digital.ai

TeamForge 18.1

TeamForge 18.1 has a lot of new features and enhancements. Here's a list of a few release-defining new features in TeamForge 18.1.

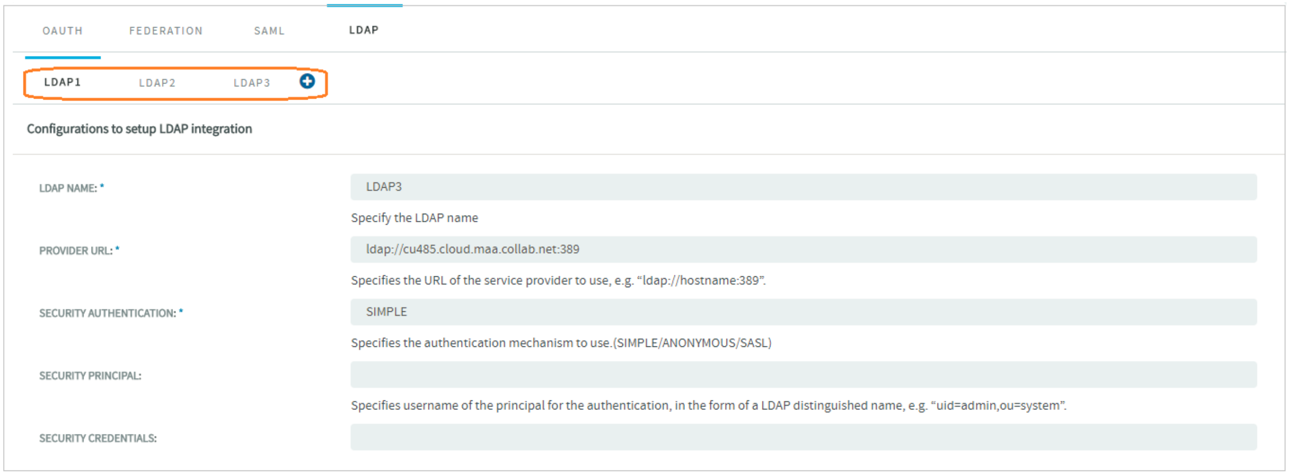
Release Information

- **Released on**: April 27, 2018
- **GA Version**: 18.1.446

Site Administration

Support for Multiple LDAP Servers

You can now configure multiple LDAP servers for LDAP authentication.



Configurations to setup LDAP integration

LDAP NAME: * LDAP3

Specify the LDAP name

PROVIDER URL: * ldap://cu485.cloud.maa.collab.net:389

Specifies the URL of the service provider to use, e.g. “ldap://hostname:389”.

SECURITY AUTHENTICATION: * SIMPLE

Specifies the authentication mechanism to use.(SIMPLE/ANONYMOUS/SASL)

SECURITY PRINCIPAL:

Specifies username of the principal for the authentication, in the form of a LDAP distinguished name, e.g. “uid=admin,ou=system”.

SECURITY CREDENTIALS:

Support for Multiple LDAP Servers

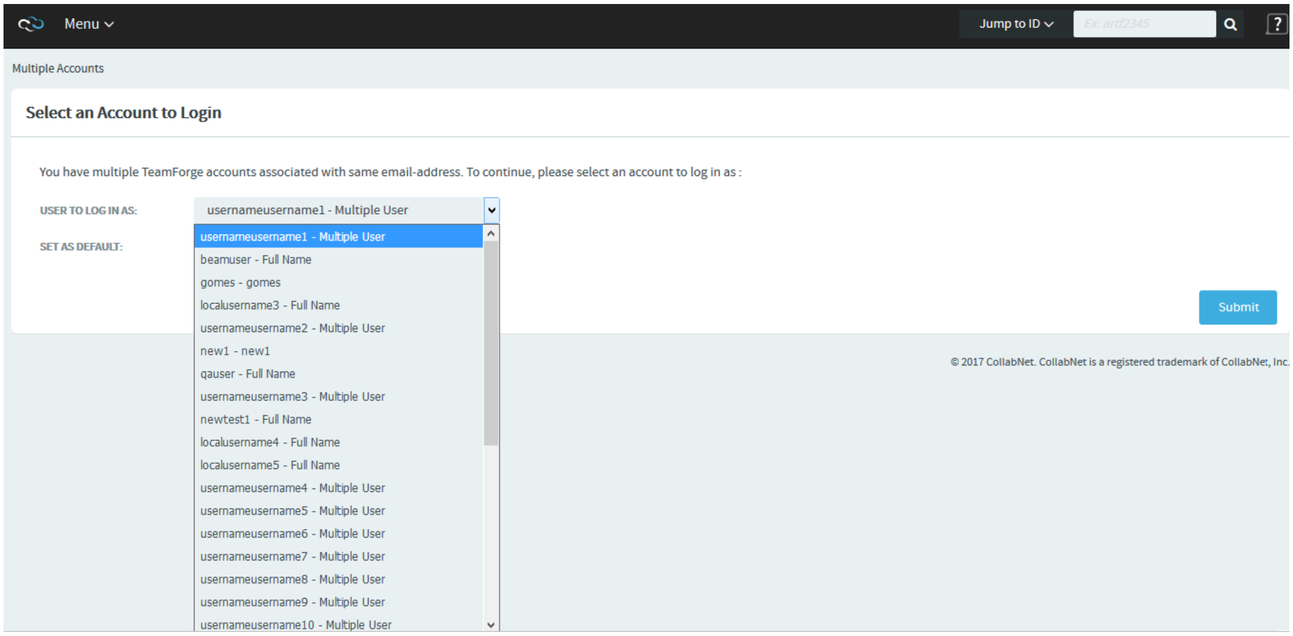
For more information, see Use LDAP for TeamForge User Authentication.

Enhancement to LDAP Configuration: Base Filter

A new search field, **BASE FILTER**, has been added to the LDAP configuration page in TeamForge to filter users belonging to a specific domain (organizational unit). For example: `_(&(sAMAccountName={0})(objectCategory=account)(objectClass=user)_`. For more information, see Use LDAP for TeamForge User Authentication.

One E-mail Address–Multiple User Accounts: Intermediate Login Page

In SAML and SAML+LDAP environments, if you have multiple user accounts mapped to one email address, you will be redirected to an intermediate login page before the third party IdP for authentication. In this intermediate login page, you can see the list of accounts associated with your email address.


Menu
Jump to ID
Ex: artf2345
Submit
usernameusername1 - Multiple User
beamuser - Full Name
gomes - gomes
localusername3 - Full Name
usernameusername2 - Multiple User
new1 - new1
qauser - Full Name
usernameusername3 - Multiple User
newtest1 - Full Name
localusername4 - Full Name
localusername5 - Full Name
usernameusername4 - Multiple User
usernameusername5 - Multiple User
usernameusername6 - Multiple User
usernameusername7 - Multiple User
usernameusername8 - Multiple User
usernameusername9 - Multiple User
usernameusername10 - Multiple User

Multiple Accounts

Select an Account to Login

You have multiple TeamForge accounts associated with same email-address. To continue, please select an account to log in as :

USER TO LOG IN AS: usernameusername1 - Multiple User

SET AS DEFAULT:

© 2017 CollabNet. CollabNet is a registered trademark of CollabNet, Inc.

Intermediate Login Page

For more information, see:

- Use SAML for TeamForge User Authentication
- Use Both SAML and LDAP for TeamForge User Authentication

Local Users

On sites with LDAP/SAML/SAML+LDAP integrations, site administrators can designate select users that do not have a SAML or LDAP account as local users. Local users can log on to TeamForge using just the TeamForge credentials, bypassing the SAML/LDAP/SAML+LDAP authentication realm.

When you select this ALLOW LOCAL USER site setting, the **Create User** and **Edit User Information** pages let site administrators to select the **Local User** check box while creating or editing user accounts.

Browser Caching in TeamForge

Unlike in the past where TeamForge data was not cached in the browser, you can now enable or disable caching in TeamForge for a better application performance. A new site-options token, `BROWSER_NO_CACHE=false`, has been added to the `runtime-options.conf` file. This means, browser caching is enabled by default. Set this token to `true` to turn off browser caching.

Project Administration

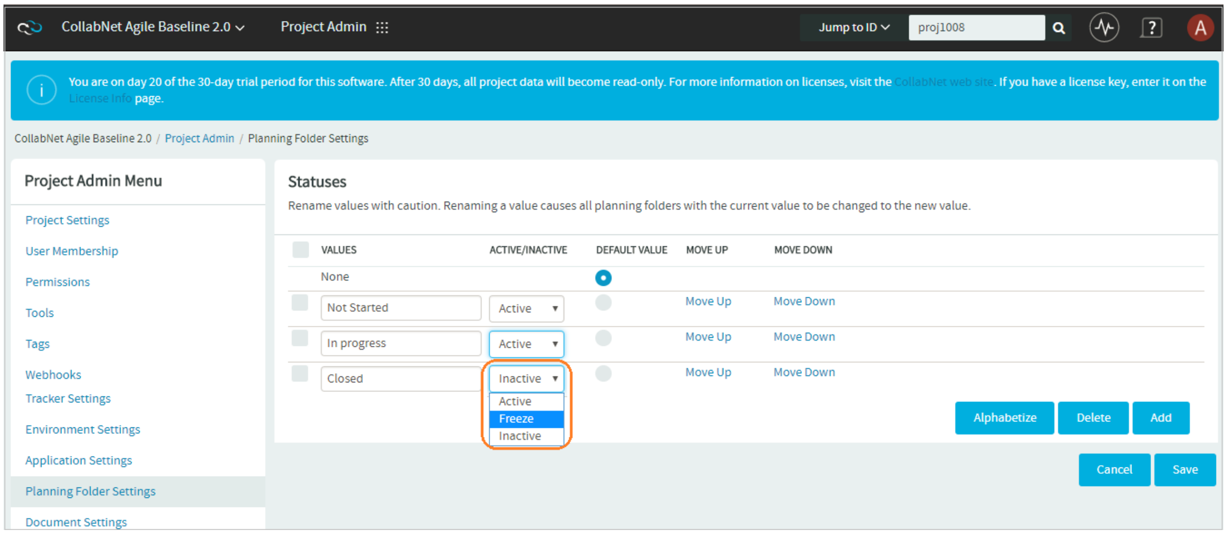
Set up Webhooks for Tracker Artifacts

You can now create webhooks to generate events whenever an artifact is created, updated, moved, cloned, and deleted. For more information, see Set up Webhooks for Tracker Artifacts.

Planning Folder Enhancements

- **A New Planning Folder Status**, **Freeze**

  You can now change the status of a planning folder to “Freeze”. A new status “Freeze” has been included in the **Active/Inactive** drop-down list in **Project Admin > Planning Folder Settings** page.



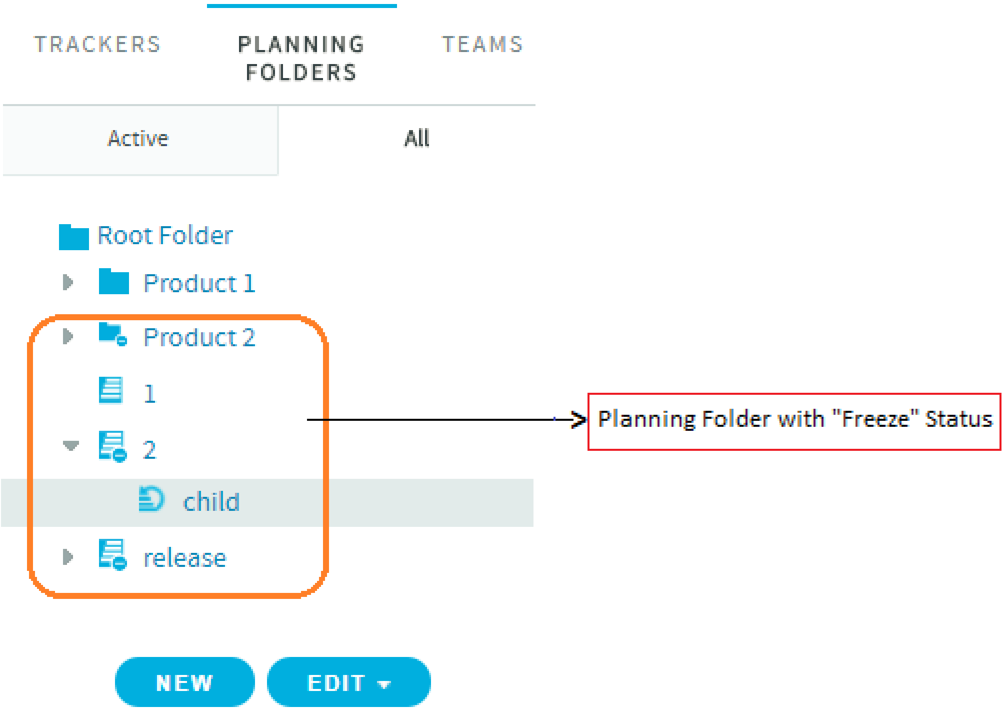
You are on day 20 of the 30-day trial period for this software. After 30 days, all project data will become read-only. For more information on licenses, visit the CollabNet web site. If you have a license key, enter it on the License Info page.

Statuses

Rename values with caution. Renaming a value causes all planning folders with the current value to be changed to the new value.

	VALUES	ACTIVE/INACTIVE	DEFAULT VALUE	MOVE UP	MOVE DOWN
	None		●		
□	Not Started	Active	○	Move Up	Move Down
□	In progress	Active	○	Move Up	Move Down
□	Closed	Inactive	○	Move Up	Move Down

Freeze, One of the Planning Folder Statuses



Planning Folders in Frozen Status

For more information, see Manage Statuses for a Planning Folder.

- **Filter Columns in Planning Folder View**

  Starting from TeamForge 18.1, the Filter Columns function is implemented for all the system-defined fields within the Planning Folder view.

Performance Improvement–Project Template Creation in the Background

Project template creation can, at times, be time consuming. Not any more! With TeamForge 18.1 and later, once you initiate the project template creation process, it is handled as a backend process and you can move on with your other activities. Once the template is created, you would get an email notification and the template would get listed in the **Project Templates** page and in the **Templates** drop-down list for creating new projects.

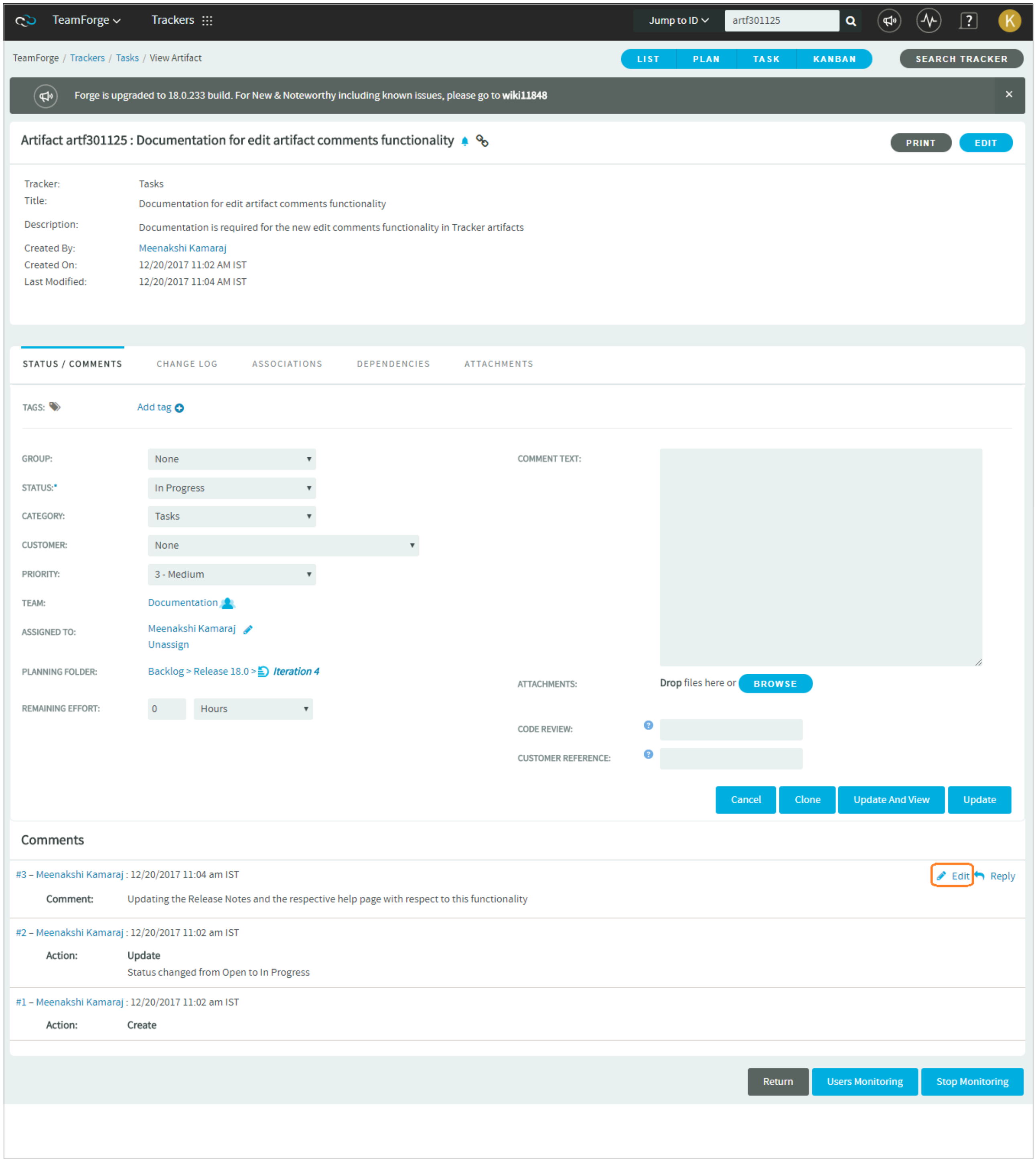
Trackers

Strip Original Email Content from Reply Email

You must be aware that whenever you reply to an email that you receive after an artifact is updated, the original email content was also carried over along with your reply as a comment to the artifact in the Web UI. Now it is just your reply that is sent without the original email content. This enhances the user experience with the UI.

Edit Comments in Tracker Artifacts

You can now edit all the comments (only the most recent comment was editable earlier) that you have added to a tracker artifact. For more information, see **Edit Your Comments** section in Edit a Tracker Artifact.



Forge is upgraded to 18.0.233 build. For New & Noteworthy including known issues, please go to **wiki11848**

Artifact artf301125 : Documentation for edit artifact comments functionality

Tracker: Tasks
Title: Documentation for edit artifact comments functionality
Description: Documentation is required for the new edit comments functionality in Tracker artifacts
Created By: Meenakshi Kamaraj
Created On: 12/20/2017 11:02 AM IST
Last Modified: 12/20/2017 11:04 AM IST

TAGS:
GROUP: None
STATUS:* In Progress
CATEGORY: Tasks
CUSTOMER: None
PRIORITY: 3 - Medium
TEAM: Documentation
ASSIGNED TO: Meenakshi Kamaraj
Unassign
PLANNING FOLDER: Backlog > Release 18.0 > *Iteration 4*
REMAINING EFFORT: 0 Hours
COMMENT TEXT:
ATTACHMENTS: **Drop** files here or
CODE REVIEW:
CUSTOMER REFERENCE:

Comments

#3 – Meenakshi Kamaraj : 12/20/2017 11:04 am IST
Comment: Updating the Release Notes and the respective help page with respect to this functionality

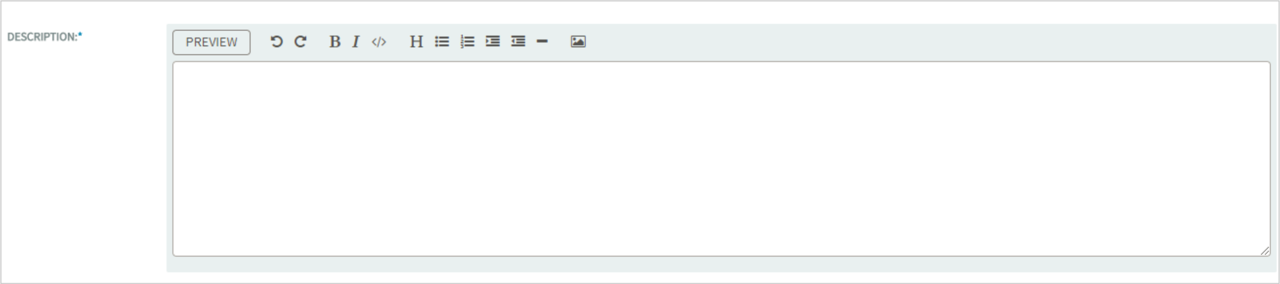
#2 – Meenakshi Kamaraj : 12/20/2017 11:02 am IST
Action: **Update**
Status changed from Open to In Progress

#1 – Meenakshi Kamaraj : 12/20/2017 11:02 am IST
Action: **Create**

Edit Button in Tracker Comments

Markdown Editor for Description and Comments of Tracker Artifacts

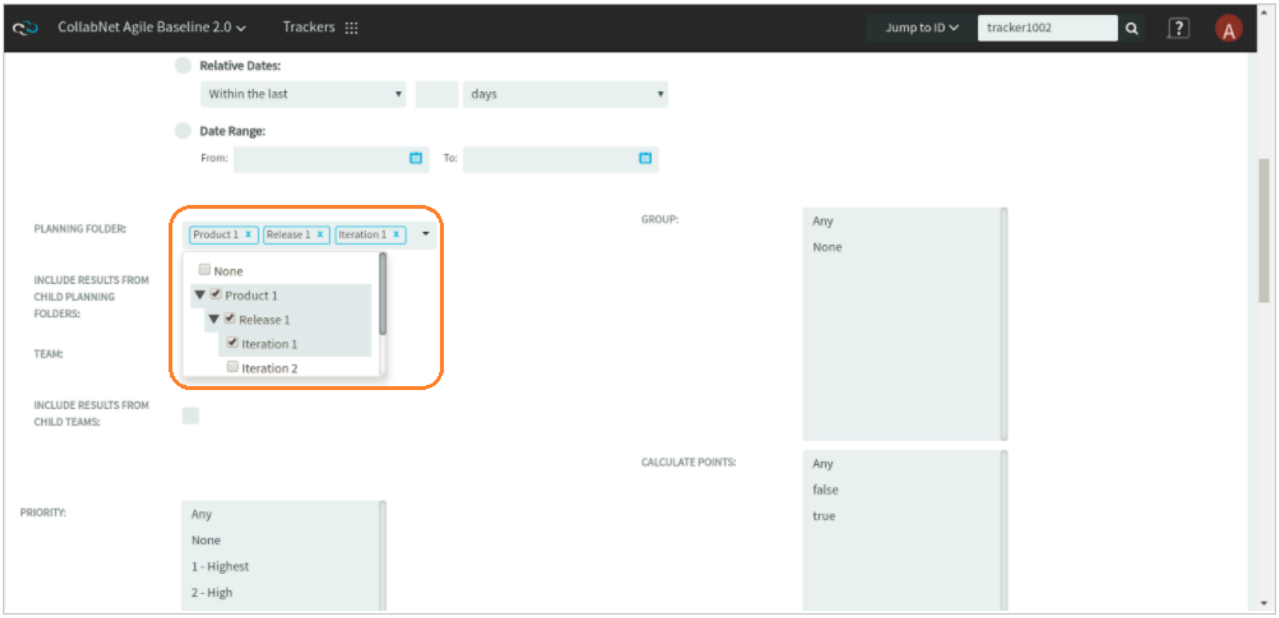
A new Markdown editor has been introduced for the **Description** and the **Comments** fields of tracker artifacts. Format your content with the wiki-like markdown syntax. TeamForge uses Showdown—a bidirectional Markdown to HTML to Markdown converter written in Javascript. For more information, see the official Showdown Documentation. Here's an abridged version of the Markdown syntax documentation.



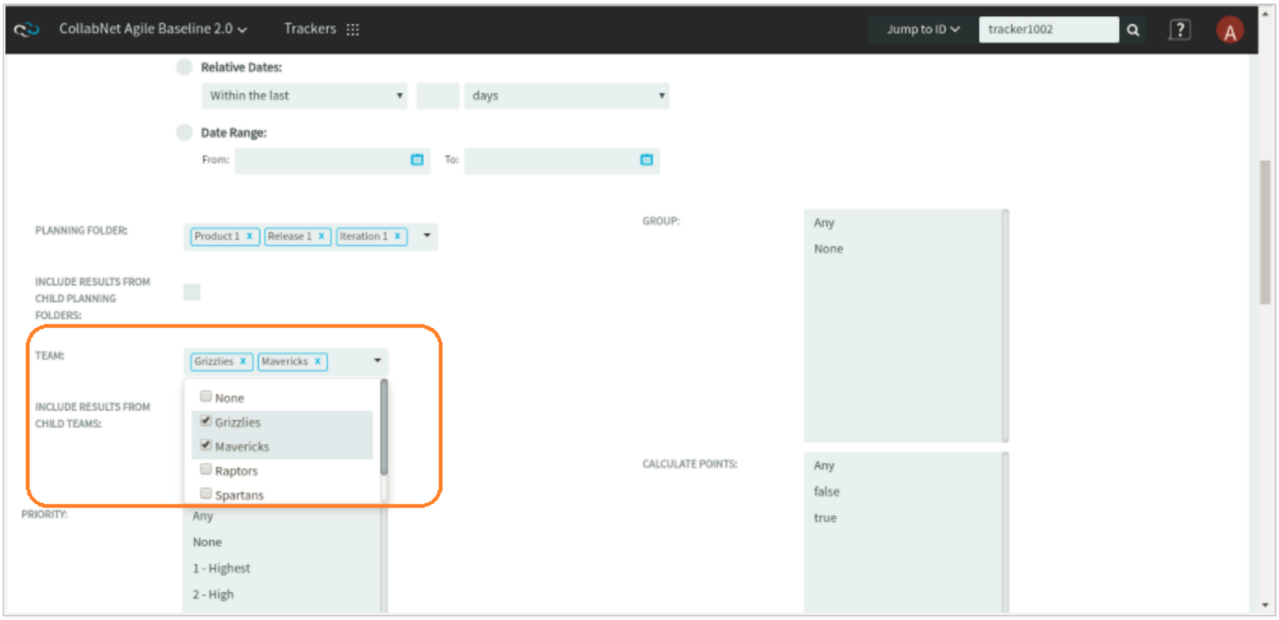
Markdown Editor

Enhancements to Search Tracker Page

When searching the tracker, you can now select one or more planning folders or teams from the respective fields on the **Search Tracker** page.



Search Tracker Page Enhancements



○ Relative Dates: Within the last ▾ days ▾

○ Date Range: From: To:

PLANNING FOLDER: Product 1 × Release 1 × Iteration 1 ×

INCLUDE RESULTS FROM CHILD PLANNING FOLDERS: □

TEAM: Grizzlies × Mavericks ×

□ None
☑ Grizzlies
☑ Mavericks
□ Raptors
□ Spartans

INCLUDE RESULTS FROM CHILD TEAMS:

PRIORITY: Any, None, 1 - Highest, 2 - High

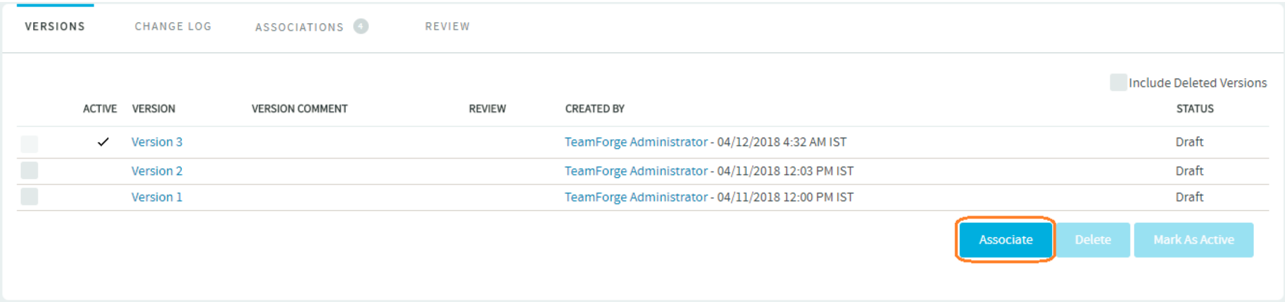
GROUP: Any, None

CALCULATE POINTS: Any, false, true

Search Tracker Page Enhancements

Documents

You can now associate TeamForge objects such as artifacts, documents, and so on with the desired version of a document. For more information, see the section Change the Active Document Version in Work with Your Documents.

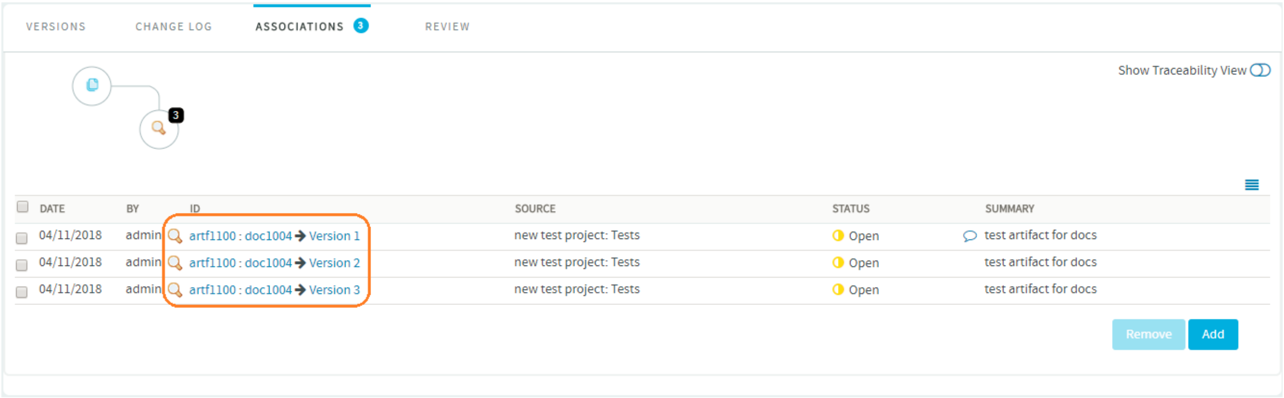


□ Include Deleted Versions

	ACTIVE	VERSION	VERSION COMMENT	REVIEW	CREATED BY	STATUS
□	✓	Version 3			TeamForge Administrator - 04/12/2018 4:32 AM IST	Draft
□		Version 2			TeamForge Administrator - 04/11/2018 12:03 PM IST	Draft
□		Version 1			TeamForge Administrator - 04/11/2018 12:00 PM IST	Draft

Associate a Document Version with TeamForge Objects

NOTE: If you haven’t selected any document version, the current active version of the document gets associated with the object.



DATE	BY	ID	SOURCE	STATUS	SUMMARY
04/11/2018	admin	artf1100 : doc1004 → Version 1	new test project: Tests	Open	test artifact for docs
04/11/2018	admin	artf1100 : doc1004 → Version 2	new test project: Tests	Open	test artifact for docs
04/11/2018	admin	artf1100 : doc1004 → Version 3	new test project: Tests	Open	test artifact for docs

Associate a Document Version with TeamForge Objects

Enhancements to HTML Notification Email Templates

Avatars are added for the **Assigned To** field in the HTML emails that you receive when a tracker artifact is created or updated and the **Created By**, **Updated By** and **Locked By** fields in the HTML emails that you receive when a document is created or updated.

Reports–Auto Healing Process for Handling Invalid XML Records

Earlier, if any corrupted XML records were found being stored during the XML to Non-XML conversion, the administrators need to manually correct them and store the corrected data into the new flex field bridge tables. To save this manual effort, an automated healing approach is now available, which replaces the invalid XML records with the correct data and makes them available for the next ETL run.

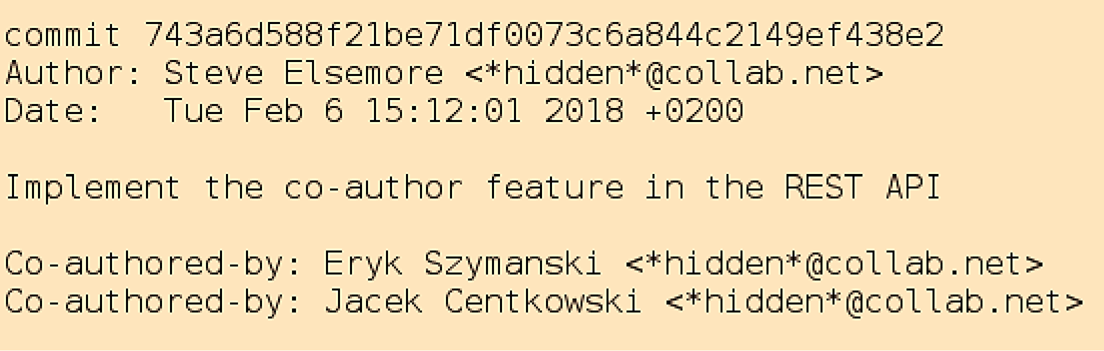
SCM/Git/Gerrit

Highlights of TeamForge-Git Integration 18.1.7-2.14.8

This section provides the highlights of TeamForge-Git Integration 18.1.7-2.14.8 released on Apr 23, 2018. For more information and bug fixes, see release notes of vanilla Gerrit version 2.14.8-1-g6ca0726997.

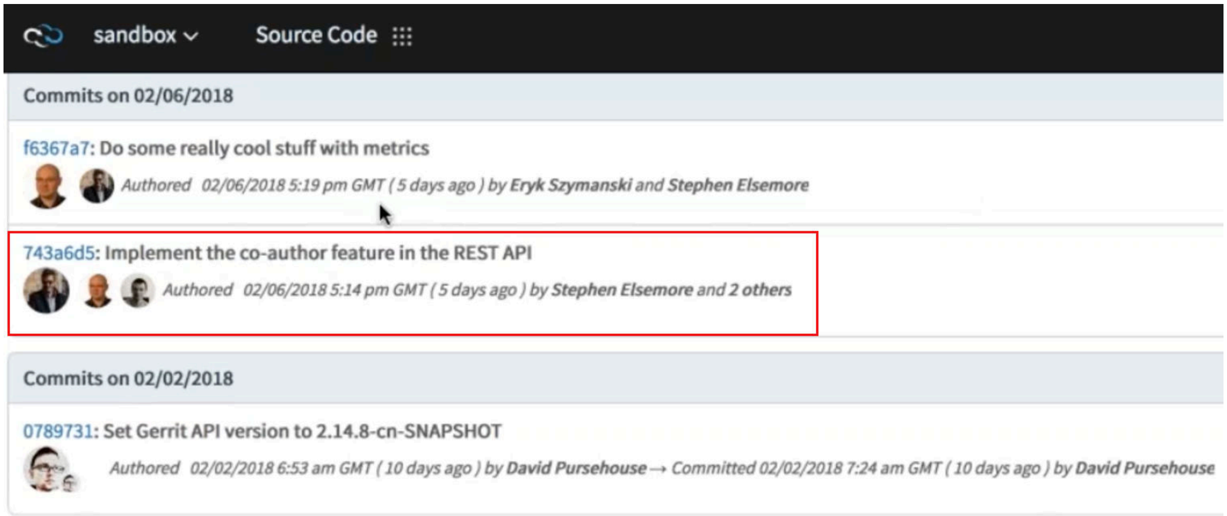
Enhancements to the Code Browser UI

- The coauthor name is also included as part of the author avatar and **Authored by** information on the changes list, whenever a change is done to the “Co-authored-by” footer text. More information can be seen in change details view.



```
commit 743a6d588f21be71df0073c6a844c2149ef438e2
Author: Steve Elsemore <*hidden*@collab.net>
Date:   Tue Feb 6 15:12:01 2018 +0200

Implement the co-author feature in the REST API

Co-authored-by: Eryk Szymanski <*hidden*@collab.net>
Co-authored-by: Jacek Centkowski <*hidden*@collab.net>
```

Add coauthors in the Commit Message



Commits on 02/06/2018

f6367a7: Do some really cool stuff with metrics
Authored 02/06/2018 5:19 pm GMT (5 days ago) by Eryk Szymanski and Stephen Elsemore

743a6d5: Implement the co-author feature in the REST API
Authored 02/06/2018 5:14 pm GMT (5 days ago) by Stephen Elsemore and 2 others

Commits on 02/02/2018

0789731: Set Gerrit API version to 2.14.8-cn-SNAPSHOT
Authored 02/02/2018 6:53 am GMT (10 days ago) by David Pursehouse → Committed 02/02/2018 7:24 am GMT (10 days ago) by David Pursehouse

List of coauthors

- The @mentions logic used in the review comments and description fields has been improved to include users that are added as reviewers already, which earlier was not the case.

- Whenever a review is opened, the recommended reviewers list is shown immediately when the **Add a New Reviewer** field gets the focus.

- When a pull request merge was done on different commits of the same submodule on different branches, the merge contained the commits of an older version of the submodule.

- Better handling of customer repository review rules.

- Performance has been improved when viewing reviews and tags on repositories having large number of tags.

- Deleted or rewritten branches are sorted in descending order to bring the most recent branches to the top of the list.

Installer Enhancements

- A new token GERRIT_USER_EMAIL is added to the `site-options.conf` file to set the email account for sending Gerrit emails. This refers to all Gerrit servers specified in `site-option.conf` file or through cluster/server specific parameters. For example, the “clusterId/serverId” in `[clusterId/serverId]:gerrit:user.email` refers to the cluster or server that is used.

- HTTP requests in JavaMelody statistics are grouped to identify issues quickly.

- Supports Java 9.

Bug Fixes in TeamForge-Git Integration 18.1.7-2.14.8

- Fixed the security vulnerability found during the URL redirection to prevent phishing.

- Associations were not created when HTTP connection to EventQ was not established due a Java HTTP client issue. This is fixed.

- An error was thrown, if the destination branches in the Code Browser UI had the URL encoded character “+” in their pull requests. This is resolved.

- Fixed the issue in which the **Delete Permanently** button was not shown on the **Deleted/Rewritten** branches tab in the Code Browser UI for non-site administrators.

- A browsing error was thrown in the Code Browser UI for the file that was tagged incorrectly as LFS file. This is fixed.

- The **maxRetries** configuration parameter was ignored by TeamForge-Git integration resulting in indefinite updates to disconnected remote replica servers. This issue is fixed.

- Gerrit’s internal log rotation and compression feature is disabled as it is handled automatically by the TeamForge runtime environment.

- During an upgrade, TeamForge provision failed due to the issue found in the Gerrit database backup directory creation for subsequent provisions. This is fixed.

- An exception was thrown when a SSH key that is not supported by the `ssh-keygen` command in CentOS 6 was generated during the SSH key generation process. This is fixed.

Install/Upgrade

- The following are the latest software versions supported by TeamForge.
    - Java 9

      TeamForge 18.1 (and later) supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens before upgrading to TeamForge 18.1.

        - JBOSS_JAVA_OPTS
        - PHOENIX_JAVA_OPTS
        - INTEGRATION_JAVA_OPTS
        - ETL_JAVA_OPTS

      TeamForge provision fails on sites that use these options post upgrade to TeamForge 18.1.

    - Wildfly 11
    - Tomcat 8.0.50
- The following Review Board scripts have been moved to `/opt/collabnet/teamforge/runtime/scripts/` directory:
    - backup-rb-data.py
    - restore-rb-data.py
    - domain_change_rb.py
    - reviewboard-psql-wrapper
    - svn-auth.py
    - svn_cache.sh
- Review Board installation and upgrade is no longer a separate process. Review Board is installed automatically when you install/upgrade TeamForge. In other words, you do not have to do a `yum install CN-reviewboard` separately.

  > **NOTE:** TeamForge 18.1 has no support for having service-specific FQDN for Review Board.

- Tasks component is no longer supported from TeamForge 18.1 and later. For more information, see CollabNet's Product End-of-Life Policy.

Back up and Restore TeamForge Data Using the teamforge.py Script

You can now use the `teamforge.py` scripts's `backup` and `restore` commands to back up and restore TeamForge data. For more information, see Back up and Restore TeamForge Data Using the teamforge.py Script.

DLM End of Life

TeamForge 18.1 and later versions no longer support the DevOps Lifecycle Management (DLM) product. As a result, an error occurs when projects that include Applications and Environments are imported via Simbel. To work around this, delete the Applications and Environments via REST APIs (the **Delete** button in the user interface (**Project Home > Project Admin > Application Settings**) is disabled in TeamForge 18.1 and later) and then try importing the project again.

Here's a list of few noteworthy issues fixed in TeamForge 18.1.

- Artifact comments that are two or more lines long were not wrapped properly; instead, a horizontal scrollbar was added. This is fixed.
- Fixed the TeamForge system error that occured when you clone an artifact (that has no attachments) with the **Include Attachment(s)** check box selected.
- Fixed an issue that prevented wiki pages with `JPEG/JPG` or `PNG` images from being exported as PDF.
- Fixed an issue with Oracle CLOB data handling that prevented the TrackerIncrementalJob from running post TeamForge upgrade.
- Fixed an issue with the parent-child field value mapping (using which a child field's value is auto-selected based on the parent field's value) when a pair of parent-child fields are made mandatory by a tracker workflow.
- An issue that prevented sorting the list by the **Apply Date** column on the **Users Awaiting Approval** page has been fixed.
- When you clone an artifact with the **Sum effort from children** field selected, the newly created artifact inherits the effort value from the artifact that it was cloned from. This is fixed.
- Fixed an issue that resets the user-selected parent field's value to `None` when the user tries to save the artifact without filling in one or more mandatory fields.

- Fixed an issue with the artifact dependency table, which was not updated when you change the parent of a child artifact.
- Document icons were not bound with the documents when you sort documents by document id or document name, which is now fixed.
- Using certain keywords such as tracker, docman, planning, frs, taskmgr, scm, wiki, and discussion in planning folder names (for example, “Metric tracker”) caused an error. In addition, the planning folder ID was not shown in the **Jump to ID** field. This is fixed.
- Changes made to the `subject` element of the HTML artifact update email notification template (`/branding/templates/mail/artifact_monitoring_template_html.vm`) were ignored. This is fixed.
- A role with `Tracker View` permission was working inconsistently. Such a role, when assigned to an individual user and a user group (which the user is a member of), lets the user view the list of teams in a project and prevents the user from viewing the list of teams in a project respectively. This is fixed.
- A user that is monitoring applications such as (Trackers, Documents, Source Code, Discussions, File Releases and Wiki) continues to monitor objects even after the user was selectively removed from monitoring one of the objects such as an artifact or a document. This is fixed.
- When a document is added to or deleted from a restricted folder, notification emails were sent to users that do not have view permission to the restricted document folder, which is now fixed.
- When you open a document (to view its details) from a multi-page document list and when you click **Return** on the **Document Details** page, you are taken to the first page of the documents list instead of the document list page from where you opened the document earlier. This is fixed.
- Creating global project roles and assigning permission to these roles had a few performance issues, which are now fixed.
- Removed the legacy CSRF token from TeamForge.
- Fixed an issue that allowed users to upload restricted file types on the Project Home page.
- Fixed the `403 (Forbidden)` error that occured when a user tries to log on to TeamForge with valid credentials immediately after a failed login attempt (due to incorrect password).
- When user accounts are created by the TeamForge site administrator, users may get an email with a link to set their password. The link, by default, expires in 72 hours. Using an expired link was taking the

user to a blank page instead of letting the user use the **Forgot Password** functionality, which is now fixed.

- Filtering columns with keywords containing leading and/or trailing spaces showed no results, which is now fixed.

- Custom web page (configured using the Publishing repository) was not shown even if the *Show custom web page* option (under **Project Admin > Project Settings**) was selected for the project. Instead, the default project home page was shown. This is now fixed.

- If you were using service-specific FQDN for Subversion, post installation of TeamForge, you had to delete the `/opt/collabnet/teamforge/var/home/apache/.subversion` directory and login once using the Subversion-specific FQDN to prevent authentication requests from Review Board from failing. This is fixed.

The following noteworthy issues, including any workarounds we may have, are known to exist in the TeamForge 18.1 release.

- CLI reports are not showing up on sites with Oracle database.
- There is an issue with ranking artifacts in planning folders. Ranking order is lost when you refresh the page. In addition, an unexpected error occurs when you rank artifacts with parent-child relationship.
- When you run a saved tracker search and export the results to Excel, the export fails if the result consists of more than 250 artifacts.
- Commits made to CVS repositories are not showing up in the ViewVC page.
- In a SAML-enabled environment, if you log out while a tracker is being exported and re-login, you would see a blank page without the exported data.

Plan your installation or upgrade setup, hardware and software requirements and so on before you begin.

TeamForge Services

Before you plan your installation or upgrade, let us understand TeamForge and its services.

A TeamForge site consists of a core TeamForge application and several tightly integrated services that support it. In addition, you can integrate TeamForge with other third party applications such as Nexus, Jenkins, Jira and so on. Some of the TeamForge services are mandatory and some are optional. You can install the services, all in one single server, or distribute them across two or more servers.

- The core TeamForge application provides the Web interface that users see, and the API that other applications can interact with. It also includes the file system where some user content is stored, such as wiki pages.
- The site database is where most of the user-created content is stored and accessed. Documents, discussion posts, tracker artifacts, project administration settings: all that sort of thing lives in the database.
- The source control server ties any number of Subversion, Git/Gerrit or CVS repositories into the TeamForge site.
- The Extract Transform and Load (ETL) server pulls data from the site database and populates the datamart to generate charts and graphs about how people are using the site. The datamart (Reports DB) is an abstraction of the site database, optimized to support the reporting functionality.
- EventQ is a TeamForge capability that provides traceability for product life cycle activities such as work items, SCM commits, continuous integration (CI) builds, and code reviews.

Here’s a list of available TeamForge services.

Service	Mandatory/ Optional	Old Name	Description
ctfcore	Mandatory	app	Main TeamForge application server
search	Mandatory	indexer	Indexing and searching
mail	Mandatory	NA (added in TeamForge 17.1)	Email server
ctfcore-database	Mandatory	database	Operational database
ctfcore-database-mirror	Optional	NA	Mirror of operational database
codesearch	Mandatory	codesearch	Code Search
etl	Optional	etl	ETL for Datamart
ctfcore-datamart	Mandatory if and only if you install `etl`	datamart	Datamart database
cliserver	Mandatory	NA	CLI Server

Service	Mandatory/ Optional	Old Name	Description
subversion	Optional	subversion	SVN Version Control
cvs	Optional	cvs	CVS Version Control
gerrit	Optional	gerrit	Git/Gerrit Version Control
gerrit-database	Mandatory if and only if you install `gerrit`	NA (added in TeamForge 17.1)	Database for Git/Gerrit. In a distributed setup, add this identifier to the server where you want to run Gerrit database. In a distributed setup with multiple Git integration servers, add this identifier to all the servers that run the Git databases. For more information, see host:SERVICES token.
binary	Optional	Optional	Artifact repository integration
binary-database	Mandatory if and only if you install `binary`	binary	Database for artifact repository integration. Binary app (binary) and database (binary-database) have to be installed on the same server.
reviewboard	Optional	reviewboard	Review Board code review tool
reviewboard-database	Mandatory if and only if you install `reviewboard`	NA (added in TeamForge 17.1)	Database for Review Board. In a distributed setup, add this identifier to the server where you want to run Review Board database.
reviewboard-adapter	Mandatory if and only if you install reviewboard	NA	Adapter for reviewboard to copy `ctfrbevents.jar`. In a distributed setup, `reviewboard-adapter` must always be installed on the TeamForge Application Server.
eventq	Mandatory	NA (added in TeamForge 17.4)	EventQ application server. In a distributed setup, add this identifier to the server where you want to run EventQ application.
redis	Mandatory	NA (added in TeamForge 17.4)	EventQ in-memory database/data structure server. In a distributed setup, add this identifier to the server where you want to run EventQ application.
mongodb	Mandatory	NA (added in TeamForge 17.4)	EventQ database server. In a distributed setup, add this identifier to the server where you want to run EventQ database.
rabbitmq	Mandatory	NA (added in TeamForge 17.4)	EventQ AMQP message server. In a distributed setup, add this identifier to the server where you want to run EventQ message queue.

These service identifiers are used in the `site-options.conf` file's `host:SERVICES` token. For more information, see host:SERVICES token.

In addition, installing TeamForge with service-specific FQDNs (instead of machine-specific host/domain names) is highly recommended so that you will be able to change the system landscape at a later point in time without having any impact on the URLs (in other words, end users do not have to notice or change anything). For example, you can create FQDNs specifically for services such as Subversion, Git, mail, Codesearch and so on. For more information, see Service-specific FQDNs.

NOTE: You cannot have a separate PUBLIC_FQDN for EventQ.

Single Server or Distributed Setup?

If you are installing TeamForge, are you planning to install on a single server or distribute TeamForge services across two or more servers? How are you going to distribute the services?

In the default setup, all services run on the same server as the main TeamForge application. But in practice, only the TeamForge application needs to run on the TeamForge application server. The other services can share that server or run on other servers, in almost any combination.

Assess your own site's particular use patterns and resources to decide how to distribute your services, if at all. For example, if you anticipate heavy use of your site, you will want to consider running the site database, the source control service, or the reporting engine on separate hardware to help balance the load. For examples on how to distribute TeamForge services, see host:SERVICES token.

In a distributed setup, it is highly recommended to have dedicated servers for TeamForge database and SCM services as these are the most sought after services in TeamForge.

When you distribute your services on multiple servers, you must do some configuration to handle communication among the services. Verify your basic networking setup. See Set Up Networking for TeamForge.

PostgreSQL or Oracle?

PostgreSQL is installed automatically when you install TeamForge. If you intend to use Oracle, CollabNet recommends that you let the installer run its course, make sure things work normally, and then set up your Oracle database and switch over to it.

PostgreSQL 9.6.5 is installed automatically when you install TeamForge 18.1. If you intend to use Oracle, CollabNet recommends that you let the installer run its course, make sure things work normally, and then set up your Oracle database and switch over to it.

If you want to use Oracle as your database, consider the following points:

- TeamForge 18.1 supports Oracle server 12c and Oracle client 12c.
- Oracle express edition is not supported for both client and server.
- Review Board 2.5.6.1 was tested with PostgreSQL 9.6.5 only. Review Board with Oracle was not tested.
- Git integration works only with PostgreSQL. The Git integration uses PostgreSQL even if your TeamForge site uses Oracle.

The efficiency of your database can have an impact on your users' perception of the site's usability. If your site uses a PostgreSQL database (which is the default), you may want to consider tuning it to fit your specific circumstances. The default settings are intended for a small-to-medium site running on a single server. See What are the right PostgreSQL settings for my site? for recommendations from CollabNet’s performance team on optimizing PostgreSQL for different conditions.

Integrations

TeamForge supports integration with a wide array of third party applications such as Nexus, Jira and so on. As a customer, you may or may not always want (or have) all of TeamForge’s supported integrated applications. It’s also quite possible that some of the integrated applications may not always run on all the platforms supported by TeamForge. To accommodate a wider audience, by default, TeamForge install and upgrade instructions include steps to integrate such third party applications with TeamForge.

However, use your discretion to ignore and skip such steps if they are not relevant to your site. See TeamForge Installation Requirements to understand what it takes to run TeamForge 18.1 with integrations.

One-hop Upgrade Compatibility

Though the TeamForge 18.1 installer supports one-hop upgrade from TeamForge 16.3 or later versions, TeamForge 18.1 upgrade instructions, in general, are for upgrading from TeamForge 17.11 (including update releases, if any) to TeamForge 18.1.

During an upgrade, the TeamForge 18.1 migration script detects the TeamForge version you run, checks if it’s TeamForge 16.3 or later, and if `true`, proceeds with the data migration. The migration script aborts with an error message if it detects TeamForge versions older than TeamForge 16.3. You must upgrade your site to TeamForge 16.3 or later and then upgrade to TeamForge 18.1.

Hardware and Backup

If you aren’t the person who first installed your current TeamForge site (or maybe, even if you are), it’s essential to catalog the hosts where your services are running and to know what configuration has been applied to them.

While upgrading to a latest TeamForge version, you can shoose to upgrade on the same hardware or on new hardware. In general, it is good to have a backup plan in place. Same hardware upgrades need no backup. However, it’s recommended to take a back up as a measure of caution. See Back up and Restore TeamForge for more information.

Other Dos and Don’ts

Here’s a list of dos, don’ts and points to remember when you install or upgrade TeamForge. Show/Hide

Dos

- Understand TeamForge installation requirements and plan your installation or upgrade.
- Get your TeamForge license key and keep it handy.
- Verify your basic networking setup before installing or upgrading TeamForge. See Set Up Networking for TeamForge.
- Look for new or modified `site-options.conf` tokens and update your `site-options.conf` file as required during the upgrade process. See Site Options Change Log.
- Set up a TeamForge Stage Server before you upgrade your Production Server.
- Stop TeamForge services on all servers in a distributed setup while upgrading to TeamForge 18.1.
- Uninstall hot fixes and add-ons, if any, before you start the TeamForge 18.1 upgrade procedure.

- TeamForge 18.1 supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.

  - JBOSS_JAVA_OPTS
  - PHOENIX_JAVA_OPTS
  - INTEGRATION_JAVA_OPTS
  - ETL_JAVA_OPTS
  - ELASTICSEARCH_JAVA_OPTS

  TeamForge provision fails on sites that use these options post TeamForge 18.1 upgrade.

Don'ts

- Do not customize your operating system installation. Select only the default packages list.
- While upgrading TeamForge, whether in place or on new hardware, always reuse the old `site-options.conf` file and make changes as necessary. **Do not** try to start with a new `site-options.conf` file. Reusing the old `site-options.conf` avoids many potential problems, particularly around the management of usernames and passwords.
- Do not manually modify TeamForge-managed site option tokens such as the `AUTO_DATA` token. See AUTO_DATA for more information.

- If you are creating symlinks, note that you must create symlinks only to the TeamForge data directory (`/opt/collabnet/teamforge/var`). You should not create symlinks to TeamForge application directories (such as `/opt/collabnet`).

Points to Remember

- Installing or upgrading TeamForge needs root privileges. You must log on as root or use a root shell to install or upgrade TeamForge.
- SSL is enabled by default and a self-signed certificate is auto-generated. However, you can use a few `site-options.conf` tokens to adjust this behavior. To generate the SSL certificates, see Generate SSL Certificates.
- For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.
- If you have Git integration on a separate server, both TeamForge and Git servers must have their time and date synchronized.
- While you can run both EventQ and TeamForge on the same server, CollabNet recommends such an approach only for testing purposes. It's always recommended to run EventQ on a separate server for optimal scalability.
- No backup is required for same hardware upgrades. However, you can create a backup as a measure of caution. See Back up and Restore TeamForge for more information.
- Always use compatible JDBC drivers meant for specific database versions. See JDBC Drivers Reference for more information. Also see: Why do ETL jobs fail post TeamForge upgrade?.
- You can run the initial load job any time after the installation of TeamForge. We recommend that you run it before you hand over the site to the users. For more information, see ETL Initial Load Jobs.
- SOAP50 APIs and event handlers are no longer supported in TeamForge 16.10 and later. Use the latest TeamForge SOAP/REST APIs.
- TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.
- Installing TeamForge with service-specific FQDNs (instead of machine-specific host/domain names) is highly recommended so that you will be able to change the system landscape at a later point in time without having any impact on the URLs (in other words, end users do not have to notice or change anything). For example, you can create FQDNs specifically for services such as Subversion, Git, mail, Codesearch and so on. For more information, see Service-specific FQDNs.
- All such service-specific FQDNs must belong to a single sub domain and it is recommended to create a new sub domain for TeamForge.
- If you are using service-specific FQDNs
  - A wildcard SSL cert is required. SNI SSL cert cannot be used.
  - When SSL is enabled and no custom SSL certificates are provided, a self-signed wildcard cert is generated for the sub domain.
  - When SSL is enabled and a custom SSL certificate is provided, the CN of the certificate is verified to be a wildcard CN.
- You cannot have a separate PUBLIC_FQDN for EventQ.

- The ability to run separate PostgreSQL instances for TeamForge database and datamart on the same server is being deprecated in TeamForge 17.11. If you have TeamForge database and datamart on separate PostgreSQL instances on the same server and if you are upgrading on a new hardware, you must Create a Single Cluster for Both Database and Datamart while upgrading to TeamForge 17.11 or later.
- While upgrading TeamForge-Git integration servers, it is important that Git master and slave servers are upgraded to the same version of TeamForge-Git integration. On sites with Git Replica Servers, you must upgrade the Git Replica Servers first and then upgrade the master Git servers.

Here's what it takes to install and run TeamForge, EventQ and other integrations supported by TeamForge.

TeamForge Hardware Requirements

The following table lists the CPU, RAM and JVM Heap Size recommendations for Small, Medium, Large and Extra-large sites.

	Small	Medium	Large	X-Large
Users	100	500	1000-5000	10000+
CPU	Octa-core	12-core	> 12-core	> 16-core
RAM	16GB	24GB	32GB	32GB
Jboss JVM Heap Size	1.5GB	3GB	6GB	> 8GB
Elasticsearch JVM Heap Size	2GB	2GB	2GB	2GB
	You must have adequate RAM to accomodate the JVM heap requirements of Elasticsearch in addition to the JVM heap requirements of other components such as Jboss, integrated applications, and so on.			
200GB (or more) hard drive. The required hard drive capacity depends on the estimated amount of document and file release uploads.				

The following table highlights the factors that can impact TeamForge performance. Numbers are indicative. Anything more than the prescribed numbers may impact the performance.

	Small	Medium	Large	X-Large
Artifacts	15000	70000	100000	100000
Flex Fields	25	50	100	100
Projects	20	80	500	500
Integrations	0-1	0-2	0-2	0-2
Integrated Applications	0-2	0-3	3+	3+

IMPORTANT: On Medium, Large, and X-Large sites, it is highly recommended that you install the TeamForge Application, Database, and SCM services on separate 64-bit servers based on the usage pattern.

Gerrit Hardware Requirements

The following table lists the CPU, RAM and Gerrit JVM Heap Size recommendations for Small, Medium, and Large sites.

For X-Large setup, see Gerrit Performance Cheat Sheet.

	Small	Medium	Large
Fetch requests per day	100k	500k	1 Million
CPU	4-core	16-core	32-core
RAM	8GB	16GB	32GB
Gerrit JVM Heap Size	4GB	12GB	28GB
200GB (or more) hard drive. The required hard drive capacity depends on the total size of all repositories.			

IMPORTANT: These numbers are indicative. Adjust your hardware based on your Gerrit server's usage. To better understand Gerrit hardware requirements and performance tuning possibilities, see Gerrit Performance Cheat Sheet.

EventQ Hardware Requirements

IMPORTANT: EventQ services can be collocated with TeamForge on the same physical server. If you do so, compare the hardware requirements for both TeamForge and EventQ and settle for the best possible hardware resources for your setup. It's always recommended to install EventQ on its own server for optimal scalability and performance.

Single Host Installation Setup

This section outlines the hardware requirements for installing TeamForge EventQ services on a single server. The single-host installation places all three TeamForge EventQ services (`App Server`, `MQ Server`, and `DB Server`) on a single operating system. This configuration is designed for trials and moderate load, while the multi-host installation setup is provided as a first step toward scaling your TeamForge EventQ instance.

Prepare for installation by setting up:

- Host running RHEL/CentOS 6.9 or 7.4
- User credentials with sudo privileges on the host

Description	Services	CPU	RAM	Storage
Single host setup with all EventQ services on the same server	App, DB and MQ servers	Quad-core 2.5 GHz	8 GB	74 GB

Multi-host Installation Setup

For high load instances or when performance is critical, install TeamForge EventQ services on separate servers or virtual machines.

Multi-host installation originates on a single server and installs core services and the TeamForge EventQ application on that first server (the `App Server`). Then, the installer remotely installs the database service on a second server (the `DB Server`) and the message queue service on a third server (the `MQ Server`).

Prepare for a multi-host installation by setting up:

- Empty hosts running RHEL or CentOS 6.9 or 7.4
- User credentials with sudo privileges on all three hosts
- ssh routes from the App server host to the other two hosts

The following are minimum resource requirements:

Description	Service	CPU	RAM	Storage	Bandwidth
EventQ App Server	App	Quad-core 2.5 GHz	4 GB	16 GB	1 Gbps to DB Server
MongoDB DB Server	DB	Quad-core 2.5 GHz	4 GB	50 GB	1 Gbps to App Server
AMQP Message Queue Server	MQ	Quad-core 2.5 GHz	4 GB	16 GB	NA

TeamForge Software Requirements

- Redhat Enterprise Linux/CentOS 6.9 and 7.4
  - Do not customize your operating system installation. Select only the default packages list.
  - Red Hat Enterprise Linux servers must have access to the Red Hat Network or equivalent (satellite server, spacewalk, or RHN proxy).
- Google Chrome 59
- Mozilla Firefox 54
- Microsoft Internet Explorer 11

TeamForge user interfaces are best viewed at screen resolution of at least 1280 x 800 (or more) pixels.

- PostgreSQL 9.6.5
- Oracle Server 12c
- Oracle Client 12c

  PostgreSQL 9.6.5 is installed by default when you install TeamForge 18.1. However, you can use Oracle if you want to. See PostgreSQL or Oracle? for more information.

- Review Board 2.5.6.1
- Git/Gerrit 2.14.x
- Subversion 1.8.19

EventQ Software Requirements

- SSH and SFTP clients are required, i.e. openssh-client. To test for this dependency, on the target server issue the command: `sftp localhost`.
- `createrepo` package is required by the disconnected mode installation. To test for this dependency, on the target server issue the command: `createrepo --version`.
- EventQ has a number of other open source software (such as Nginx, Redis and so on) dependencies, all of which are installed as part of the installation process. The following software services and their dependencies are installed:

Nginx	Installed on EventQ App Server
Phusion Passenger	Installed on EventQ App Server
Redis	Installed on EventQ App Server
MongoDB	Installed on EventQ DB Server
RabbitMQ	Installed on EventQ DB Server

IMPORTANT: In case you have an already existing instance of MongoDB and RabbitMQ, you may choose to use it instead of installing these services again. Make sure you have the required versions, though.

Supported Integrations

TeamForge 18.1 supports the following integrations:

- SubversionEdge 5.0
- CVS 1.11.x
- ViewVC 1.1.24
- Nexus 2.9.0-2.14.5[1]
- Artifactory Pro 4.7[1]
- Jira 6.3-7.4[2]
- TestLink 1.9.15 and 1.9.16
- Crucible 3.0.1
- Jenkins 1.645-2.135

Port Requirements

TeamForge Port Requirements

TeamForge components listen on a number of operating system ports. However, only a small subset must be exposed externally to enable users to access TeamForge services. Any port that is not absolutely needed must be closed.

You can select your open ports in one of these ways:

- Use the firewall configuration GUI tool that comes with your operating system. It's usually launched with a command like `system-config-selinux`.
- Open the `/etc/sysconfig/iptables` file and manually specify your open ports.

Ports Open to the Internet

Open the following operating system level ports. All other ports must be firewalled off to maintain security.

> **IMPORTANT:** Do not open port 7080 or port 8080 to the Internet. These ports are only for communications between the TeamForge application and the source code integration service, when those two site components are running on separate boxes.

Port	Description
22 (SSH)	Port 22 is the default port for the secure shell (SSH). This is required for basic SSH administrative functionality and for CVS, as all CVS transactions occur over SSH. If all Teamforge repositories are in SVN (the default for Teamforge), then this port should be closed to

Port	Description
	the public and only accessible to the system administrators. If you have to expose SSH to the Internet, the best way to protect it is to require SSH keys and not allow password authentication, and do not permit root logins over SSH. If you must use local authentication for SSH, enforce regular password changes and password complexity. **NOTE:** If you have to expose SSH internally, limit access to the port to a bastion host if you can; otherwise limit it to specific trusted hosts or subnets.
25 (SMTP)	Port 25 is the default port for SMTP (email). TeamForge discussion forums include mailing list functionality that allows users to send email to the TeamForge server. The James mail server included with TeamForge listens on port 25 to accept this mail for processing.
80 (http)	Port 80 is the default port for Web data transfer. We strongly recommend that you set up SSL and use port 80 only to redirect to port 443.
443 (https)	Port 443 is the default port for encrypted Web data transfer (https). The Apache web server should be configured to encrypt all data so that it cannot be compromised by a third party with malicious intent. Apache can be configured to force all traffic to be sent over https, even when a request is sent via port 80 (http). TeamForge can help you take care of this, if you tell it to. See Set up SSL for your TeamForge site for details.
29418 (Gerrit SSH)	Port 29418 is the default port which should be open for Gerrit SSH.

Ports for Internal Use Only

Open the `REPORTS_DATABASE_PORT` if you are granting direct access to the datamart from specific IPs using the `REPORTS_DB_ACCESS_HOSTS` `site-options.conf` token.

> **WARNING:** The ability to run separate PostgreSQL instances for TeamForge database and datamart on the same server has been deprecated in TeamForge 17.11. For more information, see Create a Single Cluster for Both Database and Datamart.

Ports to be Open in a Firewall Environment for TeamForge 18.1

	Source Server	Target Server	Port	Remarks
Apache	All	TeamForge App	80 or 443	443 for SSL
TeamForge Database	TeamForge App	TeamForge Database	5432	

	Source Server	Target Server	Port	Remarks
SVN Integration	All	SVN	80 or 443	443 for SSL
Git Integration	All	Git	80 or 443	443 for SSL
Git SSH	All	Git	29418	
Search	TeamForge App	Search	2099	
Binaries	TeamForge App	Binaries	8500	
Reports DB	TeamForge App	Reports DB	5432 or 5632	5432 is used by default as Reports DB is co-hosted with TeamForge database. 5632 can be used if you want Reports DB on a separate port.
Reports ETL	TeamForge App	Reports ETL	7010	
Code Search (Elasticsearch)	All	Code Search (Elasticsearch)	9200	

EventQ Port Requirements

Ports Used by TeamForge EventQ Services

Port	Service	Host
8844	HTTP/HTTPS	App Server
6379	Redis	App Server
27017	MongoDB	DB Server
28017	MongoDB	DB Server
5672	RabbitMQ	MQ Server
15672	RabbitMQ Management Console	MQ Server

Ports to be Open in a Firewall Environment for EventQ

The following use cases detail TeamForge EventQ’s firewall/routing requirements. By default, end-user web access is proxied through the primary TeamForge web server. TeamForge EventQ adapters supply data using the MQ layer and therefore need access to the MQ server (default port 5672). There are also private access requirements between the various installed services as detailed below.

Port	From	To	Description
443/80	App server	TeamForge server	App communication with TeamForge server
8844	TeamForge server	App server	TeamForge communication with App server
5672	TeamForge EventQ Adapters	MQ server	Message communication between Adapters and MQ server

Port	From	To	Description
5672	App server	MQ server	App communication with MQ server
5672	TeamForge server	MQ server	TeamForge communication with MQ server
15672	App server	MQ server	App administration of MQ server
27017	App server	DB server	App server communication with DB server
22	App server	MQ server	App ssh to MQ server, installation only
22	App server	DB server	App ssh to DB server, installation only

1. CollabNet releases new versions of integration plugins from time to time. It is recommended to upgrade your TeamForge-Nexus and TeamForge-Artifactory integration plugins whenever a new version is available. □ □[2]

2. Use the right version of the TeamForge Associations Add-on for the JIRA software you use. See JIRA Version Information. □

After installing the operating system, prepare the networking connections and configuration for your TeamForge site.

> **NOTE:** You must have root access to all the hosts you will be setting up for your site.

1. Use the `NetworkManager` to list the DNS servers you want to use for resolving Internet addresses.
2. Open the appropriate ports, and close all other ports. See Port Requirements.
3. Use the hostname command to verify that the machine name is resolvable on the network.
   ```
   hostname
   bigbox.supervillain.org
   ```
4. Use the `nslookup` command to verify that your hostname maps to the right IP address.
   ```
   nslookup bigbox.supervillain.org
   Server: 204.16.107.137
   Address: 204.16.107.137#53
   ```

   > **TIP:** If there is any doubt about what the system’s real IP address is, use the /sbin/ifconfig command.

5. If you are installing behind a proxy, specify your proxy settings.
   ```
   export http_proxy=http://<PROXY_USERNAME>:<PROXY_PASSWD>@<PROXY_HOST>:<PRO
   XY_PORT>
   export no_proxy=localhost,127.0.0.0/8,<hostname>
   ```
6. Use a tool such as **Nessus** to scan your server for potential vulnerabilities. See Port Requirements for detailed security recommendations.

To use HTTPS for web traffic, you will need to obtain a valid Apache SSL certificate.

When generating an Apache (mod_ssl) SSL certificate, you have two options:

- Purchase a SSL certificate from a certificate authority (CA). Searching the Web for “certificate authority” will present several choices.
- Generate a self-signed certificate. This option costs nothing and provides the same level of encryption as a certificate purchased from a certificate authority (CA). However, this option can be a mild annoyance to some users, because Internet Explorer (IE) issues a harmless warning each time a user visits a site that uses a self-signed certificate.

> **IMPORTANT:** SSL is enabled by default and a self-signed certificate is auto-generated.

Regardless of which option you select, the process is almost identical.

1. Know the fully qualified domain name (FQDN) of the website for which you want to request a certificate. If you want to access your site through `https://www.example.com`, then the FQDN of your website is `www.example.com`.
   This is also known as your common name.
2. Generate the key with the SSL `genrsa` command.

   ```
   openssl genrsa -out www.example.com.key 1024
   ```

   This command generates a 1024 bit RSA private key and stores it in the file `www.example.com.key`.

   > **TIP:** Back up your www.example.com.key file, because without this file your SSL certificate will not be valid.

3. Generate the CSR with SSL req command.

   ```
   openssl req -new -key www.example.com.key -out www.example.com.csr
   ```

   This command will prompt you for the X.509 attributes of your certificate. Give the fully qualified domain name, such as `www.example.com`, when prompted for `Common Name`.

   Do not enter your personal name here. It is requesting a certificate for a webserver, so the `Common Name` has to match the FQDN of your website.
4. Generate a self-signed certificate.

   ```
   openssl x509 -req -days 370 -in www.example.com.csr -signkey www.example.com.key -out www.example.com.crt
   ```

   This command will generate a self-signed certificate in `www.example.com.crt`.

You will now have an RSA private key in `www.example.com.key`, a Certificate Signing Request in `www.example.com.csr`, and an SSL certificate in `www.example.com.crt`. The self-signed SSL certificate that you generated will be valid for 370 days.

The easiest way to install TeamForge is to install it on a single server, dedicated to TeamForge taking the default configuration settings.

Dos and Don’ts

Here’s a list of dos, don’ts and points to remember when you install or upgrade TeamForge. Show/Hide

Dos

- Understand TeamForge installation requirements and plan your installation or upgrade.
- Get your TeamForge license key and keep it handy.
- Verify your basic networking setup before installing or upgrading TeamForge. See Set Up Networking for TeamForge.
- Look for new or modified `site-options.conf` tokens and update your `site-options.conf` file as required during the upgrade process. See Site Options Change Log.
- Set up a TeamForge Stage Server before you upgrade your Production Server.
- Stop TeamForge services on all servers in a distributed setup while upgrading to TeamForge 18.1.
- Uninstall hot fixes and add-ons, if any, before you start the TeamForge 18.1 upgrade procedure.

- TeamForge 18.1 supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.

  - JBOSS_JAVA_OPTS
  - PHOENIX_JAVA_OPTS
  - INTEGRATION_JAVA_OPTS
  - ETL_JAVA_OPTS
  - ELASTICSEARCH_JAVA_OPTS

  TeamForge provision fails on sites that use these options post TeamForge 18.1 upgrade.

Don'ts

- Do not customize your operating system installation. Select only the default packages list.
- While upgrading TeamForge, whether in place or on new hardware, always reuse the old `site-options.conf` file and make changes as necessary. **Do not** try to start with a new `site-options.conf` file. Reusing the old `site-options.conf` avoids many potential problems, particularly around the management of usernames and passwords.

- Do not manually modify TeamForge-managed site option tokens such as the `AUTO_DATA` token. See AUTO_DATA for more information.

- If you are creating symlinks, note that you must create symlinks only to the TeamForge data directory (`/opt/collabnet/teamforge/var`). You should not create symlinks to TeamForge application directories (such as `/opt/collabnet`).

Points to Remember

- Installing or upgrading TeamForge needs root privileges. You must log on as root or use a root shell to install or upgrade TeamForge.
- SSL is enabled by default and a self-signed certificate is auto-generated. However, you can use a few `site-options.conf` tokens to adjust this behavior. To generate the SSL certificates, see Generate SSL Certificates.
- For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.
- If you have Git integration on a separate server, both TeamForge and Git servers must have their time and date synchronized.
- While you can run both EventQ and TeamForge on the same server, CollabNet recommends such an approach only for testing purposes. It's always recommended to run EventQ on a separate server for optimal scalability.
- No backup is required for same hardware upgrades. However, you can create a backup as a measure of caution. See Back up and Restore TeamForge for more information.
- Always use compatible JDBC drivers meant for specific database versions. See JDBC Drivers Reference for more information. Also see: Why do ETL jobs fail post TeamForge upgrade?.
- You can run the initial load job any time after the installation of TeamForge. We recommend that you run it before you hand over the site to the users. For more information, see ETL Initial Load Jobs.
- SOAP50 APIs and event handlers are no longer supported in TeamForge 16.10 and later. Use the latest TeamForge SOAP/REST APIs.
- TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.
- Installing TeamForge with service-specific FQDNs (instead of machine-specific host/domain names) is highly recommended so that you will be able to change the system landscape at a later point in time without having any impact on the URLs (in other words, end users do not have to notice or change anything). For example, you can create FQDNs specifically for services such as Subversion, Git, mail, Codesearch and so on. For more information, see Service-specific FQDNs.
- All such service-specific FQDNs must belong to a single sub domain and it is recommended to create a new sub domain for TeamForge.
- If you are using service-specific FQDNs

    - A wildcard SSL cert is required. SNI SSL cert cannot be used.
    - When SSL is enabled and no custom SSL certificates are provided, a self-signed wildcard cert is generated for the sub domain.
    - When SSL is enabled and a custom SSL certificate is provided, the CN of the certificate is verified to be a wildcard CN.
- You cannot have a separate PUBLIC_FQDN for EventQ.
- The ability to run separate PostgreSQL instances for TeamForge database and datamart on the same server is being deprecated in TeamForge 17.11. If you have TeamForge database and datamart on separate PostgreSQL instances on the same server and if you are upgrading on a new hardware, you must Create a Single Cluster for Both Database and Datamart while upgrading to TeamForge 17.11 or later.
- While upgrading TeamForge-Git integration servers, it is important that Git master and slave servers are upgraded to the same version of TeamForge-Git integration. On sites with Git Replica Servers, you must upgrade the Git Replica Servers first and then upgrade the master Git servers.

Single Server Setup

You can install TeamForge on both RHEL/CentOS 7.4 and 6.9. In this single server setup, all the following TeamForge services are installed on a single RHEL/CentOS server.

TeamForge Application Server (server-01)

- TeamForge Application Server (ctfcore)
- Database Server (ctfcore-database and ctfcore-datamart)
- Mail Server (mail)
- Code Search Server (codesearch)
- ETL Server (etl)
- Git Integration Server (gerrit and gerrit-database)
- Review Board (reviewboard, reviewboard-database and reviewboard-adapter)
- Binary (binary and binary-database)
- SCM Integration Server (subversion and cvs)
- Search Server (search).
- TeamForge EventQ Server (rabbitmq, mongodb, redis, eventq)
- CLI Server (cliserver)

Do This Step by Step on the TeamForge Application Server (server-01)

1. Install RHEL/CentOS 7.4 or 6.9 and log on as root.

   ✓ The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux.

   ✓ See the RHEL 7.4 Installation Guide or RHEL 6.9 Installation Guide for help.

   ✓ Delete the python-crypto package if you are installing TeamForge on RHEL/CentOS 6.9. `yum erase python-crypto`

2. Check your networking setup. See Set up Networking for more information.

3. Configure the TeamForge installation repository

TeamForge Installation Repository Configuration for Sites with Internet Access

1. Contact the CollabNet Support and download the TeamForge 18.1 installation repository package to `/tmp`.
2. Install the repository package.
   ```
   yum install -y /tmp/collabnet-teamforge-repo-18.1-0-noarch.rpm
   ```
3. Refresh your repository cache.
   ```
   yum clean all
   ```

TeamForge Installation Repository Configuration for Sites without Internet Access

1. Contact the CollabNet Support to get the auxiliary installer package for TeamForge 18.1 disconnected installation and save it in `/tmp`.
   - RHEL/CentOS 7.4 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel7.x86_64.rpm`
   - In addition to the above CentOS 7.4 64 bit RPM package, you must get the following CentOS 7.4 compatibility RPM, which is required for TeamForge 18.1 disconnected media

installation on CentOS 7.4 profile: `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm`.

2. Unpack the disconnected installation package.
   ```
   rpm -ivh <package-name>
   ```
3. Unpack the `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm` package if you are installing TeamForge 18.1 on CentOS 7.4.
   ```
   rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
   ```

4. If not mounted already, mount the RHEL/CentOS installation DVD.

   The DVD contains the necessary software and utilities required for installing TeamForge without internet access. In the following commands, replace “cdrom” with the identifier for your server’s CD/DVD drive, if necessary.

   ```
   cd /media/
   mkdir cdrom
   mount /dev/cdrom ./cdrom/
   ```

   If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

5. Create a yum configuration file that points to the RHEL/CentOS installation DVD.
   ```
   vi /etc/yum.repos.d/cdrom.repo
   ```

   Here’s a sample yum configuration file.

   ```
   [RHEL-CDROM]
   name=RHEL CDRom
   baseurl=file:///media/cdrom/Server/
   gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
   enabled=1
   gpgcheck=0
   ```
6. Verify your yum configuration files.
   ```
   yum list httpd
   yum list apr
   ```

4. Install the TeamForge application packages.
   ```
   yum install teamforge
   ```
5. Set up your site’s master configuration file.
   ```
   vi /opt/collabnet/teamforge/etc/site-options.conf
   ```

host:SERVICES Token

```
server-01:SERVICES=ctfcore ctfcore-database search mail codesearch rabbitmq mongodb redis eventq etl ctfcore-datamart subversion cvs gerrit gerrit-database binary binary-database reviewboard reviewboard-database reviewboard-adapter cliserver
```

> **NOTE:** You may remove the identifiers of components you do not want. For example, remove `binary` and `binary-database` if you are not planning to install binary repository managers such as Nexus. See TeamForge services for more information.

host:PUBLIC_FQDN Token

```
server-01:PUBLIC_FQDN=my.app.domain.com
```

> **NOTE:** You cannot have a separate PUBLIC_FQDN for EventQ.

Save the `site-options.conf` file.

For further customization of your site configuration (SSL settings, password policy settings, PostgreSQL settings, LDAP settings and so on): Site options configuration contd…

SSL Tokens

SSL is enabled by default and a self-signed certificate is auto-generated. Use the following tokens to adjust this behavior.

```
SSL_CERT_FILE=
SSL_KEY_FILE=
SSL_CHAIN_FILE=
```

- To generate the SSL certificates, see Generate SSL certificates.
- Have the custom SSL certificate and private key for custom SSL certificate in place and provide their absolute paths in these tokens. SSL_CHAIN_FILE (intermediate certificate) is optional.
- You can also encrypt the data traffic between the application and database servers and between the ETL and datamart servers in a distributed setup. Use the [DATABASE_SSL] [siteoptiontokens.html#DATABASE_SSL] token to do that. See Encrypt Database Network Traffic.

Password Tokens

- TeamForge 7.1 and later support automatic password creation. See AUTO_DATA for more information.
- Set the REQUIRE_PASSWORD_SECURITY token to `true` to enforce password security policy for the site.

  If the token REQUIRE_PASSWORD_SECURITY is enabled, then set a value for the token, PASSWORD_CONTROL_EFFECTIVE_DATE.

  > **WARNING:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the `PASSWORD_CONTROL_EFFECTIVE_DATE` token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See PASSWORD_CONTROL_EFFECTIVE_DATE for more information.

  You can also set the following tokens to enforce a more stricter password policy:

  - MINIMUM_PASSWORD_LENGTH
  - MAX_PASSWORD_LENGTH
  - PASSWORD_REQUIRES_NUMBER
  - PASSWORD_REQUIRES_NON_ALPHANUM
  - PASSWORD_REQUIRES_MIXED_CASE
  - REQUIRE_PASSWORD_SECURITY
  - LOGIN_ATTEMPT_LOCK
  - PASSWORD_HISTORY_AGE
  - ALLOW_PASSWORD_DICTIONARY_WORD
- If the token REQUIRE_RANDOM_ADMIN_PASSWORD is already set to `true`, then set the token ADMIN_EMAIL with a valid email address.
  `ADMIN_EMAIL=root@{__APPLICATION_HOST__}`
- If you have LDAP set up for external authentication, you must set the REQUIRE_USER_PASSWORD_CHANGE site options token to `false`.

Prevent Cross-site Scripting

An attacker could potentially upload an HTML page to TeamForge that contains active code, such as JavaScript. This active code would then be executed by clients' browsers when they view the page, which can harm the system.

To prevent an attack of this sort, you can specify whether or not HTML code is displayed in TeamForge. This flag applies to all documents, tracker, task, and forum attachments, and files in the file release system.

Set the SAFE_DOWNLOAD_MODE token according to your requirements. For more information, see SAFE_DOWNLOAD_MODE.

PostgreSQL Tokens and Settings

Make sure the PostgreSQL tokens in the `site-options.conf` file are set as recommended in the following topic: What are the right PostgreSQL settings for my site?

Save the `site-options.conf` file.

6. Provision services.
   `teamforge provision`
   TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.
7. Verify TeamForge installation.
   1. Reboot the server and make sure all services come up automatically at startup.
   2. Log on to the TeamForge web application using the default Admin credentials.
      - Username: `admin`
      - Password: `admin`
   3. Create a sample project. See Create a TeamForge Project.
   4. Write a welcome message to your site's users. See Create a Site-wide Broadcast.

Post Install Tasks

- Supply Your TeamForge License Key
- Run TeamForge in SELinux `enabled` Mode
- Install EventQ Integration Adapters

- Users are not getting email notifications for review requests and reviews. What should I do?

Distributed setup with TeamForge, Database (including Datamart), EventQ, Review Board, SCM (Subversion, CVS and Git) and Code Search installed on separate servers.

✓ In this distributed setup, TeamForge services are distributed across six servers, server-01 through server-06 as illustrated in the following table.

✓ You can install TeamForge on both RHEL/CentOS 7.4 and 6.9. In this distributed setup, all the following services are installed on RHEL/CentOS 7.4 servers.

server-01	**server-02**	**server-03**	**server-04**	**server-05**	**server-06**
TeamForge Application Server	**TeamForge Database Server**	**EventQ Server**	**Review Board Server**	**SCM Server**	**Code Search Server**
ctfcore	ctfcore-database	eventq	reviewboard	subversion	codesearch
mail	ctfcore-datamart	rabbitmq		cvs	
etl	gerrit-database	mongodb		gerrit	
search	binary-database	redis			
reviewboard-adapter[1]	reviewboard-database				
binary					
cliserver					

Dos and Don’ts

Here’s a list of dos, don’ts and points to remember when you install or upgrade TeamForge. Show/Hide

Dos

- Understand TeamForge installation requirements and plan your installation or upgrade.
- Get your TeamForge license key and keep it handy.
- Verify your basic networking setup before installing or upgrading TeamForge. See Set Up Networking for TeamForge.
- Look for new or modified `site-options.conf` tokens and update your `site-options.conf` file as required during the upgrade process. See Site Options Change Log.
- Set up a TeamForge Stage Server before you upgrade your Production Server.
- Stop TeamForge services on all servers in a distributed setup while upgrading to TeamForge 18.1.
- Uninstall hot fixes and add-ons, if any, before you start the TeamForge 18.1 upgrade procedure.

- TeamForge 18.1 supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.
    - JBOSS_JAVA_OPTS
    - PHOENIX_JAVA_OPTS
    - INTEGRATION_JAVA_OPTS
    - ETL_JAVA_OPTS
    - ELASTICSEARCH_JAVA_OPTS

  TeamForge provision fails on sites that use these options post TeamForge 18.1 upgrade.

Don'ts

- Do not customize your operating system installation. Select only the default packages list.
- While upgrading TeamForge, whether in place or on new hardware, always reuse the old `site-options.conf` file and make changes as necessary. **Do not** try to start with a new `site-options.conf` file. Reusing the old `site-options.conf` avoids many potential problems, particularly around the management of usernames and passwords.
- Do not manually modify TeamForge-managed site option tokens such as the `AUTO_DATA` token. See AUTO_DATA for more information.

- If you are creating symlinks, note that you must create symlinks only to the TeamForge data directory (`/opt/collabnet/teamforge/var`). You should not create symlinks to TeamForge application directories (such as `/opt/collabnet`).

Points to Remember

- Installing or upgrading TeamForge needs root privileges. You must log on as root or use a root shell to install or upgrade TeamForge.
- SSL is enabled by default and a self-signed certificate is auto-generated. However, you can use a few `site-options.conf` tokens to adjust this behavior. To generate the SSL certificates, see Generate SSL Certificates.
- For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.
- If you have Git integration on a separate server, both TeamForge and Git servers must have their time and date synchronized.

- While you can run both EventQ and TeamForge on the same server, CollabNet recommends such an approach only for testing purposes. It's always recommended to run EventQ on a separate server for optimal scalability.
- No backup is required for same hardware upgrades. However, you can create a backup as a measure of caution. See Back up and Restore TeamForge for more information.
- Always use compatible JDBC drivers meant for specific database versions. See JDBC Drivers Reference for more information. Also see: Why do ETL jobs fail post TeamForge upgrade?.
- You can run the initial load job any time after the installation of TeamForge. We recommend that you run it before you hand over the site to the users. For more information, see ETL Initial Load Jobs.
- SOAP50 APIs and event handlers are no longer supported in TeamForge 16.10 and later. Use the latest TeamForge SOAP/REST APIs.
- TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.
- Installing TeamForge with service-specific FQDNs (instead of machine-specific host/domain names) is highly recommended so that you will be able to change the system landscape at a later point in time without having any impact on the URLs (in other words, end users do not have to notice or change anything). For example, you can create FQDNs specifically for services such as Subversion, Git, mail, Codesearch and so on. For more information, see Service-specific FQDNs.
- All such service-specific FQDNs must belong to a single sub domain and it is recommended to create a new sub domain for TeamForge.
- If you are using service-specific FQDNs
    - A wildcard SSL cert is required. SNI SSL cert cannot be used.
    - When SSL is enabled and no custom SSL certificates are provided, a self-signed wildcard cert is generated for the sub domain.
    - When SSL is enabled and a custom SSL certificate is provided, the CN of the certificate is verified to be a wildcard CN.
- You cannot have a separate PUBLIC_FQDN for EventQ.
- The ability to run separate PostgreSQL instances for TeamForge database and datamart on the same server is being deprecated in TeamForge 17.11. If you have TeamForge database and datamart on separate PostgreSQL instances on the same server and if you are upgrading on a new hardware, you must Create a Single Cluster for Both Database and Datamart while upgrading to TeamForge 17.11 or later.
- While upgrading TeamForge-Git integration servers, it is important that Git master and slave servers are upgraded to the same version of TeamForge-Git integration. On sites with Git Replica Servers, you must upgrade the Git Replica Servers first and then upgrade the master Git servers.

Prepare the Servers for TeamForge Installation (server-01 through server-06)

1. Install RHEL/CentOS 7.4 and log on as root.

   ✓ The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux.

   ✓ See the RHEL 7.4 Installation Guide for help.

2. Check your networking setup. See Set up Networking for more information.

3. Configure the TeamForge installation repository

TeamForge Installation Repository Configuration for Sites with Internet Access

1. Contact the CollabNet Support and download the TeamForge 18.1 installation repository package to `/tmp`.
2. Install the repository package.
   ```
   yum install -y /tmp/collabnet-teamforge-repo-18.1-0-noarch.rpm
   ```
3. Refresh your repository cache.
   ```
   yum clean all
   ```

TeamForge Installation Repository Configuration for Sites without Internet Access

1. Contact the CollabNet Support to get the auxiliary installer package for TeamForge 18.1 disconnected installation and save it in `/tmp`.
   - RHEL/CentOS 7.4 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel7.x86_64.rpm`
   - In addition to the above CentOS 7.4 64 bit RPM package, you must get the following CentOS 7.4 compatibility RPM, which is required for TeamForge 18.1 disconnected media installation on CentOS 7.4 profile: `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm`.
2. Unpack the disconnected installation package.

```
rpm -ivh <package-name>
```

3. Unpack the `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm` package if you are installing TeamForge 18.1 on CentOS 7.4.

   ```
   rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
   ```

4. If not mounted already, mount the RHEL/CentOS installation DVD.

   The DVD contains the necessary software and utilities required for installing TeamForge without internet access. In the following commands, replace “cdrom” with the identifier for your server’s CD/DVD drive, if necessary.

   ```
   cd /media/
   mkdir cdrom
   mount /dev/cdrom ./cdrom/
   ```

   If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

5. Create a yum configuration file that points to the RHEL/CentOS installation DVD.

   ```
   vi /etc/yum.repos.d/cdrom.repo
   ```

   Here’s a sample yum configuration file.

   ```
   [RHEL-CDROM]
   name=RHEL CDRom
   baseurl=file:///media/cdrom/Server/
   gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
   enabled=1
   gpgcheck=0
   ```

6. Verify your yum configuration files.

   ```
   yum list httpd
   yum list apr
   ```

Install TeamForge Services

1. Install TeamForge application services on the TeamForge Application Server (server-01)

   ```
   yum install teamforge
   ```

2. Install TeamForge database services on the TeamForge Database Server (server-02)

   ```
   yum install teamforge
   ```

3. Install EventQ services on the EventQ Server (server-03)

```
   yum install teamforge-eventq
   ```

4. Install Review Board services on the Review Board Server (server-04)
   ```
   yum install teamforge
   ```
5. Install SCM services on the SCM Server (server-05)
   ```
   yum install teamforge-scm teamforge-git
   ```
6. Install the Code Search service on the Code Search Server (server-06)
   ```
   yum install teamforge-codesearch
   ```

Set up Your Site’s Master Configuration File

1. Do this on the TeamForge Database Server (server-02).
   ```
   vi /opt/collabnet/teamforge/etc/site-options.conf
   ```

host:SERVICES Token

```
server-01:SERVICES=ctfcore search mail etl binary reviewboard-adapter cliserver
server-02:SERVICES=ctfcore-database ctfcore-datamart gerrit-database binary-database reviewboard-database
server-03:SERVICES=eventq rabbitmq mongodb redis
server-04:SERVICES=reviewboard
server-05:SERVICES=subversion cvs gerrit
server-06:SERVICES=codesearch
```

host:PUBLIC_FQDN Token

```
server-01:PUBLIC_FQDN=my.app.domain.com
```

> **NOTE:** You cannot have a separate PUBLIC_FQDN for EventQ.

Save the `site-options.conf` file.

For further customization of your site configuration (SSL settings, password policy settings, PostgreSQL settings, LDAP settings and so on): Site options configuration contd…

SSL Tokens

SSL is enabled by default and a self-signed certificate is auto-generated. Use the following tokens to adjust this behavior.

```
SSL_CERT_FILE=
SSL_KEY_FILE=
SSL_CHAIN_FILE=
```

- To generate the SSL certificates, see Generate SSL certificates.
- Have the custom SSL certificate and private key for custom SSL certificate in place and provide their absolute paths in these tokens. SSL_CHAIN_FILE (intermediate certificate) is optional.
- You can also encrypt the data traffic between the application and database servers and between the ETL and datamart servers in a distributed setup. Use the [DATABASE_SSL] [siteoptiontokens.html#DATABASE_SSL] token to do that. See Encrypt Database Network Traffic.

Password Tokens

- TeamForge 7.1 and later support automatic password creation. See AUTO_DATA for more information.

- Set the REQUIRE_PASSWORD_SECURITY token to `true` to enforce password security policy for the site.

  If the token REQUIRE_PASSWORD_SECURITY is enabled, then set a value for the token, PASSWORD_CONTROL_EFFECTIVE_DATE.

  > **WARNING:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the `PASSWORD_CONTROL_EFFECTIVE_DATE` token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See PASSWORD_CONTROL_EFFECTIVE_DATE for more information.

  You can also set the following tokens to enforce a more stricter password policy:

  - MINIMUM_PASSWORD_LENGTH
  - MAX_PASSWORD_LENGTH
  - PASSWORD_REQUIRES_NUMBER
  - PASSWORD_REQUIRES_NON_ALPHANUM

    - PASSWORD_REQUIRES_MIXED_CASE
    - REQUIRE_PASSWORD_SECURITY
    - LOGIN_ATTEMPT_LOCK
    - PASSWORD_HISTORY_AGE
    - ALLOW_PASSWORD_DICTIONARY_WORD
- If the token REQUIRE_RANDOM_ADMIN_PASSWORD is already set to `true`, then set the token ADMIN_EMAIL with a valid email address.
  ```
  ADMIN_EMAIL=root@{__APPLICATION_HOST__}
  ```
- If you have LDAP set up for external authentication, you must set the REQUIRE_USER_PASSWORD_CHANGE site options token to `false`.

Prevent Cross-site Scripting

An attacker could potentially upload an HTML page to TeamForge that contains active code, such as JavaScript. This active code would then be executed by clients' browsers when they view the page, which can harm the system.

To prevent an attack of this sort, you can specify whether or not HTML code is displayed in TeamForge. This flag applies to all documents, tracker, task, and forum attachments, and files in the file release system.

Set the SAFE_DOWNLOAD_MODE token according to your requirements. For more information, see SAFE_DOWNLOAD_MODE.

PostgreSQL Tokens and Settings

Make sure the PostgreSQL tokens in the `site-options.conf` file are set as recommended in the following topic: What are the right PostgreSQL settings for my site?

Save the `site-options.conf` file.

2. Provision the Database Server (server-02).
   ```
   teamforge provision
   ```
3. Copy the `/opt/collabnet/teamforge/etc/site-options.conf` file from the TeamForge Database Server (server-02) to the `/opt/collabnet/teamforge/etc/` directory of all other servers.

Provision Services on All the Servers

1. Provision services.
   `teamforge provision`
   TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.

✓ You must provision services in a particluar sequence. Usually you start with the Database Server, followed by the Application Server and then by other servers such as SCM, Review Board, EventQ and Code Search servers.

✓ The TeamForge installer derives this sequence from your `site-options.conf` file and shows you the order of provisioning servers when you try to provision one of the distributed servers. Follow the exact sequence as instructed.

1. Provision the Application Server (server-01)
2. Provision the SCM server (server-05)
3. Provision the EventQ Server (server-03)
4. Provision the Review Board Server (server-04)
5. Provision the Code Search Server (server-06)

Reinitialize TeamForge

1. Reinitialize TeamForge on the Review Board Server.
   `teamforge reinitialize`
2. During `teamforge provision`, the `Register SCM integration` process fails on sites that use self-signed certificates. Perform these steps in such cases.
   1. Restart JBoss on the TeamForge Application Server.
      `teamforge restart -s jboss`
   2. Reinitialize TeamForge on the SCM Server.
      `teamforge reinitialize`

   Do you have Git and other SCM tools (SVN and CVS) on two separate servers?
   Git and other SCM tools (SVN and CVS) are typically installed on a separate server dedicated for SCM. However, if you have Git and SCM (SVN and CVS) on two separate servers, restart Jboss on the TeamForge Application Server and reinitialize TeamForge on the SCM Server (SVN and CVS) as discussed earlier. In addition, you must also restart TeamForge on the Git Server.

   Restart TeamForge on the Git Server: `teamforge restart`

Verify TeamForge Installation

1. Verify TeamForge installation.
    1. Reboot the server and make sure all services come up automatically at startup.
    2. Log on to the TeamForge web application using the default Admin credentials.
        - Username: `admin`
        - Password: `admin`
    3. Create a sample project. See Create a TeamForge Project.
    4. Write a welcome message to your site's users. See Create a Site-wide Broadcast.

Post Install Tasks

- Supply Your TeamForge License Key
- Run TeamForge in SELinux enabled Mode
- Install EventQ Integration Adapters
- Users are not getting email notifications for review requests and reviews. What should I do?

1. `reviewboard-adapter` must always be installed on the TeamForge Application Server. □

Distributed setup with TeamForge, Oracle Database (including Datamart) and EventQ installed on separate servers.

✓ In this distributed setup, TeamForge, Oracle database and other services are distributed across three servers, server-01 through server-03 as illustrated in the following table.

✓ You can install TeamForge on both RHEL/CentOS 7.4 and 6.9. In this distributed setup, all the following services are installed on RHEL/CentOS 7.4 servers.

server-01	server-02	server-03
TeamForge Application Server	**Oracle Database Server**	**EventQ Server**
ctfcore	ctfcore-database	eventq
mail	ctfcore-datamart	rabbitmq
etl		mongodb
search		redis
codesearch		
gerrit		
gerrit-database		
subversion		

server-01	server-02	server-03
TeamForge Application Server	**Oracle Database Server**	**EventQ Server**
cvs		
reviewboard		
reviewboard-database		
reviewboard-adapter[1]		
binary		
binary-database		
cliserver		

Dos and Don’ts

Here’s a list of dos, don’ts and points to remember when you install or upgrade TeamForge. Show/Hide

Dos

- Understand TeamForge installation requirements and plan your installation or upgrade.
- Get your TeamForge license key and keep it handy.
- Verify your basic networking setup before installing or upgrading TeamForge. See Set Up Networking for TeamForge.
- Look for new or modified `site-options.conf` tokens and update your `site-options.conf` file as required during the upgrade process. See Site Options Change Log.
- Set up a TeamForge Stage Server before you upgrade your Production Server.
- Stop TeamForge services on all servers in a distributed setup while upgrading to TeamForge 18.1.
- Uninstall hot fixes and add-ons, if any, before you start the TeamForge 18.1 upgrade procedure.

- TeamForge 18.1 supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.

  - JBOSS_JAVA_OPTS
  - PHOENIX_JAVA_OPTS
  - INTEGRATION_JAVA_OPTS
  - ETL_JAVA_OPTS
  - ELASTICSEARCH_JAVA_OPTS

  TeamForge provision fails on sites that use these options post TeamForge 18.1 upgrade.

Don'ts

- Do not customize your operating system installation. Select only the default packages list.
- While upgrading TeamForge, whether in place or on new hardware, always reuse the old `site-options.conf` file and make changes as necessary. **Do not** try to start with a new `site-options.conf` file. Reusing the old `site-options.conf` avoids many potential problems, particularly around the management of usernames and passwords.
- Do not manually modify TeamForge-managed site option tokens such as the `AUTO_DATA` token. See AUTO_DATA for more information.

- If you are creating symlinks, note that you must create symlinks only to the TeamForge data directory (`/opt/collabnet/teamforge/var`). You should not create symlinks to TeamForge application directories (such as `/opt/collabnet`).

Points to Remember

- Installing or upgrading TeamForge needs root privileges. You must log on as root or use a root shell to install or upgrade TeamForge.
- SSL is enabled by default and a self-signed certificate is auto-generated. However, you can use a few `site-options.conf` tokens to adjust this behavior. To generate the SSL certificates, see Generate SSL Certificates.
- For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.
- If you have Git integration on a separate server, both TeamForge and Git servers must have their time and date synchronized.
- While you can run both EventQ and TeamForge on the same server, CollabNet recommends such an approach only for testing purposes. It’s always recommended to run EventQ on a separate server for optimal scalability.
- No backup is required for same hardware upgrades. However, you can create a backup as a measure of caution. See Back up and Restore TeamForge for more information.
- Always use compatible JDBC drivers meant for specific database versions. See JDBC Drivers Reference for more information. Also see: Why do ETL jobs fail post TeamForge upgrade?.
- You can run the initial load job any time after the installation of TeamForge. We recommend that you run it before you hand over the site to the users. For more information, see ETL Initial Load Jobs.
- SOAP50 APIs and event handlers are no longer supported in TeamForge 16.10 and later. Use the latest TeamForge SOAP/REST APIs.
- TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.

- Installing TeamForge with service-specific FQDNs (instead of machine-specific host/domain names) is highly recommended so that you will be able to change the system landscape at a later point in time without having any impact on the URLs (in other words, end users do not have to notice or change anything). For example, you can create FQDNs specifically for services such as Subversion, Git, mail, Codesearch and so on. For more information, see Service-specific FQDNs.
- All such service-specific FQDNs must belong to a single sub domain and it is recommended to create a new sub domain for TeamForge.
- If you are using service-specific FQDNs
  - A wildcard SSL cert is required. SNI SSL cert cannot be used.
  - When SSL is enabled and no custom SSL certificates are provided, a self-signed wildcard cert is generated for the sub domain.
  - When SSL is enabled and a custom SSL certificate is provided, the CN of the certificate is verified to be a wildcard CN.
- You cannot have a separate PUBLIC_FQDN for EventQ.
- The ability to run separate PostgreSQL instances for TeamForge database and datamart on the same server is being deprecated in TeamForge 17.11. If you have TeamForge database and datamart on separate PostgreSQL instances on the same server and if you are upgrading on a new hardware, you must Create a Single Cluster for Both Database and Datamart while upgrading to TeamForge 17.11 or later.
- While upgrading TeamForge-Git integration servers, it is important that Git master and slave servers are upgraded to the same version of TeamForge-Git integration. On sites with Git Replica Servers, you must upgrade the Git Replica Servers first and then upgrade the master Git servers.

Prepare the Servers for TeamForge Installation (server-01 through server-03)

1. Install RHEL/CentOS 7.4 and log on as root.

   ✓ The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux.

   ✓ See the RHEL 7.4 Installation Guide for help.

2. Check your networking setup. See Set up Networking for more information.

3. Configure the TeamForge installation repository

TeamForge Installation Repository Configuration for Sites with Internet Access

1. Contact the CollabNet Support and download the TeamForge 18.1 installation repository package to `/tmp`.
2. Install the repository package.
   ```
   yum install -y /tmp/collabnet-teamforge-repo-18.1-0-noarch.rpm
   ```
3. Refresh your repository cache.
   ```
   yum clean all
   ```

TeamForge Installation Repository Configuration for Sites without Internet Access

1. Contact the CollabNet Support to get the auxiliary installer package for TeamForge 18.1 disconnected installation and save it in `/tmp`.
   - RHEL/CentOS 7.4 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel7.x86_64.rpm`
   - In addition to the above CentOS 7.4 64 bit RPM package, you must get the following CentOS 7.4 compatibility RPM, which is required for TeamForge 18.1 disconnected media installation on CentOS 7.4 profile: `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm`.
2. Unpack the disconnected installation package.
   ```
   rpm -ivh <package-name>
   ```
3. Unpack the `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm` package if you are installing TeamForge 18.1 on CentOS 7.4.
   ```
   rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
   ```

4. If not mounted already, mount the RHEL/CentOS installation DVD.

   The DVD contains the necessary software and utilities required for installing TeamForge without internet access. In the following commands, replace “cdrom” with the identifier for your server’s CD/DVD drive, if necessary.

   ```
   cd /media/
   mkdir cdrom
   mount /dev/cdrom ./cdrom/
   ```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

5. Create a yum configuration file that points to the RHEL/CentOS installation DVD.
   ```
   vi /etc/yum.repos.d/cdrom.repo
   ```

   Here's a sample yum configuration file.

   ```
   [RHEL-CDROM]
   name=RHEL CDRom
   baseurl=file:///media/cdrom/Server/
   gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
   enabled=1
   gpgcheck=0
   ```
6. Verify your yum configuration files.
   ```
   yum list httpd
   yum list apr
   ```

✓ You need not configure the TeamForge installation repository on the Oracle Database Server.

Install the TeamForge Services

1. Install the TeamForge application services on the TeamForge Application Server (server-01)
   ```
   yum install teamforge
   ```
2. Install the Review Board services on the TeamForge Application Server (server-01)
   ```
   yum install teamforge
   ```
3. Download the corresponding version of Oracle client and run the following command on the TeamForge Application Server (server-01).
   ```
   yum localinstall <path to oracle client rpm>
   ```
4. Install EventQ services on the EventQ Server (server-03)
   ```
   yum install teamforge-eventq
   ```

Set up the Oracle Database Server (server-02)

1. Install Oracle 12c.

   > **NOTE:** Make sure your database uses UTF8 or AL32UTF8 encoding. This is needed to support users in Asian languages. See this Oracle knowledge base article.

2. Log on to the Oracle Database Server as a system administrator with `SYSDG` privilege and run the following query.
   ```
   alter system set parallel_threads_per_cpu=4;
   ```

3. Log in as an Oracle user and create the site database user and permissions.

   To use an Oracle database for your TeamForge data, set up the Oracle database and tell the TeamForge installer how to handle it.

TeamForge Database Setup

> **NOTE:** Make sure your database uses UTF8 or AL32UTF8 encoding. This is needed to support users in Asian languages. See this Oracle knowledge base article.

1. Connect to your Oracle database.
   ```
   SQL> connect <adminusername>@<db_name>/<adminpassword> as sysdba
   ```
2. Create the database user and password you will use to connect from TeamForge to Oracle.
   ```
   SQL> create user <sf user> identified by <sf passwd> default tablespace
    <your tablespace> temporary tablespace <temporary tablespace>;
   ```
3. Grant permissions to the user that you just created.
   ```
   SQL> grant unlimited tablespace to <sf user>;
   SQL> grant create snapshot to <sf user>;
   SQL> grant create cluster to <sf user>;
   SQL> grant create database link to <sf user>;
   SQL> grant create procedure to <sf user>;
   SQL> grant create sequence to <sf user>;
   SQL> grant create trigger to <sf user>;
   SQL> grant create type to <sf user>;
   SQL> grant create view to <sf user>;
   SQL> grant query rewrite to <sf user>;
   SQL> grant alter session to <sf user>;
   SQL> grant create table to <sf user>;
   SQL> grant create session to <sf user>;
   SQL> grant create any synonym to <sf user>;
   SQL> exit
   ```
4. Create the database read-only user that you will use to connect from TeamForge.

```
SQL> create user <database_readonly_user> identified by <database_readonly_password> default tablespace <your tablespace> temporary tablespace <temporary tablespace>;
```

5. Grant the required permissions to the read-only user that you just created.

```
SQL> grant create session to <database_readonly_user>;
SQL> exit
```

6. Connect to your Oracle database as .

```
SQL> connect <sf user>@<db_name>/<sf passwd>
```

7. Grant the ‘create synonym’ permission on TeamForge database to the read-only user that you just created.

```
SQL> begin
for i in (select table_name from user_tables) loop
execute immediate 'grant select on '|| i.table_name||' to <database_readonly_user>';
execute immediate 'create synonym <database_readonly_user>.'||i.table_name||' for '||i.table_name||'';
end loop;
end;

SQL> exit
```

TeamForge Datamart Setup

> **NOTE:** Make sure your database uses UTF8 or AL32UTF8 encoding. This is needed to support users in Asian languages. See this Oracle knowledge base article.

1. Connect to your Oracle database.

```
SQL> connect <adminusername>@<db_name>/<adminpassword> as sysdba
```

2. Create the datamart user you will use to connect from TeamForge.

```
SQL> create user <reports_database_user> identified by <reports_database_password> default tablespace <your tablespace> temporary tablespace <temporary tablespace>;
```

3. Grant permissions to the user that you just created.

```
SQL> grant unlimited tablespace to <reports_database_user>;
SQL> grant create snapshot to <reports_database_user>;
SQL> grant create cluster to <reports_database_user>;
SQL> grant create database link to <sreports_database_user>;
SQL> grant create procedure to <reports_database_user>;
SQL> grant create sequence to <reports_database_user>;
SQL> grant create trigger to <reports_database_user>;
SQL> grant create type to <reports_database_user>;
SQL> grant create view to <reports_database_user>;
SQL> grant query rewrite to <reports_database_user>;
SQL> grant alter session to <reports_database_user>;
SQL> grant create table to <reports_database_user>;
SQL> grant create session to <reports_database_user>;
SQL> grant create any synonym to <reports_database_user>;
SQL> exit
```

> **NOTE:** Replace with the datamart username specified in the site-options.conf and with the database password specified in site-options.conf.

4. Create the datamart read-only user that you will use to connect from TeamForge.

```
SQL> create user <reports_readonly_user> identified by <reports_readonl
y_password> default tablespace <your tablespace> temporary tablespace
<temporary tablespace>;
```

5. Grant the required permissions to the read-only user that you just created.

```
SQL> grant create session to <reports_readonly_user>;
SQL> exit
```

> **NOTE:** The TeamForge installer creates the tables and default values for you.

6. Connect to your Oracle database as .

```
SQL> connect <reports_database_user>@<db_name>/<reports_database_passw
ord>
```

7. Grant the ‘create synonym’ permission on TeamForge datamart to the read-only user that you just created. SQL> begin

```
for i in (select table_name from user_tables) loop
execute immediate 'grant select on '|| i.table_name||' to <reports_rea
donly_user>';
execute immediate 'create synonym <reports_readonly_user>.'||i.table_n
ame||' for '||i.table_name||'';
end loop;
end;

SQL> exit
```

4. Log on to the TeamForge Application Server and copy the Oracle Datamart setup script from `/opt/collabnet/teamforge/runtime/scripts/` to the `/tmp` directory of the Oracle Database Server (server-02).
   ```
   scp /opt/collabnet/teamforge/runtime/scripts/datamart-oracle-setup.sh <use
   rname>@<server-02>:/tmp
   ```
5. Copy the Oracle Datamart setup script to `/u1` directory.
   ```
   mkdir /u1
   cp /tmp/datamart-oracle-setup.sh /u1
   ```
6. Create the reporting user and schema.

   > **TIP:** Skip this step if you have already set up the datamart as discussed earlier. Your responses to the `datamart-oracle-setup.sh` script's prompts must match the values of the equivalent variables of the TeamForge Application Server's `site-options.conf` file.

   ```
   cd /u1
   sh datamart-oracle-setup.sh
   ```

Set up the TeamForge Application Server (server-01)

Log on to the TeamForge Application Server (server-01), set up the `site-options.conf` file, and provision the services.

1. Rename the sample Oracle site configuration file in the `/opt/collabnet/teamforge/etc/` directory.
   ```
   cd /opt/collabnet/teamforge/etc/
   cp site-options-oracle.conf site-options.conf
   ```
2. Set up your site's master configuration file.
   ```
   vi /opt/collabnet/teamforge/etc/site-options.conf
   ```

   ## host:SERVICES Token

   ```
   server-01:SERVICES=ctfcore mail etl search subversion cvs codesearch clise
   rver gerrit gerrit-database  binary binary-database reviewboard reviewboar
   d-database reviewboard-adapter
   ```

```
server-02:SERVICES=ctfcore-database ctfcore-datamart
server-03:SERVICES=eventq mongodb redis rabbitmq
```

host:PUBLIC_FQDN Token

```
server-01:PUBLIC_FQDN=my.app.domain.com
```

> **NOTE:** You cannot have a separate PUBLIC_FQDN for EventQ.

Configure the Oracle Database Tokens

Configure the Oracle database name, usernames and passwords as configured on the Oracle Database Server.

- Database type is `oracle` (`DATABASE_TYPE=oracle`)
- Database service name is the host name of the Oracle Database Server (for example, `DATABASE_SERVICE_NAME=cu349.maa.collab.net`)
- Reports database service name is the host name of the server where the datamart is (for example, `REPORTS_DATABASE_SERVICE_NAME=cu349.maa.collab.net`)

```
DATABASE_TYPE=oracle

# Adjust usernames/passwords to match what has been configured on the datab
ase server.
DATABASE_USERNAME=ctfuser
DATABASE_PASSWORD=ctfpwd
DATABASE_READ_ONLY_USER=ctfrouser
DATABASE_READ_ONLY_PASSWORD=ctfropwd
DATABASE_NAME=orcl
DATABASE_SERVICE_NAME=

# Adjust usernames/passwords to match what has been configured on the datab
ase server.
REPORTS_DATABASE_USERNAME=ctfrptuser
REPORTS_DATABASE_PASSWORD=ctfrptpwd
REPORTS_DATABASE_NAME=orcl
REPORTS_DATABASE_READ_ONLY_USER=ctfrptrouser
```

```
REPORTS_DATABASE_READ_ONLY_PASSWORD=ctfrptropwd
REPORTS_DATABASE_SERVICE_NAME=
```

Save the `site-options.conf` file.

For further customization of your site configuration (SSL settings, password policy settings, PostgreSQL settings, LDAP settings and so on): Site options configuration contd…

SSL Tokens

SSL is enabled by default and a self-signed certificate is auto-generated. Use the following tokens to adjust this behavior.

```
SSL_CERT_FILE=
SSL_KEY_FILE=
SSL_CHAIN_FILE=
```

- To generate the SSL certificates, see Generate SSL certificates.
- Have the custom SSL certificate and private key for custom SSL certificate in place and provide their absolute paths in these tokens. SSL_CHAIN_FILE (intermediate certificate) is optional.

Password Tokens

- TeamForge 7.1 and later support automatic password creation. See AUTO_DATA for more information.

- Set the REQUIRE_PASSWORD_SECURITY token to `true` to enforce password security policy for the site.

  If the token REQUIRE_PASSWORD_SECURITY is enabled, then set a value for the token, PASSWORD_CONTROL_EFFECTIVE_DATE.

  > **WARNING:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the `PASSWORD_CONTROL_EFFECTIVE_DATE` token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See PASSWORD_CONTROL_EFFECTIVE_DATE for more information.

  You can also set the following tokens to enforce a more stricter password policy:

  - MINIMUM_PASSWORD_LENGTH
  - MAX_PASSWORD_LENGTH
  - PASSWORD_REQUIRES_NUMBER
  - PASSWORD_REQUIRES_NON_ALPHANUM
  - PASSWORD_REQUIRES_MIXED_CASE
  - REQUIRE_PASSWORD_SECURITY
  - LOGIN_ATTEMPT_LOCK
  - PASSWORD_HISTORY_AGE
  - ALLOW_PASSWORD_DICTIONARY_WORD
- If the token REQUIRE_RANDOM_ADMIN_PASSWORD is already set to `true`, then set the token ADMIN_EMAIL with a valid email address.

  ```
  ADMIN_EMAIL=root@{__APPLICATION_HOST__}
  ```
- If you have LDAP set up for external authentication, you must set the REQUIRE_USER_PASSWORD_CHANGE site options token to `false`.

Prevent Cross-site Scripting

An attacker could potentially upload an HTML page to TeamForge that contains active code, such as JavaScript. This active code would then be executed by clients' browsers when they view the page, which can harm the system.

To prevent an attack of this sort, you can specify whether or not HTML code is displayed in TeamForge. This flag applies to all documents, tracker, task, and forum attachments, and files in the file release system.

Set the SAFE_DOWNLOAD_MODE token according to your requirements. For more information, see SAFE_DOWNLOAD_MODE.

JAVA_OPTS

Configure the JBOSS_JAVA_OPTS site-options.conf token. See JBOSS_JAVA_OPTS.

> **NOTE:** All JVM parameters but `-Xms1024m` and `-Xmx2048m` have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as `-XX:MaxMetaspaceSize=512m` `-XX:ReservedCodeCacheSize=128M` `-server` `-XX:+HeapDumpOnOutOfMemoryError` `-Djsse.enableSNIExtension=false`

> -Dsun.rmi.dgc.client.gcInterval=600000
> -Dsun.rmi.dgc.server.gcInterval=600000) in the site-options.conf file.

TeamForge 18.1 (and later) supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.

- JBOSS_JAVA_OPTS
- PHOENIX_JAVA_OPTS
- INTEGRATION_JAVA_OPTS
- ETL_JAVA_OPTS
- ELASTICSEARCH_JAVA_OPTS

TeamForge provision fails on sites that use these options post upgrade to TeamForge 18.1.

Save the `site-options.conf` file.

3. Provision services.
   `teamforge provision`
   TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.

Set up the EventQ Server (server-03)

Log on to the EventQ Server (server-03), set up the `site-options.conf` file, and provision the services.

1. Copy the `/opt/collabnet/teamforge/etc/site-options.conf` file from the TeamForge Application Server to the EventQ Server's `/opt/collabnet/teamforge/etc/` directory.

2. Provision services.

   `teamforge provision`
   TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.

Verify TeamForge Installation

1. Verify TeamForge installation.

1. Reboot the server and make sure all services come up automatically at startup.
2. Log on to the TeamForge web application using the default Admin credentials.
   - Username: `admin`
   - Password: `admin`
3. Create a sample project. See Create a TeamForge Project.
4. Write a welcome message to your site's users. See Create a Site-wide Broadcast.

Post Install Tasks

- Supply Your TeamForge License Key
- Run TeamForge in SELinux `enabled` Mode
- Install EventQ Integration Adapters
- Users are not getting email notifications for review requests and reviews. What should I do?

1. `reviewboard-adapter` must always be installed on the TeamForge Application Server. □

You can install TeamForge with its database installed separately on an external PostgreSQL server such as AWS RDS/Aurora.

You can install TeamForge with its database installed separately on an external PostgreSQL server such as AWS RDS/Aurora. These instructions are for installing TeamForge in a three-server distributed setup with TeamForge and EventQ on two separate servers. All database services are hosted on a third server, which is an external PostgreSQL server not directly managed by TeamForge.

✓ You can install TeamForge on both RHEL/CentOS 7.4 and 6.9.

✓ In this distributed setup, TeamForge services are distributed across three servers, server-01 through server-03 as illustrated in the following table. It is assumed that server-03 is an externally managed PostgreSQL server.

server-01	**server-02**	**server-03**
TeamForge Application Server	**EventQ Server**	**External Database Server**
ctfcore	eventq	gerrit-database
mail	mongodb	reviewboard-database
search	redis	binary-database
codesearch	rabbitmq	ctfcore-database
etl		ctfcore-datamart
gerrit		
reviewboard		

server-01	server-02	server-03
TeamForge Application Server	**EventQ Server**	**External Database Server**
reviewboard-adapter[1]		
subversion		
cvs		
binary		
cliserver		

Dos and Don’ts

Here’s a list of dos, don’ts and points to remember when you install or upgrade TeamForge. Show/Hide

Dos

- Understand TeamForge installation requirements and plan your installation or upgrade.
- Get your TeamForge license key and keep it handy.
- Verify your basic networking setup before installing or upgrading TeamForge. See Set Up Networking for TeamForge.
- Look for new or modified `site-options.conf` tokens and update your `site-options.conf` file as required during the upgrade process. See Site Options Change Log.
- Set up a TeamForge Stage Server before you upgrade your Production Server.
- Stop TeamForge services on all servers in a distributed setup while upgrading to TeamForge 18.1.
- Uninstall hot fixes and add-ons, if any, before you start the TeamForge 18.1 upgrade procedure.

- TeamForge 18.1 supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.
  - JBOSS_JAVA_OPTS
  - PHOENIX_JAVA_OPTS
  - INTEGRATION_JAVA_OPTS
  - ETL_JAVA_OPTS
  - ELASTICSEARCH_JAVA_OPTS

  TeamForge provision fails on sites that use these options post TeamForge 18.1 upgrade.

Don'ts

- Do not customize your operating system installation. Select only the default packages list.
- While upgrading TeamForge, whether in place or on new hardware, always reuse the old `site-options.conf` file and make changes as necessary. **Do not** try to start with a new `site-options.conf` file. Reusing the old `site-options.conf` avoids many potential problems, particularly around the management of usernames and passwords.
- Do not manually modify TeamForge-managed site option tokens such as the `AUTO_DATA` token. See AUTO_DATA for more information.

- If you are creating symlinks, note that you must create symlinks only to the TeamForge data directory (`/opt/collabnet/teamforge/var`). You should not create symlinks to TeamForge application directories (such as `/opt/collabnet`).

Points to Remember

- Installing or upgrading TeamForge needs root privileges. You must log on as root or use a root shell to install or upgrade TeamForge.
- SSL is enabled by default and a self-signed certificate is auto-generated. However, you can use a few `site-options.conf` tokens to adjust this behavior. To generate the SSL certificates, see Generate SSL Certificates.
- For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.
- If you have Git integration on a separate server, both TeamForge and Git servers must have their time and date synchronized.
- While you can run both EventQ and TeamForge on the same server, CollabNet recommends such an approach only for testing purposes. It’s always recommended to run EventQ on a separate server for optimal scalability.
- No backup is required for same hardware upgrades. However, you can create a backup as a measure of caution. See Back up and Restore TeamForge for more information.
- Always use compatible JDBC drivers meant for specific database versions. See JDBC Drivers Reference for more information. Also see: Why do ETL jobs fail post TeamForge upgrade?.
- You can run the initial load job any time after the installation of TeamForge. We recommend that you run it before you hand over the site to the users. For more information, see ETL Initial Load Jobs.
- SOAP50 APIs and event handlers are no longer supported in TeamForge 16.10 and later. Use the latest TeamForge SOAP/REST APIs.
- TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.

- Installing TeamForge with service-specific FQDNs (instead of machine-specific host/domain names) is highly recommended so that you will be able to change the system landscape at a later point in time without having any impact on the URLs (in other words, end users do not have to notice or change anything). For example, you can create FQDNs specifically for services such as Subversion, Git, mail, Codesearch and so on. For more information, see Service-specific FQDNs.
- All such service-specific FQDNs must belong to a single sub domain and it is recommended to create a new sub domain for TeamForge.
- If you are using service-specific FQDNs
    - A wildcard SSL cert is required. SNI SSL cert cannot be used.
    - When SSL is enabled and no custom SSL certificates are provided, a self-signed wildcard cert is generated for the sub domain.
    - When SSL is enabled and a custom SSL certificate is provided, the CN of the certificate is verified to be a wildcard CN.
- You cannot have a separate PUBLIC_FQDN for EventQ.
- The ability to run separate PostgreSQL instances for TeamForge database and datamart on the same server is being deprecated in TeamForge 17.11. If you have TeamForge database and datamart on separate PostgreSQL instances on the same server and if you are upgrading on a new hardware, you must Create a Single Cluster for Both Database and Datamart while upgrading to TeamForge 17.11 or later.
- While upgrading TeamForge-Git integration servers, it is important that Git master and slave servers are upgraded to the same version of TeamForge-Git integration. On sites with Git Replica Servers, you must upgrade the Git Replica Servers first and then upgrade the master Git servers.

Prepare the Servers for TeamForge Installation (server-01 and server-02)

1. Install RHEL/CentOS 7.4 and log on as root.

   ✓ The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux.

   ✓ See the RHEL 7.4 Installation Guide for help.

2. Check your networking setup. See Set up Networking for more information.

3. Configure the TeamForge installation repository

TeamForge Installation Repository Configuration for Sites with Internet Access

1. Contact the CollabNet Support and download the TeamForge 18.1 installation repository package to `/tmp`.
2. Install the repository package.
   ```
   yum install -y /tmp/collabnet-teamforge-repo-18.1-0-noarch.rpm
   ```
3. Refresh your repository cache.
   ```
   yum clean all
   ```

TeamForge Installation Repository Configuration for Sites without Internet Access

1. Contact the CollabNet Support to get the auxiliary installer package for TeamForge 18.1 disconnected installation and save it in `/tmp`.
   - RHEL/CentOS 7.4 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel7.x86_64.rpm`
   - In addition to the above CentOS 7.4 64 bit RPM package, you must get the following CentOS 7.4 compatibility RPM, which is required for TeamForge 18.1 disconnected media installation on CentOS 7.4 profile: `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm`.
2. Unpack the disconnected installation package.
   ```
   rpm -ivh <package-name>
   ```
3. Unpack the `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm` package if you are installing TeamForge 18.1 on CentOS 7.4.
   ```
   rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
   ```
4. If not mounted already, mount the RHEL/CentOS installation DVD.

   The DVD contains the necessary software and utilities required for installing TeamForge without internet access. In the following commands, replace “cdrom” with the identifier for your server’s CD/DVD drive, if necessary.

   ```
   cd /media/
   mkdir cdrom
   mount /dev/cdrom ./cdrom/
   ```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

5. Create a yum configuration file that points to the RHEL/CentOS installation DVD.
   ```
   vi /etc/yum.repos.d/cdrom.repo
   ```

   Here's a sample yum configuration file.

   ```
   [RHEL-CDROM]
   name=RHEL CDRom
   baseurl=file:///media/cdrom/Server/
   gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
   enabled=1
   gpgcheck=0
   ```
6. Verify your yum configuration files.
   ```
   yum list httpd
   yum list apr
   ```

Install TeamForge Services

1. Install TeamForge and Review Board services on the TeamForge Application Server (server-01)
   ```
   yum install teamforge
   ```
2. Install EventQ services on the EventQ Server (server-02)
   ```
   yum install teamforge-eventq
   ```

Prepare the External Database Server for TeamForge Installation

1. Log on to the Database Server and create the TeamForge database, datamart, Gerrit database, Binary database and Review Board database. Note down the following credentials that are required to set up the TeamForge `site-options.conf` tokens later in the process.

   - Database name (DATABASE_NAME)
   - Database username (DATABASE_USERNAME)
   - Database password (DATABASE_PASSWORD)
   - Database read-only username (DATABASE_READ_ONLY_USER)
   - Database read-only password (DATABASE_READ_ONLY_PASSWORD)
   - Reports database name (REPORTS_DATABASE_NAME)

- Reports database username (REPORTS_DATABASE_USERNAME)
- Reports database password (REPORTS_DATABASE_PASSWORD)
- Reports database read-only username (REPORTS_DATABASE_READ_ONLY_USER)
- Reports database read-only password (REPORTS_DATABASE_READ_ONLY_PASSWORD)
- Gerrit database password (GERRIT_DATABASE_PASSWORD)
- IAF database name (IAF_DBNAME)
- IAF database username (IAF_DBUSER)
- IAF database password (IAF_DBPASS)
- Review Board database name (REVIEWBOARD_DATABASE_NAME)
- Review Board database username (REVIEWBOARD_DATABASE_USER)
- Review Board database password (REVIEWBOARD_DATABASE_PASSWORD)

2. Create users and grant access rights.
   - **Access rights for read-only users**: LOGIN,NOCREATEDB,NOCREATEROLE,NOSUPERUSER
   - **Access rights for other users**: LOGIN,CREATEDB,NOCREATEROLE,NOSUPERUSER
3. Configure the TeamForge installation repository

TeamForge Installation Repository Configuration for Sites with Internet Access

1. Contact the CollabNet Support and download the TeamForge 18.1 installation repository package to `/tmp`.
2. Install the repository package.
   ```
   yum install -y /tmp/collabnet-teamforge-repo-18.1-0-noarch.rpm
   ```
3. Refresh your repository cache.
   ```
   yum clean all
   ```

TeamForge Installation Repository Configuration for Sites without Internet Access

1. Contact the CollabNet Support to get the auxiliary installer package for TeamForge 18.1 disconnected installation and save it in `/tmp`.
   - RHEL/CentOS 7.4 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel7.x86_64.rpm`
   - In addition to the above CentOS 7.4 64 bit RPM package, you must get the following CentOS 7.4 compatibility RPM, which is required for TeamForge 18.1 disconnected media

installation on CentOS 7.4 profile: `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm`.

2. Unpack the disconnected installation package.
   ```
   rpm -ivh <package-name>
   ```
3. Unpack the `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm` package if you are installing TeamForge 18.1 on CentOS 7.4.
   ```
   rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
   ```
4. If not mounted already, mount the RHEL/CentOS installation DVD.

   The DVD contains the necessary software and utilities required for installing TeamForge without internet access. In the following commands, replace “cdrom” with the identifier for your server’s CD/DVD drive, if necessary.

   ```
   cd /media/
   mkdir cdrom
   mount /dev/cdrom ./cdrom/
   ```

   If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

5. Create a yum configuration file that points to the RHEL/CentOS installation DVD.
   ```
   vi /etc/yum.repos.d/cdrom.repo
   ```

   Here’s a sample yum configuration file.

   ```
   [RHEL-CDROM]
   name=RHEL CDRom
   baseurl=file:///media/cdrom/Server/
   gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
   enabled=1
   gpgcheck=0
   ```
6. Verify your yum configuration files.
   ```
   yum list httpd
   yum list apr
   ```

4. Install TeamForge database services on the External PostgreSQL Server (server-03)
   ```
   yum install teamforge
   ```

Set up Your Site's Master Configuration File

1. Do this on the TeamForge Application Server (server-01).
   `vi /opt/collabnet/teamforge/etc/site-options.conf`

host:SERVICES Token

```
server-01:SERVICES=ctfcore mail etl search subversion cvs codesearch cliserver gerrit binary reviewboard reviewboard-adapter
server-02:SERVICES=eventq mongodb redis rabbitmq
server-03:SERVICES=tfcore-database ctfcore-datamart gerrit-database  binary-database reviewboard-database
```

host:PUBLIC_FQDN Token

```
server-01:PUBLIC_FQDN=my.app.domain.com
```

> **NOTE:** You cannot have a separate PUBLIC_FQDN for EventQ.

Set up the Following Site Option Tokens

- DATABASE_NAME=
- DATABASE_USERNAME=
- DATABASE_PASSWORD=
- DATABASE_READ_ONLY_USER=
- DATABASE_READ_ONLY_PASSWORD=
- REPORTS_DATABASE_NAME=
- REPORTS_DATABASE_USERNAME=
- REPORTS_DATABASE_PASSWORD=
- REPORTS_DATABASE_READ_ONLY_USER=
- REPORTS_DATABASE_READ_ONLY_PASSWORD=
- GERRIT_DATABASE_PASSWORD=
- IAF_DBNAME=
- IAF_DBUSER=
- IAF_DBPASS=

- REVIEWBOARD_DATABASE_NAME=
- REVIEWBOARD_DATABASE_USER=
- REVIEWBOARD_DATABASE_PASSWORD=

Save the `site-options.conf` file.

For further customization of your site configuration (SSL settings, password policy settings, PostgreSQL settings, LDAP settings and so on): Site options configuration contd…

SSL Tokens

SSL is enabled by default and a self-signed certificate is auto-generated. Use the following tokens to adjust this behavior.

```
SSL_CERT_FILE=
SSL_KEY_FILE=
SSL_CHAIN_FILE=
```

- To generate the SSL certificates, see Generate SSL certificates.
- Have the custom SSL certificate and private key for custom SSL certificate in place and provide their absolute paths in these tokens. SSL_CHAIN_FILE (intermediate certificate) is optional.
- You can also encrypt the data traffic between the application and database servers and between the ETL and datamart servers in a distributed setup. Use the [DATABASE_SSL] [siteoptiontokens.html#DATABASE_SSL] token to do that. See Encrypt Database Network Traffic.

Password Tokens

- TeamForge 7.1 and later support automatic password creation. See AUTO_DATA for more information.
- Set the REQUIRE_PASSWORD_SECURITY token to `true` to enforce password security policy for the site.

  If the token REQUIRE_PASSWORD_SECURITY is enabled, then set a value for the token, PASSWORD_CONTROL_EFFECTIVE_DATE.

  > **WARNING:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the `PASSWORD_CONTROL_EFFECTIVE_DATE` token. If a date is not set, the PCK disables,

> deletes or expires user accounts immediately. See PASSWORD_CONTROL_EFFECTIVE_DATE for more information.

You can also set the following tokens to enforce a more stricter password policy:

- MINIMUM_PASSWORD_LENGTH
- MAX_PASSWORD_LENGTH
- PASSWORD_REQUIRES_NUMBER
- PASSWORD_REQUIRES_NON_ALPHANUM
- PASSWORD_REQUIRES_MIXED_CASE
- REQUIRE_PASSWORD_SECURITY
- LOGIN_ATTEMPT_LOCK
- PASSWORD_HISTORY_AGE
- ALLOW_PASSWORD_DICTIONARY_WORD

- If the token REQUIRE_RANDOM_ADMIN_PASSWORD is already set to `true`, then set the token ADMIN_EMAIL with a valid email address.
  ```
  ADMIN_EMAIL=root@{__APPLICATION_HOST__}
  ```
- If you have LDAP set up for external authentication, you must set the REQUIRE_USER_PASSWORD_CHANGE site options token to `false`.

Prevent Cross-site Scripting

An attacker could potentially upload an HTML page to TeamForge that contains active code, such as JavaScript. This active code would then be executed by clients' browsers when they view the page, which can harm the system.

To prevent an attack of this sort, you can specify whether or not HTML code is displayed in TeamForge. This flag applies to all documents, tracker, task, and forum attachments, and files in the file release system.

Set the SAFE_DOWNLOAD_MODE token according to your requirements. For more information, see SAFE_DOWNLOAD_MODE.

PostgreSQL Tokens and Settings

Make sure the PostgreSQL tokens in the `site-options.conf` file are set as recommended in the following topic: What are the right PostgreSQL settings for my site?

Save the `site-options.conf` file.

2. Copy the `/opt/collabnet/teamforge/etc/site-options.conf` file from the TeamForge Application Server to the `/opt/collabnet/teamforge/etc/` directory of all other servers.

Provision Services on All the Servers

1. Provision services.
   `teamforge provision`
   TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.

✓ You must provision services in a particluar sequence. Usually you start with the Database Server, followed by the Application Server and then by other servers such as the EventQ server.

✓ The TeamForge installer derives this sequence from your `site-options.conf` file and shows you the order of provisioning servers when you try to provision one of the distributed servers. Follow the exact sequence as instructed.

1. Provision the Database Server (server-03)
2. Provision the Application Server (server-01)
3. Provision the EventQ Server (server-02)

Verify TeamForge Installation

1. Verify TeamForge installation.
   1. Reboot the server and make sure all services come up automatically at startup.
   2. Log on to the TeamForge web application using the default Admin credentials.
      - Username: `admin`
      - Password: `admin`
   3. Create a sample project. See Create a TeamForge Project.
   4. Write a welcome message to your site's users. See Create a Site-wide Broadcast.

Post Install Tasks

- Supply Your TeamForge License Key
- Run TeamForge in SELinux enabled Mode
- Install EventQ Integration Adapters

- Users are not getting email notifications for review requests and reviews. What should I do?

1. `reviewboard-adapter` must always be installed on the TeamForge Application Server. □

You can upgrade TeamForge on new hardware with all services on a single server.

In this single server setup, the following TeamForge services run on the TeamForge Application Server (server-01).

- TeamForge Application Server (ctfcore)
- Database Server (ctfcore-database and ctfcore-datamart)
- Codesearch Server (codesearch)
- Mail Server (mail)
- ETL Server (etl)
- Git Integration Server (gerrit and gerrit-database)
- SCM Integration Server (subversion and cvs)
- Search Server (search).
- TeamForge EventQ Server (eventq, mongodb, redis and rabbitmq)
- TeamForge CLI Server (cliserver)
- Review Board (reviewboard, reviewboard-database, reviewboard-adapter)
- CLI Server (cliserver)

Dos and Don'ts

Here's a list of dos, don'ts and points to remember when you install or upgrade TeamForge. Show/Hide

Dos

- Understand TeamForge installation requirements and plan your installation or upgrade.
- Get your TeamForge license key and keep it handy.
- Verify your basic networking setup before installing or upgrading TeamForge. See Set Up Networking for TeamForge.
- Look for new or modified `site-options.conf` tokens and update your `site-options.conf` file as required during the upgrade process. See Site Options Change Log.
- Set up a TeamForge Stage Server before you upgrade your Production Server.
- Stop TeamForge services on all servers in a distributed setup while upgrading to TeamForge 18.1.
- Uninstall hot fixes and add-ons, if any, before you start the TeamForge 18.1 upgrade procedure.

- TeamForge 18.1 supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.
  - JBOSS_JAVA_OPTS
  - PHOENIX_JAVA_OPTS

    - INTEGRATION_JAVA_OPTS
    - ETL_JAVA_OPTS
    - ELASTICSEARCH_JAVA_OPTS

  TeamForge provision fails on sites that use these options post TeamForge 18.1 upgrade.

Don'ts

- Do not customize your operating system installation. Select only the default packages list.
- While upgrading TeamForge, whether in place or on new hardware, always reuse the old `site-options.conf` file and make changes as necessary. **Do not** try to start with a new `site-options.conf` file. Reusing the old `site-options.conf` avoids many potential problems, particularly around the management of usernames and passwords.
- Do not manually modify TeamForge-managed site option tokens such as the `AUTO_DATA` token. See AUTO_DATA for more information.

- If you are creating symlinks, note that you must create symlinks only to the TeamForge data directory (`/opt/collabnet/teamforge/var`). You should not create symlinks to TeamForge application directories (such as `/opt/collabnet`).

Points to Remember

- Installing or upgrading TeamForge needs root privileges. You must log on as root or use a root shell to install or upgrade TeamForge.
- SSL is enabled by default and a self-signed certificate is auto-generated. However, you can use a few `site-options.conf` tokens to adjust this behavior. To generate the SSL certificates, see Generate SSL Certificates.
- For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.
- If you have Git integration on a separate server, both TeamForge and Git servers must have their time and date synchronized.
- While you can run both EventQ and TeamForge on the same server, CollabNet recommends such an approach only for testing purposes. It’s always recommended to run EventQ on a separate server for optimal scalability.
- No backup is required for same hardware upgrades. However, you can create a backup as a measure of caution. See Back up and Restore TeamForge for more information.
- Always use compatible JDBC drivers meant for specific database versions. See JDBC Drivers Reference for more information. Also see: Why do ETL jobs fail post TeamForge upgrade?.

- You can run the initial load job any time after the installation of TeamForge. We recommend that you run it before you hand over the site to the users. For more information, see ETL Initial Load Jobs.
- SOAP50 APIs and event handlers are no longer supported in TeamForge 16.10 and later. Use the latest TeamForge SOAP/REST APIs.
- TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.
- Installing TeamForge with service-specific FQDNs (instead of machine-specific host/domain names) is highly recommended so that you will be able to change the system landscape at a later point in time without having any impact on the URLs (in other words, end users do not have to notice or change anything). For example, you can create FQDNs specifically for services such as Subversion, Git, mail, Codesearch and so on. For more information, see Service-specific FQDNs.
- All such service-specific FQDNs must belong to a single sub domain and it is recommended to create a new sub domain for TeamForge.
- If you are using service-specific FQDNs
    - A wildcard SSL cert is required. SNI SSL cert cannot be used.
    - When SSL is enabled and no custom SSL certificates are provided, a self-signed wildcard cert is generated for the sub domain.
    - When SSL is enabled and a custom SSL certificate is provided, the CN of the certificate is verified to be a wildcard CN.
- You cannot have a separate PUBLIC_FQDN for EventQ.
- The ability to run separate PostgreSQL instances for TeamForge database and datamart on the same server is being deprecated in TeamForge 17.11. If you have TeamForge database and datamart on separate PostgreSQL instances on the same server and if you are upgrading on a new hardware, you must Create a Single Cluster for Both Database and Datamart while upgrading to TeamForge 17.11 or later.
- While upgrading TeamForge-Git integration servers, it is important that Git master and slave servers are upgraded to the same version of TeamForge-Git integration. On sites with Git Replica Servers, you must upgrade the Git Replica Servers first and then upgrade the master Git servers.

Prepare the New TeamForge Application Server (server-01)

1. Install RHEL/CentOS 7.4 and log on as root.

    ✓ The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux.

    ✓ See the RHEL 7.4 Installation Guide for help.

2. Check your networking setup. See Set up Networking for more information.

3. Configure the TeamForge installation repository.

TeamForge Installation Repository Configuration for Sites with Internet Access

1. Contact the CollabNet Support and download the TeamForge 18.1 installation repository package to `/tmp`.
2. Install the repository package.
   ```
   yum install -y /tmp/collabnet-teamforge-repo-18.1-0-noarch.rpm
   ```
3. Refresh your repository cache.
   ```
   yum clean all
   ```

TeamForge Installation Repository Configuration for Sites without Internet Access

1. Contact the CollabNet Support to get the auxiliary installer package for TeamForge 18.1 disconnected installation and save it in `/tmp`.
   - RHEL/CentOS 6.9 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel6.x86_64.rpm`
   - RHEL/CentOS 7.4 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel7.x86_64.rpm`
   - In addition to the above CentOS 7.4 64 bit RPM package, you must get the following CentOS 7.4 compatibility RPM, which is required for TeamForge 18.1 disconnected media installation on CentOS 7.4 profile: `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm`.
2. Unpack the disconnected installation package.
   ```
   rpm -Uvh <package-name>
   ```
3. Unpack the `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm` package if you are installing TeamForge 18.1 on CentOS 7.4.
   ```
   rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
   ```
4. If not mounted already, mount the RHEL/CentOS installation DVD.

The DVD contains the necessary software and utilities required for installing TeamForge without internet access. In the following commands, replace “cdrom” with the identifier for your server’s CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

5. Create a yum configuration file that points to the RHEL/CentOS installation DVD.
   ```
   vi /etc/yum.repos.d/cdrom.repo
   ```

   Here’s a sample yum configuration file.

   ```
   [RHEL-CDROM]
   name=RHEL CDRom
   baseurl=file:///media/cdrom/Server/
   gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
   enabled=1
   gpgcheck=0
   ```
6. Verify your yum configuration files.
   ```
   yum list httpd
   yum list apr
   ```

Install the TeamForge Services

1. Install TeamForge.
   ```
   yum install teamforge
   ```
2. Install the PostgreSQL 9.3 packages. Do this if and only if you are upgrading from TeamForge 17.4 or earlier versions.
   ```
   yum install postgresql93-*
   ```

Back up and Restore TeamForge Database, Data Directories and site-options.conf

See Back up and Restore TeamForge Database, Data Directories and site-options.conf

Back up and Restore the Review Board Database and Data Directories

See Back up and Restore Review Board Database and Data Directories

Configure the New TeamForge Application Server (server-01)

Log on to the TeamForge Application Server (server-01), set up the `site-options.conf` file, and provision the services.

1. Copy the `site-options.conf` file to the TeamForge installer directory.
   ```
   cp /tmp/site-options.conf /opt/collabnet/teamforge/etc/
   ```
2. Set up your site's master configuration file.

   > **IMPORTANT:** See Site options change log for a list of site option changes. While upgrading to a latest TeamForge release, make sure that obsolete site option tokens, if any, are removed from the `site-options.conf` file of the TeamForge version you are upgrading to.

   ```
   vi /opt/collabnet/teamforge/etc/site-options.conf
   ```

   ## host:SERVICES Token

   ```
   server-01:SERVICES = ctfcore ctfcore-database ctfcore-datamart mail etl se
   arch codesearch subversion cvs eventq redis mongodb rabbitmq cliserver ger
   rit gerrit-database binary binary-database reviewboard reviewboard-databas
   e reviewboard-adapter
   ```

   ## host:PUBLIC_FQDN Token

   ```
   server-01:PUBLIC_FQDN = my.app.domain.com
   ```

   > **NOTE:** You cannot have a separate PUBLIC_FQDN for EventQ.

MONGODB_APP_DATABASE_NAME

Set the `MONGODB_APP_DATABASE_NAME` token with EventQ's database name in the `site-options.conf` file. Do this if and only if you are upgrading from TeamForge 17.1 or earlier to TeamForge 17.4 or later.

```
MONGODB_APP_DATABASE_NAME=orchestrate
```

Save the `site-options.conf` file.

For further customization of your site configuration (SSL settings, password policy settings, PostgreSQL settings, LDAP settings and so on): Site options configuration contd…

SSL Tokens

SSL is enabled by default and a self-signed certificate is auto-generated. Use the following tokens to adjust this behavior.

```
SSL_CERT_FILE=
SSL_KEY_FILE=
SSL_CHAIN_FILE=
```

- To generate the SSL certificates, see Generate SSL certificates.
- Have the custom SSL certificate and private key for custom SSL certificate in place and provide their absolute paths in these tokens. SSL_CHAIN_FILE (intermediate certificate) is optional.
- You can also encrypt the data traffic between the application and database servers and between the ETL and datamart servers in a distributed setup. Use the [DATABASE_SSL] [siteoptiontokens.html#DATABASE_SSL] token to do that. See Encrypt Database Network Traffic.

Password Tokens

- TeamForge 7.1 and later support automatic password creation. See AUTO_DATA for more information.
- Set the REQUIRE_PASSWORD_SECURITY token to `true` to enforce password security policy for the site.

  If the token REQUIRE_PASSWORD_SECURITY is enabled, then set a value for the token, PASSWORD_CONTROL_EFFECTIVE_DATE.

WARNING: The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the PASSWORD_CONTROL_EFFECTIVE_DATE token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See PASSWORD_CONTROL_EFFECTIVE_DATE for more information.

You can also set the following tokens to enforce a more stricter password policy:

  - MINIMUM_PASSWORD_LENGTH
  - MAX_PASSWORD_LENGTH
  - PASSWORD_REQUIRES_NUMBER
  - PASSWORD_REQUIRES_NON_ALPHANUM
  - PASSWORD_REQUIRES_MIXED_CASE
  - REQUIRE_PASSWORD_SECURITY
  - LOGIN_ATTEMPT_LOCK
  - PASSWORD_HISTORY_AGE
  - ALLOW_PASSWORD_DICTIONARY_WORD
- If the token REQUIRE_RANDOM_ADMIN_PASSWORD is already set to `true`, then set the token ADMIN_EMAIL with a valid email address.
  `ADMIN_EMAIL=root@{__APPLICATION_HOST__}`
- If you have LDAP set up for external authentication, you must set the REQUIRE_USER_PASSWORD_CHANGE site options token to `false`.
- Verify and update the list of non-expiring TeamForge user accounts (password never expires).
  `USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin`

Prevent Cross-site Scripting

An attacker could potentially upload an HTML page to TeamForge that contains active code, such as JavaScript. This active code would then be executed by clients' browsers when they view the page, which can harm the system.

To prevent an attack of this sort, you can specify whether or not HTML code is displayed in TeamForge. This flag applies to all documents, tracker, task, and forum attachments, and files in the file release system.

Set the SAFE_DOWNLOAD_MODE token according to your requirements. For more information, see SAFE_DOWNLOAD_MODE.

PostgreSQL Tokens and Settings

Make sure the PostgreSQL tokens in the `site-options.conf` file are set as recommended in the following topic: What are the right PostgreSQL settings for my site?

JAVA_OPTS

Configure the JBOSS_JAVA_OPTS `site-options.conf` token. See JBOSS_JAVA_OPTS.

> **NOTE:** All JVM parameters but `-Xms1024m` and `-Xmx2048m` have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as `-XX:MaxMetaspaceSize=512m` `-XX:ReservedCodeCacheSize=128M` `-server` `-XX:+HeapDumpOnOutOfMemoryError` `-Djsse.enableSNIExtension=false` `-Dsun.rmi.dgc.client.gcInterval=600000` `-Dsun.rmi.dgc.server.gcInterval=600000`) in the site-options.conf file.

TeamForge 18.1 (and later) supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.

- JBOSS_JAVA_OPTS
- PHOENIX_JAVA_OPTS
- INTEGRATION_JAVA_OPTS
- ETL_JAVA_OPTS
- ELASTICSEARCH_JAVA_OPTS

TeamForge provision fails on sites that use these options post upgrade to TeamForge 18.1.

Save the `site-options.conf` file.

Generate the License Key

As you are upgrading on new hardware, contact CollabNet Support, generate the license key for the new server (IP address) and use it to replace `/opt/collabnet/teamforge/var/etc/sflicense.txt`.

If you have the TeamForge database and datamart on two separate ports on the same server, see Create a Single Cluster for Both Database and Datamart.

Provision Services

1. Do this if and only if you have EventQ integration and are upgrading from TeamForge 17.1 or earlier to TeamForge 18.1.

   Copy the `/opt/collabnet/rabbitmq/var` and `/opt/collabnet/mongodb/data` directories to `/opt/collabnet/teamforge/var/rabbitmq` and `/opt/collabnet/teamforge/var/mongodb` directories respectively.

   ```
   cp -R /opt/collabnet/rabbitmq/var /opt/collabnet/teamforge/var/rabbitmq
   cp -R /opt/collabnet/mongodb/data /opt/collabnet/teamforge/var/mongodb
   ```
2. Provision services.
   ```
   teamforge provision
   ```
   TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.

Finishing Tasks

1. Run the `/var/lib/pgsql/analyze_new_cluster.sh` script. This is required if and only if you are upgrading from TeamForge 17.1 (or earlier) to TeamForge 17.8 (or later).
   ```
   su - postgres -c "/var/lib/pgsql/analyze_new_cluster.sh"
   ```
2. If you have CVS integrations, synchronize permissions post upgrade. See Synchronize TeamForge Source Control Integrations.
3. Verify TeamForge upgrade.
   1. Reboot the server and make sure all services come up automatically at startup.
   2. Log on to the TeamForge web application using the default Admin credentials.
      - Username: `admin`
      - Password: `admin`
   3. If your site has custom branding, verify that your branding changes still work as intended. See Customize TeamForge.
   4. Let your site's users know they've been upgraded. See Create a Site-wide Broadcast.
4. Remove the backup files, if any, after the TeamForge site is up and running as expected. Remove the repository and the file system backup from the `/tmp/backup_dir` directory.

Post Upgrade Tasks

- Run TeamForge in SELinux enabled Mode

- Add EventQ to Existing Projects
- Users are not getting email notifications for review requests and reviews. What should I do?

You can upgrade TeamForge on new hardware in a distributed multi-host setup.

In this distributed setup, TeamForge services are distributed across multiple servers as illustrated in the following table.

server-01	server-02	server-03	server-04	server-05	server-06
TeamForge Application Server	**TeamForge Database Server**	**EventQ Server**	**Review Board Server**	**SCM Server**	**Code Search Server**
ctfcore	ctfcore-database	eventq	reviewboard	subversion	codesearch
mail	ctfcore-datamart	rabbitmq		cvs	
etl	gerrit-database	mongodb		gerrit	
search	binary-database	redis			
reviewboard-adapter[1]	reviewboard-database				
binary					
cliserver					

Dos and Don’ts

Here’s a list of dos, don’ts and points to remember when you install or upgrade TeamForge. Show/Hide

Dos

- Understand TeamForge installation requirements and plan your installation or upgrade.
- Get your TeamForge license key and keep it handy.
- Verify your basic networking setup before installing or upgrading TeamForge. See Set Up Networking for TeamForge.
- Look for new or modified `site-options.conf` tokens and update your `site-options.conf` file as required during the upgrade process. See Site Options Change Log.
- Set up a TeamForge Stage Server before you upgrade your Production Server.
- Stop TeamForge services on all servers in a distributed setup while upgrading to TeamForge 18.1.
- Uninstall hot fixes and add-ons, if any, before you start the TeamForge 18.1 upgrade procedure.

- TeamForge 18.1 supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.
    - JBOSS_JAVA_OPTS
    - PHOENIX_JAVA_OPTS
    - INTEGRATION_JAVA_OPTS
    - ETL_JAVA_OPTS
    - ELASTICSEARCH_JAVA_OPTS

  TeamForge provision fails on sites that use these options post TeamForge 18.1 upgrade.

Don'ts

- Do not customize your operating system installation. Select only the default packages list.
- While upgrading TeamForge, whether in place or on new hardware, always reuse the old `site-options.conf` file and make changes as necessary. **Do not** try to start with a new `site-options.conf` file. Reusing the old `site-options.conf` avoids many potential problems, particularly around the management of usernames and passwords.
- Do not manually modify TeamForge-managed site option tokens such as the `AUTO_DATA` token. See AUTO_DATA for more information.

- If you are creating symlinks, note that you must create symlinks only to the TeamForge data directory (`/opt/collabnet/teamforge/var`). You should not create symlinks to TeamForge application directories (such as `/opt/collabnet`).

Points to Remember

- Installing or upgrading TeamForge needs root privileges. You must log on as root or use a root shell to install or upgrade TeamForge.
- SSL is enabled by default and a self-signed certificate is auto-generated. However, you can use a few `site-options.conf` tokens to adjust this behavior. To generate the SSL certificates, see Generate SSL Certificates.
- For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.
- If you have Git integration on a separate server, both TeamForge and Git servers must have their time and date synchronized.

- While you can run both EventQ and TeamForge on the same server, CollabNet recommends such an approach only for testing purposes. It's always recommended to run EventQ on a separate server for optimal scalability.
- No backup is required for same hardware upgrades. However, you can create a backup as a measure of caution. See Back up and Restore TeamForge for more information.
- Always use compatible JDBC drivers meant for specific database versions. See JDBC Drivers Reference for more information. Also see: Why do ETL jobs fail post TeamForge upgrade?.
- You can run the initial load job any time after the installation of TeamForge. We recommend that you run it before you hand over the site to the users. For more information, see ETL Initial Load Jobs.
- SOAP50 APIs and event handlers are no longer supported in TeamForge 16.10 and later. Use the latest TeamForge SOAP/REST APIs.
- TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.
- Installing TeamForge with service-specific FQDNs (instead of machine-specific host/domain names) is highly recommended so that you will be able to change the system landscape at a later point in time without having any impact on the URLs (in other words, end users do not have to notice or change anything). For example, you can create FQDNs specifically for services such as Subversion, Git, mail, Codesearch and so on. For more information, see Service-specific FQDNs.
- All such service-specific FQDNs must belong to a single sub domain and it is recommended to create a new sub domain for TeamForge.
- If you are using service-specific FQDNs
  - A wildcard SSL cert is required. SNI SSL cert cannot be used.
  - When SSL is enabled and no custom SSL certificates are provided, a self-signed wildcard cert is generated for the sub domain.
  - When SSL is enabled and a custom SSL certificate is provided, the CN of the certificate is verified to be a wildcard CN.
- You cannot have a separate PUBLIC_FQDN for EventQ.
- The ability to run separate PostgreSQL instances for TeamForge database and datamart on the same server is being deprecated in TeamForge 17.11. If you have TeamForge database and datamart on separate PostgreSQL instances on the same server and if you are upgrading on a new hardware, you must Create a Single Cluster for Both Database and Datamart while upgrading to TeamForge 17.11 or later.
- While upgrading TeamForge-Git integration servers, it is important that Git master and slave servers are upgraded to the same version of TeamForge-Git integration. On sites with Git Replica Servers, you must upgrade the Git Replica Servers first and then upgrade the master Git servers.

Prepare the New Servers for TeamForge Installation (server-01 through server-06)

1. Install RHEL/CentOS 7.4 and log on as root.

   ✓ The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux.

   ✓ See the RHEL 7.4 Installation Guide for help.

2. Check your networking setup. See Set up Networking for more information.

3. Configure the TeamForge installation repository.

   ## TeamForge Installation Repository Configuration for Sites with Internet Access

   1. Contact the CollabNet Support and download the TeamForge 18.1 installation repository package to `/tmp`.
   2. Install the repository package.
      ```
      yum install -y /tmp/collabnet-teamforge-repo-18.1-0-noarch.rpm
      ```
   3. Refresh your repository cache.
      ```
      yum clean all
      ```

   ## TeamForge Installation Repository Configuration for Sites without Internet Access

   1. Contact the CollabNet Support to get the auxiliary installer package for TeamForge 18.1 disconnected installation and save it in `/tmp`.
      - RHEL/CentOS 6.9 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel6.x86_64.rpm`
      - RHEL/CentOS 7.4 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel7.x86_64.rpm`
      - In addition to the above CentOS 7.4 64 bit RPM package, you must get the following CentOS 7.4 compatibility RPM, which is required for TeamForge 18.1 disconnected media

installation on CentOS 7.4 profile: `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm`.

2. Unpack the disconnected installation package.
   ```
   rpm -Uvh <package-name>
   ```
3. Unpack the `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm` package if you are installing TeamForge 18.1 on CentOS 7.4.
   ```
   rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
   ```
4. If not mounted already, mount the RHEL/CentOS installation DVD.

   The DVD contains the necessary software and utilities required for installing TeamForge without internet access. In the following commands, replace “cdrom” with the identifier for your server’s CD/DVD drive, if necessary.

   ```
   cd /media/
   mkdir cdrom
   mount /dev/cdrom ./cdrom/
   ```

   If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

5. Create a yum configuration file that points to the RHEL/CentOS installation DVD.
   ```
   vi /etc/yum.repos.d/cdrom.repo
   ```

   Here’s a sample yum configuration file.

   ```
   [RHEL-CDROM]
   name=RHEL CDRom
   baseurl=file:///media/cdrom/Server/
   gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
   enabled=1
   gpgcheck=0
   ```
6. Verify your yum configuration files.
   ```
   yum list httpd
   yum list apr
   ```

Install the TeamForge Services

1. Install TeamForge application services on the TeamForge Application Server (server-01).
   ```
   yum install teamforge
   ```

2. Install TeamForge database services on the TeamForge Database Server (server-02).
   `yum install teamforge`
3. Install EventQ services on the EventQ Server (server-03).
   `yum install teamforge-eventq`
4. Install Review Board services on the Review Board Server (server-04).
   `yum install teamforge`
5. Install SCM services on the SCM Server (server-05).
   `yum install teamforge-scm teamforge-git`
6. Install the Code Search service on the Code Search Server (server-06).
   `yum install teamforge-codesearch`

Back up and Restore TeamForge Database, Data Directories and site-options.conf

See Back up and Restore TeamForge Database, Data Directories and site-options.conf

Back up and Restore Review Board Database and Data Directories

See Back up and Restore Review Board Database and Data Directories

Set up the site-options.conf File

1. Log on to the new TeamForge Database Server (server-02) and copy the backed up `site-options.conf` file to the TeamForge installer directory.
   `cp /tmp/site-options.conf /opt/collabnet/teamforge/etc/`
2. Do this on the TeamForge Database Server (server-02).
   `vi /opt/collabnet/teamforge/etc/site-options.conf`

   ### host:SERVICES Token

   ```
   server-01:SERVICES=ctfcore search mail etl binary reviewboard-adapter clis
   erver
   server-02:SERVICES=ctfcore-database ctfcore-datamart gerrit-database binar
   y-database reviewboard-database
   server-03:SERVICES=eventq rabbitmq mongodb redis
   ```

```
server-04:SERVICES=reviewboard
server-05:SERVICES=subversion cvs gerrit
server-06:SERVICES=codesearch
```

host:PUBLIC_FQDN Token

```
server-01:PUBLIC_FQDN=my.app.domain.com
```

NOTE: You cannot have a separate PUBLIC_FQDN for EventQ.

MONGODB_APP_DATABASE_NAME

Set the `MONGODB_APP_DATABASE_NAME` token with EventQ's database name in the `site-options.conf` file. Do this if and only if you are upgrading from TeamForge 17.1 or earlier to TeamForge 17.4 or later.

```
MONGODB_APP_DATABASE_NAME=orchestrate
```

Save the `site-options.conf` file.

For further customization of your site configuration (SSL settings, password policy settings, PostgreSQL settings, LDAP settings and so on): Site options configuration contd…

SSL Tokens

SSL is enabled by default and a self-signed certificate is auto-generated. Use the following tokens to adjust this behavior.

```
SSL_CERT_FILE=
SSL_KEY_FILE=
SSL_CHAIN_FILE=
```

- To generate the SSL certificates, see Generate SSL certificates.
- Have the custom SSL certificate and private key for custom SSL certificate in place and provide their absolute paths in these tokens. SSL_CHAIN_FILE (intermediate certificate) is optional.
- You can also encrypt the data traffic between the application and database servers and between the ETL and datamart servers in a distributed setup. Use the [DATABASE_SSL] [siteoptiontokens.html#DATABASE_SSL] token to do that. See Encrypt Database Network Traffic.

Password Tokens

- TeamForge 7.1 and later support automatic password creation. See AUTO_DATA for more information.
- Set the REQUIRE_PASSWORD_SECURITY token to `true` to enforce password security policy for the site.

  If the token REQUIRE_PASSWORD_SECURITY is enabled, then set a value for the token, PASSWORD_CONTROL_EFFECTIVE_DATE.

  > **WARNING:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the `PASSWORD_CONTROL_EFFECTIVE_DATE` token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See PASSWORD_CONTROL_EFFECTIVE_DATE for more information.

  You can also set the following tokens to enforce a more stricter password policy:

  - MINIMUM_PASSWORD_LENGTH
  - MAX_PASSWORD_LENGTH
  - PASSWORD_REQUIRES_NUMBER
  - PASSWORD_REQUIRES_NON_ALPHANUM
  - PASSWORD_REQUIRES_MIXED_CASE
  - REQUIRE_PASSWORD_SECURITY
  - LOGIN_ATTEMPT_LOCK
  - PASSWORD_HISTORY_AGE
  - ALLOW_PASSWORD_DICTIONARY_WORD
- If the token REQUIRE_RANDOM_ADMIN_PASSWORD is already set to `true`, then set the token ADMIN_EMAIL with a valid email address.
  ```
  ADMIN_EMAIL=root@{__APPLICATION_HOST__}
  ```
- If you have LDAP set up for external authentication, you must set the REQUIRE_USER_PASSWORD_CHANGE site options token to `false`.
- Verify and update the list of non-expiring TeamForge user accounts (password never expires).
  ```
  USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin
  ```

Prevent Cross-site Scripting

An attacker could potentially upload an HTML page to TeamForge that contains active code, such as JavaScript. This active code would then be executed by clients' browsers when they view the page, which can harm the system.

To prevent an attack of this sort, you can specify whether or not HTML code is displayed in TeamForge. This flag applies to all documents, tracker, task, and forum attachments, and files in the file release system.

Set the SAFE_DOWNLOAD_MODE token according to your requirements. For more information, see SAFE_DOWNLOAD_MODE.

PostgreSQL Tokens and Settings

Make sure the PostgreSQL tokens in the `site-options.conf` file are set as recommended in the following topic: What are the right PostgreSQL settings for my site?

JAVA_OPTS

Configure the JBOSS_JAVA_OPTS `site-options.conf` token. See JBOSS_JAVA_OPTS.

> **NOTE:** All JVM parameters but `-Xms1024m` and `-Xmx2048m` have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as `-XX:MaxMetaspaceSize=512m` `-XX:ReservedCodeCacheSize=128M` `-server` `-XX:+HeapDumpOnOutOfMemoryError` `-Djsse.enableSNIExtension=false` `-Dsun.rmi.dgc.client.gcInterval=600000` `-Dsun.rmi.dgc.server.gcInterval=600000`) in the site-options.conf file.

TeamForge 18.1 (and later) supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.

- JBOSS_JAVA_OPTS
- PHOENIX_JAVA_OPTS
- INTEGRATION_JAVA_OPTS
- ETL_JAVA_OPTS

- ELASTICSEARCH_JAVA_OPTS

TeamForge provision fails on sites that use these options post upgrade to TeamForge 18.1.

Save the `site-options.conf` file.

3. Provision services.
   `teamforge provision`
   TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.
4. Copy the `/opt/collabnet/teamforge/etc/site-options.conf` file from the TeamForge Database Server (server-02) to the `/opt/collabnet/teamforge/etc/` directory of all other servers.

Provision Services on All the Servers

1. Do this if and only if you have EventQ integration and are upgrading from TeamForge 17.1 or earlier to TeamForge 18.1.

   Copy the `/opt/collabnet/rabbitmq/var` and `/opt/collabnet/mongodb/data` directories to `/opt/collabnet/teamforge/var/rabbitmq` and `/opt/collabnet/teamforge/var/mongodb` directories respectively.

   ```
   cp -R /opt/collabnet/rabbitmq/var /opt/collabnet/teamforge/var/rabbitmq
   cp -R /opt/collabnet/mongodb/data /opt/collabnet/teamforge/var/mongodb
   ```
2. Provision services.
   `teamforge provision`
   TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.

✓ You must provision services in a particluar sequence. Usually you start with the Database Server, followed by the Application Server and then by other servers such as SCM, Review Board, EventQ and Code Search servers.

✓ The TeamForge installer derives this sequence from your `site-options.conf` file and shows you the order of provisioning servers when you try to provision one of the distributed servers. Follow the exact sequence as instructed.

1. Provision the Database Server (server-02)
2. Provision the Application Server (server-01)

3. Provision the SCM server (server-05)
4. Provision the EventQ Server (server-03)
5. Provision the Review Board Server (server-04)
6. Provision the Code Search Server (server-06)

Reinitialize TeamForge

1. Reinitialize TeamForge on the Review Board Server.
   `teamforge reinitialize`
2. During `teamforge provision`, the `Register SCM integration` process fails on sites that use self-signed certificates. Perform these steps in such cases.
   1. Restart JBoss on the TeamForge Application Server.
      `teamforge restart -s jboss`
   2. Reinitialize TeamForge on the SCM Server.
      `teamforge reinitialize`

   Do you have Git and other SCM tools (SVN and CVS) on two separate servers?
   Git and other SCM tools (SVN and CVS) are typically installed on a separate server dedicated for SCM. However, if you have Git and SCM (SVN and CVS) on two separate servers, restart Jboss on the TeamForge Application Server and reinitialize TeamForge on the SCM Server (SVN and CVS) as discussed earlier. In addition, you must also restart TeamForge on the Git Server.

   Restart TeamForge on the Git Server: `teamforge restart`

Finishing Tasks

1. Run the `/var/lib/pgsql/analyze_new_cluster.sh` script. This is required if and only if you are upgrading from TeamForge 17.1 (or earlier) to TeamForge 17.8 (or later).
   `su - postgres -c "/var/lib/pgsql/analyze_new_cluster.sh"`
2. If you have CVS integrations, synchronize permissions post upgrade. See Synchronize TeamForge Source Control Integrations.
3. Verify TeamForge upgrade.
   1. Reboot the server and make sure all services come up automatically at startup.
   2. Log on to the TeamForge web application using the default Admin credentials.
      - Username: `admin`
      - Password: `admin`
   3. If your site has custom branding, verify that your branding changes still work as intended. See Customize TeamForge.
   4. Let your site's users know they've been upgraded. See Create a Site-wide Broadcast.

Post Upgrade Tasks

- Run TeamForge in SELinux `enabled` Mode
- Add EventQ to Existing Projects
- Users are not getting email notifications for review requests and reviews. What should I do?

1. `reviewboard-adapter` must always be installed on the TeamForge Application Server. ↩

You can upgrade TeamForge on the same hardware with all services on a single server.

In this single server setup, the following TeamForge services run on the TeamForge Application Server (server-01).

- TeamForge Application Server (ctfcore)
- Database Server (ctfcore-database and ctfcore-datamart)
- Codesearch Server (codesearch)
- Mail Server (mail)
- ETL Server (etl)
- Git Integration Server (gerrit and gerrit-database)
- SCM Integration Server (subversion and cvs)
- Search Server (search).
- TeamForge EventQ Server (eventq, mongodb, redis and rabbitmq)
- TeamForge CLI Server (cliserver)
- Review Board (reviewboard, reviewboard-database, reviewboard-adapter)
- CLI Server (cliserver)

Dos and Don’ts

Here’s a list of dos, don’ts and points to remember when you install or upgrade TeamForge. Show/Hide

Dos

- Understand TeamForge installation requirements and plan your installation or upgrade.
- Get your TeamForge license key and keep it handy.
- Verify your basic networking setup before installing or upgrading TeamForge. See Set Up Networking for TeamForge.
- Look for new or modified `site-options.conf` tokens and update your `site-options.conf` file as required during the upgrade process. See Site Options Change Log.
- Set up a TeamForge Stage Server before you upgrade your Production Server.

- Stop TeamForge services on all servers in a distributed setup while upgrading to TeamForge 18.1.
- Uninstall hot fixes and add-ons, if any, before you start the TeamForge 18.1 upgrade procedure.

- TeamForge 18.1 supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.

  - JBOSS_JAVA_OPTS
  - PHOENIX_JAVA_OPTS
  - INTEGRATION_JAVA_OPTS
  - ETL_JAVA_OPTS
  - ELASTICSEARCH_JAVA_OPTS

  TeamForge provision fails on sites that use these options post TeamForge 18.1 upgrade.

Don'ts

- Do not customize your operating system installation. Select only the default packages list.
- While upgrading TeamForge, whether in place or on new hardware, always reuse the old `site-options.conf` file and make changes as necessary. **Do not** try to start with a new `site-options.conf` file. Reusing the old `site-options.conf` avoids many potential problems, particularly around the management of usernames and passwords.
- Do not manually modify TeamForge-managed site option tokens such as the `AUTO_DATA` token. See AUTO_DATA for more information.

- If you are creating symlinks, note that you must create symlinks only to the TeamForge data directory (`/opt/collabnet/teamforge/var`). You should not create symlinks to TeamForge application directories (such as `/opt/collabnet`).

Points to Remember

- Installing or upgrading TeamForge needs root privileges. You must log on as root or use a root shell to install or upgrade TeamForge.
- SSL is enabled by default and a self-signed certificate is auto-generated. However, you can use a few `site-options.conf` tokens to adjust this behavior. To generate the SSL certificates, see Generate SSL Certificates.
- For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

- If you have Git integration on a separate server, both TeamForge and Git servers must have their time and date synchronized.
- While you can run both EventQ and TeamForge on the same server, CollabNet recommends such an approach only for testing purposes. It's always recommended to run EventQ on a separate server for optimal scalability.
- No backup is required for same hardware upgrades. However, you can create a backup as a measure of caution. See Back up and Restore TeamForge for more information.
- Always use compatible JDBC drivers meant for specific database versions. See JDBC Drivers Reference for more information. Also see: Why do ETL jobs fail post TeamForge upgrade?.
- You can run the initial load job any time after the installation of TeamForge. We recommend that you run it before you hand over the site to the users. For more information, see ETL Initial Load Jobs.
- SOAP50 APIs and event handlers are no longer supported in TeamForge 16.10 and later. Use the latest TeamForge SOAP/REST APIs.
- TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.
- Installing TeamForge with service-specific FQDNs (instead of machine-specific host/domain names) is highly recommended so that you will be able to change the system landscape at a later point in time without having any impact on the URLs (in other words, end users do not have to notice or change anything). For example, you can create FQDNs specifically for services such as Subversion, Git, mail, Codesearch and so on. For more information, see Service-specific FQDNs.
- All such service-specific FQDNs must belong to a single sub domain and it is recommended to create a new sub domain for TeamForge.
- If you are using service-specific FQDNs
  - A wildcard SSL cert is required. SNI SSL cert cannot be used.
  - When SSL is enabled and no custom SSL certificates are provided, a self-signed wildcard cert is generated for the sub domain.
  - When SSL is enabled and a custom SSL certificate is provided, the CN of the certificate is verified to be a wildcard CN.
- You cannot have a separate PUBLIC_FQDN for EventQ.
- The ability to run separate PostgreSQL instances for TeamForge database and datamart on the same server is being deprecated in TeamForge 17.11. If you have TeamForge database and datamart on separate PostgreSQL instances on the same server and if you are upgrading on a new hardware, you must Create a Single Cluster for Both Database and Datamart while upgrading to TeamForge 17.11 or later.
- While upgrading TeamForge-Git integration servers, it is important that Git master and slave servers are upgraded to the same version of TeamForge-Git integration. On sites with Git Replica Servers, you must upgrade the Git Replica Servers first and then upgrade the master Git servers.

✓ The following instructions are valid for both RHEL/CentOS 6.9/7.4 platforms. Specific steps, if applicable only for a particular RHEL/CentOS platform, are called out explicitly.

✓ No backup is required for same hardware upgrades. However, you can create a backup as a precaution. See Back up and Restore TeamForge Database, Data Directories and site-options.conf.

Uninstall Custom Event Handlers, Hot Fixes, Add-ons and Review Board

1. Log on to the TeamForge Application Server (server-01).
2. SOAP 50 is no longer supported. Back up all your custom event handlers and remove all the event handler JAR files before starting your TeamForge upgrade process.
   1. Go to **My Workspace** > **Admin**.
   2. Click **System Tools** from the **Projects** menu.
   3. Click **Customizations**.
   4. Select the custom event handler and click **Delete**.

   > **TIP:** Post upgrade, you can add custom event handlers again from the backup while making sure that you don't have SOAP50 (deprecated) library used.

3. Uninstall hotfixes and add-ons, if any, installed on your site.

yum upgrade

1. Stop TeamForge.

   - If you are upgrading from TeamForge 16.7 or earlier releases:
     `/etc/init.d/collabnet stop`
   - If you are upgrading from TeamForge 16.10, 17.1, or 17.4 releases:
     `/opt/collabnet/teamforge/bin/teamforge stop`
   - If you are upgrading from TeamForge 17.8 or later releases:
     `teamforge stop`

2. Stop EventQ.

   - If you are upgrading from TeamForge 16.3:
     `/etc/init.d/orchestrate stop`
   - If you are upgrading from TeamForge 16.7, 16.10, or 17.1 release:

```
/etc/init.d/eventq stop
/etc/init.d/collabnet-rabbitmq-server stop
/etc/init.d/collabnet-mongod stop
```

- If you are upgrading from TeamForge 17.4 release:
  `/opt/collabnet/teamforge/bin/teamforge stop`
- If you are upgrading from TeamForge 17.8 or later releases:
  `teamforge stop`

3. Upgrade the operating system packages.

   ```
   yum upgrade
   ```

Configure the TeamForge Installation Repository

1. Configure the TeamForge installation repository.

TeamForge Installation Repository Configuration for Sites with Internet Access

1. Contact the CollabNet Support and download the TeamForge 18.1 installation repository package to `/tmp`.
2. Install the repository package.
   `yum install -y /tmp/collabnet-teamforge-repo-18.1-0-noarch.rpm`
3. Refresh your repository cache.
   `yum clean all`

TeamForge Installation Repository Configuration for Sites without Internet Access

1. Contact the CollabNet Support to get the auxiliary installer package for TeamForge 18.1 disconnected installation and save it in `/tmp`.
   - RHEL/CentOS 6.9 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel6.x86_64.rpm`
   - RHEL/CentOS 7.4 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel7.x86_64.rpm`

- In addition to the above CentOS 7.4 64 bit RPM package, you must get the following CentOS 7.4 compatibility RPM, which is required for TeamForge 18.1 disconnected media installation on CentOS 7.4 profile: `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm`.

2. Unpack the disconnected installation package.

   ```
   rpm -Uvh <package-name>
   ```

3. Unpack the `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm` package if you are installing TeamForge 18.1 on CentOS 7.4.

   ```
   rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
   ```

4. If not mounted already, mount the RHEL/CentOS installation DVD.

   The DVD contains the necessary software and utilities required for installing TeamForge without internet access. In the following commands, replace “cdrom” with the identifier for your server’s CD/DVD drive, if necessary.

   ```
   cd /media/
   mkdir cdrom
   mount /dev/cdrom ./cdrom/
   ```

   If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

5. Create a yum configuration file that points to the RHEL/CentOS installation DVD.

   ```
   vi /etc/yum.repos.d/cdrom.repo
   ```

   Here’s a sample yum configuration file.

   ```
   [RHEL-CDROM]
   name=RHEL CDRom
   baseurl=file:///media/cdrom/Server/
   gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
   enabled=1
   gpgcheck=0
   ```

6. Verify your yum configuration files.

   ```
   yum list httpd
   yum list apr
   ```

Upgrade the TeamForge Services

1. Upgrade TeamForge.
   ```
   yum install teamforge
   ```
2. Run the following command to upgrade the Binary application packages. This is required if and only if you are upgrading from TeamForge 16.10 (or earlier) to TeamForge 17.8 (or later).
   ```
   yum install cn-binary
   ```

Set up the site-options.conf File

1. Set up the `site-options.conf` file.

> **IMPORTANT:** See Site options change log for a list of site option changes. While upgrading to a latest TeamForge release, make sure that obsolete site option tokens, if any, are removed from the `site-options.conf` file of the TeamForge version you are upgrading to.

```
vi /opt/collabnet/teamforge/etc/site-options.conf
```

host:SERVICES Token

```
server-01:SERVICES = ctfcore ctfcore-database ctfcore-datamart mail etl se
arch codesearch subversion cvs eventq redis mongodb rabbitmq cliserver ger
rit gerrit-database binary binary-database reviewboard reviewboard-databas
e reviewboard-adapter
```

host:PUBLIC_FQDN Token

```
server-01:PUBLIC_FQDN = my.app.domain.com
```

> **NOTE:** You cannot have a separate PUBLIC_FQDN for EventQ.

MONGODB_APP_DATABASE_NAME

Set the `MONGODB_APP_DATABASE_NAME` token with EventQ's database name in the `site-options.conf` file. Do this if and only if you are upgrading from TeamForge 17.1 or earlier to TeamForge 17.4 or later.

```
MONGODB_APP_DATABASE_NAME=orchestrate
```

Save the `site-options.conf` file.

For further customization of your site configuration (SSL settings, password policy settings, PostgreSQL settings, LDAP settings and so on): Site options configuration contd…

SSL Tokens

SSL is enabled by default and a self-signed certificate is auto-generated. Use the following tokens to adjust this behavior.

```
SSL_CERT_FILE=
SSL_KEY_FILE=
SSL_CHAIN_FILE=
```

- To generate the SSL certificates, see Generate SSL certificates.
- Have the custom SSL certificate and private key for custom SSL certificate in place and provide their absolute paths in these tokens. SSL_CHAIN_FILE (intermediate certificate) is optional.
- You can also encrypt the data traffic between the application and database servers and between the ETL and datamart servers in a distributed setup. Use the [DATABASE_SSL] [siteoptiontokens.html#DATABASE_SSL] token to do that. See Encrypt Database Network Traffic.

Password Tokens

- TeamForge 7.1 and later support automatic password creation. See AUTO_DATA for more information.

- Set the REQUIRE_PASSWORD_SECURITY token to `true` to enforce password security policy for the site.

  If the token REQUIRE_PASSWORD_SECURITY is enabled, then set a value for the token, PASSWORD_CONTROL_EFFECTIVE_DATE.

WARNING: The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the PASSWORD_CONTROL_EFFECTIVE_DATE token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See PASSWORD_CONTROL_EFFECTIVE_DATE for more information.

You can also set the following tokens to enforce a more stricter password policy:

- MINIMUM_PASSWORD_LENGTH
- MAX_PASSWORD_LENGTH
- PASSWORD_REQUIRES_NUMBER
- PASSWORD_REQUIRES_NON_ALPHANUM
- PASSWORD_REQUIRES_MIXED_CASE
- REQUIRE_PASSWORD_SECURITY
- LOGIN_ATTEMPT_LOCK
- PASSWORD_HISTORY_AGE
- ALLOW_PASSWORD_DICTIONARY_WORD

- If the token REQUIRE_RANDOM_ADMIN_PASSWORD is already set to `true`, then set the token ADMIN_EMAIL with a valid email address.
  `ADMIN_EMAIL=root@{__APPLICATION_HOST__}`
- If you have LDAP set up for external authentication, you must set the REQUIRE_USER_PASSWORD_CHANGE site options token to `false`.
- Verify and update the list of non-expiring TeamForge user accounts (password never expires).
  `USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin`

Prevent Cross-site Scripting

An attacker could potentially upload an HTML page to TeamForge that contains active code, such as JavaScript. This active code would then be executed by clients' browsers when they view the page, which can harm the system.

To prevent an attack of this sort, you can specify whether or not HTML code is displayed in TeamForge. This flag applies to all documents, tracker, task, and forum attachments, and files in the file release system.

Set the SAFE_DOWNLOAD_MODE token according to your requirements. For more information, see SAFE_DOWNLOAD_MODE.

PostgreSQL Tokens and Settings

Make sure the PostgreSQL tokens in the `site-options.conf` file are set as recommended in the following topic: What are the right PostgreSQL settings for my site?

JAVA_OPTS

Configure the JBOSS_JAVA_OPTS `site-options.conf` token. See JBOSS_JAVA_OPTS.

> **NOTE:** All JVM parameters but `-Xms1024m` and `-Xmx2048m` have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as `-XX:MaxMetaspaceSize=512m` `-XX:ReservedCodeCacheSize=128M` `-server` `-XX:+HeapDumpOnOutOfMemoryError` `-Djsse.enableSNIExtension=false` `-Dsun.rmi.dgc.client.gcInterval=600000` `-Dsun.rmi.dgc.server.gcInterval=600000`) in the site-options.conf file.

TeamForge 18.1 (and later) supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.

- JBOSS_JAVA_OPTS
- PHOENIX_JAVA_OPTS
- INTEGRATION_JAVA_OPTS
- ETL_JAVA_OPTS
- ELASTICSEARCH_JAVA_OPTS

TeamForge provision fails on sites that use these options post upgrade to TeamForge 18.1.

Save the `site-options.conf` file.

Provision Services

1. Do this if and only if you have EventQ integration and are upgrading from TeamForge 17.1 or earlier to TeamForge 18.1.

Copy the /opt/collabnet/rabbitmq/var and /opt/collabnet/mongodb/data directories to /opt/collabnet/teamforge/var/rabbitmq and /opt/collabnet/teamforge/var/mongodb directories respectively.

```
cp -R /opt/collabnet/rabbitmq/var /opt/collabnet/teamforge/var/rabbitmq
cp -R /opt/collabnet/mongodb/data /opt/collabnet/teamforge/var/mongodb
```

2. Provision services.

```
teamforge provision
```

TeamForge 18.1 installer expects the system locale to be LANG=en_US.UTF-8. TeamForge create runtime (teamforge provision) fails otherwise.

Finishing Tasks

1. Run the /var/lib/pgsql/analyze_new_cluster.sh script. This is required if and only if you are upgrading from TeamForge 17.1 (or earlier) to TeamForge 17.8 (or later).
   ```
   su - postgres -c "/var/lib/pgsql/analyze_new_cluster.sh"
   ```
2. If you have CVS integrations, synchronize permissions post upgrade. See Synchronize TeamForge Source Control Integrations.
3. Verify TeamForge upgrade.
   1. Reboot the server and make sure all services come up automatically at startup.
   2. Log on to the TeamForge web application using the default Admin credentials.
      - Username: admin
      - Password: admin
   3. If your site has custom branding, verify that your branding changes still work as intended. See Customize TeamForge.
   4. Let your site's users know they've been upgraded. See Create a Site-wide Broadcast.

Post Upgrade Tasks

- Run TeamForge in SELinux enabled Mode
- Add EventQ to Existing Projects
- Users are not getting email notifications for review requests and reviews. What should I do?

You can upgrade TeamForge on the same hardware in a distributed multi-host setup.

In this distributed setup, TeamForge services are distributed across multiple servers as illustrated in the following table.

server-01	server-02	server-03	server-04	server-05	server-06
TeamForge Application Server	**TeamForge Database Server**	**EventQ Server**	**Review Board Server**	**SCM Server**	**Code Search Server**
ctfcore	ctfcore-database	eventq	reviewboard	subversion	codesearch
mail	ctfcore-datamart	rabbitmq		cvs	
etl	gerrit-database	mongodb		gerrit	
search	binary-database	redis			
reviewboard-adapter[1]	reviewboard-database				
binary					
cliserver					

Dos and Don’ts

Here’s a list of dos, don’ts and points to remember when you install or upgrade TeamForge. Show/Hide

Dos

- Understand TeamForge installation requirements and plan your installation or upgrade.
- Get your TeamForge license key and keep it handy.
- Verify your basic networking setup before installing or upgrading TeamForge. See Set Up Networking for TeamForge.
- Look for new or modified `site-options.conf` tokens and update your `site-options.conf` file as required during the upgrade process. See Site Options Change Log.
- Set up a TeamForge Stage Server before you upgrade your Production Server.
- Stop TeamForge services on all servers in a distributed setup while upgrading to TeamForge 18.1.
- Uninstall hot fixes and add-ons, if any, before you start the TeamForge 18.1 upgrade procedure.

- TeamForge 18.1 supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.
  - JBOSS_JAVA_OPTS
  - PHOENIX_JAVA_OPTS
  - INTEGRATION_JAVA_OPTS
  - ETL_JAVA_OPTS
  - ELASTICSEARCH_JAVA_OPTS

TeamForge provision fails on sites that use these options post TeamForge 18.1 upgrade.

Don'ts

- Do not customize your operating system installation. Select only the default packages list.
- While upgrading TeamForge, whether in place or on new hardware, always reuse the old `site-options.conf` file and make changes as necessary. **Do not** try to start with a new `site-options.conf` file. Reusing the old `site-options.conf` avoids many potential problems, particularly around the management of usernames and passwords.
- Do not manually modify TeamForge-managed site option tokens such as the `AUTO_DATA` token. See AUTO_DATA for more information.

- If you are creating symlinks, note that you must create symlinks only to the TeamForge data directory (`/opt/collabnet/teamforge/var`). You should not create symlinks to TeamForge application directories (such as `/opt/collabnet`).

Points to Remember

- Installing or upgrading TeamForge needs root privileges. You must log on as root or use a root shell to install or upgrade TeamForge.
- SSL is enabled by default and a self-signed certificate is auto-generated. However, you can use a few `site-options.conf` tokens to adjust this behavior. To generate the SSL certificates, see Generate SSL Certificates.
- For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.
- If you have Git integration on a separate server, both TeamForge and Git servers must have their time and date synchronized.
- While you can run both EventQ and TeamForge on the same server, CollabNet recommends such an approach only for testing purposes. It’s always recommended to run EventQ on a separate server for optimal scalability.
- No backup is required for same hardware upgrades. However, you can create a backup as a measure of caution. See Back up and Restore TeamForge for more information.
- Always use compatible JDBC drivers meant for specific database versions. See JDBC Drivers Reference for more information. Also see: Why do ETL jobs fail post TeamForge upgrade?.
- You can run the initial load job any time after the installation of TeamForge. We recommend that you run it before you hand over the site to the users. For more information, see ETL Initial Load Jobs.
- SOAP50 APIs and event handlers are no longer supported in TeamForge 16.10 and later. Use the latest TeamForge SOAP/REST APIs.

- TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.
- Installing TeamForge with service-specific FQDNs (instead of machine-specific host/domain names) is highly recommended so that you will be able to change the system landscape at a later point in time without having any impact on the URLs (in other words, end users do not have to notice or change anything). For example, you can create FQDNs specifically for services such as Subversion, Git, mail, Codesearch and so on. For more information, see Service-specific FQDNs.
- All such service-specific FQDNs must belong to a single sub domain and it is recommended to create a new sub domain for TeamForge.
- If you are using service-specific FQDNs
  - A wildcard SSL cert is required. SNI SSL cert cannot be used.
  - When SSL is enabled and no custom SSL certificates are provided, a self-signed wildcard cert is generated for the sub domain.
  - When SSL is enabled and a custom SSL certificate is provided, the CN of the certificate is verified to be a wildcard CN.
- You cannot have a separate PUBLIC_FQDN for EventQ.
- The ability to run separate PostgreSQL instances for TeamForge database and datamart on the same server is being deprecated in TeamForge 17.11. If you have TeamForge database and datamart on separate PostgreSQL instances on the same server and if you are upgrading on a new hardware, you must Create a Single Cluster for Both Database and Datamart while upgrading to TeamForge 17.11 or later.
- While upgrading TeamForge-Git integration servers, it is important that Git master and slave servers are upgraded to the same version of TeamForge-Git integration. On sites with Git Replica Servers, you must upgrade the Git Replica Servers first and then upgrade the master Git servers.

✓ The following instructions are valid for both RHEL/CentOS 6.9/7.4 platforms. Specific steps, if applicable only for a particular RHEL/CentOS platform, are called out explicitly.

✓ No backup is required for same hardware upgrades. However, you can create a backup as a precaution. See Back up and Restore TeamForge Database, Data Directories and site-options.conf.

Uninstall Custom Event Handlers, Hot Fixes and Add-ons

Log on to the TeamForge Application Server.

1. SOAP 50 is no longer supported. Back up all your custom event handlers and remove all the event handler JAR files before starting your TeamForge upgrade process.
   1. Go to **My Workspace** > **Admin**.

2. Click **System Tools** from the **Projects** menu.
3. Click **Customizations**.
4. Select the custom event handler and click **Delete**.

> **TIP:** Post upgrade, you can add custom event handlers again from the backup while making sure that you don’t have SOAP50 (deprecated) library used.

2. Uninstall hotfixes and add-ons, if any, installed on your site.

yum upgrade

1. Stop TeamForge.

> **IMPORTANT:** Stop TeamForge on all the servers in a distributed setup.

- If you are upgarding from TeamForge 16.7 or earlier releases:
  `/etc/init.d/collabnet stop`
- If you are upgrading from TeamForge 16.10, 17.1, or 17.4 releases:
  `/opt/collabnet/teamforge/bin/teamforge stop`
- If you are upgrading from TeamForge 17.8 or later releases:
  `teamforge stop`

2. Log on to the EventQ Server. Stop EventQ.

- If you are upgrading from TeamForge 16.3:
  `/etc/init.d/orchestrate stop`
- If you are upgrading from TeamForge 16.7, 16.10, or 17.1 release:
  ```
  /etc/init.d/eventq stop
  /etc/init.d/collabnet-rabbitmq-server stop
  /etc/init.d/collabnet-mongod stop
  ```
- If you are upgrading from TeamForge 17.4 release:
  `/opt/collabnet/teamforge/bin/teamforge stop`
- If you are upgrading from TeamForge 17.8 or later releases:
  `teamforge stop`

3. Upgrade the operating system packages.

```
yum upgrade
```

NOTE: Run `yum upgrade` on all the servers.

Configure the TeamForge Installation Repository

1. Configure the TeamForge installation repository.

TeamForge Installation Repository Configuration for Sites with Internet Access

1. Contact the CollabNet Support and download the TeamForge 18.1 installation repository package to `/tmp`.
2. Install the repository package.
```
yum install -y /tmp/collabnet-teamforge-repo-18.1-0-noarch.rpm
```
3. Refresh your repository cache.
```
yum clean all
```

TeamForge Installation Repository Configuration for Sites without Internet Access

1. Contact the CollabNet Support to get the auxiliary installer package for TeamForge 18.1 disconnected installation and save it in `/tmp`.
   - RHEL/CentOS 6.9 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel6.x86_64.rpm`
   - RHEL/CentOS 7.4 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel7.x86_64.rpm`
   - In addition to the above CentOS 7.4 64 bit RPM package, you must get the following CentOS 7.4 compatibility RPM, which is required for TeamForge 18.1 disconnected media installation on CentOS 7.4 profile: `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm`.
2. Unpack the disconnected installation package.
```
rpm -Uvh <package-name>
```
3. Unpack the `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm` package if you are installing TeamForge 18.1 on CentOS 7.4.
```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

4. If not mounted already, mount the RHEL/CentOS installation DVD.

   The DVD contains the necessary software and utilities required for installing TeamForge without internet access. In the following commands, replace “cdrom” with the identifier for your server’s CD/DVD drive, if necessary.

   ```
   cd /media/
   mkdir cdrom
   mount /dev/cdrom ./cdrom/
   ```

   If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

5. Create a yum configuration file that points to the RHEL/CentOS installation DVD.
   ```
   vi /etc/yum.repos.d/cdrom.repo
   ```

   Here’s a sample yum configuration file.

   ```
   [RHEL-CDROM]
   name=RHEL CDRom
   baseurl=file:///media/cdrom/Server/
   gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
   enabled=1
   gpgcheck=0
   ```
6. Verify your yum configuration files.
   ```
   yum list httpd
   yum list apr
   ```

Upgrade the TeamForge Services

1. Upgrade the TeamForge application services on the TeamForge Application Server (server-01).
   ```
   yum install teamforge
   ```
2. Upgrade the TeamForge database services on the TeamForge Database Server (server-02).
   ```
   yum install teamforge
   ```
3. Upgrade the EventQ services on the EventQ Server (server-03).
   ```
   yum install teamforge-eventq CN-eventq collabnet-nginx collabnet-passenger
   ```
4. Upgrade the Review Board services on the Review Board Server (server-04).
   ```
   yum install teamforge
   ```
5. Upgrade the SCM services on the SCM Server (server-05).
   ```
   yum install teamforge-scm teamforge-git
   ```

6. Upgrade the Code Search service on the Code Search Server (server-06).
   ```
   yum install teamforge-codesearch
   ```

Set up the site-options.conf File

1. Log on to the TeamForge Database Server (server-02) and set up the `site-options.conf` file.

   > **IMPORTANT:** See Site options change log for a list of site option changes. While upgrading to a latest TeamForge release, make sure that obsolete site option tokens, if any, are removed from the `site-options.conf` file of the TeamForge version you are upgrading to.

   ```
   vi /opt/collabnet/teamforge/etc/site-options.conf
   ```

host:SERVICES Token

```
server-01:SERVICES=ctfcore search mail etl binary reviewboard-adapter clis
erver
server-02:SERVICES=ctfcore-database ctfcore-datamart gerrit-database binar
y-database reviewboard-database
server-03:SERVICES=eventq rabbitmq mongodb redis
server-04:SERVICES=reviewboard
server-05:SERVICES=subversion cvs gerrit
server-06:SERVICES=codesearch
```

host:PUBLIC_FQDN Token

```
server-01:PUBLIC_FQDN=my.app.domain.com
```

> **NOTE:** You cannot have a separate PUBLIC_FQDN for EventQ.

MONGODB_APP_DATABASE_NAME

Set the `MONGODB_APP_DATABASE_NAME` token with EventQ's database name in the `site-options.conf` file. Do this if and only if you are upgrading from TeamForge 17.1 or earlier to TeamForge 17.4 or later.

```
MONGODB_APP_DATABASE_NAME=orchestrate
```

Save the `site-options.conf` file.

For further customization of your site configuration (SSL settings, password policy settings, PostgreSQL settings, LDAP settings and so on): Site options configuration contd…

SSL Tokens

SSL is enabled by default and a self-signed certificate is auto-generated. Use the following tokens to adjust this behavior.

```
SSL_CERT_FILE=
SSL_KEY_FILE=
SSL_CHAIN_FILE=
```

- To generate the SSL certificates, see Generate SSL certificates.
- Have the custom SSL certificate and private key for custom SSL certificate in place and provide their absolute paths in these tokens. SSL_CHAIN_FILE (intermediate certificate) is optional.
- You can also encrypt the data traffic between the application and database servers and between the ETL and datamart servers in a distributed setup. Use the [DATABASE_SSL] [siteoptiontokens.html#DATABASE_SSL] token to do that. See Encrypt Database Network Traffic.

Password Tokens

- TeamForge 7.1 and later support automatic password creation. See AUTO_DATA for more information.

- Set the REQUIRE_PASSWORD_SECURITY token to `true` to enforce password security policy for the site.

  If the token REQUIRE_PASSWORD_SECURITY is enabled, then set a value for the token, PASSWORD_CONTROL_EFFECTIVE_DATE.

  > **WARNING:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don’t meet the password security requirements starting from the date set for the `PASSWORD_CONTROL_EFFECTIVE_DATE` token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See PASSWORD_CONTROL_EFFECTIVE_DATE for more information.

You can also set the following tokens to enforce a more stricter password policy:

  - MINIMUM_PASSWORD_LENGTH
  - MAX_PASSWORD_LENGTH
  - PASSWORD_REQUIRES_NUMBER
  - PASSWORD_REQUIRES_NON_ALPHANUM
  - PASSWORD_REQUIRES_MIXED_CASE
  - REQUIRE_PASSWORD_SECURITY
  - LOGIN_ATTEMPT_LOCK
  - PASSWORD_HISTORY_AGE
  - ALLOW_PASSWORD_DICTIONARY_WORD
- If the token REQUIRE_RANDOM_ADMIN_PASSWORD is already set to `true`, then set the token ADMIN_EMAIL with a valid email address.
  `ADMIN_EMAIL=root@{__APPLICATION_HOST__}`
- If you have LDAP set up for external authentication, you must set the REQUIRE_USER_PASSWORD_CHANGE site options token to `false`.
- Verify and update the list of non-expiring TeamForge user accounts (password never expires).
  `USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin`

Prevent Cross-site Scripting

An attacker could potentially upload an HTML page to TeamForge that contains active code, such as JavaScript. This active code would then be executed by clients' browsers when they view the page, which can harm the system.

To prevent an attack of this sort, you can specify whether or not HTML code is displayed in TeamForge. This flag applies to all documents, tracker, task, and forum attachments, and files in the file release system.

Set the SAFE_DOWNLOAD_MODE token according to your requirements. For more information, see SAFE_DOWNLOAD_MODE.

PostgreSQL Tokens and Settings

Make sure the PostgreSQL tokens in the `site-options.conf` file are set as recommended in the following topic: What are the right PostgreSQL settings for my site?

JAVA_OPTS

Configure the JBOSS_JAVA_OPTS `site-options.conf` token. See JBOSS_JAVA_OPTS.

> **NOTE:** All JVM parameters but `-Xms1024m` and `-Xmx2048m` have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as `-XX:MaxMetaspaceSize=512m` `-XX:ReservedCodeCacheSize=128M` `-server` `-XX:+HeapDumpOnOutOfMemoryError` `-Djsse.enableSNIExtension=false` `-Dsun.rmi.dgc.client.gcInterval=600000` `-Dsun.rmi.dgc.server.gcInterval=600000`) in the site-options.conf file.

TeamForge 18.1 (and later) supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.

- JBOSS_JAVA_OPTS
- PHOENIX_JAVA_OPTS
- INTEGRATION_JAVA_OPTS
- ETL_JAVA_OPTS
- ELASTICSEARCH_JAVA_OPTS

TeamForge provision fails on sites that use these options post upgrade to TeamForge 18.1.

Save the `site-options.conf` file.

2. Provision services.
   `teamforge provision`
   TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.
3. Copy the `/opt/collabnet/teamforge/etc/site-options.conf` file from the TeamForge Database Server (server-02) to the `/opt/collabnet/teamforge/etc/` directory of all other servers.

Provision Services on All the Servers

1. Do this if and only if you have EventQ integration and are upgrading from TeamForge 17.1 or earlier to TeamForge 18.1.

Copy the `/opt/collabnet/rabbitmq/var` and `/opt/collabnet/mongodb/data` directories to `/opt/collabnet/teamforge/var/rabbitmq` and `/opt/collabnet/teamforge/var/mongodb` directories respectively.

```
cp -R /opt/collabnet/rabbitmq/var /opt/collabnet/teamforge/var/rabbitmq
cp -R /opt/collabnet/mongodb/data /opt/collabnet/teamforge/var/mongodb
```

2. Provision services.
   `teamforge provision`
   TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.

✓ You must provision services in a particluar sequence. Usually you start with the Database Server, followed by the Application Server and then by other servers such as SCM, Review Board, EventQ and Code Search servers.

✓ The TeamForge installer derives this sequence from your `site-options.conf` file and shows you the order of provisioning servers when you try to provision one of the distributed servers. Follow the exact sequence as instructed.

1. Provision the Database Server (server-02)
2. Provision the Application Server (server-01)
3. Provision the SCM server (server-05)
4. Provision the EventQ Server (server-03)
5. Provision the Review Board Server (server-04)
6. Provision the Code Search Server (server-06)

Reinitialize TeamForge

1. Reinitialize TeamForge on the Review Board Server.
   `teamforge reinitialize`
2. During `teamforge provision`, the `Register SCM integration` process fails on sites that use self-signed certificates. Perform these steps in such cases.
   1. Restart JBoss on the TeamForge Application Server.
      `teamforge restart -s jboss`
   2. Reinitialize TeamForge on the SCM Server.
      `teamforge reinitialize`

   Do you have Git and other SCM tools (SVN and CVS) on two separate servers?
   Git and other SCM tools (SVN and CVS) are typically installed on a separate server dedicated for SCM. However, if you have Git and SCM (SVN and CVS) on two separate servers, restart Jboss on the TeamForge Application Server and reinitialize TeamForge on the SCM Server (SVN and CVS) as

discussed earlier. In addition, you must also restart TeamForge on the Git Server.

Restart TeamForge on the Git Server: `teamforge restart`

Finishing Tasks

1. Run the `/var/lib/pgsql/analyze_new_cluster.sh` script. This is required if and only if you are upgrading from TeamForge 17.1 (or earlier) to TeamForge 17.8 (or later).
   `su - postgres -c "/var/lib/pgsql/analyze_new_cluster.sh"`
2. If you have CVS integrations, synchronize permissions post upgrade. See Synchronize TeamForge Source Control Integrations.
3. Verify TeamForge upgrade.
   1. Reboot the server and make sure all services come up automatically at startup.
   2. Log on to the TeamForge web application using the default Admin credentials.
      - Username: `admin`
      - Password: `admin`
   3. If your site has custom branding, verify that your branding changes still work as intended. See Customize TeamForge.
   4. Let your site's users know they've been upgraded. See Create a Site-wide Broadcast.

Post Upgrade Tasks

- Run TeamForge in SELinux enabled Mode
- Add EventQ to Existing Projects
- Users are not getting email notifications for review requests and reviews. What should I do?

1. `reviewboard-adapter` must always be installed on the TeamForge Application Server. □

Distributed setup with TeamForge, Oracle Database (including Datamart) and EventQ installed on separate servers.

✓ In this setup, TeamForge, Oracle database and other services are distributed across three servers, server-01 through server-03 as illustrated in the following table.

✓ You can install TeamForge on both RHEL/CentOS 7.4 and 6.9. In this distributed setup, all the following services are installed on RHEL/CentOS 7.4 servers.

server-01	server-02	server-03
TeamForge Application Server	**Oracle Database Server**	**EventQ Server**
ctfcore	ctfcore-database	eventq
mail	ctfcore-datamart	rabbitmq
etl		mongodb
search		redis
codesearch		
gerrit		
gerrit-database		
subversion		
cvs		
reviewboard		
reviewboard-database		
reviewboard-adapter[1]		
binary		
binary-database		
cliserver		

Dos and Don’ts

Here’s a list of dos, don’ts and points to remember when you install or upgrade TeamForge. Show/Hide

Dos

- Understand TeamForge installation [requirements][requirements] and [plan your installation or upgrade][plan_your_installation_upgrade].
- Get your [TeamForge license][teamforgelicense] key and keep it handy.
- Verify your basic networking setup before installing or upgrading TeamForge. See [Set Up Networking for TeamForge][setupnetworking].
- Look for new or modified `site-options.conf` tokens and update your `site-options.conf` file as required during the upgrade process. See [Site Options Change Log][siteoptionschangelog].
- Set up a TeamForge Stage Server before you upgrade your Production Server.
- Stop TeamForge services on all servers in a distributed setup while upgrading to TeamForge 18.1.
- Uninstall hot fixes and add-ons, if any, before you start the TeamForge 18.1 upgrade procedure.

- TeamForge 18.1 supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.
  - JBOSS_JAVA_OPTS
  - PHOENIX_JAVA_OPTS
  - INTEGRATION_JAVA_OPTS
  - ETL_JAVA_OPTS
  - ELASTICSEARCH_JAVA_OPTS

  TeamForge provision fails on sites that use these options post TeamForge 18.1 upgrade.

Don'ts

- Do not customize your operating system installation. Select only the default packages list.
- While upgrading TeamForge, whether in place or on new hardware, always reuse the old `site-options.conf` file and make changes as necessary. **Do not** try to start with a new `site-options.conf` file. Reusing the old `site-options.conf` avoids many potential problems, particularly around the management of usernames and passwords.
- Do not manually modify TeamForge-managed site option tokens such as the `AUTO_DATA` token. See [AUTO_DATA][siteoptiontokens.html#auto_data] for more information.

- If you are creating symlinks, note that you must create symlinks only to the TeamForge data directory (`/opt/collabnet/teamforge/var`). You should not create symlinks to TeamForge application directories (such as `/opt/collabnet`).

Points to Remember

- Installing or upgrading TeamForge needs root privileges. You must log on as root or use a root shell to install or upgrade TeamForge.
- SSL is enabled by default and a self-signed certificate is auto-generated. However, you can use a few `site-options.conf` tokens to adjust this behavior. To generate the SSL certificates, see [Generate SSL Certificates][generatesslcerts].
- For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.
- If you have Git integration on a separate server, both TeamForge and Git servers must have their time and date synchronized.

- While you can run both EventQ and TeamForge on the same server, CollabNet recommends such an approach only for testing purposes. It’s always recommended to run EventQ on a separate server for optimal scalability.
- No backup is required for same hardware upgrades. However, you can create a backup as a measure of caution. See [Back up and Restore TeamForge][backupandrestore] for more information.
- Always use compatible JDBC drivers meant for specific database versions. See JDBC Drivers Reference for more information. Also see: [Why do ETL jobs fail post TeamForge upgrade?][database-faqs.html#etl_fails_post_upgrade].
- You can run the initial load job any time after the installation of TeamForge. We recommend that you run it before you hand over the site to the users. For more information, see [ETL Initial Load Jobs][database-faqs.html#etl_initial_load_jobs].
- SOAP50 APIs and event handlers are no longer supported in TeamForge 16.10 and later. Use the latest TeamForge SOAP/REST APIs.
- TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.
- Installing TeamForge with service-specific FQDNs (instead of machine-specific host/domain names) is highly recommended so that you will be able to change the system landscape at a later point in time without having any impact on the URLs (in other words, end users do not have to notice or change anything). For example, you can create FQDNs specifically for services such as Subversion, Git, mail, Codesearch and so on. For more information, see [Service-specific FQDNs][siteoptiontokens.html#servicespecificfqdns].
- All such service-specific FQDNs must belong to a single sub domain and it is recommended to create a new sub domain for TeamForge.
- If you are using service-specific FQDNs
  - A wildcard SSL cert is required. SNI SSL cert cannot be used.
  - When SSL is enabled and no custom SSL certificates are provided, a self-signed wildcard cert is generated for the sub domain.
  - When SSL is enabled and a custom SSL certificate is provided, the CN of the certificate is verified to be a wildcard CN.
- You cannot have a separate PUBLIC_FQDN for EventQ.
- The ability to run separate PostgreSQL instances for TeamForge database and datamart on the same server is being deprecated in TeamForge 17.11. If you have TeamForge database and datamart on separate PostgreSQL instances on the same server and if you are upgrading on a new hardware, you must [Create a Single Cluster for Both Database and Datamart][movedbdmintoonepginstance] while upgrading to TeamForge 17.11 or later.
- While upgrading TeamForge-Git integration servers, it is important that Git master and slave servers are upgraded to the same version of TeamForge-Git integration. On sites with Git Replica Servers, you must upgrade the Git Replica Servers first and then upgrade the master Git servers.

Back up Your Oracle Database

- See Oracle Database Backup and Recovery User's Guide
- See Oracle Database Backup and Recovery FAQ

Uninstall Custom Event Handlers, Hot Fixes and Add-ons

Log on to the TeamForge Application Server.

1. SOAP 50 is no longer supported. Back up all your custom event handlers and remove all the event handler JAR files before starting your TeamForge upgrade process.
    1. Go to **My Workspace** > **Admin**.
    2. Click **System Tools** from the **Projects** menu.
    3. Click **Customizations**.
    4. Select the custom event handler and click **Delete**.

    > **TIP:** Post upgrade, you can add custom event handlers again from the backup while making sure that you don't have SOAP50 (deprecated) library used.

2. Uninstall hotfixes and add-ons, if any, installed on your site.

yum upgrade

1. Stop TeamForge.

    > **IMPORTANT:** Stop TeamForge on all the servers in a distributed setup.

    - If you are upgarding from TeamForge 16.7 or earlier releases:
      `/etc/init.d/collabnet stop`
    - If you are upgrading from TeamForge 16.10, 17.1, or 17.4 releases:
      `/opt/collabnet/teamforge/bin/teamforge stop`
    - If you are upgrading from TeamForge 17.8 or later releases:
      `teamforge stop`

2. Log on to the EventQ Server. Stop EventQ.

- If you are upgrading from TeamForge 16.3:
  `/etc/init.d/orchestrate stop`
- If you are upgrading from TeamForge 16.7, 16.10, or 17.1 release:
  ```
  /etc/init.d/eventq stop
  /etc/init.d/collabnet-rabbitmq-server stop
  /etc/init.d/collabnet-mongod stop
  ```
- If you are upgrading from TeamForge 17.4 release:
  `/opt/collabnet/teamforge/bin/teamforge stop`
- If you are upgrading from TeamForge 17.8 or later releases:
  `teamforge stop`

3. Upgrade the operating system packages.

   ```
   yum upgrade
   ```

   NOTE: Run `yum upgrade` on all the servers.

Configure the TeamForge Installation Repository

1. Configure the TeamForge installation repository.

TeamForge Installation Repository Configuration for Sites with Internet Access

1. Contact the CollabNet Support and download the TeamForge 18.1 installation repository package to `/tmp`.
2. Install the repository package.
   `yum install -y /tmp/collabnet-teamforge-repo-18.1-0-noarch.rpm`
3. Refresh your repository cache.
   `yum clean all`

TeamForge Installation Repository Configuration for Sites without Internet Access

1. Contact the CollabNet Support to get the auxiliary installer package for TeamForge 18.1 disconnected installation and save it in `/tmp`.
   - RHEL/CentOS 6.9 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel6.x86_64.rpm`
   - RHEL/CentOS 7.4 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel7.x86_64.rpm`
   - In addition to the above CentOS 7.4 64 bit RPM package, you must get the following CentOS 7.4 compatibility RPM, which is required for TeamForge 18.1 disconnected media installation on CentOS 7.4 profile: `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm`.
2. Unpack the disconnected installation package.
   ```
   rpm -Uvh <package-name>
   ```
3. Unpack the `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm` package if you are installing TeamForge 18.1 on CentOS 7.4.
   ```
   rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
   ```
4. If not mounted already, mount the RHEL/CentOS installation DVD.

   The DVD contains the necessary software and utilities required for installing TeamForge without internet access. In the following commands, replace “cdrom” with the identifier for your server’s CD/DVD drive, if necessary.

   ```
   cd /media/
   mkdir cdrom
   mount /dev/cdrom ./cdrom/
   ```

   If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.
5. Create a yum configuration file that points to the RHEL/CentOS installation DVD.
   ```
   vi /etc/yum.repos.d/cdrom.repo
   ```

   Here’s a sample yum configuration file.

   ```
   [RHEL-CDROM]
   name=RHEL CDRom
   ```

```
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

6. Verify your yum configuration files.
```
yum list httpd
yum list apr
```

Upgrade the TeamForge Services

1. Upgrade the TeamForge and Review Board application services on the TeamForge Application Server (server-01).
```
yum install teamforge
```
2. Upgrade the EventQ services on the EventQ Server (server-03).
```
yum install teamforge-eventq CN-eventq collabnet-nginx collabnet-passenger
```

Back up the TeamForge Data Directories

On sites running TeamForge 16.7 or earlier versions:

1. Back up the following data directories.

> **TIP:** In a distributed setup, you must backup specific directories such as `/svnroot` and `/cvsroot` from the server that hosts those SCM services.

Directory	Contents
/opt/collabnet/teamforge/var	User-created data, such as artifact attachments
/opt/collabnet/reviewboard	Review Board data
/svnroot	Subversion source code repositories
/sf-svnroot	Subversion repository for branding data
/cvsroot	CVS source code repositories (required only if you have CVS)
/gitroot	Git source code repositories

```
cp -Rpf /svnroot /sf-svnroot /cvsroot /gitroot /opt/collabnet/teamforge/va
r /opt/collabnet/reviewboard /tmp/backup_dir
```
2. Back up the `/opt/collabnet/gerrit` directory if you have Git integration.

> **TIP:** Do this on the server that hosts the TeamForge-Git integration services.

```
mkdir /tmp/backup_dir/gerrit
cp -Rpfv /gitroot /tmp/backup_dir
cp -Rpfv /opt/collabnet/gerrit/ /tmp/backup_dir/gerrit
```

3. Back up the EventQ YAML files and directories if you have EventQ installed.

> **TIP:** Do this on the server that hosts TeamForge EventQ services.

```
cp -Rpfv /opt/collabnet/eventq/config/*.yml /opt/collabnet/mongodb /opt/col
labnet/rabbitmq /tmp/backup_dir
```

On sites running TeamForge 16.10 or later versions:

1. Back up the `/opt/collabnet/teamforge/var` directory.

> **TIP:** Do this on both the TeamForge Application and Database servers in case you have them running on two separate servers.

```
mkdir -p /tmp/backup_dir
cp -Rpfv /opt/collabnet/teamforge/var /tmp/backup_dir
```

2. Back up the `/opt/collabnet/gerrit` directory if you have Git integration.

> **TIP:** Do this on the server that hosts the TeamForge-Git integration services.

```
mkdir /tmp/backup_dir/gerrit
cp -Rpfv /opt/collabnet/gerrit/ /tmp/backup_dir/gerrit
```

3. Back up the EventQ YAML files and directories if you have EventQ installed.

> **TIP:** Do this on the server that hosts TeamForge EventQ services.

```
cp -Rpfv /opt/collabnet/eventq/config/*.yml /opt/collabnet/mongodb /opt/col
labnet/rabbitmq /tmp/backup_dir
```

Back up and Restore Review Board Database and Data Directories

See [Back up and Restore Review Board Database and Data Directories][backupandrestore.html#backuprb]

Set up the site-options.conf File and Provision Services

1. Log on to the TeamForge Application Server (server-01), set up the `site-options.conf` file, and provision the services.
   `vi /opt/collabnet/teamforge/etc/site-options.conf`

   ## host:SERVICES Token

   ```
   server-01:SERVICES=ctfcore mail etl search subversion cvs codesearch cliserver gerrit gerrit-database  binary binary-database reviewboard reviewboard-database reviewboard-adapter cliserver
   server-02:SERVICES=ctfcore-database ctfcore-datamart
   server-03:SERVICES=eventq mongodb redis rabbitmq
   ```

   ## host:PUBLIC_FQDN Token

   ```
   server-01:PUBLIC_FQDN=my.app.domain.com
   ```

   ## Configure the Oracle Database Tokens

   Configure the Oracle database name, usernames and passwords as configured on the Oracle Database Server.

   - Database type is `oracle` (`DATABASE_TYPE=oracle`)
   - Database service name is the host name of the Oracle Database Server (for example, `DATABASE_SERVICE_NAME=cu349.maa.collab.net`)
   - Reports database service name is the host name of the server where the datamart is (for example, `REPORTS_DATABASE_SERVICE_NAME=cu349.maa.collab.net`)

```
DATABASE_TYPE=oracle

# Adjust usernames/passwords to match what has been configured on the database server.
DATABASE_USERNAME=ctfuser
DATABASE_PASSWORD=ctfpwd
DATABASE_READ_ONLY_USER=ctfrouser
DATABASE_READ_ONLY_PASSWORD=ctfropwd
DATABASE_NAME=orcl
DATABASE_SERVICE_NAME=

# Adjust usernames/passwords to match what has been configured on the database server.
REPORTS_DATABASE_USERNAME=ctfrptuser
REPORTS_DATABASE_PASSWORD=ctfrptpwd
REPORTS_DATABASE_NAME=orcl
REPORTS_DATABASE_READ_ONLY_USER=ctfrptrouser
REPORTS_DATABASE_READ_ONLY_PASSWORD=ctfrptropwd
REPORTS_DATABASE_SERVICE_NAME=
```

Save the `site-options.conf` file.

For further customization of your site configuration (SSL settings, password policy settings, PostgreSQL settings, LDAP settings and so on): Site options configuration contd…

SSL Tokens

SSL is enabled by default and a self-signed certificate is auto-generated. Use the following tokens to adjust this behavior.

```
SSL_CERT_FILE=
SSL_KEY_FILE=
SSL_CHAIN_FILE=
```

- To generate the SSL certificates, see [Generate SSL certificates][generatesslcerts].
- Have the custom SSL certificate and private key for custom SSL certificate in place and provide their absolute paths in these tokens. SSL_CHAIN_FILE (intermediate certificate) is optional.

Password Tokens

- TeamForge 7.1 and later support automatic password creation. See AUTO_DATA for more information.
- Set the [REQUIRE_PASSWORD_SECURITY] [siteoptiontokens.html#REQUIRE_PASSWORD_SECURITY] token to `true` to enforce password security policy for the site.

  If the token REQUIRE_PASSWORD_SECURITY is enabled, then set a value for the token, PASSWORD_CONTROL_EFFECTIVE_DATE.

  > **WARNING:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the `PASSWORD_CONTROL_EFFECTIVE_DATE` token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See PASSWORD_CONTROL_EFFECTIVE_DATE for more information.

  You can also set the following tokens to enforce a more stricter password policy:

  - MINIMUM_PASSWORD_LENGTH
  - MAX_PASSWORD_LENGTH
  - PASSWORD_REQUIRES_NUMBER
  - PASSWORD_REQUIRES_NON_ALPHANUM
  - PASSWORD_REQUIRES_MIXED_CASE
  - REQUIRE_PASSWORD_SECURITY
  - LOGIN_ATTEMPT_LOCK
  - PASSWORD_HISTORY_AGE
  - ALLOW_PASSWORD_DICTIONARY_WORD
- If the token REQUIRE_RANDOM_ADMIN_PASSWORD is already set to `true`, then set the token ADMIN_EMAIL with a valid email address.
  ```
  ADMIN_EMAIL=root@{__APPLICATION_HOST__}
  ```
- If you have LDAP set up for external authentication, you must set the REQUIRE_USER_PASSWORD_CHANGE site options token to `false`.

Prevent Cross-site Scripting

An attacker could potentially upload an HTML page to TeamForge that contains active code, such as JavaScript. This active code would then be executed by clients' browsers when they view the page, which can harm the system.

To prevent an attack of this sort, you can specify whether or not HTML code is displayed in TeamForge. This flag applies to all documents, tracker, task, and forum attachments, and files in the file release system.

Set the SAFE_DOWNLOAD_MODE token according to your requirements. For more information, see [SAFE_DOWNLOAD_MODE][siteoptiontokens.html#SAFE_DOWNLOAD_MODE].

JAVA_OPTS

Configure the JBOSS_JAVA_OPTS site-options.conf token. See [JBOSS_JAVA_OPTS][siteoptiontokens.html#jboss_java_opts].

> **NOTE:** All JVM parameters but `-Xms1024m` and `-Xmx2048m` have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as `-XX:MaxMetaspaceSize=512m` `-XX:ReservedCodeCacheSize=128M` `-server` `-XX:+HeapDumpOnOutOfMemoryError` `-Djsse.enableSNIExtension=false` `-Dsun.rmi.dgc.client.gcInterval=600000` `-Dsun.rmi.dgc.server.gcInterval=600000`) in the site-options.conf file.

TeamForge 18.1 (and later) supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.

- JBOSS_JAVA_OPTS
- PHOENIX_JAVA_OPTS
- INTEGRATION_JAVA_OPTS
- ETL_JAVA_OPTS
- ELASTICSEARCH_JAVA_OPTS

TeamForge provision fails on sites that use these options post upgrade to TeamForge 18.1.

Save the `site-options.conf` file.

2. Provision services.
   `teamforge provision`
   TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.

Set up the EventQ Server (server-03)

Log on to the EventQ Server (server-03), set up the `site-options.conf` file, and provision the services.

1. Copy the `/opt/collabnet/teamforge/etc/site-options.conf` file from the TeamForge Application Server to the EventQ Server's `/opt/collabnet/teamforge/etc/` directory.

2. Provision services.

   ```
   teamforge provision
   ```
   TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.

Verify TeamForge Upgrade

1. Verify TeamForge upgrade.
   1. Reboot the server and make sure all services come up automatically at startup.
   2. Log on to the TeamForge web application using the default Admin credentials.
      - Username: `admin`
      - Password: `admin`
   3. If your site has custom branding, verify that your branding changes still work as intended. See [Customize TeamForge][customize].
   4. Let your site's users know they've been upgraded. See [Create a Site-wide Broadcast][siteadmin-createsitewidebroadcast].

Post Upgrade Tasks

- [Run TeamForge in SELinux `enabled` Mode][setupselinux]
- [Add EventQ to Existing Projects][add_eventq_to_projects_py]

- [Users are not getting email notifications for review requests and reviews. What should I do?][reviewboard-faqs.html#rbemailsettings]

1. `reviewboard-adapter` must always be installed on the TeamForge Application Server. □

The ability to run separate PostgreSQL instances for TeamForge database and datamart on the same server has been deprecated in TeamForge 17.11.

The ability to run separate PostgreSQL instances for TeamForge database and datamart on the same server has been deprecated in TeamForge 17.11. If you have the TeamForge database and datamart on separate PostgreSQL instances on the same server and if you are upgrading on a new hardware, follow these instructions to create a dump of both the database and datamart and load them into one PostgreSQL instance on the same server.

- During TeamForge installation, the `REPORTS_DATABASE_PORT` token should no longer be used to assign a separate port for datamart on the server that also runs the TeamForge database. The following warning shows up if you use the `REPORTS_DATABASE_PORT` token with a custom port number (other than the default value, which is 5432).

  ```
  Using two separate Postgres clusters for database and datamart on the same
   machine is deprecated. Consider deploying the two clusters on two machine
  s or using a single cluster for both databases.
  ```

- If you have the TeamForge database and datamart running on separate PostgreSQL instances on the same server:
  - **New Hardware Upgrade**: If you are upgrading on a new hardware, it is highly recommended to create a dump of both the database and datamart and load them into the same PostgreSQL instance.
  - **Same Hardware Upgrade**: If you are upgrading on the same hardware, you may still choose to use the `REPORTS_DATABASE_PORT` and have the database and datamart running on two separate PostgreSQL instances. However, support for `REPORTS_DATABASE_PORT` token may end in one of the future TeamForge releases, when you may have to dump and load both the database and datamart on the same PostgreSQL instance anyway.

> **NOTE:** The following instructions assume that you are upgrading from TeamForge 17.8 (with database and datamart on separate ports) to TeamForge 17.11 (or later) on a new hardware.

1. Do this on the existing TeamForge Application Server where the database and datamart runs on two separate ports.
   1. Make a dump file of your site database.

```
su - postgres
/usr/bin/pg_dumpall > /var/lib/pgsql/9.x/backups/teamforge_data_backup.dmp
exit
mkdir /tmp/backup_dir
cp /var/lib/pgsql/9.x/backups/teamforge_data_backup.dmp /tmp/backup_dir/
```

   2. Make a dump of your datamart.

      ```
      /usr/bin/pg_dumpall -p <reports_database_port> > /var/lib/pgsql/9.x/backups/teamforge_reporting_data_backup.dmp
      ```

2. Do this on the new TeamForge server.
   1. Recreate the runtime environment (do not provision TeamForge directly).

      ```
      teamforge deploy
      ```

   2. Reload the site database.

      ```
      su - postgres
      /usr/bin/psql < /tmp/backup_dir/teamforge_data_backup.dmp
      exit
      ```

   3. Reload the datamart.

      ```
      su - postgres -c "/usr/bin/psql -p <reports_database_port> < /tmp/backup_dir/teamforge_reporting_data_backup.dmp"
      ```

> **WARNING:** You must move the PostgreSQL 9.x directory (`mv /var/lib/pgsql/9.x /var/lib/pgsql/9.x_old`) after reloading the database dump, failing which the `teamforge provision` command will not be successful.

3. Provision services.

   ```
   teamforge provision
   ```

Save a copy of your TeamForge site's data to a location from where you can quickly retrieve it to your TeamForge site.
Before You Begin

- The following instructions, though applicable in general to any PostgreSQL version, assume that you run a TeamForge version that runs on PostgreSQL 9.3, which you want to back up. Replace 9.3 with the PostgreSQL version you run, if required.
- If you are upgrading by installing TeamForge 18.1 on new hardware, then you'll need the backed-up site data to complete the upgrade.
- If you are upgrading your site on the same hardware, then you won't need to back up but you should create a backup anyway, as a measure of caution.

Back up and Restore TeamForge Database, Data Directories and site-options.conf

1. Log on to the TeamForge server that you want to back up.

2. SOAP 50 is no longer supported. Back up all your custom event handlers and remove all the event handler JAR files before starting your TeamForge upgrade process.

   > **TIP:** Do this on the TeamForge Application Server.

   1. Go to **My Workspace** > **Admin**.
   2. Click **System Tools** from the **Projects** menu.
   3. Click **Customizations**.
   4. Select the custom event handler and click **Delete**.

      > **TIP:** Post upgrade, you can add custom event handlers again from the backup while making sure that you don't have SOAP50 (deprecated) library used.

3. Stop TeamForge.

   > **IMPORTANT:** Stop TeamForge on all the servers in a distributed setup.

   - If you are upgarding from TeamForge 16.7 or earlier releases:
     ```
     /etc/init.d/collabnet stop
     ```
   - If you are upgrading from TeamForge 16.10, 17.1, or 17.4 releases:
     ```
     /opt/collabnet/teamforge/bin/teamforge stop
     ```
   - If you are upgrading from TeamForge 17.8 or later releases:
     ```
     teamforge stop
     ```

4. Stop EventQ.

   - If you are upgrading from TeamForge 16.3:
     ```
     /etc/init.d/orchestrate stop
     ```
   - If you are upgrading from TeamForge 16.7, 16.10, or 17.1 release:
     ```
     /etc/init.d/eventq stop
     /etc/init.d/collabnet-rabbitmq-server stop
     /etc/init.d/collabnet-mongod stop
     ```
   - If you are upgrading from TeamForge 17.4 release:

```
/opt/collabnet/teamforge/bin/teamforge stop
```

- If you are upgrading from TeamForge 17.8 or later releases:
  ```
  teamforge stop
  ```

5. Back up your site data.

   On sites running TeamForge 16.7 or earlier versions:

   1. Back up your site database.

      > **TIP:** Do this on the server that hosts the TeamForge database and datamart services.

      ```
      mkdir -p /tmp/backup_dir
      cd /var/lib
      tar -zcvf /tmp/backup_dir/pgsql.tgz pgsql/9.3
      ```
   2. Back up the following data directories.

      > **TIP:** In a distributed setup, you must backup specific directories such as `/svnroot` and `/cvsroot` from the server that hosts those SCM services.

Directory	Contents
/opt/collabnet/teamforge/var	User-created data, such as artifact attachments
/opt/collabnet/reviewboard	Review Board data
/svnroot	Subversion source code repositories
/sf-svnroot	Subversion repository for branding data
/cvsroot	CVS source code repositories (required only if you have CVS)
/gitroot	Git source code repositories

      ```
      cp -Rpf /svnroot /sf-svnroot /cvsroot /gitroot /opt/collabnet/teamforg
      e/var /opt/collabnet/reviewboard /tmp/backup_dir
      ```
   3. Back up the `/opt/collabnet/gerrit` directory if you have Git integration.

      > **TIP:** Do this on the server that hosts the TeamForge-Git integration services.

      ```
      mkdir /tmp/backup_dir/gerrit
      cp -Rpfv /gitroot /tmp/backup_dir
      cp -Rpfv /opt/collabnet/gerrit/ /tmp/backup_dir/gerrit
      ```

4. Back up the EventQ YAML files and directories if you have EventQ installed.

   > **TIP:** Do this on the server that hosts TeamForge EventQ services.

   ```
   cp -Rpfv /opt/collabnet/eventq/config/*.yml /opt/collabnet/mongodb /opt
   /collabnet/rabbitmq /tmp/backup_dir
   ```
5. Compress your backed up data.
   ```
   cd /tmp
   tar czvf backup.tgz backup_dir
   ```

On sites running TeamForge 16.10 or later versions:

1. Back up the `/opt/collabnet/teamforge/var` directory.

   > **TIP:** Do this on both the TeamForge Application and Database servers in case you have them running on two separate servers.

   ```
   mkdir -p /tmp/backup_dir
   cp -Rpfv /opt/collabnet/teamforge/var /tmp/backup_dir
   ```
2. Back up the `/opt/collabnet/gerrit` directory if you have Git integration.

   > **TIP:** Do this on the server that hosts the TeamForge-Git integration services.

   ```
   mkdir /tmp/backup_dir/gerrit
   cp -Rpfv /opt/collabnet/gerrit/ /tmp/backup_dir/gerrit
   ```
3. Back up the EventQ YAML files and directories if you have EventQ installed.

   > **TIP:** Do this on the server that hosts TeamForge EventQ services.

   ```
   cp -Rpfv /opt/collabnet/eventq/config/*.yml /opt/collabnet/mongodb /opt
   /collabnet/rabbitmq /tmp/backup_dir
   ```
4. Compress your backup data.
   ```
   cd /tmp
   tar czvf backup.tgz backup_dir
   ```

6. Back up your SSH keys, if any.

7. Back up your SSL certificates and keys, if any.

8. If you are upgrading TeamForge on new hardware, copy the backed up data and the `site-options.conf` file to the new server.
   ```
   scp /tmp/backup.tgz username@newbox:/tmp
   scp /opt/collabnet/teamforge/etc/site-options.conf username@newbox:/tmp
   ```

9. Restore TeamForge data.

   If you are upgrading from TeamForge 16.7 or earlier versions:

   1. Log on to the server where you want to restore the data and unpack the `backup.tgz` file.
      ```
      cd /tmp
      tar xzvf backup.tgz
      ```
   2. Restore the database and data directories.
      ```
      cd /opt/collabnet/teamforge/
      mkdir var
      cd var
      tar zvxf /tmp/backup_dir/pgsql.tgz
      cp -Rpfv /tmp/backup_dir/svnroot /svnroot
      cp -Rpfv /tmp/backup_dir/cvsroot /cvsroot
      cp -Rpfv /tmp/backup_dir/sf-svnroot /sf-svnroot
      cp -Rpfv /tmp/backup_dir/var /opt/collabnet/teamforge/
      ```
   3. Restore the Git data directories on the server that hosts TeamForge-Git integration.
      ```
      cp -Rpfv /tmp/backup_dir/gitroot /gitroot
      cp -Rpfv /tmp/backup_dir/gerrit/gerrit/etc /opt/collabnet/gerrit
      cp -Rpf /tmp/backup_dir/gerrit/gerrit/.ssh /opt/collabnet/gerrit
      cp -Rpf /tmp/backup_dir/gerrit/gerrit/bin /opt/collabnet/gerrit
      cp -Rpf /tmp/backup_dir/gerrit/gerrit/index /opt/collabnet/gerrit
      ```
   4. Restore the EventQ YAML files and data directories on the server that hosts EventQ.
      ```
      yes | cp -Rf /tmp/backup_dir/*.yml /opt/collabnet/eventq/config/
      yes | cp -Rf /tmp/backup_dir/mongodb /opt/collabnet
      yes | cp -Rf /tmp/backup_dir/rabbitmq /opt/collabnet
      ```

   If you are upgrading from TeamForge 16.10 or later versions:

   1. Log on to the server where you want to restore the data and unpack the `backup.tgz` file.
      ```
      cd /tmp
      tar xzvf backup.tgz
      ```
   2. Restore the database and data directories.
      ```
      cp -Rpfv /tmp/backup_dir/var /opt/collabnet/teamforge/
      ```
   3. Restore the Git data directories on the server that hosts TeamForge-Git integration.

```
cp -Rpfv /tmp/backup_dir/gerrit/gerrit/etc /opt/collabnet/gerrit
cp -Rpf /tmp/backup_dir/gerrit/gerrit/.ssh /opt/collabnet/gerrit
cp -Rpf /tmp/backup_dir/gerrit/gerrit/bin /opt/collabnet/gerrit
cp -Rpf /tmp/backup_dir/gerrit/gerrit/index /opt/collabnet/gerrit
```

4. Restore the EventQ YAML files and data directories on the server that hosts EventQ.

```
yes | cp -Rf /tmp/backup_dir/*.yml /opt/collabnet/eventq/config/
yes | cp -Rf /tmp/backup_dir/mongodb /opt/collabnet
yes | cp -Rf /tmp/backup_dir/rabbitmq /opt/collabnet
```

Back up and Restore Review Board Database and Data Directories

The Review Board database should have been backed up already when you backed up TeamForge. So, it is not necessary to take a back up of the Review Board database again.

In case you have Review Board on a separate server outside of the TeamForge Application Server, you must back up the `/opt/collabnet/teamforge/var/pgsql` directory from the Review Board Server that hosts the Review Board database service (`reviewboard-database`).

```
mkdir -p /tmp/backup_dir
cd /opt/collabnet/teamforge/var
tar -zcvf /tmp/backup_dir/reviewboard_pgsql.tgz pgsql/9.3
```

Copy the `/tmp/reviewboard_pgsql.tgz` file to the `/tmp` directory of the new server if you are upgrading Review Board on a new hardware.

```
scp /tmp/reviewboard_pgsql.tgz username@newRBbox:/tmp
```

> **TIP:** The default Review Board data directory has been changed from `/opt/collabnet/reviewboard/data` to `/opt/collabnet/teamforge/var/reviewboard/data` in TeamForge 17.4.

Are You Upgrading from TeamForge 17.1 or Earlier to TeamForge 17.4 or Later?

If you are upgrading from TeamForge 17.1 or earlier to TeamForge 17.4 or later, regardless of whether you upgrade Review Board on the same or new hardware, you must back up your Review Board data directory from `/opt/collabnet/reviewboard/data` and restore it to `/opt/collabnet/teamforge/var/reviewboard/data`.

1. Back up the Review Board data directory.

```
cd /opt/collabnet
tar -zcvf /tmp/reviewboard_data.tgz reviewboard
```

2. Copy the `/tmp/reviewboard_data.tgz` file to the `/tmp` directory of the new server if you are upgrading Review Board on a new hardware.
   `scp /tmp/reviewboard_data.tgz username@newRBbox:/tmp`

Are You Upgrading from TeamForge 17.4 to TeamForge 17.8 or Later?

1. Back up the Review Board data directory.

```
cd /opt/collabnet/teamforge/var
tar -zcvf /tmp/reviewboard_data.tgz reviewboard
```

> **TIP:** If you are upgrading from TeamForge 17.4 (or later), the `/opt/collabnet/teamforge/var` directory would have been backed up already as part of your TeamForge upgrade process, in which case you can skip backing up the `/opt/collabnet/teamforge/var` directory again.

2. Copy the `/tmp/reviewboard_data.tgz` file to the `/tmp` directory of the new server where you plan to have Review Board.

Restore Review Board Data

1. Restore the Review Board database (on the new server where you plan to have the Review Board database).

```
cd /opt/collabnet/teamforge/var/
tar -zxvf /tmp/reviewboard_pgsql.tgz
```

2. Restore the Review Board data directories (on the new server where you plan to have Review Board).

✓ The default Review Board data directory has been changed from `/opt/collabnet/reviewboard/data` to `/opt/collabnet/teamforge/var/reviewboard/data` in TeamForge 17.4. If you are upgrading from TeamForge 17.1 or earlier to TeamForge 17.4 or later, regardless of whether you upgrade Review Board on the same or new hardware, you must back up your Review Board data directory from `/opt/collabnet/reviewboard/data` and restore it to `/opt/collabnet/teamforge/var/reviewboard/data`.

✓ If you are upgrading from TeamForge 17.4 (or later), the `/opt/collabnet/teamforge/var` directory would have been restored already as part of your TeamForge upgrade process, in which case you can skip restoring the `/opt/collabnet/teamforge/var` directory again.

✓ If you are upgrading on a new hardware, ensure that you have already copied the backup of the Review Board data directory to the `/tmp` directory of the new server.

```
cd /opt/collabnet/teamforge/var/
tar -zxvf /tmp/reviewboard_data.tgz
```

Create a TeamForge Database Dump

1. Run the following commands to create a dump file of your site's database. Do this on your database server in case you have your database running on a separate server.

   > **NOTE:** These commands are for a PostgreSQL database, which is the default. If your site uses an Oracle database, follow the Oracle Backup Procedure instead.

   ```
      su - postgres
      /usr/bin/pg_dumpall > /opt/collabnet/teamforge/var/pgsql/9.3/backups/te
   amforge_data_backup.dmp
      exit
      mkdir /tmp/backup_dir
      cp /opt/collabnet/teamforge/var/pgsql/9.3/backups/teamforge_data_backup
   .dmp /tmp/backup_dir/
   ```

2. If your reporting database is running on a separate port, back up your reporting database too.

   > **WARNING:** The ability to run separate PostgreSQL instances for TeamForge database and datamart on the same server has been deprecated in TeamForge 17.11. If you are still running TeamForge database and datamart on two separate ports on the same server, Create a Single Cluster for Both Database and Datamart while upgrading to TeamForge 18.1.

   ```
   /usr/bin/pg_dumpall -p <reports_database_port> > /opt/collabnet/teamforge/
   var/pgsql/9.3/backups/teamforge_reporting_data_backup.dmp
   cp /opt/collabnet/teamforge/var/pgsql/9.3/backups/teamforge_reporting_data
   _backup.dmp /tmp/backup_dir/
   ```

Related Links

Back up and Restore TeamForge Data Using the teamforge.py Script

When you purchase a TeamForge license, you get the right to assign licenses to a specified number of users.

TeamForge License Framework

TeamForge supports a more flexible and granular approach to tool instantiation. TeamForge's license model consists of the following license types: ALM, SCM, Version Control, Collaboration, and Trackers. These license types are tailored to suit the needs of specific set of users that need access to certain tools and functions. A TeamForge user's license type determines the tools available to the user in TeamForge.

While TeamForge supports more selective tool options with these new license changes, there's no impact on customers, both new and existing, requiring all the tools that TeamForge supports.

The following table lists the license types and the tools that go with them (refer to the table at the end of this topic for a complete list of tools and functions).

License Type	Available Tools
ALM	Offers the full range of ALM tools and functions to users.
SCM	Offers core Source Code Management tools and functions to users. Includes all the ALM tools and functions but Tracker and Documents component.
Version Control	Offers Source Code Management, File Releases and Review tools and functions to users.
Collaboration	Offers a range of collaboration tools such as Documents, Wiki and Discussion Forums to users.
Trackers	Offers TeamForge's Tracker capability (Trackers, Planning Folder, Teams, Planning Board, Task Board and Kanban Board) to users. Also includes File Releases.

In addition to the tools offering, the Reporting framework is also controlled by the licenses you have. Meaning, you can view and generate reports based on the license types assigned to you. For example, you must have SCM license to view or generate SCM activity reports. Check with your CollabNet representative if you aren't sure how many licenses or what kind of licenses you want/have.

- Your license key contains a few important numbers:
    - The number of users eligible to use specific licenses such as ALM, SCM, Version Control, Collaboration and Trackers.
    - The IP address of the machine that your site runs on.

      For example, if your organization has 80 users who will use only the source code management features, 100 users who need the Application Lifecycle Management features, your license key string may look like this:

      ```
      ALM=100:SCM=80:supervillaininc:144.16.116.25.:302D02150080D7853DB3E5C6
      F67EABC65BD3AC17D4D35CB3Z00214141D70455B18583BF0A5000CA56B34817ADF8DBF
      I32353A6E657492617369633A38372E3139342E3136102E31322E
      ```

- Your license key only works for the IP address (or range of addresses) of the hardware that your CollabNet TeamForge is running on, as specified in your order form. If your site uses a separate server for its database or source code repositories, make sure your license key reflects the IP addresses of all the necessary servers. If any of these addresses change, ask your CollabNet representative to generate a new key.
- When you create a user account, you can assign the user with available licenses.
- You can purchase a TeamForge license for as many years as you want. The validity period is encoded into your license when it is generated by your CollabNet representative.
- How many users your site can support depends on the type of license. Check with your CollabNet representative if you aren't sure what kind of licenses you have.
- Your license key is attached to the IP address of the server where your site runs. You can get a license key for a single IP address or for a range of IP addresses.
- Your service year starts the first time you log into your site, or the first time you create or edit any item on your site, such as a tracker artifact or a document. Whichever of these events comes first starts the clock.
- The expiration date of your license is shown on the License Info page. (Go to My Workspace > Admin and select Projects > License Info).
- When your service year expires, you can still see the project data on your site, but you cannot make any changes to it. However, you can still carry out some critical maintenance functions for your site:
    - Enter a new license key.
    - Disable or delete users.
    - Change user passwords.
    - Get forgotten user passwords.
- Except deleted users, all other users in TeamForge such as active users, pending users, disabled users, and expired users continue to consume licenses.

Tools Availability Matrix

Tools	ALM	SCM	Version Control	Collaboration	Tracker
Agile Task and Planning Boards	✓				✓
Tracker	✓				✓
Git/SVN Repository Management and Replication	✓	✓	✓		
Code Review	✓	✓	✓		

Tools	ALM	SCM	Version Control	Collaboration	Tracker
Build Automation	✓	✓	✓		
Test Management	✓				✓
File Releases	✓	✓	✓		✓
Binary Repository Management	✓	✓	✓		
EventQ/Activity Stream	✓	✓	✓		✓
Access Controls	✓	✓	✓	✓	✓
Project Work Spaces	✓	✓	✓	✓	✓
Wiki and Discussion Forums	✓	✓		✓	
Document Management	✓			✓	
User Management	✓	✓	✓	✓	✓
Flexible Process and Toolchain Templates	✓	✓	✓	✓	✓
Reusable Dashboards	✓	✓	✓	✓	✓
Categories and Groups	✓	✓	✓	✓	✓
Cross Project Visibility	✓	✓			✓
Cross Project Reporting and Dashboards	✓		✓		✓
Cross Project Search	✓	✓	✓	✓	✓
Site-wide Administration	✓	✓	✓	✓	✓

Tools	ALM	SCM	Version Control	Collaboration	Tracker
Custom Branding and Custom Integrations	✓	✓	✓	✓	✓
Security Architecture	✓	✓	✓	✓	✓
Onsite and Hosted Provisioning	✓	✓	✓	✓	✓
SVN Auto Updates	✓	✓	✓		
SVN Repository Backup and Monitoring	✓	✓	✓		

Supply Your TeamForge License Key via Teamforge User Interface

Your license key enables you to use CollabNet TeamForge for the period of your contract.

Your license key will only work for the IP address of the machine that your CollabNet TeamForge is running on, as specified in your order form.

These steps are for installing your license key via the web interface. If you prefer, you can install it as a text file instead. See Supply your TeamForge license key as a text file.

1. Locate the confirmation email you received from your CollabNet representative when you purchased your contract.
2. Log into your site as a Site Administrator.

   > **NOTE:** A Site Administrator is different from the root user on the server that runs your TeamForgse site.

3. Click **Admin > License Key**.

   If you have entered a license before, the IP address and current licensed number of users on your site are listed on the License Key page. Verify that the IP address is the same as the one you entered in your order form.

4. Click **Enter License Key**.

5. Copy your new license key from the confirmation email and paste it into the Enter License Key field.

A license key string looks like this:

```
ALM=100:SCM=80:supervillaininc:144.16.116.25.:302D02150080D7853DB3E5C6F67E
ABC65BD3AC17D4D35CB3Z00214141D70455B18583BF0A5000CA56B34817ADF8DBFI32353A6
E657492617369633A38372E3139342E3136102E31322EE
```

IMPORTANT: Save this license key in case you need to reinstall CollabNet TeamForge.

6. Click **Save**.
7. Verify that the new value for Licensed Number of Users matches the total number of licensed users in your contract.

Supply Your TeamForge License Key as a Text File

Your license key enables you to use CollabNet TeamForge for the period of your contract.

Your license key will only work for the IP address of the machine that your CollabNet TeamForge is running on.

WARNING: If you are upgrading from a site with a limited number of users to an enterprise-scale site, you must install your license key before starting CollabNet TeamForge. Otherwise, your site could be rendered inoperable.

1. Locate the confirmation email you received from your CollabNet representative when you purchased your contract.

2. Create a text file and copy-paste your license key from the confirmation email into it.

   For example, if your organization has 80 users who will use only the source code management features and 100 users who need the full range of application lifecycle management features, your license key string may look like this:

   ```
   alm=100:SCM=80:supervillaininc:144.16.116.25.:302D02150080D7853DB3E5C6F67E
   ABC65BD3AC17D4D35CB3Z00214141D70455B18583BF0A5000CA56B34817ADF8DBFI32353A6
   E657492617369633A38372E3139342E3136102E31322E
   ```

   NOTE: Save this license key in case you need to reinstall CollabNet TeamForge.

3. Save the text file as `/opt/collabnet/teamforge/var/etc/sflicense.txt`

> **TIP:** Save your license key somewhere remote too, in case you need to reinstall CollabNet TeamForge and your `sflicense.txt` file is not accessible.

4. Make the license file usable by the application.

   ```
   chmod 0664 /opt/collabnet/teamforge/var/etc/sflicense.txt
   chown <APP_USER>:<APP_GROUP> /opt/collabnet/teamforge/var/etc/sflicense.txt
   ```

   Change the values of `<APP_USER>` and `<APP_GROUP>` with the values of `APP_USER` and `APP_GROUP` tokens respectively from the `/opt/collabnet/teamforge/runtime/conf/runtime-options.conf` file.

View License Information

You can obtain a summary of the license information from the **License Info** page.

The License Info page provides you with all the basic information about the licenses you purchased for your TeamForge site. This includes details such as the number of TeamForge licenses you had obtained, how many you have used, expiration date and so on.

1. Go to **My Workspace > Admin**.
2. Select **LICENSE INFO** from the **Projects** menu.

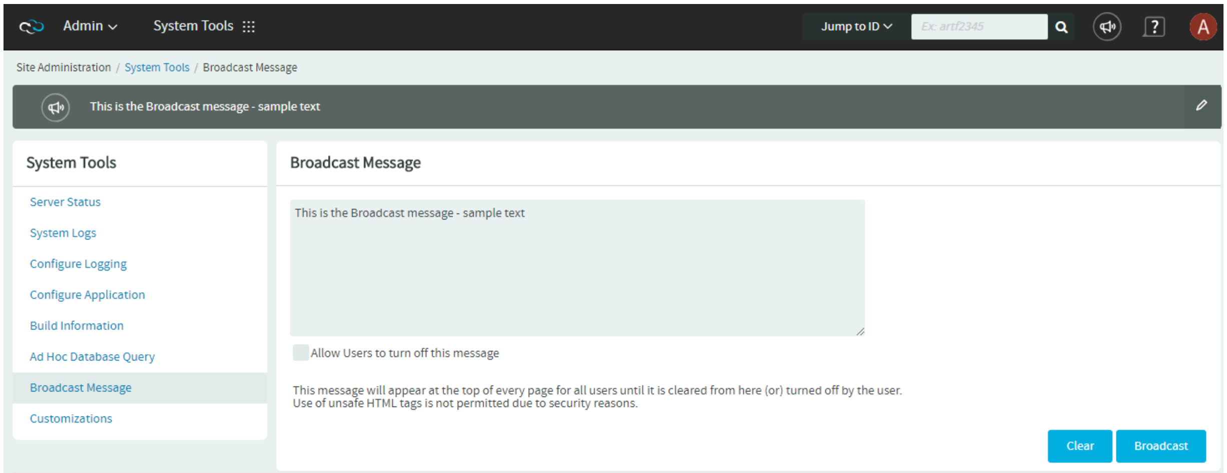
To say something to everyone who uses your site, post a site-wide broadcast message.

For example, if you plan to upgrade your site, you may want to let users know a few days before your upgrade that the site will be unavailable for a short time. The broadcast message can be viewed even without logging into TeamForge

1. Go to **My Workspace > Admin**.
2. Click **SYSTEM TOOLS** from the **Projects** menu.
3. Click **Broadcast Message**.

4. In the **Broadcast Message** page, type the message you want to broadcast and click **Broadcast**.

   The message is displayed at the top on all the pages of the site until it is deleted or replaced.



This is the Broadcast message - sample text

Broadcast Message

This is the Broadcast message - sample text

□ Allow Users to turn off this message

This message will appear at the top of every page for all users until it is cleared from here (or) turned off by the user.
Use of unsafe HTML tags is not permitted due to security reasons.

5. Select the check box **Allow Users to turn off this message** to enable users to close the broadcast message box.

To edit a broadcast message quickly, you do not need to go to the specific broadcast message page. You can edit it from any page, using the Edit icon that appears at the far right end of the broadcast message box.

You can redesign some aspects of your site to suit your organization's needs and preferences.

These instructions support only some basic types of customization. Almost infinite varieties of customization are possible. To get into specific customization options in more detail, search or post a question on the TeamForge discussion forum or talk to your CollabNet representative.

Customize TeamForge Using a Custom `.jar` File

You can customize your TeamForge site by building a maven project and uploading the customization jar file that extends or customizes TeamForge.

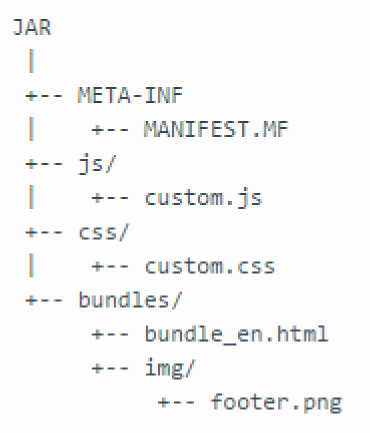
Maven projects are built and packaged to generate TeamForge customization jar and `MANIFEST.MF` files. The generated customization jar file is then uploaded to TeamForge. When you upload a customization jar, it is processed and if it has a custom event, it is registered. Later, if it has customizations, they are cached by the customization mechanism for a cost-free access at every request. Cached customizations are then served by the following three servlets:

Servlet	Description
/ctf/api/main/js-customization	Retrieves all the Javascript customizations.
/ctf/api/main/css-customization	Retrieves all the CSS customizations.
/ctf/js/modules/customization-<customization-name>/<resource-name>;	Resolves the resource relative to the main folder configured for the given customization name.

The customization mechanism provides access to all the enabled customizations in the cache.

A customization jar can contain:

- Custom events
- Javascript customizations
- CSS customizations
- Custom bundles



```
JAR
 |
+-- META-INF
|     +-- MANIFEST.MF
+-- js/
|     +-- custom.js
+-- css/
|     +-- custom.css
+-- bundles/
      +-- bundle_en.html
      +-- img/
           +-- footer.png
```

Sample jar File Structure

Here's a sample customization jar file.

While custom events are configured through an `events.xml` file in the `META-INF` folder in the jar file, Javascript, CSS and custom bundles are configured through `META-INF/MANIFEST.MF` entries.

Here's a list of `META-INF/MANIFEST.MF` entries:

MANIFEST.MF entries	Description
CTF-Customizations-Enabled	The entry to enable or disable a customization. This entry applies to custom event and customizations. • Custom event and customizations are applied if this entry is set to `True` (default value). • Custom event and customizations are not applied if this entry is set to `False`.
CTF-Customization-Name	The entry to set the name of the customization to be used for getting bundles.
CTF-Customizations-Priority	The entry to set the priority for customizations. Allows you to specify the `priority` of the customization. Customizations are sorted by the servlets based on the `priority`. Customizations

	with low priority are included at the end. The priority value could be from 1 to 100, 100 being the default value.
CTF-JS-Customization	Path to a Javascript file.
CTF-CSS-Customization	Path to CSS stylesheet.
CTF-Bundle-Customization	Path to the main bundles directory.

An Illustration of How to Add a CSS Customization

1. Build a customization project.

   Maven project structure that includes a custom stylesheet file as a resource:

   ```
   css-customization% find -type f
   ./pom.xml
   ./src/main/resources/custom/custom.css
   css-customization %
   css-customization % cat src/main/resources/custom/custom.css
   div.core-footer {
       background-image:url('/ctf/js/modules/customization-mybundles/img/footer.png');
   }
   css-customization %
   ```

   The `pom.xml` descriptor, uses the packaging plugin for setting the needed `MANIFEST.MF` properties:

   ```
   <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
     xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
     <modelVersion>4.0.0</modelVersion>

     <groupId>com.ctf.customizations.samples</groupId>
     <artifactId>css-customization</artifactId>
     <version>1.0-SNAPSHOT</version>
     <packaging>jar</packaging>
   ...
     <build>
       <plugins>
   ```

```
       <plugin>
       <groupId>org.apache.maven.plugins</groupId>
       <artifactId>maven-jar-plugin</artifactId>
       <version>2.4</version>
         <configuration>
           <archive>
             <manifestEntries>
               <CTF-Customizations-Enabled>True</CTF-Customizations-Enabled
>
               <CTF-Customization-Name>mystyles</CTF-Customization-Name>
               <CTF-CSS-Customization>custom/custom.css</CTF-CSS-Customizat
ion>
             </manifestEntries>
           </archive>
         </configuration>
       </plugin>
     </plugins>
   </build>
 ...
 </project>
```

2. Package the Maven project.

```
 css-customization % mvn package
 [INFO] Scanning for projects...
 ...
 [INFO] --- maven-jar-plugin:2.4:jar (default-jar) @ stylesheet ---
 [INFO] Building jar: /home/matias/workspaces/ctf/css-customization/target
/mystyles.jar
 [INFO] ------------------------------------------------------------------
------
 [INFO] BUILD SUCCESS
 [INFO] ------------------------------------------------------------------
------
 [INFO] Total time: 2.959s
 [INFO] Finished at: Wed May 21 20:26:28 ART 2014
 [INFO] Final Memory: 8M/105M
 [INFO] ------------------------------------------------------------------
------
 css-customization %
```

Generated jar:

```
css-customization % jar tvf target/mystyles.jar
   207 Wed May 21 20:26:28 ART 2014 META-INF/MANIFEST.MF
    17 Wed May 21 20:26:26 ART 2014 custom/custom.css
  1092 Wed May 21 20:16:06 ART 2014 META-INF/maven/com.ctf.customizations
.samples/css-customization/pom.xml
   132 Wed May 21 20:26:28 ART 2014 META-INF/maven/com.ctf.customizations
.samples/css-customization/pom.properties
css-customization %
```

Generated `MANIFEST.MF`:

```
Manifest-Version: 1.0
Built-By: matias
Build-Jdk: 1.7.0_55
CTF-Customization-Name: mystyles
CTF-CSS-Customization: custom/custom.css
CTF-Customizations-Enabled: True
Created-By: Apache Maven 3.0.4
Archiver-Version: Plexus Archiver
```

3. Upload the generated jar file as a custom event so that your customization is applied to TeamForge pages.
    1. Log on to TeamForge as a site administrator.
    2. Select **My Workspace > Admin**.
    3. Select **Projects > System Tools > Customizations** and click **Create**.
    4. Click **Choose File**, select the customization jar file and click **Add**.
4. Enable, disable, delete or download a customization.
    1. Select **My Workspace > Admin**.
    2. Select **Projects > System Tools > Customizations**.
    3. Select one or more customizations (check boxes) you want to enable or disable and click **Enable** or **Disable**.
    4. Select one or more customizations (check boxes) you want to delete and click **Delete**.
    5. Select one or more customizations (check boxes) you want to download and click **Download**.

Customize a Page, Picture, Text String, or Other Elements on Your Site

Follow these general instructions to customize a page, picture, text string, or other element on your site.

IMPORTANT: Custom branding changes can be overridden when your site is upgraded to a new version. You may have to reapply any look-and-feel modifications after an upgrade.

1. Download the sample branding files. Choose one of these files:
   - Basic branding package: Contains the files you need to do most of your branding tasks. Safest to use this file if you are doing your own branding.
   - Advanced branding package: Contains all the files that can be customized. For use when someone from CollabNet is helping you with your branding.

   NOTE: It is important that you have the most recent version of this archive as a starting point. Check that the version number at the top of the `readme.txt` file in your copy of the branding package is the same as your version of the application. If it is not the same, check www.collab.net to see if there is a more recent version.

2. In the `look` project, check out the branding repository.
3. Copy the default version of the appropriate file from the branding zip file to the equivalent directory in your local copy of the branding repository.
4. Change the file to produce the results you want. For example:
   - To change a logo on your site's home page, overwrite the `home.gif` file with a new file of the same name.
   - To change a logo on a project home page, overwrite the `project.gif` file with a new file of the same name.
5. Commit the changed files into your site's branding repository.

   IMPORTANT: Your branding repository does not have to contain all the files that are in the sample branding zip file, but the structure of your repository must be an exact mirror of the structure of the sample file set.

Customize the Home Page of Your Site

To change the content of your site's main page, replace the `home.vm` file or add either `domain_home.html` or `DomainHome.html.html` file to the `html` folder in the branding repository.

NOTE: For the general steps for changing the look and feel of a page, see Customize a Page, Picture, Text String, or Other Elements on Your Site.

The \branding\templates\sfmain\home.vm template controls the look, feel and structure of the standard home page. The default version allows users to log in and sign up for new user accounts, if CollabNet TeamForge is configured to allow user self-creation.

If the DomainHome.html.html or domain_home.html file is checked into the branding repository, the contents of the file are displayed as the site home page.

TIP: If both DomainHome.html.html and domain_home.html files exist in the repository, the contents of the DomainHome.html.html are displayed.

Edit the home.vm template to produce the page you want. You can change these objects on the site home page:

Object	Description
siteNews	The html block that shows site news. • The siteNews html block itself is not customizable. • By uncommenting this siteNews object and commenting out communityNews object, site news can be displayed across all projects in the site. • In addition to enabling the siteNews html block, you must set ENABLE_SITE_NEWS token to true if you want site news published on your site's home page.
communityNews	The html block that shows community news. By uncommenting this object, and commenting out siteNews object, community news (news from the look project) can be displayed across all projects in the site.
mostActiveProjects	The html block that shows the most active projects. The html block itself is not customizable.
displayActivityGraph	A flag that indicates that the activity graph should be displayed.
displayTeamForgeLinks	A flag that indicates that CollabNet TeamForge quick links should be displayed.

Customize the Home Page of Projects

To change the default main pages of the projects on your site, edit the project_home.vm file.

NOTE: For the general steps for changing the look and feel of a page, see Customize a Page, Picture, Text String, or Other Elements on Your Site.

Edit the `project_home.vm` template to produce the project page you want. You can change these objects on the project home page:

Object	Description
projectData	The object that contains the information about the project. It implements the interface `com.collabnet.ce.customization.IProjectData`.
adminList	The list of project administrators. Each object of the list implements the interface `com.collabnet.customization.IUserRow`.
memberList	The list of project members. Each object of the list implements the interface `com.collabnet.customation.IUserRow`.
projectMember	A flag that indicates that the user is a member of the project.
joinProjectButton	The button that contains the link to the Join Project page. It returns a `com.collabnet.ce.customization.widgets.Button`.
useCustomHomePage	A flag that indicates that the page shows the Wiki Home page instead of the standard Home page.
customHomePage	The html that displays as the Project Home page.
editCustomHomePageButton	The button that is used to edit the custom Home page. It returns a `com.collabnet.ce.customization.widgets.Button`.
projectAdmin	A flag that indicates whether or not the current user is a Project Admin.
useCustomProjectLogo	A flag that indicates that the Wiki project logo image will be used instead of the standard project logo.
customLogoPathString	The url from where the custom project logo image can be loaded.

Change Your Site’s Outgoing Emails

When you site sends out automated emails, the text of the emails can be customized to fit your site’s specific needs.

NOTE: Before customizing your site, download the branding files. See Customize a Page, Picture, Text String, or Other Elements on Your Site.

You control screen labels and messages by overriding the resource bundle keys that specify the text strings that appear in Velocity macros and JSPs.

1. In your local copy of the branding repository, create a directory called templates/mail.
2. In the `templates/mail` directory, create a file containing the custom content for an email that the system sends out.

   Give the file the same name as the equivalent sample email file in the branding files package. For example, to override the email that is sent out to new members of the site, name the file `templates/mail/user_welcome.vm`. Use Velocity syntax to identify the parts of the email, like this:

```
 ##subject
 Welcome to our TeamForge site!
 ##subject
 ##body
 Here is the content that I want to appear in emails coming from my site..
.
 ##body
```

   > **NOTE:** To customize a template in a specific language, identify the locale as an extension to the file name. For example, to create a user welcome file in Japanese, name the file `templates/mail/user_welcome_ja.vm`

3. Commit your new and changed files into the repository.

Customize Your Apache Configuration

The following instructions illustrate how you can include custom configuration to Apache and disable the same if not required.

1. Create `conf.d/httpd/httpd.conf.d` under `/opt/collabnet/teamforge/etc/` directory.
2. Include `custom.conf` under `/opt/collabnet/teamforge/etc/conf.d/httpd/httpd.conf.d/`.
3. Provision services.
   `teamforge provision`

The following warning message is displayed, which you can ignore.

```
   Custom configuration found in /opt/collabnet/teamforge/etc/conf.d/httpd/http
d.conf.d has been applied. Please be informed that such configuration may impac
t the reliability of TeamForge.
```

The following line is added to `/etc/httpd/conf/httpd.conf`:

```
Include /opt/collabnet/teamforge/etc/conf.d/httpd/httpd.conf.d/
```

1. Run the `httpd -e info` command to know the Apache configuration/syntax errors, if any.

Remove Custom Apache Configuration

1. To remove custom configuration:
   ```
   cd /opt/collabnet/teamforge/etc/conf.d/
   mv httpd/ httpd_old
   ```
2. Provision services.
   ```
   teamforge provision
   ```

Customize Your PostgreSQL Configuration

The following instructions illustrate how you can include custom configuration to PostgreSQL and disable the same if not required.

1. Create `conf.d/pgsql/pg_hba.conf.d/` under `/opt/collabnet/teamforge/etc/` directory.

   If the reporting service is running on a separate port (see Create a Single Cluster for Both Database and Datamart), create `conf.d/reports-pgsql/pg_hba.conf.d/` under `/opt/collabnet/teamforge/etc/`.

2. Include `custom.conf` under `/opt/collabnet/teamforge/etc/conf.d/pgsql/pg_hba.conf.d/`.

   If the reporting service is running on a separate port, include `custom.conf` under `/opt/collabnet/teamforge/etc/conf.d/reports-pgsql/pg_hba.conf.d/`

3. Provision services.

   ```
   teamforge provision
   ```

The following warning message is displayed, which you can ignore.

```
  Custom configuration found in /opt/collabnet/teamforge/etc/conf.d/pgsql/pg_h
ba.conf.d has been applied. Please be aware of that such configuration may impa
ct the reliability of TeamForge.
```

If the reporting service is running on a separate port:

```
   Custom configuration found in /opt/collabnet/teamforge/etc/conf.d/reports-pg
sql/pg_hba.conf.d has been applied. Please be aware of that such configuration
may impact the reliability of TeamForge.
```

Configuration settings from `custom.conf` are included in `/var/lib/pgsql/9.6/data/pg_hba.conf`.

If the reporting service is running on a separate port, configuration settings from `custom.conf` are included in `/var/lib/pgsql/9.6/reports/pg_hba.conf`.

1. Check the `postgresql.log` file for any syntax errors: `/opt/collabnet/teamforge/log/pgsql/postgresql.log`.

Remove Custom PostgreSQL Configuration

1. To remove custom configuration:
   ```
   cd /opt/collabnet/teamforge/etc/conf.d/
   mv pgsql pgsql_old
   ```

   If the reporting service is running on a separate port:

   ```
   cd /opt/collabnet/teamforge/etc/conf.d/
   mv reports-pgsql reports-pgsql_old
   ```
2. Provision services.
   ```
   teamforge provision
   ```

If SELinux is active on the server that runs your TeamForge site, configure it to allow the services that TeamForge requires.

✓ In case of same hardware upgrade using RHEL/CentOS, it is recommended to upgrade the OS to RHEL/CentOS 7.4 or later versions.

✓ Log on as root or use a root shell while setting up SELinux.

TeamForge SELinux Policies

TeamForge implements SELinux policies for most of its services such as JBoss, Apache, ETL, Tomcat and so on. However, you can use these instructions to revert these policies (not recommended) if required.

Here's a list of SELinux modules that are implemented (use the `semodule -l|grep tf_` command to see the list of TeamForge SELinux modules):

- tf_apache
- tf_branding
- tf_cvs

- tf_daemon-base
- tf_etl
- tf_integration-base
- tf_jboss
- tf_phoenix
- tf_postgresql
- tf_runtime-base
- tf_subversion
- tf_tomcat

While you can revert these policies, you can contact CollabNet Support to get help in fixing the issue with TeamForge SELinux policies.

- To Revert the TeamForge SELinux Policies:
  `/opt/collabnet/teamforge/runtime/scripts/fix_data_selinux_permissions.sh`
- If JBoss is using agents such as takipi, run the following command to apply selinux context for the takipi agent:
  ```
  semanage fcontext --add -t tf_jboss_rw_t '/opt/takipi(/.*)?'
  restorecon -R /opt/takipi
  ```

Do This If SELinux Is `disabled`

Verify SELinux mode using `getenforce` command. Do this if you have SELinux running in `disabled` mode.

1. Stop TeamForge.

   > **IMPORTANT:** Stop TeamForge on all the servers in a distributed setup.

   - If you are upgarding from TeamForge 16.7 or earlier releases:
     `/etc/init.d/collabnet stop`
   - If you are upgrading from TeamForge 16.10, 17.1, or 17.4 releases:
     `/opt/collabnet/teamforge/bin/teamforge stop`
   - If you are upgrading from TeamForge 17.8 or later releases:
     `teamforge stop`
2. Edit the file `/etc/sysconfig/selinux` and set `SELINUX=enforcing`.
3. Turn off TeamForge startup on boot.
   `chkconfig collabnet off`
4. Reboot the server and verify if SELInux is set to `enforcing` mode.

```
getenforce
```

5. Turn on TeamForge startup on boot.
   ```
   chkconfig collabnet on
   ```
6. Apply TeamForge SELinux policies.
   ```
   teamforge apply-selinux
   ```
7. Provision services.
   ```
   teamforge provision
   ```

Do This If SELinux Is `permissive`

Verify SELinux mode using `getenforce` command. Do this if you have SELinux running in `permissive` mode.

1. Set SELinux to run in `enforcing` mode again.
   ```
   setenforce 1
   ```
2. Restart TeamForge.
   ```
   teamforge restart
   ```

Memcached caches Subversion (SVN) authentication and authorization information and serves the mod_authnz_ctf module's authentication and authorization requests thereby reducing the number of SOAP calls, which in turn results in less load on the TeamForge Application Server.
Before You Begin

- See this wiki page for more information about Memcached.
- Memcached can run on the TeamForge Application Server or on a separate server (in case Subversion is on a separate server). This document assumes that you install Memcached on the TeamForge Application Server that also hosts Subversion.

Do This on the TeamForge Application Server

1. Install Memcached.

   Add the `subversion-caching` identifier to the `SERVICES` token. For example:

   ```
   localhost:SERVICES=ctfcore ctfcore-database mail etl ctfcore-datamart sear
   ch subversion cvs codesearch cliserver eventq mongodb redis rabbitmq  gerr
   it gerrit-database binary binary-database reviewboard reviewboard-database
    reviewboard-adapter subversion-caching
   ```

It is also possible to use an externally managed Memcached server. To use an externally managed Memcached server, add the `subversion-caching` service to the `SERVICES` token as shown below:

```
localhost:SERVICES=ctfcore ctfcore-database mail etl ctfcore-datamart sear
ch subversion cvs codesearch cliserver eventq mongodb redis rabbitmq  gerr
it gerrit-database binary binary-database reviewboard reviewboard-database
 reviewboard-adapter
myexternalmemcachedserver:SERVICES=subversion-caching
```

Where, `myexternalmemcachedserver` hosts the Memcached service.

2. Configure the OPTIONS key in the Memcached configuration file (`/etc/sysconfig/memcached`) and start Memcached.

   The OPTIONS key in the memcached configuration file is used to set additional options during Memcached startup. Add the -l flag to have Memcached listen to . This is an important option to consider as there is no other way to secure the installation. Binding to an internal or firewalled network interface is recommended.

   ```
   vi /etc/sysconfig/memcached
   ```

   > **IMPORTANT:** Remove the -l flag from the OPTIONS key to have Memcached listen to the server's default IP address or host name, including the 'localhost'.

3. Provision services.

   ```
   teamforge provision
   ```

TeamForge EventQ is a TeamForge capability that provides traceability for product life cycle activities such as work items, SCM commits, continuous integration (CI) builds, and code reviews.

Problem Statement and Solution

Organizations currently tend to have modern mixed-vendor, heterogeneous environments with complex lifecycles including work item, commit, review, build/test, deploy, and other tasks that are monitored or managed by stand-alone tools. The tools may be vendor-supplied or open-source, and may reside on-premises or in private, public, or hybrid clouds. These tools, however, lack the ability to associate with one another and do not lend themselves to end-to-end traceability: organizations cannot easily see the connections between activities derived from disparate lifecycle tools. Yet to achieve traceability, organizations are often forced to use an all-in-one, monolithic solution that excludes popular point tools. TeamForge EventQ offers a traceability solution that preserves the advantages of your best-of-breed tools.

While TeamForge EventQ is extensible, this version provides the following packaged adapters for connecting lifecycle activities:

Type of Service	Products
SCM/Version Control	Git, Subversion
Code Review	ReviewBoard, Gerrit, Atlassian Crucible
Build/Test	Jenkins
Issue Tracker	CollabNet TeamForge, Atlassian JIRA

TeamForge EventQ Vision

In a modern and frequently fractured development environment employing tools from different vendors, TeamForge EventQ aims to provide: Aggregation — Using packaged adapters and an extensible architecture, TeamForge EventQ obtains and stores metadata about key lifecycle activities like work items, commits, builds/tests, and code reviews across vendors and platforms. Traceability — TeamForge EventQ draws associations between activities across the entire lifecycle and enables traceability through the various steps or stages of the lifecycle.

Feature Overview

TeamForge EventQ aims to aggregate lifecycle metadata across various tools and establishes networks of associations across those lifecycle activities. The use cases for TeamForge EventQ include visibility into development activities, visibility into associations between lifecycle activities, and requirements traceability for auditing purposes.

Activity Streams: Activity streams provide a “project chronology” showing the most recent activities at the top and the oldest at the bottom. See Activity Stream for more information.

Extensibility

While TeamForge EventQ ships with some adapters, its extensibility ensures that users can write custom adapters to extend TeamForge EventQ to other work item trackers, version control, CI, and code review systems. New product classes may also be extended using Extensible Data Sources (XDS). See Extend TeamForge EventQ for more information.

Install or Upgrade TeamForge-EventQ Integartion

You can install EventQ on the TeamForge Application Server or on a separate server. For more information about installing and upgrading EventQ, see TeamForge install and upgrade instructions.

SCM Commits in TeamForge with EventQ Integration

Consider the following while integrating TeamForge and EventQ.

Uploading the Event Handler JAR File

After adding EventQ to your site, it is mandatory to upload the custom event handler JAR file to get the pre and post commit notifications. Click here for steps to add a custom event handler.

Implications of TeamForge-EventQ Integration on SCM Commits and Associations

Pre and post commit notifications will be sent from TeamForge to EventQ only if the **Association Required on Commit** is enabled for the repository and if the **require-scm-integration** is set to `true` for the EventQ application.

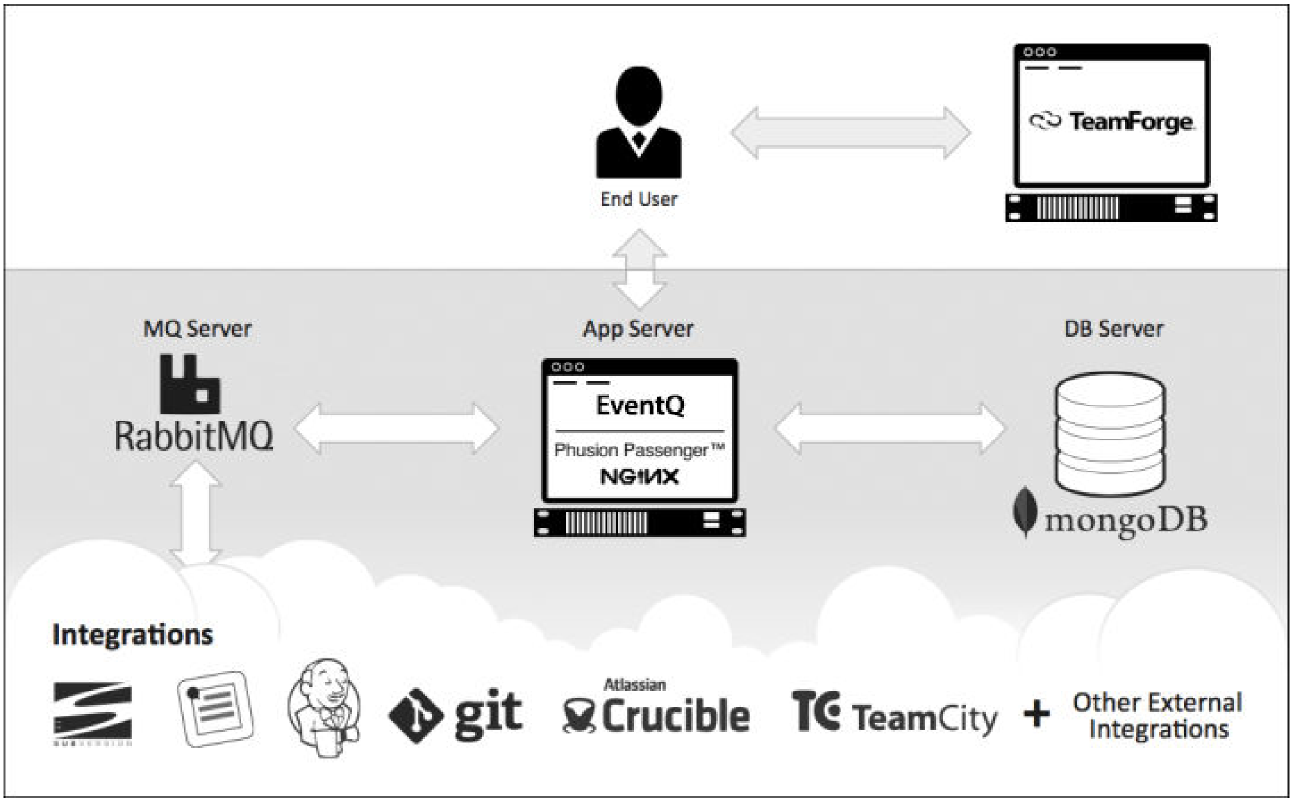
Post TeamForge-EventQ integration, SCM commits will fail in TeamForge if all the following conditions are true and if the EventQ application is down or if there are errors while processing the commit request (on the EventQ's side):

- The **Association Required on Commit** is enabled for a repository.
- The **require-scm-integration** is set to `true`.
- The commit message contains the integrated application's ID. In other words, the EventQ application's ID in this case.

SCM commits can also fail if the custom event handler exists in the TeamForge even after the removal of the EventQ application from the TeamForge site. It is highly recommended to remove the custom event handler once you remove the EventQ application.

The TeamForge EventQ architecture aims for extensibility, fault tolerance, and scalability.

End users experience TeamForge EventQ as a seamless component of TeamForge. However, under the covers EventQ is a separate application with a distinct architecture. This article intends to give an overview of that architecture to assist with installation and administration of the underlying services as well as an overview of the message queue (MQ) architecture used to integrate with and collect data from external sources.



Hosting Considerations

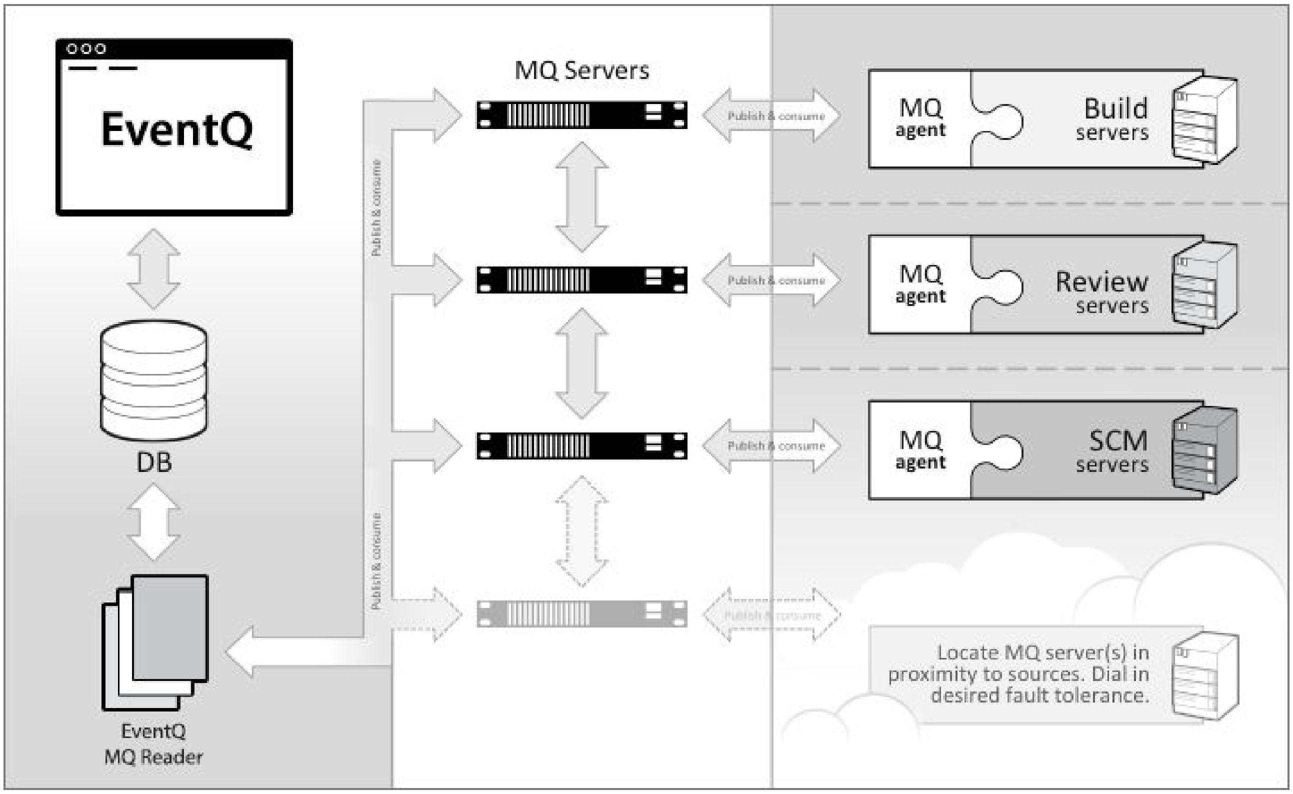
TeamForge EventQ is designed for host separation, where each component resides on different operation systems. Any component may be run on virtualized hardware. In the absence of separate hardware or virtualization, EventQ components may be collocated.

Load Balancing and Clustering

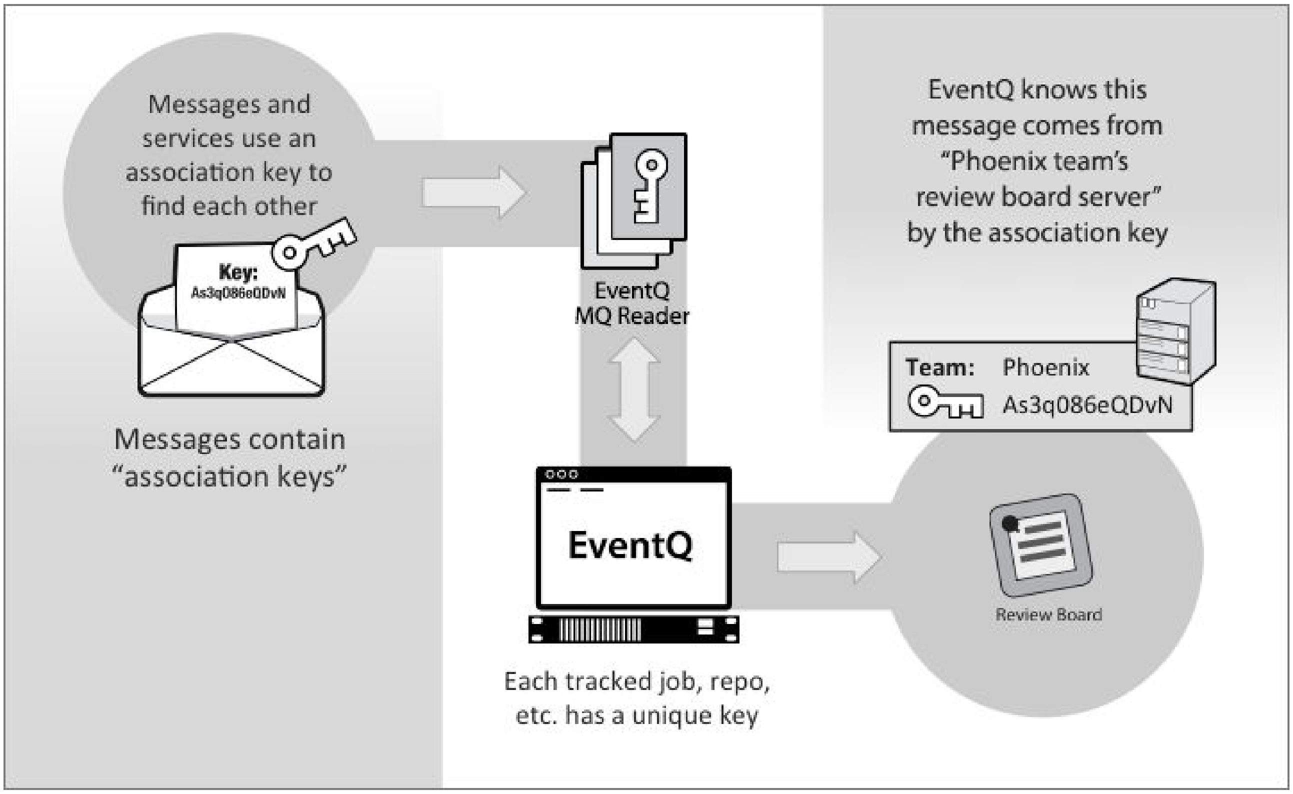
TeamForge EventQ components may be load balanced and/or clustered. The EventQ application server may be installed on multiple hosts and load balanced (session clustering is not required). MongoDB and RabbitMQ both support clustered modes as well. However, such configurations are not documented or supported and are only offered as supplemental services from CollabNet.

Integration with External Sources

TeamForge EventQ uses a MQ architecture to collect data from external sources like work item, SCM, build/CI, and code review systems. The diagram below demonstrates a clustered network of MQ servers that share data collected from external sources. When a relevant activity occurs, the external source uses an MQ agent to publish a specially formatted message to the MQ network. MQ servers can be clustered and strategically located to prevent loss of messages during to outages.



Once a message is published by an external source, it is consumed by a daemon service running on the EventQ application server. The message is parsed and the data is persisted. MQ adapters are configured with “association keys”: unique identifiers that enable EventQ to file incoming data into the appropriate source.


Messages and services use an association key to find each other
Key:
As3q086eQDvN
Messages contain "association keys"
EventQ MQ Reader
EventQ
Each tracked job, repo, etc. has a unique key
EventQ knows this message comes from "Phoenix team's review board server" by the association key
Team: Phoenix
As3q086eQDvN
Review Board

Extensibility

EventQ ships with a handful of packaged adapters to common work item, SCM, build/CI, and review systems. But what if you are using different tools in your process? You can integrate your tools by building adapters that work by publishing MQ messages that are consumed by EventQ. See the Extensibility documentation for more on writing TeamForge EventQ adapters: Extend TeamForge EventQ.

User Authentication and Authorization

TeamForge EventQ is authenticated and authorized by TeamForge, using TeamForge users, permissions and roles. To access TeamForge EventQ features, TeamForge users need additional permissions on a project by project basis.

TeamForge EventQ requires a valid TeamForge user session for access. If you try to access TeamForge EventQ when you are not logged in or when your session has expired, you will be directed to the TeamForge login page. After you successfully log in, you will be automatically redirected to the target TeamForge EventQ screen.

Permissions

TeamForge EventQ permissions are set in the “Project Admin” area on a project by project basis. The following EventQ permissions may be used in creating TeamForge roles:

- **EventQ READ** — The minimum required permission users need to view EventQ based data such as the Activity Stream and EventQ-based associations.
- **EventQ EDIT** — Required to modify EventQ data sources via the **Project Admin > Tools** interface. Users with the “EventQ EDIT” permission are implicitly granted the “EventQ READ” permission.
- **EventQ CREATE** — Required to create new EventQ data sources via the **Project Admin > Tools** interface. Users with the “EventQ CREATE” permission are implicitly granted the “EventQ READ” and “EventQ EDIT” permissions.
- **EventQ API** — Required for read and write access to EventQ HTTP APIs such as the Source API and the XDS schema API. TeamForge Site Administrators do not explicitly require this permission; however, all other project members, including Project Administrators, require this permission to use the EventQ HTTP API.
- **REPORTING API** — Required to retrieve data from the EventQ Reporting API.

> **NOTE:** TeamForge EventQ does not immediately reflect permission changes. Permissions are automatically refreshed daily. To reflect permission changes more quickly, log out of TeamForge, wait 10 minutes, and then log back in again.

TeamForge EventQ integrates with different flavors of work item, SCM/commit, build, and review servers using adapters.

EventQ Adapters

TeamForge EventQ collects metadata about key lifecycle activities from external sources like work item trackers, SCM repositories, CI/build servers, and code review systems. To do so, TeamForge EventQ relies on `adapters` to collect the metadata from sources. TeamForge EventQ packages the following adapters:

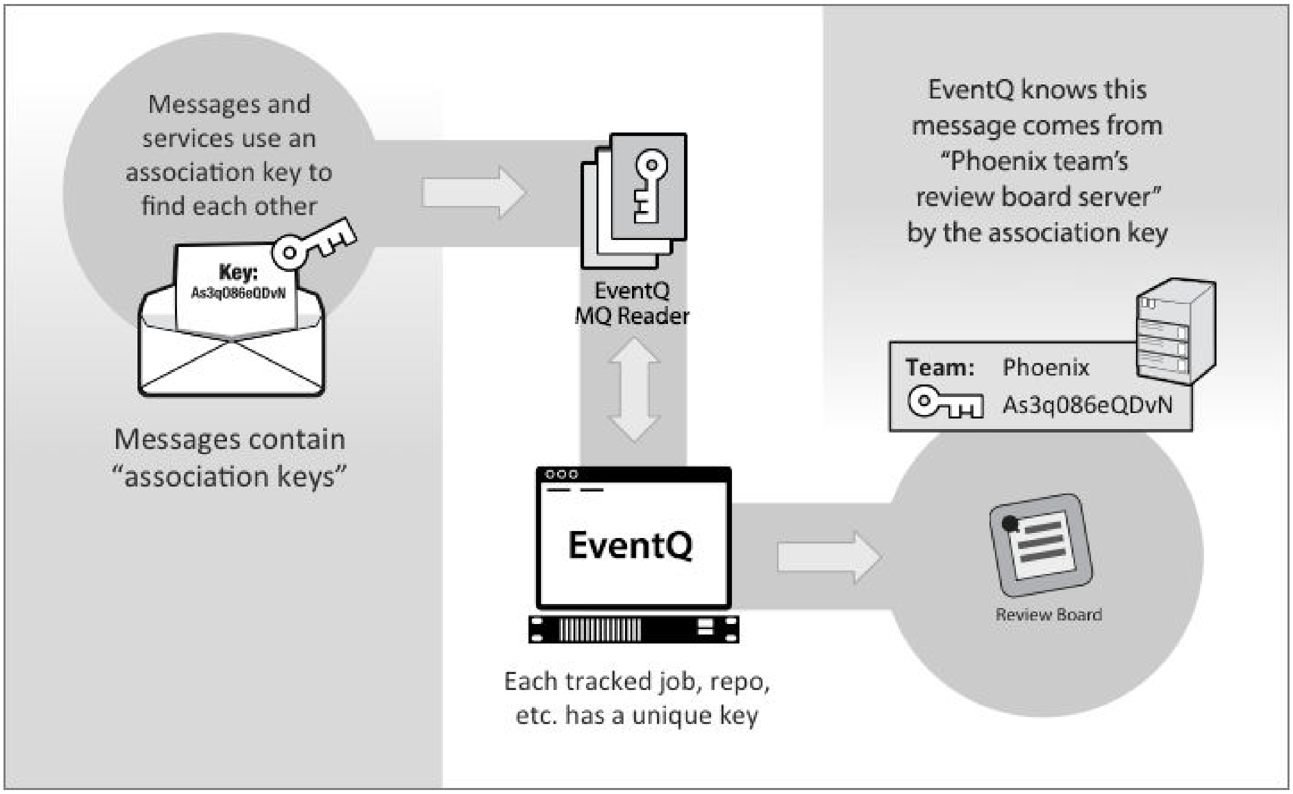
Type of Service	Product Description
SCM/Version Control	Git, Subversion
Builds/CI server	Jenkins
Code Review	ReviewBoard, Atlassian Crucible
Work Item/Issue Tracker	CollabNet TeamForge, Atlassian JIRA

Source Association Keys

TeamForge EventQ uses source association keys to uniquely identify incoming data from activity sources that it monitors.

Adapters supply data to TeamForge EventQ through the message queue (MQ) layer. When a relevant activity occurs in source systems, TeamForge EventQ adapters generate a message with key metadata about the activity and place that message on the designated MQ server. When TeamForge EventQ receives the message, it must associate it to the proper source somehow. To do this, TeamForge EventQ supplies a unique “source association key” for each external source created. Adapters must supply this unique source association key with each message for the data to correlate with the appropriate project source.

Note that commit sources based on TeamForge project repositories do not need source associations keys since they are internal and are identified by other means. All other sources require source association keys.


Messages and services use an association key to find each other
Key:
As3q086eQDvN
Messages contain “association keys”
EventQ MQ Reader
EventQ
Each tracked job, repo, etc. has a unique key
EventQ knows this message comes from “Phoenix team’s review board server” by the association key
Team: Phoenix
As3q086eQDvN
Review Board

Locate a Source Association Key

Source association key for existing activity sources are located on the corresponding source configuration page.

1. From the activity stream, click **Manage Sources**.

2. Select the desired source type toggle. TeamForge EventQ displays all existing sources.
3. Click **Edit** on the source whose key you wish to obtain. You see the configuration page for the selected source.
4. Locate the source association key for the selected source. On most browsers you can copy the key by clicking the small clipboard icon.

JIRA Integration

The TeamForge EventQ JIRA integration enhances further the extensible toolchain model of TeamForge by offering association and traceability between JIRA, TeamForge, and various orchestrated third-party tools. The TeamForge EventQ JIRA adapter enables JIRA as an alternative tracker for use with TeamForge by detecting and storing associations between JIRA issues and TeamForge managed SCM activities, like commits, builds and so on.

JIRA Version Information

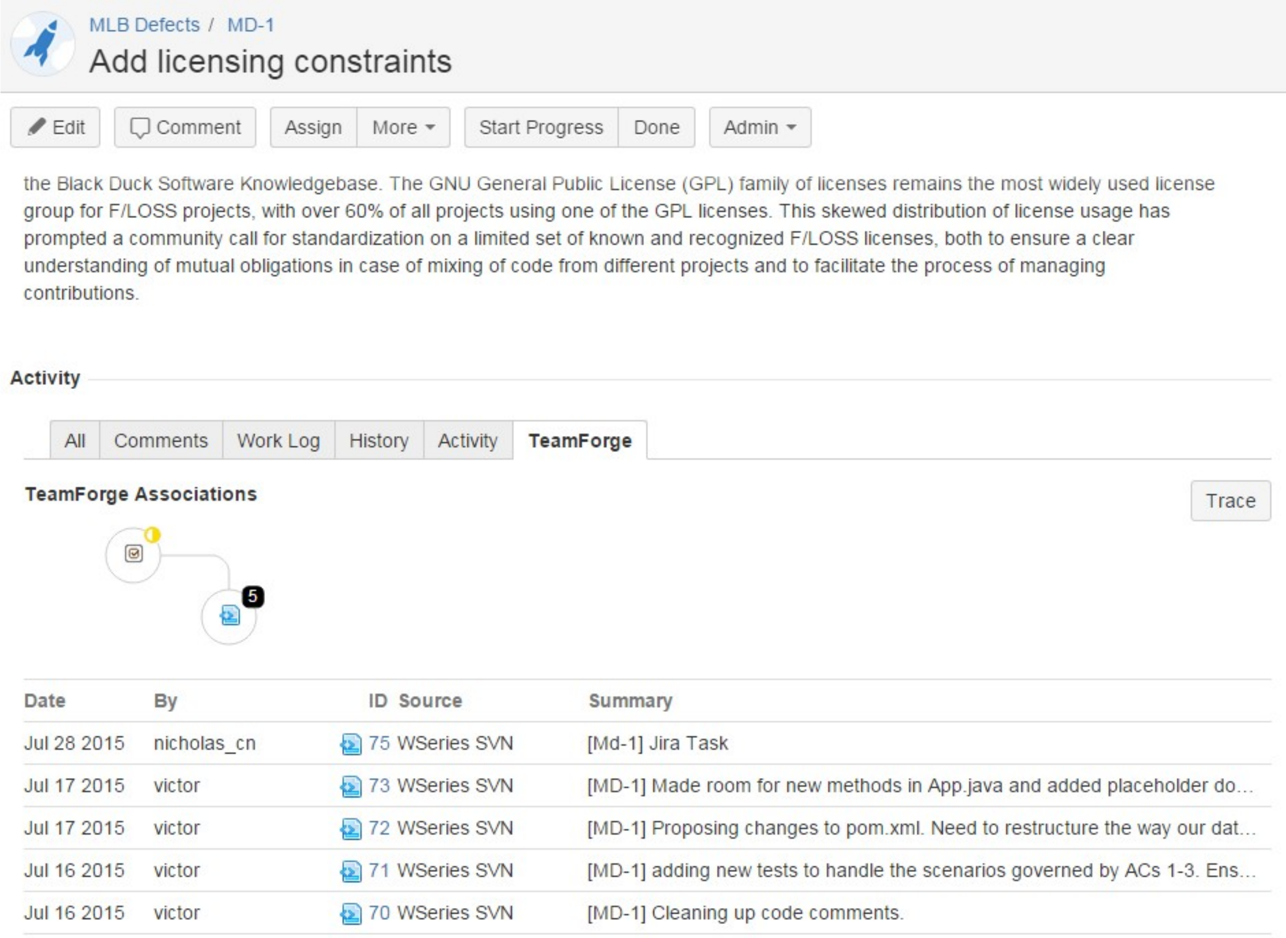
The following table lists the compatible JIRA and TeamForge Associations Add-on versions. You can integrate TeamForge with JIRA using either the TeamForge-JIRA Associations Add-on or Webhooks.

Product	Version(s) Supported	Integrations Supported
JIRA Software	6.3 - 6.4.14	TeamForge Associations Add-on 2.11-6, Webhooks
	7.0 - 7.4.4	TeamForge Associations Add-on 2.11-7, Webhooks
	7.6.0 - 8.0.2	TeamForge Associations Add-on 2.13-7, Webhooks
JIRA Software Data Center	6.3 - 6.4.14	TeamForge Associations Add-on 2.11-6, Webhooks
	7.0 - 7.4.4	TeamForge Associations Add-on 2.11-7, Webhooks
	7.6.0 - 8.0.2	TeamForge Associations Add-on 2.13-7, Webhooks
JIRA Core	6.3 - 6.4.14	TeamForge Associations Add-on 2.11-6, Webhooks
	7.0 - 7.4.4	TeamForge Associations Add-on 2.11-7, Webhooks
	7.6.0 - 8.0.2	TeamForge Associations Add-on 2.13-7, Webhooks

Integrate TeamForge and JIRA Using the Associations Add-on

This method of integration is based on TeamForge-JIRA associations add-on and is recommended for on-premises installation of JIRA. The TeamForge EventQ JIRA integration brings associations and traceability to JIRA. The TeamForge EventQ JIRA adapter enables JIRA as an alternative tracker for use with TeamForge by detecting and storing associations between JIRA issues and TeamForge managed SCM activities, like commits. Traceability is therefore possible between JIRA, TeamForge, and various EventQ-based third-party tools.

Inside TeamForge, JIRA issues are visible in activity streams and association visualizations. Within JIRA, all the SCM related activities for the particular JIRA issue are tracked and displayed under the *TeamForge* tab on the JIRA issue page.



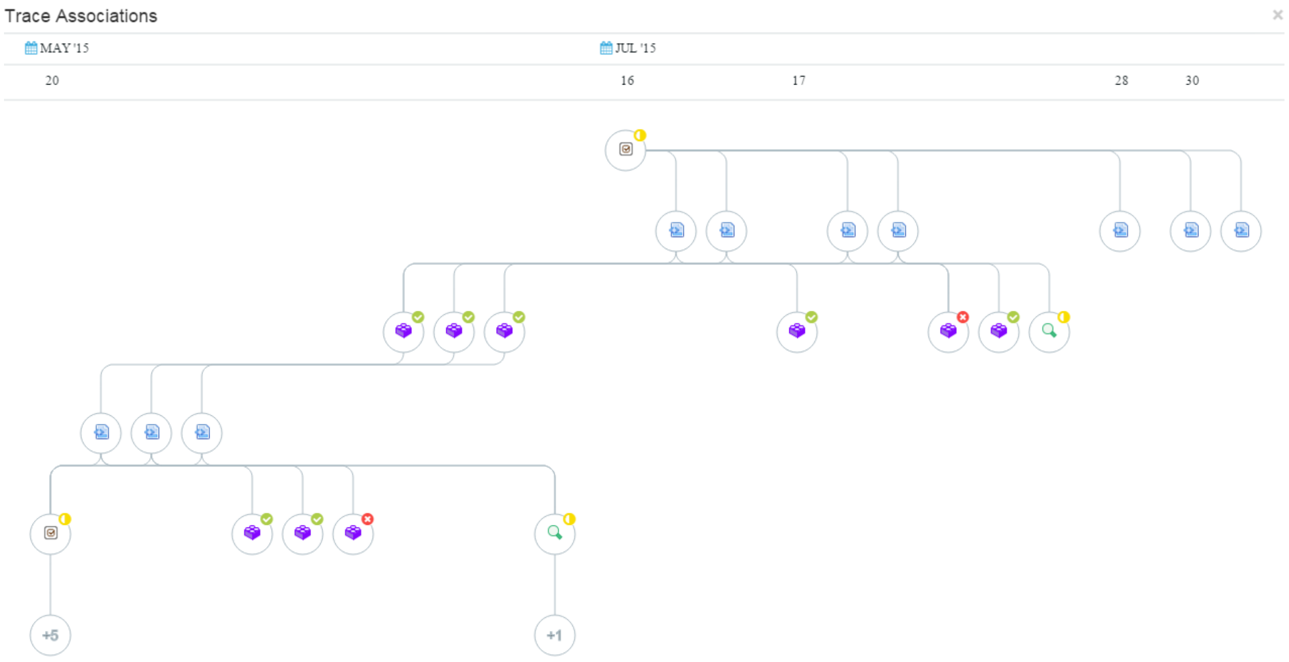
Add licensing constraints

the Black Duck Software Knowledgebase. The GNU General Public License (GPL) family of licenses remains the most widely used license group for F/LOSS projects, with over 60% of all projects using one of the GPL licenses. This skewed distribution of license usage has prompted a community call for standardization on a limited set of known and recognized F/LOSS licenses, both to ensure a clear understanding of mutual obligations in case of mixing of code from different projects and to facilitate the process of managing contributions.

TeamForge Associations

Date	By	ID	Source	Summary
Jul 28 2015	nicholas_cn	75	WSeries SVN	[Md-1] Jira Task
Jul 17 2015	victor	73	WSeries SVN	[MD-1] Made room for new methods in App.java and added placeholder do…
Jul 17 2015	victor	72	WSeries SVN	[MD-1] Proposing changes to pom.xml. Need to restructure the way our dat…
Jul 16 2015	victor	71	WSeries SVN	[MD-1] adding new tests to handle the scenarios governed by ACs 1-3. Ens…
Jul 16 2015	victor	70	WSeries SVN	[MD-1] Cleaning up code comments.

The **TeamForge** tab provides a summary of associations and also details of a full listing of associations. This is a listing of immediate associations, activities with direct relationships to the JIRA issue at hand. To explore the chain of associations, click the **Trace** button. The traceability chain that includes commits, builds, reviews, deploys, and other XDS-based integration is displayed. On the **Trace Associations** page, clicking any node lights up all the association paths to that node.In addition, a small pop-up appears with details about the node in question. You can use **SET AS TARGET** button to expand the associations from the selected point. Another function exists to **SEE** more details about the node, opening a new browser tab.



The TeamForge EventQ JIRA adapter is packaged as a JIRA “add-on”. Once installed and configured, the add-on supplies issue tracker “work item” metadata to TeamForge EventQ. The JIRA adapter needs configuration on a per project basis in JIRA, such that JIRA (many) to TeamForge (one) project mappings are established. It is therefore required to configure a TeamForge project URL and a set of TeamForge credentials for each JIRA project. Note that once configured with the aforementioned URL and credentials, the JIRA adapter will create and manage the needed sources and source associations keys. For more about source association keys, see Source Association Keys.

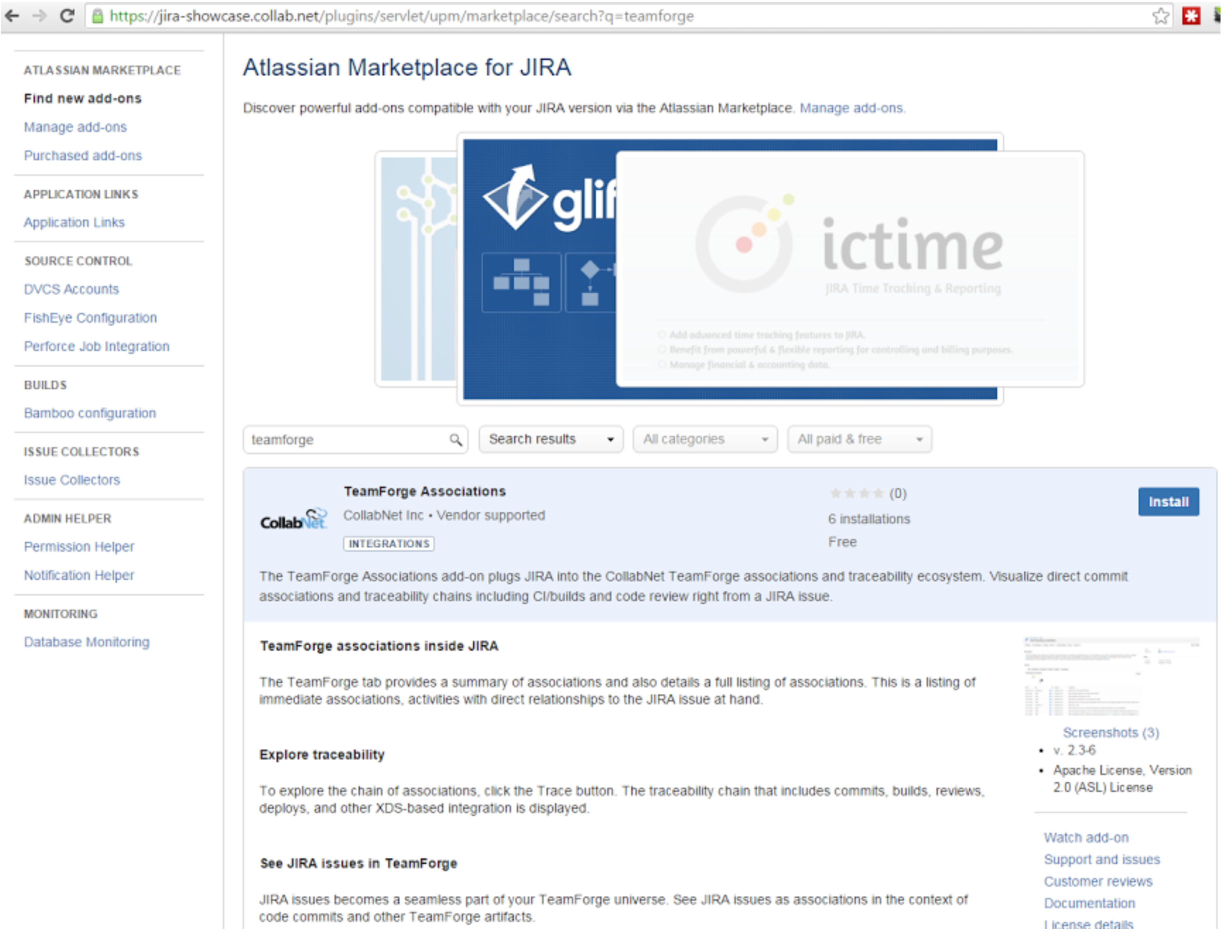
Install a JIRA Adapter

Use the JIRA adapter to notify EventQ of updates to JIRA issues and to vizualize the associations between JIRA and other tools. The JIRA adapter must be installed once on each JIRA server you wish to connect to TeamForge EventQ. Install the adapter using the JIRA add-on Manager. Refer to the Atlassian Marketplace for supported versions.

Install the TeamForge-JIRA Associations Add-on

1. As a privileged JIRA user, navigate to **Administration > Add-ons > Find new add-ons**.
2. Search the marketplace for the **TeamForge Associations** add-on for JIRA. Type ‘TeamForge Associations’ and search.

3. Click **Install**.



Atlassian Marketplace for JIRA

Discover powerful add-ons compatible with your JIRA version via the Atlassian Marketplace. Manage add-ons.

TeamForge Associations

CollabNet Inc • Vendor supported

(0)

6 installations

Free

The TeamForge Associations add-on plugs JIRA into the CollabNet TeamForge associations and traceability ecosystem. Visualize direct commit associations and traceability chains including CI/builds and code review right from a JIRA issue.

TeamForge associations inside JIRA

The TeamForge tab provides a summary of associations and also details a full listing of associations. This is a listing of immediate associations, activities with direct relationships to the JIRA issue at hand.

- v. 2.3.6
- Apache License, Version 2.0 (ASL) License

Explore traceability

To explore the chain of associations, click the Trace button. The traceability chain that includes commits, builds, reviews, deploys, and other XDS-based integration is displayed.

See JIRA issues in TeamForge

JIRA issues becomes a seamless part of your TeamForge universe. See JIRA issues as associations in the context of code commits and other TeamForge artifacts.

Upgrade the TeamForge-JIRA Associations Add-on

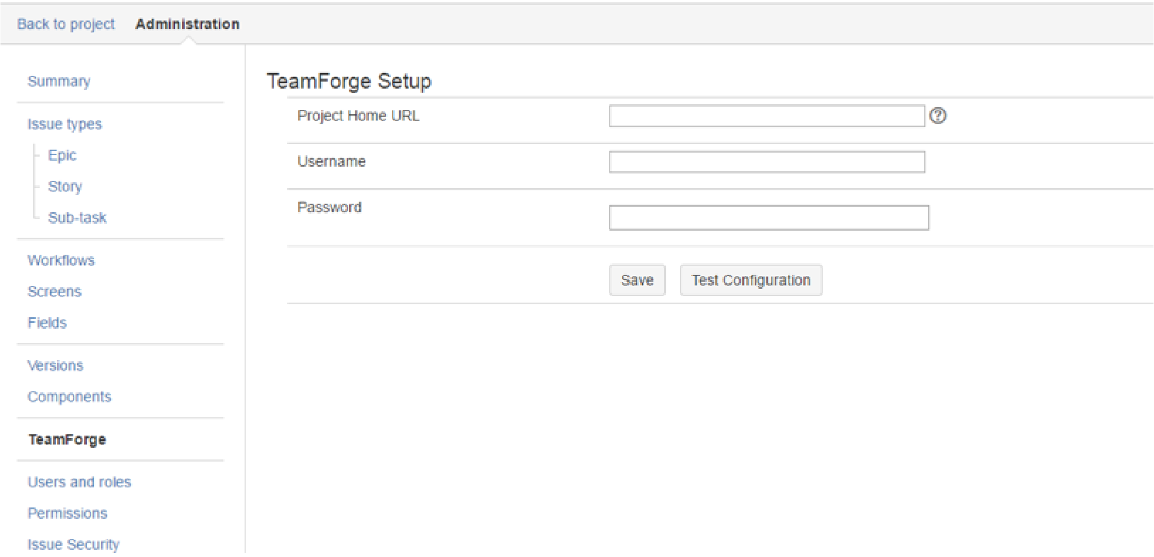
New versions of the TeamForge-JIRA associations add-on will be visible in the **Manage Add-ons** section (**Administration > Add-ons > Manage Add-ons**).

Configure a JIRA Adapter

Configure "TeamForge Associations" to notify *TeamForge EventQ* about work item activities.

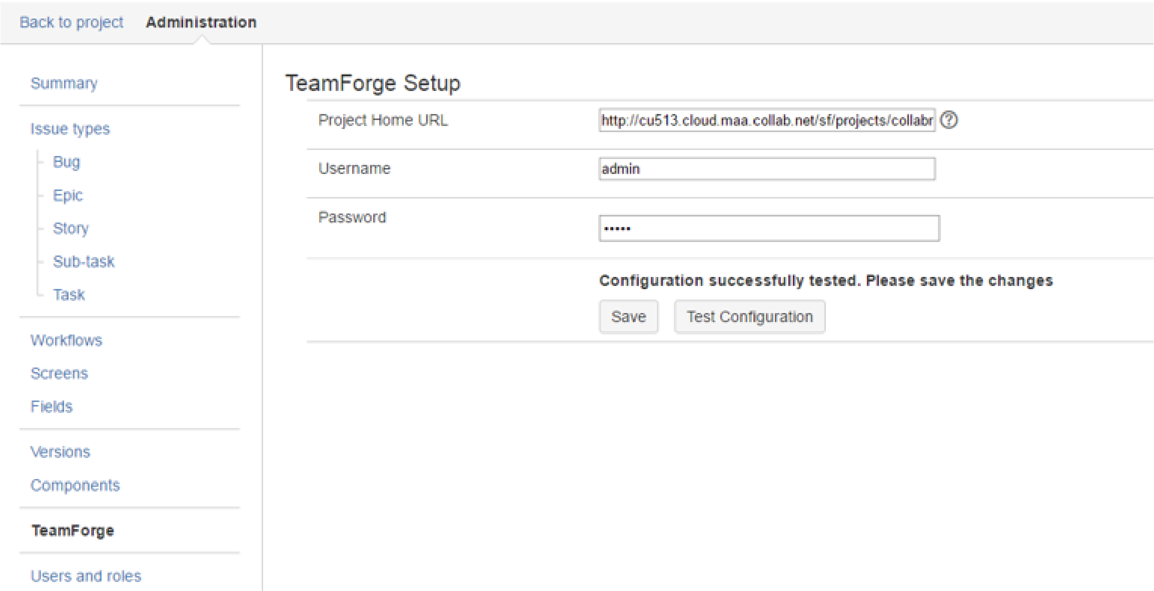
Before you start configuration, make sure that you have installed the TeamForge Associations add-on: verify that **TeamForge Associations** is present in the **User-installed Add-ons** list.

1. Configure the add-on from the **Project Administration** page to notify TeamForge-EventQ about the work item activities.
   1. Navigate to **Administration > TeamForge**.



The **TeamForge Setup** page appears.

2. Find the TeamForge project to associate with and enter its URL in the **Project Home URL** field. Obtain the TeamForge `Project Home URL` by navigating to the target TeamForge project and clicking **Project Home**. Copy the URL from the browser.
3. Using your valid TeamForge login credentials, click **Test Configuration**. The user whose credentials are supplied must have API permissions in both TeamForge and EventQ, or should be a `Project Admin`.



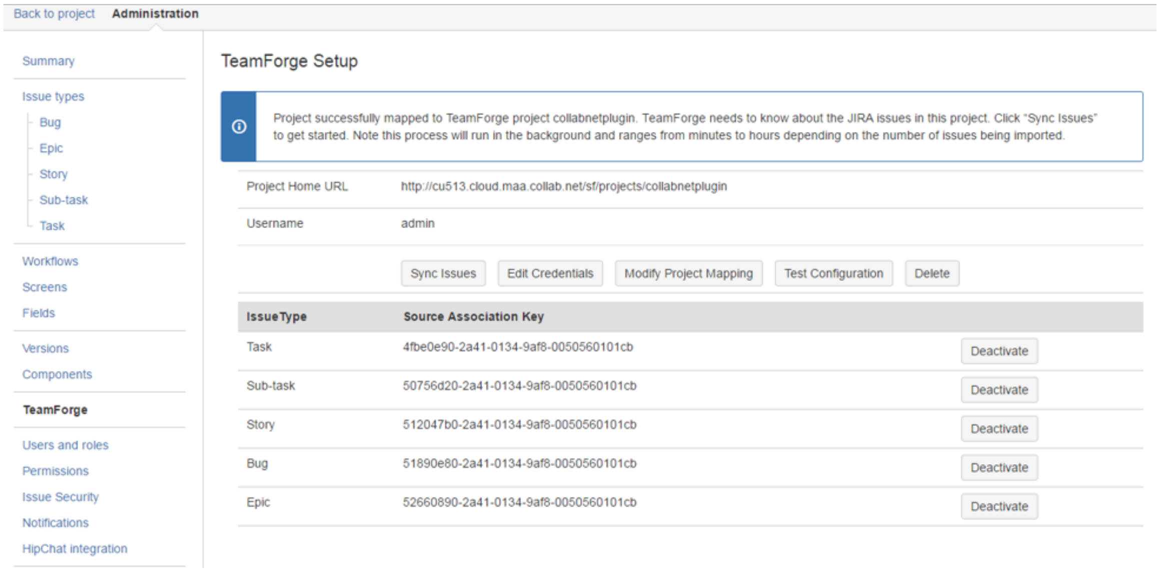
Project Home URL: http://cu513.cloud.maa.collab.net/sf/projects/collabr

Username: admin

Password: •••••

Configuration successfully tested. Please save the changes

4. Click **Save** to save the configuration. When you save your configuration, for each issue type in the JIRA project a corresponding source is created in the TeamForge EventQ.



TeamForge Setup

Project successfully mapped to TeamForge project collabnetplugin. TeamForge needs to know about the JIRA issues in this project. Click "Sync Issues" to get started. Note this process will run in the background and ranges from minutes to hours depending on the number of issues being imported.

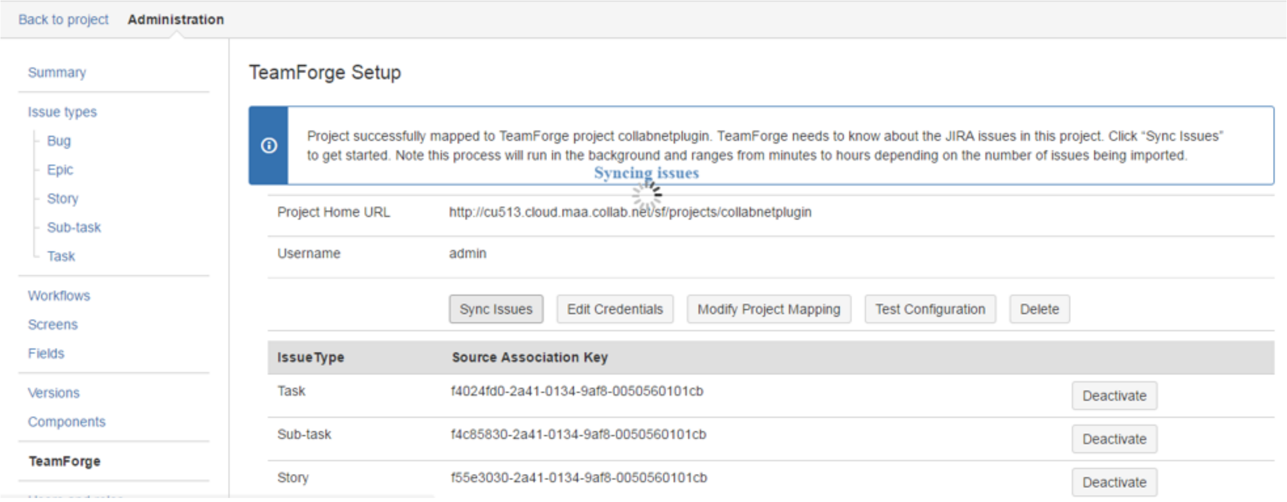
Project Home URL	http://cu513.cloud.maa.collab.net/sf/projects/collabnetplugin
Username	admin

IssueType	Source Association Key	
Task	4fbe0e90-2a41-0134-9af8-0050560101cb	Deactivate
Sub-task	50756d20-2a41-0134-9af8-0050560101cb	Deactivate
Story	512047b0-2a41-0134-9af8-0050560101cb	Deactivate
Bug	51890e80-2a41-0134-9af8-0050560101cb	Deactivate
Epic	52660890-2a41-0134-9af8-0050560101cb	Deactivate

Sync Existing JIRA Issues into TeamForge

To associate TeamForge objects to existing JIRA issues, click **Sync Issues**.

This step bootstraps JIRA issue data from the current project into the TeamForge EventQ data store. Note that this process may take several minutes (even hours) to complete, depending on the number of issues in the JIRA project.



TeamForge Setup

Project successfully mapped to TeamForge project collabnetplugin. TeamForge needs to know about the JIRA issues in this project. Click "Sync Issues" to get started. Note this process will run in the background and ranges from minutes to hours depending on the number of issues being imported.

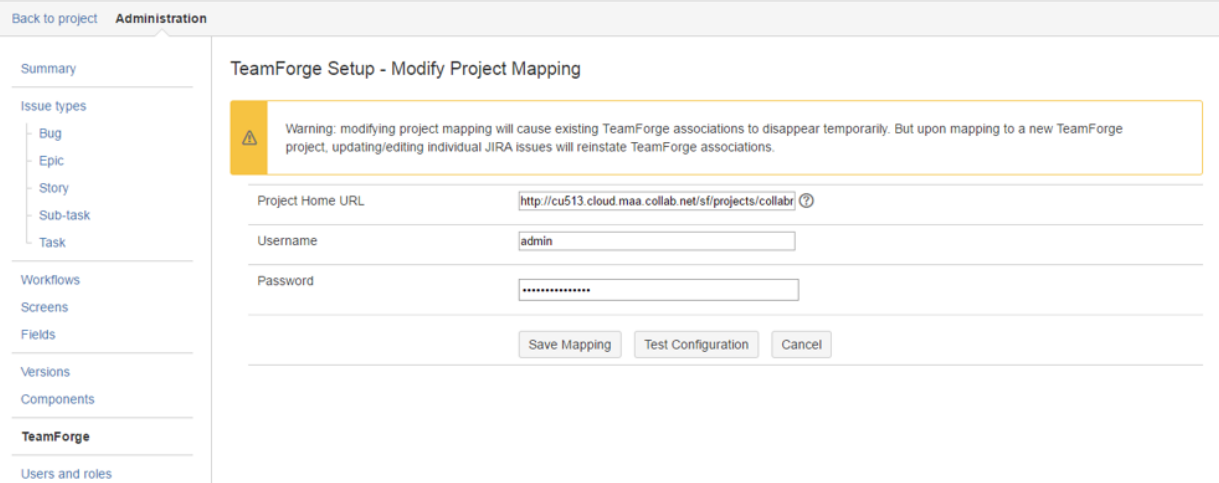
Project Home URL	http://cu513.cloud.maa.collab.net/sf/projects/collabnetplugin
Username	admin

IssueType	Source Association Key	
Task	f4024fd0-2a41-0134-9af8-0050560101cb	Deactivate
Sub-task	f4c85830-2a41-0134-9af8-0050560101cb	Deactivate
Story	f55e3030-2a41-0134-9af8-0050560101cb	Deactivate

Test or Modify Configuration

1. Test the current configuration by clicking **Test Configuration**. The system indicates whether the supplied URL and credentials work properly.
2. To modify the supplied credentials, click **Edit Credentials**.

   1. Modify either the **Username** or **Password** fields.
   2. Click **Test Configuration** to verify your settings.
   3. Click **Save** once testing is satisfactory.
3. To modify the TeamForge project mapping, click **Modify Project Mapping**.



TeamForge Setup - Modify Project Mapping

⚠ Warning: modifying project mapping will cause existing TeamForge associations to disappear temporarily. But upon mapping to a new TeamForge project, updating/editing individual JIRA issues will reinstate TeamForge associations.

Project Home URL: http://cu513.cloud.maa.collab.net/sf/projects/collabr

Username: admin

Password: ••••••••••••••••

   1. Modify the **ProjectHome URL** field to reflect the desired TeamForge project.
   2. If required, update the credentials to match the TeamForge project.
   3. Click **Test Configuration** to verify that the supplied user had adequate permissions in the selected TeamForge project.
   4. Click **Save Mapping** once testing is satisfactory.

> **NOTE:** Upon successfully re-mapping the JIRA project, run **Sync Issues** again to populate the TeamForge data store.

Delete Project Mapping Configuration

Click **Delete** if you wish to completely disassociate the JIRA project from the configured TeamForge project.

> **WARNING:** Deleting project mapping abandons all existing association data. The JIRA project can be mapped to another project subsequently, but existing associations are lost. Do this if you were testing the integration using a production JIRA project but now wish to remove any test association data.

Integrate TeamForge and JIRA Using WebHooks

The TeamForge Integration for JIRA® Webhooks collects issue data from your JIRA Serve or JIRA Cloud for purposes of associations, traceability, and event reporting.

This integration relies on JIRA's webhooks mechanism to send issue data to TeamForge EventQ for purposes of associations, traceability, and event reporting. Being webhooks-based, there is no adapter or software to install; simply configure TeamForge and JIRA webhooks to communicate with each other and start creating associations, traceability, and events reports based on JIRA data within TeamForge. With no adapter to install, TeamForge Webhooks Integration for JIRA is ideal for use with JIRA Cloud.

Viewing Associations and Traceability

Because there's no need to install plugins, there is no mechanism to visualize associations between TeamForge objects and JIRA issues from within JIRA. To visualize associations, please use the Source Code tool in TeamForge.

Prerequisites

This integration is subject to limitations imposed by JIRA Webhooks and JIRA Cloud. See Atlassian® Documentation for specifications. In particular:

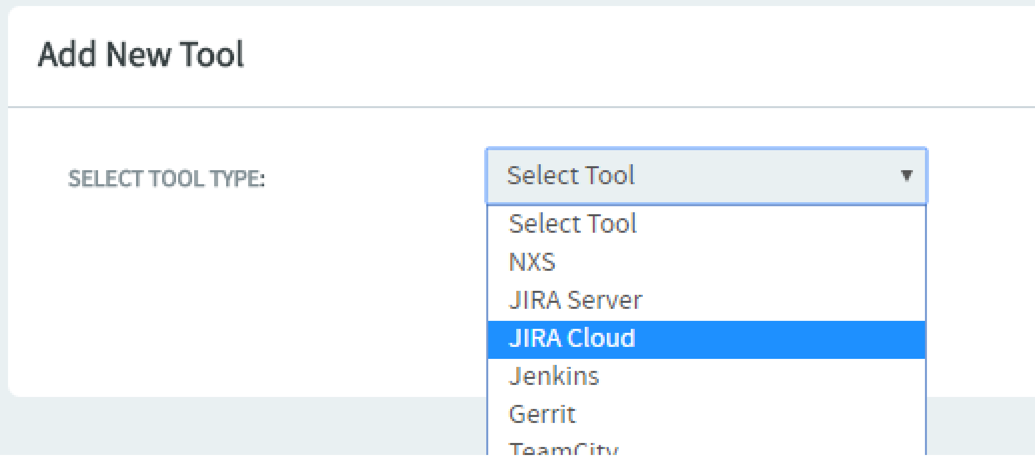
- TeamForge HTTP or HTTPS access must be configured on ports 80 or 443 respectively. JIRA Cloud does not support alternate port configurations for webhooks.
- If using HTTPS, TeamForge must be configured with a **valid SSL certificate signed by a certificate authority** to be an eligible JIRA Cloud hooks consumer. Self-signed certificates are not supported by JIRA Cloud. This is a limitation JIRA Cloud imposes on webhooks consumers.
- TeamForge Integration for JIRA Webhooks is a point-to-point integration, therefore the following ports must be configured for access:

Port	From	To	Description
443/80	JIRA web server	TeamForge web server	JIRA sends webhooks payload to TeamForge

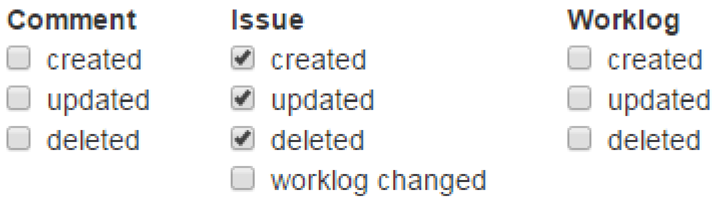
Configuring the JIRA Cloud WebHooks

This integration documentation walks you through the creation of one or more webhooks triggers in JIRA. Each webhook should be scoped to a specific JIRA project issue type. It is therefore required that a separate webhook be set up for each issue type you wish to collect data from for purposes of associations and event reporting.

1. In TeamForge, navigate to the Project that houses the source code repositories you wish to associate to the JIRA issues in question.
2. Select **Project Home > Project Admin > Tools** and click **Add Tool**.
3. Select **JIRA Cloud** from the **Select Tool Type** drop-down list.



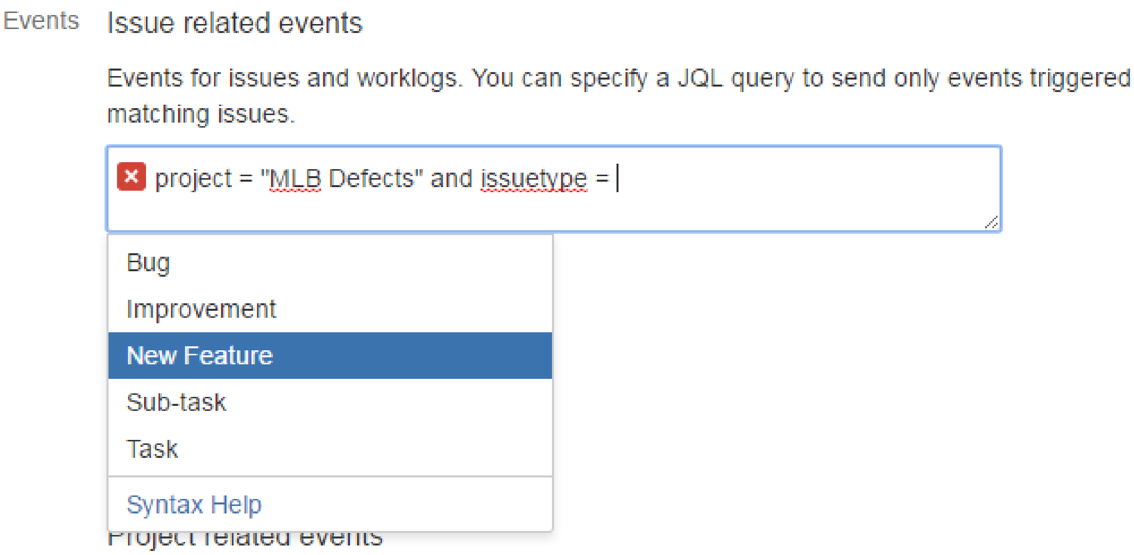
4. Type the **Display Name**. The **Display Name** is used to differentiate your JIRA projects/instances associated to this TeamForge project.
5. Select the **Include Traceability** check box.
6. Type the **Source Name**. This value differentiates the sources you define for this JIRA Project/instance. It is recommended that you name your new source to match the Issue Type in JIRA you're configuring currently.
7. Click **Generate Webhook**. You will be presented with a unique Webhook URL. This URL should only be used once, as it maps to a specific Project and Issue Type in JIRA.
8. Copy the entire Webhook URL and retain it for future steps.
9. Click **Save** to finish adding your tool (the new JIRA project/instance).
10. In JIRA, as a privileged user navigate to **System Administration > Webhooks**.
11. Click **Create a Webhook** and give it a descriptive name and toggle "Status" to **Enabled**.
12. Paste the Webhook URL into **URL** field.
13. Under the Events section, check the following issue events: `created`, `updated`, `deleted`.



Comment	Issue	Worklog
□ created	☑ created	□ created
□ updated	☑ updated	□ updated
□ deleted	☑ deleted	□ deleted
	□ worklog changed	

14. Under the **Events** section, input JQL to narrow the events triggering webhooks payloads to TeamForge as follows:
    - Supply a Project by entering: **project = [select project name]**.
    - Supply an issue type by entering: **and issuetype = [select issue type]**.

> **TIP:** Notice that a drop-down menu assists with project and issue type selection once the = symbol has been inputted, followed by an empty space.

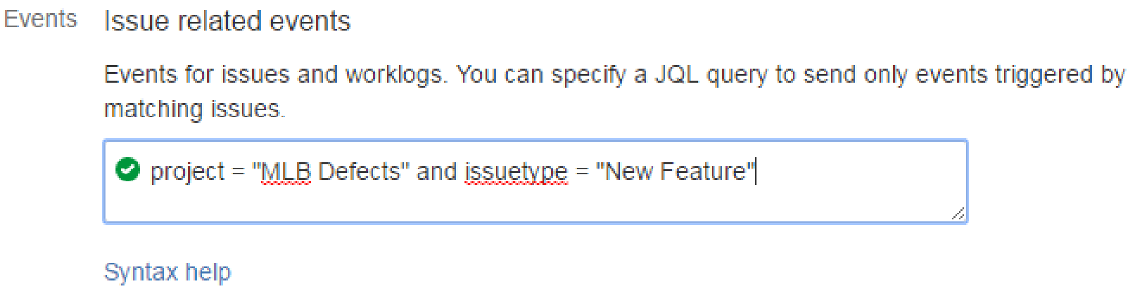


Events Issue related events

Events for issues and worklogs. You can specify a JQL query to send only events triggered matching issues.

☒ project = "MLB Defects" and issuetype =

A green tick mark indicates you’ve properly configured the webhook.



Events Issue related events

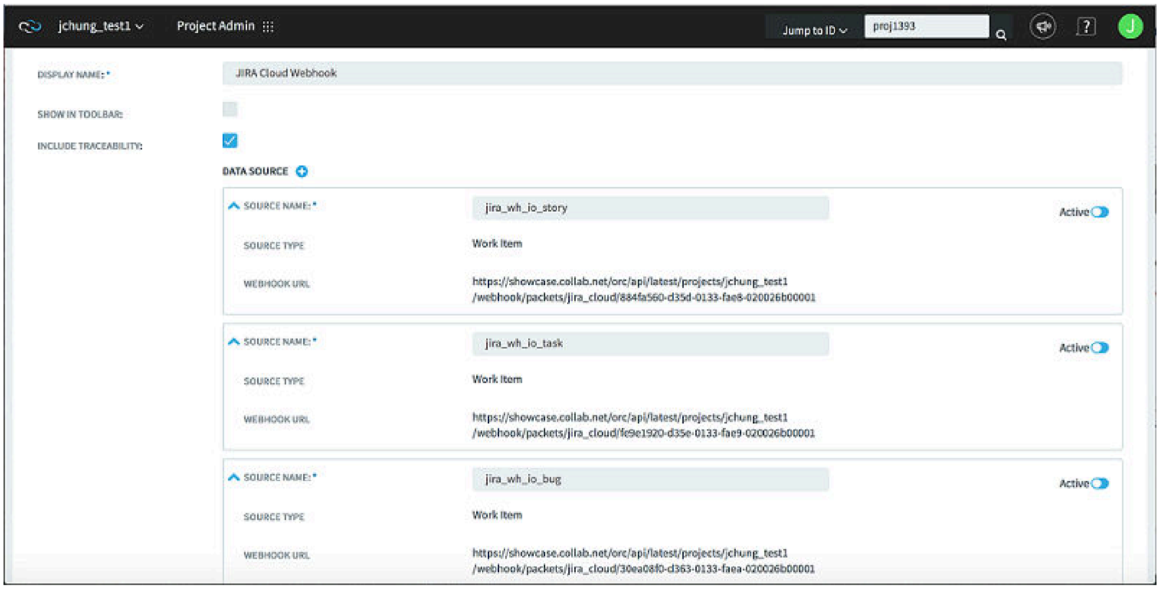
Events for issues and worklogs. You can specify a JQL query to send only events triggered by matching issues.

✓ project = "MLB Defects" and issuetype = "New Feature"

15. Click **Save**. Webhook configuration is now complete for the issue type in question. You should see the newly configured webhook in the listing within the JIRA **System Administration > Webhooks** page.
16. To verify the configuration, create a new issue matching the issue type in question and you should see a new representative activity in the EventQ Activity Stream. It is recommended that you house all issue types relating to a single JIRA project under the same “Tool” in TeamForge, creating a separate “source” for each issue type.
    1. Navigate to the newly created JIRA Cloud Tool in TeamForge: **Project Admin > Tools > [Click the tool created in the steps above]**.

       > **NOTE:** You can see that your existing source is housed here.

    2. Click **Data source +** to create a second source to house the second JIRA issue type.
    3. Continue with the steps as above.



DISPLAY NAME: * JIRA Cloud Webhook

SHOW IN TOOLBAR: □

INCLUDE TRACEABILITY: ☑

DATA SOURCE

SOURCE NAME: * jira_wh_io_story Active
SOURCE TYPE Work Item
WEBHOOK URL https://showcase.collab.net/orc/api/latest/projects/jchung_test1
/webhook/packets/jira_cloud/884fa560-d35d-0133-fae8-020026b00001

SOURCE NAME: * jira_wh_io_task Active
SOURCE TYPE Work Item
WEBHOOK URL https://showcase.collab.net/orc/api/latest/projects/jchung_test1
/webhook/packets/jira_cloud/fe9e1920-d35e-0133-fae9-020026b00001

SOURCE NAME: * jira_wh_io_bug Active
SOURCE TYPE Work Item
WEBHOOK URL https://showcase.collab.net/orc/api/latest/projects/jchung_test1
/webhook/packets/jira_cloud/30ea08f0-d363-0133-faea-020026b00001

17. Repeat the above steps for each desired issue type.

Associate JIRA Issues to Version Control Commits

Associations between JIRA® issues and TeamForge-aware version control commits can be created via commit message references. Commits to TeamForge project repositories and external repositories that have been configured as EventQ sources are also supported.

Creating Associations

1. Using your desired terminal or IDE, instantiate a version control commit to your repository of choice.
2. In the commit message, make reference to JIRA ID(s) to which you wish to associate the commit surrounded by square brackets.,br>
   Sample Commit Messages
   ```
   “[DEMO-123] This commit message will result in an association between
   JIRA issue DEMO-123 and this commit.”

   “[DEMO-123, DEMO-124] Here I’m associating two JIRA issues with project
   identifiers DEMO.”
   ```

Viewing Associations Inside JIRA

1. Navigate to the desired JIRA issue details page.
2. Click the **TeamForge** tab.
3. A list of immediate associations appears. In other words, these objects are directly associated to the JIRA issue in question.

4. Click **Trace** to view the first three levels of the traceability chain.

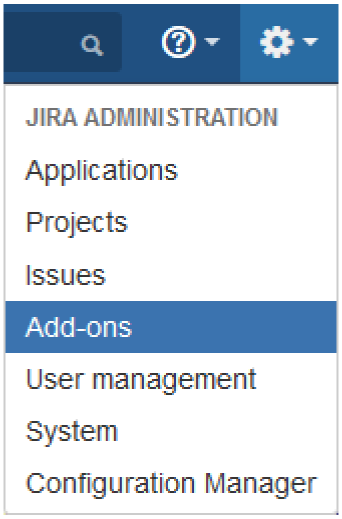
    - ✓ All activities are mapped chronologically.
    - ✓ Lines indicate direct associations.
    - ✓ Use the “+” icon to explore further levels of traceability.

Remove TeamForge Project Mappings

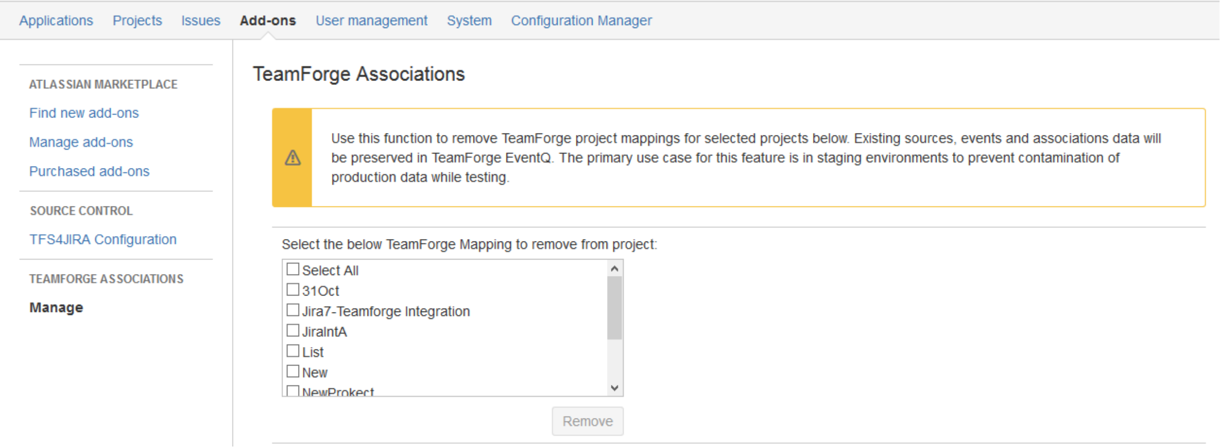
As a JIRA site administrator, you can disable active JIRA-TeamForge project mappings for one or more or all the projects (**Administration > Add-ons > TeamForge Associations**) if you want to prevent a JIRA server that’s mirrored into a staging/testing environment from triggering events back to TeamForge (when there are changes to issues in JIRA projects that are mapped to TeamForge) thereby polluting the production TeamForge event data store.

Use this feature to disable all production JIRA-TeamForge mappings in stage environments and then test the TeamForge Associations add-on by creating a new mapping between a staging JIRA server and TeamForge server.

1. Log on to the JIRA server as a site administrator.
2. Select **JIRA Administration > Add-ons**.



3. Select **TeamForge Associations > Manage**.



TeamForge Associations

Use this function to remove TeamForge project mappings for selected projects below. Existing sources, events and associations data will be preserved in TeamForge EventQ. The primary use case for this feature is in staging environments to prevent contamination of production data while testing.

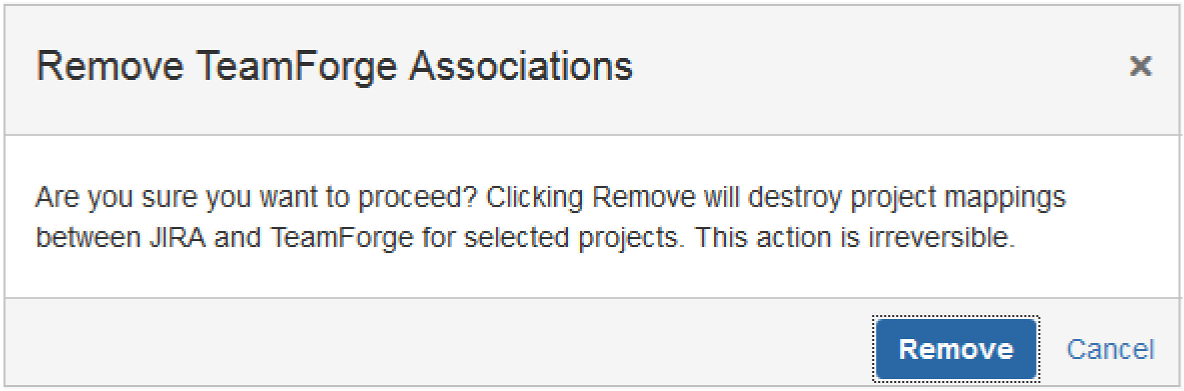
Select the below TeamForge Mapping to remove from project:

□ Select All
□ 31Oct
□ Jira7-Teamforge Integration
□ JiraIntA
□ List
□ New
□ NewProkect

4. Select one or more projects from the list to remove the project mapping.

> **TIP:** You can select the **Select All** check box to select all the projects.

5. Click **Remove**. A confirmation message is displayed.



Remove TeamForge Associations

Are you sure you want to proceed? Clicking Remove will destroy project mappings between JIRA and TeamForge for selected projects. This action is irreversible.

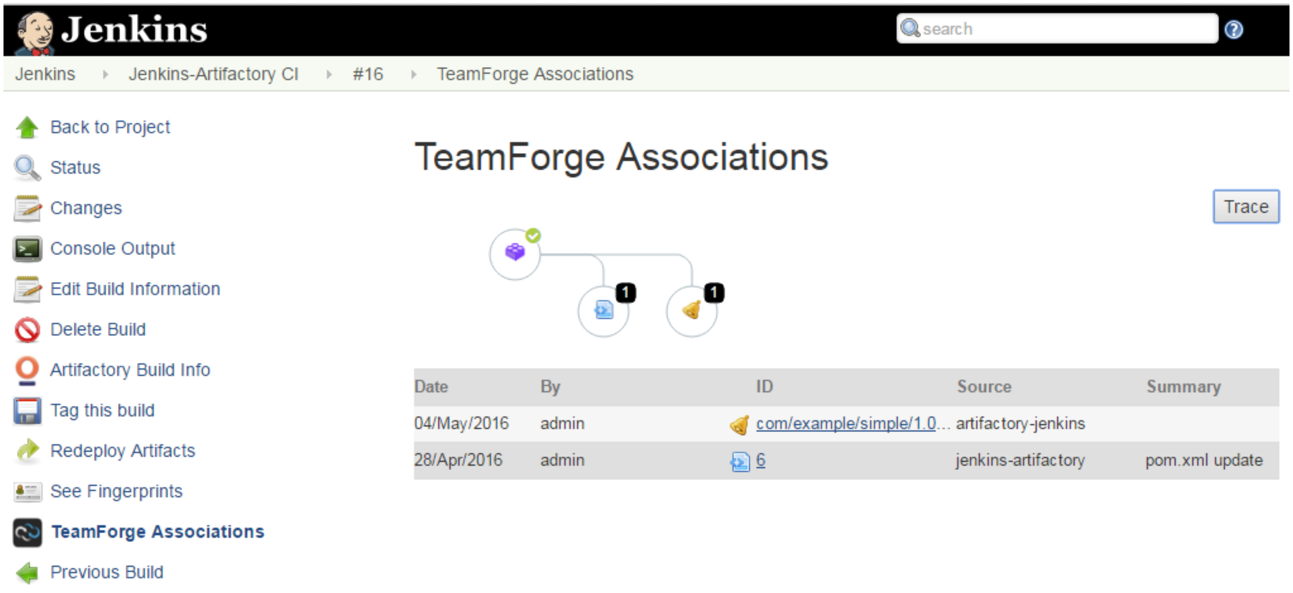
6. Click **Remove** to delete the mapping.

Jenkins Integration

The TeamForge EventQ Jenkins adapter supplies build data from your Jenkins continuous integration server to TeamForge EventQ for purposes of metadata archival, traceability and event reporting.

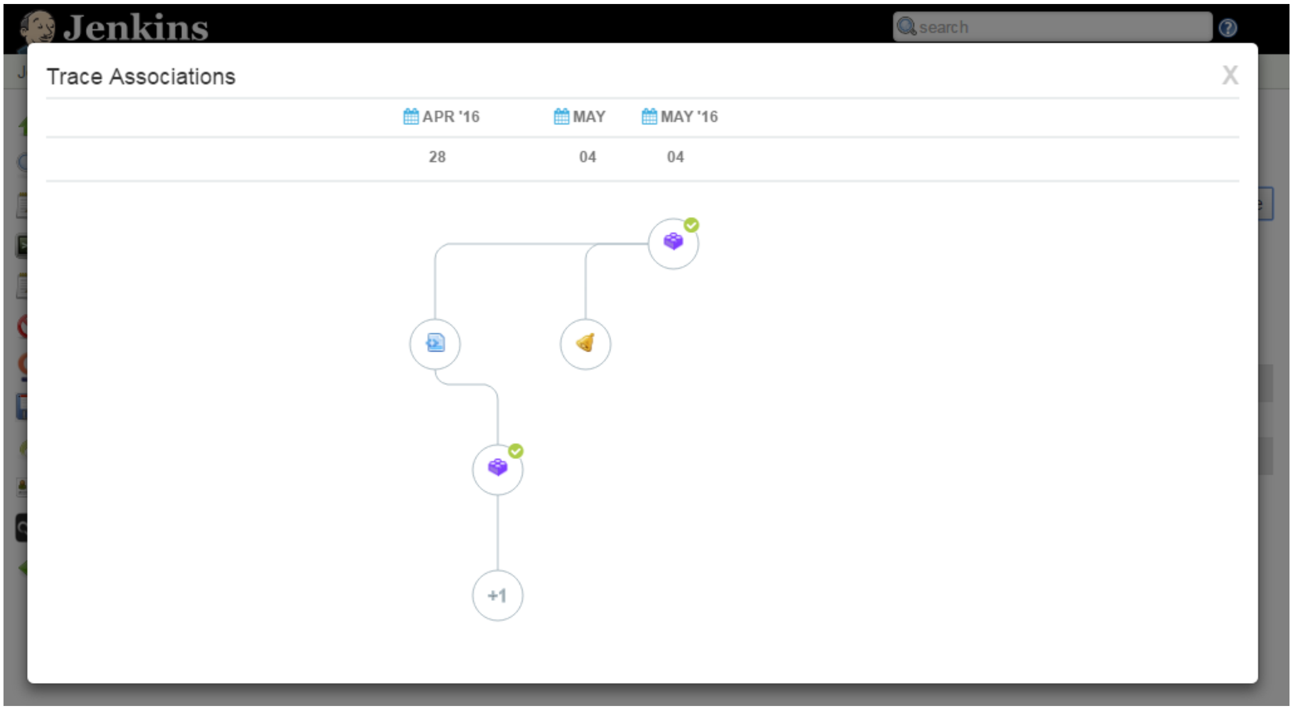
The TeamForge EventQ Jenkins integration brings associations and traceability to Jenkins. The adapter displays associations between Jenkins builds and other tools within the TeamForge landscape like version control commits and binary artifacts. Traceability is therefore possible between Jenkins build, TeamForge, and various orchestrated third-party tools.

Inside TeamForge, Jenkins build details are visible in activity streams and association visualizations. Within Jenkins, associations are visible from the **TeamForge Associations** tab.



Date	By	ID	Source	Summary
04/May/2016	admin	com/example/simple/1.0...	artifactory-jenkins	
28/Apr/2016	admin	6	jenkins-artifactory	pom.xml update

Clicking **Trace** provides a view of the build in the context of other associated activities. Use cases include root cause analysis and auditable traceability.



The TeamForge EventQ Jenkins adapter is packaged as a Jenkins plug-in. Once installed and configured, the plug-in supplies build results metadata for configured Jenkins jobs to TeamForge EventQ.

Jenkins Version Information

The following table lists the compatible Jenkins and TeamForge-Jenkins Plugin versions.

Product	Version(s) Supported	Integrations Supported
Jenkins (Job and Pipeline)	1.645-2.135	collabnet.hpi 2.0.4

Install a Jenkins Adapter

A new CollabNet Plugin for Jenkins (v2.0) is available that functionally subsumes both CollabNet Plugin v1.2.0 and EventQ Jenkins Adapter v2.0. Install this plugin using the web interface. Existing TeamForge-Jenkins integrations that use either CollabNet Plugin v1.2.0 (or earlier), or EventQ Jenkins Adapter v2.0 (or earlier), or both, must be migrated (uisng the `migrate_jenkins_plugin.sh` script) after installing the new CollabNet plugin.

CollabNet Plugin Features

- Notify TeamForge EventQ when builds complete. The CollabNet Plugin must be installed once on each Jenkins server you wish to connect to TeamForge EventQ.
- Authenticate users from TeamForge. If setup as the “Build & Test” appplication, it can even use Single Sign-On.
- Authorization from TeamForge, including the ability to set permissions in Jenkins based on roles in your TeamForge project.
- Upload the build log or workspace artifacts to the TeamForge Documents.
- Upload workspace artifacts to the TeamForge File Release System, as a post-build publishing task or as a build promotion task.
- Open/update/close TeamForge Tracker artifacts based on the Jenkins build status.
- Upload workspace artifacts to the Lab Management Project Build Library. (Requires CollabNet Lab Management).

  Click here to know more about the requirements for installing the latest CollabNet plugin.

1. If you are integrating TeamForge and Jenkins for the first time:
   1. Log on to the Jenkins Server as a previliged Jenkins user, navigate to **Manage Jenkins > Manage Plugins > Available**.
   2. Select the latest CollabNet Pluginand install the plugin.
   3. Restart your Jenkins server.
2. Existing TeamForge-Jenkins integrations that use CollabNet Plugin v1.2.0 (or earlier):
   1. Log on to the Jenkins Server as a privileged Jenkins user, navigate to **Manage Jenkins > Manage Plugins > Updates**.

   2. Select the latest CollabNet Plugin and install the plugin.
   3. Download the `migrate_jenkins_plugin.sh` script and save it to `<JENKINS_HOME_DIRECTORY>/jobs/`.

      > **TIP:** Jenkins default home directory is */var/lib/jenkins/*.

   4. Change ownership of the `migrate_jenkins_plugin.sh` file.
      `chmod 755 migrate_jenkins_plugin.sh`
   5. Run the `migrate_jenkins_plugin.sh` script.
      `./migrate_jenkins_plugin.sh`
3. Existing TeamForge-Jenkins integrations that use either EventQ Jenkins Adapter v2.0 (or earlier) or both CollabNet Plugin v1.2.0 (or earlier) and EventQ Jenkins Adapter v2.0 plugins:
   1. Log on to the Jenkins Server as a privileged Jenkins users, navigate to **Manage Jenkins > Manage Plugins > Installed**.
   2. Select the EventQ Jenkins Adapter v2.0 and click **Uninstall**.
   3. If you have CollabNet Plugin v1.2.0 (or earlier) installed already, select the *Updates* tab. Select the *Available* tab otherwise.
   4. Select the latest CollabNet Plugin and install the plugin.
   5. Download the `migrate_jenkins_plugin.sh` script and save it to `<JENKINS_HOME_DIRECTORY>/jobs/`.

      > **TIP:** Jenkins default home directory is */var/lib/jenkins/*.

   6. Change ownership of the `migrate_jenkins_plugin.sh` file.
      `chmod 755 migrate_jenkins_plugin.sh`
   7. Run the `migrate_jenkins_plugin.sh` script.
      `./migrate_jenkins_plugin.sh`

Configure a Jenkins Adapter

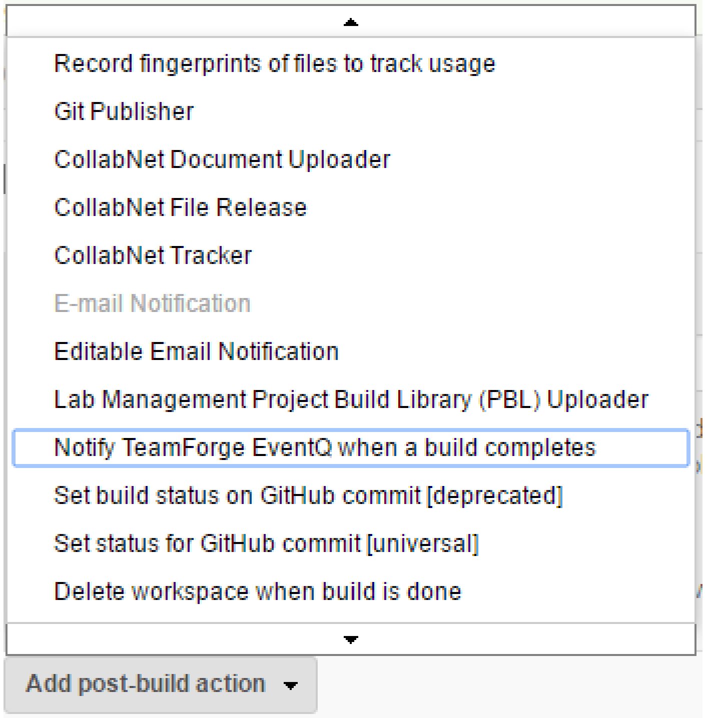
Configure the Jenkins adapter to notify TeamForge EventQ when builds complete.

Before you start the configuration, make sure that you have installed the Jenkins plug-in using the web interface. The TeamForge EventQ Jenkins adapter has a system configuration as well as job level configurations. The system configuration sets defaults to be used by subsequently configured job level configurations.

> **IMPORTANT:** The source association key should be unique for each job configured.

Configuring an Individual Jenkins Job

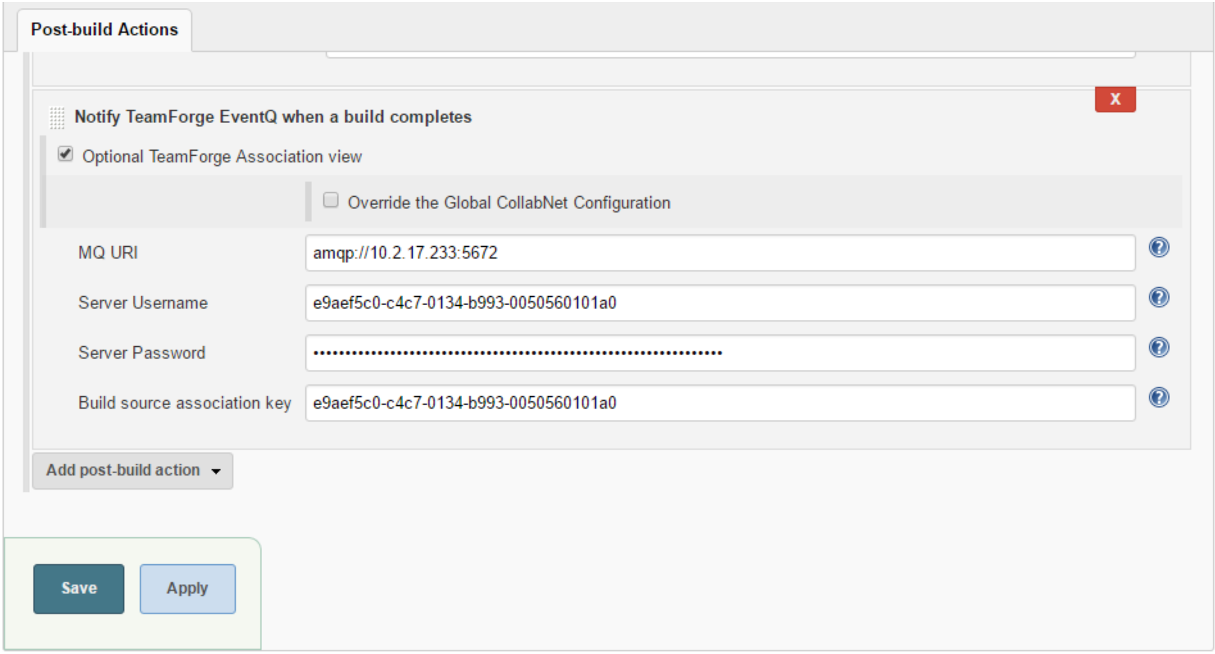
1. Create a build source and have the configuration details handy for the following steps.
2. As a privileged Jenkins user, locate the job you wish to report build data to TeamForge EventQ and navigate to its configuration page.
3. Add a post-build action to **Notify TeamForge EventQ when a build completes**.


Record fingerprints of files to track usage
Git Publisher
CollabNet Document Uploader
CollabNet File Release
CollabNet Tracker
E-mail Notification
Editable Email Notification
Lab Management Project Build Library (PBL) Uploader
Notify TeamForge EventQ when a build completes
Set build status on GitHub commit [deprecated]
Set status for GitHub commit [universal]
Delete workspace when build is done
Add post-build action

4. Using the values you obtained in Step 1, copy and paste the Queue Server value into the **Server URL** field.

> **NOTE:** Include the amqp protocol and the port information.

5. Fill in the **Server Username** and **Server Password** fields. Copy and paste the queue username and password values from the values obtained in Step 1.
6. Populate the Build source association key field with the value obtained in Step 1. For more information about association keys, see Source Association Keys.



Post-build Actions

Notify TeamForge EventQ when a build completes

☑ Optional TeamForge Association view

□ Override the Global CollabNet Configuration

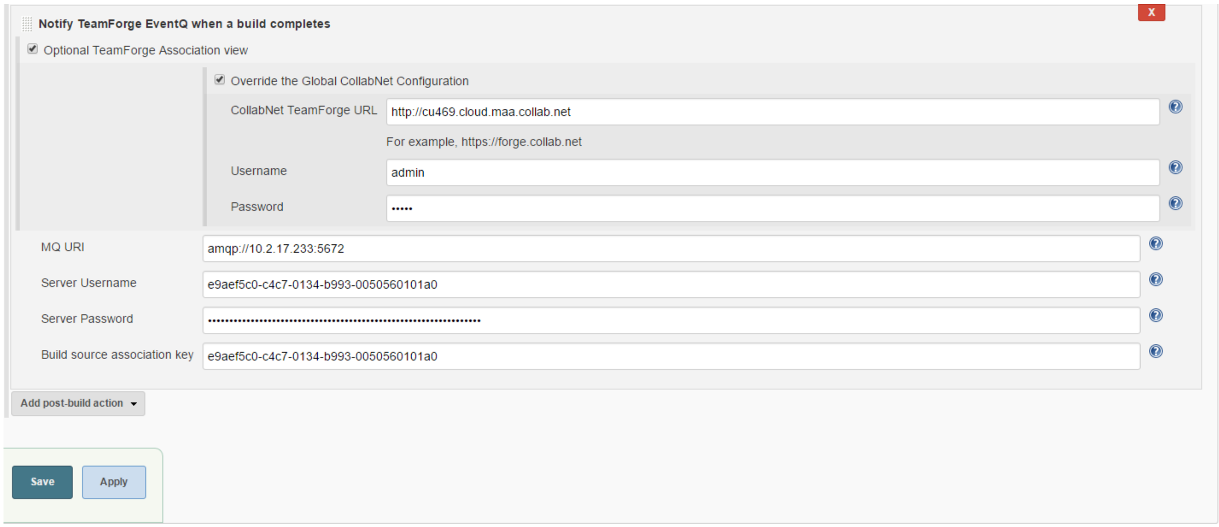
MQ URI: amqp://10.2.17.233:5672

Server Username: e9aef5c0-c4c7-0134-b993-0050560101a0

Server Password: ••

Build source association key: e9aef5c0-c4c7-0134-b993-0050560101a0

7. By default, the **Optional TeamForge Association View** check box is selected. If required, you can override the global configuration. Select the **Override the Global CollabNet Configuration** check box and type the TeamForge URL and user credentials.



Notify TeamForge EventQ when a build completes

☑ Optional TeamForge Association view

☑ Override the Global CollabNet Configuration

CollabNet TeamForge URL: http://cu469.cloud.maa.collab.net

For example, https://forge.collab.net

Username: admin

Password: •••••

MQ URI: amqp://10.2.17.233:5672

Server Username: e9aef5c0-c4c7-0134-b993-0050560101a0

Server Password: ••

Build source association key: e9aef5c0-c4c7-0134-b993-0050560101a0

8. Save the job configuration.
9. Run a build to test the new configuration and verify configuration. Information and errors will be reported to your Jenkins log and to the build console.

Configuring the Default Jenkins Global Settings

Now that you've set up your first job level configuration, you can save some effort next time around by defining system values for the Server URL, Server Username and Server Password. Note that each job needs a unique Source Association Key, so that will still need to be configured per job.

1. Refer to Step 1 and obtain values for the Server URL, Server Username and Server Password.
2. As a privileged Jenkins user, navigate to **Manage Jenkins > Configure System** page on the target Jenkins server.
3. On the Jenkins configuration page, find the TeamForge EventQ section.
    1. Fill in the **Server URL** field. Copy and paste the Queue Server value into the Server URL field in Jenkins.

        > **IMPORTANT:** Include the `amqp` protocol and the port information.

    2. Fill in the **Server Username** and **Server Password** fields.
4. Save the configuration.

External Git Repository Integration

With a customized post-receive extension (hook) installed, Git servers external to TeamForge can notify TeamForge EventQ of Git "push" activity.

TeamForge project Git repositories are already wired to supply push activity data to TeamForge EventQ; no special configuration needed. This section instead details integration of Git repositories external to TeamForge.

The TeamForge EventQ Git adapter is packaged as a Git post-receive hook script. Once installed and configured, the hook script supplies push metadata for the selected repository to TeamForge EventQ. The push data is associated to the proper source using source association keys (see Source Association Keys). Configuration therefore requires that the message queue server URL and credentials be supplied, along with the appropriate source association key.

> **IMPORTANT:** The TeamForge EventQ-Git adapter v1.0 supports Git 1.8.3.1 or later.

Install a Git Adapter

Install a Git adapter as a post-receive hook on Mac OS X/Linux.

Minimum Requirements:

- Git 1.8.3.1 or later.
- Python 2.6. Python v3.0 and later are not compatible.

Each external Git repository you wish to monitor with TeamForge EventQ requires the installation of a customized post-receive hook script as follows.

Install Git Post-receive Hooks

Contact the CollabNet Support to download the `eventQ-git-adapter-[version].zip` adapter file and unpack it inside your repository's hooks directory.

For example, if your repository's hooks directory is located at `/path/to/repo.git/hooks`, `cd /path/to/repo.git/hooks`, download the zip file to the current location, then unzip `eventQ-git-adapter-1.0.zip`.

This will unpack the following into your hooks directory:

- **Files**: `LICENSE.txt`, `README.txt`, `post_receive_orc_publish_amqp.conf.sample`, `post_receive_orc_publish_amqp.py`
- **Directories**: `eventq/` and `pika/`

Configuration

1. In a web browser, log into TeamForge EventQ as a privileged EventQ user and edit (or add) the relevant Commit source you wish to associate with this Git repository. **Manage Sources > Toggle Commit > Add or Edit a Source > External repository**
2. On the **Edit Source** page, locate the section titled "Adapter Configuration Information" and copy all values; these values are used later in the configuration process.
3. Switch back to your command line console and copy `post_receive_orc_publish_amqp.conf.sample` to `post_receive_orc_publish_amqp.conf`.
4. Edit `post_receive_orc_publish_amqp.conf` and modify options for your specific installation, for example:
   - Logging Section
     - **filename** — Full path to the desired log file, for example, `/tmp/post-commit.log`.
     - **level** — The logging level, which is one of these: **debug** , **info**, **warning**, **error**, or **critical**.
   - RabbitMQ (amqp) Section
     - **host** — The hostname of the amqp server. Copy and paste the hostname from the **Queue Server** value from the **Edit Source** page in TeamForge EventQ.
     - **port** — The port number of the amqp server (default is 5672). Copy and paste the port number the **Queue Server** value from the **Edit Source** page in TeamForge EventQ.

  - **username** — Username used to access the amqp server. Copy and paste the queue username value from the **Edit Source** page in TeamForge EventQ.
  - **password** — Password used to access the amqp server. Copy and paste the queue password value from the **Edit Source** page in TeamForge EventQ.
- TeamForge EventQ Section
  - **source_association_key** — key provided by TeamForge EventQ. Copy and paste the source association key value from the **Edit Source** page in TeamForge EventQ. For more on association kys, see Source Association Keys.

Activate Post-receive Hook

Copy or create a symlink called “post-receive” linking to `post_receive_orc_publish_amqp.py`.

External Subversion Repository Integration

With a customized post-commit extension (hook) installed, Subversion servers external to TeamForge can notify TeamForge EventQ of Subversion commit activity.

TeamForge project Subversion repositories are already wired to supply commit activity data to TeamForge EventQ; no special configuration needed. This section instead details integration of Subversion repositories external to TeamForge.

The TeamForge EventQ Subversion adapter is packaged as a Subversion post-commit hook script. Once installed and configured, the hook script supplies commit metadata for the selected repository to TeamForge EventQ. The commit data is associated to the proper source using source association keys (see Source Association Keys). Configuration therefore requires that the message queue server URL and credentials be supplied, along with the appropriate source association key.

> **IMPORTANT:** The TeamForge EventQ-Subversion adapter v1.0 supports Subversion 1.7 or later.

Install a Subversion Adapter on Mac OS X/Linux

Install a Subversion adapter as a post-commit hook on Mac OS X/Linux.

Minimum Requirements:

- Subversion 1.7 or later.
- Python 2.6. Python v3.0 and later are not compatible.

Each external Subversion repository you wish to monitor with TeamForge EventQ requires the installation of a customized post-commit hook script as follows.

Install Subversion Post-commit Hooks

Contact the CollabNet Support to download the `eventQ-subversion-adapter-[version].zip` adapter file and unpack it inside your repository's hooks directory.

For example, if your repository's hooks directory is located at `/path/to/svn/repos/my_repo/hooks`, `cd /path/to/svn/repos/my_repo/hooks`, download the zip file to the current location, then unzip `eventQ-svn-adapter-1.0.zip`.

This will unpack the following into your hooks directory:

- LICENSE.txt
- README.txt
- configdefaults.py
- configwrapper.py
- eventq_commit_gate_hook.py
- orchestrator_commit_gate_hook_snippet.sh
- pika/
- post-commit.bat.sample
- post_commit_orc_publish_amqp.conf.sample
- post_commit_orc_publish_amqp.py

Configuration

1. In a web browser, log into TeamForge EventQ as a privileged EventQ user and edit (or add) the relevant Commit source you wish to associate with this Subversion repository. **Manage Sources > Toggle Commit > Add or Edit a Source > External repository**
2. On the **Edit Source** page, locate the section titled “Adapter Configuration Information” and copy all values; these values are used later in the configuration process.
3. Switch back to your command line console and copy `post_commit_orc_publish_amqp.conf.sample` to `post_commit_orc_publish_amqp.conf`.
4. Edit `post_commit_orc_publish_amqp.conf` and modify options for your specific installation, for example:
   - Logging Section
     - **filename** — Full path to the desired log file, for example, `/tmp/post-commit.log`.
     - **level** — The logging level, which is one of these: **debug** , **info**, **warning**, **error**, or **critical**.
   - svnlook Section
     - svnlook — The path to your svnlook executable, for example, `/usr/bin/svnlook` or `/opt/subversion/bin/svnlook`.

- RabbitMQ (amqp) Section
  - **host** — The hostname of the amqp server. Copy and paste the hostname from the **Queue Server** value from the **Edit Source** page in TeamForge EventQ.
  - **port** — The port number of the amqp server (default is 5672). Copy and paste the port number the **Queue Server** value from the **Edit Source** page in TeamForge EventQ.
  - **username** — Username used to access the amqp server. Copy and paste the queue username value from the **Edit Source** page in TeamForge EventQ.
  - **password** — Password used to access the amqp server. Copy and paste the queue password value from the **Edit Source** page in TeamForge EventQ.
- TeamForge EventQ Section
  - **source_association_key** — key provided by TeamForge EventQ. Copy and paste the source association key value from the **Edit Source** page in TeamForge EventQ. For more on association kys, see Source Association Keys.

Activate Post-commit Hook

Copy or create a symlink called “post-commit” linking to `post_commit_orc_publish_amqp.py`.

Install a Subversion Adapter on Windows

Install a Subversion adapter as a post-commit hook on Windows.

Minimum Requirements:

- Subversion 1.7 or later.
- Python 2.6. Python v3.0 and later are not compatible.

> **NOTE:** This procedure assumes that you have installed CollabNet Subversion Edge, version 4.0 or later, although other Windows distributions can be similarly configured. Each external Subversion repository you wish to monitor with TeamForge EventQs requires the installation of a customized post-commit hook script as follows.

Install Subversion Post-commit Hooks

Contact the CollabNet Support to download the `eventQ-subversion-adapter-[version].zip` adapter file and unpack it inside your repository’s hooks directory.

For example, if your repository’s hooks directory is located at `C:\csvn\repositories\my_repo\hooks`, cd `C:\csvn\repositories\my_repo\hooks`, download the zip file to the current location, then unzip `eventQ-svn-adapter-1.0.zip`.

This will unpack the following into your hooks directory:

- LICENSE.txt
- README.txt
- configdefaults.py
- configwrapper.py
- eventq_commit_gate_hook.py
- orchestrator_commit_gate_hook_snippet.sh
- pika/
- post-commit.bat.sample
- post_commit_orc_publish_amqp.conf.sample
- post_commit_orc_publish_amqp.py

Configuration

1. In a web browser, log into TeamForge EventQ as a privileged EventQ user and edit (or add) the relevant Commit source you wish to associate with this Subversion repository. **Manage Sources > Toggle Commit > Add or Edit a Source > External repository**
2. On the **Edit Source** page, locate the section titled "Adapter Configuration Information" and copy all values; these values are used later in the configuration process.
3. Switch back to your command line console and copy `post_commit_orc_publish_amqp.conf.sample` to `post_commit_orc_publish_amqp.conf`.
4. Edit `post_commit_orc_publish_amqp.conf` and modify options for your specific installation, for example:
   - Logging Section
     - **filename** — Full path to the desired log file, for example, `C:\Temp\post-commit.log`.

       > **IMPORTANT:** Ensure that the log file exists, or create it, before proceeding to the next step.

     - **level** — The logging level, which is one of these: **debug** , **info**, **warning**, **error**, or **critical**.
   - svnlook Section
     - svnlook — The path to your svnlook executable, for example, `C:\csvn\bin\svnlook.exe`.
   - RabbitMQ (amqp) Section
     - **host** — The hostname of the amqp server. Copy and paste the hostname from the **Queue Server** value from the **Edit Source** page in TeamForge EventQ.

    - **port** — The port number of the amqp server (default is 5672). Copy and paste the port number the **Queue Server** value from the **Edit Source** page in TeamForge EventQ.
    - **username** — Username used to access the amqp server. Copy and paste the queue username value from the **Edit Source** page in TeamForge EventQ.
    - **password** — Password used to access the amqp server. Copy and paste the queue password value from the **Edit Source** page in TeamForge EventQ.

    > **NOTE:** A RabbitMQ server is optionally set up during the installation process. The installation process sets a default password for the `eventq` user and stores it for reference in the following file on the RabbitMQ host (root access required): `/opt/collabnet/rabbitmq/.rabbit_passwords`

- TeamForge EventQ Section
    - **source_association_key** — key provided by TeamForge EventQ. Copy and paste the source association key value from the **Edit Source** page in TeamForge EventQ. For more on association kys, see Source Association Keys.

Activate Post-commit Hook

1. Copy the `post-commit.bat.sample` to `post-commit.bat`.
2. Edit the `post-commit.bat` for your specific installation: set `PYTHON_EXE` to the full path to `python.exe` executable. For CollabNet Subversion Edge, this is typically `C:\csvn\Python25\python.exe`.

Related Links

- Installing Pika
- Implementing Repository Hooks
- Subversion Adapter FAQs

Crucible Integration

The TeamForge EventQ Crucible adapter enables communication between your Crucible server and TeamForge EventQ.

The TeamForge EventQ Crucible adapter is packaged as a Crucible add-on. Once installed and configured, the add-on supplies code review metadata for selected Crucible projects to TeamForge EventQ. The code review metadata is associated to the proper source using source association keys (see Source Association

Keys). Configuration therefore requires that the MQ server URL and credentials be supplied, along with the appropriate source association key.

The TeamForge EventQ-Crucible adapter v1.0 supports Crucible 3.0.1

Install a Crucible Adapter

Install the Crucible adapter using the Crucible Add-on Manager.

TeamForge EventQ currently supports Crucible 3.0.1. Use the Crucible adapter to notify TeamForge EventQ when reviews are created or modified, or when files are added or removed from reviews. The Crucible adapter must be installed once on each Crucible server you wish to connect to TeamForge EventQ. Install the add-on by using the Crucible add-on Manager.

1. Contact the CollabNet Support to download the `eventQ-Crucible-adapter-[version].jar` Crucible plug-in file.
2. As a privileged Crucible user, navigate to **Administration > Manage Add-ons > Upload Add-on**.
3. In the **From my computer** field, select your `eventQ-Crucible-adapter-[version].jar` file, then click **Upload**. You are now returned to the **Manage Add-ons** view and the **TeamForge EventQ Adapter** now be visible under **User-installed Add-ons**.

> **NOTE:** Going forward, uploading new versions of the adapter automatically upgrades the old adapter while retaining your old settings.

Configure a Crucible Add-on

Configure the "TeamForge EventQ Adapter" add-on to notify TeamForge EventQ about review changes.

Before you start configuration, make sure that you have installed the Crucible add-on: verify that **TeamForge EventQ Adapter** is present in the **User-installed Add-ons** list.

1. As a privileged EventQ user, edit (or add) the relevant Review Source you wish to associate with this Crucible project. **Manage Sources > Toggle Review > Edit the Source**
2. On the **Edit Source** page, find the **Adapter Configuration Information** section and copy all values; these values are used later in the configuration process.
3. As a privileged Crucible user, navigate to **Administration > Manage Add-ons**.
4. Click **TeamForge EventQ Adapter** from the **User-installed Add-ons** list. The **TeamForge EventQ Adapter** section displays new options, including "Configure".
5. Click **Configure**. You see a configuration form for each of your projects.\
6. Find the project to associate with TeamForge EventQ and enter the following information.

    1. Populate the **Server URL** field. Copy and paste the **Queue Server** value from the **Edit Source** page in TeamForge EventQ. Make sure you include the “amqp” protocol and the port information.
    2. Populate the **Server Username** and **Server Password** fields. Copy and paste the queue username and password value from the **Edit Source** page in TeamForge EventQ.
    3. Populate the **Source Association Key** field. Copy and paste the source association key value from the **Edit Source** page in TeamForge EventQ. For more on association keys, see Source Association Keys.

7. Click **Save**. Crucible now reports code review events for the configured project to TeamForge EventQ.

Related Links

Crucible Adapter FAQs

Review Board Integration

The TeamForge EventQ-Review Board adapter enables communication between your Review Board server and TeamForge EventQ.

The Review Board adapter is packaged as a Review Board extension, and once installed provides web-based configuration inside Review Board.

The adapter supplies code review metadata for selected repositories to TeamForge EventQ. Code review metadata is associated to the proper source using source association keys (see Source Association Keys). Configuration therefore requires that the MQ server URL and credentials be supplied, along with the appropriate source association key.

> **IMPORTANT:** TeamForge EventQ-Review Board adapter supports Review Board v1.7 and later.

Install or Upgrade a Review Board Adapter on Mac OS X/ Linux

Install a Review Board adapter as a Review Board extension to retrieve code review data from instances of Review Board 1.7 and later.

Minimum Requirements

- Review Board 1.7 or higher
- Python 2.6 or 2.7. Version 3.0 and later are not compatible.
- setuptools 2.1 or later
- pip 1.5 - 6.0.6 (6.0.7 has issues and is not supported)

1. Upgrade pip. Upgrade pip if you do not have the minimum required version.
   ```
   sudo pip install --upgrade pip
   ```
2. Upgrade setuptools. Upgrade setuptools if you do not have the minimum required version.
   ```
   sudo pip install --upgrade setuptools
   ```
3. Install or upgrade the TeamForge EventQ-Review Board adapter as a Review Board extension.
   - **Installation**
     - If your server has internet access, use the command below to install the EventQ adapter as an extension.
       ```
       sudo pip install --use-wheel --no-index --find-links=http://eventq.collab.net/RHEL/6/x86_64/wheelhouse eventq_rb
       ```
     - If your server cannot access the internet, see Install a Review Board Adapter Without Internet Access instead.
   - **Upgrade**
     - Add the -U flag to upgrade to the latest version of the Review Board adapter.
       ```
       sudo pip install -U --use-wheel --no-index --find-links=http://eventq.collab.net/RHEL/6/x86_64/wheelhouse eventq_rb
       ```
4. Restart Review Board's web server to register the newly installed EventQ adapter.

Install or Upgrade a Review Board Adapter Without Internet Access

The following instructions pertain to situations where your Review Board server does not have outbound access to the internet.

1. Contact the CollabNet Support to download the `eventq_rb-[version].zip` archive file.
2. Extract the archive file on the target server.
   ```
   unzip eventq_rb-[version].zip
   ```
3. Install or upgrade the TeamForge EventQ-Review Board adapter as a Review Board extension.
   - Use the command below to install the EventQ adapter as an extension.
     ```
     sudo pip install --use-wheel --no-index --find-links=PATH_TO_WHEELHOUSE eventq_rb
     ```

     Where `PATH_TO_WHEELHOUSE` is the path to the directory where you unzipped the adapter file. Suppose you unzipped the adapter file to the `/tmp` directory. In such a case, the `PATH_TO_WHEELHOUSE` is `/tmp`.

     Example:

     ```
     sudo pip install --use-wheel --no-index --find-links=/home/myusername/wheelhouse/ eventq_rb
     ```

- Add the -U flag to upgrade to the latest version of the Review Board adapter.

  ```
  sudo pip install -U --use-wheel --no-index --find-links=PATH_TO_WHEELHO
  USE eventq_rb
  ```

4. Restart Review Board's web server to register the newly installed EventQ adapter.

Configure the Review Board Adapter (TeamForge Integrated)

Configure a TeamForge-integrated Review Board adapter to retrieve code review metadata.

Once the adapter has been installed successfully, configure Review Board by following these steps. The instructions below pertain to Review Board instances integrated with CollabNet TeamForge. See Configure the Review Board Adapter (Standalone) for configuration instructions when using a standalone version of Review Board.

Navigate to Review Board Extension Administration

1. Log into TeamForge as a site-admin user.
2. Select **Admin > Integrated Apps**.
3. Select the Review Board option and click **Administer**. You should now see Review Board administrative page.
4. Click **Extensions** from the menu bar to list your currently installed extensions. `eventq-rb` should be listed.
5. Click **Enable** to activate the EventQ extension.
6. Click **Configure** on the `eventq-rb` extension.

   The eventq-rb configuration page shows up. From here you can select and configure the repository whose reviews you would like to push to an EventQ review source. To create a review source in EventQ see: Manage Review Sources.

Configure a Review Board Respository

1. Select the repository from the **Repository** drop-down list.
2. Copy and paste the fields from the EventQ review source that you wish to associate to this repository.
3. Click **Save**.
4. Repeat steps 1 through 3 for each Review Board repository you wish to monitor in TeamForge EventQ.

Configure the Review Board Adapter (Standalone)

Configure a standalone instance of Review Board adapter to retrieve code review metadata.

Once the adapter has been installed successfully, configure Review Board by following these steps. The instructions below pertain to standalone Review Board instances.

Navigate to Review Board Extension Administration

1. Log into your Review Board instance as a privileged user.
2. Click the **Admin** link to reach the administrative page.
3. Click **Extensions** on the menu bar to list your currently installed extensions. `eventq-rb` should be listed.
4. Click **Enable** to activate the EventQ extension.
5. Click **Configure** on the `eventq-rb` extension.

   The `eventq-rb` configuration page shows up. From here you can select and configure the repository whose reviews you would like to push to an EventQ review source. To create a review source in EventQ see: Manage Review Sources.

Configure a Review Board Respository

1. Select the repository from the Repository drop-down list.
2. Copy and paste the fields from the EventQ review source that you wish to associate to this repository.
3. Click **Save**.
4. Repeat steps 1 through 3 for each Review Board repository you wish to monitor in TeamForge EventQ.

EventQ provides a mechanism, allowing you to add relevant project tools to the TeamForge project context. This is accomplished through—sources—which represent light-weith integrations to external products. Add sources that populate activity streams with relevant activity data such as work item updates, commits, CI builds, and code reviews.

What is a Source?

TeamForge is an open development platform with a strong notion associations. Within TeamForge, any two objects can be associated together, and when multiple associations are chained together, they create "traceability". But what about tools outside of TeamForge? EventQ provides just such a mechanism, allowing you to add relevant project tools to the TeamForge project context. This is accomplished through—sources—which represent light-weith integrations to external products. Once established, "sources" capture activity data from external tools, bringing them into the TeamForge project context and associations fabric.

- For work items, sources are issue trackers.
- For SCM commits, sources are defined as the code repository of interest.
- For CI builds, sources are “jobs” or “build configurations”.
- For code reviews, sources are defined as the project or repository of interest.

How do Sources work?

Whenever an “activity” occurs in a tool that has been setup as a project source, a special “adapter” sends notification of the activity to TeamForge EventQ. For example, if a specific Jenkins job is configured as a source, build information will be sent to TeamForge EventQ when that Jenkins job is executed. The resulting activity is mapped to the TeamForge project in question. A TeamForge project can have multiple sources, all sending their activity data to the TeamForge project.

Associations

TeamForge EventQ creates associations between activities such as reviews, commits, builds, and work items.

Associations are relationships between activities like work items, commits, builds, and reviews. You can see associations for a particular activity on its TeamForge EventQ detail page or create a traceability graph of associations using the Trace associations button from an activity details page. Most associations in TeamForge EventQ center around commits. That is, builds, work items, and reviews associate directly to commits, while a build-to-review association are implied through a common commit.

Builds are associated to commits by the underlying CI/build system. Typically, CI/build systems obtain source code from a target repository and build based on that specific revision of source code. TeamForge EventQ therefore relies on the CI/build system to report the associated revisions, which then get mapped as associations between commits and builds in TeamForge EventQ.

Reviews associate to commits in one of two ways, depending on the process being employed and the source code review product.

When creating code reviews, note that the code review’s repository URL must match the one specified while configuring the related SCM source. Otherwise, TeamForge EventQ will not be able to register associations between commits and reviews.

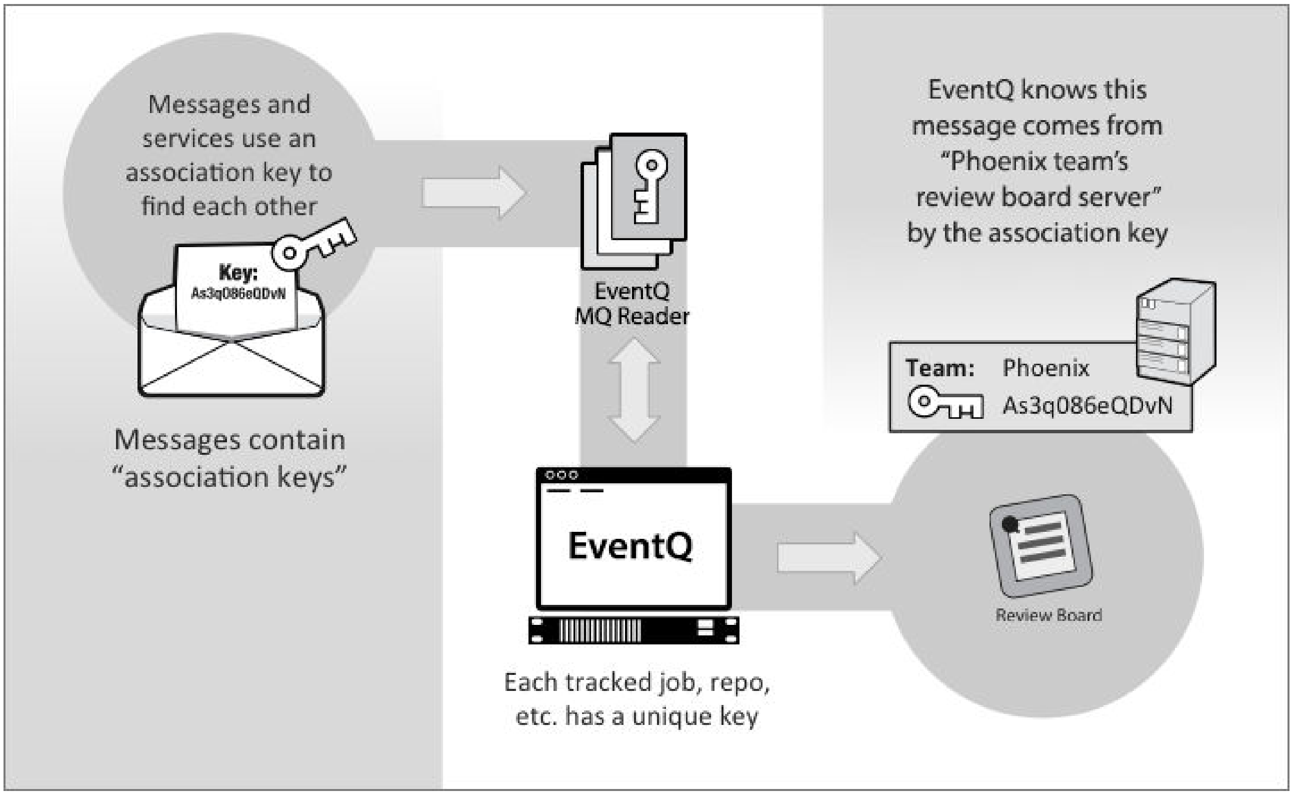
Work items can be directly associated to commits by referencing the work item artifact ID in the commit message.

Source Association Keys

TeamForge EventQ uses source association keys to uniquely identify incoming data from activity sources that it monitors.

Adapters supply data to TeamForge EventQ through the message queue (MQ) layer. When a relevant activity occurs in source systems, TeamForge EventQ adapters generate a message with key metadata about the activity and place that message on the designated MQ server. When TeamForge EventQ receives the message, it must associate it to the proper source somehow. To do this, TeamForge EventQ supplies a unique “source association key” for each external source created. Adapters must supply this unique source association key with each message for the data to correlate with the appropriate project source.

Note that commit sources based on TeamForge project repositories do not need source associations keys since they are internal and are identified by other means. All other sources require source association keys.


Messages and services use an association key to find each other
Key: As3q086eQDvN
Messages contain “association keys”
EventQ MQ Reader
EventQ
Each tracked job, repo, etc. has a unique key
EventQ knows this message comes from “Phoenix team’s review board server” by the association key
Team: Phoenix
As3q086eQDvN
Review Board

Locate a Source Association Key

Source association key for existing activity sources are located on the corresponding source configuration page.

1. From the activity stream, click **Manage Sources**.
2. Select the desired source type toggle. TeamForge EventQ displays all existing sources.
3. Click **Edit** on the source whose key you wish to obtain. You see the configuration page for the selected source.
4. Locate the source association key for the selected source. On most browsers you can copy the key by clicking the small clipboard icon.

Extensible Data Source (XDS) Overview

EventQ can be configured to accept data from a wide range of sources through the Extensible Data Source (or XDS) feature. In addition to the stock activity APIs defined for commits, builds, reviews, and work items, the XDS feature enables you to create your own activity APIs by defining an “XDS Schema” that describes the expected activity message format for a particular tool or product domain in question. See Extend TeamForge EventQ for more information.

“XDS Sources” can be added to your project through the “Custom” source type. Once added, XDS activities will be visible in the activity stream. XDS sources can also have associations just like any other source resulting in traceability between XDS and other activities, even between multiple XDS sources. For more information, see Manage XDS Sources.

Manage Build Sources

Add build sources to bring automated build job results into the TeamForge project and traceability context.

Build sources bring build/CI data into EventQ for archival of metadata, participation in traceability, and activity reporting.

Create or Find an Existing Tool to Group Your Build Source Under

1. From the desired project context, navigate to **Project Admin > Tools**.
2. Determine whether a **Tool** exists that represents a logical container for your desired source. For instance, if you are adding a Jenkins source, look for a tool representing the instance of Jenkins in question.
    1. If a tool exists already, click the tool’s title to edit the tool.
    2. If a tool does not exist, click **Add Tool**.

Edit an Existing Build Source

1. To edit an existing build source, find the **Data Source** in the list of sources associated with the tool. You can edit the display name or associate a different commit source for this build source. If you do not select a commit source, no associations will be drawn for the build activities coming from this build source. TeamForge displays a warning if a build source is not associated with a commit source. See Manage Commit Sources.
2. Click **Update** to save your changes.

> **NOTE:** While editing an existing source configuration, you can toggle a source from `Active` to `Inactive`, which will stop TeamForge from collecting data from that source; toggle back to `Active` to resume data collection from the source.

Create a New Build Source

1. If this is a new tool, first select the **Tool Type** from the drop-down list. For instance, if you're adding a Jenkins tool instance, select **Jenkins**.
2. Select the **Include Traceability** check box. A blank data source appears. If this is an existing tool with existing sources, click the **Data Source + (Add Source)** icon to add a new blank source.
    1. Provide a display name for the build source. This display name will help you differentiate the source in lists throughout the user interfaces. The display name can be up to 100 alphanumeric characters in length.

    2. Select a commit source. Select the commit source from the list of existing commit sources (or create a new one).

        ✓ You must select a commit source from the list of existing commit sources to let TeamForge create associations between your build and commit activities.

        ✓ If you do not select a commit source, no associations will be drawn for the build activities coming from this build source. TeamForge displays a warning if a build source is not associated with a commit source. See Manage Commit Sources for more information about adding a commit source.

        ✓ For most source types, TeamForge generates a unique "source association key". The source association key uniquely identifies and helps route data from sources properly in TeamForge. You must supply this string while configuring adapters for most source types. You can copy the key by clicking on the small clipboard icon. No association key is necessary when you configure a TeamForge project code repository, though.

3. Click **Update**.

    TeamForge saves the new build source and activates it.

Manage Review Sources

Add review sources to bring code reviews into the TeamForge project and traceability context.

Review sources bring code review data into EventQ for archival of meta data, participation in traceability, and activity reporting.

Create or Find an Existing Tool to Group Your Review Source Under

1. From the desired project context, navigate to **Project Admin > Tools**.
2. Determine whether a tool exists that represents a logical container for your desired source. For instance, if you are adding a Review Board source, look for a tool representing the instance of Review Board in question.
    1. If a tool exists already, click the tool's title to edit the tool.
    2. If a tool does not exist, click **Add Tool**.

Edit an Existing Review Source

1. To edit an existing review source, locate the **Data Source** in question on the **Edit Tool** screen. You can edit the display name or commit association prefix for this source.

2. Click **Update** to save your changes.

> **NOTE:** While editing an existing source configuration, you can toggle a source from `Active` to `Inactive`, which will stop TeamForge from collecting data from that source; toggle back to `Active` to resume data collection from the source.

Create a New Review Source

1. If this is a new tool, first select the **Tool Type** from the drop-down list. For instance, if you're adding a Review Board tool instance, select **Review Board**.
2. Select the **Include Traceability** check box. A blank data source appears. If this is an existing tool with existing sources, click the **Data Source + (Add Source)** icon to add a new blank source.
    1. Provide a display name for the review source. This display name will help you differentiate the source in lists throughout the user interfaces. The display name can be up to 100 alphanumeric characters in length.
    2. Enter the Commit Association Prefix. This prefix is used to associate code commits with code reviews using the commit message. To associate a review with a code commit, you must enter the association prefix and review ID in the commit message in this format:
    `[key:review_ID]`

Where key is any alphanumeric string. A commit association key may be up to 20 characters, including alphanumeric and underscore (“_”) characters.

3. Click **Update**.

   TeamForge saves the new review source, and activates it.

Manage Commit Sources

Commit sources enable your external repositories to participate in traceability.

Commit sources bring commit data into TeamForge for archival of metadata, participation in traceability, and activity reporting.

> **NOTE:** TeamForge Project SCM repositories are added automatically as sources, both on upgrade and when new repositories are defined in the TeamForge project context.

Create or Find an Existing Tool to Group Your Commit Source Under

1. From the desired project context, navigate to **Project Admin > Tools**.
2. Determine whether a tool exists that represents a logical container for your desired source. For instance, if you are adding a source to represent an external Git repository, look for a tool representing the Git server in question.
   1. If a tool exists already, click the tool’s title to edit the tool.
   2. If a tool does not exist, click **Add Tool**.

Edit an Existing Commit Source

1. To edit an existing commit source, locate the **Data Source** in question on the **Edit Tool** screen. You can edit the display name for any commit source. Commit sources can be defined with one of two repository types:
   - **Project repositories**: housed within TeamForge projects.
   - **External repositories**: housed outside of TeamForge projects.

   > **WARNING:** Once defined, you cannot switch the repository type. For external repositories, you can edit the repository URL. For project repositories, `Source Code view` permission is

> required and once selected and saved to a source, the project repository selection may not be altered. To work around this constraint, create a new source with a new repository and deactivate the old source.

2. Click **Update** to save your changes.

> **NOTE:** While editing an existing source configuration, you can toggle a source from `Active` to `Inactive`, which will stop TeamForge from collecting data from that source; toggle back to `Active` to resume data collection from the source.

Create a New Commit Source

1. If this is a new tool, first select the **Tool Type** from the drop-down list. For instance, if you're adding an external Git tool instance, select **External Git**.
2. Select the **Include Traceability** check box. A blank data source appears. If this is an existing tool with existing sources, click the **Data Source + (Add Source)** icon to add a new blank source.
   1. Provide a display name for the commit source. This display name will help you differentiate the source throughout the user interfaces where this source appears. The display name can be up to 100 alphanumeric characters in length.
   2. Select the repository type. Your source may be either a TeamForge project repository or an "external" repository.
      - **Project repositories**: housed within TeamForge projects.
      - **External repositories**: housed outside of TeamForge projects.

      > **NOTE:** Sources for Project repositories are created automatically.You need `Source Code View` permission to see available project repositories.

   3. For External repositories, enter your repository URL. This is likely the URI used to check out code or a WebDAV-enabled URL.
      - **For Subversion repositories**: run the `svn info` command inside the working copy and copy/paste the value of the "URL" field.

      - **For Git repositories**: run the `git remote show origin` inside the working tree and use the value of the "Fetch URL" field, without the "username@".

        For example, if your Subversion repository URL is set to https://forge.example.com/svn/repos/myproject, TeamForge EventQ collects messages from the "myproject" repository and all repositories under “myproject” such as https://forge.example.com/svn/repos/myproject/branches/mybranch.

3. Click **Update**.

   TeamForge saves the new commit source, and activates it.

   > **NOTE:** TeamForge EventQ displays all the defined sources for a step in the order in which they were defined.

Manage Work Item Sources

Add work item sources to tickets, issues, defects and other work items to bring work items into the TeamForge project and traceability context.

Work item sources bring tracker data into TeamForge for archival of metadata, participation in traceability, and activity reporting.

> **NOTE:** TeamForge trackers are added automatically as sources, both on upgrade and when new trackers are defined in the TeamForge project context.

Create or Find an Existing Tool to Group Your Work Item Source Under

1. From the desired project context, navigate to **Project Admin > Tools**.
2. Determine whether a tool exists that represents a logical container for your desired source. For instance, if you are adding a JIRA® source, look for a tool representing the instance of JIRA in question.
   1. If a tool exists already, click the tool’s title to edit the tool.
   2. If a tool does not exist, click **Add Tool**.

Edit an Existing Work Item Source

1. To edit an existing work item source, locate the **Data Source** in question on the **Edit Tool** screen. You can edit the display name, while all other properties are system defined.

2. Click **Update** to save your changes.

> **NOTE:** While editing an existing source configuration, you can toggle a source from `Active` to `Inactive`, which will stop TeamForge from collecting data from that source; toggle back to `Active` to resume data collection from the source.

Create a New Work Item Source

1. If this is a new tool, first select the **Tool Type** from the drop-down list. For instance, if you're adding an external JIRA Tool instance, select the appropriate **Tool Type**.
2. Select the **Include Traceability** check box. A blank data source appears. If this is an existing tool with existing sources, click the **Data Source + (Add Source)** icon to add a new blank source.
    1. Provide a display name for the work item source. This display name will help you differentiate the source throughout the user interfaces where this source appears. The display name can be up to 100 alphanumeric characters in length.
3. Click **Update**.

    TeamForge EventQ saves the new work item source, and activates it.

Manage XDS Sources

Add Extensible Data Sources (XDS) to bring a wide variety of tools into the TeamForge project and traceability context.

XDS sources provide a means to integrate tools that do not fall under the stock activity classes (commit, build, review, and work item). You can add XDS sources to represent a wide range of tools and product domains.

Create or Find an Existing Tool to Group Your XDS Source Under

1. From the desired project context, navigate to **Project Admin > Tools**.
2. Determine whether a tool exists that represents a logical container for your desired source. If you are adding a Chef™ source, look for a tool representing the Chef instance in question.
    1. If a tool exists already, click the tool's title to edit the tool.
    2. If a tool does not exist, click **Add Tool**.

Edit an Existing XDS Source

1. To edit an existing XDS source, locate the **Data Source** in question on the **Edit Tool** screen. You can edit the display name, look and feel, and tags. If you save an XDS source without defining the **Associated Source**, you may edit the XDS source later and add an **Associated Source**. Once the **Associated Source** has been defined it may not be altered.

2. Click **Update** to save your changes.

> **NOTE:** While editing an existing source configuration, you can toggle a source from `Active` to `Inactive`, which will stop TeamForge from collecting data from that source; toggle back to `Active` to resume data collection from the source.

Create a New XDS Source

1. If this is a new tool, first select the **Tool Type** from the drop-down list. For instance, if you're adding a Nexus tool instance, select the matching XDS Schema name from the drop-down list. Alternatively, select **Other** if no applicable choice exists yet.
2. Select the **Include Traceability** check box. A blank data source appears. If this is an existing tool with existing sources, click the **Data Source + (Add Source)** icon to add a new blank source.
   1. Provide a display name for the XDS source that is depicted on all TeamForge EventQ user interfaces. The display name can be up to 100 alphanumeric characters in length.
   2. Select an icon and set the icon background color to differentiate your source visually.
   3. Select an Associated Source.
      - Select a source from the list of existing sources. TeamForge EventQ associates activities from the XDS source to the selected source. For instance, if you want the XDS source in question to associate to builds from job XYZ, select the source corresponding to job XYZ in the **Associated Source** drop-down list.
      - If you do not select an associated source, no associations will be drawn to the activities coming from this XDS source.

   4. Add tags to the Source as a means to organize and categorize XDS Sources. Tags are particularly useful as filters for reporting via the Reporting API.

      > **WARNING:** Tags are shared across the entire site. Exercise caution so that no superfluous tags are created which may make reporting more difficult.

3. Click **Update**.

   TeamForge EventQ saves the new custom source, and activates it.

The Activity Stream provides a feed of recent activities and updates from within the TeamForge project context. Activities include Tracker, Source Code, and external activities via EventQ sources.

Overview

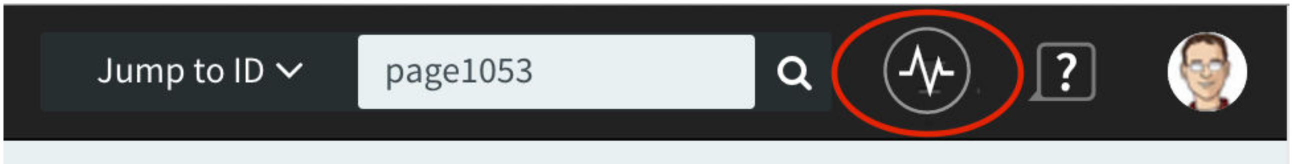
The Activity Stream is a feed of recent project activities, listed in reverse chronological order (newest activities on top). There is one activity stream per TeamForge project. The Activity Stream is useful to keep apprised of recent project updates without switching your context.

The Activity Stream includes events from:

- Tracker
- Source Code (Git and Subversion)
- Pull Request and Gerrit code reviews
- All configured EventQ activity sources (for example, Jenkins, JIRA®, Chef, Nexus, Artifactory, Testlink, Reviewboard, and so on)

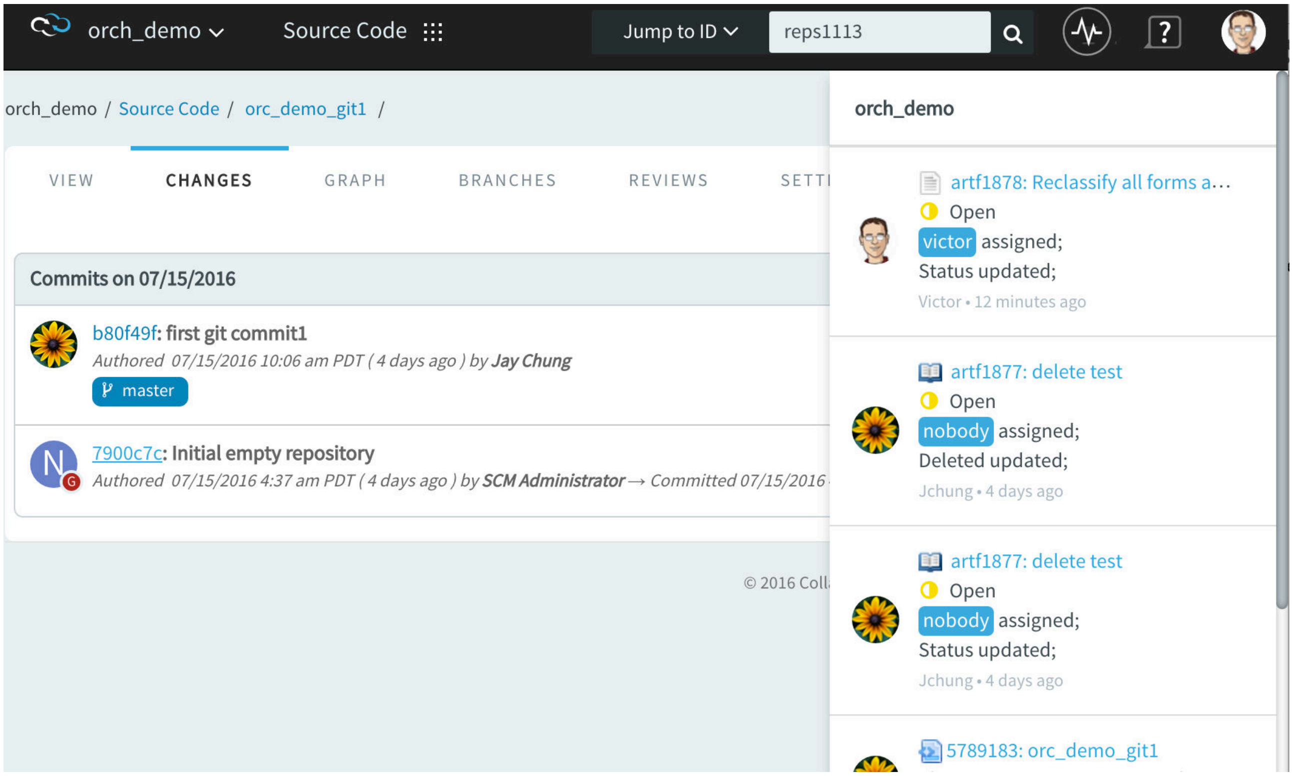
Usage

The Activity Stream is activated by clicking the “Activity Stream” icon in the TeamForge header (only available in a project context).



WARNING: In a distributed TeamForge setup where TeamForge and EventQ run on separate servers, you must make sure that there are no time-synchronization issues. If you see the following error message when you click the **Activity Stream** icon, check if the servers are in sync with network’s time. `Please contact your system Administrator. Something went wrong.`

When activated, the Activity Stream expands and lays over the page content, anchored to the right side of the browser. Activities appear in reverse chronological order (newest at the top) and are scrollable. As you scroll near the bottom, additional activities load automatically providing more scrolling real estate (and so on).



orch_demo / Source Code / orc_demo_git1 /

Commits on 07/15/2016

b80f49f: first git commit1
Authored 07/15/2016 10:06 am PDT (4 days ago) by ***Jay Chung***
master

7900c7c: Initial empty repository
Authored 07/15/2016 4:37 am PDT (4 days ago) by ***SCM Administrator*** *→ Committed 07/15/2016*

© 2016 Coll

orch_demo

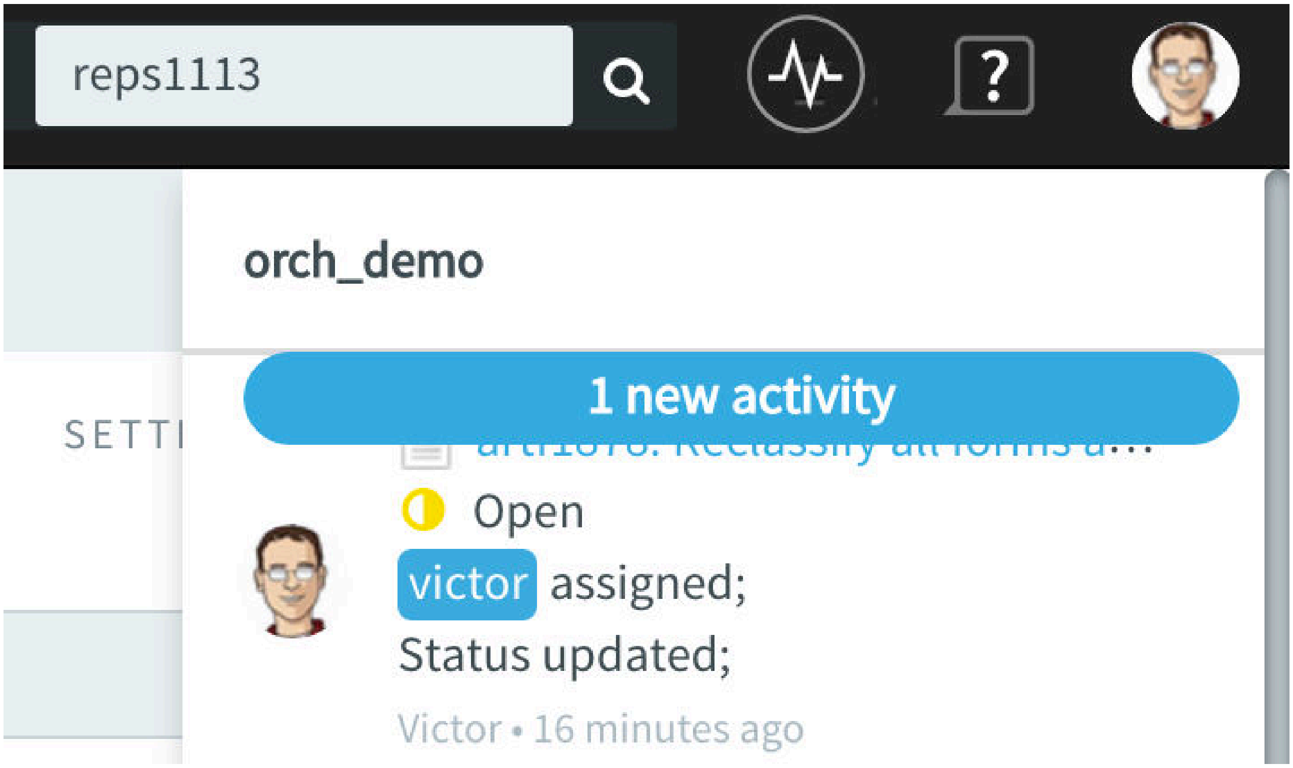
artf1878: Reclassify all forms a…
Open
victor assigned;
Status updated;
Victor • 12 minutes ago

artf1877: delete test
Open
nobody assigned;
Deleted updated;
Jchung • 4 days ago

artf1877: delete test
Open
nobody assigned;
Status updated;
Jchung • 4 days ago

5789183: orc_demo_git1

Each activity represents a distinct event in TeamForge and integrated tools. When available, you can click links and they will open the object in question in a new tab. You will see a notice in your Activity Stream when new activities occur. Click the notice to new activities.



Click the **Activity Stream** icon in the header a second time to collapse the Activity Stream.

IMPORTANT: Users must have the EventQ **PIPELINE READ** permission to access the Activity Stream.

Manage the TeamForge EventQ server-side components.
The scope of this topic is limited to key administrative features like starting/stopping services, log locations, and so on. Administration and configuration of dependent services (e.g., nginx, MongoDB, RabbitMQ) is not covered; see respective product documentation. It is assumed that as TeamForge EventQ administrator you are familiar with basic Linux administration commands and terminology.

Background Services

TeamForge EventQ installs and starts six background services on the App server. Each service is managed using init scripts, which are installed and started as part of the installation process. Init scripts are all located in `/etc/init.d/` on the respective hosts. Once started, each service creates a pid file.

Init Script	PID File	Host	Description
collabnet-mongod	/var/run/collabnet-mongodb/mongodb.pid	DB server	Database service
collabnet-nginx	/var/run/collabnet-nginx/nginx.pid	App server	Web and application server
collabnet-rabbitmq-server	/var/run/collabnet-rabbitmq/rabbitmq.pid	MQ server	AMQP message service

Init Script	PID File	Host	Description
collabnet-redis	/var/run/collabnet-redis/redis.pid	App server	Internal job queue service
eventq-ctf-jobs	/var/run/eventq/ctf_job_runner.pid	App server	Background communication with TeamForge
eventq-queue-commit	/var/run/eventq/scm_queue.pid	App server	Processes message from commit queue
eventq-queue-build	/var/run/eventq/build_queue.pid	App server	Processes message from build queue
eventq-queue-review	/var/run/eventq/review_queue.pid	App server	Processes message from review queue

Starting and Stopping Services

Use the init scripts to start, stop, restart, or get status on a service individually.

- **status** Is the process currently running? Does it have a PID (even if it's not running - e.g. in the case of a crash)
- **start** Starts the process
- **stop** Stop the process
- **restart** Stop and then start the process

Example Usage

```
sudo service eventq-queue-commit restart
```

While each service can be managed individually, the eventq init script will start and stop all App server services: `/etc/init.d/eventq` **Example Usage** `shell sudo service eventq start `` `

Service Administration Passwords

During the installation of TeamForge EventQ, if RabbitMQ and MongoDB were installed as part of the installation process then the administrative passwords have been written to disk for reference:

- **DB server**: `/opt/collabnet/mongodb/.mongo_passwords`
- **MQ server**: `/opt/collabnet/rabbitmq/.rabbit_passwords`

Log Files

The following log files are relevant to TeamForge EventQ services and may be useful for debugging. Log rotation on all log files is enabled by default.

EventQ Application Server Logs

- The `server.log` is the main log file for TeamForge EventQ, detailing the server's page serves and any errors that EventQ may encounter. It relates to the Phusion Passenger service, which is started with nginx. (`/var/log/eventq/`)
    - build_queue.log
    - ctf_job_runner.log
    - nginx-error.log
    - nginx.log
    - review_queue.log
    - scm_queue.log
    - server.log
- Redis Log: `/var/log/collabnet-redis/`
    - redis.log
- MongoDB Server Log: `/var/log/collabnet-mongodb/`
    - mongodb.log
- RabbitMQ Server Log: `/var/log/collabnet-rabbitmq/`
    - rabbit@localhost.log
    - rabbit@localhost-sasl.log
    - startup_err
    - startup_log

Monitoring TeamForge EventQ with New Relic

TeamForge EventQ may be configured for monitoring using New Relic, a third party service.

This section discusses how to set up monitoring of TeamForge EventQ using a third party product, New Relic. Once set up, configure New Relic to alert you of outages, monitor application performance, observe errors, and so on. It is assumed that you have a New Relic account and license key ready for use.

> **NOTE:** New Relic is not affiliated with CollabNet and may cost you for usage of their services.

1. Create a New Relic account and obtain your license key. If you haven't done so already, create an account on the New Relic web site. Obtain the license key associated with your account.
2. Contact CollabNet Support and request the TeamForge EventQ New Relic monitoring script.
3. Copy the New Relic script onto the TeamForge EventQ application server host (App server).
   `cp newrelic.sh /tmp cd /tmp chmod 755 newrelic.sh`
4. Execute the New Relic script with root privileges, passing your license key as an argument.

```
sudo ./newrelic.sh LICENSE_KEY
```

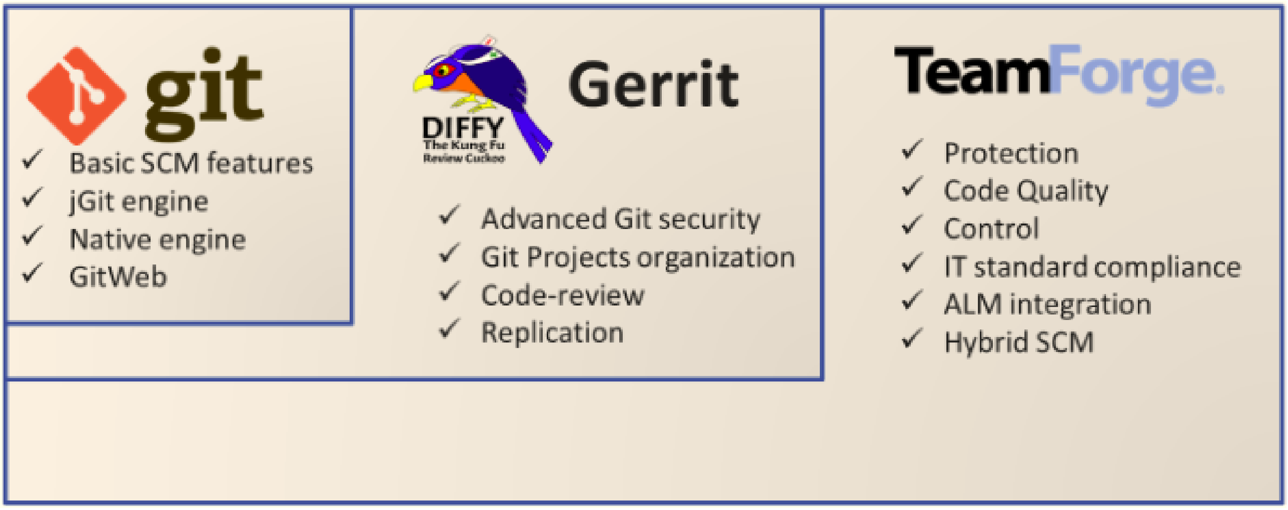
Where `LICENSE_KEY` is replaced with your New Relic provided license key.

After a few moments, your New Relic account will reflect the TeamForge EventQ server. Follow New Relic’s instructions for setting up monitoring, alerts or other desired functionality.

TeamForge supports integration with Git, a distributed version control tool powered by Gerrit.

Although Git is the world's leading distributed version control system, the enterprise has been slow and tentative in its adoption. Concerned with security breaches, compliance violations and lack of governance, many organizations have chosen to take a "wait and see" approach. With TeamForge, Git is ready for the enterprise. TeamForge lets you realize all the benefits of Git while ensuring the security, governance and manageability your business demands. With TeamForge, you can even manage Git and Subversion together, within each individual project.

Gerrit is an open source code review system designed to work with Git. Gerrit supports various access control mechanisms. The TeamForge Git integration uses Gerrit as a vehicle to bring TeamForge project roles and permissions into Git.



Install or Upgrade TeamForge-Git Integration

You can install Git on the TeamForge Application Server or on a separate server dedicated for SCM. For more information about installing and upgrading Git, see TeamForge install and upgrade instructions.

Git Integration Blog Posts

You can also read the CollabNet blog posts on Git and follow the latest developments in the CollabNet TeamForge-Git integration space.

Add Git as a Linked Application

Once you have installed Git, you can add Git as a linked application on your TeamForge site.

✓ In TeamForge 7.2 and later versions, installing Git for the first time creates a site-wide linked application automatically.

✓ In TeamForge 8.0 and later versions, in addition to a site-wide linked application, a project-wide linked application is also created for projects in TeamForge that have at least one CVS repository.

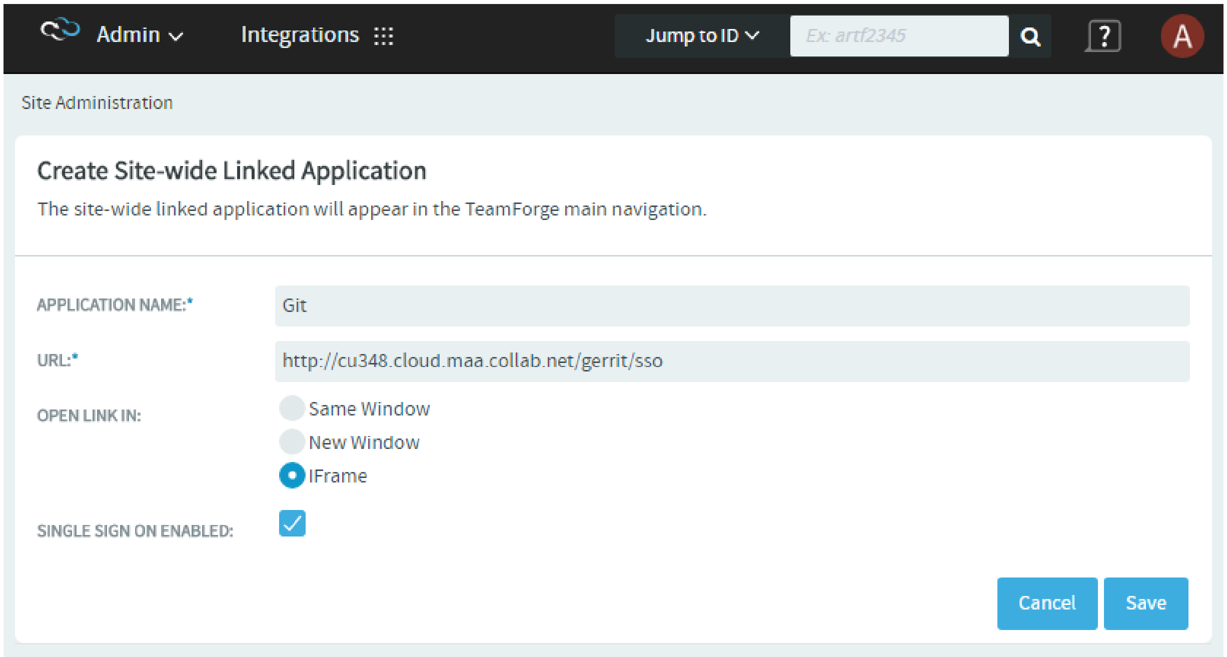
✓ However, this behavior can be controlled by the `teamforge.createTFProjectLinkedApps` Gerrit config (`gerrit.config`) property.

1. Set up the URL `http://<TEAMFORGEHOSTNAME>/gerrit/sso/`.

   > **NOTE:** The `/` at the end of the URL matters. Make sure you have it.

2. For instructions on setting up a site-wide linked application in TeamForge, see Create a Site-wide Linked Application.

   Here's an example for Git:

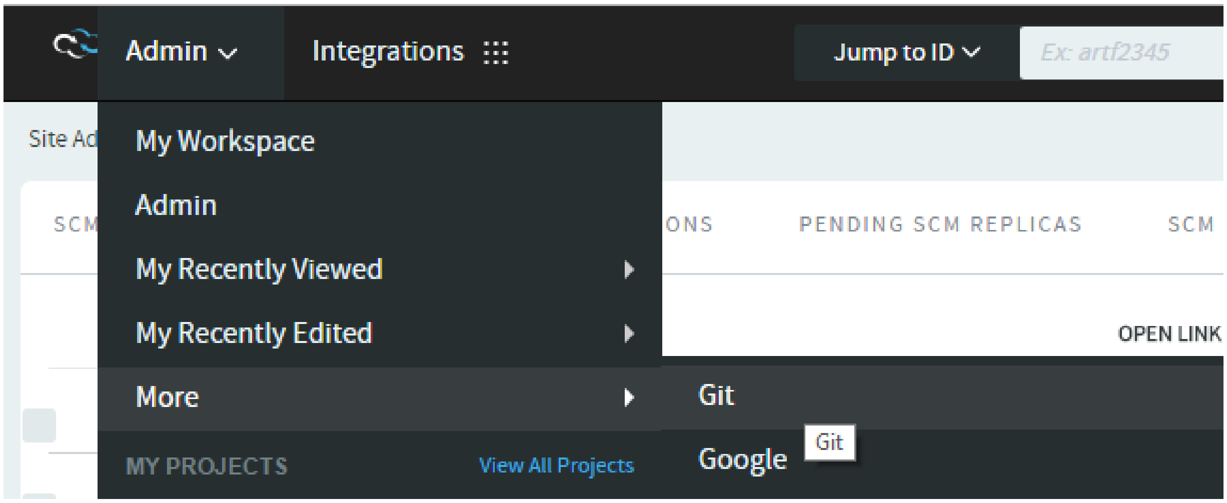


Create Site-wide Linked Application

The site-wide linked application will appear in the TeamForge main navigation.

APPLICATION NAME:*	Git
URL:*	http://cu348.cloud.maa.collab.net/gerrit/sso
OPEN LINK IN:	○ Same Window ○ New Window ● IFrame
SINGLE SIGN ON ENABLED:	☑

A link for Git is added to the More menu in your TeamForge navigation bar.

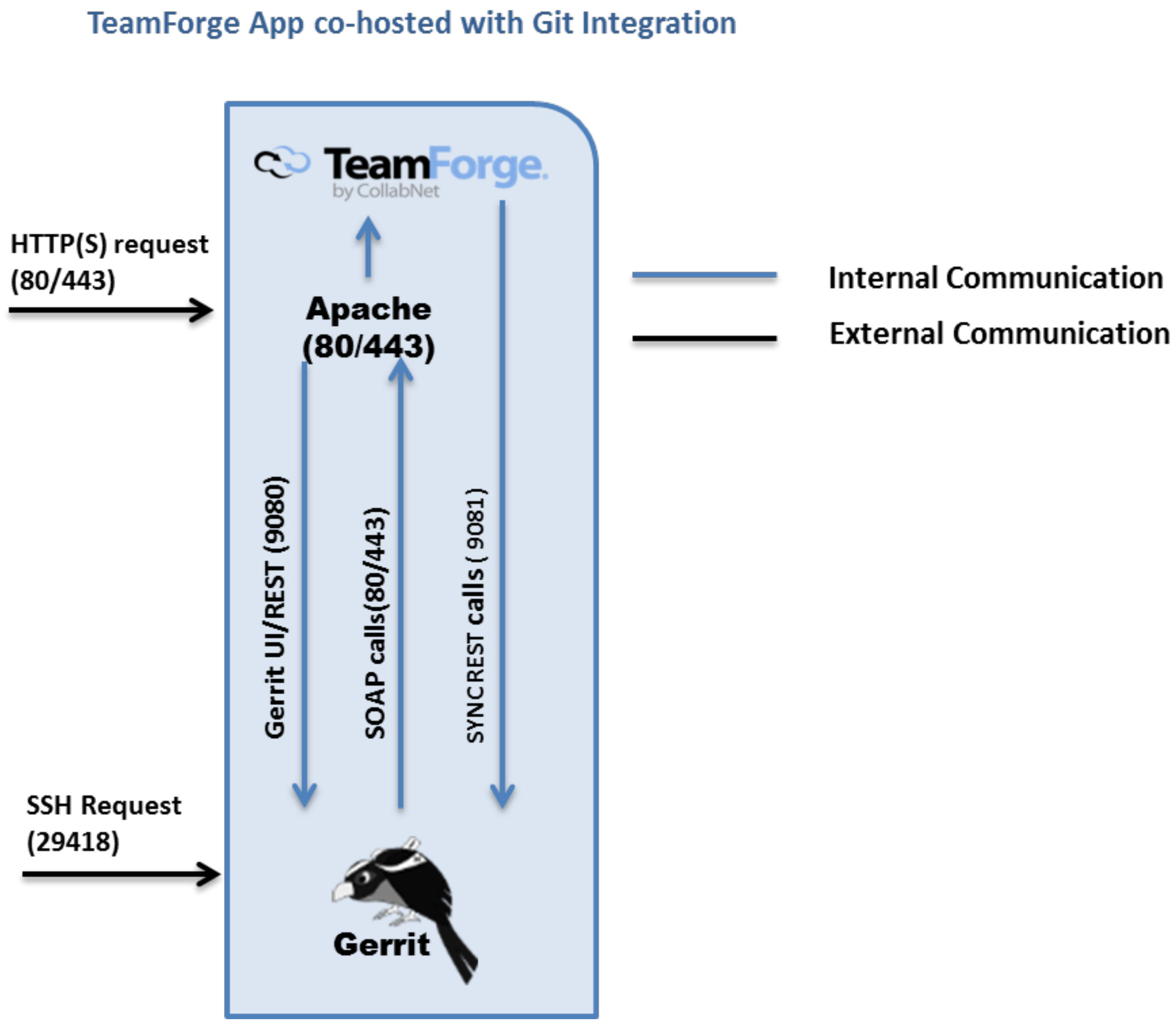


Clicking **Git** displays the Git console in the main TeamForge window.

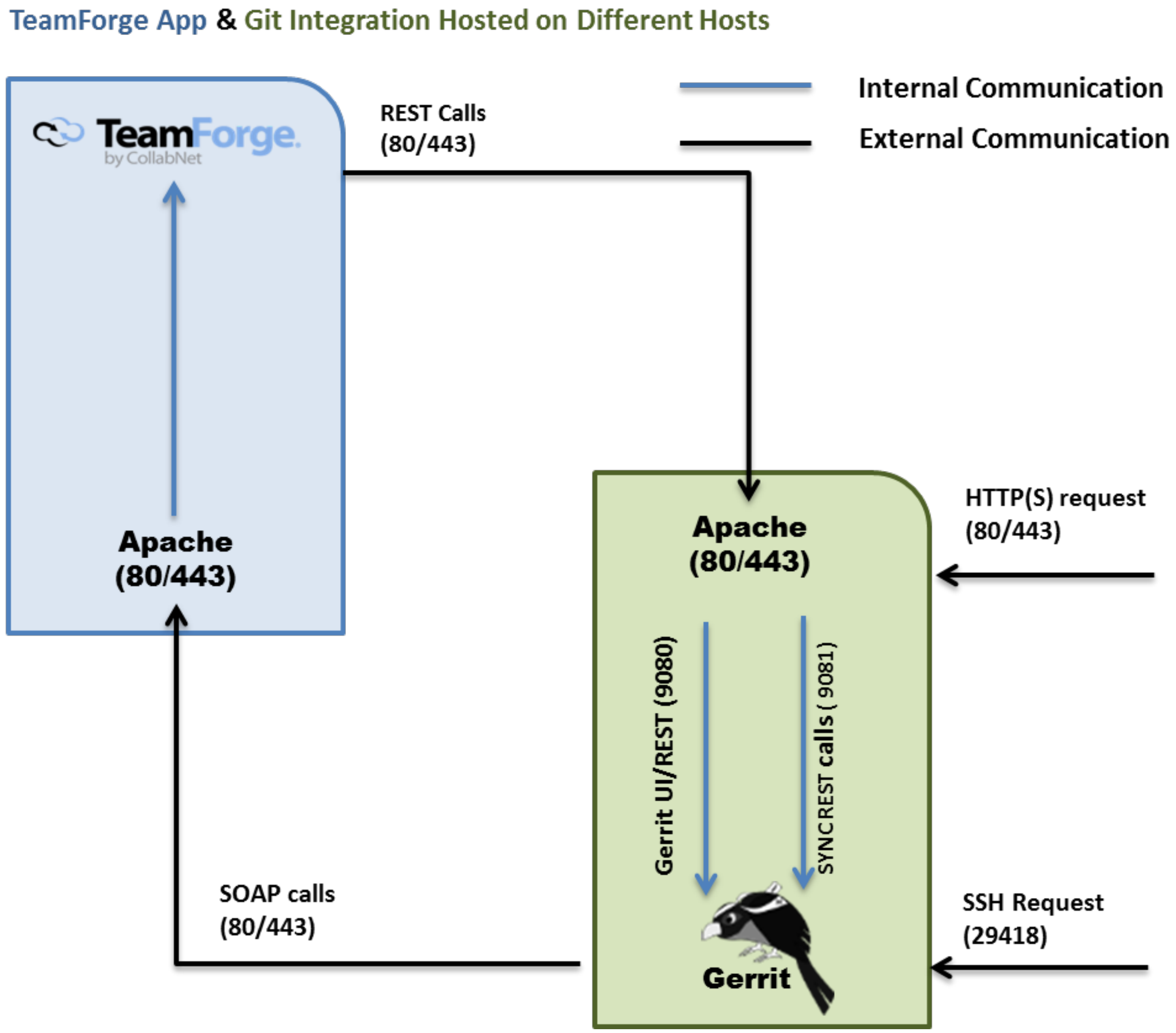
Illustrations on TeamForge-Gerrit Communication

The following illustrations help you understand the communication flow between TeamForge and Gerrit in a single host and distributed environments.

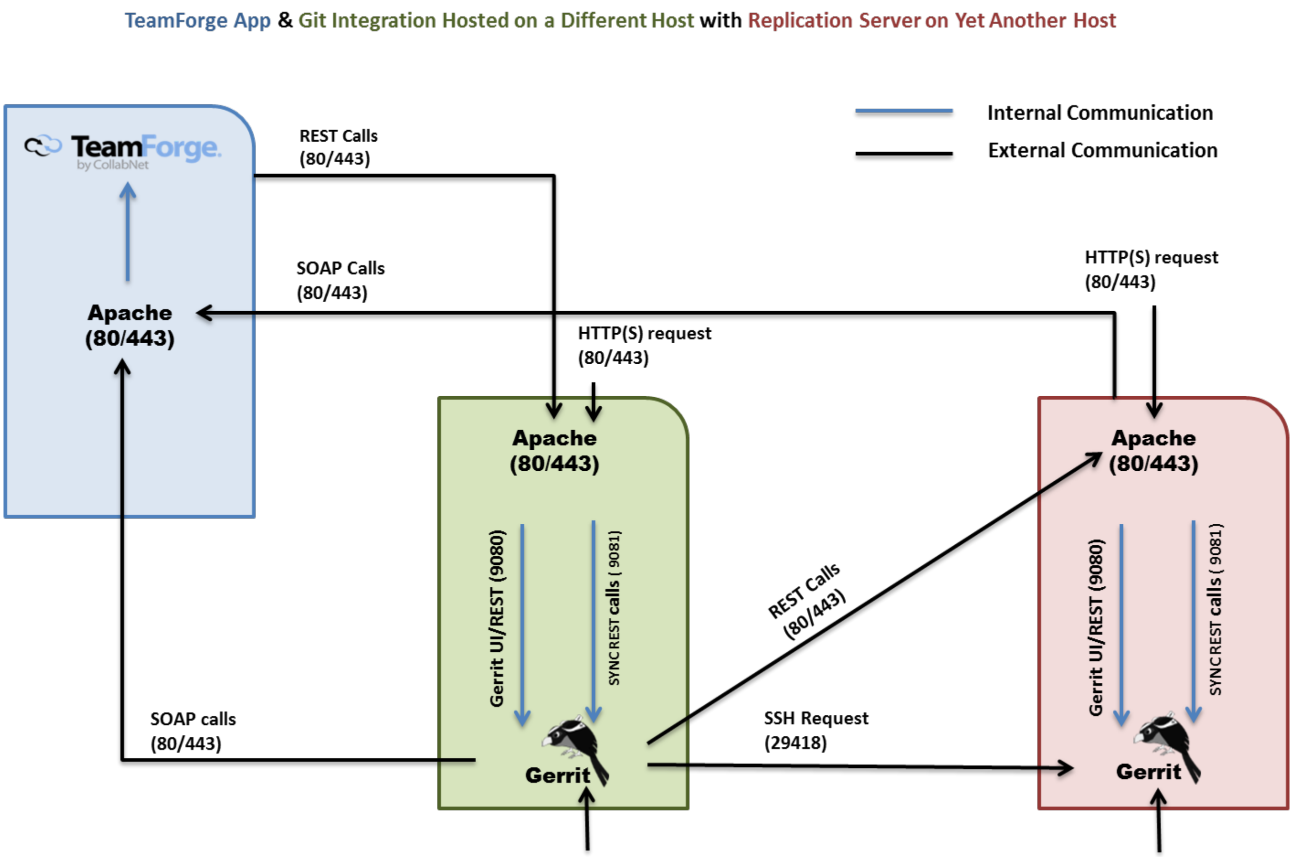
TeamForge and Git/Gerrit on a Single Host



TeamForge and Git/Gerrit in a Distributed Two-server Setup



TeamForge, Git/Gerrit and Replica Server in a Three-server Distributed Setup

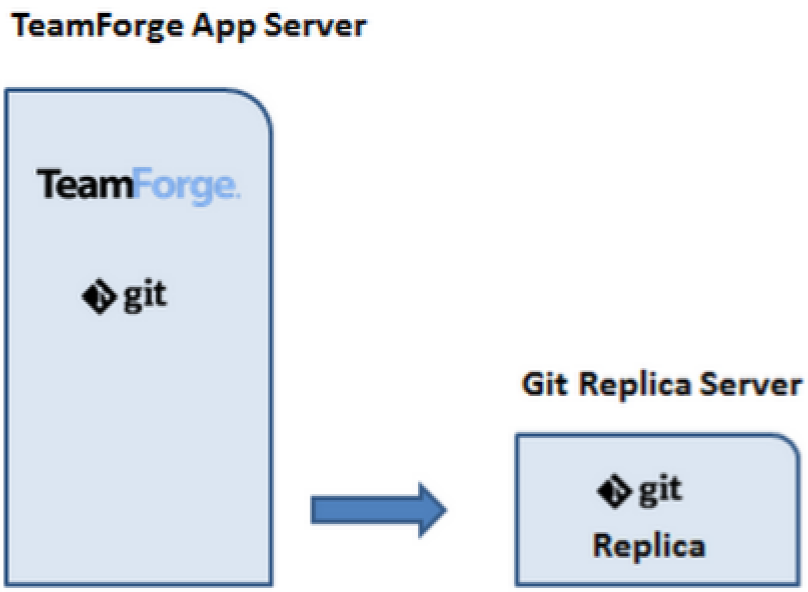


On sites distributed across multiple geographic locations, Git Replica Servers are local and remote mirror servers that can provide up-to-date copies of the central repositories. If set up, Git Replica Servers can address load balancing and fetch performance issues. You can set up one or more Git Replica Servers (also referred to as slave or mirror servers) with TeamForge 8.1 and later.

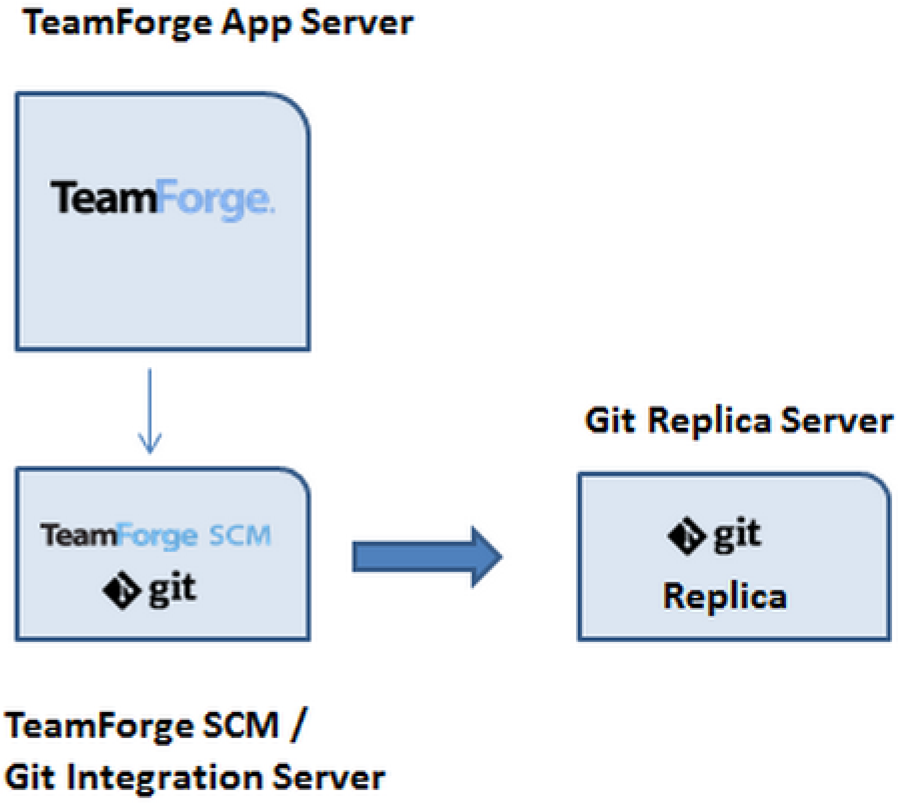
Set up a Git Replica Server

- A Git Replica Server has one and only one `master` Git server.
- It's not possible to set up both Git master and slave on the same server. However, you can have multiple master and slave servers in your TeamForge environment.
- Git replication servers can be set up with TeamForge 8.1 or later only.

You can have your master Git integration server installed on the TeamForge Application Server or on a separate server dedicated to Git/SCM integration.



TeamForge and Git (master) on the Same Server



TeamForge and Git (master) on Separate Servers

Before You Begin

- Make sure you have upgraded your Git integration to TeamForge-Git v8.4.6 or later.
- Have the master Git integration server's `externalSystemId` handy.

  Open the `/opt/collabnet/gerrit/etc/gerrit.config` file on the master Git integration server and note down the `externalSystemId` from the **[teamforge]** section.

  Alternatively, log on to the TeamForge Application Server as a Site Administrator, click **Admin > Integrations > SCM Integrations**, select the master Git integration server, click **Edit** and look for a token such as `exsy####`, for example `exsy1002`, in the browser URL. This is the external system ID of your Git integration server.
- Open the TeamForge Application Server's `site-options.conf` file and keep the values of the following tokens handy.
  ```
  SCM_DEFAULT_SHARED_SECRET=
  ```

  Note down the values of the following tokens if and only if obfuscation is enabled (`OBFUSCATION_ENABLED=true`):

  ```
  OBFUSCATION_ENABLED=
  OBFUSCATION_KEY=
  OBFUSCATION_PREFIX=
  AUTO_DATA=
  ```

1. Install Red Hat Enterprise Linux/CentOS 7.4 and log on as root.

   The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux.

   See the Red Hat Installation Guide for help.
2. Check your basic networking setup. See Set up Networking for more information.
3. Upgrade the operating system packages.
   ```
   yum upgrade
   ```
4. Reboot the server.
   ```
   reboot
   ```
5. Configure the TeamForge installation repository

TeamForge Installation Repository Configuration for Sites with Internet Access

1. Contact the CollabNet Support and download the TeamForge 18.1 installation repository package to `/tmp`.
2. Install the repository package.
   ```
   yum install -y /tmp/collabnet-teamforge-repo-18.1-0-noarch.rpm
   ```
3. Refresh your repository cache.
   ```
   yum clean all
   ```

TeamForge Installation Repository Configuration for Sites without Internet Access

1. Contact the CollabNet Support to get the auxiliary installer package for TeamForge 18.1 disconnected installation and save it in `/tmp`.
   - RHEL/CentOS 7.4 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel7.x86_64.rpm`
   - In addition to the above CentOS 7.4 64 bit RPM package, you must get the following CentOS 7.4 compatibility RPM, which is required for TeamForge 18.1 disconnected media installation on CentOS 7.4 profile: `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm`.
2. Unpack the disconnected installation package.
   ```
   rpm -ivh <package-name>
   ```
3. Unpack the `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm` package if you are installing TeamForge 18.1 on CentOS 7.4.
   ```
   rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
   ```
4. If not mounted already, mount the RHEL/CentOS installation DVD.

   The DVD contains the necessary software and utilities required for installing TeamForge without internet access. In the following commands, replace “cdrom” with the identifier for your server’s CD/DVD drive, if necessary.

   ```
   cd /media/
   mkdir cdrom
   mount /dev/cdrom ./cdrom/
   ```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

5. Create a yum configuration file that points to the RHEL/CentOS installation DVD.
   ```
   vi /etc/yum.repos.d/cdrom.repo
   ```
   Here's a sample yum configuration file.
   ```
   [RHEL-CDROM]
   name=RHEL CDRom
   baseurl=file:///media/cdrom/Server/
   gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
   enabled=1
   gpgcheck=0
   ```
6. Verify your yum configuration files.
   ```
   yum list httpd
   yum list apr
   ```

6. Install the Git packages.
   ```
   yum install teamforge-git
   ```
7. Set up the `site-options.conf` tokens for the Git Replica Server.
   ```
   vi /opt/collabnet/teamforge/etc/site-options.conf
   ```
   It is assumed that:

   ✓ `my.app.domain.com` is the Fully Qualified Domain Name (FQDN) of your TeamForge Application Server.

   ✓ `my.git.domain.com` is the Fully Qualified Domain Name (FQDN) of your Git Integration Server.

   ✓ `my.gitreplica.domain.com` is the Fully Qualified Domain Name (FQDN) of your Git Replica Server.

   1. Set up the `SERVICES` tokens.
      ```
      my.gitreplica.domain.com:SERVICES=gerrit gerrit-database
      my.app.domain.com:SERVICES=ctfcore ctfcore-database ctfcore-datamart e
      tl search subversion cvs binary binary-database
      ```
   2. Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file.
      To generate the SSL certificates, see Generate SSL Certificates.
      ```
      SSL=on
      SSL_CERT_FILE=
      SSL_KEY_FILE=
      SSL_CHAIN_FILE=
      ```

        - The SSL_CHAIN_FILE is optional.
        - If you use certificates that are generated in-house, self-signed, or signed by a non-established Certificate Authority, they must be registered with each client system that will connect to the TeamForge server.
        - For the setup discussed in this topic, add the certificate of `my.app.domain.com` to the JVM of `my.git.domain.com` and `my.gitreplica.domain.com`. In addition, add the certificate of `my.gitreplica.domain.com` to the JVM of `my.git.domain.com`. Click here for more information.
    3. Set the gerrit replication server mode.
       ```
       GERRIT_REPLICATION_MODE=slave
       ```
    4. Set the external system ID of the master Git integration server.
       ```
       GERRIT_REPLICATION_MASTER_EXTERNAL_SYSTEM_ID=exsy####
       ```
    5. Set the obfuscation related tokens.
    6. Save the `site-options.conf` file.
8. Provision services.
   ```
   teamforge provision
   ```

Now, the gerrit service is running in replica mode. You can now find the newly created Git Replica Server listed on TeamForge Application Server by accessing the following url: `http://<TF_HOST>/sf/sfmain/do/listSystems`.

Once you have set up one or more Git Replica Servers, you can replicate repositories.

Upgrade Git Replica Servers

> **IMPORTANT:** When upgrading TeamForge-Git integration servers, it is important that Git master and slave servers are upgraded to the same version of TeamForge-Git integration. On sites with Git Replica Servers, you must upgrade the Git Replica Servers first and then upgrade the master Git servers. For more information about upgrading master Git servers, see TeamForge upgrade instructions.

To upgrade existing Git Replica Servers:

1. Log on to the Git Replica Server and move the existing TeamForge repository from `/etc/yum.repos.d`.
2. Remove the `collabnet-teamforge-internal-repo.rpm`.
   ```
   yum erase collabnet-teamforge-internal-repo rpm
   ```
3. Configure the TeamForge installation repository

TeamForge Installation Repository Configuration for Sites with Internet Access

1. Contact the CollabNet Support and download the TeamForge 18.1 installation repository package to `/tmp`.
2. Install the repository package.
   ```
   yum install -y /tmp/collabnet-teamforge-repo-18.1-0-noarch.rpm
   ```
3. Refresh your repository cache.
   ```
   yum clean all
   ```

TeamForge Installation Repository Configuration for Sites without Internet Access

1. Contact the CollabNet Support to get the auxiliary installer package for TeamForge 18.1 disconnected installation and save it in `/tmp`.
   - RHEL/CentOS 7.4 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel7.x86_64.rpm`
   - In addition to the above CentOS 7.4 64 bit RPM package, you must get the following CentOS 7.4 compatibility RPM, which is required for TeamForge 18.1 disconnected media installation on CentOS 7.4 profile: `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm`.
2. Unpack the disconnected installation package.
   ```
   rpm -ivh <package-name>
   ```
3. Unpack the `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm` package if you are installing TeamForge 18.1 on CentOS 7.4.
   ```
   rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
   ```
4. If not mounted already, mount the RHEL/CentOS installation DVD.

   The DVD contains the necessary software and utilities required for installing TeamForge without internet access. In the following commands, replace “cdrom” with the identifier for your server’s CD/DVD drive, if necessary.

   ```
   cd /media/
   mkdir cdrom
   mount /dev/cdrom ./cdrom/
   ```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

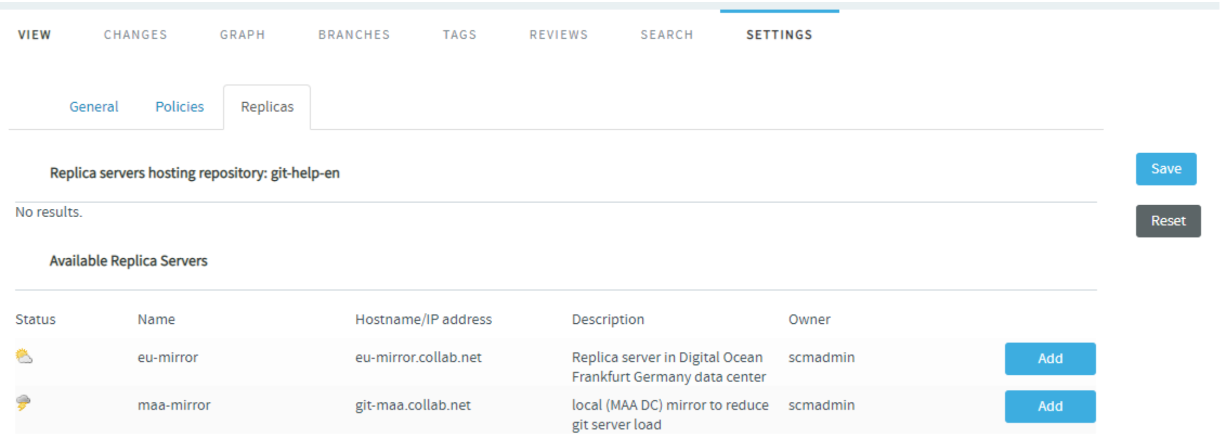
5. Create a yum configuration file that points to the RHEL/CentOS installation DVD.
   ```
   vi /etc/yum.repos.d/cdrom.repo
   ```
   Here’s a sample yum configuration file.
   ```
   [RHEL-CDROM]
   name=RHEL CDRom
   baseurl=file:///media/cdrom/Server/
   gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
   enabled=1
   gpgcheck=0
   ```
6. Verify your yum configuration files.
   ```
   yum list httpd
   yum list apr
   ```

4. Refresh your repository cache.
   ```
   yum clean all
   ```
5. Upgrade the Git packages.
   ```
   yum install teamforge-git
   ```
6. Provision services.
   ```
   teamforge provision
   ```

Replicate Repositories with Git Replica Servers

It is assumed that you already have one or more Teamforge projects that consists of one or more Git repositories that you want to replicate.

1. To start replicating a repository–go to the TeamForge project–select the Git repository you want to replicate, select the **Settings** tab and then select the **Replicas** tab.



Replica servers hosting repository: git-help-en

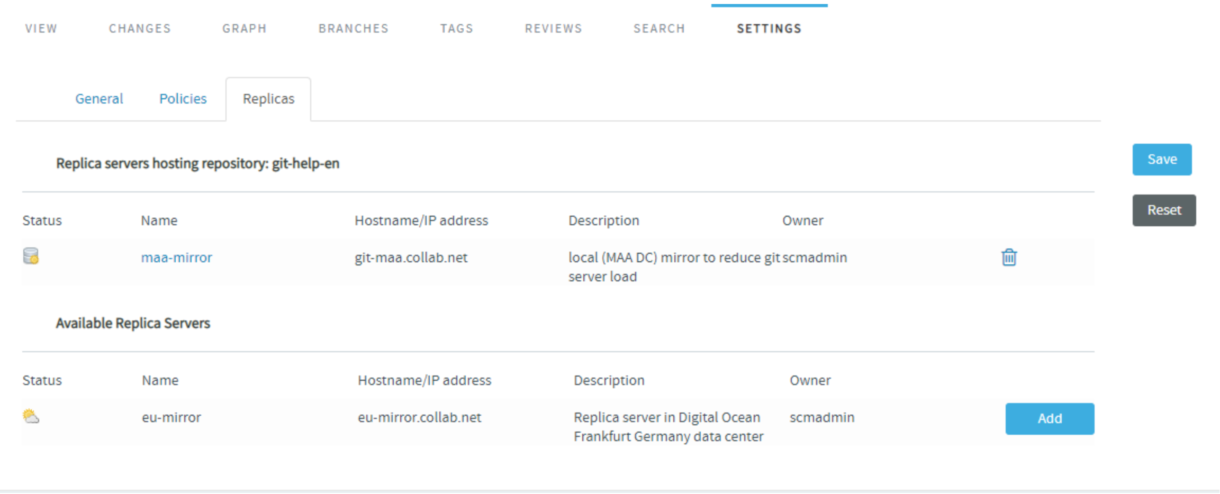
No results.

Available Replica Servers

Status	Name	Hostname/IP address	Description	Owner	
⛅	eu-mirror	eu-mirror.collab.net	Replica server in Digital Ocean Frankfurt Germany data center	scmadmin	Add
⛈	maa-mirror	git-maa.collab.net	local (MAA DC) mirror to reduce git server load	scmadmin	Add

This page lists the available Git Replica Servers.

2. From the list of Replica Servers, click the **Add** button of one or more Replica Servers to have the server(s) replicate the selected repository.



Replica servers hosting repository: git-help-en

Status	Name	Hostname/IP address	Description	Owner	
🛢	maa-mirror	git-maa.collab.net	local (MAA DC) mirror to reduce git server load	scmadmin	🗑

Available Replica Servers

Status	Name	Hostname/IP address	Description	Owner	
⛅	eu-mirror	eu-mirror.collab.net	Replica server in Digital Ocean Frankfurt Germany data center	scmadmin	Add

3. Click **Save**.
4. Push a commit and verify if it's replicated on the Replica Servers.

Related Links

- Replicate a Subversion Repository

Git Large File Storage (LFS) is a Git extension for versioning large files. Git LFS replaces large files such as audio samples, videos, datasets, and graphics with text pointers inside Git, while storing the file contents on a separate server (typically a remote server).

LFS is supported by TeamForge-Git/Gerrit integration 16.7.10-2.13.2 and later. LFS is controlled by two levels of configuration in TeamForge. First, integration level LFS configuration that provides default values for a given Gerrit instance. Second, repository level LFS configuration, which by default derives system level configuration that can be further adjusted.

In practice, it is assumed that the Gerrit integration server is LFS ready by default and one (Project Owner/ Site Admin) decides on enabling LFS at the repository level with or without maximum object size limitation. This configuration scenario supports a model where LFS is enabled for specific repositories only while the rest of the system remains unaffected.

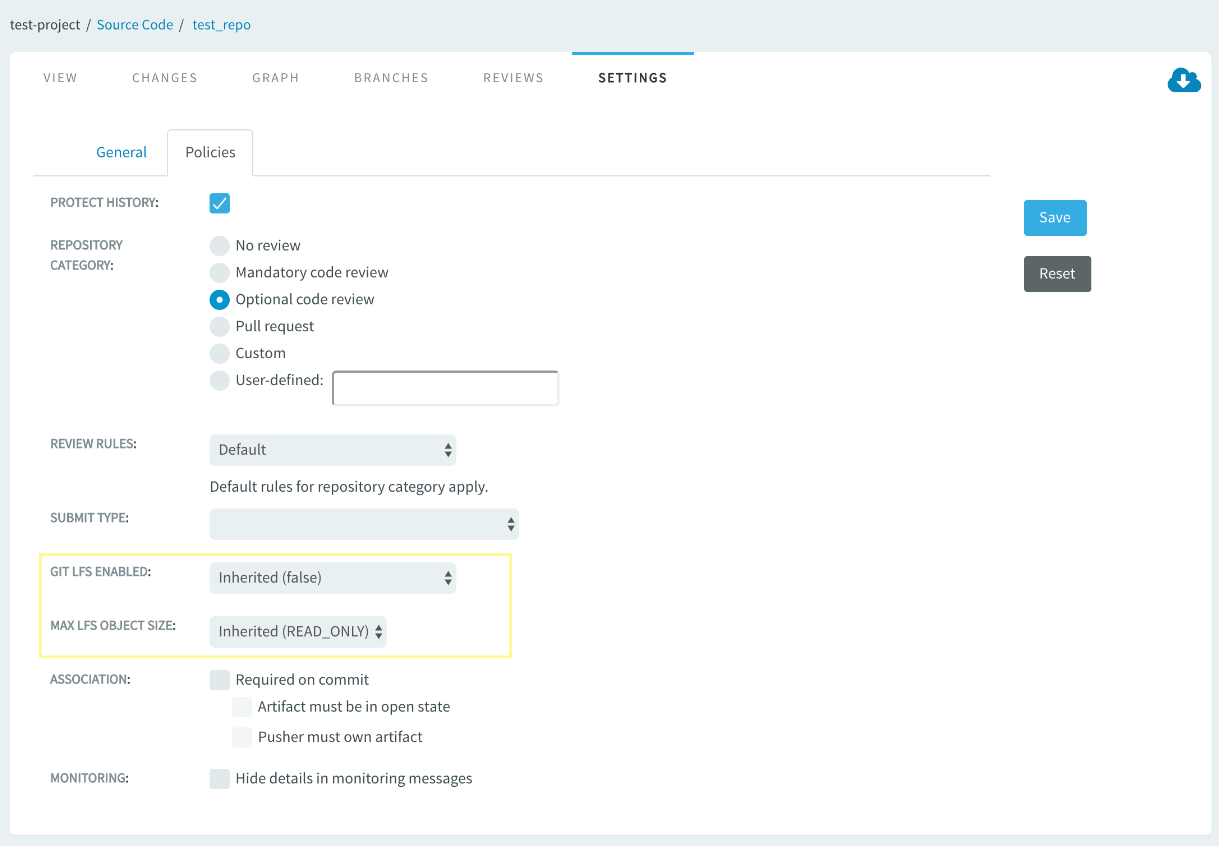
Enable LFS for a Repository

You can enable LFS for both existing and new repositories.

This section provides instructions to set up LFS for both exiting and new repositories.

Setting up LFS for Existing Repositories

1. Log on to TeamForge, select **Project Home > Source Code**, and select (click) a repository.
2. Select **Settings > Polices**.
3. To enable LFS, you must select the values for **MAX LFS OBJECT SIZE** and **GIT LFS ENABLED** fields.



test-project / Source Code / test_repo

PROTECT HISTORY: ☑

REPOSITORY CATEGORY:
○ No review
○ Mandatory code review
● Optional code review
○ Pull request
○ Custom
○ User-defined:

REVIEW RULES: Default

Default rules for repository category apply.

SUBMIT TYPE:

GIT LFS ENABLED: Inherited (false)

MAX LFS OBJECT SIZE: Inherited (READ_ONLY)

ASSOCIATION: □ Required on commit
□ Artifact must be in open state
□ Pusher must own artifact

MONITORING: □ Hide details in monitoring messages

MAX LFS OBJECT SIZE (required field)

Select one of the values: `Inherited`, `Unlimited`, `Read-only` or `Limited to`.

By default Inherited value is READ_ONLY. It means that once LFS data is pushed into repository it is always available for fetch/clone operation. Even if you switch to Unlimited, for example, and then decide to go back to READ_ONLY at a later point in time for a given repository or integration, repository consistency is preserved and data would always be available. This is necessary to prevent situations where crucial binary data is always readable unless you rewrite the repository history to render such binary data unavailable. Select:

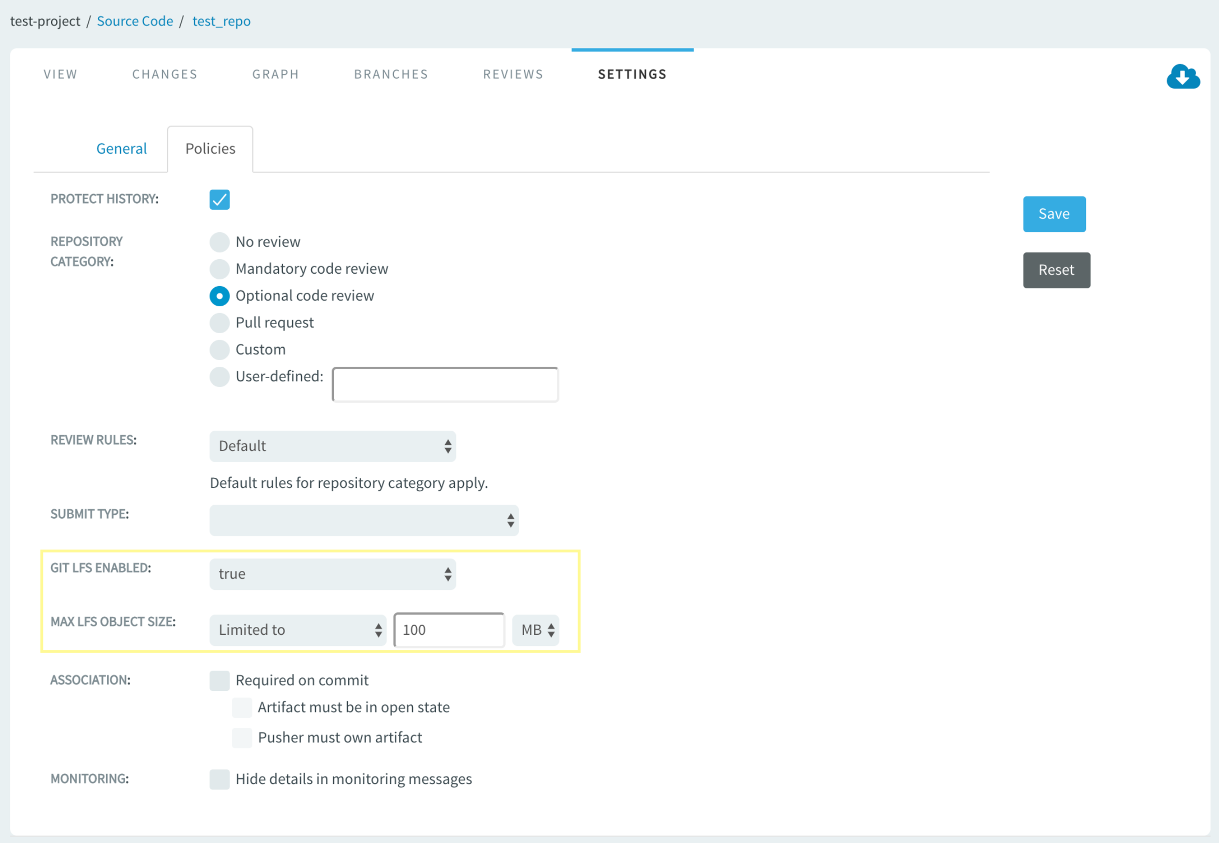
- **Inherited** that makes this repository inherit the default Git-integration settings. Note that in case of Inherited, current default integration setting is shown for your reference.
- **Unlimited** to support unlimited object size (size is not proactively limited by Gerrit but space availability still applies).
- **Limited** to to limit maximum object size to a reasonable value, for example, 100MB.
- **Read-only** that turns LFS in this repository to read-only mode.

GIT LFS ENABLED (optional field)

Select one of the values: `Inherited`, `true` or `false`.

The inherited value is `false` by default. LFS is served from master Gerrit instance over HTTP/HTTPS protocol and therefore you must enable this parameter to extend the checkout URLs with LFS specific part for SSH protocol and replication. This is required as the LFS client, by default, uses the same URL (derives the protocol from it) that is used for fetch/clone/push operations, while pointing to the master Gerrit instance over HTTP/HTTPS to read/write data.

The following illustration shows a typical LFS configuration where the **MAX LFS OBJECT SIZE** is limited to 100 MB for a repository that's server over SSH:



test-project / Source Code / test_repo

PROTECT HISTORY: ☑

REPOSITORY CATEGORY:
○ No review
○ Mandatory code review
● Optional code review
○ Pull request
○ Custom
○ User-defined:

REVIEW RULES: Default

Default rules for repository category apply.

SUBMIT TYPE:

GIT LFS ENABLED: true

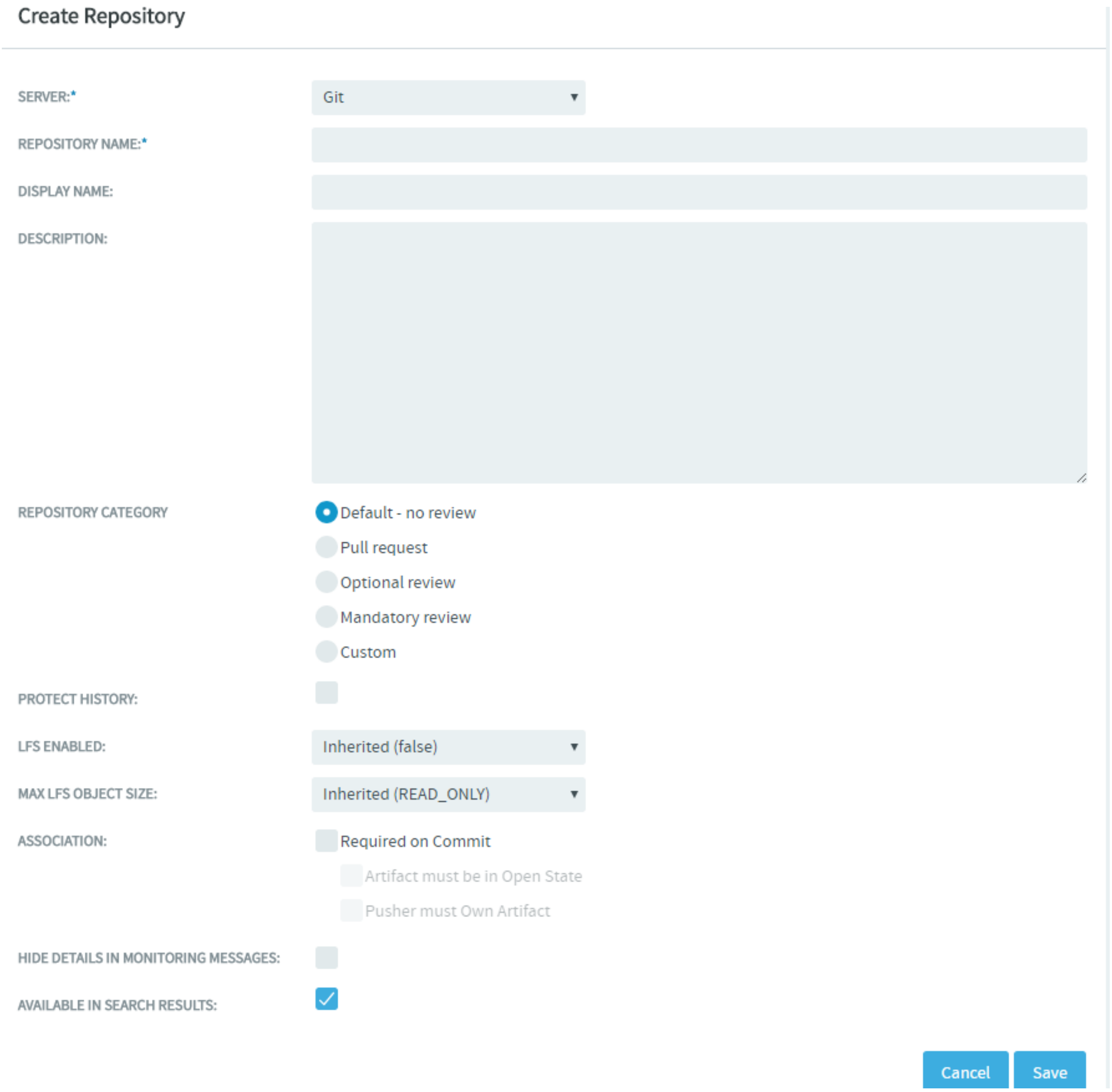
MAX LFS OBJECT SIZE: Limited to 100 MB

ASSOCIATION:
□ Required on commit
□ Artifact must be in open state
□ Pusher must own artifact

MONITORING: □ Hide details in monitoring messages

Setting up LFS When You Create a New Repository

1. Log on to TeamForge, select **Project Home > Source Code**, and click **Create Repository**.



Create Repository

SERVER:* Git ▾

REPOSITORY NAME:*

DISPLAY NAME:

DESCRIPTION:

REPOSITORY CATEGORY
● Default - no review
○ Pull request
○ Optional review
○ Mandatory review
○ Custom

PROTECT HISTORY: □

LFS ENABLED: Inherited (false) ▾

MAX LFS OBJECT SIZE: Inherited (READ_ONLY) ▾

ASSOCIATION: □ Required on Commit
□ Artifact must be in Open State
□ Pusher must Own Artifact

HIDE DETAILS IN MONITORING MESSAGES: □

AVAILABLE IN SEARCH RESULTS: ☑

2. Select the values for **MAX LFS OBJECT SIZE** and **GIT LFS ENABLED** fields while creating the new repository and click **Save**.

Set up LFS Client and Work with Large Files

Download the Git LFS client for your platform. These instructions are valid for Git LFS client 1.3 and later. The checkout URLs are automatically extended with LFS part and you do not have to modify the checkout URL manually to have it working out of the box for SSH protocol or replication scenarios.

For downloading Git LFS client, see Git Large File Storage page.

Working with LFS over HTTP/HTTPS (Without Replication)

1. Use extended checkout URL. Example:

   ```
   git clone -c
         'lfs.url=http://product_developer@main.server.collab.net/gerrit/test
   _repo.git/info/lfs'
         ssh://product_developer@gerrit.server.collab.net:29418/test_repo &&
   cd "test_repo" && git
         config user.name "Nancy S." && git config user.email "nancy@example.co
   m" && git config
         url."ssh://gerrit.server.collab.net:29418".insteadOf "ssh://main.ser
   ver.collab.net:29418"
         && git config url."ssh://product_developer@gerrit.server.collab.net:2
   9418".insteadOf
         "ssh://product_developer@main.server.collab.net:29418" && git config
         url."ssh://product_developer@main.server.collab.net:29418".pushInste
   adOf
         "ssh://product_developer@gerrit.server.collab.net:29418" && scp -P 2
   9418
         product_developer@gerrit.server.collab.net:hooks/commit-msg .git/hoo
   ks/
   ```

2. Select the file types you'd like Git LFS to manage. You can configure additional file extensions anytime.

   The following command tracks all `.jpg` images in a given working directory.

   ```
   git lfs track *.jpg
   ```

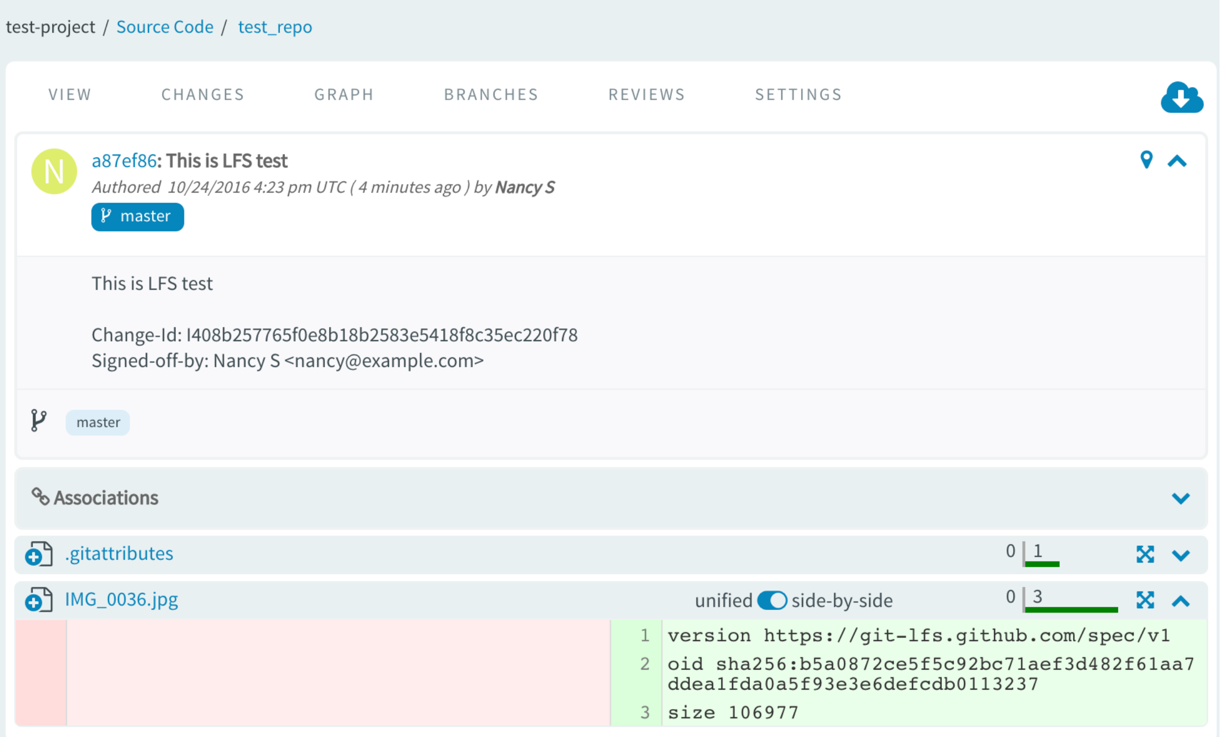
3. Create a commit by adding the binary file and the technical file (`.gitattributes`) that is modified by Git LFS client.

   ```
   git add IMG_0036.jpg .gitattributes
   ```

4. Commit and push the file(s) to the remote repository.

   ```
   git commit -sm 'This is LFS test' && git push origin HEAD:master
   ```

   The binary file that is successfully pushed to LFS manifests itself by having a reference file committed to Gerrit and you can check its content by going to TeamForge code browser for the given repository. Here is an example reference file. It contains Git LFS protocol version specification along with Git LFS object SHA and its size.



test-project / Source Code / test_repo

a87ef86: This is LFS test
Authored 10/24/2016 4:23 pm UTC (4 minutes ago) by Nancy S

This is LFS test

Change-Id: I408b257765f0e8b18b2583e5418f8c35ec220f78
Signed-off-by: Nancy S <nancy@example.com>

.gitattributes 0 | 1

IMG_0036.jpg unified side-by-side 0 | 3

```
version https://git-lfs.github.com/spec/v1
oid sha256:b5a0872ce5f5c92bc71aef3d482f61aa7
ddea1fda0a5f93e3e6defcdb0113237
size 106977
```

You can control all Gerrit Code Review features directly from TeamForge by specifying a code review policy as part of the TeamForge Git repository's description field.

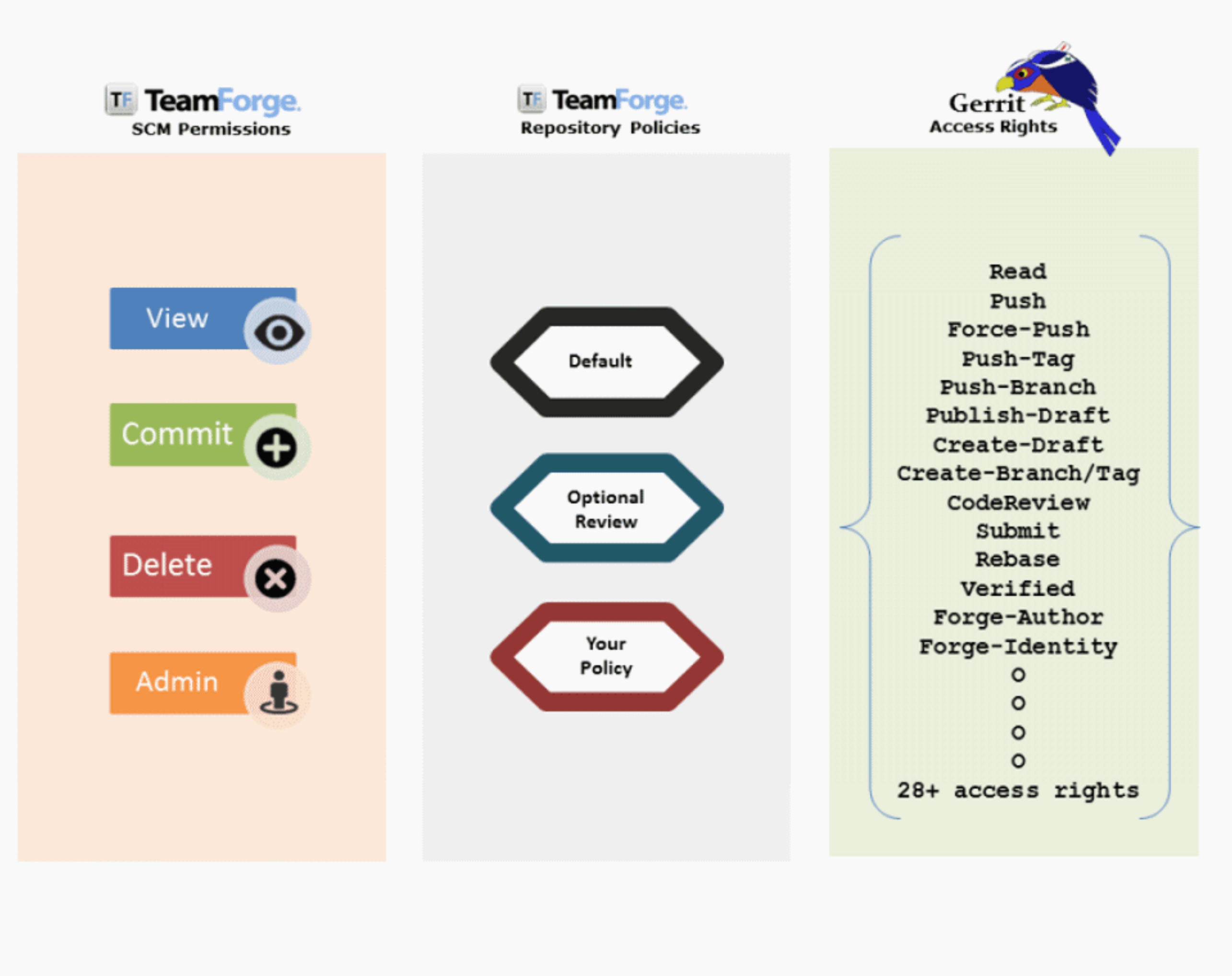
For more information on Gerrit Code Review, see the Gerrit documentation.

By default, the following code review policy options are provided:

- **No review**: All Gerrit review features are turned off and read/write access is enforced.
- **Mandatory** review: All code changes must be reviewed and read/write access is enforced.
- **Optional** review: The review feature is turned on but can be bypassed if necessary; read/write access is enforced.
- **Pull request** review: Pull requests allow developers to collaborate with each other on a code change before merging it into another branch on a Git repository. Using a pull request, you notify others about a feature or fix change that needs attention.
- **Custom**: Access rights must be set manually in the Gerrit web interface; they will not be overridden by TeamForge. This specification is intended for advanced users who are familiar with Gerrit access rights and want to turn off “auto pilot”.
- **User-defined** review: You can add your own categories, if you have access to the `TeamForgeGerritMappings.xml` file. For more information on adding a user-defined repository category, see Create a User-defined Repository Category.

If you have access to the `TeamForgeGerritMappings.xml` file, you can add your own categories.

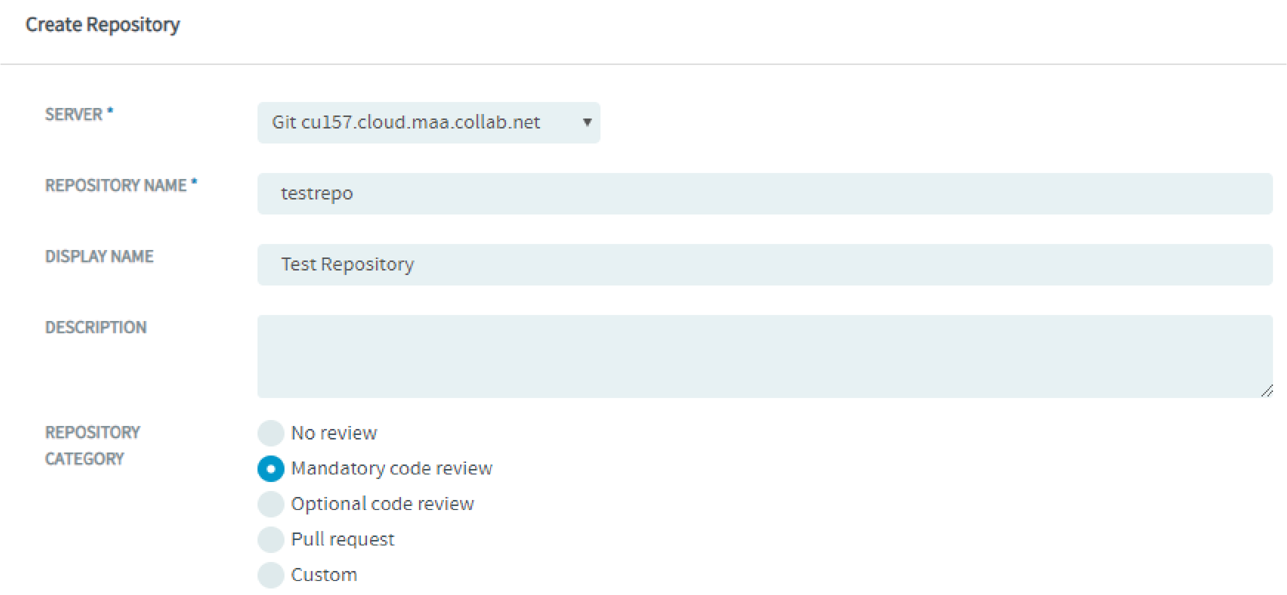
The following animation illustrates the detailed mapping between SCM permissions, repository policies, and Gerrit access rights.



Mapping between SCM permissions, repository policies, and Gerrit access rights

Mandatory Code Reviews for Git Repositories

When a mandatory review is specified, every change pushed to the repository must pass through a review process before it can get committed (merged) to the repository.



Create Repository

SERVER * Git cu157.cloud.maa.collab.net

REPOSITORY NAME * testrepo

DISPLAY NAME Test Repository

DESCRIPTION

REPOSITORY CATEGORY
○ No review
● Mandatory code review
○ Optional code review
○ Pull request
○ Custom

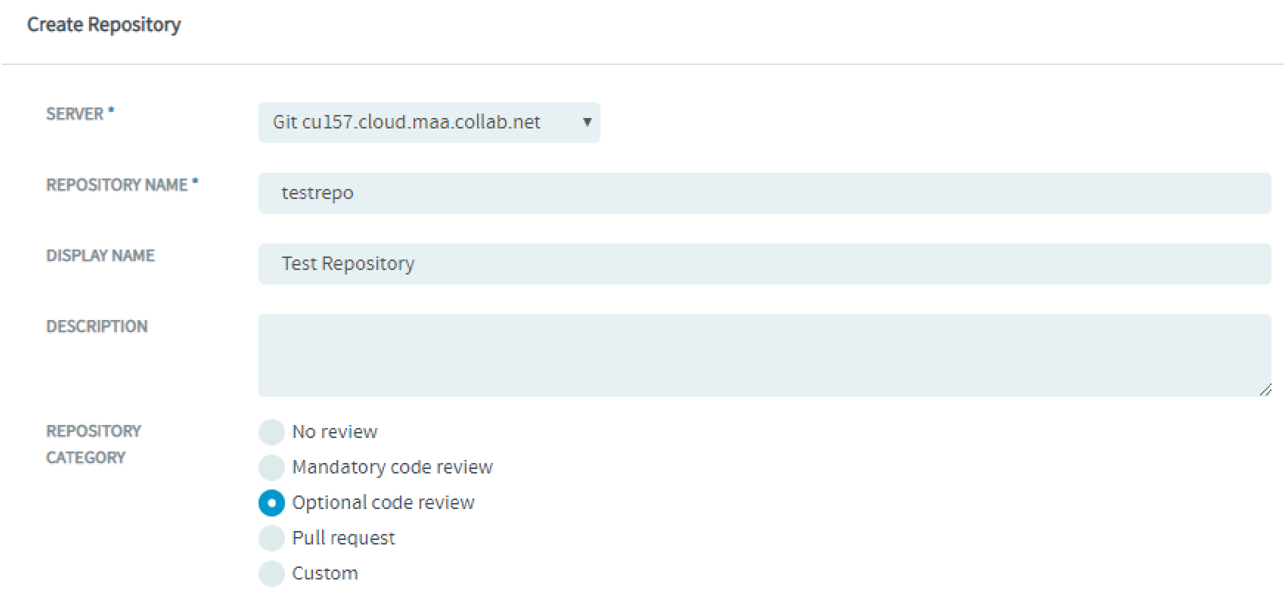
NOTE: Only TeamForge users with the **Source Code Admin** permission can bypass reviews.

Here's a list of permissions and what users with these permissions can do:

- No access: Users with no permissions cannot do anything.
- View only: Users with read permissions can read branches and push for reviews, and have -1 and +1 for reviews.
- Commit/View: Users with commit permissions can do everything read permissions would grant and in addition have -2, +2 for reviews. They can verify and submit permissions but have no right to bypass reviews.
- Delete/View: Users with delete permissions can do everything commit permissions would grant.
- Source Code Admin: Users with admin permissions can do everything delete permissions would grant and in addition push to and create any branch (bypassing review). They can rewrite history, forge the identity of the Gerrit server, and have the right to push tags, the right to upload merges, and the right to fine tune access rights in Gerrit for the Gerrit project involved.

Optional Code Review for Git Repositories

When an optional review is specified, every change submitted to the repository can be pushed for code review or directly pushed to the repository bypassing review. This depends on the TeamForge user having the appropriate permissions — source code Delete/View or Commit/View permission for the former, or Source Code Admin permission for the latter.



Create Repository

SERVER * Git cu157.cloud.maa.collab.net

REPOSITORY NAME * testrepo

DISPLAY NAME Test Repository

DESCRIPTION

REPOSITORY CATEGORY

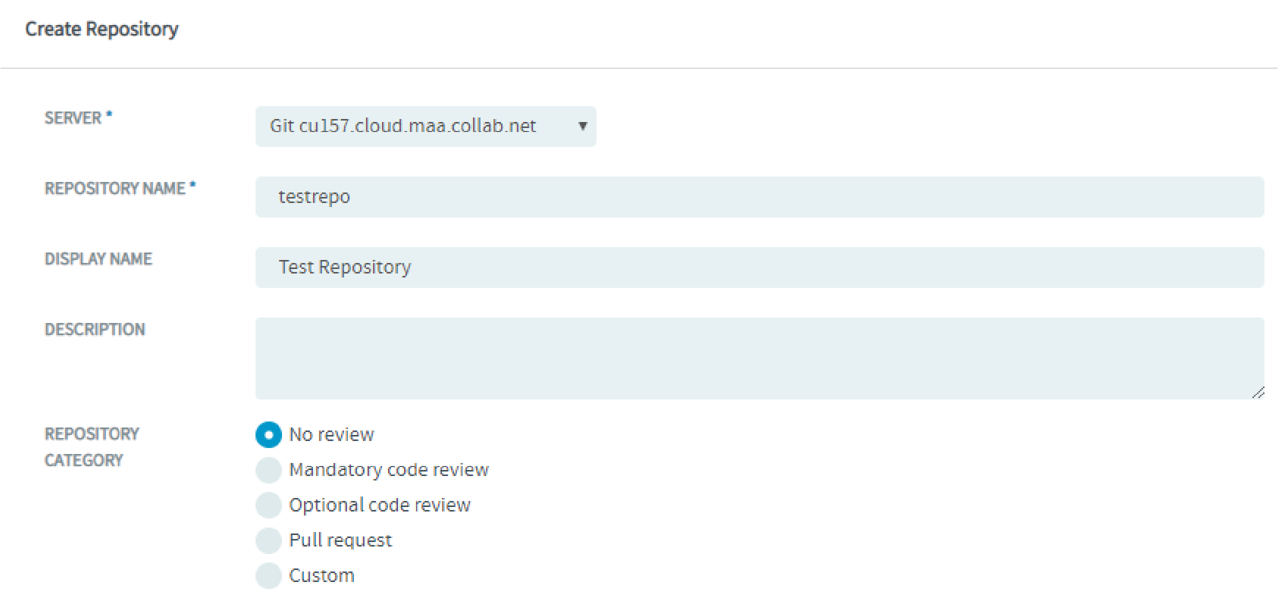
○ No review
○ Mandatory code review
● Optional code review
○ Pull request
○ Custom

Here's a list of permissions and what users with these permissions can do:

- No access: Users with no permissions cannot do anything.
- View only: Users with read permissions can read branches and push for reviews, and have -1 and +1 for reviews.
- Commit/View: Users with commit permissions can do everything read permissions would grant and in addition have -2, +2 for reviews. They can verify and submit permissions, push to/create any branch (bypassing review) and push tags.
- Delete/View: Users with delete permissions can do everything commit permissions would grant and in addition, have the right to rewrite history, upload merges and forge identity.
- Source Code Admin: Users with admin permissions can do everything delete permissions would grant and in addition push to/create any branch (bypassing review). They can rewrite history, forge the identity of the Git server, and have the right to push tags, the right to upload merges, and the right to fine tune access rights in Git for the Git project involved.

No Code Review for Git Repositories

In TeamForge 8.0 and later, the **No review** policy is selected unless you choose some other policy.



Create Repository

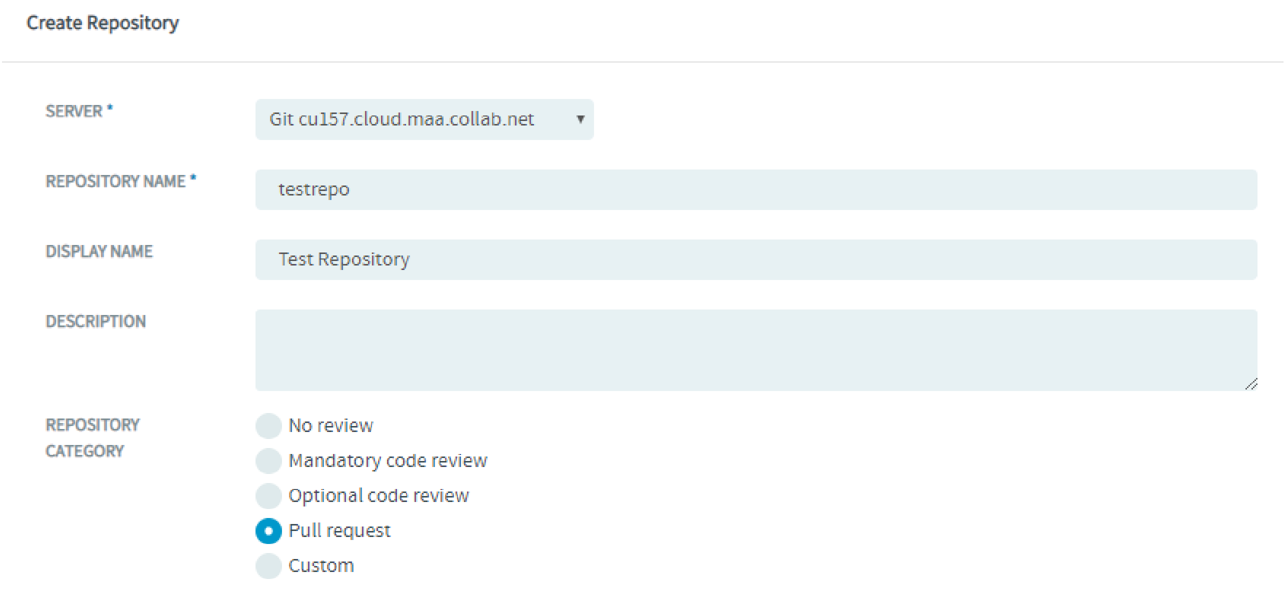
SERVER *	Git cu157.cloud.maa.collab.net ▾
REPOSITORY NAME *	testrepo
DISPLAY NAME	Test Repository
DESCRIPTION	
REPOSITORY CATEGORY	● No review ○ Mandatory code review ○ Optional code review ○ Pull request ○ Custom

Here’s a list of permissions and what users with these permissions can do:

- No access: Users with no permissions cannot do anything.
- View only: Users with read permissions can only read branches.
- Commit/View: Users with commit permissions can do everything read permissions would grant and in addition, push to/create any branch and push tags.
- Delete/View: Users with delete permissions can do everything commit permissions would grant and in addition, have the right to rewrite history, upload merges and forge identity.
- Source Code Admin: Users with admin permissions can do everything delete permissions would grant. In addition, they can forge the identity of the Gerrit server, and have the right to fine tune access rights in Git for the Git project involved.

Pull Request Reviews for Git Repositories

Pull requests allow developers to collaborate with each other on a code change before merging it into another branch on a Git repository. Using a pull request, you notify others about a feature or fix change that needs attention.



Create Repository

SERVER * : Git cu157.cloud.maa.collab.net ▾

REPOSITORY NAME * : testrepo

DISPLAY NAME : Test Repository

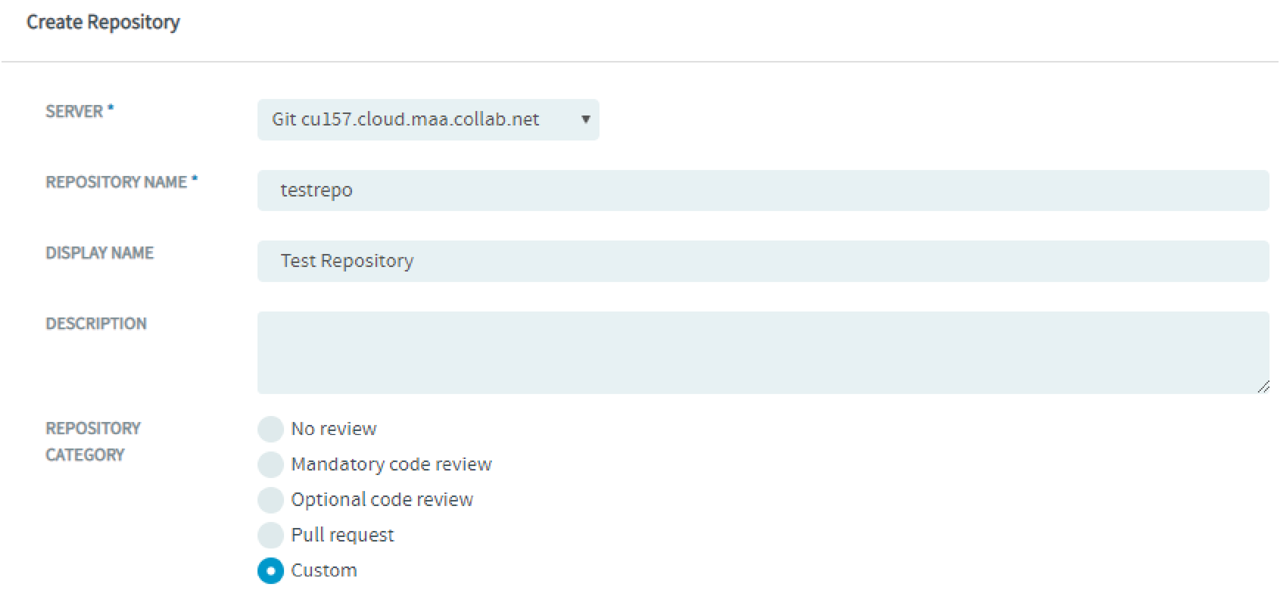
DESCRIPTION :

REPOSITORY CATEGORY :
- ○ No review
- ○ Mandatory code review
- ○ Optional code review
- ● Pull request
- ○ Custom

For more information about pull requests, see Pull Request.

Custom Code Review for Git Repositories

When a custom code review is specified, users with the TeamForge Source Code Admin permission can directly fine tune permissions (access rights) in gerrit’s web interface. Those changes will not be overridden by TeamForge.



Create Repository

SERVER * : Git cu157.cloud.maa.collab.net ▾

REPOSITORY NAME * : testrepo

DISPLAY NAME : Test Repository

DESCRIPTION :

REPOSITORY CATEGORY :
- ○ No review
- ○ Mandatory code review
- ○ Optional code review
- ○ Pull request
- ● Custom

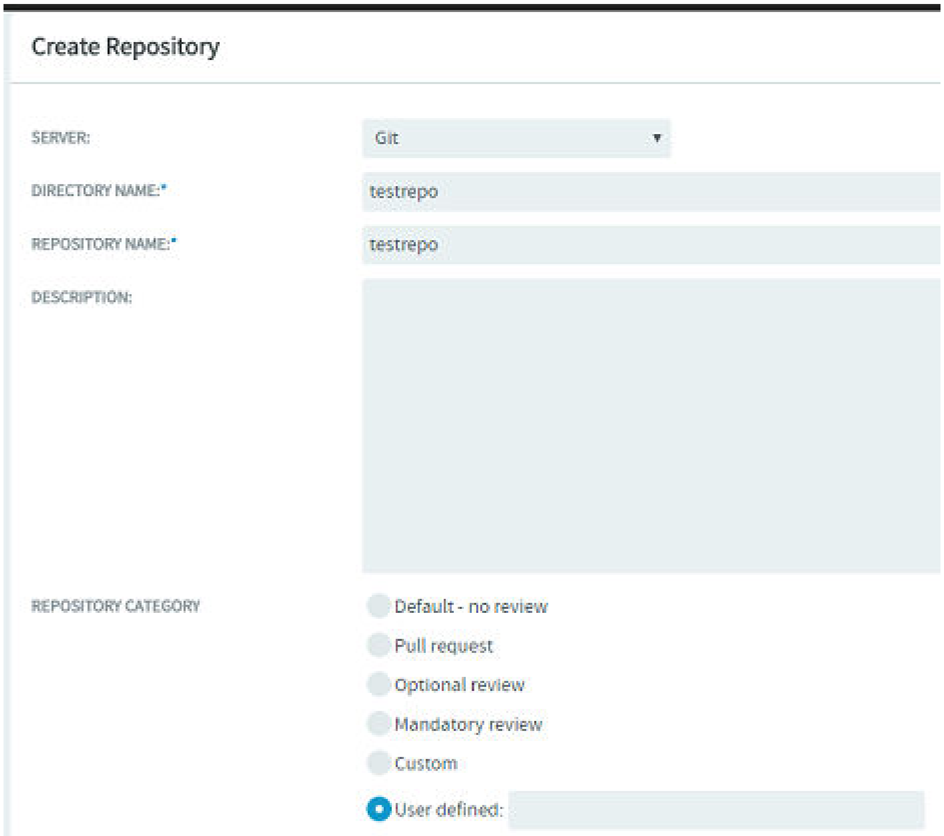
For information on manually defining access rights in the Gerrit web interface, see Update Git repository access permissions in Gerrit.

Here’s a list of permissions and what users with these permissions can do:

- No access: Users with no permissions cannot do anything.
- View only: Users with read permissions cannot do anything unless added in Gerrit.
- Commit/View: Users with commit permissions cannot do anything unless added in Gerrit.
- Delete/View: Users with delete permissions cannot do anything unless added in Gerrit.
- Source Code Admin: Users with admin permissions have the right to fine tune access rights in Gerrit for the Gerrit project involved.

User-defined Reviews for Git Repositories

Users can define their own code review policy.



Create Repository

SERVER: Git

DIRECTORY NAME:* testrepo

REPOSITORY NAME:* testrepo

DESCRIPTION:

REPOSITORY CATEGORY
○ Default - no review
○ Pull request
○ Optional review
○ Mandatory review
○ Custom
● User defined:

Create a User-defined Repository Category

You can add your own categories, if you have access to the `TeamForgeGerritMappings.xml` file.

To add a new user-defined repository category, follow these steps:

1. Create an empty Git repository, say `test-git-repo`.

   ```
   git init `test-git-repo`
   ```

2. Change to the directory `test-git-repo`.

   ```
   cd test-git-repo
   ```

3. Download the commits, files, and refs from the remote repository to your local repository.

   ```
   git fetch ssh://admin@<your_domain>:29418/TF-Projects refs/meta/config:meta
   -config
   ```

4. Check out the `TeamForgeGerritMappings.xml` file.

   ```
   git checkout meta-config
   ```

5. Open the `TeamForgeGerritMappings.xml` file in the editor.

   ```
   vim TeamForgeGerritMappings.xml
   ```

   Add a new repository category, say “pull_request_new” to it.

   ```
   <RepoCategory name="pull_request_new" keepRightsAddedInGerrit="false">
        <ScmAdmin>
            <GerritRead value="ALLOW" refPattern="refs/*" exclusive="false"↗
            <GerritCodeReview upperRange="2" lowerRange="-2" refPattern="refs
   /*" exclusive="false"↗
            <GerritVerify upperRange="1" lowerRange="-1" refPattern="refs/*"
   exclusive="false"↗
            <GerritSubmit value="ALLOW" refPattern="refs/*" exclusive="false"
   ↗
            <GerritPush forcePush="true" value="ALLOW" refPattern="refs/*" ex
   clusive="false"↗
            <GerritCreateReference value="ALLOW" refPattern="refs/*" exclusiv
   e="false"↗
   ```

```
         <GerritForgeAuthorIdentity value="ALLOW" refPattern="refs/*" excl
usive="false"/>
         <GerritForgeCommitterIdentity value="ALLOW" refPattern="refs/*" e
xclusive="false"/>
         <GerritForgeServerIdentity value="ALLOW" refPattern="refs/*" excl
usive="false"/>
         <GerritOwner value="ALLOW" refPattern="refs/*" exclusive="false"/>
         <GerritAbandon value="ALLOW" refPattern="refs/*" exclusive="false
"/>
         <GerritPushMerges value="ALLOW" refPattern="refs/for/refs/*" excl
usive="false"/>
         <GerritPush forcePush="false" value="ALLOW" refPattern="refs/for/
refs/*" exclusive="false"/>
         <GerritRebase value="ALLOW" refPattern="refs/*" exclusive="false"
/>
         <GerritPushAnnotatedTag forcePush="false" value="ALLOW" refPatter
n="refs/tags/*" exclusive="false"/>
         <GerritPushSignedTag value="ALLOW" refPattern="refs/tags/*" exclu
sive="false"/>
         <!-- protected branches-->
         <GerritPush forcePush="true" value="ALLOW" refPattern="refs/heads
/{RepoParams/@protectedBranches}" exclusive="true"/>
         <GerritSubmit value="ALLOW" refPattern="refs/for/refs/heads/{Repo
Params/@protectedBranches}" exclusive="true"/>
     </ScmAdmin>
     <ScmDeleteView>
         <GerritRead value="ALLOW" refPattern="refs/*" exclusive="false"/>
         <GerritCodeReview upperRange="2" lowerRange="-2" refPattern="refs
/*" exclusive="false"/>
         <GerritVerify upperRange="1" lowerRange="-1" refPattern="refs/*"
exclusive="false"/>
         <GerritSubmit value="ALLOW" refPattern="refs/*" exclusive="false"
/>
         <GerritPush forcePush="true" value="ALLOW" refPattern="refs/*" ex
clusive="false"/>
         <GerritCreateReference value="ALLOW" refPattern="refs/*" exclusiv
e="false"/>
         <GerritForgeAuthorIdentity value="ALLOW" refPattern="refs/*" excl
usive="false"/>
```

```
         <GerritForgeCommitterIdentity value="ALLOW" refPattern="refs/*" e
xclusive="false"/>
         <GerritPushMerges value="ALLOW" refPattern="refs/for/refs/*" excl
usive="false"/>
         <GerritPush forcePush="false" value="ALLOW" refPattern="refs/for/
refs/*" exclusive="false"/>
         <GerritRebase value="ALLOW" refPattern="refs/*" exclusive="false"
/>
         <GerritPushAnnotatedTag forcePush="false" value="ALLOW" refPatter
n="refs/tags/*" exclusive="false"/>
         <GerritPushSignedTag value="ALLOW" refPattern="refs/tags/*" exclu
sive="false"/>
         <!-- protected branches-->
         <GerritPush forcePush="false" value="DENY" refPattern="refs/heads
/{RepoParams/@protectedBranches}" exclusive="true"/>
         <GerritSubmit value="DENY" refPattern="refs/for/refs/heads/{RepoP
arams/@protectedBranches}" exclusive="true"/>
     </ScmDeleteView>
     <ScmCommitView>
         <GerritRead value="ALLOW" refPattern="refs/*" exclusive="false"/>
         <GerritCodeReview upperRange="2" lowerRange="-2" refPattern="refs
/*" exclusive="false"/>
         <GerritVerify upperRange="1" lowerRange="-1" refPattern="refs/*"
exclusive="false"/>
         <GerritSubmit value="ALLOW" refPattern="refs/*" exclusive="false"
/>
         <GerritPush forcePush="false" value="ALLOW" refPattern="refs/*" e
xclusive="false"/>
         <GerritCreateReference value="ALLOW" refPattern="refs/*" exclusiv
e="false"/>
         <GerritPush forcePush="false" value="ALLOW" refPattern="refs/for/
refs/*" exclusive="false"/>
         <GerritRebase value="ALLOW" refPattern="refs/*" exclusive="false"
/>
         <GerritPushAnnotatedTag forcePush="false" value="ALLOW" refPatter
n="refs/tags/*" exclusive="false"/>
         <GerritPushSignedTag value="ALLOW" refPattern="refs/tags/*" exclu
sive="false"/>
         <GerritPushMerges value="ALLOW" refPattern="refs/for/refs/*" excl
```

```
usive="false"/>
         <!-- protected branches-->
         <GerritPush forcePush="false" value="DENY" refPattern="refs/heads
/{RepoParams/@protectedBranches}" exclusive="true"/>
         <GerritSubmit value="DENY" refPattern="refs/for/refs/heads/{RepoP
arams/@protectedBranches}" exclusive="true"/>
     </ScmCommitView>
     <ScmViewOnly>
         <GerritRead value="ALLOW" refPattern="refs/*" exclusive="false"/>
         <GerritCodeReview upperRange="1" lowerRange="-1" refPattern="refs
/*" exclusive="false"/>
         <GerritPushMerges value="ALLOW" refPattern="refs/for/refs/*" excl
usive="false"/>
         <GerritPush forcePush="false" value="ALLOW" refPattern="refs/for/
refs/*" exclusive="false"/>
         <GerritRebase value="ALLOW" refPattern="refs/*" exclusive="false"
/>
     </ScmViewOnly>
 </RepoCategory>
```

6. Run this command to add the changes to your local directory.

```
git add TeamForgeGerritMappings.xml
```

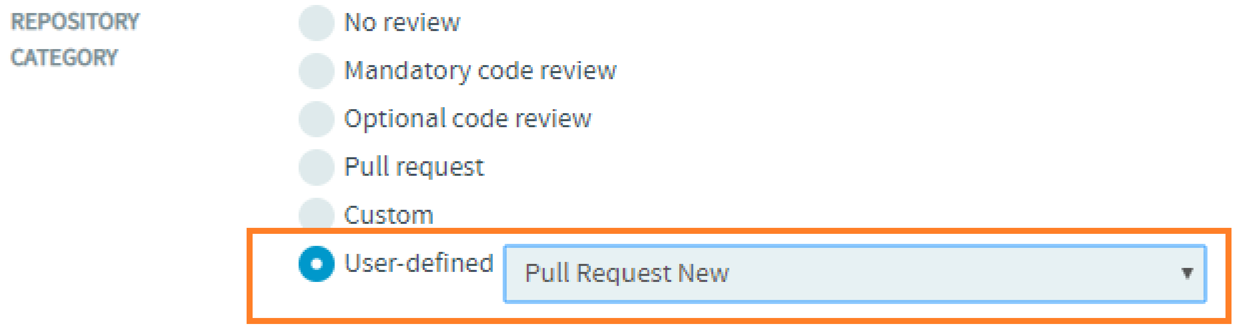
7. Commit the changes.

```
git commit -m "add user-defined repo type 'pull_request_new'"
```

8. Check-in the changes to your remote repository.

```
git push ssh://admin@<your-domain>:29418/TF-Projects meta-config:refs/meta/
config
```

Now the user-defined category `Pull Request New` is added successfully.

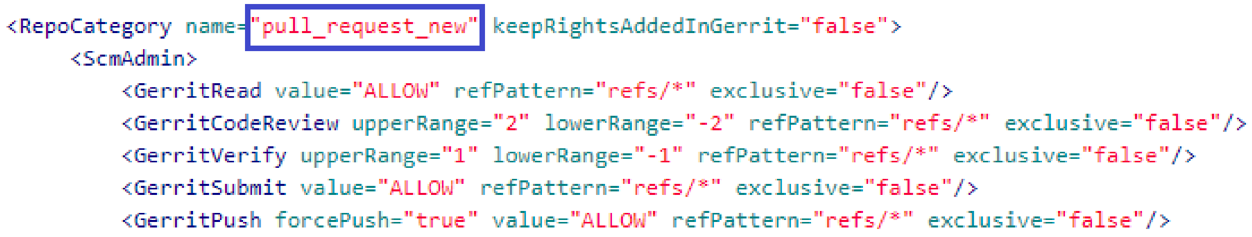


REPOSITORY CATEGORY

○ No review
○ Mandatory code review
○ Optional code review
○ Pull request
○ Custom
● User-defined Pull Request New ▾

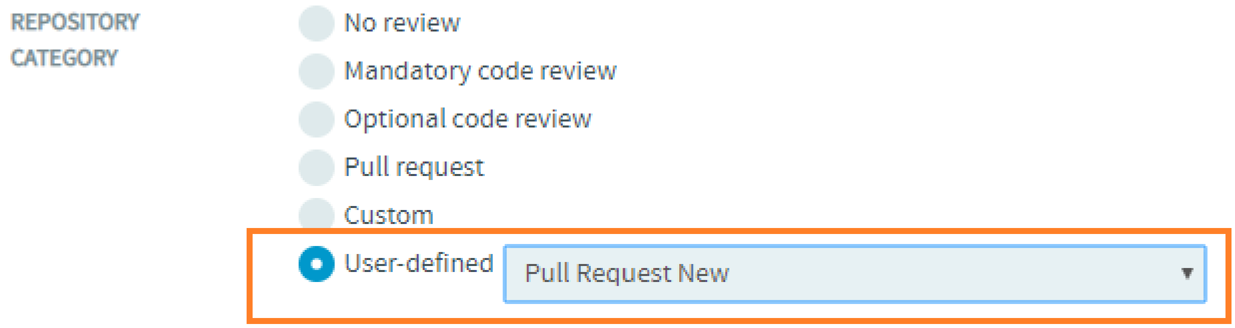
User-defined repository category "Pull Request New"

The `master` branch becomes the default protected branch for repositories that belong to the user-defined repository category, provided that its name is prefixed with “Pull Request”.



```
<RepoCategory name="pull_request_new" keepRightsAddedInGerrit="false">
     <ScmAdmin>
         <GerritRead value="ALLOW" refPattern="refs/*" exclusive="false"/>
         <GerritCodeReview upperRange="2" lowerRange="-2" refPattern="refs/*" exclusive="false"/>
         <GerritVerify upperRange="1" lowerRange="-1" refPattern="refs/*" exclusive="false"/>
         <GerritSubmit value="ALLOW" refPattern="refs/*" exclusive="false"/>
         <GerritPush forcePush="true" value="ALLOW" refPattern="refs/*" exclusive="false"/>
```

User-defined repository category "pull_request_new" in `TeamForgeGerritMappings.xml` file

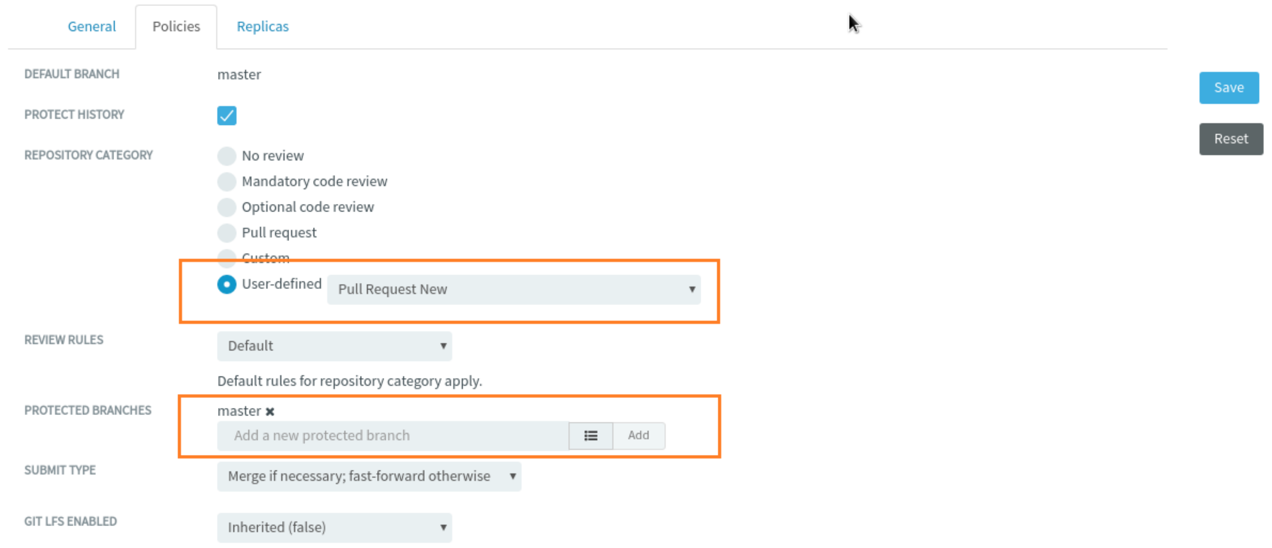


REPOSITORY CATEGORY

○ No review
○ Mandatory code review
○ Optional code review
○ Pull request
○ Custom
● User-defined Pull Request New ▾

User-defined repository category "pull_request_new" shown as "Pull Request New" in the UI

Once the repository is created, the `master` branch becomes a protected branch of the repository by default.



DEFAULT BRANCH master

PROTECT HISTORY ☑

REPOSITORY CATEGORY
○ No review
○ Mandatory code review
○ Optional code review
○ Pull request
○ Custom
● User-defined Pull Request New ▾

REVIEW RULES Default ▾

Default rules for repository category apply.

PROTECTED BRANCHES master ✖
Add a new protected branch

SUBMIT TYPE Merge if necessary; fast-forward otherwise ▾

GIT LFS ENABLED Inherited (false) ▾

"master" added as the default protected branch for user-defined repository category "Pull Request New"

Related Links

- Review Code

History protection archives rewritten changes and keeps backups of deleted branches. If history changes occur, an immutable backup `ref` is created in the remote repository, notification emails are sent to all members of the Gerrit Administrators group, and an event is logged in the audit log.

History rewrites are non-fast-forward updates of remote refs and associated objects. History rewrites happen when a branch in a remote repository gets deleted, previously pushed commits get amended or filtered and forcefully re-pushed, or a remote branch/tag is pointed to an entirely different commit history.

History may get rewritten without leaving any trace of the previous state. Sometimes this behavior may be wanted — for example, in the case of removing code violating intellectual property, removing mistakenly committed large binary files or removing merged feature branches. The TeamForge-Git integration therefore does not disable the history rewrite feature, but instead enables it for SCM Administrators alone. However, since rewriting history might be easily abused and result in accidental data loss, we’ve introduced the History Protection feature as a safety net and necessity for ensuring proper audit compliance.

History protection archives rewritten changes and keeps backups of deleted branches. If history changes occur, an immutable backup `ref` is created in the remote repository, notification emails are sent to all members of the Gerrit Administrators group, and an event is logged in the audit log. The backed up `ref` can be restored into a new branch with any Git client (without needing physical file access to the Gerrit server). Gerrit site administrators can still decide to remove selected backup refs permanently.

Enable History Protection

History protection is enabled at the site level by default in TeamForge 17.4 and later versions. However, site administrators can disable history protection at the site level if need be, after which project administrators can choose to have history protection enabled or disabled for individual repositories.

To turn on/off history protection for an individual Git repository in a TeamForge project, select or clear the **Protect History** check box respectively while creating the repository.

For an existing repository:

1. On the **Source Code** page, select the Git repository and click **Edit**.
2. Select the **Protect History** check box.
3. Click **Save**.

You can turn history protection on or off any time. However, your change will not be reflected in Gerrit immediately. It will be effective after the time that you defined as the regular refresh interval while installing the Git integration.

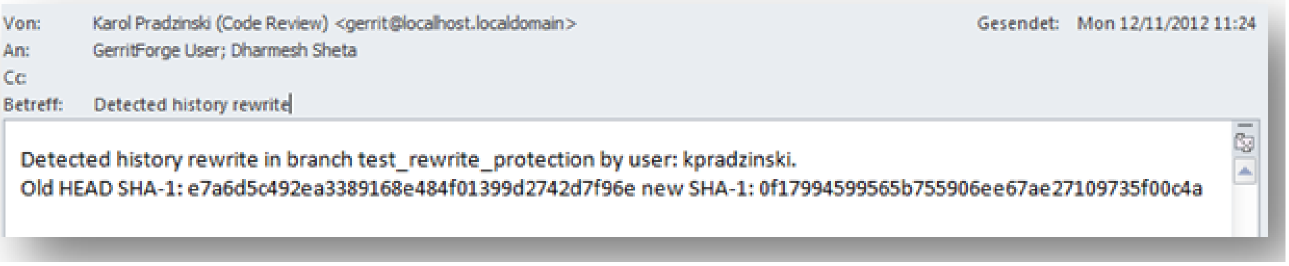
If you want your change to take effect immediately, do this right after you select or clear the **Protect History** check box: as a user with Source Code Admin permission, temporarily remove any user having a project role with any SCM permission, and then add that user back. This will trigger an immediate sync which will enable history protection. After that, the Gerrit Administrator will be able to see History Protection enabled in the Gerrit web interface (by logging in as a Gerrit Administrator and clicking the General link for the project with the name of the Git repository).

History Protection Reports

Once history protection is turned on, any non-fast-forward push to a remote repository or deletion of a branch or tag on a remote repository is recorded and reported.

Email Notifications

When history is rewritten, an email is sent to the Administrator group members in Gerrit.



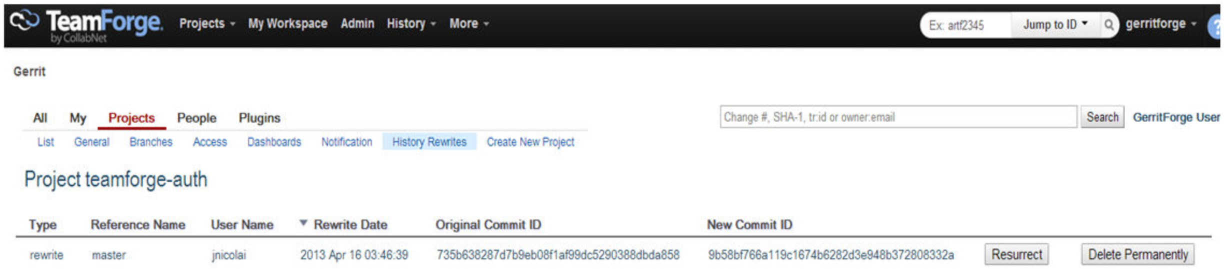
Von: Karol Pradzinski (Code Review) <gerrit@localhost.localdomain>
Gesendet: Mon 12/11/2012 11:24
An: GerritForge User; Dharmesh Sheta
Cc:
Betreff: Detected history rewrite

Detected history rewrite in branch test_rewrite_protection by user: kpradzinski.
Old HEAD SHA-1: e7a6d5c492ea3389168e484f01399d2742d7f96e new SHA-1: 0f17994599565b755906ee67ae27109735f00c4a

Gerrit Web Interface

Every history rewrite event is logged and stored in the Gerrit database and visible in the Gerrit web interface. As Gerrit Administrator, you can:

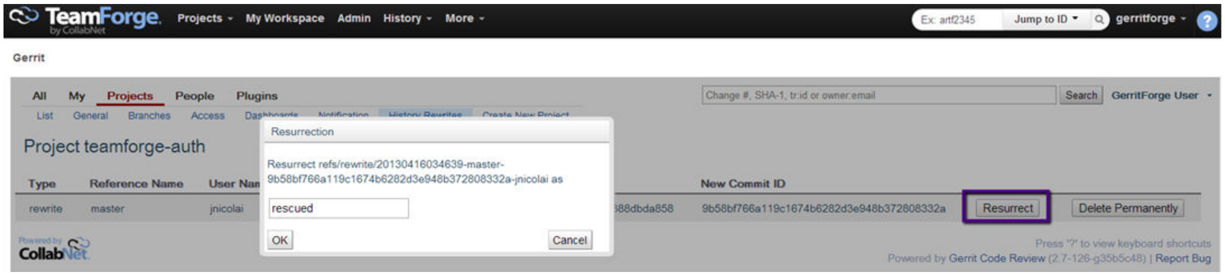
- See rewritten history from **Project > Rewritten History**.



Project teamforge-auth

Type	Reference Name	User Name	Rewrite Date	Original Commit ID	New Commit ID		
rewrite	master	jnicolai	2013 Apr 16 03:46:39	735b638287d7b9eb08f1af99dc5290388dbda858	9b58bf766a119c1674b6282d3e948b372808332a	Resurrect	Delete Permanently

- Restore history by clicking **Resurrect** and providing a name for the new branch.



Project teamforge-auth

Resurrection

Resurrect refs/rewrite/20130416034639-master-9b58bf766a119c1674b6282d3e948b372808332a-jnicolai as

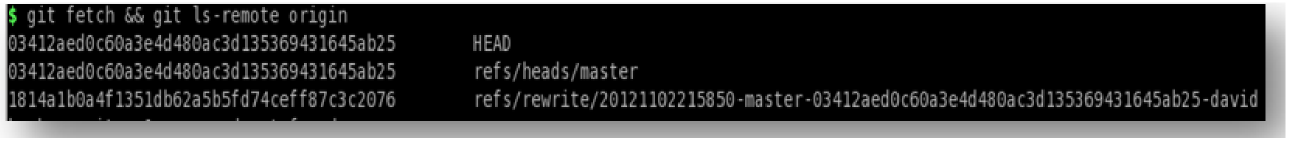
rescued

rewrite master jnicolai 388dbda858 9b58bf766a119c1674b6282d3e948b372808332a

- Permanently remove a branch by clicking **Delete Permanently**.

Git Command Line

You can use a standard Git client and run `git fetch && git ls-remote` for information on rewritten and deleted branches.



```
$ git fetch && git ls-remote origin
03412aed0c60a3e4d480ac3d135369431645ab25        HEAD
03412aed0c60a3e4d480ac3d135369431645ab25        refs/heads/master
1814a1b0a4f1351db62a5b5fd74ceff87c3c2076        refs/rewrite/20121102215850-master-03412aed0c60a3e4d480ac3d135369431645ab25-david
```

You can view entries in `refs/rewrite` (for non-fast-forward pushes) and `refs/delete` using the Git `ls-remote` command only if read access is granted to `refs/*`. Gerrit will prevent any other action such as delete/force-update on those special `refs` for all users including administrators.

Audit Log Entries

The following events are logged in `/opt/collabnet/gerrit/logs/gerrit.audit.log`:

- Remote branches are deleted.
- History is rewritten (non-fast-forward push).
- Backup branches are resurrected.
- Backup branches are permanently deleted.
- History Protection is turned on or off.

Appendix

- History Protection FAQs
- History Protection Slide Deck
- Git reflog vs History Protection
- Gerrit Performance Cheat Sheet

This topic discusses the mappings between TeamForge and Gerrit, Gerrit access rights, directory structure, connectivity, logs and configuration properties, and differences compared to vanilla Gerrit.

Git Integration Blog Posts

You can read the CollabNet blog posts on Git integration and follow the latest developments in the CollabNet TeamForge-Git integration space.

Here's a list of few useful blog posts:

- Bulletproof, Military Grade Security – Visualizing the Access Control Mechanisms of Your SCM Solution
- You shall not pass – Control your code quality gates with a wizard – Part I
- You shall not pass – Control your code quality gates with a wizard – Part II
- Migrating from Subversion to Git: What Your PCI-DSS Guy Will Not Tell You, Part I
- Migrating from Subversion to Git: What Your PCI-DSS Guy Will Not Tell You, Part II
- Seamlessly navigate between TeamForge projects and related Gerrit reviews
- TeamForge Git /Gerrit Integration with Jenkins CI
- CollabNet Gerrit Notifications – For all who miss the good ol' git push notifications
- TeamForge Just Got Even Better with Git Pull Request Feature!
- Gerrit Rebranding – The missing Guide to a customized Look & Feel
- Easy guide to mappings between Gerrit Access Control and TeamForge Source Code Permissions

Mappings Between TeamForge and Gerrit

These tables shows how objects and relationships are mapped between TeamForge and Gerrit.

TeamForge Object	Gerrit Object
TeamForge project	Project
SCM repository in TeamForge project (containing project roles with SCM permissions)	Project
Project Role	Group
User Group	Group
User	User
Site-wide role (TeamForge 8.0 and later)	Group

TeamForge Relationship	Gerrit Relationship
Git repository is part of a TeamForge project.	Gerrit project corresponding to the Git repository inherits from the Gerrit project corresponding to the TeamForge project (`TeamForge-Projects/<TeamForge project id>`).
TeamForge project <child> has a parent TeamForge project <parent>.	Gerrit project <child> inherits from the Gerrit project <parent>.
TeamForge project top is a top-level project.	Gerrit project <top> inherits from Gerrit project. TeamForge-Projects which in turn inherits from All-Projects.
User has a TeamForge Project Role.	User is part of the Group which corresponds to the TeamForge Project Role.
User is part of a User Group that is assigned a Project Role.	User is part of a Group (which corresponds to a TeamForge Project Role).
User is part of a User Group.	User is part of a Group (which corresponds to a TeamForge User Group.
Project Role is assigned an SCM permission (such as Admin, Delete and View, View and Commit, View Only, None).	Corresponding group is assigned Gerrit access rights matching the assigned TeamForge SCM permissions. Those access rights are determined by the code review policy of the corresponding TeamForge repository.
Site-wide role is assigned an SCM permission. (TeamForge 8.0 and later only).	Corresponding Gerrit groups are assigned Gerrit access rights matching the assigned TeamForge SCM permissions. Those access rights are determined by the code review policy of the TeamForge repository and hence may vary between repositories.
Guests, All Site Users, All Logged in Users, All Non-Restricted Users or Project Members have SCM permissions associated using TeamForge’s Default Access Permissions (TeamForge 8.0 and later only).	Corresponding Gerrit groups are assigned Gerrit access rights matching the assigned TeamForge SCM permissions. Those access rights are determined by the code review policy of the TeamForge repository and hence may vary between repositories.
User is a site admin in TeamForge.	User is part of Gerrit groups. TeamForge: Site Admins. TeamForge: Site-wide Project Admin Access. Private Project - Site-wide Admin Access. Public Project - Site-wide Admin Access. Gated Project - Site-wide Admin Access. Site admins have OWN and READ permissions for all Gerrit projects and the rights granted by the SCM Admin permission (depends on the code review policy of the Git repository in question).
User is a project admin in TeamForge.	User is part of Gerrit group. TeamForge: Project Admin for <TF project id>, which has OWN and READ permissions for all Git repositories of the corresponding TeamForge project.
User is non restricted in TeamForge (TeamForge 8.0 and later only).	User belongs to Gerrit group. TeamForge: Non-restricted Users.

TeamForge Relationship	Gerrit Relationship
User is a member of a TeamForge project (TeamForge 8.0 and later only).	User belongs to Gerrit group. TeamForge: Direct Project Member of <TF project id>.
User is member of a user group associated to a TeamForge project role (TeamForge 8.0 and later only).	User belongs to Gerrit group. TeamForge: Project Member of <TF project id>.
User has a site-wide role that has SCM permissions or a site-wide project admin permissions (TeamForge 8.0 and later only).	User is part of Gerrit group. TeamForge : Site-wide Role: <name of TeamForge Site-wide role> and - depending on the prevent inheritance to private projects flag, SCM permissions and project admin permissions - TeamForge - Site-wide Project Admin Access Public Project - Site-wide Admin Access Gated Project - Site-wide Admin Access Private Project - Site-wide Admin Access Public Project - Site-wide Delete Access Gated Project - Site-wide Delete Access Private Project - Site-wide Delete Access Public Project - Site-wide Commit Access Gated Project - Site-wide Commit Access Private Project - Site-wide Commit Access Public Project - Site-wide View Access Gated Project - Site-wide View Access Private Project - Site-wide View Access
User has a TeamForge account.	User belongs to the Gerrit group. Registered Users.
User is not logged into TeamForge yet.	User belongs to the Gerrit group. Anonymous Users. (as all logged in users do too).

Access Rights in Gerrit 2.13.x

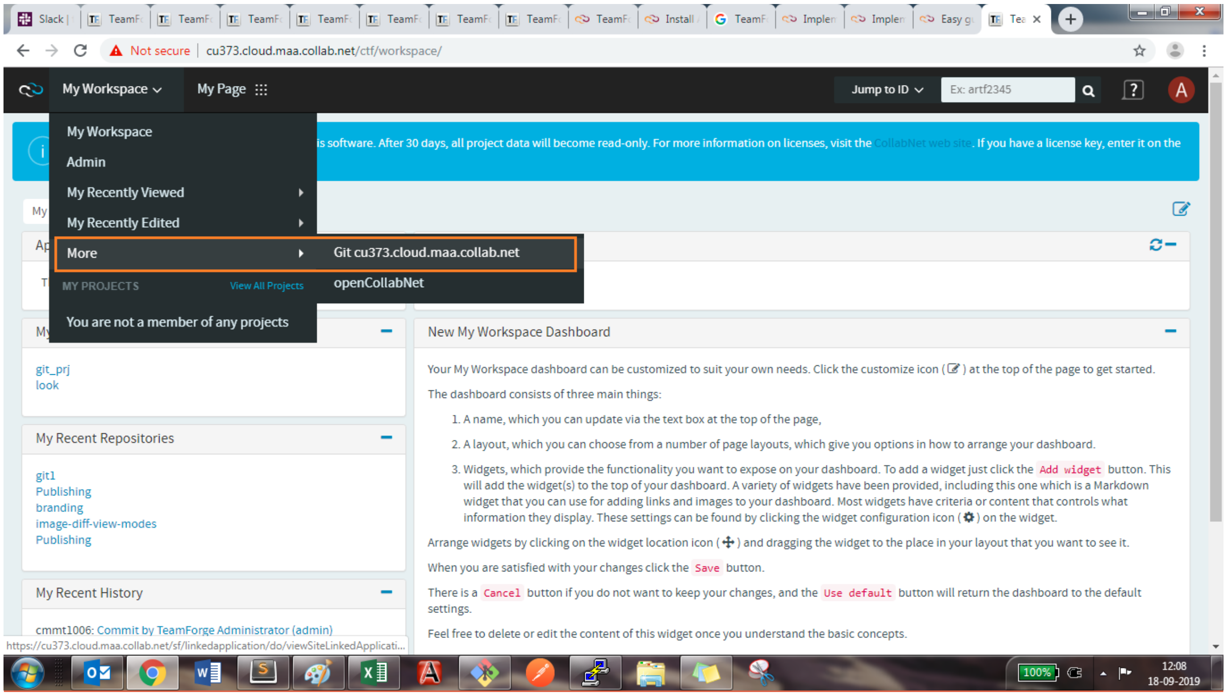
The Git integration maps Gerrit access rights to TeamForge Role Based Access Control (RBAC) permissions.

The mappings file `TeamForgeGerritMappings.xml` is located in the `refs/meta/config` branch of `TF-Projects` project.

How to view/access the `TeamForgeGerritMappings.xml` file?

1. Log on to TeamForge as a Site Administrator.
2. Select **My Workspace > More > Git <hostname>**.

> **NOTE:** `hostname` refers to the server where your Git integration is hosted.



is software. After 30 days, all project data will become read-only. For more information on licenses, visit the CollabNet web site. If you have a license key, enter it on the

New My Workspace Dashboard

Your My Workspace dashboard can be customized to suit your own needs. Click the customize icon ( ) at the top of the page to get started.

The dashboard consists of three main things:

1. A name, which you can update via the text box at the top of the page,
2. A layout, which you can choose from a number of page layouts, which give you options in how to arrange your dashboard.
3. Widgets, which provide the functionality you want to expose on your dashboard. To add a widget just click the `Add widget` button. This will add the widget(s) to the top of your dashboard. A variety of widgets have been provided, including this one which is a Markdown widget that you can use for adding links and images to your dashboard. Most widgets have criteria or content that controls what information they display. These settings can be found by clicking the widget configuration icon ( ) on the widget.

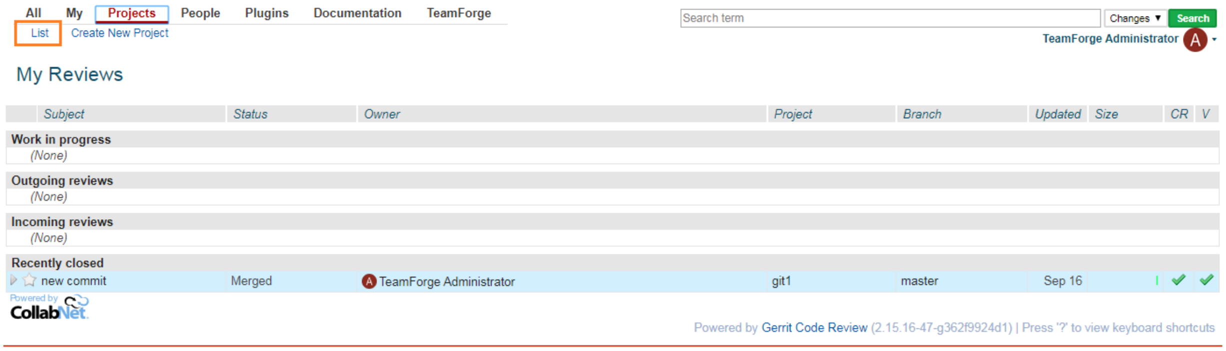
Arrange widgets by clicking on the widget location icon ( ) and dragging the widget to the place in your layout that you want to see it.

When you are satisfied with your changes click the `Save` button.

There is a `Cancel` button if you do not want to keep your changes, and the `Use default` button will return the dashboard to the default settings.

Feel free to delete or edit the content of this widget once you understand the basic concepts.

3. Select **Projects > List**.

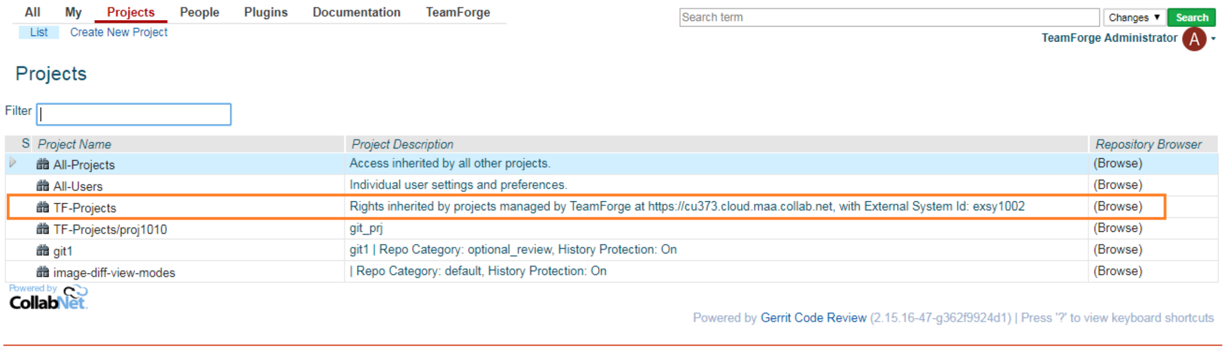


My Reviews

Subject	Status	Owner	Project	Branch	Updated	Size	CR	V
Work in progress								
(None)								
Outgoing reviews								
(None)								
Incoming reviews								
(None)								
Recently closed								
new commit	Merged	TeamForge Administrator	git1	master	Sep 16		✓	✓

Powered by Gerrit Code Review (2.15.16-47-g362f9924d1) | Press '?' to view keyboard shortcuts

4. Select **TF-Projects** from the list of projects.



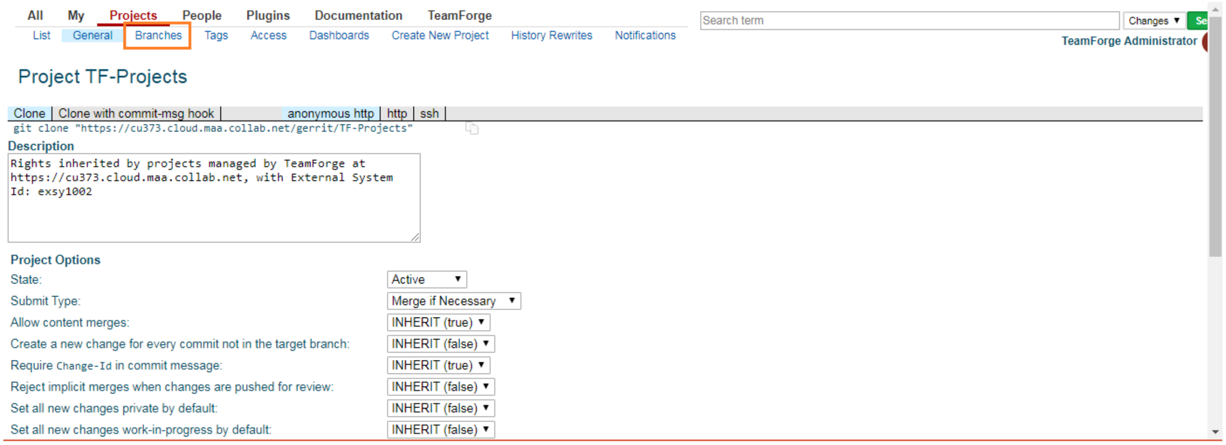
Projects

Filter

S	Project Name	Project Description	Repository Browser
	All-Projects	Access inherited by all other projects.	(Browse)
	All-Users	Individual user settings and preferences.	(Browse)
	TF-Projects	Rights inherited by projects managed by TeamForge at https://cu373.cloud.maa.collab.net, with External System Id: exsy1002	(Browse)
	TF-Projects/proj1010	git_prj	(Browse)
	git1	git1 | Repo Category: optional_review, History Protection: On	(Browse)
	image-diff-view-modes	| Repo Category: default, History Protection: On	(Browse)

Powered by Gerrit Code Review (2.15.16-47-g362f9924d1) | Press '?' to view keyboard shortcuts

5. Select the **Branches** tab.



Project TF-Projects

```
git clone "https://cu373.cloud.maa.collab.net/gerrit/TF-Projects"
```

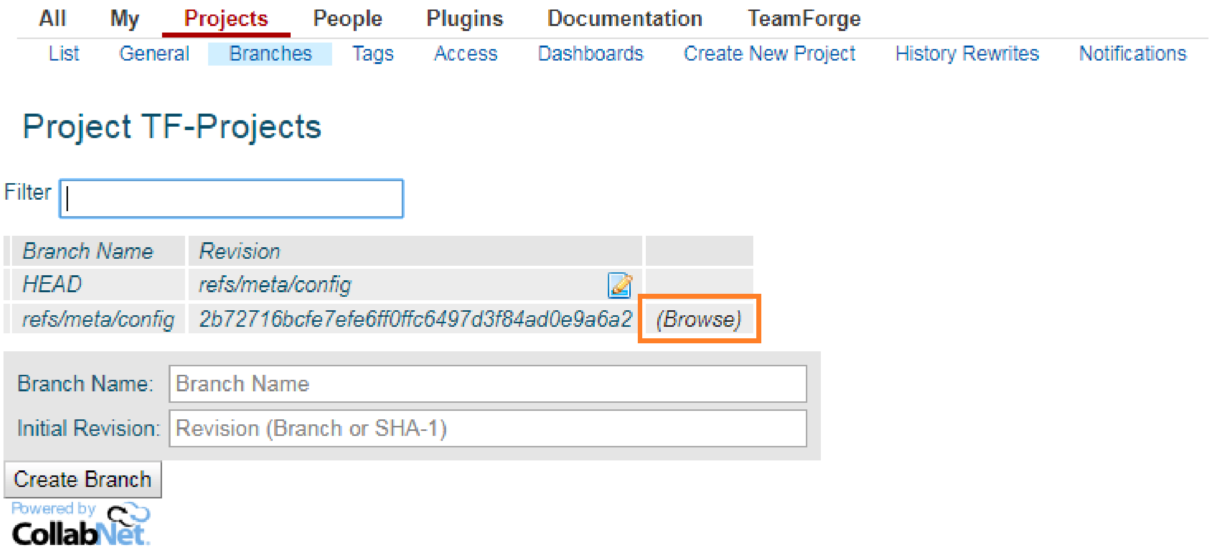
Description

```
Rights inherited by projects managed by TeamForge at
https://cu373.cloud.maa.collab.net, with External System
Id: exsy1002
```

Project Options

State:	Active ▾
Submit Type:	Merge if Necessary ▾
Allow content merges:	INHERIT (true) ▾
Create a new change for every commit not in the target branch:	INHERIT (false) ▾
Require `Change-Id` in commit message:	INHERIT (true) ▾
Reject implicit merges when changes are pushed for review:	INHERIT (false) ▾
Set all new changes private by default:	INHERIT (false) ▾
Set all new changes work-in-progress by default:	INHERIT (false) ▾

6. Click **Browse** against the `refs/meta/config` branch name.

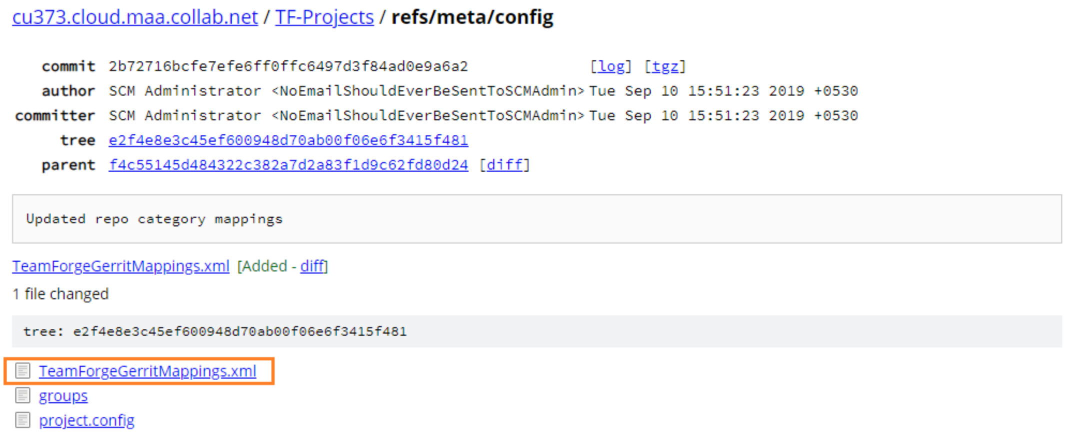


Project TF-Projects

Branch Name	Revision	
HEAD	refs/meta/config	
refs/meta/config	2b72716bcfe7efe6ff0ffc6497d3f84ad0e9a6a2	(Browse)

Branch Name:	Branch Name
Initial Revision:	Revision (Branch or SHA-1)

The `TeamForgeGerritMappings.xml` file can be found here.



cu373.cloud.maa.collab.net / TF-Projects / **refs/meta/config**

```
   commit 2b72716bcfe7efe6ff0ffc6497d3f84ad0e9a6a2            [log] [tgz]
   author SCM Administrator <NoEmailShouldEverBeSentToSCMAdmin> Tue Sep 10 15:51:23 2019 +0530
committer SCM Administrator <NoEmailShouldEverBeSentToSCMAdmin> Tue Sep 10 15:51:23 2019 +0530
     tree e2f4e8e3c45ef600948d70ab00f06e6f3415f481
   parent f4c55145d484322c382a7d2a83f1d9c62fd80d24 [diff]
```

```
Updated repo category mappings
```

TeamForgeGerritMappings.xml [Added - diff]

1 file changed

```
tree: e2f4e8e3c45ef600948d70ab00f06e6f3415f481
```

- TeamForgeGerritMappings.xml
- groups
- project.config

The following table shows how TeamForge RBAC permissions are now mapped to Gerrit access rights by default.

Code Review Policy	TeamForge Permission Cluster	Gerrit Access Right
Default	SCM None	-
	SCM View Only	Read
	SCM Commit/View	Read
		Push
		Create Reference
		Push Annotated Tag (refs/tags/*)
		Push Signed Tag (refs/tags/*)
	SCM Delete/View	Read
		Push (forcePush)
		Create Reference
		Forge Author Identity
		Forge Committer Identity
		Push Annotated Tag (refs/tags/*)
		Push Signed Tag (refs/tags/*)
	SCM Admin	Read
		Push (forcePush)
		Create Reference
		Forge Author Identity
		Forge Committer Identity
		Forge Server Identity
		Owner

		Abandon
		Push Annotated Tag (refs/tags/*)
		Push Signed Tag (refs/tags/*)
Optional Review	SCM None	-
	SCM View Only	Read
		View Drafts
		Publish Drafts
		Code Review -1,1
		Push (refs/for/refs/*)
		Rebase(refs/for/refs/*)
	SCM Commit/View	Read
		View Drafts
		Publish Drafts
		Code Review -2,2
		Verify -1,1
		Submit
		Push
		Create Reference
		Rebase (refs/for/refs/*)
		Push Annotated Tag(refs/tags/*)
		Push Signed Tag (refs/tags/*)
	SCM Delete/View	Read
		View Drafts
		Publish Drafts
		Code Review -2,2
		Verify -1,1
		Submit
		Push (forcePush)
		Create Reference
		Rebase (refs/for/refs/*)
		Create References
		Push Signed Tag (refs/tags/*)
		Push Annotated Tag (refs/tags/*)
		Push Merges(refs/for/refs/*)
		Forge Author Identity

		Forge Committer Identity
	SCM Admin	Read
		View Drafts
		Publish Drafts
		Delete Drafts
		Code Review -2,2
		Verify -1,1
		Submit
		Push (forcePush)
		Create Reference
		Owner
		Abandon
		Rebase (refs/for/refs/*)
		Create References
		Push Signed Tag (refs/tags/*)
		Push Annotated Tag (refs/tags/*)
		Push Merges(refs/for/refs/*)
		Forge Author Identity
		Forge Committer Identity
		Forge Server Identity
Mandatory Review	SCM None	-
	SCM View Only	Read
		View Drafts
		Publish Drafts
		Code Review -2,2
		Push (refs/for/refs/*)
		Rebase (refs/for/refs/*)
	SCM Commit/View	Read
		View Drafts
		Publish Drafts
		Code Review -2,2
		Verify -1,1
		Submit
		Push(refs/for/refs/*)
		Rebase (refs/for/refs/*)

	SCM Delete/View	Read
		View Drafts
		Publish Drafts
		Code Review -2,2
		Verify -1,1
		Submit
		Push(refs/for/refs/*)
		Rebase (refs/for/refs/*)
	SCM Admin	Read
		View Drafts
		Publish Drafts
		Delete Drafts
		Code Review -2,2
		Verify -1,1
		Submit
		Push (forcePush)
		Create Reference
		Owner
		Abandon
		Rebase (refs/for/refs/*)
		Push Annotated Tag(refs/tags/*)
		Push Signed Tag (refs/tags/*)
		Create References
		Push Merges(refs/for/refs/*)
		Forge Author Identity
		Forge Committer Identity
		Forge Server Identity

To make changes to the mappings, modify the `TeamForgeGerritMappings.xml` file in the `refs/meta/config` branch of `TF-Projects` project on the server where your Git integration is hosted. For instance, if you want to add a user-defined category to your repository, first you need to add the user-defined category to the `TeamForgeGerritMappings.xml` file. For instructions, see Create a User-defined Repository Category.

NOTE: Make sure that the resulting XML structure complies with this schema: https://forge.collab.net/gerrit/static/TeamForgeGerritMappings-8.0.0.xsd.

Gerrit Configuration Options

Gerrit provides many configuration options. In addition, CollabNet Gerrit plugins also have configuration options.

For more information on Gerrit's configuration options, see Gerrit Code Review - Configuration.

In addition, see Gerrit Performance Cheat Sheet to know more about tuning Gerrit for optimal performance.

CollabNet Gerrit plugins have these configuration options:

Section.teamforge

Options	Description
teamforge.cache-path	Location where Gerrit and CollabNet Gerrit plugin store caches. By default, this is at `/opt/collabnet/gerrit/cache`. We advise that it not be changed.
teamforge.cache-ttl	Time-to-live for Gerrit caches in seconds. The default value is 300.
teamforge.apiPort	Port over which TeamForge communicates with the Git integration. The default value is 9081.
teamforge.refreshTimeOut	Interval in seconds after which the Git integration synchronizes with TeamForge. The default value is 3600.
teamforge.jumboPushThreshold	The number of commits in one Git push beyond which the Git integration creates only a single commit object in TeamForge. The default value is 30.
teamforge.externalSystemId	ID of the TeamForge external integration system. The value of this property is set by the post-installation script when the Git integration is first installed.
teamforge.url	Host URL of the TeamForge site with which Git is integrated. The value of this property is set by the post-installation script when the Git integration is first installed.
teamforge.allowPushIfTeamForgeConnectionIsDown	TeamForge commit objects are validated prior to creation. When the value of this property is `false` and connection to TeamForge is down, validation fails. When the value of this property is `true`, validation and creation of commit objects are postponed until the connection to TeamForge is restored. The default value is `false`.
teamforge.parallelRemoteCallLimit	TeamForge is able to handle a certain number of parallel connections. This parameter was introduced in order to avoid TeamForge "is out of service" issues. The default value is 9.
teamforge.maxRemoteCallRetry	This parameter was introduced in order to specify the number of retry attempts for calls to TeamForge before connection failure is returned. The default value is 3.

teamforge.credentialsCache	When the value of this property is set to true, users' credentials are cached for the `teamforge.credentialsCacheTimeOut` amount of time and used to authorize actions in case of TeamForge connection outage. The default value is `true`.
teamforge.credentialsCacheTimeOut	Interval (in Seconds) after which the credentials cache expires. The default value is 3600.
teamforge.reconnectInterval	When the "TeamForge connection is down" state is detected, and the number of seconds exceeds the value of this parameter, attempts to restore connection are performed periodically. The default value is 30.
teamforge.repositoryroot	Location where all Git repositories are stored physically. The default value is set to the value of the Gerrit configuration property `gerrit.basePath`, which is set to `/gitroot` by default.
teamforge.maxFilesListedInTFCommitObject	Restricts the number of entries in the SCM files list view for a particular TeamForge commit object. This is especially useful for repository initial commit objects as they could contain a thousand entries that get processed by TeamForge. The default value is 250.
teamforge.notificationMaxSize	Number of bytes in notification message that will be sent out by git-multimail–part of the notification plugin. If message is larger than specified limit, it will be truncated. The default value is 25000.
teamforge.notificationMaxPythonExecutors	Number of Python processes used to create git-multimail notification. Each process will create one notification at a time. The default value is 2.
teamforge.syncTeamForgeProjectHierarchy	Turns the Project Hierarchy feature on. New Gerrit installs will have this value set to true, existing ones to false.
teamforge.supportSiteWideRoles	Enables TeamForgesite-wide role support. New Gerrit installs will have this value set to `true`, existing ones to `false`. This feature requires at least TeamForge 8.0 (will be ignored before).
teamforge.supportDefaultAccessPermissions	Enables TeamForge Default Access Permission support. New Gerrit installs will have this value set to `true`, existing ones to `false`. This feature requires at least TeamForge 8.0 (will be ignored before).
teamforge.commitProcessingTimeOut	Maximum time allocated to process each Git commit to create a TeamForge commit object. If processing takes longer, processing of this commit is canceled, no corresponding TeamForge commit object will be created and the next commit will be processed. The default time is 15 min.
teamforge.createTFProjectLinkedApps	If enabled creates Project linked application with target to Gerrit Dashboard for that TeamForge project given project contains at least one Git repository. This feature requires at

	least TeamForge 8.0 (will be ignored before). The default value is true.
teamforge.teamForgeMenuHeader	Specifies the name of the menu that contains the links back to TeamForge user's Workspace and repositories list for a given TeamForge project. The default value is TeamForge.
teamforge.ensureStreamEventsForRegisteredUsers	If set to true, the `RegisteredUsers` group will have the StreamEvents global capability assigned during Gerrit startup. The default value is true.
teamforge.ensureAdminRightsForSiteAdmins	If set to `true`, the `TF: Site Admins` group will have `Administrate Server` global capability assigned during Gerrit startup. The default value is `true`.

Replication Configuration

This feature requires TeamForge 8.1 or later. These options are ignored if you have TeamForge 8.0 or earlier.

Options	Description
teamforge.replicationMode	Sets the server mode (replication master or slave) of the Git integration server. This property is set by the TeamForge installer depending on the value specified in the site-options.conf file's GERRIT_REPLICATION_MODE token. Therefore, this property should not be edited manually within the gerrit.config file. The default value set by the TeamForge installer is master.

Replication Master Configuration

Options	Description
plugin.teamforge-replication.replicationDelay	The delay (in seconds) between a push to the source repository and the actual replication attempt to the replica server. If further push activities happen between this delay, those will be bundled into the same replication attempt, avoiding bursts of replication attempts in case of repository mass updates. The default value is 15s and should not be set below 3s.
plugin.teamforge-replication.threads	The number of threads that are used to push changes for each replica server. The default value is 4.
plugin.teamforge-replication.replicationRetry	The maximum wait time before the next replication attempt is performed (upon previous connection failure). It is increased progressively (after each failure per mirror) starting with 1m to the power of 2 and up to the parameter value. For example, if the value is 5m, replication will be reattempted (considering that connection failure still occurs) after 1m then after 2m then after 4m and then after 5m and further attempts will be performed at 5m intervals. The default is 5m.
plugin.teamforge-replication.sshConnectionTimeout	The timeout duration for establishing SSH connections during a replication attempt or when an SSH command is performed. This prevents the SSH queue from being blocked while waiting to connect to a mirror that is not responding. The default value is 15s.
plugin.teamforge-replication.sshCommandTimeout	The timeout duration for replication SSH command execution (for example, project creation, HEAD change, and so on), after which the command fails. This prevents the SSH

	queue from being blocked while waiting to connect to a mirror that is not responding. The default is 30s.
plugin.teamforge-replication.pushTimeout	The timeout duration for a replication push (push time after SSH connection is established), after which the push fails. This prevents the SSH queue from being blocked while waiting to connect to a mirror that is not responding. The default is 30s.

Replication Mirror Configuration

Options	Description
plugin.teamforge-slave.replicaId	The replica ID of the replication slave created in TeamForge if GERRIT_REPLICATION_MODE is set as slave. This property is set automatically by Gerrit upon start up and hence should not be edited manually.
plugin.teamforge-slave.allowGroup	The group or groups that are allowed to push directly to the replication mirror. By default, only Administrator groups can do this.

Log Files

From TeamForge 18.1, Gerrit's internal log rotation and compression feature is disabled as it is handled automatically by the TeamForge runtime environment.

Appendix

- History Protection FAQs
- History Protection Slide Deck
- Git reflog vs History Protection
- Gerrit Performance Cheat Sheet

Install Review Board on your site before you can make it available as an integrated application to project managers on your TeamForge site.

✓ You can install the Review Board application (`reviewboard`) on the TeamForge Application Server or on a seprate server of its own.

✓ Review Board database (`reviewboard-database`) can be installed on the TeamForge PostgreSQL Database Server on sites with database running on a separate server.

✓ To install Review Board successfully, ensure that other repositories such as EPEL (Extra Packages for Enterprise Linux) are disabled apart from the CollabNet and Operating System repositories.

✓ This procedure is for those who are installing the Review Board for the first time.

✓ In this scenario, both TeamForge and Review Board use PostgreSQL.

✓ TeamForge 18.1 supports Review Board 2.5.6.1 on RHEL/CentOS 6.9 and 7.4.

✓ Installing Review Board needs root privileges. You must log on as root or use a root shell to install Review Board.

> **IMPORTANT:** TeamForge has no support for having service-specific FQDN for Review Board.

Install Review Board on the TeamForge Application Server

In this setup, you install Review Board on the TeamForge Application Server (server-01) that already has TeamForge installed on it.

1. If you have TeamForge installed, you should have the TeamForge installation repository configured already.

   For more information, see: Configure the TeamForge installation repository

   ## TeamForge Installation Repository Configuration for Sites with Internet Access

   1. Contact the CollabNet Support and download the TeamForge 18.1 installation repository package to `/tmp`.

2. Install the repository package.
   `yum install -y /tmp/collabnet-teamforge-repo-18.1-0-noarch.rpm`
3. Refresh your repository cache.
   `yum clean all`

TeamForge Installation Repository Configuration for Sites without Internet Access

1. Contact the CollabNet Support to get the auxiliary installer package for TeamForge 18.1 disconnected installation and save it in `/tmp`.
   - RHEL/CentOS 7.4 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel7.x86_64.rpm`
   - In addition to the above CentOS 7.4 64 bit RPM package, you must get the following CentOS 7.4 compatibility RPM, which is required for TeamForge 18.1 disconnected media installation on CentOS 7.4 profile: `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm`.
2. Unpack the disconnected installation package.
   `rpm -ivh <package-name>`
3. Unpack the `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm` package if you are installing TeamForge 18.1 on CentOS 7.4.
   `rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm`

4. If not mounted already, mount the RHEL/CentOS installation DVD.

   The DVD contains the necessary software and utilities required for installing TeamForge without internet access. In the following commands, replace “cdrom” with the identifier for your server’s CD/DVD drive, if necessary.

   ```
   cd /media/
   mkdir cdrom
   mount /dev/cdrom ./cdrom/
   ```

   If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

5. Create a yum configuration file that points to the RHEL/CentOS installation DVD.
   `vi /etc/yum.repos.d/cdrom.repo`

   Here’s a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

6. Verify your yum configuration files.

   ```
   yum list httpd
   yum list apr
   ```

2. Install Review Board.

   ```
   yum install teamforge
   ```

3. Make sure that `reviewboard`, `reviewboard-database` and `reviewboard-adapter` identifiers have been added to the `SERVICES` token of the TeamForge Application Server.

   ```
   server-01:SERVICES=ctfcore ctfcore-database mail search codesearch etl ctf
   core-datamart subversion cvs gerrit gerrit-database binary binary-database
    reviewboard reviewboard-database reviewboard-adapter cliserver
   ```

4. Do this on sites without internet access.
   1. Contact the CollabNet Support and get the `python-modules-sources.zip` file.
   2. Unzip the `python-modules-sources.zip` file to `/opt/collabnet/teamforge/service/reviewboard/resources/SOURCES/python-modules-sources`.

      ```
      unzip python-modules-sources.zip -d /opt/collabnet/teamforge/service/r
      eviewboard/resources/SOURCES/python-modules-sources
      ```

5. Provision services.

   ```
   teamforge provision
   ```

6. If SCM is installed on a separate box, run the following script to authenticate a scmviewer user against a TeamForge Subversion repository for creating a new review request.

   ```
   /opt/collabnet/teamforge/runtime/scripts/svn-auth.py --repo-path=https://<
   scm_domain>/svn/repos/<repo_dir_name>
   ```

   You should now have a Review Board instance ready to work with TeamForge.

Install Review Board with Database on a Separate Server

You can install the Review Borad database on the TeamForge Database Server on sites with a dedicated Database Server. In this setup, you install TeamForge and Review Board on a two-server distributed setup with database services running on a separate server.

Install Review Board services on the TeamForge Application Server (server-01)

1. If you have TeamForge installed, you should have the TeamForge installation repository configured already.

   For more information, see: Configure the TeamForge installation repository

   ## TeamForge Installation Repository Configuration for Sites with Internet Access

   1. Contact the CollabNet Support and download the TeamForge 18.1 installation repository package to `/tmp`.
   2. Install the repository package.
      ```
      yum install -y /tmp/collabnet-teamforge-repo-18.1-0-noarch.rpm
      ```
   3. Refresh your repository cache.
      ```
      yum clean all
      ```

   ## TeamForge Installation Repository Configuration for Sites without Internet Access

   1. Contact the CollabNet Support to get the auxiliary installer package for TeamForge 18.1 disconnected installation and save it in `/tmp`.
      - RHEL/CentOS 7.4 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel7.x86_64.rpm`
      - In addition to the above CentOS 7.4 64 bit RPM package, you must get the following CentOS 7.4 compatibility RPM, which is required for TeamForge 18.1 disconnected media installation on CentOS 7.4 profile: `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm`.
   2. Unpack the disconnected installation package.
      ```
      rpm -ivh <package-name>
      ```
   3. Unpack the `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm` package if you are installing TeamForge 18.1 on CentOS 7.4.
      ```
      rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
      ```

4. If not mounted already, mount the RHEL/CentOS installation DVD.

   The DVD contains the necessary software and utilities required for installing TeamForge without internet access. In the following commands, replace “cdrom” with the identifier for your server’s CD/DVD drive, if necessary.

   ```
   cd /media/
   mkdir cdrom
   mount /dev/cdrom ./cdrom/
   ```

   If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

5. Create a yum configuration file that points to the RHEL/CentOS installation DVD.

   ```
   vi /etc/yum.repos.d/cdrom.repo
   ```

   Here’s a sample yum configuration file.

   ```
   [RHEL-CDROM]
   name=RHEL CDRom
   baseurl=file:///media/cdrom/Server/
   gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
   enabled=1
   gpgcheck=0
   ```

6. Verify your yum configuration files.

   ```
   yum list httpd
   yum list apr
   ```

2. Install Review Board.

   ```
   yum install teamforge
   ```

3. Make sure that `reviewboard`, `reviewboard-database` and `reviewboard-adapter` identifiers have been added to the SERVICES token as required.

   ```
   server-01:SERVICES = ctfcore mail search codesearch cliserver etl subversi
   on cvs gerrit binary binary-database reviewboard reviewboard-adapter
   server-02:SERVICES = ctfcore-database ctfcore-datamart gerrit-database rev
   iewboard-database
   ```

4. Do this on sites without internet access.
   1. Contact the CollabNet Support and get the `python-modules-sources.zip` file.
   2. Unzip the `python-modules-sources.zip` file to `/opt/collabnet/teamforge/service/reviewboard/resources/SOURCES/python-modules-sources`.

```
unzip python-modules-sources.zip -d /opt/collabnet/teamforge/service/reviewboard/resources/SOURCES/python-modules-sources
```

5. Provision services.
   ```
   teamforge provision
   ```

Provision the Database Server (server-02) with reviewboard-database Added to It

1. If you have TeamForge installed, you should have the TeamForge installation repository configured already.

   For more information, see: Configure the TeamForge installation repository

TeamForge Installation Repository Configuration for Sites with Internet Access

1. Contact the CollabNet Support and download the TeamForge 18.1 installation repository package to `/tmp`.
2. Install the repository package.
   ```
   yum install -y /tmp/collabnet-teamforge-repo-18.1-0-noarch.rpm
   ```
3. Refresh your repository cache.
   ```
   yum clean all
   ```

TeamForge Installation Repository Configuration for Sites without Internet Access

1. Contact the CollabNet Support to get the auxiliary installer package for TeamForge 18.1 disconnected installation and save it in `/tmp`.
   - RHEL/CentOS 7.4 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel7.x86_64.rpm`
   - In addition to the above CentOS 7.4 64 bit RPM package, you must get the following CentOS 7.4 compatibility RPM, which is required for TeamForge 18.1 disconnected media installation on CentOS 7.4 profile: `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm`.
2. Unpack the disconnected installation package.

```
    rpm -ivh <package-name>
    ```

  3. Unpack the `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm` package if you are installing TeamForge 18.1 on CentOS 7.4.

     ```
     rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
     ```

  4. If not mounted already, mount the RHEL/CentOS installation DVD.

     The DVD contains the necessary software and utilities required for installing TeamForge without internet access. In the following commands, replace “cdrom” with the identifier for your server’s CD/DVD drive, if necessary.

     ```
     cd /media/
     mkdir cdrom
     mount /dev/cdrom ./cdrom/
     ```

     If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

  5. Create a yum configuration file that points to the RHEL/CentOS installation DVD.

     ```
     vi /etc/yum.repos.d/cdrom.repo
     ```

     Here’s a sample yum configuration file.

     ```
     [RHEL-CDROM]
     name=RHEL CDRom
     baseurl=file:///media/cdrom/Server/
     gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
     enabled=1
     gpgcheck=0
     ```

  6. Verify your yum configuration files.

     ```
     yum list httpd
     yum list apr
     ```

2. Install Review Board.

   ```
   yum install teamforge
   ```

3. Make sure that `reviewboard`, `reviewboard-database` and `reviewboard-adapter` identifiers have been added to the SERVICES token as required.

   ```
   server-01:SERVICES = ctfcore mail search codesearch cliserver etl subversion cvs gerrit binary binary-database reviewboard reviewboard-adapter cliserver
   ```

```
server-02:SERVICES = ctfcore-database ctfcore-datamart gerrit-database rev
iewboard-database
```

4. Provision services.
   ```
   teamforge provision
   ```

Do This on the TeamForge Application Server (server-01)

If SCM is installed on a separate box, run the following script to authenticate a scmviewer user against a TeamForge Subversion repository for creating a new review request.

```
/opt/collabnet/teamforge/runtime/scripts/svn-auth.py --repo-path=https://<scm_
domain>/svn/repos/<repo_dir_name>
```

You should now have a Review Board instance ready to work with TeamForge.

Install Review Board on a Separate Server

In this setup, you install TeamForge and Review Board on a two-server distributed setup with Review Board services running on a separate server.

Provision the TeamForge Application Server (server-01) with reviewboard-adapter Added to It

1. If you have TeamForge installed, you should have the TeamForge installation repository configured already.

   For more information, see: Configure the TeamForge installation repository

   ### TeamForge Installation Repository Configuration for Sites with Internet Access

   1. Contact the CollabNet Support and download the TeamForge 18.1 installation repository package to `/tmp`.
   2. Install the repository package.
      ```
      yum install -y /tmp/collabnet-teamforge-repo-18.1-0-noarch.rpm
      ```
   3. Refresh your repository cache.
      ```
      yum clean all
      ```

TeamForge Installation Repository Configuration for Sites without Internet Access

1. Contact the CollabNet Support to get the auxiliary installer package for TeamForge 18.1 disconnected installation and save it in `/tmp`.
   - RHEL/CentOS 7.4 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel7.x86_64.rpm`
   - In addition to the above CentOS 7.4 64 bit RPM package, you must get the following CentOS 7.4 compatibility RPM, which is required for TeamForge 18.1 disconnected media installation on CentOS 7.4 profile: `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm`.
2. Unpack the disconnected installation package.
   ```
   rpm -ivh <package-name>
   ```
3. Unpack the `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm` package if you are installing TeamForge 18.1 on CentOS 7.4.
   ```
   rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
   ```
4. If not mounted already, mount the RHEL/CentOS installation DVD.

   The DVD contains the necessary software and utilities required for installing TeamForge without internet access. In the following commands, replace “cdrom” with the identifier for your server’s CD/DVD drive, if necessary.

   ```
   cd /media/
   mkdir cdrom
   mount /dev/cdrom ./cdrom/
   ```

   If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

5. Create a yum configuration file that points to the RHEL/CentOS installation DVD.
   ```
   vi /etc/yum.repos.d/cdrom.repo
   ```

   Here’s a sample yum configuration file.

   ```
   [RHEL-CDROM]
   name=RHEL CDRom
   baseurl=file:///media/cdrom/Server/
   gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
   ```

```
    enabled=1
    gpgcheck=0
    ```
    6. Verify your yum configuration files.
    ```
    yum list httpd
    yum list apr
    ```
2. Install Review Board.
   ```
   yum install teamforge
   ```
3. Make sure that `reviewboard`, `reviewboard-database` and `reviewboard-adapter` identifiers have been added to the SERVICES token as required.
   ```
   server-01:SERVICES = ctfcore ctfcore-database ctfcore-datamart gerrit-database mail search codesearch cliserver etl subversion cvs gerrit binary binary-database reviewboard-adapter cliserver
   server-02:SERVICES = reviewboard reviewboard-database
   ```
4. Provision services.
   ```
   teamforge provision
   ```

Install Review Board Services on the Review Board Server (server-02)

1. If you have TeamForge installed, you should have the TeamForge installation repository configured already.

   For more information, see: Configure the TeamForge installation repository

   ### TeamForge Installation Repository Configuration for Sites with Internet Access

   1. Contact the CollabNet Support and download the TeamForge 18.1 installation repository package to `/tmp`.
   2. Install the repository package.
      ```
      yum install -y /tmp/collabnet-teamforge-repo-18.1-0-noarch.rpm
      ```
   3. Refresh your repository cache.
      ```
      yum clean all
      ```

TeamForge Installation Repository Configuration for Sites without Internet Access

1. Contact the CollabNet Support to get the auxiliary installer package for TeamForge 18.1 disconnected installation and save it in `/tmp`.
   - RHEL/CentOS 7.4 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel7.x86_64.rpm`
   - In addition to the above CentOS 7.4 64 bit RPM package, you must get the following CentOS 7.4 compatibility RPM, which is required for TeamForge 18.1 disconnected media installation on CentOS 7.4 profile: `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm`.
2. Unpack the disconnected installation package.
   ```
   rpm -ivh <package-name>
   ```
3. Unpack the `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm` package if you are installing TeamForge 18.1 on CentOS 7.4.
   ```
   rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
   ```
4. If not mounted already, mount the RHEL/CentOS installation DVD.

   The DVD contains the necessary software and utilities required for installing TeamForge without internet access. In the following commands, replace “cdrom” with the identifier for your server’s CD/DVD drive, if necessary.

   ```
   cd /media/
   mkdir cdrom
   mount /dev/cdrom ./cdrom/
   ```

   If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

5. Create a yum configuration file that points to the RHEL/CentOS installation DVD.
   ```
   vi /etc/yum.repos.d/cdrom.repo
   ```

   Here’s a sample yum configuration file.

   ```
   [RHEL-CDROM]
   name=RHEL CDRom
   baseurl=file:///media/cdrom/Server/
   gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
   ```

```
enabled=1
gpgcheck=0
```

   6. Verify your yum configuration files.

   ```
   yum list httpd
   yum list apr
   ```

2. Install Review Board.

   ```
   yum install teamforge
   ```

3. Make sure that `reviewboard`, `reviewboard-database` and `reviewboard-adapter` identifiers have been added to the SERVICES token as required.

   ```
   server-01:SERVICES = ctfcore ctfcore-database ctfcore-datamart gerrit-database mail search codesearch cliserver etl subversion cvs gerrit binary binary-database reviewboard-adapter cliserver
   server-02:SERVICES = reviewboard reviewboard-database
   ```

4. Do this on sites without internet access.
   1. Contact the CollabNet Support and get the `python-modules-sources.zip` file.
   2. Unzip the `python-modules-sources.zip` file to `/opt/collabnet/teamforge/service/reviewboard/resources/SOURCES/python-modules-sources.`

      ```
      unzip python-modules-sources.zip -d /opt/collabnet/teamforge/service/reviewboard/resources/SOURCES/python-modules-sources
      ```

5. Provision services.

   ```
   teamforge provision
   ```

6. Reinitialize TeamForge on the Review Board Server.

   ```
   teamforge reinitialize
   ```

7. If SCM is installed on a separate box, run the following script to authenticate a scmviewer user against a TeamForge Subversion repository for creating a new review request.

   ```
   /opt/collabnet/teamforge/runtime/scripts/svn-auth.py --repo-path=https://<scm_domain>/svn/repos/<repo_dir_name>
   ```

   You should now have a Review Board instance ready to work with TeamForge.

Post Install Tasks

- Add Review Board to Projects
- Users are not getting email notifications for review requests and reviews. What should I do?

Bootstrap Review Board Post Install or Upgrade

Use the following instructions if you want to bootstrap Review Board (drop Review Board database tables and recreate them again) for some reason post installation or upgrade.

1. Log on to the server that hosts the Review Board.
2. Select **My Workspace > Admin**.
3. Select **Projects > Integrated Apps**.
4. Select **Review Board** and click **Delete**.
5. Stop TeamForge.
   ```
   teamforge stop
   ```
6. Start the TeamForge database services.
   ```
   teamforge start -s postgres
   ```
7. Bootstrap the Review Board database.
   ```
   teamforge bootstrap -s reviewboard-database-postgres
   ```
8. Bootstrap the Review Board.
   ```
   teamforge bootstrap -s reviewboard
   ```
9. Start TeamForge.
   ```
   teamforge start
   ```

Use these instructions to upgrade Review Board to a latest build.
Before You Begin

- TeamForge 18.1 supports Review Board 2.5.6.1 on RHEL/CentOS 6.9 and 7.4.
- This procedure is for those who have Review Board already and are upgrading Review Board to a latest build on RHEL/CentOS 6.9 or 7.4.
- You may choose to upgrade Review Board on the same server or on a new server.
- In this scenario, both TeamForge and Review Board use PostgreSQL.
- To install Review Board successfully, ensure that other repositories such as EPEL (Extra Packages for Enterprise Linux) are disabled apart from the CollabNet and Operating System repositories.
- Upgrading Review Board needs root privileges. You must log on as root or use a root shell to upgrade Review Board.

Back up the Review Board Data Directory

The default Review Board data directory has been changed from `/opt/collabnet/reviewboard/data` to `/opt/collabnet/teamforge/var/reviewboard/data` in TeamForge 17.4.

Are You Upgrading from TeamForge 17.1 or Earlier to TeamForge 17.4 or Later?

If you are upgrading from TeamForge 17.1 or earlier to TeamForge 17.4 or later, regardless of whether you upgrade Review Board on the same or new hardware, you must back up your Review Board data directory from `/opt/collabnet/reviewboard/data` and restore it to `/opt/collabnet/teamforge/var/reviewboard/data`.

> **TIP:** The Review Board database is backed up already when you have upgraded TeamForge. So, it is not necessary to take a back up of the Review Board database again.

1. Back up the Review Board data directory.
   ```
   cd /opt/collabnet
   tar -zcvf /tmp/reviewboard_data.tgz reviewboard
   ```
2. Copy the `/tmp/reviewboard_data.tgz` file to the `/tmp` directory of the new server if you are upgrading Review Board on a new hardware.

Are You Upgrading from TeamForge 17.4 to TeamForge 17.8 or Later?

1. Back up the Review Board data directory.
   ```
   cd /opt/collabnet/teamforge/var
   tar -zcvf /tmp/reviewboard_data.tgz reviewboard
   ```
   > **TIP:** If you are upgrading from TeamForge 17.4 (or later), the `/opt/collabnet/teamforge/var` directory would have been backed up already as part of your TeamForge upgrade process, in which case you can skip backing up the `/opt/collabnet/teamforge/var` directory again.
2. Copy the `/tmp/reviewboard_data.tgz` file to the `/tmp` directory of the new server.

Upgrade Review Board

> **NOTE:** TeamForge 18.1 has no support for having service-specific FQDN for Review Board.

1. Make sure that `reviewboard`, `reviewboard-database` and `reviewboard-adapter` identifiers have been added to the `SERVICES` token of the TeamForge Application Server (server-01).
   ```
   server-01:SERVICES=ctfcore ctfcore-database mail search codesearch etl ctf
   core-datamart subversion cvs gerrit gerrit-database binary binary-database
    reviewboard reviewboard-database reviewboard-adapter cliserver
   ```
   It is assumed that the Review Board is running on the TeamForge Application Server. In case you have a separate Review Board Server, add the `reviewboard` and `reviewboard-database` identifiers to the Review Board server's `SERVICES` token.
2. Configure the TeamForge installation repository.

TeamForge Installation Repository Configuration for Sites with Internet Access

1. Contact the CollabNet Support and download the TeamForge 18.1 installation repository package to `/tmp`.
2. Install the repository package.
   ```
   yum install -y /tmp/collabnet-teamforge-repo-18.1-0-noarch.rpm
   ```
3. Refresh your repository cache.
   ```
   yum clean all
   ```

TeamForge Installation Repository Configuration for Sites without Internet Access

1. Contact the CollabNet Support to get the auxiliary installer package for TeamForge 18.1 disconnected installation and save it in `/tmp`.
   - RHEL/CentOS 6.9 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel6.x86_64.rpm`
   - RHEL/CentOS 7.4 64 bit: `CTF-Disconnected-media-18.1.446-1261.rhel7.x86_64.rpm`
   - In addition to the above CentOS 7.4 64 bit RPM package, you must get the following CentOS 7.4 compatibility RPM, which is required for TeamForge 18.1 disconnected media installation on CentOS 7.4 profile: `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm`.
2. Unpack the disconnected installation package.
   ```
   rpm -Uvh <package-name>
   ```

3. Unpack the `compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm` package if you are installing TeamForge 18.1 on CentOS 7.4.
   ```
   rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
   ```
4. If not mounted already, mount the RHEL/CentOS installation DVD.

   The DVD contains the necessary software and utilities required for installing TeamForge without internet access. In the following commands, replace “cdrom” with the identifier for your server’s CD/DVD drive, if necessary.

   ```
   cd /media/
   mkdir cdrom
   mount /dev/cdrom ./cdrom/
   ```

   If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

5. Create a yum configuration file that points to the RHEL/CentOS installation DVD.
   ```
   vi /etc/yum.repos.d/cdrom.repo
   ```

   Here’s a sample yum configuration file.

   ```
   [RHEL-CDROM]
   name=RHEL CDRom
   baseurl=file:///media/cdrom/Server/
   gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
   enabled=1
   gpgcheck=0
   ```
6. Verify your yum configuration files.
   ```
   yum list httpd
   yum list apr
   ```

> **TIP:** If you have TeamForge installed, you would have the installation repository already configured.

3. Upgrade Review Board.
   ```
   yum install teamforge
   ```
4. Do this on sites without internet access.
   1. Contact the CollabNet Support and get the `python-modules-sources.zip` file.
   2. Unzip the `python-modules-sources.zip` file to `/opt/collabnet/teamforge/service/reviewboard/resources/SOURCES/python-modules-sources`.

```
unzip python-modules-sources.zip -d /opt/collabnet/teamforge/service/r
eviewboard/resources/SOURCES/python-modules-sources
```

5. Provision services.
   ```
   teamforge provision
   ```

6. Restore the Review Board data.

   ✓ The default Review Board data directory has been changed from `/opt/collabnet/reviewboard/data` to `/opt/collabnet/teamforge/var/reviewboard/data` in TeamForge 17.4. If you are upgrading from TeamForge 17.1 or earlier to TeamForge 17.4 or later, regardless of whether you upgrade Review Board on the same or new hardware, you must back up your Review Board data directory from `/opt/collabnet/reviewboard/data` and restore it to `/opt/collabnet/teamforge/var/reviewboard/data`.

   ✓ If you are upgrading from TeamForge 17.4 (or later), the `/opt/collabnet/teamforge/var` directory would have been restored already as part of your TeamForge upgrade process, in which case you can skip restoring the `/opt/collabnet/teamforge/var` directory again.

   ✓ If you are upgrading on a new hardware, ensure that you have already copied the backup of the Review Board data directory to the `/tmp` directory of the new server.
   ```
   cd /opt/collabnet/teamforge/var/
   tar -zxvf /tmp/reviewboard_data.tgz
   ```

7. If SCM is installed on a separate box, run the following script to authenticate a scmviewer user against a TeamForge Subversion repository for creating a new review request.
   ```
    python ./svn-auth.py --repo-path=https://<scm_domain>/svn/repos/<repo_dir
   _name>
   ```

Post Upgrade Tasks

- Add Review Board to Projects
- Users are not getting email notifications for review requests and reviews. What should I do?

Bootstrap Review Board Post Install or Upgrade

Use the following instructions if you want to bootstrap Review Board (drop Review Board database tables and recreate them again) for some reason post installation or upgrade.

1. Log on to the server that hosts the Review Board.
2. Select **My Workspace > Admin**.

3. Select **Projects > Integrated Apps**.
4. Select **Review Board** and click **Delete**.
5. Stop TeamForge.
   ```
   teamforge stop
   ```
6. Start the TeamForge database services.
   ```
   teamforge start -s postgres
   ```
7. Bootstrap the Review Board database.
   ```
   teamforge bootstrap -s reviewboard-database-postgres
   ```
8. Bootstrap the Review Board.
   ```
   teamforge bootstrap -s reviewboard
   ```
9. Start TeamForge.
   ```
   teamforge start
   ```

When TeamForge Site Administrator has made the Review Board application available, Project Administrators can add it as one of their project tools.

When you add Review Board to your project, it works just like the other TeamForge tools, authorization, authentication, go-urls, association, linkification and source code management support.

For this example, we'll call your project "testproject" and we'll assume you have Project Admin rights in that project.

1. Click **Project Admin** from the **Project Home** menu.
2. Click **Tools** to see the list of integrated applications available in the site.
3. Click **Add Tool**.

4. Select **Review Board** from the list of tools displayed.

   Set the values that make sense for your TeamForge installation, then click **Save**.

Option	Description
Prefix	Specify a unique alphanumeric string that will identify this tool throughout the site. For example, suppose you set your prefix to `ZZ`. Entering `ZZ_1` in the `Jump to ID` redirect you to review request id 1 of the "testproject" project. Another may also add Review Board as a tool, with a prefix of YY.Jumping to `YY_1` will redirect to review request id 1 of that project. **NOTE:** The tool prefix cannot be changed after you have set it.

Option	Description
Exclude Repositories	Use a comma to separate the directory names of the repositories that you want to exclude from syncing with Review Board. To include all repositories, enter “None”. **IMPORTANT:** You must sync repositories with Review Board (select **Synchronize Repositories** check box) to have the repositories listed in this **Exclude Repositories** text box excluded.
Synchronize Repositories	Synchronize TeamForge Subversion repositories with Review Board. **NOTE:** Select this check box every time you want to sync repositories to Review Board. When you add Review Board to your project with this check box selected, the SVN repositories are synced once and the check box is cleared. For any new repositories you create, you must edit the Review Board (**Project Admin > Project Toolbar > Edit Integrated Application**) and sync new repositories with Review Board by selecting this check box.

If a Review Board button appears along with the prefix as a tooltip when you mouse over the button, your job is done.

An important aspect of the end-to-end development lifecycle is the creation and storage of software packages that are often binary artifacts. In the Java world, these are usually reusable jars that are used by other projects. Binary artifact repository managers are software systems that manage, version, and store binary artifacts. Examples of such repository managers are JFrog Artifactory, and Sonatype Nexus.

Here is the design overview of integrating TeamForge, a full ALM suite with binary repository managers.

What is a binary artifact repository?

A binary artifact repository stores binary artifacts along with the metadata in a defined directory structure, conceptually similar to a source code repository. The metadata describes the binary software artifact and includes information such as dependencies, versioning, and build promotions. Maven is the widely used tool for dependency management, especially for Java projects. Maven represents dependencies in an XML file called Project Object Model (POM). Other tools can use similar approaches to store documentation archives, source archives, Flash libraries and applications, and Ruby libraries.

How does a binary artifact repository manager help?

Some of the advantages of using a binary artifact repository manager are:

- **Dependency management**: Nexus and Artifactory can act as Maven repositories. Maven is a widely used Java dependency management and build tool.
- **Efficient builds**: With the help of a binary artifact repository manager, you can save the download time from public repositories as the artifacts once downloaded are cached locally.
- **Predictability and release stability**: Once published onto a release repository, the binary artifact and metadata do not change. It ensures predictable and repeatable builds.
- **Control and audit**: If you want to standardize libraries that are used in your software, the binary artifact repository helps track the versions of your software components. Also it enables you to audit the licenses of your third-party components used in your software.
- **Promotes collaboration**: The binary artifact repository enables you to share components with other teams.

How to integrate Nexus with TeamForge?

> **IMPORTANT:** TeamForge-Nexus integration is not supported in SUSE Linux platform. See TeamForge Installation Requirements.

TeamForge supports integration with Nexus in both the ALM and SCM modes. Nexus integration has been tested by CollabNet for Nexus and Nexus professional versions 2.9, 2.10, and 2.11.

To integrate Nexus with TeamForge:

1. Download and install the Nexus OSS if you do not have a Nexus instance running.
2. Download and install the CollabNet Nexus integration plugin.
3. Change your build system and use the CollabNet supplied Maven deploy plugin for end-to-end traceability.
4. Set up the TeamForge EventQ activity source to provide your teams with end-to-end visibility from requirements to source code all the way to deployed binary artifacts.

Install the TeamForge-Nexus Integration Plugin

You need to have the following information handy before you start off with the installation:

- Installation path of the running Nexus instance.
- TeamForge host's URL.
- TeamForge site administrator credentials.
- A suitable name for your Nexus instance; the Binaries App in TeamForge refers to this name.

You must have a Nexus instance running for the integration. If you are upgrading from an earlier version of the plugin, ensure that the old plugin is completely removed from the directory and the new plugin is unzipped on the same directory before you restart the Nexus instance.

- To set up a Nexus server, see Install Nexus.
- To install the TeamForge-Nexus integration plugin, see Install the TeamForge-Nexus Integration Plugin.
- To upgrade the TeamForge-Nexus integration plugin, see Upgrade the TeamForge-Nexus Integration Plugin.

Accessing Nexus through TeamForge: You have to introduce a TeamForge project context in Nexus and allow authentication to use TeamForge credentials for logging into Nexus directly. Accessing Nexus through the TeamForge project toolbar provides you with Single Sign-on (SSO). It logs you into Nexus automatically with the project context. You can allow RBAC (Role Based Access Control) using TeamForge roles.

How to integrate Artifactory with TeamForge?

> **IMPORTANT:** TeamForge-Artifactory integration is not supported in SUSE Linux platform. See TeamForge Installation Requirements.

TeamForge supports integration with Artifactory in both the ALM and SCM modes. Artifactory integration has been tested by CollabNet for Artifactory Pro 4.7 or later.

To integrate Artifactory with TeamForge:

1. You must have an Artifactory Pro instance running for the integration. Download and install the Artifactory Pro 4.7 or later if you do not have an Artifactory Pro instance running.
2. Download and install the CollabNet Artifactory integration plugin.
3. Change your build system and use the CollabNet supplied Maven deploy plugin for end-to-end traceability.
4. Set up the TeamForge EventQ activity source to provide your teams with end-to-end visibility from requirements to source code all the way to deployed binary artifacts.

Install the TeamForge-Artifactory Integration Plugin

You need to have the following information handy before you start off with the installation:

- TeamForge Host URL.
- TeamForge site administrator credentials.
- A suitable name for your Artifactory instance; the Binaries App in TeamForge refers to this name.
- Artifactory Host URL.
- To set up an Artifactory server, see Install Artifactory Pro.
- To install the TeamForge-Artifactory integration plugin, see Install the TeamForge-Artifactory Integration Plugin.

Accessing Artifactory through TeamForge: You have to introduce a TeamForge project context in Artifactory and allow authentication to use TeamForge credentials for logging into Artifactory directly. Accessing Artifactory through the TeamForge project toolbar provides you with Single Sign-on (SSO). It logs you into Artifactory automatically with the project context. You can allow RBAC (Role Based Access Control) using TeamForge roles.

Authentication Policies

Nexus

Your site administrator can enable the integration with the following two authentication mechanisms:

- TeamForge and native Nexus login (default)
- TeamForge only
  In both the cases, you can use your TeamForge credentials to log on to Nexus. If your Site Administrator has used the default setup, you can use your pre-existing Nexus credentials.

Roles and Permissions

Following are the two administrative privileges in Nexus:

- Nexus Admin (Site Admin in TeamForge will be a Nexus Admin)
- Project admin (permissions to create, update, and delete binary artifact repositories.)

For all the other users, privileges are based on the TeamForge RBAC setup.

Artifactory

Your site administrator can enable the integration with the following two authentication mechanisms:

- TeamForge and native Artifactory login (default)
- TeamForge only

In both the cases, you can use your TeamForge credentials to log on to Artifactory. If your Site Administrator has used the default setup, you can use your pre-existing Artifactory credentials.

Roles and Permissions

- Artifactory Admin (Site Admin in TeamForge will be an Artifactory Admin)
- A user with any one of the following permissions in TeamForge becomes an Artifactory Admin: **VIEW/CREATE REPOSITORIES**, **VIEW/UPDATE REPOSITORIES** and **VIEW/DELETE REPOSITORIES**.
- All other permissions in TeamForge are mapped as is in Artifactory.

For all the other users, privileges are based on the TeamForge RBAC setup.

TeamForge-Nexus Integration Plugin Release Notes

Here’s the release notes for TeamForge-Nexus integration plugin. Show/Hide

IMPORTANT: You must use the latest Nexus plugin regardless of the TeamForge version you have installed or upgraded to.

TeamForge-Nexus Integration Plugin v2.0

Name of the plugin: `CTF-Nexus-Integration-Plugin-2.0`.

Release Highlights

- Usability of the Nexus installer (used to integrate Nexus with TeamForge) has been enhanced.
- Improved peformance while loading Nexus UI.

Bug Fixes

- When users log on to Nexus using CTF user credentials, the Nexus UI showed poor performance. This is fixed.
- Fixed the time delay due to the presence of Nexus plugin in Maven builds.

TeamForge-Nexus Integration Plugin v2.0.1

Name of the plugin: `CTF-Nexus-Integration-Plugin-2.0.1`.

Release Highlights

- Minor patch release.
- Enhanced the TeamForge top navigation bar supported by Angular JS.
- Released with TeamForge 16.7.

TeamForge-Nexus Integration Plugin v2.1.1

Name of the plugin: `CTF-Nexus-Integration-Plugin-2.1.1`.

Release Highlights

- User session management between Nexus application and TeamForge integrated with Nexus has been handled in a much better way so that communication between TeamForge and Nexus happens smoothly as expected. This enhancement is done for the Nexus versions 2.9 through 2.14.

- Improved peformance while loading Nexus UI.
- Released with TeamForge 17.4.

Bug Fixes

- Fixed the issue in which the `500: Internal Server Error` was shown instead of `401: Unauthorized Error` for any user activities in Nexus after a session timeout for the Nexus versions 2.9, 2.10, and 2.11. This fix is applicable for Nexus versions 2.9 through 2.14.
- A `403: Forbidden` error was shown as Nexus authentication failed for users whose session has timed out due to the expiration of OAuth token. This is fixed.

Also see: Known Limitations with TeamForge-Nexus/Artifactory Integrations

Known Limitations with TeamForge-Nexus/ Artifactory Integrations

Here is a list of known limitations of TeamForge-Nexus/Artifactory integration.

TeamForge-Nexus Integration

- While it is possible for users (all TeamForge users but site administrators) with ‘create repository’ permission to create binary repositories via the TeamForge **Binaries** application, such users cannot create repositories directly on the Nexus server (using the Nexus UI) as there is no TeamForge project mapping available in Nexus.
- A TeamForge user cannot be configured as an anonymous user in Nexus as TeamForge users are not available in the Nexus database.
- TeamForge broadcast messages and license notifications are not visible in Sonatype Nexus pages in TeamForge. This is due to a limitation with the TeamForge-Nexus integration plugin.
- When the TeamForge Project administrator changes the existing binary permissions for a user, the changes will not immediately take place due to the cache implementation in Nexus to improve the performance of session handling. Due to this limitation, when a user tries to create a new Nexus repository, he will get the permission denied error. Hence the user must wait for the changes to take effect. However, the user can view the changes immediately on a standalone Nexus application by restarting his session on it.

TeamForge-Artifactory Integration

Artifactory repository names must be unique. You cannot have two or more repositories with the same name even if the repositories are of different types such as Remote or Hosted repositories. However, if you want to create two (or more) repositories with like-sounding names, you can devise and follow a repo naming convention to uniquely identify repositories of different types. For example, a “central” repo can be named as “central-local” and “central-remote” to uniquely refer to the “Hosted” and “Remote” repository types respectively.

The following instructions are for installing Nexus Open Source version 2.9.0 or later as a stand-alone server and integrating it with TeamForge.

Nexus comes bundled with a Jetty instance that listens to all configured IP addresses on a host (0.0.0.0) and runs on port 8081 by default.

Installing Nexus is straightforward. Unzip the Nexus bundle in a directory and start Nexus.

> **IMPORTANT:** Though Nexus can be installed on Mac OS, CollabNet did not test Nexus integration on Mac OS.

1. Log on to the Nexus server.

2. Download the Nexus 2.9.0 or later zip file and unzip the content to a directory of your choice. See Installing Nexus. You can find two directories, a directory that contains Nexus installation files and folders (hereinafter referred to as `<nexus-install-directory>`) and a Nexus work directory (hereinafter referred to as `<nexus-work-directory>`).

   ✓ `<sonatype-work>` is the default Nexus work directory. As a notation, `<nexus-work-directory>` is used in place of `<sonatype-work>` in this document.

   ✓ Make sure you have full access permissions on all Nexus folders.

3. Open the command prompt and start Nexus.
   - Linux:
     ```
     cd <nexus-install-directory>
     ```
     Add the following token in the `<nexus-install-directory>/conf/nexus.properties` file and start Nexus:
     ```
     nexus.ui.keepAlive=false
     ./bin/nexus start
     ```
   - Windows:
     ```
     cd <nexus-install-directory>
     ```

Add the following token in the `<nexus-install-directory>\conf\nexus.properties` file and start Nexus:

```
nexus.ui.keepAlive=false
\bin\nexus start
```

4. Verify if Nexus is running by accessing the URL: `<nexus host name>:port/nexus/index.html`.

> **TIP:** Default port is 8081. In case you have multiple Nexus instances, modify the application-port token in `/<nexus-install-directory>/conf/nexus.properties` file.

5. Stop Nexus.
   - Linux:
     ```
     ./bin/nexus stop
     ```
   - Windows:
     ```
     \bin\nexus stop
     ```

Once you have your Nexus server set up, install the TeamForge-Nexus integration plugin.

You must keep the following information handy before installing the TeamForge-Nexus integration plugin:

- **Nexus absolute path**: This is the path to the directory where you have Nexus installed. In other words, the path to the directory where you have your Nexus files unzipped. For example, `/u1/nexus/nexus-2.11.4-01/`.
- **TeamForge host URL**: The URL to access TeamForge. For example, http://ctf.cloud.collab.net.
- TeamForge administrator user name and password.
- **Nexus Application Name**: The name given to your Nexus integration. In other words, the name found in the Nexus integrated application configuration file.
- **Nexus Application Prefix**: The prefix chosen for Nexus. In other words, the prefix found in the Nexus integrated application configuration file.
- **Nexus URL**: The fully qualified Nexus URL.

1. Log on to the Nexus server.
2. Stop Nexus if it’s running.
   - Linux:
     ```
     ./bin/nexus stop
     ```
   - Windows:
     ```
     \bin\nexus stop
     ```
3. Download the `CTF-Nexus-Integration-Plugin-2.1.1.zip` file.
4. Unzip the `CTF-Nexus-Integration-Plugin-2.1.1.zip` file.

  - Linux:

    ```
    cd <nexus-work-directory>/nexus/plugin-repository
    unzip CTF-Nexus-Integration-Plugin-2.1.1.zip
    ```
  - Windows: Use a utility such as WinRAR.

5. Install the TeamForge-Nexus integration plug-in.

   ```
   sudo java -jar <nexus-work-directory>/nexus/plugin-repository/CTF-Nexus-Integration-Plugin-2.1.1/util/installer.jar -enable
   ```
6. Enter the Nexus absolute path and TeamForge host URL when prompted.
7. Start Nexus.
   - Linux:

     ```
     cd <nexus-install-directory>
     ./bin/nexus start
     ```
   - Windows:

     ```
     cd <nexus-install-directory>
       \bin\nexus start
     ```
8. Once Nexus is up and running, upload the Nexus IAF descriptors to TeamForge.

   ```
   java -jar <nexus-work-directory>/nexus/plugin-repository/CTF-Nexus-Integration-Plugin-2.1.1/util/installer.jar -installxml
   ```

   Enter the TeamForge Host URL, TeamForge admin user name and password, Nexus application name, Nexus application prefix and Nexus URL when prompted.

Related Links

TeamForge-Binary Integration FAQs

CollabNet releases new versions of the TeamForge-Nexus integration plugin. It is recommended to upgrade your TeamForge-Nexus integration plugin whenever a new version is available.

TeamForge-Nexus Integration Plugin Release Notes

Here's the release notes for TeamForge-Nexus integration plugin. Show/Hide

IMPORTANT: You must use the latest Nexus plugin regardless of the TeamForge version you have installed or upgraded to.

TeamForge-Nexus Integration Plugin v2.0

Name of the plugin: `CTF-Nexus-Integration-Plugin-2.0`.

Release Highlights

- Usability of the Nexus installer (used to integrate Nexus with TeamForge) has been enhanced.
- Improved peformance while loading Nexus UI.

Bug Fixes

- When users log on to Nexus using CTF user credentials, the Nexus UI showed poor performance. This is fixed.
- Fixed the time delay due to the presence of Nexus plugin in Maven builds.

TeamForge-Nexus Integration Plugin v2.0.1

Name of the plugin: `CTF-Nexus-Integration-Plugin-2.0.1`.

Release Highlights

- Minor patch release.
- Enhanced the TeamForge top navigation bar supported by Angular JS.
- Released with TeamForge 16.7.

TeamForge-Nexus Integration Plugin v2.1.1

Name of the plugin: `CTF-Nexus-Integration-Plugin-2.1.1`.

Release Highlights

- User session management between Nexus application and TeamForge integrated with Nexus has been handled in a much better way so that communication between TeamForge and Nexus happens smoothly as expected. This enhancement is done for the Nexus versions 2.9 through 2.14.
- Improved peformance while loading Nexus UI.
- Released with TeamForge 17.4.

Bug Fixes

- Fixed the issue in which the `500: Internal Server Error` was shown instead of `401: Unauthorized Error` for any user activities in Nexus after a session timeout for the Nexus versions 2.9, 2.10, and 2.11. This fix is applicable for Nexus versions 2.9 through 2.14.
- A `403: Forbidden` error was shown as Nexus authentication failed for users whose session has timed out due to the expiration of OAuth token. This is fixed.

Also see: Known Limitations with TeamForge-Nexus/Artifactory Integrations

Upgrade the TeamForge-Nexus Integration Plugin

1. Log on to the Nexus server.
2. Stop Nexus if it's running.
   - Linux:
     ```
     ./bin/nexus stop
     ```
   - Windows:
     ```
     \bin\nexus stop
     ```
3. Modify the `ctf_nexus.properties` file.
   ```
   vim sonatype-work/nexus/conf/ctf_nexus.properties
   ```

   Replace `TIME_TO_HOLD_CACHE=60` with the following two properties:

   ```
   TIME_TO_HOLD_USER_CACHE=1800
   TIME_TO_HOLD_PERMISSION_CACHE=1200
   ```

   > **TIP:** `TIME_TO_HOLD_USER_CACHE` and `TIME_TO_HOLD_PERMISSION_CACHE` are in seconds.

4. Back up the existing TeamForge-Nexus integration plugin directory: `<nexus-work-directory>/nexus/plugin-repository/<CTF-Nexus-Integration-Plugin>`.
5. Download the `CTF-Nexus-Integration-Plugin-2.1.1.zip` file.
6. Unzip the `CTF-Nexus-Integration-Plugin-2.1.1.zip` file you downloaded into the `<nexus-work-directory>/nexus/plugin-repository/` directory.
   - Linux:
     ```
     cd <nexus-work-directory>/nexus/plugin-repository
     unzip CTF-Nexus-Integration-Plugin-2.1.1.zip
     ```
   - Windows: Use a utility such as WinRAR.
7. Start Nexus.
   - Linux:

```
cd <nexus-install-directory>
./bin/nexus start
```

- Windows:

```
cd <nexus-install-directory>
   \bin\nexus start
```

Related Links

TeamForge-Binary Integration FAQs

The following instructions are for installing Artifactory Pro 4.7 as a stand-alone server and integrating it with TeamForge.

Installing Artifactory Pro is straightforward. Unzip the Artifactory Pro bundle in a directory and start Artifactory Pro.

> **NOTE:** Though Artifactory Pro can be installed on Mac OS, CollabNet did not test Artifactory Pro integration on Mac OS.

1. Log on to the Artifactory Pro server.
2. Download the Artifactory Pro zip file.
3. To install Artifactory Pro manually, simply unzip the downloaded Artifactory Pro zip file to a location on your file system. This will be your `$ARTIFACTORY_HOME` location. See Installing Artifactory Pro.
4. Run Artifactory Pro and verify the installation.

   Artifactory Pro can be accessed using the following URL: http://SERVER_DOMAIN:8081/artifactory

   For example, if you are testing on your local server you would use: http://localhost:8081/artifactory

Once you have your Artifactory server set up, install the TeamForge-Artifactory integration plugin.

You must keep the following information handy before installing the TeamForge-Artifactory integration plugin:

- **TeamForge Host URL**: The URL to access TeamForge. For example, http://ctf.cloud.collab.net.
- TeamForge administrator user name and password.
- **Artifactory Application Name**: The name given to your Artifactory integration. In other words, the name found in the Artifactory integrated application configuration file.
- **Artifactory Host URL**: The fully qualified Artifactory URL.

1. Log on to the Artifactory server.
2. Download the `artifactory-teamforge-plugin-1.0.tar` file.

3. Unzip the `artifactory-teamforge-plugin-1.0.tar` file.
    - Linux:
      ```
      cd $ARTIFACTORY_HOME/etc/plugins/
      tar -xf Artifactory-teamforge-plugin.tar
      ```
    - Windows: Use a utility such as WinRAR.
4. Install the TeamForge-Artifactory integration plugin passing the -m flag to import TeamForge certificates. This is required only on SSL-enabled sites (https sites).
   ```
   sudo java -jar $ARTIFACTORY_HOME/etc/plugins/lib/installer-1.0.jar -m
   ```

   Enter the TeamForge Host URL when prompted.

5. Start Artifactory.
6. Once Artifactory is up and running, upload the Artifactory IAF descriptors to TeamForge.
   ```
   sudo java -jar $ARTIFACTORY_HOME/etc/plugins/lib/installer-1.0.jar -i
   ```

   Enter the TeamForge Host URL, TeamForge Administrator user name and password, Artifactory application name and Artifactory Host URL when prompted.

Related Links

- TeamForge-Binary Integration FAQs

With TeamForge—Nexus integration enabled, you can create one or more binary repositories and link them to your project.

Create a Binary Artifact Repository

> **NOTE:** Before you can create a repository for binary artifacts, a site administrator must set up TeamForge-Nexus integration and add one or more binary servers to your TeamForge site.

1. Click **BINARIES** from the **Project Home** menu.
2. Click **Create Repository**. The **Create Repository** page appears.
3. **Server**: Select a Nexus or Artifactory binary repository server from the drop-down list.
4. If you are creating a Nexus repository, type or select the values for the following fields in the **Create Repository** page.

   > **TIP:** Hover your mouse over the Help icon (question mark icon) for relevant tooltip.

    - **Repository Name** and **Repository ID**: Type a name and ID for the binary repository.

- **Repository Type**: Select one of the following types from the drop-down list: **Hosted Repository** or **Proxy Repository** or **Virtual Repository**.
- **Provider**: Select the content provider of the repository from the drop-down list.
- **Repository Policy**: Repositories can store artifacts of a release or snapshot or both. Select a policy from the drop-down list.
- **Override Local Storage Location**: Leave this blank to use the default local storage location. To change the local storage location, type the path (URL) in the following format.
    - **Windows**: file:/{drive-letter}:/
    - **Other operating systems**: file://

1. Modify the default access settings such as **Deployment Policy**, **Allow File Browsing**, **Include in Search** and **Publish URL**, if required.
2. Modify the **Not Found Cache TTL** duration (in minutes), if required.
3. Click **Save**.

5. If you are creating an Artifactory repository, type or select the values for the following fields in the **Create Repository** page.

> **TIP:** Hover your mouse over the Help icon (question mark icon) for relevant tooltip.

- **Repository Type**: Select one of the following types from the drop-down list: **Local Repository** or **Remote Repository** or **Virtual Repository**.
- **Package Type**: Type the package type. For example, “maven”. Package type cannot be changed once the repository is created.
- **Repository Key**: Type the repository key. The repository key must be unique within an Artifactory instance. Can contain no spaces or special characters. Cannot begin with a number.

1. Fill in the fields (modify the default values, if required) listed under the `General`, `Include/Exclude Patterns`, and `Maven Settings` sections.

> **TIP:** Hover your mouse over the Help icon (question mark icon) for relevant tooltip.

2. Click **Save**.

The binary repository is created.

After creating a binary repository, you can now configure traceability for the repository so that your Nexus or Artifactory runtime events are published to EventQ. See Configure Traceability for Binary Artifact Repositories.

Configure Traceability for Binary Artifact Repositories

Once a binary repository is created, configuring the repository to publish runtime data to EventQ is the next step that lets you track and visualize binary artifacts in real time using TeamForge EventQ.

1. Click **BINARIES** from the **Project Home** menu.
2. Select the binary repository for which you want to configure runtime.
3. Click **Configure Traceability**.
4. Click **Create**.
5. Change the **Associated Source** from the drop-down list and click **Update** to change the associated source.
6. Clicking **De-Activate** and **Re-Activate** disables and enables traceability for the repository respectively.
7. Clicking **Force Delete** deletes the traceability configuration including all historical associations.

   > **WARNING:** Exercise caution before deleting the traceability configuration.

8. Click **Close** to exit the **Traceability Configuration** page.

Link Binary Artifact Repository to Project

You can link existing binary artifact repositories, if any, to your project.

1. Click **BINARIES** from the **Project Home** menu.
2. Click **Link Existing Repository**.
3. Select a repository from the list of binary repositories and click **Link to Project**.

Configure Maven Build

CollabNet provides you with a Maven deploy plugin that completely inherits the default (vanilla) Maven deploy plugin. Additionally, it posts binary artifact creation event information on to TeamForge EventQ enabling end-to-end traceability.

To configure your Maven build:

1. Replace the standard deploy with CollabNet's plugin in your POM.xml.

   ```
   <pluginRepositories>
   <pluginRepository>
     <id>collabnet</id>
   ```

```
  <name>Collabnet Public Repo</name>
  <url>http://mvn.collab.net/nexus/content/groups/public/</url>
</pluginRepository>
</pluginRepositories>
<plugins>
....
  <plugin>
    <groupId>org.apache.maven.plugins</groupId>
    <artifactId>maven-deploy-plugin</artifactId>
    <version>2.8.2</version>
    <configuration>
      <skip>true</skip>
    </configuration>
  </plugin>
  <plugin>
    <groupId>net.collab.maven.deploy</groupId>
    <artifactId>collabnet-deploy-maven-plugin</artifactId>
    <version>8.2</version>
    <extensions>true</extensions>
    <executions>
      <execution>
        <id>default-deploy</id>
        <phase>deploy</phase>
        <goals>
          <goal>deploy</goal>
        </goals>
      </execution>
    </executions>
    <configuration>
      <skipLinkToBinaries>true</skipLinkToBinaries>
      <amqpServer>amqp://host.with.amqp</amqpServer>
      <sourceAssociationKey>as-per-orchestrate-setup</sourceAssociationKey>
      <associatedBuildNumber>${env.BUILD_NUMBER}</associatedBuildNumber>
    </configuration>
  </plugin>
</plugins>
```

IMPORTANT: Make sure the <skip> tag is set to `true` to prevent more than one Nexus/Artifactory notification for a single Nexus/Artifactory artifact deployment. If <skip> is not set to true,

> notifications are sent by both the maven-deploy-plugin and the collabnet-deploy-maven-plugin for a single binary artifact.

2. Setup the Nexus/Artifactory and EventQ credentials in `settings.xml`. You may find this file in the Maven home directory. For example, in the following illustration, your distribution management section has a repository id of the local-nexus and the amqpServer, amqp://host.with.amqp is setup as mentioned in Step 1:

```
<settings>
 <servers>
  <server>
   <id>local-nexus</id>
   <username>your_ctf_username</username>
   <password>xxxxxxxx</password>
  </server>
  <server>
   <id>amqp://host.with.amqp</id>
   <username>myproj_nexus</username>
   <password>xxxxx</password>
  </server>
 </servers>
</settings>
```

If you already have a set of credentials against an AMQP URL in `settings.xml`, all jobs including the one that posts events against different sources can use the same credentials. The following table lists the available configuration items.

Configuration Parameter	Description	Mandatory	Default Value	Example
amqpServer	An amqp url for the form amqp://host.domain where EventQ is listening to.	true	None	amqp://example.com
sourceAssociationKey	The source association key from orchestrate custom event setup.	true	None	uuid format
orchestrateSchemaId	The schema id posted to EventQ. Do not set it in v1.4.1 and later.	true	binary_artifact	binary_artifact
orchestrateSchemaVersion	The version of the schema, usually 1. Omit	false	1	1

Configuration Parameter	Description	Mandatory	Default Value	Example
	to default to 1. Do not set it in v1.4.1 and later.			
amqpUserName	The AMQP user name. It is recommended not to set it in the project POM but using in settings.xml. Take it from the custom event setup in EventQ.	false	None	proj_custom
amqpPassword	The AMQP password. It is recommended not to set it in the project POM but using in settings.xml. Take it from the EventQ settings.	false	None	af78aaee0c5610d21….
amqpQueueName	Do not specify. Defaults to orchestrate.custom.	false	orchestrate.custom	orchestrate.custom
associatedBuildNumber	Specify to the env variable depending on your build system process. Set to ${env.BUILD_NUMBER} for Jenkins.	false	None	${env.BUILD_NUMBER}
skipOrchestrateNotification	Set to true to disable notification.	false	false	False
dieOnOrchestrateError	Defaults to true. Set to false to treat orchestrate notification errors as non-fatal.	false	true	true
component	Used to identify a specific binary artifact as a component in a larger application.	false	None	An ALM platform has several components such as an application server, an indexer, an SCM integration server and so on. These components have their own build process. This property is used to uniquely identify such components in EventQ.
componentOf	Associated with the ‘component’ parameter to	false	None	SCM as a component of Teamforge.

Configuration Parameter	Description	Mandatory	Default Value	Example
	store the details of the component.			

Delete a Binary Artifact Repository

You must have the required permission to delete binary repositories. As a project admin, you can use the delete option only to unlink the repository from a project. If you want to delete a repository from the Nexus, login as a Nexus admin.

1. Click **BINARIES** from the **Project Home** menu.
2. In the list of the repositories, select the repository you want to delete and click **Delete**. The following confirmation message appears:
   `This repository will no longer be accessible. Your site administrator can relink it later. Are you sure you want to unlink this repository?`
3. Click **OK** to delete.

The repository is deleted.

> **NOTE:** This deletion disassociates the repository from your project; only the Site Admin can reinstate the repository.

TestLink is a web-based Test Management system that supports all the various components and processes involved in a testing process. Using TestLink, you can create test specifications, execute test cases, create custom reports, generate test plan metrics and so on.

✓ CollabNet supports only the integration between TestLink and TeamForge. For TestLink support, contact TestLink directly. Click here for more information.

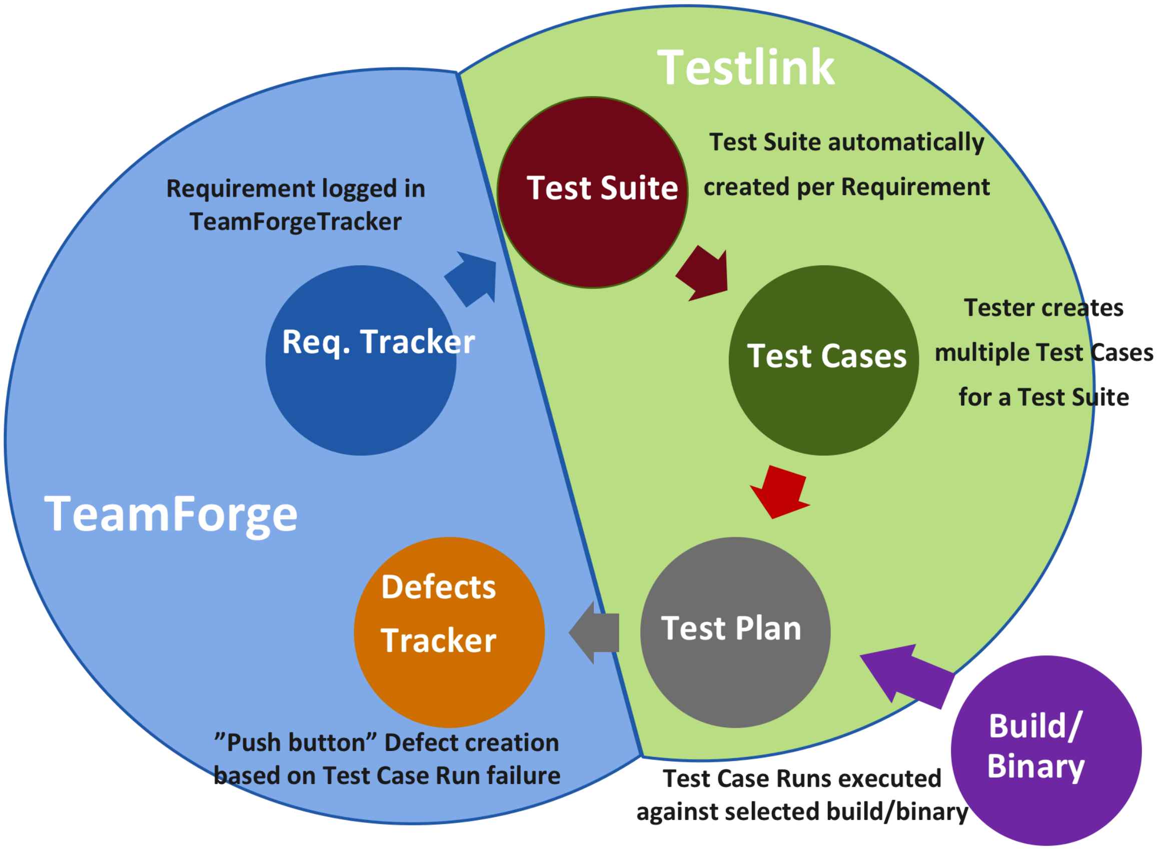
✓ TeamForge 16.7 and later releases support integration only with TestLink 1.9.15 and 1.9.16. TestLink 1.9.17 and later versions are not supported. If you are on earlier versions of TestLink, upgrade to one of the supported TestLink versions and integrate it with TeamForge. This integration does not provide backward compatibility (Data reliability and Migration) to older TeamForge-TestLink 1.9.11 integration that is based on TeamForge's Integrated Application Framework (IAF).

✓ The TeamForge-TestLink Integration Plugin, `collabnet-testlink-1.0.2.jar`, supports integration with Testlink 1.9.15 and 1.9.16.

TeamForge 18.1 supports integration with TestLink 1.9.15 and 1.9.16 to track and synchronize requirement and defect tracker artifacts between these two systems.

The TeamForge-TestLink integration has two primary touch points:

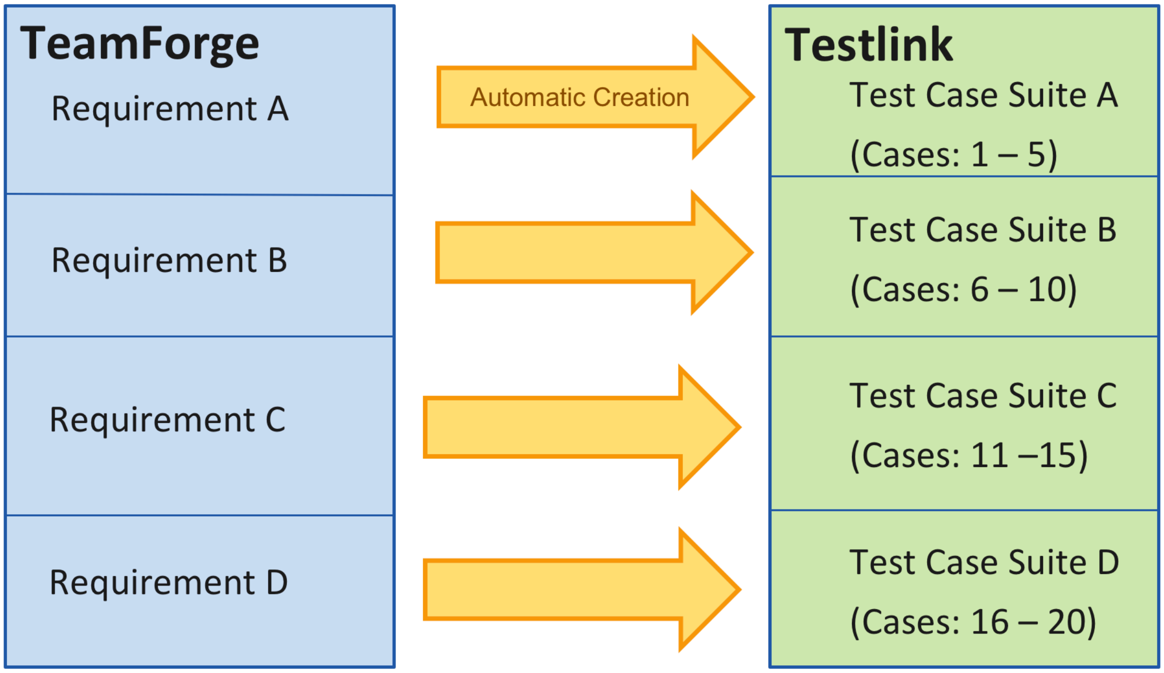
- **Requirements** (TeamForge) to **Test Suites** (TestLink)
- **Test Case Run Failures** (TestLink) to **Defects** (TeamForge)


Testlink
Test Suite
Test Suite automatically created per Requirement
Requirement logged in TeamForgeTracker
Req. Tracker
Test Cases
Tester creates multiple Test Cases for a Test Suite
TeamForge
Defects Tracker
Test Plan
Build/ Binary
"Push button" Defect creation based on Test Case Run failure
Test Case Runs executed against selected build/binary

Here is a sample workflow to illustrate these touch points:

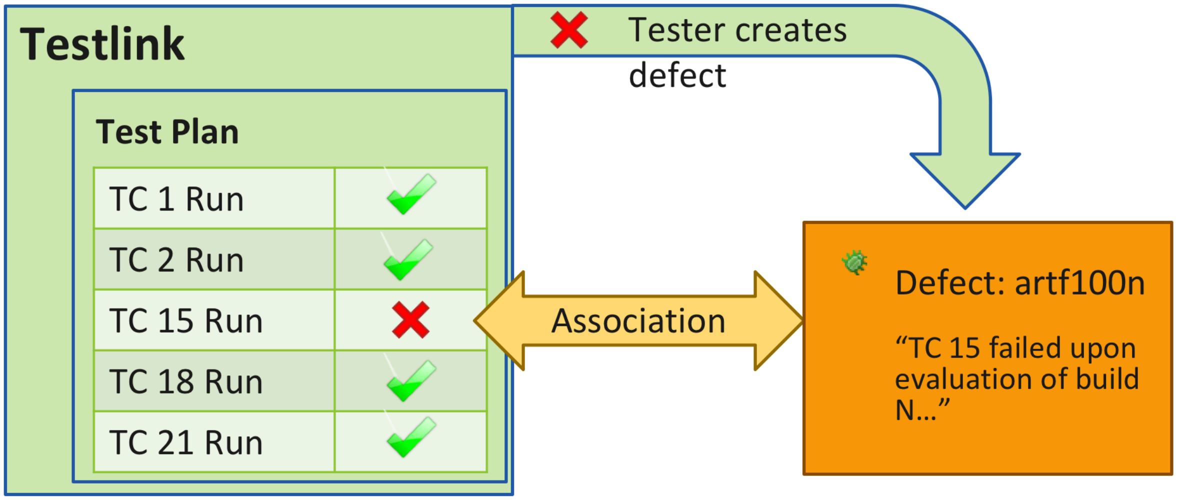
1. Map TeamForge and TestLink projects The first step is to establish a project mapping between the intended TeamForge and TestLink projects. During mapping, in TestLink you'll be able to specify the TeamForge "Requirements" and "Defects" trackers for your project. These trackers will be used throughout the workflow.

2. Create Requirements and Test Suites When a requirement artifact is created in the TeamForge tracker specified in Step 1, you will be given the opportunity to create a matching "Test Suite" in TestLink straight from the artifact creation screen in TeamForge.

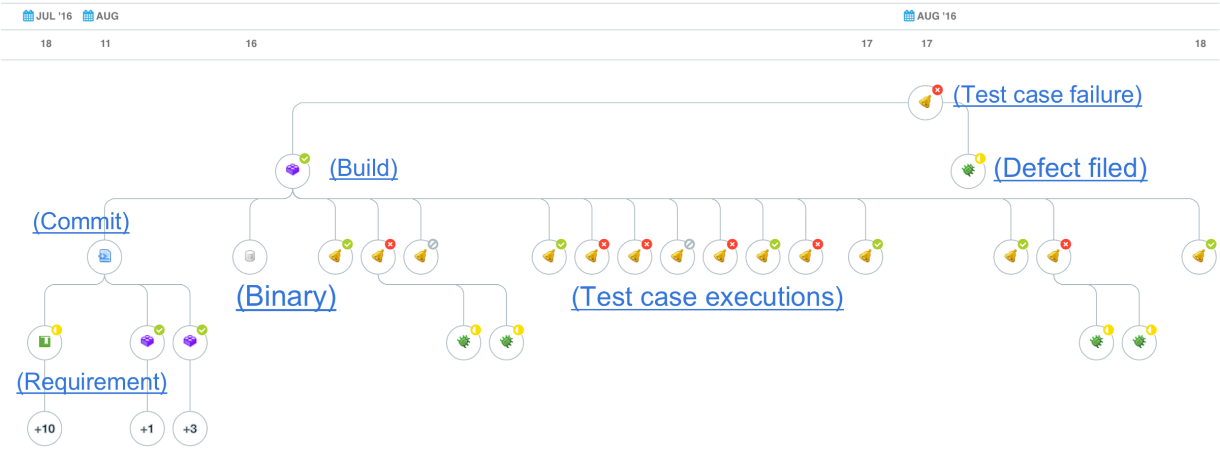


3. Populate Test Suite with Test Cases Within TestLink you can find and populate your Test Suite with Test Cases.
4. Run Test Cases against Builds Once the Test Cases have been defined, you can run them against desired builds, tracking results in TestLink. Test Case runs are associated to the build/binary being tested for traceability purposes.

5. Create Defects for failed Test Case runs If you encounter failing Test Case runs, you have the option to click a button inside TestLink to create a defect in TeamForge. This is a “push button” defect that opens a new TeamForge window pre-populated with Test Case run failure data. An association will be created between the failing Test Case run and the new Defect.



6. Visualize Traceability Once this cycle is complete, you can view the associations chain between Test Case runs and any associated defects, the Builds or Binaries they exercised, the Commits which originated the Build, and ultimately the Requirements that started it all.



Instructions to integrate TeamForge with TestLink 1.9.15 and later.

Installation Requirements for TeamForge-TestLink Integration

- TeamForge Installation Requirements
- TestLink Installation Requirements

✓ CollabNet supports only the integration between TestLink and TeamForge. For TestLink support, contact TestLink directly. Click here for more information.

✓ TeamForge 16.7 and later releases support integration only with TestLink 1.9.15 and 1.9.16. TestLink 1.9.17 and later versions are not supported. If you are on earlier versions of TestLink, upgrade to one of the supported TestLink versions and integrate it with TeamForge. This integration does not provide backward compatibility (Data reliability and Migration) to older TeamForge-TestLink 1.9.11 integration that is based on TeamForge's Integrated Application Framework (IAF).

✓ The TeamForge-TestLink Integration Plugin, `collabnet-testlink-1.0.2.jar`, supports integration with Testlink 1.9.15 and 1.9.16.

✓ By default, only TeamForge users with "Site Administrator" privileges can create test suites. If you want other users to create test suites, you must create a site-wide role in TeamForge, grant "CREATE/VIEW" access to "All Projects" for the role and assign this role to users.

Install the TeamForge-TestLink Integration Plugin on the TeamForge Application Server

1. Log on to the TeamForge Application Server.
2. A new version of TestLink integration jar (`collabnet-testlink-1.0.2.jar`) is available . If you have integrated TestLink with TeamForge using the `collabnet-testlink-1.0.jar` file in the past:
   1. Go to **My Workspace > Admin**.
   2. Select **Projects > System Tools > Customizations**.
   3. Select `collabnet-testlink-1.0.jar`.
   4. Click **Delete**. A confirmation message is displayed. Click **OK**.
3. Download the collabnet-testlink-1.0.2.jar file.
4. Go to **My Workspace > Admin**.
5. Select **Projects > System Tools > Customizations** and click **Create**.
6. Click **Choose File** and select the `collabnet-testlink-1.0.2.jar` file.
7. Click **Add**.

Set up the TestLink Server

1. Log on to the TestLink Server, download and install TestLink.

   Make sure that you have the following RPMs available during TestLink installation:

   ✓ php-xml

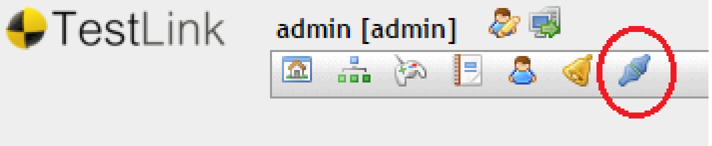
- ✓ php-mcrypt
- ✓ php-mbstring
- ✓ php-bcmath

2. Download the collabnet-testlink-1.0.1.tar file to the `/tmp` directory and untar the file to `<testlink-installation-directory>/plugins/ directory`.

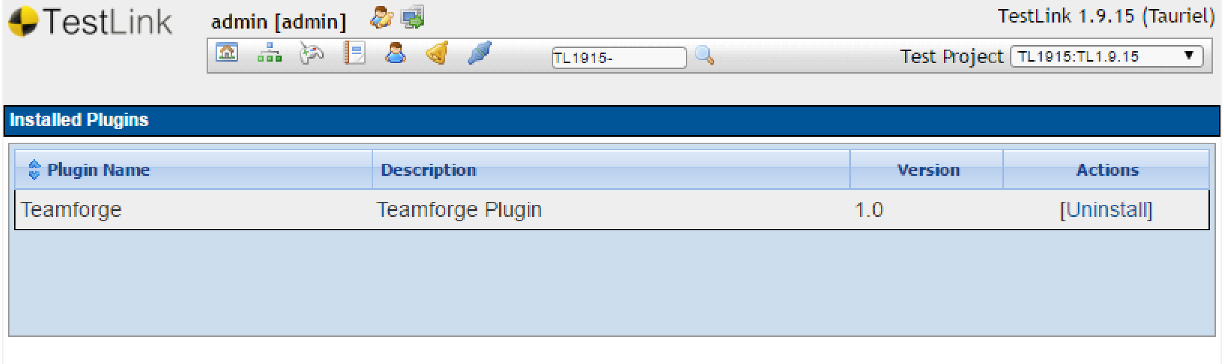
```
cd <testlink-installation-directory>/plugins/
tar -xvf /tmp/collabnet-testlink-1.0.1.tar
```

3. Log on to TestLink.

4. Click the Plugins Management icon from the toolbar.



5. Identify the TeamForge-TestLink integration plugin from the list of Availabe Plugins and click **Install**. The TeamForge-TestLink integration plugin is installed and shows up in the Installed Plugins section.

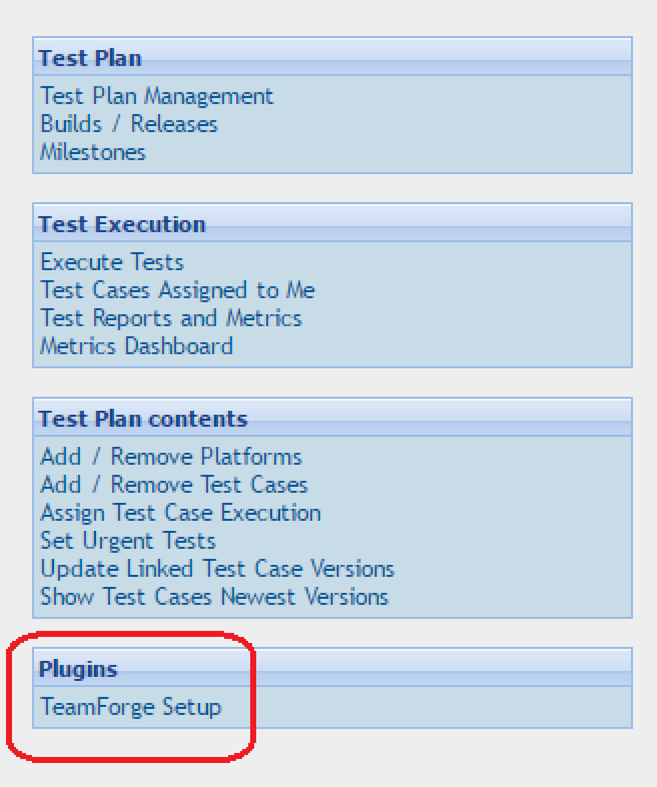


Installed Plugins

Plugin Name	Description	Version	Actions
Teamforge	Teamforge Plugin	1.0	[Uninstall]

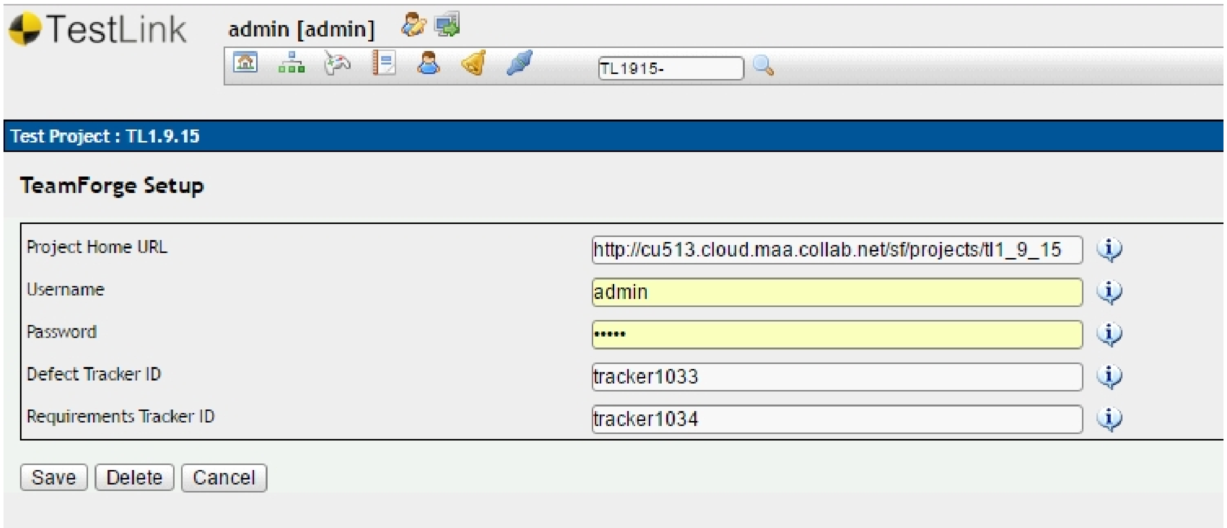
6. Click the **Home** icon from the toolbar.
7. Create a TestLink project.
8. Click **Test Project Management** and click **Create**.
9. Define project attributes such as the name, description, project prefix and so on. Click **Save**.

10. Go to **TestLink Home** and click the **TeamForge Setup** link.



11. Type the TeamForge **Project Home URL**, **Username**, **Password**, **Defect Tracker ID** and **Requirements Tracker ID** and click **Save**.

> **IMPORTANT:** It is assumed that you have a TeamForge project, requirements tracker and defect tracker created already. If not, create them first and then perform this step of setting up TeamForge in TestLink. Have the requirements and defect tracker ID handy while setting up TeamForge in TestLink.



Test Project : TL1.9.15

TeamForge Setup

Project Home URL	http://cu513.cloud.maa.collab.net/sf/projects/tl1_9_15
Username	admin
Password	•••••
Defect Tracker ID	tracker1033
Requirements Tracker ID	tracker1034

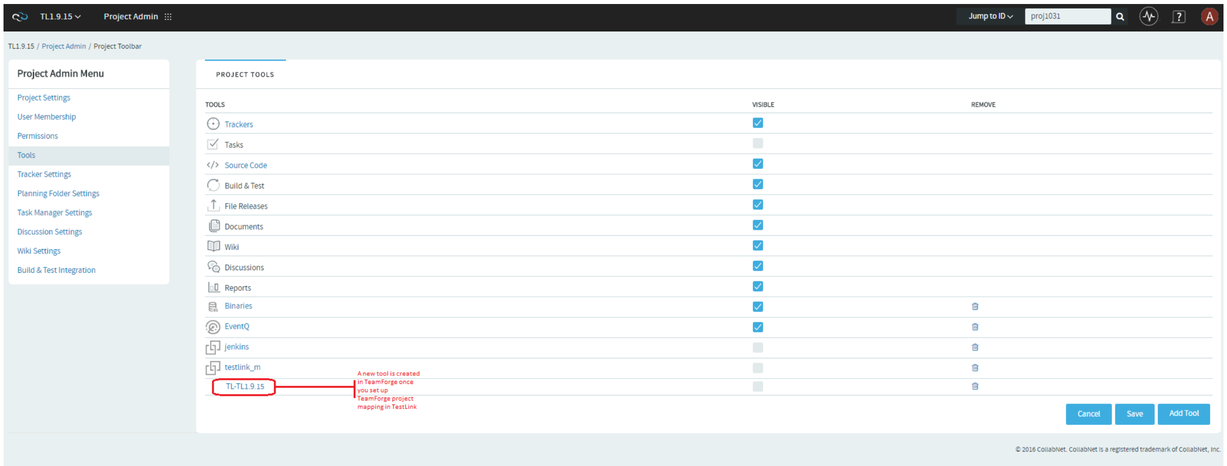
For more information on creating a TeamForge project and setting up trackers, see:

- Create a TeamForge Project
- Create a Tracker

Once you set up the TeamForge project mapping in TestLink, a new TestLink tool is created in TeamForge (**Project Home > Project Admin > Tools**).

Set up the Testlink Activity Source in TeamForge

1. Log on to the Teamforge Application Server.
2. Go to **Project Home > Project Admin > Tools** and select the **TestLink** tool.



TOOLS	VISIBLE	REMOVE
Trackers	☑	
Tasks	□	
Source Code	☑	
Build & Test	☑	
File Releases	☑	
Documents	☑	
Wiki	☑	
Discussions	☑	
Reports	☑	
Binaries	☑	🗑
EventQ	☑	🗑
jenkins	□	🗑
testlink_m	□	🗑
TL-TL1.9.15	□	🗑

A new tool is created in TeamForge once you set up TeamForge project mapping in TestLink

Cancel
Save
Add Tool
© 2016 CollabNet. CollabNet is a registered trademark of CollabNet, Inc.

The Edit Tool page appears.

3. Type a Source Name.
4. Select **Custom Activity Source** from the **Source Type** drop-down list.
5. If required, select an **Associated Source** from the drop-down list. For example, select **Build**.
6. Change the icon, background color and add tags, if you want to.
7. Click **Show Configuration** to view the configuration data.

> **IMPORTANT:** Copy and keep the custom source information such as the Source Association Key, Queue Server, Queue Username and Queue Password handy to be used for Test Suite configuration later in the process.

8. Click **Update**.

Uninstall

Instructions to uninstall the TeamForge-TestLink integration.

1. Log on to the TestLink Server.

2. Delete the TeamForge project mapping.

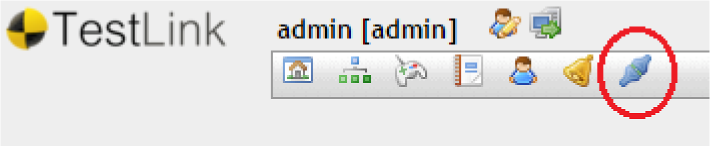
   Before uninstalling the TeamForge-TestLink integration plugin, it is a best practice but not mandatory to delete the TeamForge project mapping in TestLink.

   > **IMPORTANT:** While deleting the project mapping in TestLink, TeamForge custom sources turn inactive. The TestLink tool in TeamForge should be removed manually, if required.

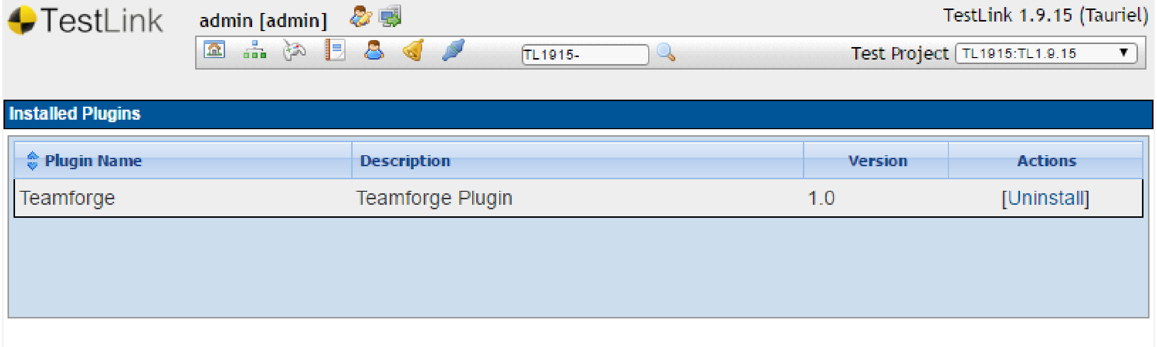
   1. Click **TeamForge Setup**.
   2. Click **Delete**. A confirmation message is displayed.
   3. Click **OK**. The TeamForge project mapping is deleted.

3. Uninstall the TeamForge-TestLink integration plugin.

   1. Click the **Plugins Management** icon from the toolbar.



   2. Identify the TeamForge-TestLink integration plugin from the list of **Installed Plugins** and click **Uninstall**.


TestLink admin [admin] TestLink 1.9.15 (Tauriel)

TL1915- Test Project TL1915:TL1.9.15

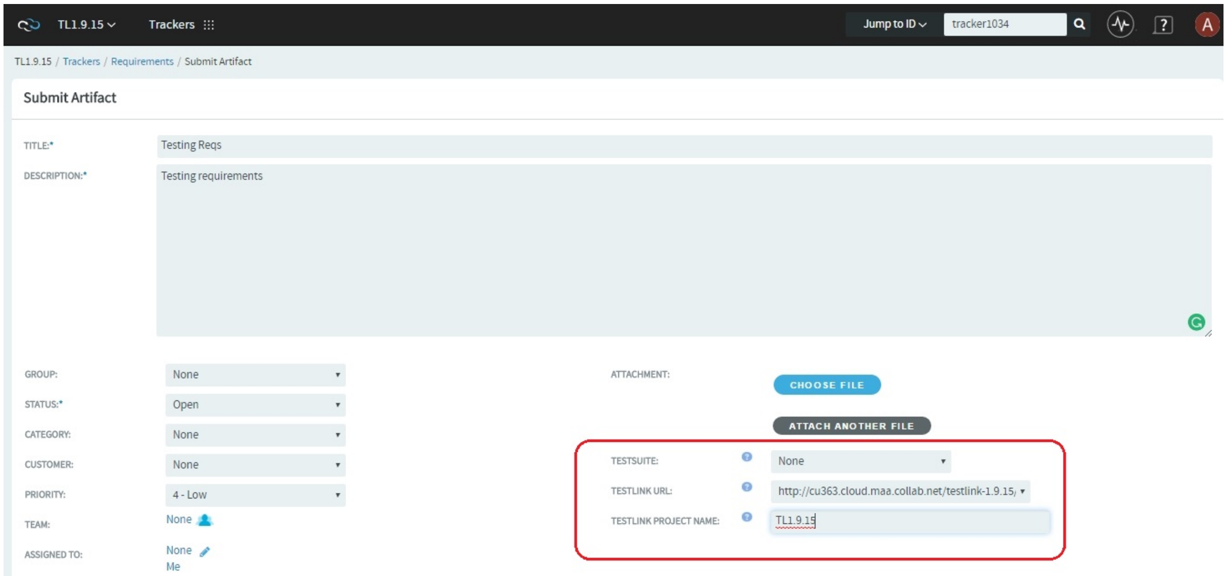
Installed Plugins

Plugin Name	Description	Version	Actions
Teamforge	Teamforge Plugin	1.0	[Uninstall]

The TeamForge-TestLink integration plugin is uninstalled.

Define the requirements in TeamForge project's requirements tracker, create test plans and test cases, execute test cases and file defects for failed test runs.

1. Log on to TeamForge, go to the TeamForge project and define the requirements in the requirements tracker. Create and submit an artifact.

   TL1.9.15 / Trackers / Requirements / Submit Artifact

   Submit Artifact

   TITLE:* Testing Reqs

   DESCRIPTION:* Testing requirements

   GROUP: None

   STATUS:* Open

   CATEGORY: None

   CUSTOMER: None

   PRIORITY: 4 - Low

   TEAM: None

   ASSIGNED TO: None
Me

   ATTACHMENT:

   TESTSUITE: None

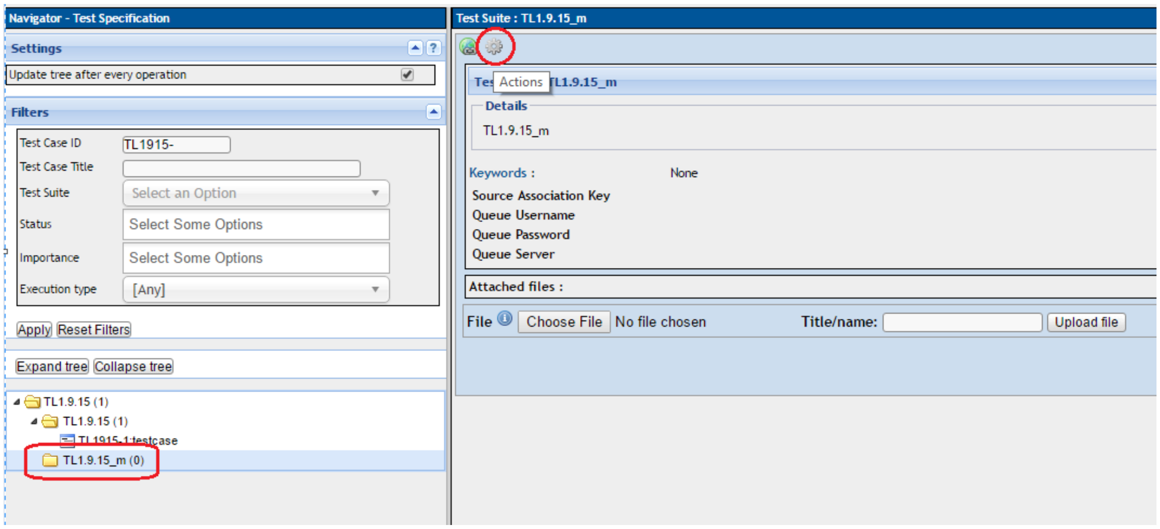
   TESTLINK URL: http://cu363.cloud.maa.collab.net/testlink-1.9.15/

   TESTLINK PROJECT NAME: TL1.9.15

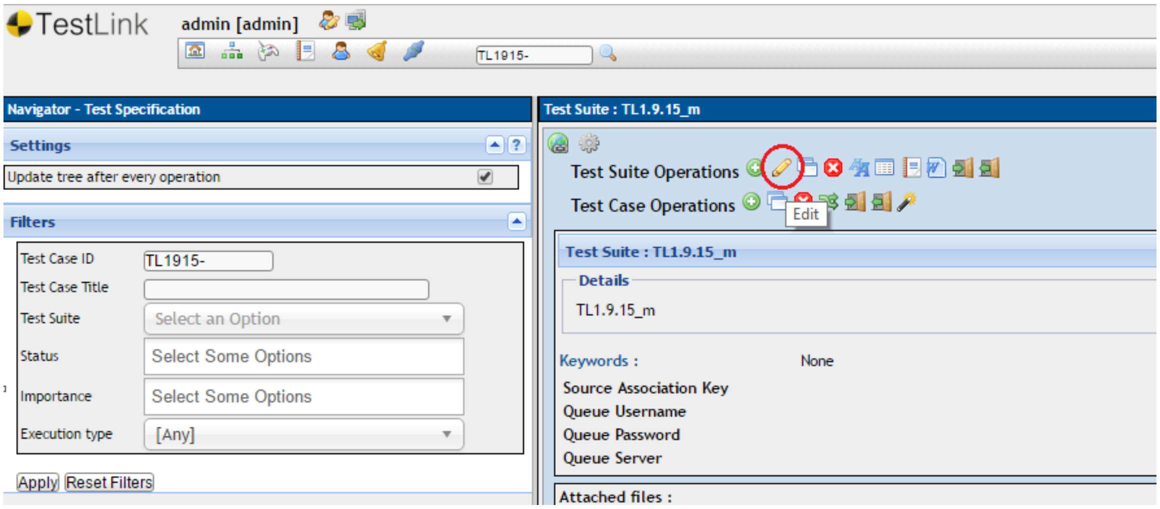
   WARNING: Test Suite titles in TestLink can be of 95 (or less) characters long. Therefore, you must limit requirement artifact titles to 95 characters or less while creating the requirement artifacts in TeamForge. Test Suite creation fails for TeamForge requirement artifacts with titles more than 95 characters.

   1. Select **Create** from the **TESTSUITE** drop-down list. The TESTLINK URL is auto-filled. Leave it as is.
   2. Type the **TESTLINK PROJECT NAME**.

3. Click **Save**.
4. Log on to TestLink and verify if a new Test Suite has been created for the requirements you just created in TeamForge (go to **TestLink Home > Test Specification**).
5. Go to **TestLink Home > Test Specification**, select the Test Suite from the tree and click **Actions**.

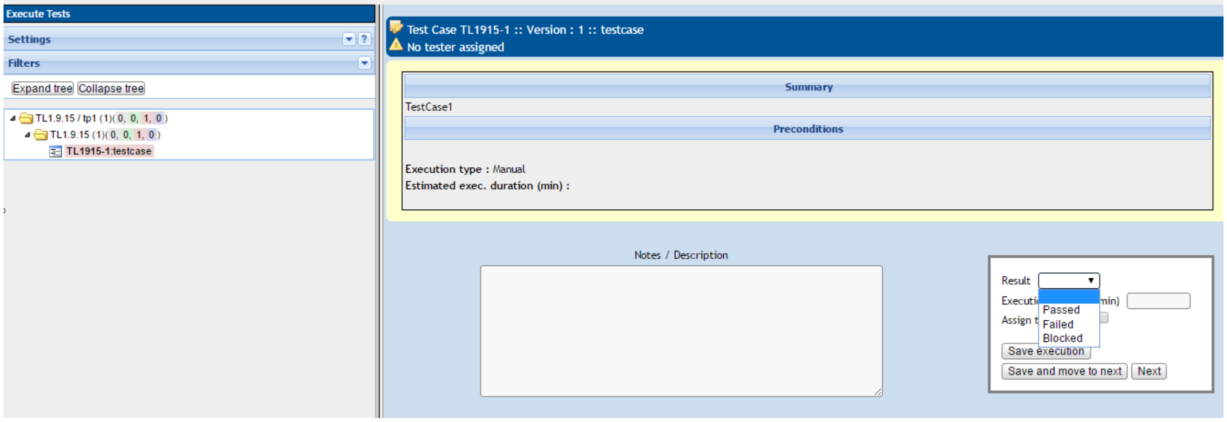


6. Click the **Test Suite Operations > Edit** icon and update the custom source fields such as **Source Association Key**, **Queue Server**, **Queue Username** and **Queue Password** with the values you noted down earlier while setting up the custom source in TeamForge.

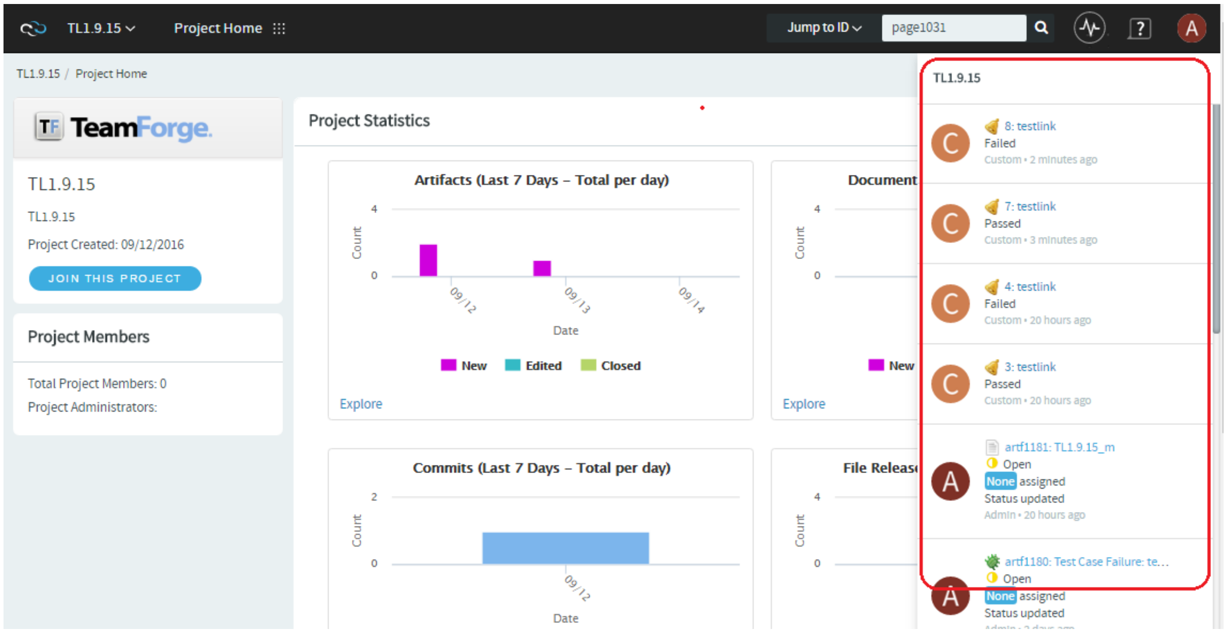


7. Click **Save**.

2. Select the Test Suite you just configured from the tree, click **Actions** and create new test cases and builds. Once you have the test cases and builds created, add specific test cases to builds.
3. Go to **TestLink Home > Execute Tests**, select the test case from the tree, update the **Result** and **Save Execution**.



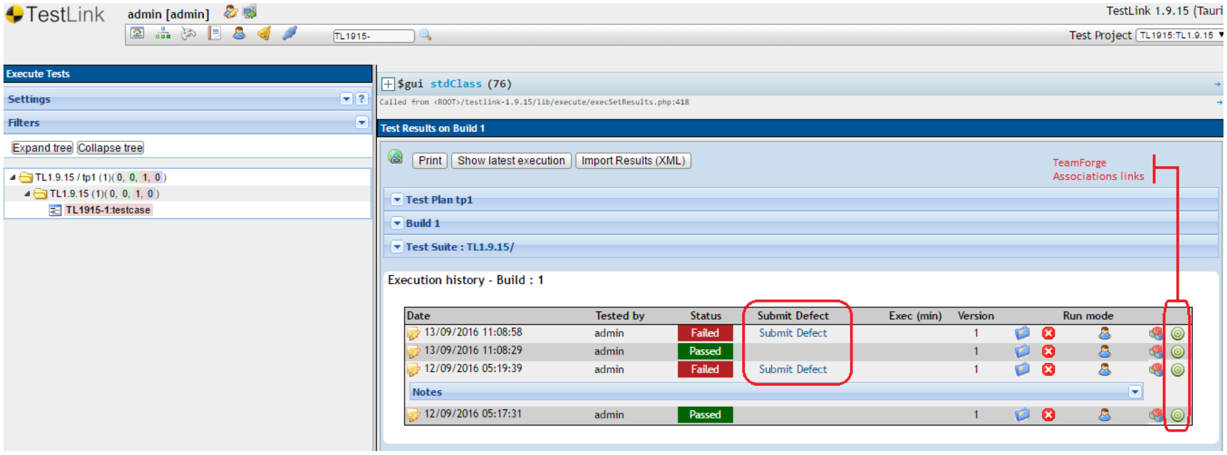
Notifications are added to the TestLink project's Activity Stream in TeamForge for your TestLink activities and events.



The **Execution History** section lists date-wise test case execution data with a **Submit Defect** link to file defects for "Failed" test runs. Clicking the **Submit Defect** link takes you to the TeamForge **Submit Artifact** page.

> **NOTE:** Some of the test execution details such as the Tester's username, build name, time of execution and notes are automatically included in the artifact's `Description` field if and only if you have an active TeamForge session. Clicking **Submit Defect** just takes you to the TeamForge Login page otherwise.

You can also trace associations using the “TeamForge Associations” link for every test case execution.



TestLink admin [admin]

TestLink 1.9.15 (Tauri

TL1915-

Test Project TL1915:TL1.9.15

Execute Tests

Settings

Filters

Expand tree Collapse tree

TL1.9.15 / tp1 (1)(0, 0, 1, 0)

TL1.9.15 (1)(0, 0, 1, 0)

TL1915-1:testcase

```
$gui stdClass (76)
Called from <ROOT>/testlink-1.9.15/lib/execute/execSetResults.php:418
```

Test Results on Build 1

Print | Show latest execution | Import Results (XML)

TeamForge Associations links

Test Plan tp1

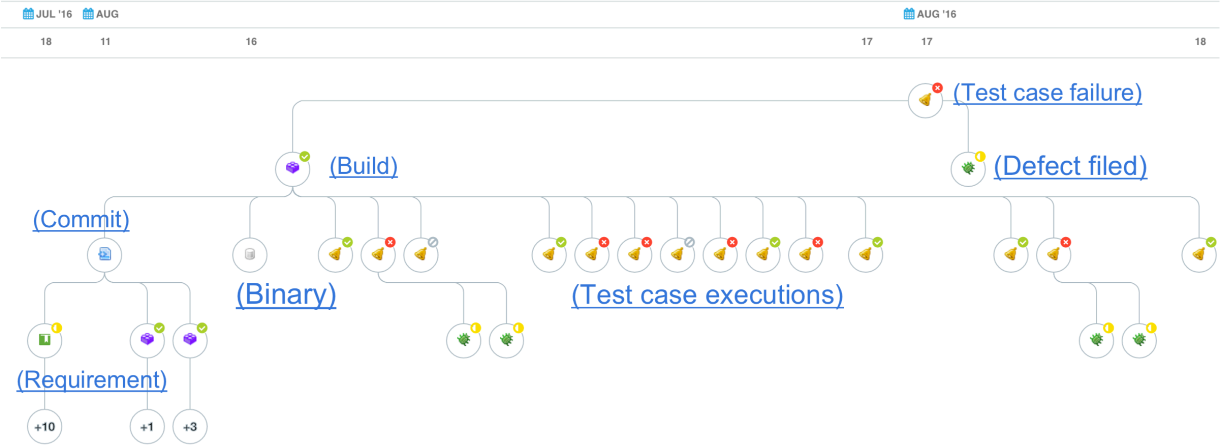
Build 1

Test Suite : TL1.9.15/

Execution history - Build : 1

Date	Tested by	Status	Submit Defect	Exec (min)	Version	Run mode
13/09/2016 11:08:58	admin	Failed	Submit Defect		1	
13/09/2016 11:08:29	admin	Passed			1	
12/09/2016 05:19:39	admin	Failed	Submit Defect		1	
Notes						
12/09/2016 05:17:31	admin	Passed			1	

4. Click **Submit Defect** and file a defect in TeamForge.
5. Click the **TeamForge Associations** link and trace associations.



TeamForge site administration involves managing several aspects of TeamForge including setting up users and a role based access control, managing users, managing scm and other integrated applications, setting up SSL, managing the database and datamart, managing projects and so on.

Here's a list of site administration tasks.

- Manage Site-wide roles
- Manage SCM tools
- Manage global project roles
- Manage projects and project groups
- Manage users
- Manage email settings
- Monitor the site
- Integrate and link external applications
- Set up SSL
- Manage the database and datamart

To assist in the administration of the TeamForge site, a person must have a site administrator user account with a corresponding role on that site.

TeamForge administrators can create suitable site-wide roles and delegate site administration responsibilities.

> **NOTE:** You can choose site administration permissions through site-wide roles.

1. Go to **My Workspace > Admin**
2. Click **Roles** from the **Projects** menu.
3. Click **Create**.
4. On the *Site-Wide Role* tab, write a name and description for the role. The role name is case-sensitive.
5. To prevent inheritance of the role into private projects, select the **Prevent Access** option.

   > **NOTE:** Selecting the option to prevent role inheritance does not affect access to public and gated projects. On selecting **Prevent access**, the user may not be allowed to project-permissions related tasks in private projects.

6. Click **Create**. The restricted site administrator role is created. The **Edit Site-wide Role Permissions** page appears.

   > **NOTE:** You can select the permission for site administration tools as well as for applications available across all projects.

7. Select the appropriate site administration and/or project permissions liste on the *Role Permissions* tab to match the responsibility assigned to a user with that role.

   > **TIP:** You may not want to risk delegating the task of deleting projects, users, groups, roles or categories.

   > **NOTE:** If you are creating a site-wide role that has Project Tracker's "Configure - Site" permission, you must also assign the "Role - View" permission.

   > **NOTE:** To manage artifact types globally, users must have project administrator permissions in a site-wide role.

The role is created. You can assign it to site members at any time.

If restricted site administrators need to do things that are not allowed by a role you have assigned to them, you may need to change the permissions associated with that role.

When you edit a role, all restricted site administrators with that role get the updated permissions automatically.

> **TIP:** You may have prevented the access permission into private projects earlier. Now you can modify the restriction or change other relevant permissions.

1. Go to **My Workspace > Admin**.
2. Click **Roles** from the **Projects** menu.
3. From the list of roles, click the role you want to edit or select the check box and click **Edit**.
4. On the **Edit Site-wide Role Permissions** page, make the changes you need.
   - To edit the title or description of the role, click **Edit**.
   - To edit the site administration and/or project permissions, choose an application from the left side of the page and select or deselect permissions and resources.
   - To edit the site members to whom the role is assigned, click **Assigned Users** tab.
5. Click **Save**.

You can empower site users to assist in site administration by giving them a suitable role. Based on the permissions you grant via site-wide roles, you can select site users who could be granted the privilege.

1. Go to **My Workspace > Admin**.
2. Click **Roles** from the **Projects** menu. The existing site-wide roles are listed.
3. Click the role that you want to assign to the site users.

4. On the **Edit Site-wide Role Permissions** page, click the *Assigned Users* tab. All users who currently have the role are displayed.
5. Click **Add**.
6. In the **Find a User** window, select the site users you want to add, and move them from the **Found Users** list to the **Selected Users** list. Click **Add**.

> **NOTE:** You can search by full or partial user name or full name to find the desired site members.

7. Click **OK**.

The additional site administrators are now ready to act! Their names are added to the **Assigned Users** list.

To allow some CollabNet TeamForge users to use one or more CollabNet TeamForge tools across several projects, create site-wide roles with specific project permissions, minus site administrative permissions. Assign these site-wide roles to those who may need to access the project tools in any project.

For example, you may want a user to be able to use the “Tracker” across several projects. You don’t need to create and assign a role supporting the task individually across all projects. Just do it one time as a site-wide role and assign it to the user.

1. Go to **My Workspace > Admin**.
2. Click **Roles** from the **Projects** menu.
3. Click **Create**.
4. On the **Create Site-wide Role** page, write a name and description for the role. The role name is case-sensitive.
5. To prevent inheritance of the role into private projects, select the **PREVENT ACCESS** option.

> **NOTE:** Selecting the option to prevent role inheritance does not affect access to public and gated projects. On selecting Prevent access, the user may not be allowed to do project-permissions related tasks in private projects.

6. Click **Create**. The site-wide role is created. The **Edit Site-wide Role Permissions** page appears.

> **NOTE:** You can select the permissions for applications and resources available across all projects.

7. Select the required project permissions listed on the **ROLE PERMISSIONS** tab, to match the tasks you want the user with that role to perform.

> **TIP:** Select `Tracker-Create` permission if you want the user to be able to create new trackers.

The role is created. You can assign it to site members at any time.

A site must have one or more servers to handle source code repositories and users. The source code server can be the same server as the application server or a separate server.

When you set up a managed software configuration management (SCM) server, you enable users to create, manage, and share repositories through CollabNet TeamForge.

> **NOTE:** The ability to add integration servers depends on the value of the DISABLE_CREATE_INTEGRATION_SERVERS flag in the `site-options.conf` file. You can add new integration servers when the flag is set to its default value of FALSE.

You can integrate more than one source code server of a given type. For example, you can have two or more Subversion servers on your site. Consult a system administrator about the requirements for setting this up.

> **TIP:** If you use a source code solution other than Subversion or CVS, you can integrate it using the CollabNet TeamForge SOAP APIs. This enables you to exchange commit data with any SCM application. Consult your CollabNet TeamForge system administrator.

> **NOTE:** CVS servers that integrate with CollabNet TeamForge must use the native UNIX/Linux authentication method, and not external authentication mechanisms such as NIS, NIS+, Winbind, Active Directory, or LDAP. TeamForge creates and manipulates local system accounts using the default `useradd`, `usermod`, `groupadd`, `groupdel`, and `userdel` commands. It expects to find any accounts or groups it create in `/etc/passwd` and `/etc/group`.

1. Go to **My Workspace > Admin**.
2. Click **Projects > INTEGRATIONS**.
3. On the **SCM INTEGRATIONS** page, click **Create**.
4. On the **Create Integration** page, write a name and description for the integration.
5. Choose the type of SCM server you want.

   > **NOTE:** When you give a group access to a CVS or Wandisco Subversion repository, members of the group can view the repository but cannot do repository actions, such as commit and update. You must assign those permissions to users individually.

   > **NOTE:** The SCM Adapter option only works if you have created your own SCM integration using the CollabNet TeamForge SOAP APIs.

6. Supply the host name for the **Soap Service Host**. This is the network address of the machine on which the integrated service such as Subversion is running.

> **NOTE:** The default `localhost` will work only if the integration server is on the same server as the CollabNet TeamForge server.

7. Leave the default values in the **SOAP Service Port** field.
8. Specify whether users will use SSL to connect to their repositories.
9. Change the **Repository Root** value if you want to store the repository on your server in a different location. The repository root is the top-level directory under which all source code repositories reside.
10. Select the **Requires Approval** check box if you want an administrator to approve all repositories created on the server. By default, unmanaged servers require approval for all repositories, because repositories must be created and integrated manually.

> **NOTE:** By default, repositories created (or deleted) by site administrators and users with site-wide role (with Integrations, especially SCM INTEGRATIONS permission) need no approval.

11. Supply the URL by which your users will access the service. This will be of the form `http://<myscmserver.com>/integration/viewvc/viewvc.cgi`.

> **IMPORTANT:** If your system administrator has upgrade your site from SourceForge Enterprise Edition 4.4 or earlier, remove the port number in the SCM Viewer URL.

> **TIP:** If you are working with a CVS server that uses Pserver authentication, ask your system administrator for the right URL.

12. The **Use Internal Code Browser** option is selected by default to allow the users to access the TeamForge code browser for a Subversion or Git server. For more information about this feature, see Get the Code.

✓ Software requirements for using TeamForge code browser, if Subversion or Git is running on a separate server: Subversion Edge 5.1.0 and TeamForge - Git integration 8.4.4 or later.

✓ You need to specify the **SCM Viewer URL** (see step 11), the way you would for ViewVC or Gitweb. The ‘http://’ or ‘https://’, the host name and the port number (if any) need to be set correctly. As a minimum requirement, the URL should point to the root of the SCM http server and the domain name.

✓ If the SCM server is not the same as the TeamForge server, make sure you install SSL certificates or visit the URL for the site directly from your browser, and import and trust the certificate into the browser.

13. Click **Save**. CollabNet TeamForge attempts to validate the SCM viewer URL. If it cannot validate the URL, you can:
    - Correct it if you have entered it incorrectly.
    - Select **Save with errors** if the URL is different for an end user than it is for the CollabNet TeamForge server; for example, if you have a firewall in place.

If you are adding a managed source code server, it is now added. All projects can now establish repositories on the server. If you are adding an unmanager CVS server, all projects can now request repositories on the server. A CollabNet TeamForge administrator must create and integrate them manually.

> **NOTE:** Only CVS servers can run unmanaged.

It is recommended to change the scmviewer password after installing TeamForge.

Follow these steps to change the scmviewer password:

1. Stop TeamForge.

   ```
   teamforge stop
   ```
2. Create an encrypted password using the password_util.sh.
   ```
   /opt/collabnet/teamforge/runtime/scripts/password_util.sh -encrypt '<new_password_text>'
   ```
3. Set the encrypted password to the SCM_USER_ENCRYPTED_PASSWORD token in the `/opt/collabnet/teamforge/etc/site-options.conf` file.
4. Provision services.
   ```
   teamforge provision
   ```

Any time you upgrade your TeamForge site or a source control application, you must ensure that your users can still access their source code.

1. Click **Admin** in the TeamForge navigation bar.
2. Select **Projects > Integrations**.
3. For each source control service you are supporting, verify that the right paths are specified.
   - **SOAP service host** should be `localhost` or the host name of the server on which you just installed TeamForge.
   - **Repository base URL** should be the URL for the top level of your source code server (which may be the same as your application server). For example, `http://<myscmbox>/svn/repos`
   - **SCM Viewer URL** should be the URL for the ViewVC application on your source control server. For example, `http://<myscmbox>/integration/viewvc/viewvc.cgi`

> **NOTE:** If you want to turn on TeamForge code browser, specify the appropriate URL.

4. Select all your CVS integrations and click **Synchronize Permissions**. This updates the permissions on your code repositories so that users can access them from the new site.

> **NOTE:** By default, the DISABLE_CREATE_INTEGRATION_SERVERS token in the `site-options.conf` file is set to `false`, which allows users to create new external integrations. To restrict users from adding new integrations, set this token to `true` and recreate the runtime environment before making the site available to users.

When a user requests a source code repository on a source control server for which you have required approval, a TeamForge administrator must approve the request before the repository is created.

- When adding a source control server integration, you have the option to require TeamForge administrator approval for all repositories created on the server.
- When a user requests a source code repository on a managed SCM server, the repository is created automatically after it is approved by a TeamForge administrator.

> **IMPORTANT:** Before approving a source code repository request for an unmanaged SCM server, a TeamForge administrator must create and integrate the repository manually.

1. Click **Admin** in the TeamForge navigation bar.
2. Click **Integrations** from the **Projects** menu.
3. Click the **PENDING SCM INTEGRATIONS** tab.

> **NOTE:** Non-site administrators can now access the SCM Integrations tab if they have permission to manage SCM integrations.

4. From the list of pending SCM repository requests, select the SCM repositories that you want to approve.
5. Click **Approve** to approve the repository.

6. Click **Reject** to reject the project and remove it from the list.

   The person who requested the repository receives an email notification when the repository is approved or rejected. If you entered a comment, that also appears in the email notification.

When a user asks for access to an unmanaged SCM server (such as an unmanaged CVS server), an administrator must approve or reject the request.

- When a project administrator assigns a role that provides SCM access to a project member, a TeamForge administrator must manually create the user account on the unmanaged SCM server. Because user creation on unmanaged SCM servers is not managed by TeamForge, TeamForge cannot verify that the user account has been created. A TeamForge administrator must confirm that he or she has created the account.
- Requests for SCM access removal are also submitted for approval by the TeamForge administrator.

When you get a repository access request on an unmanaged SCM server, log on to the server and create the requested user account on the unmanaged SCM server.

1. Log on to the TeamForge application server.
2. Go to **My Workspace > Admin**.
3. Click **Integrations** from the **Projects** menu.
4. Click the **SCM ACCESS REQUESTS** tab.
5. As the user account is created, select the repository access request from the **Repository Access Requests** section and click **Approve**. You may also click **Reject** to reject the repository access request.

   The user receives an email notification that the user account has been created and that the repository access request has been approved or rejected.

SCM system maintenance requests (such as a repository delete request) must be approved (or rejected) by an administrator.

1. Go to **My Workspace > Admin**.
2. Click **Integrations** from the **Projects** menu.
3. Click the **SCM ACCESS REQUESTS** tab.
4. Select the repository deletion request from the **System Maintenance Requests** section and click **Approve**. You can click **Reject** to reject the repository deletion request.

   The user receives an email notification that the repository deletion request has been approved or rejected.

You can move or reconfigure a source control server without having to reintegrate the server into TeamForge.

> **TIP:** If you edit or lose the permissions on your SCM server, use the **Synchronize Permissions** button on the **SCM Integrations** page to recreate the correct permissions on your SCM server from TeamForge.

1. Click **Admin** in the TeamForge navigation bar.

2. Click **Integrations** from the **Projects** menu.
3. On the **SCM INTEGRATIONS** tab, click the name of the SCM integration you want to edit.
4. On the **Edit Integration** page, make the changes you need and click **Save**.

When the existing code base for an application may need to be managed by a different team or project, you can move the source code repository from the first project into the other one.

1. Open the project from which you want to move a source code repository.
2. Click **SOURCE CODE** from the **Project Home** menu.
3. Select the repository you want to move and click **Cut**.
4. Open the project into which you want to move the repository.
5. Click **SOURCE CODE** from the **Project Home** menu.
6. Click **Paste**.

When a Subversion Edge server has been converted to a SCM Integration server in TeamForge, you can log into its management console from within TeamForge.

1. Go to **My Workspace > Admin**.
2. Click **Integrations** from the **Projects** menu.
3. In the list of servers on the **SCM INTEGRATIONS** page, click the **Open Console** link for the Subversion Edge server you want to connect to.

   > **NOTE:** Only Subversion Edge servers have this link.

4. The login page for the server's management console appears. You can log in using your TeamForge administrator credentials and view statistics such as network throughput and disk space usage for the server.

A replica server in TeamForge is a Subversion Edge server that replicates the content of an existing core SCM integration server.

A replica server in TeamForge is a Subversion Edge server that replicates the content of an existing core SCM integration server.

Approve a Replica Server Request

When there is a request for a replica of a core SCM integration server, a TeamForge administrator must approve the request before the replica is created.

Replica requests from a TeamForge admin user or site administrator are automatically approved. Replicas requested by other users need approval by a TeamForge administrator.

1. Go to **My Workspace > Admin**.

2. Click **Integrations** from the **Projects** menu.
3. On the **SCM INTEGRATIONS** tab, click the **Pending SCM Replicas** tab.
4. From the list of pending SCM replica requests, select the SCM replicas that you want to approve.
5. Click **Approve** to approve the replica.
6. Click **Reject** to reject the replica and remove it from the list.

   When a replica is approved, it is listed in the SCM INTEGRATIONS tab beneath the master SVN server.

Edit Replica Settings

As a TeamForge site administrator, you can configure replica settings for the polling frequency of the master, and repository initialization and synchronization events.

Replication events, such as creating new repositories and synching commits, are stored in a queue on the TeamForge Application Server. The replica server polls the TeamForge Application Server for new events and then processes those events on the replica server.

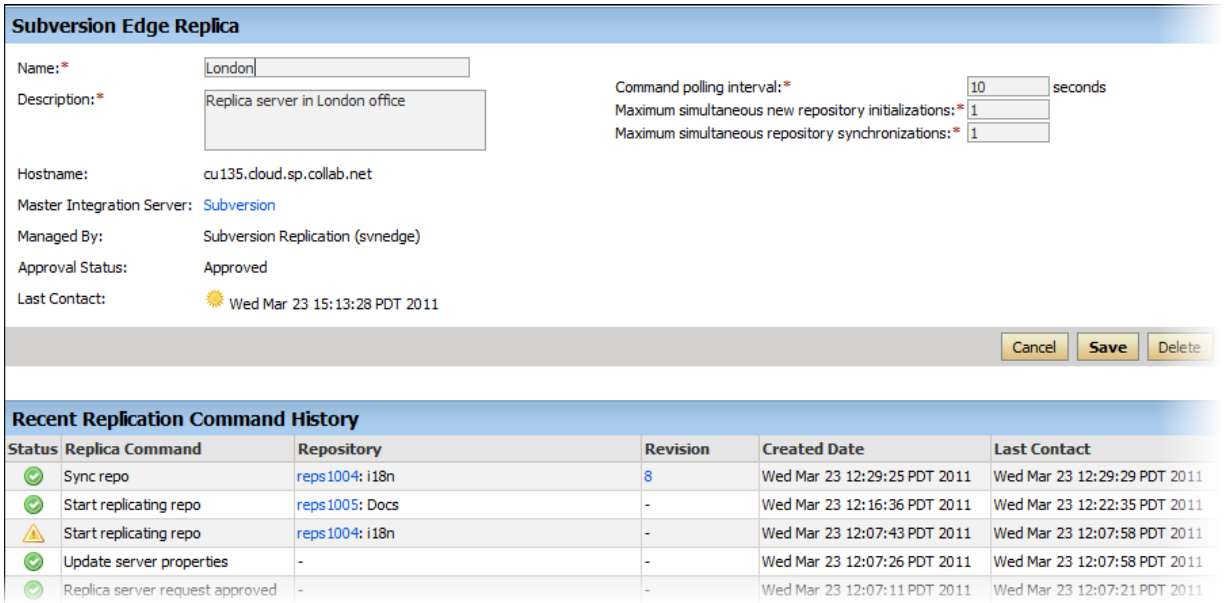
These events are divided into two separate pools:

- New repository initializations–this includes creating the repository and performing the initial synchronization of the content.
- All other events

When existing repositories are selected for replication, this can take a long time. It could take many hours or even days to fully replicate an entire repository across a WAN. These big events are processed in their own thread pool, so that other repositories which are already synchronized don’t have to wait in line for them to finish.

For each pool, you can define how many simultaneous events will be processed. The higher the number, the greater the potential load on both the TeamForge replica server and the Subversion master. However, this can also decrease the wait time for a given commit to appear on the replica server.

1. Go to **My Workspace > Admin**.
2. Click **Integrations** from the **Projects** menu.
3. On the **SCM INTEGRATIONS** tab, click the name of the Subversion Edge replica you want to edit.
4. The **Edit System** page for the replica appears. Here’s an example:



Subversion Edge Replica

Name:*	London
Description:*	Replica server in London office
Command polling interval:*	10 seconds
Maximum simultaneous new repository initializations:*	1
Maximum simultaneous repository synchronizations:*	1
Hostname:	cu135.cloud.sp.collab.net
Master Integration Server:	Subversion
Managed By:	Subversion Replication (svnedge)
Approval Status:	Approved
Last Contact:	Wed Mar 23 15:13:28 PDT 2011

Recent Replication Command History

Status	Replica Command	Repository	Revision	Created Date	Last Contact
✓	Sync repo	reps1004: i18n	8	Wed Mar 23 12:29:25 PDT 2011	Wed Mar 23 12:29:29 PDT 2011
✓	Start replicating repo	reps1005: Docs	-	Wed Mar 23 12:16:36 PDT 2011	Wed Mar 23 12:22:35 PDT 2011
⚠	Start replicating repo	reps1004: i18n	-	Wed Mar 23 12:07:43 PDT 2011	Wed Mar 23 12:07:58 PDT 2011
✓	Update server properties	-	-	Wed Mar 23 12:07:26 PDT 2011	Wed Mar 23 12:07:58 PDT 2011
✓	Replica server request approved	-	-	Wed Mar 23 12:07:11 PDT 2011	Wed Mar 23 12:07:21 PDT 2011

5. Change the replica name or description if required.

> **TIP:** Including the geographic location would help users select a nearby replica.

6. Set the **Command polling interval** to define how frequently the replica polls the master looking for new events.

   The replica will process all new events when it polls. The default value for this setting is 60 seconds, but it can range from 5 to 1000000 seconds.

7. Set the **Maximum simultaneous new repository creations** to a low value.

   New repository initializations can take a long time and generate a lot of load. So you wouldn’t want to allow too many of them to run at once. This value can range from 1 to 100, but we suggest you keep it at 3 or less.

8. Set the **Maximum simultaneous repository synchronizations** taking into account how many repositories you will be replicating and how many you think are likely to have commits occurring within the polling interval.

   This value can range from 1 to 100. You may want to set this higher than the previous field, but we suggest you keep it at 10 or less. There’s no reason to enter too high a number because you are merely specifying how many synchs can run at the exact same time – and it never runs more than one per repository.

9. When you've made your changes, click **Save**.

Remove a Replica

When you remove a replica from TeamForge, it is restored to a Subversion Edge server in standalone mode.

1. Go to **My Workspace > Admin**.
2. Click **Integrations** from the **Projects** menu.
3. In the **Edit System** page for the replica, click **Delete**.

   The replica is removed from TeamForge. The repositories that existed on the replica are deleted.

Here's how you can integrate an unmanaged CVS repository with TeamForge.

After creating the repository, you must modify two triggers, `verifymsg` and `loginfo`, to complete the integration with TeamForge.

The triggers must be modified manually, because much of the information in the triggers is specific to your TeamForge installation.

Each trigger must be modified to run a `.java` program. Here's an annotated example of how to modify and install the trigger files.

✓ You must create the CVS repository before you modify and install the trigger files.

✓ CVS servers that integrate with TeamForge must use the native UNIX/Linux authentication method, and not external authentication mechanisms such as NIS, NIS+, Winbind, Active Directory, or LDAP.

✓ TeamForge creates and manipulates local system accounts using the default `useradd`, `usermod`, `groupadd`, `groupdel`, and `userdel` commands. It expects to find any accounts or groups it created in `/etc/passwd` and `/etc/group`.

Create a `loginfo` Trigger File

Here's how to create the `loginfo` trigger file.

1. Check out the module `CVSROOT` from the CVS repository.

   The `CVSROOT` module is automatically created in all new CVS repositories, and contains the `verifymsg` and `loginfo` files. When created, the files contain only comments.

2. Begin modifying the file by inserting `ALL` followed by a tab. This enables TeamForge to see all modules in the repository.
3. Modify the path to your Java home directory as in this example and add it to the file, followed by a single space.

Example	Modification
`/usr/local/java/j2sdk1.4.1_01/bin/java -cp`	Replace `/usr/local/java/j2sdk1.4.1_01/` with the directory in which you have installed Java.

4. Modify the paths to the `.jar` files as shown here, then add them to the file separated by colons.

Examples	Modifications

`/yourdirectory/sourceforge_home/lib/saaj.jar`
`/yourdirectory/sourceforge_home/lib/axis-1.1rc1-va.jar`
`/yourdirectory/sourceforge_home/lib/saturn.jar`
`/yourdirectory/sourceforge_home/lib/jaxrpc.jar`
`/yourdirectory/sourceforge_home/lib/externalintegration.jar`
`/yourdirectory/sourceforge_home/lib/commons-discovery.jar`
`/yourdirectory/sourceforge_home/lib/commons-logging.jar`
`/yourdirectory/sourceforge_home/lib/log4j-1.2.8.jar`
	Replace `/yourdirectory/` with the directory in which you have installed TeamForge.

5. Modify the parameters as follows:

Example	Modification
-Dintegration.name=exsy1001	Replace `exsy1001` with the ID of your SCM server integration with TeamForge. **NOTE:** To find the ID, click **Admin > Integrations > Integration Name**. The ID appears at the end of the URL.

-Dcvs.cvsroot=$CVSROOT	No change
-Dappserver.url=http://localhost:8080/ce-soap60/services/ScmListener	Replace `localhost` with the name of your machine running TeamForge. **NOTE:** If needed, replace 8080 with your SOAP server port.
-Dcvs.username=$USER	No change
-Dlog4j.configuration=file:/yourdirectory/sourceforge_home/etc/externalintegration/log4j.xml	Replace `/yourdirectory/` with the directory in which you have installed TeamForge.
-Dexternalintegration.triggers.log.dir=/tmp	No change

6. Add the following Java class containing the body of the trigger.
```
com.vasoftware.sf.externalintegration.triggers.cvstriggers.LogInfo
                              %{sVv}
```
7. Check your file against the completed example shown on Completed loginfo trigger file.
8. When you are finished, check the modified file back into your CVS repository. If you have not already done so, repeat the process for the verifymsg trigger file.

> **NOTE:** If you make an error in the file, commits made to the repository will not fail, but you will receive error messages.

Create a Modified `verifymsg` Trigger File

Here’s how to modify the `verifymsg` trigger file.

1. Check out the module `CVSROOT` from the CVS repository.

   The `CVSROOT` module is automatically created in all new CVS repositories, and contains the `verifymsg` and `loginfo` files. When created, the files contain only comments.

2. Begin modifying the file by inserting `.*` followed by a tab. This enables TeamForge to see all modules in the repository.

3. Modify the path to your Java home directory as indicated below. Add it to the file, followed by a single space.

Example	Modification
/usr/local/java/j2sdk1.4.1_01/bin/ java -cp	Replace /usr/local/java/j2sdk1.4.1_01/ with the directory in which you have installed Java.

4. Modify the paths to the .jar files as shown here, then add them to the file separated by colons.

Examples	Modifications
/yourdirectory/sourceforge_home/lib/saaj.jar /yourdirectory/sourceforge_home/lib/axis-1.1rc1-va.jar /yourdirectory/sourceforge_home/lib/saturn.jar /yourdirectory/sourceforge_home/lib/jaxrpc.jar `/yourdirectory/sourceforge_home/lib/externalintegration.jar /yourdirectory/sourceforge_home/lib/commons-discovery.jar /yourdirectory/sourceforge_home/lib/commons-logging.jar /yourdirectory/sourceforge_home/lib/log4j-1.2.8.jar	Replace /yourdirectory/ with the directory in which you have installed TeamForge.

5. Modify the parameters as follows:

Example	Modification
-Dintegration.name=exsy1001	Replace exsy1001 with the ID of your SCM server integration with TeamForge. **NOTE:** To find the ID, click **Admin > Integrations > Integration Name**. The ID appears at the end of the URL.
-Dcvs.cvsroot=$CVSROOT	No change

-Dappserver.url=http://localhost:8080/ce-soap60/services/ScmListener	Replace `localhost` with the name of your machine running TeamForge. **NOTE:** If needed, replace 8080 with your SOAP server port.
-Dcvs.username=$USER	No change
-Dlog4j.configuration=file:/yourdirectory/sourceforge_home/etc/externalintegration/log4j.xml	Replace `/yourdirectory/` with the directory in which you have installed TeamForge.
-Dexternalintegration.triggers.log.dir=/tmp	No change

6. Add the following Java class containing the body of the trigger: `com.vasoftware.sf.externalintegration.triggers.cvstriggers.VerifyMessage`
7. Check your file against the completed example shown on Completed verifymsg trigger file.
8. Check in the modified file back into your CVS repository.
9. If you have not already done so, repeat the process for the `loginfo` trigger file.

> **NOTE:** If you make an error in the file and check it back into your CVS repository, all commits made to the repository will fail. To correct the file, you cannot simply check it out and fix it. You must fix it on the repository. After fixing the file on the repository, you must then check it out, fix it again, then check it back in.

You might want to create roles that projects across the site can use with minimum effort and maintenance. Using global project roles is an easy way of enforcing role-based similarities and removing role duplication across projects. You can suggest to the project administrators to use common global project roles while assigning project tasks in CollabNet TeamForge, instead of creating and managing several similar roles for their individual projects.

A global project role is a ready-to-use role available in all projects. Only site administrators or restricted site administrators can create and manage a ready-to-use role.

As a project administrator, you can use global project roles provided by the site administrators instead of creating and managing roles tailored to your projects.

> **NOTE:** You can use the ready-to-use roles to set up your team faster and with little fuss. However, you may not be able to edit the ready-to-use roles.

Before you create a role in your project, it is a good idea to check all the available ready-to-use roles. You are likely to get ones that grant the desired set of permissions.

Global project roles serve a different purpose from that of a site-wide role. The site-wide roles enable site administrators to create restricted site administrators for providing assitance in site management. Besides that, site-wide roles can be used to grant tool/application access across the site to a user.

To help project managers get their project members set up quickly, provide ready-made project roles that any project on your site can use.

> **NOTE:** You need the `Role-Create` permission to create global project roles. All site administrators and some restricted site administrators have this permission.

1. Go to **My Workspace > Admin**.
2. Click **Roles** from the **Projects** menu.
3. Click the **GLOBAL PROJECT ROLES** tab. All the existing Global Project Roles are listed here. It is a good idea to check this list before you create another role.

   > **NOTE:** You can suggest that the project administrators check the **Project Admin > Permissions > Roles > View: Global Project Roles** list before creating any new roles in their projects.

4. Click **Create**.
5. On the **Create Global Project Role** page, write a name and description for the role. The role name is case-sensitive.

   > **TIP:** Remember that the role name can not be the same as a site-wide role name.

6. To allow inheritance of the role's permissions into private sub-projects, clear the **PREVENT INHERITANCE** check box.
7. To allow the project members to be able to request this role, select **PROJECT MEMBERS CAN REQUEST THIS ROLE**. Project members can submit requests for `Available upon Request` roles. For a project, the project administrators can set an `Available upon Request` role to be automatically granted to the project member requesting it.
8. Click **Create**. The new global project role is created. The **Edit global project role permissions** page appears.
9. Select the application permissions that are relevant to the role, from those listed on the **ROLE PERMISSIONS** tab.

   > **TIP:** You may want to restrain providing project or application administration permissions, until required.

The role is created. The project administrators can assign it to their project members any time.

You may need to add or remove certain permissions from an existing global project role to assign new tasks or change the access permissions given via the role.

> **NOTE:** Only site administrators or restricted site administrators with `Role-Edit` permission can edit global project roles.

1. Go to **My Workspace > Admin**.
2. Click **Roles** from the **Projects** menu.
3. Click the **GLOBAL PROJECT ROLES** tab. All the existing global project roles are listed here.
4. Select the role that you want to edit and click **Edit** or just click the hyperlinked role name. The **Edit global project role permissions** page appears.
5. Click **Edit** to make changes to the role details.
6. Modify the **Role Name** or **Description**, if required. The role name is case-sensitive and must not be the same as a site-wide role.
7. Change the inheritance setting to prevent or allow inheritance of the role's permissions into private sub-projects.
8. To make the role requestable or non-requestable, change the Project members can request this role setting. Project members can submit requests for `Available upon Request` roles. For a project, the project administrators can set a `Available upon Request` role to be automatically granted to the project member requesting it.
9. Click **Update**. The global project role is modified.
10. Select the application permissions that are relevant to the role, from those listed on the **ROLE PERMISSIONS** tab and click **Save**.

    > **TIP:** You may want to restrain removing project or application administration permissions as the change impacts existing users too.

    The role is modified.

If you no longer need a global project role, you should delete it. On deleting a global project role, all the user associations in the projects are removed.

> **NOTE:** Only site administrators or restricted site administrators with `Role-Delete` permission can delete global project roles.

1. Go to **My Workspace > Admin**.
2. Click **Roles** from the **Projects** menu.
3. Click the **GLOBAL PROJECT ROLES** tab. All the existing global project roles are listed here.

4. Select the role that you want to delete and click **Delete**. A confirmation message appears.

5. Click **OK** to continue with deleting the selected role.

   The selected global project role is deleted and all its associations are removed.

TeamForge administrators can do a variety of things to help projects on the site be successful.

Create a New Project

TeamForge administrators can create new projects without having to submit them for approval.

> **NOTE:** When a TeamForge administrator creates a new project, he or she is not made a member of the project, and the `Founder Project Admin` role is not created. To designate a project administrator, you must add the user to the project, then create and assign a project administrator role manually.

1. Go to **My Workspace > Admin**.
2. In the list of TeamForge projects, click **Create Project**.
3. On the **Create Project** page, provide a name for the project. This is the name that will appear in all project lists and on the project home page.
4. Enter a URL name for the project, if appropriate. This is the name that will appear in the project's URL.
5. Write a description of the project.

6. Select a project template. A project template is used to pre-populate new projects with the structure and configuration of an existing project. If you do not want to use a project template, choose **None**.

   > **NOTE:** If you create a project from a template that contains an integrated application, you may have to provide some information specific to the integrated application. For example, for Project Tracker you must set a new artifact prefix that is different from the prefix in the template.

7. Click **Create**.

   The project is created.

Approve a New Project

Any registered TeamForge user can request a new project. A new project is activated only after a site administrator approves it.

Before approving a new project, you have the option to review the project details.

1. Go to **My Workspace > Admin**.
2. Click the **PENDING PROJECTS** tab.
3. Select the projects you want to approve from the pending projects list.
4. Click **Approve** to approve the project and move it to the **All Projects** page with status Active.

5. Click **Reject** to reject the project and remove it from the list.

   The project requester receives an email notification when the project is approved or rejected. If you entered a comment, that also appears in the email notification.

Rename a Project

As the focus of a project shifts, its name or description can become obsolete. A site administrator can update the name or description to help keep the project current.

1. Go to **My Workspace > Admin**.
2. In the TeamForge project list, click the project you want to edit.
3. Click **PROJECT ADMIN** from the **Project Home** menu.
4. On the **Project Admin Menu**, click **Project Settings** and make the changes you need.
5. Click **Save**.

Delete a Project

If you no longer need a project or any of the data in it, you should delete it.

> **WARNING:** Delete a project only if you are sure that you no longer need any of the data within it. Move any items that you want to save.

- Deleting a project deletes all of the data within it, with the exception of source code data, which is maintained separately from the rest of the site's content.
- You can delete a parent project only when its members, user groups and roles are not in use in any other project.
- When you delete a parent project, the direct subprojects are moved one level up; that is, under the immediate parent of the deleted project.
- If the deleted project has no parent, its subprojects become root-level projects.

1. Go to **My Workspace > Admin**.

2. From the TeamForge project list, choose the project that you want to delete and click **Delete**.

The project is deleted.

Lock or Unlock a Project

To ensure that no changes occur in a project while you are collating or migrating project data, lock the project. You must have project administration permissions or be a site administrator to lock or unlock a project.

1. To lock or unlock a project in TeamForge, go to Project Settings and lock/unlock the project.

   > **NOTE:** A locked project does not allow any member (including project administrators and site administrators) to make any changes to the project. Besides that, a locked project can not be set as the parent project for any other project and tasks like adding, editing or deleting integrated applications are also not allowed.

2. Click **PROJECT ADMIN** from the **Project Home** menu.

   The project is locked or unlocked as desired.

   The lock icon 🔒 Locked appears on all the project pages while the project is locked.

   > **NOTE:** If a locked project has an integrated application, for example, project tracker, all the project tracker pages are also non-editable while the project is locked. The user who has access permissions for the integrated application can only view the pages.

To help users navigate your site, help them sort projects into categories that make sense.

To help users navigate your site, help them sort projects into categories that make sense.

Add a Project Category

When you set up project categories for your site, project administrators can use this taxonomy to organize their projects.

You can create any number of top-level categories and any number of sub-category levels.

1. Go to **My Workspace > Admin**.
2. Click **CATEGORIES** from the **Projects** menu.
3. In the **Project Categories** tree, find the location where you want to create the new category.
   - Highlighting **Project Categories** creates a new top-level category.

- Highlighting any category creates a sub-category beneath it.

4. Click **New**.
5. In the **Create Category** page, write a name and description for the category.
6. Click **Save**.

   The category is created. It appears in the **Project Categories** navigation tree, and is available for use by all project administrators when categorizing their projects.

Edit a Project Category

A project category's membership and function may change over time. If it does, you can update the category's name or description.

1. Go to **My Workspace > Admin**.
2. On the site administration navigation bar, click **CATEGORIES**. The **Project Categories** tree displays the hierarchy of existing categories.
3. In the **Project Categories** tree, find the category that you want to edit.
4. Make the changes you need and click **Update**.

Move a Project Category

You can reorganize projects by moving a project category to another place in the project category hierarchy.

You can move a project category in the following ways:

- From a top-level category to a sub-category.
- From a sub-category to a top-level category.
- From a sub-category to another sub-category.

When you move a project category, any sub-categories that it contains are also moved to the destination category.

1. Go to **My Workspace > Admin**.
2. Click **CATEGORIES** from the **Projects** menu.
3. On the **Edit Category** page, in the **Project Categories** section, click the project category you want to move.
4. On the **Edit** menu, click **Cut**.
5. Find the location to which you want to move the selected project category. You can move a project category either to the root category or into any other project category.

6. Select **Paste** from the **Edit** menu.

   The project category is now moved to the selected destination.

Delete a Project Category

If you no longer need a project category, you should delete it.

When you delete a project category, all of its sub-categories are also deleted.

1. Go to **My Workspace > Admin**.
2. Click **CATEGORIES** from the **Projects** menu. The **Project Categories** tree displays the hierarchy of existing categories.
3. Using the document tree, find the project category that you want to delete.
4. Choose **Delete** from the **EDIT** menu.

   The project category and all of its sub-categories are deleted.

Stop Using Project Categories

If you do not need to sort projects into categories, remove the ability to do so on your site.

By default, project categorization is disabled for new TeamForge installations.

> **NOTE:** Disabling project categorization does not delete categories you have already set up.

1. Go to **My Workspace > Admin**.
2. Click **CATEGORIES** from the **Projects** menu.
3. On the **Edit Category** page, select **Disabled** for **SITE-WIDE CATEGORIZATION** and click **Update**.

   Project categorization is disabled for your TeamForge site.

You can coordinate work among multiple projects; enable the user, user group and role inheritance by adding a common parent project to several projects.

A parent project is the base from which a subproject’s members, user groups and roles, with their corresponding permissions, are derived. A subproject can inherit project members, user groups and roles from its parent project.

You can create a parent project to track and manage several smaller assignments as subprojects.

When you define users and roles with specific permissions in a project, those users and roles are passed down to any subprojects that belong to that project. This helps you avoid the repeated effort of defining users, user groups and roles across projects.

✓ A subproject can have only one parent. You can change or remove that at any time.

✓ You must be a project administrator or site administrator to add, edit or remove the parent projects.

1. Click **Project Admin** from the **Project Home** menu.
2. On the **Project Settings** page, click **Add Parent**.
3. Choose a parent project that it makes sense for your project to belong to, and click **Update**.

   > **NOTE:** You cannot add a parent project for the `Look` project as it is a special project in itself.

When a subproject grows beyond its original scope, you may want to make it stand-alone project or move it to a different project hierarchy. By removing the association with a parent project, you can manage the subproject as a separate project.

> **NOTE:** Only one parent project can be selected for a subproject. However, the parent project can be changed or removed, as required.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Settings** page, click **Edit Parent**.
3. Change the parent project as required.

   You can be a project administrator or a site administrator to change or remove a parent project. As a project administrator, you can remove or change a parent project only if you have administrator permissions for both the projects that are being linked.

   > **NOTE:** A parent project can be removed or changed only when its members, user groups and roles are not in use in any other project. In other words, you can not remove/change a parent project while its members, user groups or roles are in use in any other project.

4. On the **Choose a Parent Project** page, select the desired parent project and click **Update**.

   > **NOTE:** If the project hierarchy exists, the project and its subprojects are moved only under the project from which members, user groups or roles are inherited. If project hierarchy does not exist and no inheritance is in use, the project is made a `Root` project.

5. Click **Remove Parent** in the **Project Settings** page to remove the association with parent project.

6. On the pop-up message box, click **OK**, if you wish to make the project a `Root` project.

   The project hierarchy is changed or removed in accordance with role based access control and inheritance rules.

To begin controlling multiple projects at one go, create a project group.

To manage two or more independent projects, create a project group in TeamForge. Similar to projects, a project group provides the platform for sharing project members, roles and permissions across a group of projects. With some well planned settings, you can manage several of your projects and also effectively control the project members accessing each project.

To begin managing your several projects together, create a project group in TeamForge.

> **NOTE:** When you create a project group, you are granted administration rights for the group and will be able to perform actions such as adding or removing a project from the group. You will be listed as an administrator in the group's Project Group Details. (Site administrators are exempt from this, since they can perform all actions on the site without requiring specific roles).

1. Go to **My Workspace > Admin**.
2. Click **PROJECT GROUPS** from the **Projects** menu. The existing project groups are listed here.
3. To create a new project group, click **Create Project** Group.

4. On the **Create Project Group** page, write a name and description for the project group. The project group name must be unique, however, it can be the same as any of the projects.

   The project group is created.

   In your Project Group page, you can add projects, add project group members or specify roles for the project group.

   > **TIP:** You can always come back to this page later to specify projects, users or roles that affect your project group.

You can make changes to a project group to keep it updated with the various projects you may be managing with that in CollabNet TeamForge. You must have the administrator permissions for project groups to make any modifications.

> **TIP:** You could be either a site administrator or a project administrator, but you must have the project groups administration permissions to manage projects as a group.

1. Go to **My Workspace > Admin**.
2. Click **PROJECT GROUPS** from the **Projects** menu. The existing project groups are listed here.
3. Click the project group that you want to modify. The **Project Group Details** page appears.
4. Click **Edit** and make the changes as required.

   You can update the project group name, description as well as the administrators.

5. To add one or more administrators for the project group, click the Search icon next to `Administrators`.
6. On the **Find a User** page, select the required administrators, click **Add** and click **OK**.

   The project group administrators are listed on the **Project Group Details** page.

   The project group is modified.

   From the **Project Group Details** page, you can add the projects, add project group members or specify the desired roles for the project group.

If the projects being managed under a project group have achieved their targets you may not need the project group anymore. You can delete a project group in CollabNet TeamForge. You must have the administrator permissions for project groups.

> **TIP:** You could be either a site administrator or a project administrator, but you must have the project groups administration permissions to manage projects as a group.

1. Go to **My Workspace > Admin**.
2. Click **PROJECT GROUPS** from the **Projects** menu. The existing project groups are listed here.
3. To delete a project group, select the project group from the list and click **Delete**. You may get a warning message as the projects being managed under the group may have active role assignments. Click **OK** on the message to proceed with deleting the project group. The project group is deleted.

   > **TIP:** The relationship between projects and the project group is broken and you can no longer manage projects using the deleted project group.

You may be interested to start using the project group that brings your projects together. Add your projects to your project group as the initial step.

> **TIP:** You could be either a site administrator or a project administrator, but you must have the project groups administration permissions to manage projects as a group.

If you have just created your project group, you might already be on the Project Group Details page. Skip the first three steps in that case.

1. Go to **My Workspace > Admin**.
2. Click **PROJECT GROUPS** from the **Projects** menu. The existing project groups are listed here.
3. Click your project group. The **Project Group Details** page appears.
4. On the **Project Group Details** page, click **Add** to associate your projects with the project group. A list of projects appears.

   > **TIP:** You can only add the projects for which you are the project administrator.

5. From the **PROJECT** list, select the projects and click **Add**. The selected projects are added to the project group.

6. If you add an irrelevant project to the group, you can select it from the **Project Group Details** page and click **Remove** to remove it from the project group.

   > **TIP:** You can always come back to this page later to add more projects, specify users or roles that affect your project group.

Your project group needs to be set up with project group members to facilitate any administrative tasks that you may want to do.

> **TIP:** You could be either a site administrator or a project administrator, but you must have the project groups administration permissions to manage projects as a group.

If you have just created your project group, you might already be on the Project Group Details page. Skip the first three steps in that case.

1. Go to **My Workspace > Admin**.
2. Click **PROJECT GROUPS** from the **Projects** menu. The existing project groups are listed here.
3. Click your project group. The **Project Group Details** page appears.
4. From the left navigation pane, click the **User Membership** link to add users to the project group.
5. On the **PROJECT GROUP MEMBERSHIP** tab, click **Add**.
6. On the **Add User** page, find the users you want by one of these methods:
   - Under **Search for Users**, filter the list of site users eligible to join this project group. You can filter by full or partial name or user name.

     > **NOTE:** Search text is not case-sensitive.

- Browse the list of registered users on the site. Sort them by name, user name, email address or membership status.

> **NOTE:** If a site has a great many users, you must filter them first to narrow down the list. This helps avoid slowing down the system.

7. Select the users you want to add.
8. Under **Assign Roles (Optional)**, select the roles you want the users to have. You can select any available global project role or role created just for this project.

> **TIP:** If you prefer, you can skip this step and assign roles later on.

9. Save your changes.
10. Click **Save** to return to the **Project Group Membership** page.
11. Click **Save and Add More** to keep adding users. The selected users are granted membership to the project group.

    If you add a user who may not need to be a member of the project group, you can select the user from the **Project Group Membership** page and click **Remove** to remove the user's membership.

As a project group administrator, you know that the key to any user's access to the project group and the projects belonging to the group, is in the role you assign to the user. You can do almost all the role-related activities with project groups that you could do with individual projects.

You can create site-wide roles and assign the users to those roles and then bring them into your project group. You can create roles specifically for your project group and assign users to those or use global project roles, as required. The permissions inherited through the assigned roles are subject to the “Prevent Inheritance” option setting done while creating the roles.

Give a Role to Project Group Members

A role can be assigned to many project group users at once.

While the user-role matrix provides a convenient way to add project group members to a role, it can become unwieldy if the project has a large number of users or roles. When that is the case, try assigning roles to multiple project group users at one time.

1. Go to **My Workspace > Admin**.
2. Click **PROJECT GROUPS** from the **Projects** menu. The existing project groups are listed here.
3. Click your project group. The **Project Group Details** page appears.

4. From the left navigation pane, click the **Permissions** link to specify user roles applicable to the project group.
5. On the **Roles** tab, click the **Global Project Roles** or **Roles Created For a Project** option of **View**, to display existing global roles or roles created just for this project.
6. Click the name of the role that you want to assign to project group members.
7. On the **Edit Role** page, click the **Assigned Users** tab. The **Assigned Users** page shows all users who currently have the role.
8. Click **Add**.
9. In the **Find a User** window, select the project group members you want to add, and move them from the **Found Users** list to the **Selected Users** list.
10. Click **Add** to move selected users.
11. Click **Add All** to move all users.

    > **NOTE:** You can search by full or partial user name or full name to find the desired project group members.

12. Click **OK**.

    The project group members are now assigned the role.

Give Roles to a Project Group Member

A project group member can have any number of roles. As project group administrator, you must assign each project group member's roles with care. The roles would impact not just an individual project, but would also grant same permissions across the projects in a project group.

Permissions are cumulative. The project group member has all of the access permissions allowed by all of the assigned roles, plus any permissions that may have been assigned globally using application permissions.

1. Go to **My Workspace > Admin**.
2. Click **PROJECT GROUPS** from the **Projects** menu. The existing project groups are listed here.
3. Click your project group. The **Project Group Details** page appears.
4. From the left navigation pane, click the **Permissions** link to specify user roles applicable to the project group.
5. Click the **User-Role Matrix** tab. Observe the users listed on the left and all the available roles (global and direct) on the right. Users can be assigned global roles and roles created just for this project.
6. Select roles for each project member.
7. Click **Save**.

The roles are now assigned to each project group member.

Assign roles in multiple projects to a user group

Project managers can assign a role to multiple users at once by assigning the role to a user group that contains all those users. As a site administrator, you can do the same thing across multiple projects, by treating the projects as part of a project group.

1. Go to **My Workspace > Admin**.
2. Click **PROJECT GROUPS** from the **Projects** menu. The existing project groups are listed here.
3. On the **Project Group** page, click **Permissions**.
4. On the **User Group-Role Matrix** tab, add the user group you want, then select the roles you want to assign to that user group.
5. Click **Finish** or **Finish and Add More**.

> **IMPORTANT:** When you give a group access to a CVS or Wandisco Subversion repository, members of the group can view the repository but cannot do repository actions, such as commit and update. You must assign those permissions to users individually.

Assign User Groups to a Role

To manage permissions for a lot of groups or roles at once, try assigning user groups to roles.

1. Go to **My Workspace > Admin**.
2. Click **PROJECT GROUPS** from the **Projects** menu. The existing project groups are listed here.
3. Click your project group. The **Project Group Details** page appears.
4. From the left navigation pane, click the **Permissions** link to specify user roles applicable to the project group.
5. On the **Roles** tab, to display existing global or direct roles, click the global project roles or **Direct Roles** option of **View**. You can assign the project group users or user groups to a global project role or a role created just for this project.
6. Click the name of the role that you want to assign to the user group members in the project group.
7. On the **Edit Role** page, click the **Assigned User Groups** tab. The **Assigned User Groups** table shows all user groups who currently have the role.
8. Click **Add**.
9. Type some of the group’s name in the Name (search) box and click **Find.**
10. From the **Add User Group to Role** table, select the user groups that you want to add, and click **Finish**.

> **IMPORTANT:** When you give a group access to a CVS or Wandisco Subversion repository, members of the group can view the repository but cannot do repository actions, such as commit and update. You must assign those permissions to users individually.

Create a Role in a Project Group

A role defines the applications that project group members with that role can use, and the specific things project group members can do in each application.

Any project groups administrator can create and assign a role. It is a good idea to check the existing global project roles before creating any new role for a project group. Note: Any existing site-wide or global project role can be associated with a project group. It is advisable to check the permissions granted via a role before assigning it to users or user groups in a project group as it would impact more than a single project.

1. Go to **My Workspace > Admin**.
2. Click **PROJECT GROUPS** from the **Projects** menu. The existing project groups are listed here.
3. Click your project group. The **Project Group Details** page appears.
4. From the left navigation pane, click the **Permissions** link to specify user roles applicable to the project group.
5. On the **Roles** tab, click **View: Roles Created For this Project Group**. You can view global project roles by selecting **View: Global Project Roles** before creating a new role for this project.
6. Click **Create**.
7. On the **Create Role** page, write a name and description for the role.
8. To allow the inheritance of the role into private subprojects, clear the **Prevent Inheritance** check box.

   > **NOTE:** By default, the role inheritance into private subprojects is prevented. For example, you may not want administrator roles to be inherited in subprojects, until required. Selecting the option to prevent role inheritance does not affect public and gated projects.

9. Click **Create**. The role is created. The **Edit Role** page appears.
10. For each application listed on the **Role Permissions** page, select the permissions and resources you want to make available to users with this role.

    > **NOTE:** You can specify access to individual top-level folders, but not to specific subfolders.

11. Click **Save**.

    The role is created. You can assign it to project group members or user groups associated with the project group at any time.

Edit a Role in a Project Group

As a project group administrator, you may need to update the permissions granted by a role being used across the projects.

While modifying a role, it is better to be cautious about granting more access than required by the users or a user group. In the case of project groups, as a role could be mapped across projects, consider being restrictive.

1. Go to **My Workspace > Admin**.
2. Click your project group. The **Project Group Details** page appears.
3. From the left navigation pane, click the **Permissions** link to specify user roles applicable to the project group.
4. On the **Roles** tab, click **View: Roles Created For a Project**. The existing roles created for this project are listed. You can view global project roles by selecting View: Global Project Roles, if required.
5. Select the role that you want to modify and click **Edit**.

6. On the **Edit Role** page, make the desired changes.

   You can update the role name, description and inheritance settings on clicking **Edit**.

   The role’s permissions, assigned users or assigned user groups can also be modified as required.

7. Click **Save** after making the changes. The role in the project group is updated.

Delete a Role in a Project Group

When you no longer need a role that you created for a project, it’s a good idea to delete that role.

Any project groups administrator can create and assign a role. It is good to keep the project role tray as small and as manageable as possible.

> **NOTE:** Any existing site-wide or global project role can be associated with a project group. It is advisable to check the permissions granted via a role before assigning it to users or user groups in a project group as it would impact more than a single project.

1. Go to **My Workspace > Admin**.
2. Click **PROJECT GROUPS** from the **Projects** menu. The existing project groups are listed here.
3. Click your project group. The **Project Group Details** page appears.
4. From the left navigation pane, click the **Permissions** link to specify user roles applicable to the project group.

5. On the **Roles** tab, click **View: Roles Created For a Project**. You can view global project roles by selecting **View: Global Project Roles**. However, you can only delete roles created for this project here.
6. Select the role and click **Delete**.

   You may get a warning message if the role you are trying to delete is in use.
7. Click **OK** to proceed with deleting the non-required role.

   The selected role is deleted.

To participate in a TeamForge site, a person must have a user account on that site. TeamForge administrators can create these user accounts. This topic applies to sites with no LDAP authentication.

✓ If your TeamForge site uses LDAP authentication, TeamForge administrators cannot create new user accounts. On a site with LDAP authentication, each user must log into TeamForge using his or her LDAP credentials.

✓ On sites with LDAP/SAML/SAML+LDAP integrations, site administrators can designate select users that do not have a SAML or LDAP account as local users. Local users can log on to TeamForge using just the TeamForge credentials while bypassing the SAML/LDAP/SAML+LDAP authentication realm. A local user can also change and reset his password. For more information, see ALLOW LOCAL USER.

1. Go to **My Workspace > Admin**.
2. Click **USERS** from the **Projects** menu.
3. Click the drop-down arrow next to **Create** and click **Single User**.
4. On the **Create User** page, enter the field values appropriately.
   1. Enter a user name for the user.

      Your user name must meet these criteria:

      - User name is case-sensitive. However, to make usernames case-insentive set the site-options token ALLOW_CASE_INSENSITIVE_LOGIN to `true`.
      - Minimum number of characters as specified in the site-options.conf file.
      - No spaces.
      - Should have at least one letter.
      - The first character is a letter.
   2. Enter and confirm a password for the user, if you prefer to set the user's password yourself.

      > **TIP:** To invite users to create their own password, leave the PASSWORD field blank. A password ticket email will be sent to users to let them create a password.
   3. Enter the **FULL NAME** and **EMAIL ADDRESS** of the user.

TIP: You can add more email addresses for the user after you finish creating their profile.

4. Enter the user's organization.

   Organization can be a geographic designation, a corporate division, or whatever you want. It's advised to keep it consistent across your site.

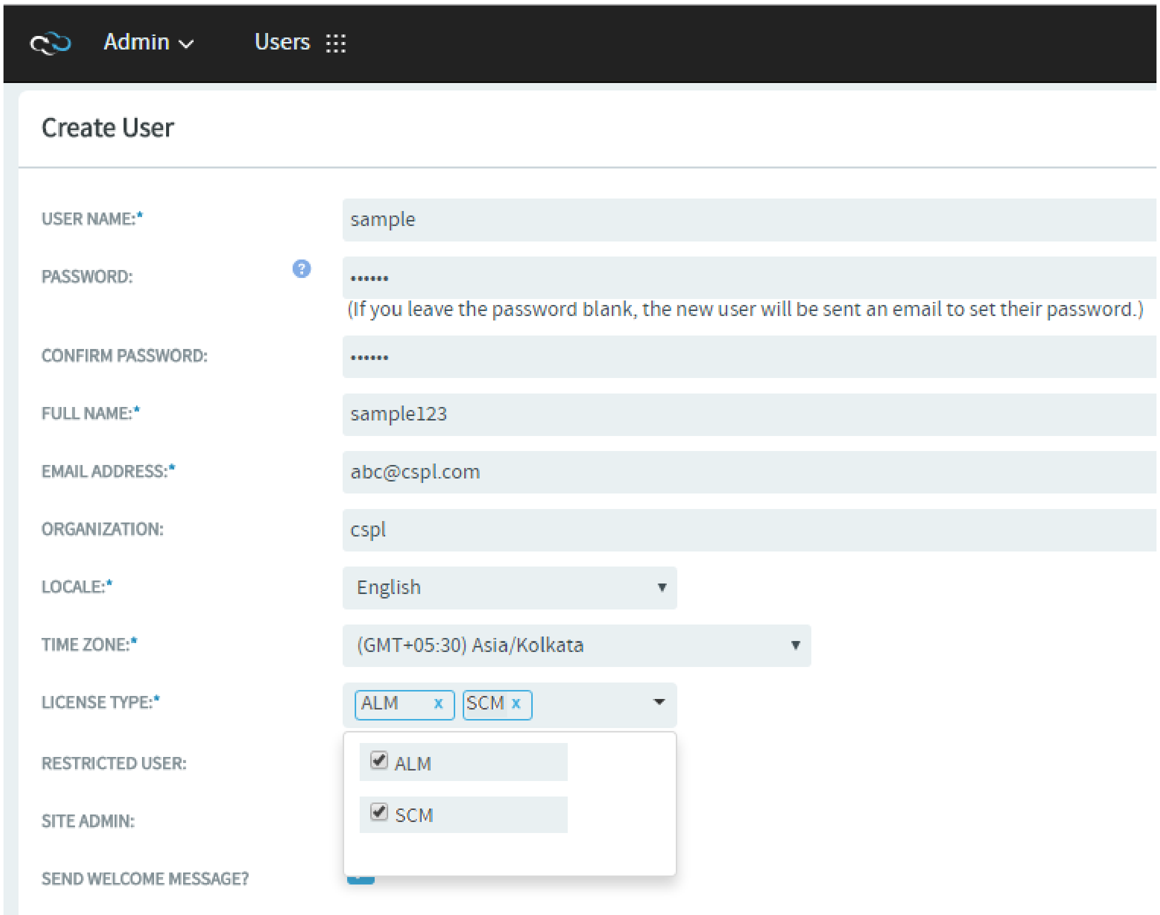
5. Select the language from the **LOCALE** drop-down list.

   NOTE: TeamForge supports English, Chinese, Japanese and Korean languages.

6. From the **TIME ZONE** drop-down list, select the preferred time zone for the user.

   IMPORTANT: Selecting the time zone overrides the default time zone set by the site options token, DISPLAY_TIMEZONE. It reflects in all the email notifications and TeamForge pages excluding integrated application pages.

7. Choose the user's TeamForge LICENSE TYPE.
   - You can assign users a combination of multiple license types such as ALM and SCM.



Create User

USER NAME:*	sample
PASSWORD:	•••••• (If you leave the password blank, the new user will be sent an email to set their password.)
CONFIRM PASSWORD:	••••••
FULL NAME:*	sample123
EMAIL ADDRESS:*	abc@cspl.com
ORGANIZATION:	cspl
LOCALE:*	English
TIME ZONE:*	(GMT+05:30) Asia/Kolkata
LICENSE TYPE:*	ALM x SCM x ☑ ALM ☑ SCM
RESTRICTED USER:	
SITE ADMIN:	
SEND WELCOME MESSAGE?	

8. Choose a user type. You can choose only one user type for each user.
   - SITE ADMIN: Administrators have unlimited access to all the data in TeamForge.
   - RESTRICTED USER: Restricted users can only access projects of which they are members.

   > **IMPORTANT:** If you do not select **RESTRICTED USER**, the user will be unrestricted and able to access all projects that have not been made private by a project administrator.

9. To send a welcome message to the user, select **SEND WELCOME MESSAGE?**.

5. Click **Create**.

   The user account is created.

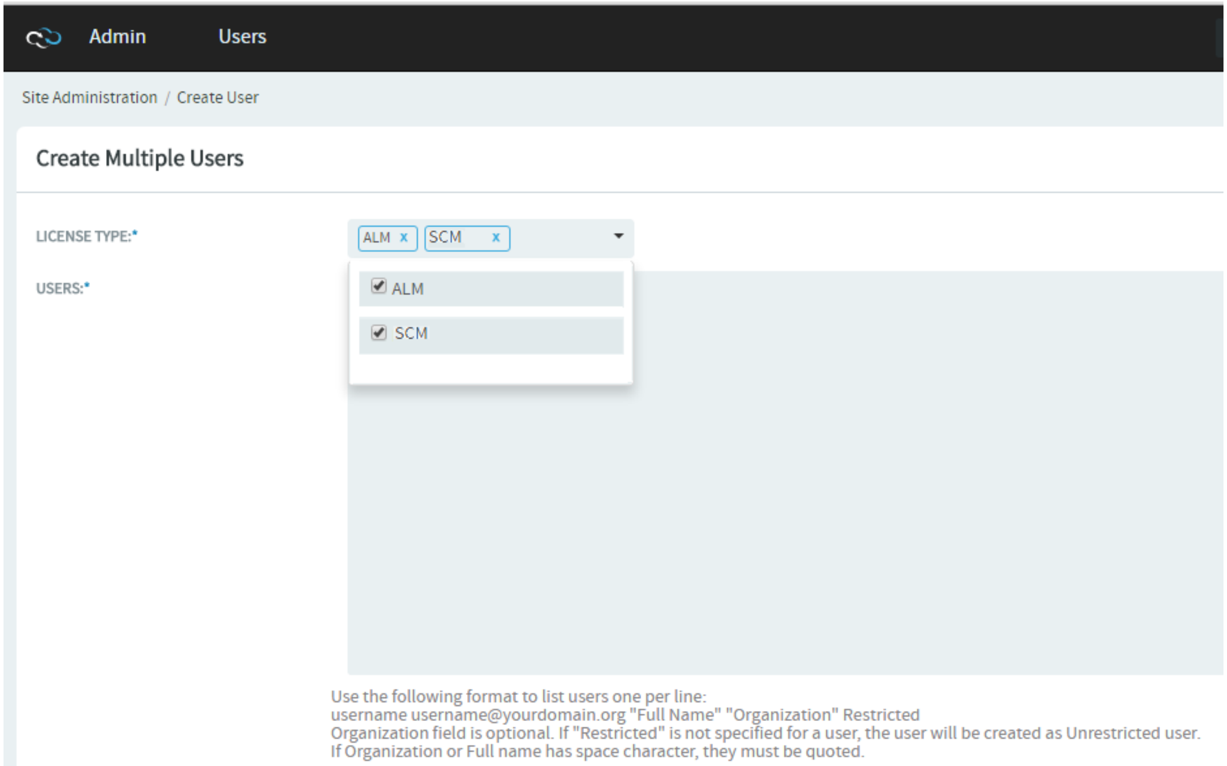
To participate in a TeamForge site, a person must have a user account on that site. TeamForge administrators can provide access to multiple users by creating their accounts together.

IMPORTANT: If your TeamForge site uses LDAP authentication, TeamForge administrators cannot create new user accounts. On a site with LDAP authentication, each user must log into TeamForge using his or her LDAP user name and password.

1. Go to **My Workspace > Admin**.
2. Click **USERS** from the **Projects** menu.
3. Click the drop-down arrow next to **Create** and click **Multiple Users**.

4. Choose the user's TeamForge **LICENSE TYPE** on **Create Multiple Users** page.

   Multi select option is now enabled. Users can now use combination of license types such as ALM and SCM.



Create Multiple Users

LICENSE TYPE:* ALM x SCM x

☑ ALM

☑ SCM

USERS:*

Use the following format to list users one per line:
username username@yourdomain.org "Full Name" "Organization" Restricted
Organization field is optional. If "Restricted" is not specified for a user, the user will be created as Unrestricted user.
If Organization or Full name has space character, they must be quoted.

5. On the **Create Multiple Users** page, enter up to 25 lines like this, one user per line:
   `username username@yourdomain.org name organization Restricted`

   Usernames must meet these criteria:

   - 1 to 31 characters.
   - Only alphanumeric characters.
   - No spaces.

- At least one letter.
- The first character is a letter.

In addition:

- Organization field is optional.
- To create an unrestricted user, omit **Restricted**.
- Restricted users can only access projects of which they are members, while unrestricted users can access all projects that have not been made private by a project administrator.
- Use quotes around the full name or the organization information if it is more than a single word.
- A maximum of twenty-five user accounts can be created at one time.

6. Click **Create**.

   The user accounts are created and password e-mails are sent to all the new users.

When a user has trouble accessing the site, you may need to reset the user's password or change the user's account status.

Edit a User Account

✓ If your TeamForge site uses LDAP for single-sign-on, passwords must be reset in the LDAP system, not on the Web administration pages. Ask your system administrator for help.

✓ To avoid disasters, TeamForge makes it impossible to delete or deactivate the TeamForge admin account. You also can’t remove the TeamForge admin flag or mark the admin user as a restricted user.

✓ On sites with LDAP/SAML/SAML+LDAP integrations, site administrators can designate select users that do not have a SAML or LDAP account as local users. Local users can log on to TeamForge using just the TeamForge credentials while bypassing the SAML/LDAP/SAML+LDAP authentication realm. A local user can also change and reset his password. For more information, see ALLOW LOCAL USER.

1. Go to **My Workspace > Admin**.
2. Click **USERS** from the **Projects** menu.
3. On the **USERS** tab, click the name of the user whose account you want to edit.
4. On the **User Details** page, click **Edit**.
5. On the **Edit User Information** page, make your changes and click **Update**. You can specify up to a maximum of three alternate email addresses, if required.

Act on Multiple User Accounts at Once

A TeamForge administrator can edit the status of multiple user accounts simultaneously.

For example, if you have multiple pending new accounts to approve, you can approve them in a batch instead of individually editing each account.

> **NOTE:** A pending user is a user who has requested an account but has not yet confirmed his or her email addresses.

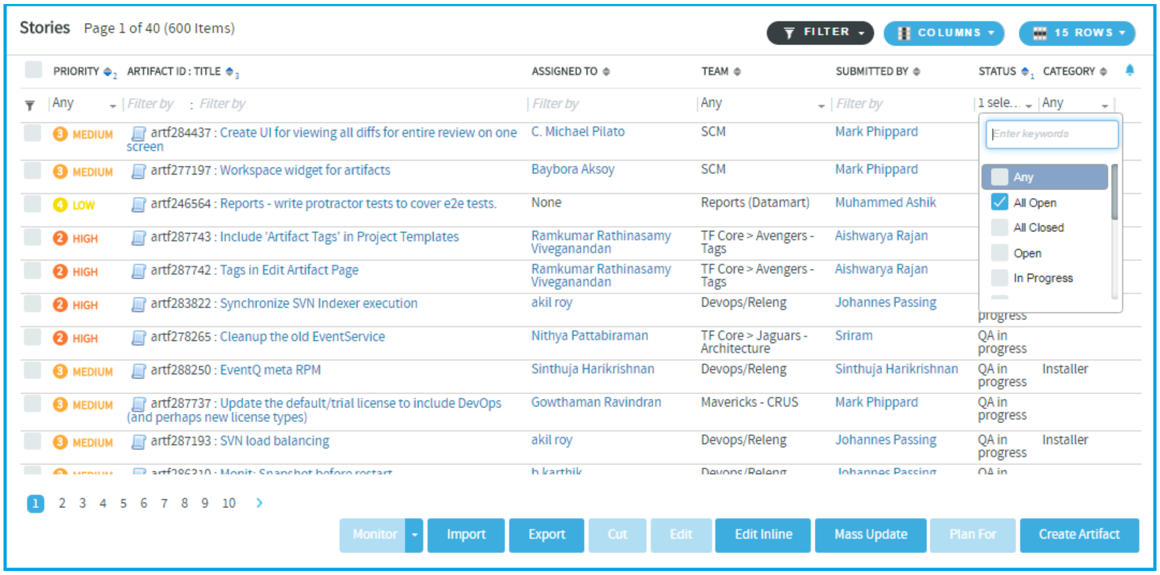
In the case of TeamForge admin accounts, you cannot make any of these edits:

- ✓ Delete the account.
- ✓ Change the account status to anything but active.
- ✓ Remove the TeamForge admin flag.
- ✓ Mark it as a restricted user.

1. Go to **My Workspace > Admin**.
2. Click **USERS** from the **Projects** menu.
3. On the **Users** page, select the users whose status you want to edit.
4. Click the desired status change.
    - Delete - Deleted users are removed from all projects. All assigned items are removed from the user. Deleted users do not count against your TeamForge license count.
    - Disable - Disabled users cannot log in to TeamForge and do not receive notification messages, but they remain members of projects and selection lists.
    - Activate - Active users have full use of TeamForge, subject to RBAC permissions.

To find a user, filter the list all CollabNet TeamForge users on your site.

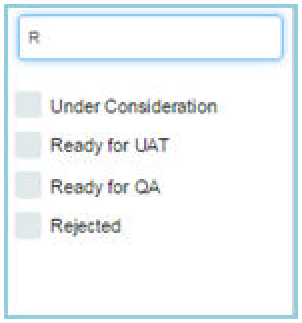
1. Go to **My Workspace > Admin**.
2. Click **USERS** from the **Projects** menu.
3. Specify the filter criteria in one or more filter fields (at the top of each column) and click **FILTER**.
    - You can find a filter field at the top of each column in most of the tables in the TeamForge application.
    - The filter field could be a text box or a drop-down list with multi-select check boxes.



Stories Page 1 of 40 (600 Items)

☐	PRIORITY	ARTIFACT ID : TITLE	ASSIGNED TO	TEAM	SUBMITTED BY	STATUS	CATEGORY
	Any	Filter by : Filter by	Filter by	Any	Filter by	1 sele...	Any
☐	3 MEDIUM	artf284437 : Create UI for viewing all diffs for entire review on one screen	C. Michael Pilato	SCM	Mark Phippard		
☐	3 MEDIUM	artf277197 : Workspace widget for artifacts	Baybora Aksoy	SCM	Mark Phippard		
☐	4 LOW	artf246564 : Reports - write protractor tests to cover e2e tests.	None	Reports (Datamart)	Muhammed Ashik		
☐	2 HIGH	artf287743 : Include 'Artifact Tags' in Project Templates	Ramkumar Rathinasamy Viveganandan	TF Core > Avengers - Tags	Aishwarya Rajan		
☐	2 HIGH	artf287742 : Tags in Edit Artifact Page	Ramkumar Rathinasamy Viveganandan	TF Core > Avengers - Tags	Aishwarya Rajan		
☐	2 HIGH	artf283822 : Synchronize SVN Indexer execution	akil roy	Devops/Releng	Johannes Passing	progress	
☐	2 HIGH	artf278265 : Cleanup the old EventService	Nithya Pattabiraman	TF Core > Jaguars - Architecture	Sriram	QA in progress	
☐	3 MEDIUM	artf288250 : EventQ meta RPM	Sinthuja Harikrishnan	Devops/Releng	Sinthuja Harikrishnan	QA in progress	Installer
☐	3 MEDIUM	artf287737 : Update the default/trial license to include DevOps (and perhaps new license types)	Gowthaman Ravindran	Mavericks - CRUS	Mark Phippard	QA in progress	
☐	3 MEDIUM	artf287193 : SVN load balancing	akil roy	Devops/Releng	Johannes Passing	QA in progress	Installer
☐	MEDIUM	artf286310 : Monit: Snapshot before restart	h karthik	Devops/Releng	Johannes Passing	QA in	

- You can type your filter criteria in the text boxes. The search text is case-insensitive.
- You can also select the filter values from one or more drop-down lists. By default, you can only select up to 10 filter values in a drop-down list. However, you can set a value that suits your requirement for the FILTER_DROPDOWN_MAX_SELECTION token in the site-options.conf file to increase or decrease the count.
- **Filter-as-you-type**: You can find the **Enter keywords** text box in all filter drop-down lists. As you type your filter keyword, instant search results are shown in the drop-down list. For example, in the following illustration, typing “R” instantly shows all statuses having the alphabet “R”. The search text is case-insensitive.



R

☐ Under Consideration
☐ Ready for UAT
☐ Ready for QA
☐ Rejected

- Some search filters may not appear if your site administrator has not enabled them.

4. After filtering, if you want to clear the filters, click **FILTER** and select **Clear** from the drop-down list.

All users meeting your filter criteria are displayed.

To manage multiple users at once, create a group and add users to such user groups.

Create a User Group

To manage multiple users at once, create a group that represents them.

1. Go to **My Workspace > Admin**.
2. Select **USER GROUPS** from the **Projects** menu.
3. Click **Create** and provide a name for the group and a description of its purpose.

> **NOTE:** If your project is a child of another project, it may have inherited one or more user groups from its parent project. To work with inherited users and user groups, you must go to the project that they belong to.

4. Click **Create**.

Add a User to a User Group

Put together multiple users who share characteristics in a user group.

1. Go to **My Workspace > Admin**.
2. Select **User Groups** from the **Projects** menu.

> **TIP:** In TeamForge 18.1 and later, **Groups** (in earlier versions of the product), has been renamed to **User Groups** to better distinguish user groups from project groups.

3. Under **User Groups**, click the group to which you want to add the user.
4. On the **Edit Group** page, click **Add**.
5. Use the picker to move users into the group, and click **OK**. You can select the inherited project members also from the list.
6. Click **Return**.

Find a User’s Groups

You can get a consolidated view of all user groups, which a user is a member of.

1. Go to **My Workspace > Admin**.
2. Click **USERS** from the **Projects** menu.

3. On the **Users Details** page, click the **USER GROUP MEMBERSHIP** tab.

The groups listed on this tab are the groups that this user is a member of.

Edit a User Group

1. Go to **My Workspace > Admin**.
2. Select **User Groups** from the **Projects** menu.
3. Under **Groups**, click the group's name you want to edit.
4. Click **EDIT**.
5. Make your changes and click **Update**.

If your TeamForge installation authenticates against an LDAP directory, follow these instructions to reset your admin account password.

If your installation does not validate against LDAP, click Forgot Your Password on the TeamForge home page to reset the password for the admin account.

1. With a web browser, go to the URL `http://<host>sf/sfmain/do/forgotAdminPassword`.
2. On the **Admin Account Password Retrieval** page, Click **Send Email**. TeamForge sends an email to the address specified for the admin user.
3. Check your email and click the link provided to reset your password.
4. On the **Reset Password** page, enter and confirm a new password.
5. Click **Reset Password**.

   You can now log into CollabNet TeamForge with your new password.

A user can have multiple roles in different projects either by being directly assigned those roles or by inheriting them. You might find it useful to see all the roles assigned to a user in a TeamForge site before adding or removing a role.

1. Go to **My Workspace > Admin**.
2. Click **USERS** from the **Projects** menu.
3. In the **ROLES** tab, select a role type in the **View** drop-down–**Roles created for a Project**, **Roles Inherited From Parent Project** or **Site-wide Roles**.
   - Roles created for this project include the roles the user is directly assigned or assigned through a user group in projects and project groups.
   - Inherited roles include the roles the user inherits from parent projects and project groups.

These are additional details you can follow while configuring your email settings.

Remove Users from Monitoring Objects

As a site or project administrator, if one or more users are no longer project members, you can remove them from monitoring selected TeamForge objects they once subscribed for monitoring.

However, you cannot remove a user from the monitoring list if the user is monitoring applications such as trackers, documents, tasks, and so on instead of individual TeamForge objects.

By default, this feature is disabled. To enable this feature, set the USER_MONITORING_REMOVE_ENABLED token to `true` in the `site-options.conf` file.

> **NOTE:** Every user removal operation is being logged in the database for audit purposes.

1. Go to the item, from which you want to remove users from monitoring.
2. Select users to remove from monitoring list.
3. If you want to remove one or more users from monitoring one of the items, select the item, then click **Monitor > Users Monitoring Selected**.

   The Users Monitoring This Item window appears.
4. If you want to remove one or more users from monitoring more than one item, select all the items, then click **Monitor > Users Monitoring This Folder**.
5. In the case of team monitoring, click **Monitor > Users Monitoring This Team**.
6. In the following window, select one or more check boxes corresponding to the users you want to remove from monitoring.
7. Click **Remove**. The `Are you sure you want to remove the selected user(s) from monitoring?` message appears.
8. Click **OK**.

   The selected users are removed from monitoring the selected object. An e-mail notification is sent to all active users that are removed from monitoring selected objects.

Limit the Size of Message Attachments

To avoid overtaxing your mail server or your storage volume, you may want to set a ceiling on the size of the attachments that users can send to a forum via email.

When a user sends an attachment that is larger than the limit, the message is rejected and the user gets an email from the Site Administrator explaining that the attachment exceeded the limit.

> **TIP:** Before imposing a file attachment size limit, it's a good idea to point your users to better ways of collaborating around large files. Consider suggesting source code repositories, backup systems, or other appropriate solutions.

1. Open the `site-options.conf` file, the master configuration file that controls your TeamForge site.
   ```
   vi /opt/collabnet/teamforge/etc/site-options.conf
   ```
2. Set the value of the DISCUSSION_MAX_ATTACHMENT_SIZE token.

   For example, if your users are given to using Microsoft Word documents on the site, you might set `DISCUSSION_MAX_ATTACHMENT_SIZE` to 10 MB, and increase the value by two or three MB at a time if users need more headroom.
3. Review the changes, then save the `site-options.conf` file.

Limit the Size of Document Attachments

When many users store very large documents on your site, you may sometimes notice a slowdown in your site's performance. You can reduce the impact of such a use pattern by telling TeamForge not to attach documents larger than a certain size.

> **TIP:** It's also a good idea to let your users know that the Documents tool in TeamForge is not designed primarily as a storage device. As a best practice, upload documents to make them available for collaboration, not for backup or long-term storage.

1. Open the `site-options.conf` file, the master configuration file that controls your TeamForge site.
   ```
   vi /opt/collabnet/teamforge/etc/site-options.conf
   ```
2. Add the DOCUMENT_MAX_FILE_UPLOAD_SIZE parameter and give it a value equal to the maximum size (in megabytes) of documents to be uploaded.
3. Review the changes, then save the `site-options.conf` file.

Relay Emails Through SMTP Gateway with Authentication

You can set up TeamForge to relay emails through an SMTP gateway (such as Amazon AES) that uses authentication. By default, James sends emails directly. However, you may prefer relaying emails through an enterprise relay server. Configuring the JAMES_GATEWAY_* tokens let you do that.

> **NOTE:** By default, TeamForge uses the user's email address (as registred in TeamForge user profile) as the sender address (`From`). Therefore, it is important that the mail relay does not impose any restrictions on the email sender address and accepts all emails to be relayed.

To relay emails through an SMTP gateway:

1. Set the following site-options.conf tokens.
   ```
   vi /opt/collabnet/teamforge/etc/site-options.conf
   JAMES_GATEWAY_HOST=
   JAMES_GATEWAY_PORT=
   JAMES_GATEWAY_USERNAME=
   JAMES_GATEWAY_PASSWORD=
   ```
2. Save the `site-options.conf` file.
3. Provision services.
   ```
   teamforge provision
   ```

To facilitate troubleshooting, keep an eye on key data about your CollabNet TeamForge site.

Check Your Server's Status

As a first troubleshooting step, check the **Server Status** page to see if your servers are running and connected to the main TeamForge server.

> **NOTE:** The **Server Status** page provides only the current status of each server. It is not a diagnostic tool, and you must correct any connection or configuration issues on the server in question.

1. Go to **My Workspace > Admin**.
2. On the site administration navigation bar, click **SYSTEM TOOLS**.
3. On the **System Tools** menu, click **Server Status**.
4. Find the status of the server you are interested in.

   For each server, you see one of the following status messages:

   - **OK**–The server is running and connected to the main TeamForge server.
   - **Could Not Connect**–The server is not running or is not connected to the main TeamForge server.

Get Build Information

For solving some kinds of problems, it may be useful to know which software components are installed, the exact build number, and other technical information about your TeamForge site.

1. Go to **My Workspace > Admin**.
2. On the site administration navigation bar, click **SYSTEM TOOLS**.
3. On the **System Tools** menu, click **Build Information**. The **Build Information** page consists of the following information: TeamForge version number, build number, operating system name and version number, instance (any packages of customizations that have been applied) and a list of installed RPMs.

Get Reports on Site Activity

You can keep track of various kinds of user activity, such as the number of user logins, source code commits, and tracker activity on your site.

- Monitor the total number of commits on your site by all users, in all projects, per day. You can track SCM commits to Subversion and CVS repositories, and to Git repositories as well, if you have the TeamForge Git integration set up.
- Check how many users log into your site in a given period. For example, to assess the impact of changes you've made to your site's look and feel (see Customize TeamForge), look for trends in the user login numbers for the week following the changes.
- Check the artifacts that are created and closed in a given period.

1. Go to **My Workspace > Admin**.
2. Click **REPORTS** from the **Projects** menu and select one of the available reports.
3. Click a **Zoom** option to see a week, a month, a quarter, 6 months, or a year of activity. The **Max** option shows all the data that's ever been collected.
4. Use the **From** and **To** date fields to zero in on the exact period of activity you are interested in.
5. Roll your cursor over any data point on the graph to see the exact numbers that determine that point's position.
6. Click **Grid View** to see the data as a table. You can toggle between chart and grid view as needed.

If your site's users need access to an application or web site that is not part of TeamForge, you can make it available by linking or integrating from within your TeamForge site.

What is a linked application?

A linked application is an external application or site that users can get to from inside a TeamForge project.

You can use linked applications to incorporate these types of applications into your TeamForge project:

- Third party applications
- Internally developed applications
- Integrations developed using the TeamForge SOAP APIs
- Company intranet sites
- External websites

After you create a linked application, a button is added to your project navigation bar. Clicking the button displays the linked application in the main TeamForge project window (or in the case of a site-wide linked application, in a separate window).

> **NOTE:** You can create as many linked applications per project as you wish. However, because each linked application adds a button to the project navigation bar, creating a large number of linked applications can cause horizontal scrolling.

TeamForge administrators can also create site-wide linked applications that appear in all TeamForge projects.

Create or Edit Linked Applications

To bring an external tool into your environment quickly and easily, set it up as a linked application. When you create a site-wide linked application, it appears in all projects on your TeamForge site. Site-wide linked applications are especially useful for incorporating corporate standard external applications, such as a company intranet sites, into your TeamForge installation.

Create a Site-wide Linked Application

1. Go to **My Workspace > Admin**.
2. On the site administration navigation bar, click **INTEGRATIONS > SITE-WIDE LINKED APPLICATIONS**.
3. Click **Create**.
4. On the **Create Site-wide Linked Application** page, provide a name for the linked application. This name appears on the link in the TeamForge navigation bar.
5. Enter the server location or URL for the linked application.
6. Select whether you want to enable single sign-on for the linked application.
   - If you use single sign-on, access to the linked application is managed through the TeamForge authentication system. Users are not required to log into the linked application after they have logged into TeamForge.
   - If you do not use single sign-on, users will be required to log in to the linked application using its native authentication system.

7. Choose how you want the linked application to appear when a user clicks it.
    - **In the same window**: The linked application takes over the entire browser window, replacing whatever the user was looking at.
    - **In a new window**: The linked application launches in a separate browser window.
    - **In an iframe**: The linked application appears in a box in the same window, framed by the TeamForge site's header and navigation controls.
8. Click **Save**.

    A link for the site-wide linked application is added to your TeamForge navigation bar. Clicking the link displays the application in the main TeamForge window.

Edit a Site-wide Linked Application

When the use patterns of a linked application change, you may need to change the way the application integrates with TeamForge.

1. Go to **My Workspace > Admin**.
2. On the site administration navigation bar, click **INTEGRATIONS > SITE-WIDE LINKED APPLICATIONS**.
3. Click the name of the site-wide linked application that you want to edit.
4. On the Edit Site-wide Linked Application page, make the changes you need. You can edit these elements:
    - The name of the application.
    - The application's URL.
    - Whether the application uses single sign-on.
    - Whether the application is displayed in a new window, in the same window, or in an IFrame.

    > **NOTE:** You cannot change the application icon.
5. Click **Save**.

What is an integrated application?

An integrated application is a stand-alone application that can seamlessly integrate into any TeamForge project.

You can use integrated applications to incorporate these types of applications into your TeamForge project:

- Third party applications

- Internally developed applications
- Integrations developed using the TeamForge SOAP APIs
- External websites

When you add an integrated application to your project, an icon is added to your project navigation bar. Clicking the icon displays the integrated application in the main TeamForge project window.

TeamForge site-administrators can register site-wide integrated applications that project administrators can opt to use across projects.

Site administrators or users with site-wide roles with the administration permissions for integrated applications can enable/disable integrated applications.

TIP: Disabling an integrated application restricts it from being added to projects. However, disabling an integrated application does not affect the projects where the integrated application might already be in use.

After your site administrator registers an integrated application on the site level, on adding it to your project, an icon is added to your project navigation bar. Clicking the icon displays the integrated application in the main TeamForge project window.

NOTE: You can register and integrate as many applications per project as you wish. However, because each integrated application adds an icon to the project navigation bar, creating a large number of integrated applications can cause horizontal scrolling.

Create Integrated Applications

To make a tool comprehensively available to your users, set it up as an integrated application.

Before you can make an integrated application available to Project Administrators, your system administrator must integrate the application with your site. This may involve modifying the application. How this is done depends in part on the application.

Integrate an External Application into a TeamForge Site

When you integrate an external application into your TeamForge site, your site’s project administrators can choose to include the integrated application alongside the built-in tools in their projects.

When you have integrated the application, project administrators on your site can add it to their set of collaboration tools. Objects they create will share the core TeamForge features, such as authorization, authentication, go-urls, association, linkification, templating, Project Pages components, search, and source code management support.

1. Find the XML files that describes your integrated application.
2. Log into TeamForge as an admin user.
3. Go to **My Workspace > Admin**.
4. Click **INTEGRATED APPS** from the **Projects** menu.
5. Click **Create**.
6. Use the **Browse** window to find the configuration file you created, then click **Next**.
7. On the **Preview** page, review the parameters you set in the configuration file.

   > **NOTE:** You may have to revise one or more values to ensure they are valid.

8. Click **Save**.

The application is now available for all projects on your site. You can direct project administrators to these instructions to add it to their own project toolbars.

Enable or Disable Integrated Applications

Site administrators can enable or disable integrated applications.

In TeamForge, enable an integrated application to make it available for use in the projects.

> **TIP:** You can also disable an integrated application to restrict it from being added to projects.

1. Go to **My Workspace > Admin**.
2. Click **INTEGRATED APPS** from the **Projects** menu.
3. To make the integrated application available for use, select the integrated application and click **Enable**.
4. To stop making the integrated application available for use, click **Disable**. The system asks for confirmation before disabling an integrated application. Click **OK** to disable the integrated application.

   A disabled integrated application can be re-enabled when there is a need to use that integrated application in a project.

Remove a Tool (Integrated Application) from Project Admin Menu

To stop making an application or site outside of TeamForge available to your users from inside your TeamForge projects seamlessly, disintegrate an application.

> **NOTE:** You may have integrated several external applications per project to maximize the integrated applications feature’s utility. However, because each integrated application adds an icon to the project navigation bar, a large number of integrated applications can cause horizontal scrolling. Consider removing the integrated applications that are not in use.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **Tools**.
3. To remove an integrated application from the list of tools displayed , clear the **Visible** check box and click **Save**. The selected integrated application is removed from the project. The icon of the removed integrated application disappears from your **Project Home** menu.

   > **NOTE:** Removing an integrated application, removes all associations to the integrated application too. Removing an integrated application, removes the component type in the **Project Home Create Component** page.

Set Site-level Permissions for an Integrated Application

In TeamForge, as the site administrator you can set the site-level permissions for an integrated application.

The default access permissions are usually set using the configuring xml file for each integrated application. However, you may want to provide permission access at site-level for some of the users.

1. Go to **My Workspace > Admin**.
2. Click **INTEGRATED APPS** from the **Projects** menu.
3. Set the permissions for the integrated application and click **Submit**. The permissions are set for the selected integrated application. It is possible that an integrated application may be added to a project by a restricted user whose project administration permissions come from a global project role. In such a case, you could be left with no one authorized to administrate the integrated application. TeamForge handles that situation in one of two ways:
   - If one or more users with the “Founder project administrator” role is a member of the project, then the restricted user who did the integration gets that role too.
   - If there is no project member with the “Founder project administrator” role, the restricted user who did the integration gets a new administrator role called “<integrated_application_name> Administrator.”

View Integrated Application Information

TeamForge site administrators, and users with site-wide roles and the IAF permission, can view information about an integrated application’s configuration and other details.

1. Go to **My Workspace > Admin**.
2. Click **INTEGRATED APPS** from the **Projects** menu.
3. Click the name of the integrated application that you want to view. The default category tab, **GENERIC**, displays all the existing integrated applications associated with the site. You can have your integrated application displayed under a separate **CATEGORY** tab by defining it in the deployment configuration file.

   The configuration details of the integrated application are displayed.

Edit an Integrated Application

TeamForge site administrators can update an integrated application’s URLs (SOAP endpoint URL, administration URL, browser URL) and other parameters by uploading XML files containing the changes.

1. Go to **My Workspace > Admin**.
2. Click **INTEGRATED APPS** from the **Projects** menu.
3. Select the integrated application you want to edit.
4. Click **Edit**.
5. On the **Edit Integrated Application** page, make the changes you need. You can edit these elements:
   - The Deployment Configuration File.
   - The Application Configuration File.
6. Click **Browse** and select the files to attach.
7. Click **Next**.

Export an Integrated Application

TeamForge site administrators can export an integrated application’s variables in XML format and have them available for editing.

1. Go to **My Workspace > Admin**.
2. Click **INTEGRATED APPS** from the **Projects** menu.
3. To export the integrated application, select the integrated application name.
4. From the **View Integrated Application** page, click **Export**.

The integrated application is exported.

If you have registered Secure Socket Layer (SSL) certificates, your site's users can use SSL when they set up an SCM integration server.

Register SSL Certificates

If you use certificates that are generated in-house, self-signed, or signed by a non-established Certificate Authority, they must be registered with each client system that will connect to the TeamForge server. Registration consists of importing custom certificates into the Java runtime's global keystore on each server.

> **WARNING:** This affects any other Java applications on the server that use the same Java runtime.

1. Collect the server certificates from all servers. On RHEL, CentOS and other RedHat-based distributions, these are contained in `/etc/httpd/conf/ssl.crt/server.crt`.

   > **IMPORTANT:** Be sure to use exactly this path, as there are other files with similar names, plus server certificates are not really secret, but some other files are. So, files must be copied (e.g., via scp) to the same directory, and renamed if necessary to avoid conflicts. It's recommended that you use the short server name of the corresponding server for this.

2. Locate the Java keystore.

   This is `PATH_TO_JAVA/jre/lib/security/cacerts`. For example, this may be `/usr/local/j2sdk1.4.2_10/jre/lib/security/cacerts`.

3. Locate the Java keytool utility.

   This is `PATH_TO_JAVA/bin/keytool` For example, `/usr/local/j2sdk1.4.2_10/bin/keytool`.

4. Import each server certificate into the keystore.

   ```
   PATH_TO_JAVA/bin/keytool -import -keystore PATH_TO_JAVA/jre/lib/security/c
   acerts -file <server>.crt -alias <server>
   ```

   > **NOTE:** Any value is accepted for server in -alias .

5. At the password prompt, use `changeit`. Confirm that you trust the certificate by typing yes.

6. Verify that all your certificates are added.
   ```
   PATH_TO_JAVA/bin/keytool -list -keystore PATH_TO_JAVA/jre/lib/security/cacerts |less
   ```

   > **NOTE:** The list will contain many more certificates. These are top-level CA certificates, provided with Java.

7. If you are running more than one separate server, repeat these steps for each server.
8. Restart TeamForge

From now on, you can select the **Use SSL** check box, if required, when creating an SCM integration.

Encrypt Database Network Traffic (On Sites with Remote Database Servers)

To prevent your data from being exposed in a readable format on the network, use the Secure Socket Layer (SSL) to encrypt the network traffic between the Application and the Database servers.

If you have a dedicated database server (operational database or datamart), encrypt the data traffic between the application and database servers and between the ETL and datamart servers.

> **IMPORTANT:** The following steps are relevant for a distributed setup only.

1. Stop TeamForge.

   > **IMPORTANT:** Stop TeamForge on all the servers in a distributed setup.

   - If you are upgarding from TeamForge 16.7 or earlier releases:
     `/etc/init.d/collabnet stop`
   - If you are upgrading from TeamForge 16.10, 17.1, or 17.4 releases:
     `/opt/collabnet/teamforge/bin/teamforge stop`
   - If you are upgrading from TeamForge 17.8 or later releases:
     `teamforge stop`
2. Add the following site option tokens to all the TeamForge servers.
   1. If the operational database is running on a separate server, include the token `DATABASE_SSL=on`.

   2. If the datamart is running on a separate server, include the token `REPORTS_DATABASE_SSL=on`.

> **NOTE:** It is mandatory to include these tokens on all the servers.

3. Provision services.
   `teamforge provision`
4. Verify that your PostgreSQL database is running in the SSL mode.
   1. Log on to the Database Server and run the following command:
      ```
      grep "ssl = " var/lib/pgsql/9.6/data/postgresql.conf
      ```
      Observe:”ssl = on”

Provide sample projects to help users get started quickly. TeamForge comes with a sample template useful for agile development projects. Site administrators and project managers can use this template to jumpstart a project without a lot of manual setup steps.

Project templates are installed by default when you install TeamForge. TeamForge comes with a sample template useful for agile development projects. Site administrators and project managers can use this template to jumpstart a project without a lot of manual setup steps.

To install project templates using the TeamForge startup script, set the following tokens and restart the CollabNet services:

```
INSTALL_TEMPLATES=true
REQUIRE_USER_PASSWORD_CHANGE=false
```

✓ If the project templates are already installed, you cannot re-install them using the TeamForge startup script.

✓ You may choose to delete the sample project templates. After deleting the sample project templates, you must set the *INSTALL_TEMPLATES* token to `false`. Otherwise, the project templates, if not found in the database, are installed automatically every time you restart the CollabNet services.

Update a Project Template

To revise or correct an existing project template, overwrite it with a template of the same name.

- You must be a site administrator to overwrite an existing project template.
- Revise the project that will serve as the basis for the new project template.

> **TIP:** You can disable a project template while you make changes to the project. To disable a project template, use the **My Workspace > Projects > Templates** page.

1. Select **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Settings** page, click **Create Project Template**.

3. Select **Replace Existing Template** and choose the template you want to overwrite.
4. Provide the description for the template. (If you want to change the template name too, create a new template from the same project and disable the existing template.)

> **TIP:** It’s a good idea to use the description to note the changes from the previous version of the template.

5. Select the items you want to be available when new projects are created from this template.
6. The replacement project template is created. Its name and description appear on the **Template** tab of the **Projects** list, accessible from the navigation bar.

Enable or Disable Project Templates

Site administrators or users with site-wide roles with project administration permissions can enable/disable project templates.

In TeamForge, enable a project template to make it available for use for creating new projects.

1. On **My Page**, click **Projects** and select the **PROJECT TEMPLATES** tab.
2. To make the project template available for use, select the template and click **Enable**.
3. To stop making the project template available for use, click **Disable**.

> **IMPORTANT:** You can create new projects only from the list of enabled project templates.

- The Template Name field displays the name of the template using which the project was created. However, post upgrade to TeamForge 16.7 (or later), you will see the Template Name as “not available” for projects created in TeamForge 16.3 or earlier.
- The Template Name field shows a hyphen (-) in cases where projects are created not from a template.
- The template name is struck through in cases where the template used to create the project was deleted.

> **NOTE:** Projects created using a template will now have the information about the template used for project creation.

Set the interval at which you want yourTeamForge site's data extracted to the datamart from which reports are generated.

Each extract-transform-load (ETL) job consists of extracting the data from the production database, transforming it to support reporting, and loading it into the datamart.

By default, this is done every night at 2:30 a.m., by the host's local clock.

1. Open the `site-options.conf` file, the master configuration file that controls your TeamForge site.

   ```
   vi /opt/collabnet/teamforge/etc/site-options.conf
   ```

   > **NOTE:** `vi` is an example. Any *nix text editor will work.

2. Set the *ETL_JOB_TRIGGER_TIME* variable to the interval at which you want ETL jobs to run. For example, a value of `0 0/15 * * * ?` will run an ETL job every 15 minutes.
3. Review the variables you have changed, then save the `site-options.conf` file.

You can query the database if you are a site administrator or have been given access to System Tools by another site administrator.

You can raise a database query using the **Admin > System Tools > Ad Hoc Database Query** page.

> **NOTE:** This feature is not available for sites that use the Oracle database.

In the **Ad Hoc Database Query** page, select a data store, `Operational Datastore` or `Datamart`, type the "select" query and click **Run Query**. The query is executed against the CollabNet TeamForge database and the results are dispayed.

For security reasons, the `sfuser` and `password_history` tables of the operational datastore are restricted for ad hoc querying from the **Admin > System Tools > Ad Hoc Database Query** page. Alternatively, use the following views, `sfuser_view` and `password_history_view`, for retrieving all other data but passwords.

> **NOTE:** The results of your query may be limited or your query session may timeout as per the settings in the `site-options.conf` file.

You can submit read-only queries of the format:

```
SELECT [FROM][WHERE][GROUP BY] [HAVING] [ORDER BY]
```

You can use the following special keywords while drafting the query:

- "\d" or "show tables" - To list all the tables.
- "\d <table name> - To view the description of a specific table.
- "select * from <table name> - To view all the contents of a specific table.

When you set up Lab Management, your team members can use TeamForge to access their own virtual machines for developing and testing.

1. Go to your **Lab Management** project.
2. Select **Project Admin** from the **Project Home** menu.
3. Click **Tools** from the Project Admin menu.
4. Click **Add Tool**.
5. Select **Other** from **Select Tool Type** drop-down list.
6. Enter the display name as **Lab Mgmt**.
7. Select the **Show in Toolbar** check box.
8. Enter the server location or URL of your Lab Management server.
9. If you are a TeamForge administrator, select whether you want to use single sign-on for the linked application.
   - If you use single sign-on, TeamForge manages authentication for Lab Management, and users don’t have to log into Lab Management after they have logged into TeamForge.
   - If you do not use single sign-on, users must log into Lab Management using its native authentication system.
10. Click **Browse** and select the icon ( ) for the linked application.
11. Click **Save**.

A **Lab Mgmt** button is added to your **Project Home** menu. Clicking it launches the application in the main TeamForge project window.

To enable a project member to use a system, you must reallocate the system to that user.
You must be a Domain Administrator or Project Administrator to free a host that does not already belong to you.

1. Notify the user you’re taking the system from through email or some other means, so that they don’t lose any important work which they may not have saved on that system. The user whose system was deallocated will get an email after the deallocation, but it is a courtesy to notify the user beforehand.

2. On your project’s **Project Home** page, click the host you want to reallocate.

3. Click **Free Host**. The host is now free and can be allocated by anyone in the project. TeamForge Lab Management emails the user that their system has been deallocated by an administrator.

4. To discourage other users from allocating it, put a note in the host’s **Description** stating something like “This host reserved for Jane.”

5. Notify the user who is going to allocate the new system that their system is now ready.

Defining scope is an iterative, interactive process. As you go through it, you'll find elements of your scope expanding, shrinking or changing shape in response to feedback from analyzing and planning out the work.

NOTE: It's a good idea to get the feature definition process under way *before* setting up planning folders (see Creating a Planning Folder), because defining features gives you the raw material for your planning process.

1. Before you get started, you'll do some kind of user research, even if it's only a few phone calls, to get an idea of the needs of the customers you hope to satisfy. Express these needs and desires as stories about what a user can do with your product. Then create an artifact for each user story you identify. A user story describes the situation after your product has been launched: What can the user do now that they couldn't do before?

   TIP: For best results, your user story artifact should by limited to a clear description of the capability you want in the product. Remember that a user story is not an implementation plan. Details about the implementation will be recorded in the user story artifact, but when you write a user story, try to leave the implementation up to the developer who will be responsible for it.

2. You may want to define one or more fictional users who resemble the real-life users of your product. This can help simplify and focus your thinking.
3. Use the Priority field to express your opinion of how important the story is to the user. In general, the most eagerly desired capabilities will be addressed first. (During implementation planning, your Priority setting will be used as one input in summing up the effort involved in each priority level.)
4. Make it as clear as you can at the outset what degree of functionality is acceptable. For example, if your team is creating an airplane, how high must it be able to fly? How far must it go before refueling? How many passengers must it carry? Stating your acceptance criteria concretely helps reduce the time needed for ongoing reviews and changes.

Create a new project when you have identified work to be done that has its own distinct character, dependencies or schedule.

What constitutes a project depends on your organization. Some organizations favor a small number of big, centralized projects. Others prefer a larger number of smaller, specialized projects. Your site administrator can help you decide if your work should be part of a larger project or a project of its own.

A project is a workspace where people can use the CollabNet TeamForge applications to collaborate and to create, store, and share data.

All the work you do with CollabNet TeamForge is organized into projects. Any registered CollabNet TeamForge user can create a project, subject to approval by a CollabNet site administrator. After a new project is approved, the project creator can configure project applications, add project members, and create and assign roles to govern each user's individual access permissions and the access permissions of groups of users. A registered CollabNet can also request membership in any CollabNet project. Requests to join projects are submitted to the project's administrators for approval.

How CollabNet TeamForge projects are organized is up to you and your organization. You might choose to create one large, centralized CollabNet TeamForge project in which to manage all of your organization's development work. Or you might choose instead to create a number of smaller projects for each team or sub-project.

Any registered user on the site can create a project, subject to approval by a CollabNet TeamForge site administrator.

You can use a project template to pre-populate new projects with the structure and configuration of an existing project.

When you create a project, it is submitted to the CollabNet TeamForge administrator for approval. You will receive an email notification when the site administrator approves or rejects your project. When your project is approved, you are assigned the Founder `Project Admin` role and made a project administrator. You can access the project from **My Projects** or **View All Projects** menu option under **My Workspace**.

1. Access the **Projects** page through either of the following ways:
   - Go to **My Workspace** > **View All Projects**
   - Go to **My Page** > **Projects**.
2. Click **Create New Project**.
3. On the **Create Project** page, give the project a name and a brief description.
   - The name will appear in project lists and on the project's home page.
   - A terse description is recommended. There will be unlimited room to discuss the project's aims and methods in detail on the project pages themselves.
4. Provide a URL name for the project, if you want the URL for the project to be different from the internal project name.

   If you do not enter a URL name, the project URL will be the same as the project name.
5. If your site administrator has provided project templates (see Create a Project Template), select the appropriate one for your new project.

   Project templates give you ready-made artifact types, work flow support, user roles and other start-up content appropriate to the kind of project you are creating.

   > **TIP:** The DEFAULT PROJECT ACCESS and PROJECT ACCESS EDITABLE options are displayed based on the project settings. If you would like to change these settings, ask your site administrator.
6. Click **Create**. The project is submitted to the TeamForge site administrator for approval. You will receive an email notification when the site administrator approves or rejects your project. When your project is

approved, you can get to it from your **MY PROJECTS** tab available under **PROJECTS** menu in the **My Workspace** page or from the **Projects** menu in your navigation bar.

The Project Dashboard offers a centralized view into all development projects managed in TeamForge .

The Project Dashboard provides managers with an at-a-glance overview of the status of each of their projects. It provides summary information on the number and status of the tasks and tracker artifacts in each project, and calculates project overrun and underrun statistics.

The Project Dashboard also provides overview information such as project start and end dates and project ranking.

You can see the Project Dashboard if you have both the View Tracker and View Task permissions for one or more projects. Only those projects for which you have both the View Tracker and View Task permissions appear on your Project Dashboard.

In the TeamForge navigation bar, click **My Workspace > Dashboard**.

You can view the Project Dashboard if you have both the View Tracker and View Task permissions for one or more projects. Only those projects for which you have both the View Tracker and View Task permissions are displayed on your Project Dashboard. Contents

For each project, the Project Dashboard displays the following information:

- **Project Activity ranking**–The activity of the project in relation to all other TeamForge projects.
- **Start Date or End Date**–The start and end date of the project, based on the start and end dates of all project tasks.
- **Task Status**–The status of the project, based on the “rolled-up” status of all project tasks and task folders. You can configure the “roll up” criteria for each project from the project’s Task Manager Settings page.
- **Status History**–The history of the project’s “rolled-up” status color. These figures are calculated in real time, but do not calculate time that the project’s status was Not Started or Completed.

  Click the status bar to go to the project’s Task Summary page.
- **Task and Tracker Effort** - The project’s current overrun or underrun, based on the difference between estimated and actual effort spent on project tasks and tracker artifacts.

  Only completed and closed tasks and tracker artifacts, with values in the estimated and actual effort fields, contribute to the overrun or underrun calculations.
- **Tracker Status**–The number of open tracker artifacts in the project, per priority value. The number of open tracker artifacts is indicated in parentheses.

  Click the status bar to go to the project’s Tracker Summary page.

As a project administrator in TeamForge Lab Management, you can edit certain properties for your project.

1. On the project home page for the project that you wish to edit, click **Edit Project**.
2. Edit the following parameters:
   - **Project Summary** - A brief, one-line summary of the project.
   - **Project Description** - A more detailed explanation of the project.
   - **Project MOTD (Message of the Day)** - The Project MOTD is displayed to all users in your projects. This message also may display to users when they log in to client nodes in your project.
   - **Project-specific host allocation time limits** - In this section, project administrators can control how long users in their project can allocate hosts for. If you set this value to 0 (zero), there is no limit on the time a host can be allocated.

   If you reduce this value, users' hosts in your project may be deallocated. If deallocations will occur as a result of your lowering of the maximum allocation time, you will be warned which systems will be affected, and given a chance to change your mind.

   - **Delete this Project/Undelete this Project** - If your project is not deleted and has no hosts, you will be given the option to delete this project from TeamForge Lab Management. If your project is deleted, you will be given the option to undelete it.

   > **NOTE:** Only Domain Admins will be able to delete and undelete projects.

To make it easier to start projects, provide project templates based on existing projects.

A project template is used to populate new projects with the structure and configuration of the Source project. It can also include the actual content of the project it is based on, such as tasks, tracker items and documents.

When you create a project template, it is available only to projects on your own site. To make it available more widely, you can export the project template and share it with others. Ask your CollabNet representative for more information about doing this.

The access settings for a project template are the same as the access settings for the project on which the template is based. For example:

✓ If you create a template from a project that has hidden applications, any project you create from that template will have the same applications hidden.

✓ If you create a template from a project that is private, any project created from that template will also be private.

1. Set up the project that will serve as the basis for the new project template.

   > **TIP:** If you need a clean project template, you may want to create a new project specifically for this purpose.

2. Click **Project Admin** from the **Project Home** menu.
3. On the **Project Settings** page, click **Create Project Template**.
4. Write a name and description for the template.

   > **TIP:** Consider the uses that you or other project managers might have for this project template, and include keyworkds in your description that are related to those uses.

5. Every new project includes the following project related components. Choose the items that you want to make available in new projects that are created from this template.

   - **Trackers** component into which you can add new trackers.

     > **NOTE:** When you select **Artifacts and their dependencies** from optional content, it copies the artifacts along with their associated tags.

   - **Planning Folder** into which you can add new planning folders.
   - **Tasks** into which you can add more task folders and tasks.
   - **File Releases** into which you can add packages and releases.
   - **Documents** folder into which you can add more document folders and documents.
   - **Wiki** into which you can add wiki pages to share project informaiton.
   - **Discussions** into which you can add discussion forums and respective discussions.
   - **Teams** into which you can add project teams and the members.
   - **Task Board** which you can configure to see a set of tasks for each project.
   - **Kanban Board** which you can configure to see different stages of your project development activities, specify workflow constraints for each state and map them to your tracker statuses.

   If you wish, you can include the actual project components from the current project's components.

6. Click **OK**.

The project template is created. The template is available to all users who have the required permission to create new projects. When the templates are used, the number of projects associated to individual template is displayed in the **projects created** field.

The template name and description appear on the *Template* tab of the **Projects** list, accessible from your personal navigation bar.

Related Links

- What is a project template?
- What is in a project template?

Organizing projects by categories can help users find what they need on a site quickly and easily. Project categories express the relationships among projects. If your project is in a category, it is visible to users browsing projects from the main **Project Categories** page.

Your project can be in one or more project categories.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click the *PROJECT CATEGORIZATION* tab. A list of all the categories that the project belongs to is displayed.
3. Click **Add**. The **Select Category** window shows all the available project categories.
4. If you find a project category that your project should be in, select that category and click **Add**.

Your project is now a member of the selected project category. You can repeat this process to add your project to any number of project categories.

You can make it easy for project members to use a wide variety of applications and sites from within TeamForge.

Link an External Application

To make an application or site outside of CollabNet TeamForge available in your project, create a linked application.

You can create as many linked applications per project as you need.

> **TIP:** Some sites employ page code that disables this feature. For example, some Google apps automatically generate user login data and append it to their URLs, which prevents them from opening in an iFrame.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **Tools**. A list of all the applications in the project is displayed on the **Project Tools** section.
3. Click **Add Tool**.
4. On the **Add Tool** page, enter a name for the linked application. This name will appear, with an icon you choose, in the **Project Home** menu.
5. Enter the server location or URL for the linked application.
6. If you are a TeamForge administrator, select whether you want to use single sign on for the linked application.
   - By default, only site administrators can edit (turn on/off) single sign on (SSO) for linked applications. However, you can set the site options token ONLY_SITE_ADMIN_CAN_EDIT_SINGLE_SIGN_ON to `false` to have both site and project administrators turn SSO on and off. For more information, see ONLY_SITE_ADMIN_CAN_EDIT_SINGLE_SIGN_ON.
   - If you use single sign on, TeamForge users can automatically log into the linked application.
   - If you do not use single sign on, users must log into the linked application using its native authentication system.
7. Click **Choose File** and select a `.gif`, `.jpg`, or `.png` file to serve as the new icon for the linked application. Make the image 25 pixels wide and 20 pixels high. This icon appears with the application name in the **Project Home** menu.
8. Click **Save**.

A button for the linked application is added to your **Project Home** menu. Clicking it launches the application in the main TeamForge project window.

Edit a Linked External Application

When the use patterns of a linked application change, you may need to change the way the application integrates with CollabNet TeamForge.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. From the list of applications, click the application you want to edit. The **Edit Tool** page is displayed.

3. Edit the name of the linked application, if required.
4. Edit the linked application URL, if required.
5. Enable or disable single sign on for the linked application. By default, only site administrators can edit (turn on/off) single sign on (SSO) for linked applications. However, you can set the site options token ONLY_SITE_ADMIN_CAN_EDIT_SINGLE_SIGN_ON to `false` to have both site and project administrators turn SSO on and off. For more information, see ONLY_SITE_ADMIN_CAN_EDIT_SINGLE_SIGN_ON.
6. Click **Choose File** and select a `.gif`, `.jpg`, or `.png` file to serve as the new icon for the linked application. Make the image 25 pixels wide and 20 pixels high. This icon appears with the application name in the **Project Home** menu.
7. Click **Update**.

Integrate an External Application into Your Project

To make an application or site outside of TeamForge available to your users seamlessly from inside your TeamForge project, bring it in as an integrated application.

If the application you want to use is not yet available for your project, ask your site administrator to set it up.

You can use as many integrated applications as you wish, after your site administrator has made them available.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **Tools**. The **Project Tools** section displays the list of all the current applications in the project.
3. Click **Add Tool**.
4. On the **Add New Tool** page, select the desired application. All the relevant parameters to configure the tool are displayed.
   1. If your site administrator has made it possible, specify a prefix for the resources created by the application you are setting up. A prefix enables TeamForge to provide handy links between objects managed by this application and other TeamForge objects. For example, if you are bringing in a blogging application:
      - You can connect a blog post with a TeamForge artifact using a special link called an "association."

- For each blog post, you can give readers a simplified address, known as a “go URL.”

> **NOTE:** For project-level associations and go URLs the application’s prefix is permanent after you save it.

2. Set any other configuration parameters you need and save your changes.

    - Click **Save** to return to the Project Tools page.

An icon for the integrated application is added to your **Project Home** menu. Clicking it launches the application in the main TeamForge project window.

How is an integrated application described?

An integrated application is described using two XML files - a deployment configuration file and an application configuration file - that provide information to TeamForge about the configuration options exposed by the application.

In TeamForge version 6.1.1 and later, you have the ability to configure some integrated application settings using the user interface. You can also export these settings in XML format and make changes. To edit configuration settings, you would upload the XML file containing the updates.

Integrated application settings

> **NOTE:** Some of the tags are internationalized so that the application will display languages based on the browser locale. See Internationalize Your Integrated Application for more information.

<name> This is the title of the integrated application. When the integrated application is added to a project, the button that appears on the project pages has this name. This name must be unique – you cannot use it for any other integrated application on the same TeamForge server.

This tag is used in both deployment and application configuration files.

<adminurl>

When an application has an administration screen for configuring its parameters, this field contains that URL. It is optional.

This tag is used in the deployment configuration file.

<baseurl>

This is the URL to which a user will be directed on clicking the integrated application button in a project.

This tag is used in the deployment configuration file.

<endpoint>

This is the SOAP endpoint for the integrated application. The endpoint contains the various methods exposed by the integrated application that are called during the lifecycle of TeamForge.

This tag is used in the deployment configuration file.

<gourl>

This indicates which URL must be used when an object id for an integrated application is specified (either via `Jump_to_id` or on the URL as `/sf/go/<objectid>`). This URL can support a couple of dynamic parameters.

- %o - The object id entered by the user will be dynamically replaced here.
- %p - The project id for the object entered will be dynamically replaced here.

For example, if the Go URL is http://go.tourl.com/tracking?id=%o and the object ID entered is XYZ123, then the URL will be replaced and redirected to http://go.tourl.com/tracking?id=XYZ123.

This tag is used in the deployment configuration file.

<category>

This is used to display the integrated application under a separate *Category* tab. It is an optional parameter that is manually inserted based on the requirement. If the category is not defined, the integrated application appears under the *Generic* tab.

This tag is used in the deployment configuration file.

<isbaseapp>

This validates whether the integrated application is a base application configured for the specified category. It is an optional parameter that is manually inserted based on the requirement.

This tag is used in the deployment configuration file.

<is-search-supported>

This validates whether the integrated application needs to be displayed as an object category in the drop-down list for search options in TeamForge such as **Jump to ID** and **Advanced Search**. It is an optional parameter that is manually inserted based on the requirement.

This tag is used in the application configuration file.

<config-parameters>

There can be any number of configuration parameters for an integrated application and they are displayed when associating the application to a project. These parameters are filled in by the project administrator and are available in the integrated application SOAP interface as configuration parameters. The integrated application gets a chance to validate these parameters and indicate back to TeamForge whether this project association is successful by passing in a “TRUE”. It can return a “FALSE” if it doesn’t want this project association to succeed. Each configuration parameter is placed inside the “param” tag, which can contain multiple elements to describe the parameter.

- **<title>**

  The internationalized title that appears for a project administrator to fill in while associating the integrated application to a project.

- **<name>**

  The Java variable under which the value for this parameter will be available on the integrated application.

- **<description>**

  The internationalized description that appears when a project administrator fills in or enters a configuration parameter.

- **<default value>**

  The default value for the parameter that will appear in the user interface during the association of an integrated application to a project.

- **<display type>**

  This is the type of display control used for the configuration parameter. We support “TEXT” for text fields, “CHECKBOX” for checkbox type controls, “RADIO” for radio buttons, and “SELECT” for select dropdowns. This field can also take an attribute that says what the value type for the field should be – whether it should be an “Integer”, “String” and so on. So if the field is expecting numbers, then entering “foo” as a value will throw a validation failure.

- **<option>**

  If the display type is “RADIO” or “SELECT”, then these fields contain the individual options available for the display controls. This will contain a “name” attribute that will be sent to the integrated application when that option is selected from the UI. The value of this option should be an internationalized field as it is the value visible to the user.

- **<editable>**

This specifies whether the configuration parameter should be editable once the integrated application is associated to a project. These configuration parameters are available when you add or edit an integrated project. If a parameter should not be “edited” post association, setting this to “false” will make it non-editable.

This tag is used in the application configuration file.

<description>

This is an internationalized string for the integrated application’s description. It contains information for TeamForge project and site administrators to know what the application does.

This tag is used in the application configuration file.

<id-pattern>

When trying to link to an integrated application id, this regular expression gets used for mapping. By default (if no value is provided), it looks for alphanumerical characters; in case you need specific characters to be matched (for example, JIRA, which has hyphens in ids), this value is used.

This tag is used in the application configuration file.

<page-component>

These settings are used for Project Content Editors. The integrated application content can become part of the standard Page Component data that appears in project home pages. The settings indicate the type of information that will be available from the integration application.

- **<input-type>**

  This is the input type control for an integrated app Page Component. We only support 2 types now. Either “select” so that the inputs can be shown from a “SELECT” dropdown and the users will be able to pick a value from there. Else, it can be a “text” where a simple “text” field will be entered for taking the user input.

- **<result-format>**

  This is the format in which the output of Page Component is returned. This can be a “list” which indicates that it will be a Table like output. The integrated app will send the results in an XML format and the Integrated app framework converts this into a list of records. The other option is “html”, where the output from the Integrated application is just displayed on the screen.

- **<page-component-description>**

  The description that will appear when you add an Integrated application Page Component (Link to the page where “ Add component” is available)

- **<page-component-title>**

  The title that will appear when adding an Integrated application Page Component (Link to the page where “PCE Add component” is available)

This tag is used in the application configuration file.

<permissions>

This is a collection of permissions that are exposed by the integrated application. There could be any number of such permissions. These permissions will appear as a part of the project’s roles (existing ones, as well as ones newly added) and can be assigned along with other tool permissions. You can map one of these permissions with a “dapMappedTo” attribute – this indicates the permission to be used when a user logs in without authentication (for example, for public projects). Typically, this is the permission to read data so that it doesn’t need a login name; it varies from one application to another.

This tag is used in the application configuration file.

<prefix>

If the “require-per-project-prefix” attribute is false, the value of this tag is used for identifying the integrated application in Go URLs, associations, and linkifications. If the “require-per-project-prefix” attribute is true, the value is used only for the “Host” project. Each project must fill in its value as part of adding the integrated application. Click here for steps to add integrated applications to a project.

This tag is used in the application configuration file. The prefix can contain alpha-numeric characters and cannot be more than six characters in length. For more information about prefix, see How does an integrated application interact with other TeamForge tools?

<require-per-project-prefix>

An integrated application can indicate to TeamForge whether the object ids that it generates are uniquely identifiable across the entire application (if yes, the value for the attribute is “false”) or whether they need to be project-specific (in this case, the value for the attribute is “true”). If an integrated application needs per-project prefix, you must enter the prefix value when the integrated application is added to a project.

This tag is used in the application configuration file.

<require-scm-integration>

This indicates whether SCM commits need to be vaildated. Some applications might have business rules which indicate that a commit can be made only if certain conditions are met. If the integrated application has any such rules, the value for the attribute should be “true”. There are also a couple of methods to be implemented in the SOAP endpoint.

This tag is used in the application configuration file.

<require-page-component>

Some integrated applications choose not to expose details as Page Components. For those that don't, set this tag to "false" and for those that do, set it to "true". If the value is "true", you must provide the "page-component-details" tags as well.

This tag is used in the application configuration file.

<servicetype>

TeamForge 6.1.0 and earlier releases supported only SOAP as the mechanism to talk from TeamForge to the integrated application. TeamForge 6.1.1 and later support REST calls. The servicetype tag indicates whether the protocol used for communication is REST or SOAP.

This tag is used in the deployment configuration file.

For examples of how these tags are used in the integration of the Pebble bogging application, see pebble-dep.xml and pebble-app.xml.

✓ The associations between CollabNet TeamForge and an integrated application can be created only from CollabNet TeamForge to the integrated application and not vice-versa.

✓ To associate an object in an integrated application from within CollabNet TeamForge, use the `[<prefix_objectid>]` format. Successful associations appear hyperlinked.

✓ Each integrated application displays its prefix on moving the mouse over the application name in the tool bar.

How does an integrated application interact with other TeamForge tools?

When you integrate an external application into your TeamForge site, the application can take full advantage of object IDs, links and Go URLs.

To look at how this works, we'll use the Pebble application as an example. Pebble is a blogging tool that you can quickly integrate with TeamForge.

Object IDs

Integrated application object IDs are of the form "prefix_objectId". Object IDs uniquely identify a TeamForge object so that you can access and use it in different contexts. For example, to get to artifact `artf1234` quickly, you just enter `artf1234` in the Jump To ID box. In the Pebble tutorial application, the date of a blog post, in YYYYMMDD format, is used as the object ID.

A prefix is an alphanumeric string attached to the beginning of an object ID that TeamForge uses to manage object IDs from different tools. For example, in the Pebble app, `<prefix>_20100601` gets you a page showing all the blog posts in the project that were published on June 1, 2010.

In an object ID such as “prefix_objectId”, the “prefix” is case-insensitive, whereas the “objectId” is case-sensitive. For example, the two object IDs, “PT_SC1” and “pt_SC1”, refer to the same object in TeamForge. Whereas, the two object IDs, “PT_SC1” and “PT_sc1”, refer to two different objects in TeamForge. Here, `PT` and `pt` are case-insensitive and the `SC1` and `sc1` are case-sensitive.

The prefix can either be the one specified when an integrated application is added to a project by project administrator, or the one in the XML Application configuration file depending on the “require-per-project-prefix” setting. The “require-per-project-prefix” setting can be true or false. If it is false, each project integration would not need to provide a project prefix; so the one provided in the XML application configuration file takes effect. If the “require-per-project-prefix” setting is true, a prefix needs to be provided by the user during every project association.

The amount of information the prefix carries depends on the kind of application you are integrating into your TeamForge site.

- With applications that use object IDs, such as Project Tracker and JIRA, you can identify the project that the object belongs to from its object ID.
- For applications that don’t have uniquely identified objects, or don’t have the notion of “project,” such as MoinMoin or Review Board, you can choose a prefix that’s specific to the project where the integrated tool is used.

Setting up Multiple Prefixes for Integrated Applications

At times, you may want to use more than one prefix for an integrated application such as the TeamForge EventQ. It is possible to have multiple prefixes set up for integrated applications. You must have the prefixes, separated by commas, included in the XML application configuration file and upload the file to your TeamForge site. For more information about uploading the application configuration file, see Edit an integrated application.

Consider the following while setting up multiple prefixes for an integrated application:

- Prefixes, once set up using the XML application configuration file, cannot be modified.
- A prefix can be up to six alpha-numeric characters in length. However, the combined length of all the prefixes cannot exceed 128 alpha-numeric characters.
- The “require-per-project-prefix” must be set to `false` in the application configuration file. In case it is set to true, an error message appears when you upload the application configuration file.
- Do not use existing prefixes. You cannot upload an application configuration file consisting of one or more prefixes already in use in TeamForge.

Go URLs

Go URLs allow a user to get to a particular object ID with a short, handy URL. To use this for Pebble, construct a URL like this: `https://mysite.com/sf/go/<prefix>_<date in format YYYYMMDD>`.

For example, if the Pebble tool in your project has the prefix PA, and you want to send someone all the blog posts published on app June 1, 2010, send them this link: https://mysite.com/sf/go/PA_20100601.

Associations

The object ID can be used to associate objects with other TeamForge objects. For example, if you want to associate a document with the blogs published on June 1, 2010, go to the document's *Associations* tab and add an association to `PA_20100601` as the object ID.

Automatic Links

When you type text of the format `<prefix> _<date in YYYYMMDDD>` in any TeamForge text field, the text is converted to a link. When you click the link you see the blog posts for that date, if any.

Export an Integrated Application

TeamForge site administrators can export an integrated application's variables in XML format and have them available for editing.

1. Go to **My Workspace > Admin**.
2. Click **INTEGRATED APPS** from the **Projects** menu.
3. To export the integrated application, select the integrated application name.
4. From the **View Integrated Application** page, click **Export**.

Hide / Remove a Linked or Integrated Application from a Project

To stop making an application or site outside of TeamForge available to your users from inside your TeamForge projects seamlessly, hide or disintegrate an application.

> **NOTE:** You may have integrated several external applications per project to maximize the integrated applications feature's utility. However, because each integrated application adds an icon to the project

navigation bar, a large number of integrated applications can cause horizontal scrolling. Consider removing the linked or integrated applications that are not in use.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **Tools**.
3. To hide an integrated application from the list of tools displayed, clear the **Visible** check box and click **Save**. The selected integrated application is hidden for the project. The icon of the hidden linked or integrated application disappears from your **Project Home** menu.

   WARNING: Hiding a linked or integrated application, hides all associations to the application too. It also hides the component type in the Project Home's **Create Component** page.

4. You can also click the **Delete** icon (🗑) to delete a linked or integrated application permanently from your project.

   WARNING: Deleting an integrated application (such as Review Board, EventQ or Binaries) means a permanent removal of the application including all related integrated application data from your project. Exercise caution before deleting an integrated application. For example, deleting Review Board from a project will delete any and all review requests, reviews, diffs, or other data associated with the Subversion repositories in the project.

Before a person can work on a project, you have to make him a member of the project.

You can make any registered user on your TeamForge site a project member. You can assign roles to the user at the same time.

NOTE: Project members, roles and the associated permissions can be inherited via project hierarchy and reused in subprojects. If your project is a subproject of any other project, you may have inherited some roles, project members or user groups.

1. Click **Project Admin** from the **Project Home** menu.
2. Click **User Membership** from the **Project Admin** menu.
3. Click **Add**.
4. On the **Add Users** page, find the users you want by one of these methods:
   - Under **FIND USERS**, filter the list of site users eligible to join this project. You can filter by full or partial name or user name.
     Search text is not case-sensitive.

- Browse the list of registered users on the site. Sort them by name, user name, email address or membership status.
  The inherited project members continue to hold the inherited roles with corresponding permissions, as specified in the source projects. If a site has a great many users, you must filter them first to narrow down the list. This helps avoid slowing down the system.

5. Select the users you want to add.

6. Under **Assign Roles**, select the roles you want the users to have.

   You can select any global project role, role created just for this project, or inherited role that is available.

   If your project is a child project of another project, the members of the parent project become inherited members of your project. The user roles specified in the parent project are available in your project provided the role inheritance is not prevented. If you assign a role in your project to a user, that user becomes a direct member of your project.
   If you prefer, you can skip this step and assign roles later on the **Project Admin > Permissions** page. Note that using the **Assigned Project Members** page, you can assign roles only to the direct project members.
7. Save your changes.
   - Click **Save** to return to the **User Membership** page.
   - Click **Save and Add More** to keep adding users.

Related Links

- Remove a User from a Project
- Handle a Request for Project Membership
- Handle a Request to Leave a Project

When you remove a user from a project, all items such as tasks and tracker artifacts that were assigned to the user are re-assigned to None.
If your project is a subproject of any other project, your project may have inherited some users from the parent project. To remove an inherited user, you must go to the parent project where that user is a direct member.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Admin Menu**, click **User Membership**.

3. From the list of current project members, select the user you want to remove and click **Remove**.

Related Links

- Add Users to a Project
- Handle a Request for Project Membership
- Handle a Request to Leave a Project

A registered CollabNet TeamForge user can ask to be a member of a project. As the project administrator, it's up to you to approve or reject such requests.

When a CollabNet TeamForge user submits a request for project membership, the request is placed in the **User Membership** section of the **Project Administration** page, pending approval by a project administrator. The request is also displayed in the **Items Pending My Approval** section of each project administrator's **My Page**.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **User Membership**, then click the *PENDING REQUESTS* tab.
3. Under **Users Requesting to Join Project**, select the user you want to approve or reject.
   - Click **Approve** to approve the request and add the user to the project.
   - Click **Reject** to deny the request.
4. To view details about the user or add a comment before approving or rejecting the request, click the user name. This is optional.

The user receives an email notification when the request is approved or rejected.

Related Links

- Add Users to a Project
- Remove a User from a Project
- Handle a Request to Leave a Project

A TeamForge user who wants to leave a project must submit a request. The project administrator can approve or reject the request.

A request to leave is placed in the **User Membership** section of the **Project Administration** page, pending approval by a project administrator. The request is also displayed in the **Items Pending My Approval** section of each project administrator's **My Page**.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **User Membership**, then click the *PENDING REQUESTS* tab.

3. Under **Users Requesting to Leave Project**, select the user whose request you want to approve or reject.
   - Click **Approve** to approve the request and remove the user from the project.
   - Click **Reject** to deny the request.
4. To view the user details or add a comment before approving or rejecting the request, click the user name. This is optional.

The user receives an email notification when the request is approved or rejected.

Related Links

- Add Users to a Project
- Remove a User from a Project
- Handle a Request for Project Membership

To control which operating system profiles the users in your project can build hosts with, adjust the settings in your profile library.

1. On the **Profile Library** page, click **Add/Remove Profiles**.

2. In the list of all potential profiles that you can add to your project, select the profiles you wish to be allowed in your project and click **Save Changes**. Your changes take effect immediately.

> **NOTE:** To revert any changes you have made, click **Cancel** or **Reset**.

You can coordinate work among multiple projects; enable the user, user group and role inheritance by adding a common parent project to several projects.

A parent project is the base from which a subproject's members, user groups and roles, with their corresponding permissions, are derived. A subproject can inherit project members, user groups and roles from its parent project.

You can create a parent project to track and manage several smaller assignments as subprojects.

When you define users and roles with specific permissions in a project, those users and roles are passed down to any subprojects that belong to that project. This helps you avoid the repeated effort of defining users, user groups and roles across projects.

✓ A subproject can have only one parent. You can change or remove that at any time.

✓ You must be a project administrator or site administrator to add, edit or remove the parent projects.

1. Click **Project Admin** from the **Project Home** menu.

2. On the **Project Settings** page, click **Add Parent**.
3. Choose a parent project that it makes sense for your project to belong to, and click **Update**.

> **NOTE:** You cannot add a parent project for the `Look` project as it is a special project in itself.

When a subproject grows beyond its original scope, you may want to make it a standalone project or move it to a different project hierarchy.

By removing the association with a parent project, you can manage the subproject as a separate project.

> **NOTE:** Only one parent project can be selected for a subproject. However, the parent project can be changed or removed, as required.

1. Click **Project Admin** from the **Project Home** menu.
2. On the **Project Settings** page, click **Edit Parent**.
3. Change the parent project as required. You can be a project administrator or a site administrator to change or remove a parent project. As a project administrator, you can remove or change a parent project only if you have administrator permissions for both the projects that are being linked.

   > **NOTE:** A parent project can be removed or changed only when its members, user groups and roles are not in use in any other project. In other words, you can not remove/change a parent project while its members, user groups or roles are in use in any other project.

   - On the **Choose a Parent Project** page, select the desired parent project and click **Update**.

     > **NOTE:** If the project hierarchy exists, the project and its subprojects are moved only under the project from which members, user groups or roles are inherited. If project hierarchy does not exist and no inheritance is in use, the project is made a ‘Root’ project.

   - Click **Remove Parent** in the **Project Settings** page to remove the association with parent project.
   - On the pop-up message box, click **OK**, if you wish to make the project a ‘Root’ project.

   The project hierarchy is changed or removed in accordance with role based access control and inheritance rules.

Project members and other users need to know how to interact with your project. You can give them the information and tools they need by creating and maintaining a project website tailored to them. You can build your project home page from scratch and assemble it from building blocks provided by TeamForge.

For quick, flexible site building, use the ready-made web page components that come with your TeamForge project site.

There are components for showing text, wiki pages, charts and graphs, and other purposes. These components make it easy to put together a sophisticated project site in little time.

Create a Project Page

To provide information and functionality to people viewing your project, build one or more project pages.

1. Go to the page to which your new page will belong.
   Any project page can have sub-pages belonging to it. A page that belongs directly to the project home page is called a *top-level page*.
2. Click **Configure: On**.
3. Choose where your new page will fit in your project's structure.
   - To create a page just under the project home page, click **Add top-level page**. A top-level page's title is always visible in the navigation tree at left.
   - To create a page under the page you are on right now, click **Add sub-page**.
4. Give your new project page a title. Keep the title brief and descriptive.
5. Choose who can see this page.
   - Your choice will apply to all subpages that you create under this page.
   - To show this page to anyone with the necessary permissions, select Visible. For example, if you have defined a group of users who have access to your project, your new project page is visible only to those users. If your project is open to the public, anyone in the world can see it. Use this option when the information on this page is ready for a wide audience.
   - To show this page only to users with the project administrator role, select Hidden. Use this option if you are drafting content that you aren't ready to share yet, or want to share only with other project managers.
6. Click **Save**.
7. Click **Configure: Off**.

Now you are ready to build functionality into your project page with components such as text, news or tracker queries.

Control Access to a Project Page

Before you put information or functionality on your project page, make sure it is accessible to the people it is intended for.

> **NOTE:** This is only relevant if your page is not hidden. If you page is hidden, users who are not project administrators cannot see the page even if their role-based permissions would allow it.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Settings** page, click **Permissions**.
3. Click the role to which you want to give access to your project page.

   > **TIP:** If the appropriate role does not exist, you must create it first.

4. On the **Edit Role** page, click **Project Pages**.
5. Under **Project Pages Permissions**, select the pages that users with this role can see and edit.
   - To enable users with this role to create, read, and modify all project pages, select the **Project Pages Admin** permission.
   - To allow users with this role to see pages but not edit them, select the appropriate page in the **View** section.
   - To allow users with this role to modify the contents of a text component project pages, select the appropriate pages in the **Edit Text Content** section.

   > **NOTE:** The project home page is always visible to any user who is authorzied to see the project.

Reorder Project Pages

Facilitate your users’ experience in your project by putting your project pages in an order that matches their needs.

1. Click **Configure: On**.
2. Select a project page and click **Edit Structure**.
3. Rearrange the structure of this page in any of these ways:
   - Select a page, and use **Cut** and **PastE** to place it in a different location.

> **NOTE:** When you move a page, all of its sub-pages will also be moved.

- Drag and drop a page to a different position.
- Click **Add New Page** to add a sub-page to the current page.
- Select one or more pages and click **Delete** to remove them.

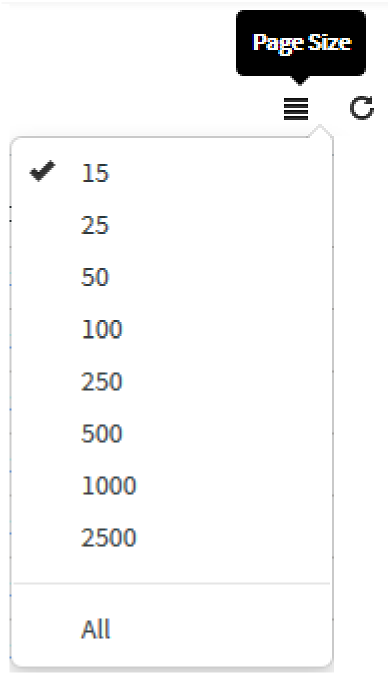
4. Click **Save Changes**.
5. Click **Configure: Off**.

Add Reports to a Project Page

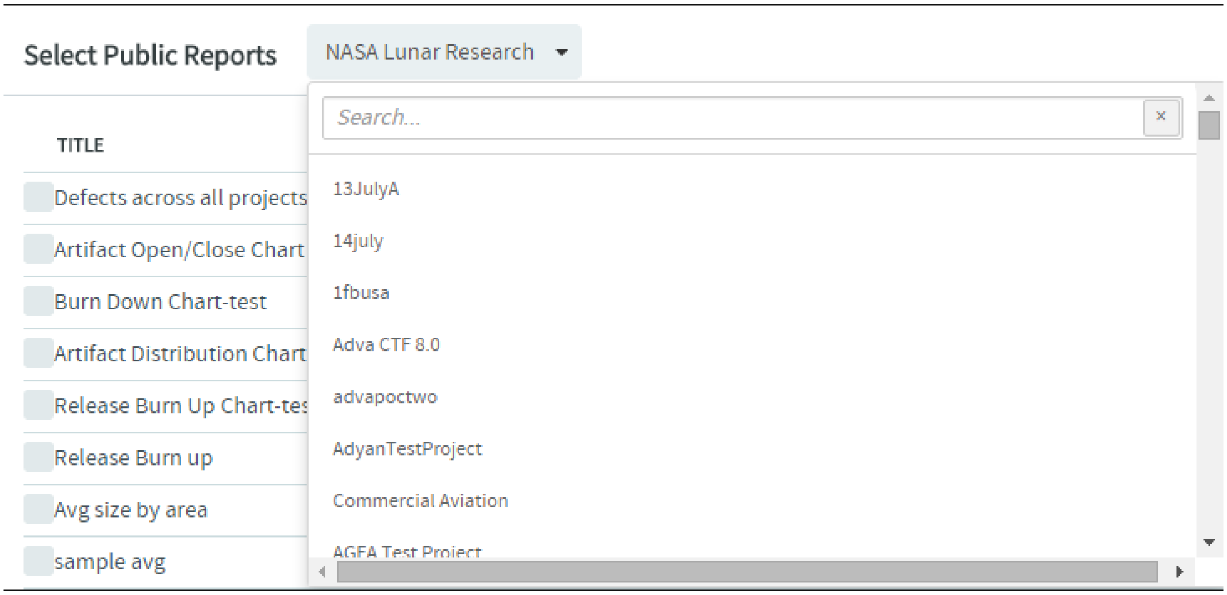
Add one or more reports to your project home page to publish your project status to other project members. You can add only reports of type *Public* to your project home page.

For more information about TeamForge reports, *Public* and *Private* reports, see [Reporting in TeamForge] [\reports.html].

1. On the project page, click **Configure: On**.
2. Click **Add New Component**. The **Create Component** page appears.
3. Type a title for the report component.
4. Select **Reports** component type. A list of *Public* reports for the project in context is shown.
5. Select **Visibility** and **Location**.
6. Select one or more *Public* reports to add to the project page.
    - You can add up to thress reports per Reports component.
    - The *Public* reports list may span over multiple pages if the number of reports exceeds **Page Size**. You can navigate through pages by clicking the page numbers at the bottom of the list.
    - However, you can change the **Page Size**.



7. If you would like to add reports from other project(s) of which you are a member, select the required project from the **Select Public Reports** drop-down list.



The list of reports existing in the selected project is displayed from which you can select the relevant reports.

> **NOTE:** Private reports and Table reports are not displayed in this list. You can add a maximum of three reports per Reports component.

8. Click **Save**.

The selected reports are added to the project page.

Hide a Project Page

While you are preparing a project page, you may wish to keep it hidden.

You can hide a page from everyone except other project adminsitrators.

1. Click **Configure: On**.
2. Click **Edit Page Properties**.
3. For **Visibility**, select **Hidden** and click **Save**.

Now no one who does not have the Project Admin role can see the page, even if they have access to the project and to this page's parent page.

When you are finished building your page, don't forget to switch the page to "Visible" so that the intended users can see it.

Rename a Project Page

As a page's focus evolves, it's a good idea to rename it to match its changing function.

1. Click **Configure: On**.
2. Click **Edit Page Properties**.
3. Change the **Title** as appropriate to its current function, and click **Save**.
4. Click **Configure: Off**.

Create a Project Page Component

Put information and resources in your users' hands with project page components.

For example, to let people know about important new developments, create a news component. To enable project members to find tracker items quickly, create a query component.

1. On the project page, click **Configure: On**.

2. Click **Add New Component**.

3. On the **Create Component** page, give your new component a title. Keep the title brief and descriptive.

4. Select one of the following component types that suits your need.

    - **Text** - Write free-form messages, reports or rants, in plain text or HTML.
    - **Reports** - Add various reports and charts to your project page.
    - **Documents** - Let project members exchange and review documents from the project page.
    - **Wiki PagE** - Open up your project page to two-way communication.
    - **Tracker Search Results** - Make saved search results available from the project page.
    - **Tracker Metrics** - Add charts about tracker metrics to your project page.
    - **Project News** - Maintain a journal or blog about your project, share information and make announcements.
    - **Project Statistics** - Show visual measures of your project activities on the project page.
    - **Subprojects** - Add a list of subprojects to your project page.

5. Choose who can see this component.

    - To show this component to anyone with the necessary permissions, select **Visible**.

    For example, if you have defined group of users who have access to trackers in your project, a query component will be visible only to those users. If your project's trackers are open to anyone, all users who view this project page will see the query component. Use this option when you are sure the component is ready for general use.

    - To keep this component under wraps until you are ready to show it, select **Hidden**.

    Now only users with the project administrator role can see this component. Use this option if you are drafting content that you aren't ready to share yet or want to share only with other project managers.

6. Select one of the locations, `Top of page` or `Bottom of page`, where the component shows up on the project page.
7. Depending on the component type you selected, set the properties of the component.

Component type	In the Properties of this component section...
Text	• Type your free-form messages, rants or announcements in the text box. • Click **Save**.
Reports	For more information about adding reports, see Add Reports to a Project Page and Reporting in TeamForge
Documents	• Select a folder from the list to display its contents on the project page. • Click **Save**.
Wiki Page	• Type a title for the wiki page. • Click **Save**.
Tracker Search Results	Make sure one or more shared tracker searches are available to add to the project page. For more information about sharing saved tracker searches, see Share a Saved Tracker Search. • Click **Add Saved Tracker Searches**. • Select one or more shared tracker searches from the *Select from Shared Tracker Searches* window. • Click **Add Selected**. • Select the number of rows of the search results to display from the **Display Rows** drop-down list. • Click **Save**.
Tracker Metrics	• Select the number of charts from the drop-down list. You can add up to three tracker charts to your project page. • Select one of the chart types: *Burndown*, *Open by Priority* or *Open vs Closed*. • Select a data source for your chart, a tracker or a planning folder. • Click **Save**.
Subprojects	• Select the number of subprojects to be displayed on the project page. • Click **Save**.

Edit a Project Page Component

You may want to make some changes to your project page component to make it more useful.

1. On the project page, click **Configure: On**.

2. In the title bar of the project page component you want to edit, click the edit icon 🔧 .

3. Update the title as appropriate to its current functiona, if required.

4. Select the **Hidden** option for **VISIBILITY** to hide the page while you work on it. You can hide a page component from everyone except other project administrators.

5. Change the following settings depending upon the project page component that yor are editing.

    - For a *Sub-projects* component, you can change the number of projects to be displayed.

    - For a **Documents** component, you can change the folder locaiton.

6. Click **Configure: Off**.

The project page component is displayed with modified content and/or settings.

Reorder project page components

To make things easy for your project members, lay out your project components in a useful order.

1. On the project page, click **Configure: On**.

2. In the title bar of the project page component you want to move, click the up arrow ⬆ or the down arrow ⬇ .

3. Click **Configure: Off**.

Hide a Project Page Component

While you are preparing a project page component, you may wish to keep it hidden.

You can hide a page component from everyone except other project administrators.

1. On the project page, clic =k **Configure: On**.

2. In the title bar of the project page component you want to hide, click the edit icon 🔧

3. Select the **Hidden** option for **VISIBILITY**, and click **Save**.

4. Click **Configure: Off**.

Now no one who does have the Project Admin role can see the component, even if they have access to the project and to this page. Uses with the Project Admin role can see this component only when **Configure** is set to **On**.

Rename a Project Page Component

As a page component's focus evolves, it's a good idea to rename it to match its changing function.

1. On the project page, click **Configure: On**.
2. In the title bar of the project page component you want to rename, click the edit button 🔧 .
3. Update the title as appropriate to its current function, and click **Save**.
4. Click **Configure: Off**.

Delete a Project Page Component

When a component is no longer useful, remove it from the page to avoid distracting users.

1. On the project page, click **Configure: On**.
2. In the title bar of the project page component you want to delete, click the **Delete this Page** button.
3. Click **Configure: Off**.

Create a Custom Project Home Page

To fully custom control your project website pages, build your own HTML and check it into the project's publishing repository. You can link to these pages as you would to any normal HTML page.

If you decide to hand-code your project hom page, the page works like any HTML page.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Settings** page, select **Show custom web page** from **PROJECT HOME OPTIONS**.
    - **Show default project pages and components** is selected by default.
    - Site administrators can restrict public access to Publishing Repositories at the site level and you cannot view or modify **PROJECT HOME OPTIONS** in such case. See DISABLE_REMOTE_PUBLISHING site-options.conf variable for more information.

3. Add the `index.html` file to the `publishing/www` Subversion repository. The content of your index.html shows up immediately.

> **NOTE:** When you switch to a hand-coded home page, you can still access any project pages you have created.

4. Click **Save**. Your cutome **Project Home** page is now active.

> **NOTE:** If your project has any subprojects, the subprojects are also listed. To see them, make sure your `index.html` file exists and is in the right repository location.

See What's in Your Publishing Repository

You can view the files in your TeamForge site's `publishing` repository through relative URLs.

For example:

1. Endter the URL: `https://forge.collab.net/sf/projects/sampleproject/index.html` in the address bar to display contents of the `index.html` file. In this case, your `index.html` file must be in the `publishing` repository under the *www/sampleproject* path. The contents of the `index.html` file are displayed in the **Project Home** page.

2. Enter the URL: `https://forge.collab.net/sf/projects/sampleproject/roles/roledetails.html` in the address bar to display contents of the `roledetails.html` file. In this case, your `roledetails.html` file must be in the `publishing`repository under the *www/sampleproject/roles* path. The contents of the `roledetails.html` file are displayed in the **Project Home** page.

To control who can access your project, consider the purpose of your project and the appropriate type of user.

Control Access by User Role

Project administrators use existing global project roles or create and assign roles to project members to define what those project members can do in the project.

If you choose to use existing global project roles, you can quickly assign relevant roles to your project members. It is a good practice to create a new role in your project only when a suitable global project role is not available.

NOTE: If your project is a subproject of any other project, your project may have inherited some roles from the parent project. To work with the properties and permissions associated with inherited roles, you must go to the parent project where those roles are specified.

Create a User Role

A role defines the applications that project members with that role can use, and the specific things project members can do in each application.

Any project administrator can create and assign a role.

TIP: It is a good idea to check the existing global project roles via `Project Admin > Permissions > Roles > View: global project roles` before creating any new role in a project.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **Permissions**.
3. On the *ROLES* tab, select **View: Roles Created For a Project**.
4. Click **Create**.
5. On the **Create Role** page, write a name and description for the role.
6. To let this role be automatically added to any private subprojects of this project, clear the **PREVENT INHERITANCE** option.

   By default, roles are inherited by public and gated subprojects, but not private subprojects.

   TIP: If this project is a subproject of any other project, you may already have inherited some roles.

7. To allow project members to be able to request this role, select **PROJECT MEMBERS CAN REQUEST THIS ROLE**. Project members can submit requests for requestable roles, which the project administrator can approve or reject.

   TIP: Select GRANT AUTOMATICALLY ON REQUEST to skip the need to approve or reject a request.

8. Click **Create**. The role is created. The **Edit Role** page appears.

9. For each application listed on the **Role Permissions** page, select the permissions and resources you want to make available to users with this role.

    > **TIP:** You can specify the permissions for Binaries and integrated applications here too.

    > **NOTE:** You can specify access to individual top-level folders, but not to specific subfolders.However, in the case of documents, you can specify access to individual subfolders as well.

10. Click **Save**.

The role is created. You can assign it to project members at any time.

> **TIP:** Based on the earlier settings, a project member may be able to submit a request for the role.

Create a Project Administrator Role

The project administrator is responsible for managing the project's users and roles.

The project creator is assigned the Founder Project Admin role, a special role granting all project and application administration permissions for the project. You can transfer the Founder Project Admin role to another user.

If the project creator is a CollabNet TeamForge administrator, no Founder Project Admin is created.

> **NOTE:** By default, project adminstrators do not have application administration permissions, such as Tracker Admin or Task Admin. Application administration permissions can be included in a project administrator role, but must be assigned separately.

1. On the **Role Permissions** page, select **Project Admin Permissions**.

    > **NOTE:** If this is an inherited role, you can not edit the permissions associated with it. You can edit the project members and user groups to whom this inherited role is assigned.

    - If you want the role to contain only the project administrator permissions to manage users and roles, **Project Admin Permissions** is all you need to select.

- If you want the role to contain additional application administration or other permissions, check the additional permissions.

2. Click **Save**.

All project members assigned this role have project administrator permissions to manage the project's users and roles.

Change a Role

If users need to do things that are not allowed by a role you have assigned to them, you may need to change the permissions associated with that role.

When you edit a role, all project members with that role get the updated permissions automatically.

> **NOTE:** If your project is a subproject of any other project, your project may have inherited some roles from the parent project. To work with the properties and permissions associated with inherited roles, you must go to the parent project where those roles are specified.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin** menu, click **Permissions**.
3. From the list of roles created for this project, click the role you want to edit.

   > **NOTE:** You can assign a role to direct project members and user groups of the project regardless of whether the role belongs directly to this project or is inherited from a parent project.

4. On the **Role** page, make the changes you need.
   - To edit the title or other role details, click **Edit**. Make the required changes and click **Update**.
   - To edit the role's permissions, choose an application from the left side of the page and select or deselect permissions and resources.
   - To edit the project members to whom the role is assigned, click **Assigned Project Members**.
   - To edit the user groups to whom the role is assigned, click **Assigned Groups**.
5. Click **Save**.

The role is modified.

Give Roles to a Project Member

A project member can have any number of roles. As project administrator, you must assign each project member’s roles.

Permissions are cumulative. The project member has all of the access permissions allowed by all of the assigned roles, plus any permissions that may have been assigned globally using application permissions.

> **NOTE:** If your project is a subproject of any other project, you may have inherited some roles, project members or user groups from the parent project. You can assign any role to any project member, regardless of whether it’s a global project role, or the role belongs directly to the project, or is inherited from a parent project.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **Permissions**, then click the *User-Role Matrix* tab. Observe the users listed on the left and all the available roles (global, direct and inherited) on the right.

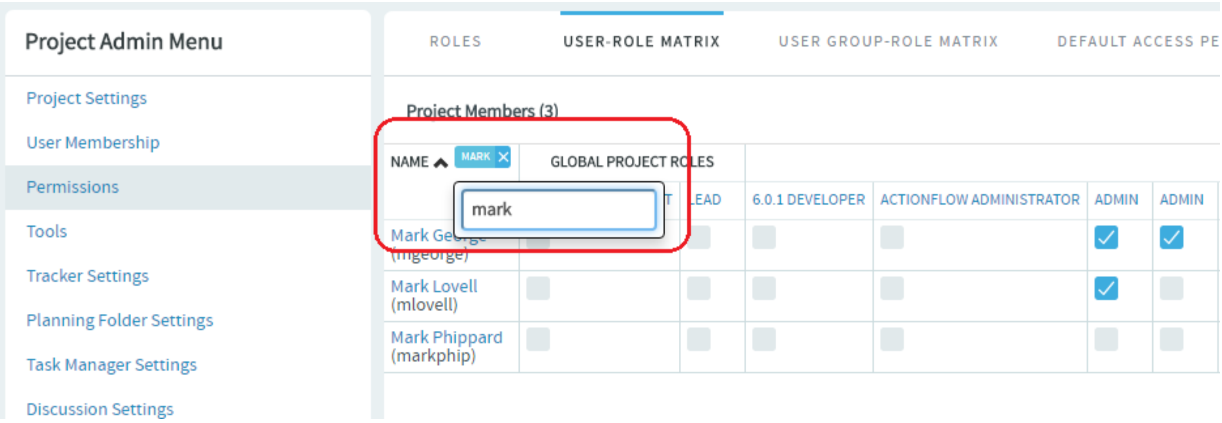
   Users can be assigned global, direct and inherited roles. Inherited users have all the permissions that they have in the parent project.

   ✓ If you assign another inherited role to an inherited user, that user gets both the roles.

   ✓ If you assign a role to a user in a project, that user becomes a current member in that project.
3. Select roles for each project member.

   > **NOTE:** A user’s license type also influences what the user can see and do on your site. A user’s license type supersedes any role assignments. Ask your site administrator how many licenses of each kind are available for your users. For more information, see How do TeamForge Licenses Work?.

   The **USER_ROLE MATRIX** page is also equipped with a smart search (filter) function that makes it easy to filter a user by name and assign roles. Use it to quickly search for a user to which you want to assign one or more roles.



NAME	GLOBAL PROJECT ROLES					
	T	LEAD	6.0.1 DEVELOPER	ACTIONFLOW ADMINISTRATOR	ADMIN	ADMIN
Mark George (mgeorge)		□	□	□	☑	☑
Mark Lovell (mlovell)	□	□	□	□	☑	□
Mark Phippard (markphip)	□	□	□	□	□	□

4. Click **Save**.

The roles are now assigned to each project member.

Give a Role to Multiple Project Members

A role can be assigned to many users at once.

The user-role matrix provides a convenient way to add project members to a role, but it can become unwieldy if the project has a large number of users or roles. When that is the case, try assigning roles to multiple users at one time.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **Permissions**.
3. Click the name of the role that you want to assign to project members.
4. On the **Role Permissions** page, click the *Assigned Project Members* tab. The **Assigned Project Members** page shows all users who currently have the role.

> **TIP:** You can assign the role only to the direct project members of the project.

> **NOTE:** A user’s license type also influences what the user can see and do on your site. A user’s license type supersedes any role assignments. Ask your site administrator how many licenses of each kind are available for your users. For more information, How Do TeamForge Licenses Work?

5. Click **Add** and use the **Find a User** window to move your desired users into the **Selected Users** list.

> **TIP:** You can search by full or partial user name or full name to find the right project members.

6. Click **OK**.

The project members are now assigned the role.

Handle a Role Request from a Project Member

A project member in CollabNet TeamForge can ask to be granted a role. As the project administrator, it’s up to you to approve or reject such requests.

When a CollabNet TeamForge project member submits a request for a role in a project, the request is placed in the **User Membership** section of the **Project Administration** page, pending approval by a project administrator. The request is also displayed in the **Items Pending My Approval** section of each project administrator’s **My Page**.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **User Membership**, then click the *Pending Requests* tab.
3. Under **Role Requests**, select the user whose role request you want to approve or reject.
   - Click **Approve** to approve the request and assign the role to the user.
   - Click **Reject** to deny the request.

> **TIP:** To view the permissions granted via the role, click the role name, if required.

> **NOTE:** Before approving or rejecting a role request, you can check the roles already assigned to the user by clicking the user name and selecting the *Roles* tab.

The user receives an email notification when the request is approved or rejected.

Assign a Global Project Role on Request

As a project administrator you can edit a requestable global project role for your project. You can update the settings to immediately assign the role to a project member on request.

> **NOTE:** Note: Only site administrators or restricted site administrators with `Role-Edit` permission can edit global project role details. Project administrators can only edit requestable global project role’s grant settings for their projects.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **Permissions**.
3. From the list of global project roles, click the requestable global project role you want to edit.
4. On the **Edit Role** page, click **Edit**.
5. Select the **Grant Automatically on Request** option and click **Save**. Selecting this option enables the project member requesting the role to be assigned the same immediately, without waiting for an approval.

The role is modified for your project.

Assign Roles to a User Group

Enable multiple users to do something all at once by giving their group a role.

A user group can have any number of roles. Each member of the user group has all the access permissions allowed by all of the assigned roles, plus any permissions that may have been assigned by other methods, such as application permissions or individually assigned roles.

Roles and the associated permissions can be inherited. If your project is a subproject of any other project, you may have inherited some roles or user groups. You can assign any inherited role to any user group.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin** menu, click **Permissions**, then click the *Group-Role Matrix* tab. Observe the user groups listed on the left and all the available roles (global, direct and inherited) on the right.

   User groups can be assigned global, direct and inherited roles. The inherited user groups have all the permissions that they have in the parent project.

   ✓ If you assign another inherited role to an inherited group, the members of that group get both the roles.

   ✓ If you assign a role to a user group in a project, that user group becomes a direct user group in that project.

3. Select the roles you want for each user group and click **Save**.

> **RESTRICTION:** When you give a group access to a CVS or Wandisco Subversion repository, members of the group can view the repository but cannot do repository actions, such as commit and update. You must assign those permissions to users individually.

The roles are now assigned to each user group.

Assign User Groups to a Role

To manage permissions for a lot of groups or roles at once, try assigning user groups to roles.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **Permissions**.
3. To display existing global, direct or inherited roles, click the **Global project roles**, **Roles Created For a Project** or **Roles Inherited From Parent Project** option of **View**.

   > **NOTE:** If your project is a subproject of another project, you may have inherited some roles or user groups. The inherited role details like role name, description and the source project name are listed.

   You can assign direct user groups to a global/direct/inherited role.
4. Click the name of the role that you want to assign to user groups.
5. On the **Role Permissions** page, click the *Assigned Groups* tab. The **Assigned Groups** page shows all user groups that are currently assigned the role.

   > **TIP:** You can assign the role only to the direct user groups of the project.
6. Click **Add**.
7. Type some of the group’s name in the **Name (search)** box and click **Find**.
8. In the **Find a user group** window, select the user groups that you want to add, and click **Finish**.

> **RESTRICTION:** When you give a group access to a CVS or Wandisco Subversion repository, members of the group can view the repository but cannot do repository actions, such as commit and update. You must assign those permissions to users individually.

The user groups are now assigned the role.

View Users and User Groups Assigned to a Role

Before adding a role to a project member or user groups, see all the users and user groups who are assigned that role through inheritance.

A user or user group can have any number of roles. Roles and the associated permissions can be inherited via project hierarchy or project groups.

1. Click **Project Admin** from the **Project Home** menu.
2. On the **Project Admin** menu, click **Permissions**.
3. In the **View** drop-down, select **Global project roles** and from the results, a specific role.
4. In the *Assigned Project Members* tab, click the **View** drop-down and make a selection.
    - **Direct Members** displays the users who are directly assigned the role in the project.
    - **Inherited Members** displays the users who inherit the role from parent projects as well as project groups.
5. In the *Assigned Groups* tab, click the **View** drop-down and make a selection.
    - **Direct User Groups** displays the user groups that are directly assigned the role in the project.
    - **Inherited User Groups** displays the user groups that inherit the role from parent projects as well as project groups.

The roles are now assigned to each user group.

Control Access by User Class

To avoid having to create and assign a lot of similar roles for individual users, give access to applications to whole classes of users whenever possible.

For each application (tasks, documents, file releases, trackers, and discussion forums), you can assign permissions globally based on user type.

For example, if you know that you want all project members to be able to view and submit to all project trackers, set the application's permissions to reflect this. You need to configure these settings only once.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin** page, configure your project access settings and click **Next**.
3. On the **Edit Default Access Application Permissions** page, click the + symbol to expand the section for which you want to assign permissions.
4. For each application and resource, choose a user type from the drop-down menus.
   - All users of the selected type will have the specified permissions: `view`, `submit/view`, or `post/view`.
   - For discussion forums and trackers, you can also specify submit or post permissions.

   > **NOTE:** You can specify access to top-level folders, but not to specific subfolders.

   > **NOTE:** If you want to control access to an application or resource that is not displayed on the **Edit Default Access Application Permissions** page, you can do so using role-based access control.
5. Click **Finish**.

Control Access by Project Type

Projects can be open only to project members, open to everyone in the world, or something in between.

By default, all new projects are created as private projects, accessible only to project members. Your system administrator can change the default access level for new projects.

> **IMPORTANT:** Users who do not have access to a project cannot see it on the Home page, in the **All Projects** list, or in search or reporting results.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Admin Menu**, click **Permissions**, then click the *DEFAULT ACCESS PERMISSIONS* tab to see the project's current access setting.

3. Click **Edit**.

4. On the **Edit Default Access Permissions** page, select the kind of access you want to allow to your project.

    - **Private** - Project members only.

    - **Gated community** - Project members and unrestricted users.

    - **Public** - All users.

    > **NOTE:** If the option is not available, your system administrator has prohibited changing the access levels of projects on the site.

Allow Users to See Other Users' Roles

You can enable some project members to view the roles assigned to other project members.

For example, if your project includes both core team members and consultants, you may want to restrict full visibility of user details to the core team members.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Admin Menu**, click **Permissions**.

3. On the *ROLES* tab, click the role you want to edit. For example, if you have divided project members into a "Core Team" role and a "Consultant" role, click the "Core Team" role.

4. Under **Project Admin Permissions**, select **View User Membership**.

Now only project members who have the role you edited can see the roles held by other project members.

> **NOTE:** They still can't see the actual permissions included in those roles, unless they have Project Admin status.

Lock or Unlock a Project

To ensure that no changes occur in a project while you are collating or migrating project data, lock the project. You must have project administration permissions or be a site administrator to lock or unlock a project.

To lock or unlock a project in TeamForge, go to **Project Settings** and lock/unlock the project.

> **NOTE:** A locked project does not allow any member (including project administrators and site administrators) to make any changes to the project. Besides that, a locked project can not be set as the parent project for any other project and tasks like adding, editing or deleting integrated applications are also not allowed.

Click **PROJECT ADMIN** from the **Project Home** menu.

The project is locked or unlocked as desired. The lock (🔒 Locked) icon appears on all the project pages while the project is locked.

> **NOTE:** If a locked project has an integrated application, for example, project tracker, all the project tracker pages are also non-editable while the project is locked. The user who has access permissions for the integrated application can only view the pages.

Control Access to Source Code

It's a good idea to make sure your source code can only be used by people who have business with it.

You can control which users can view or commit source code. You can make these distinctions at the repository level or at the path level within a repository.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Settings** page, click **Permissions**.
3. Click the role whose access to source code you want to control.

   > **TIP:** If the appropriate role does not exist, you must create it first. See Create a User Role.

4. On the **Edit Role** page, click **Source Code**.

5. Under **Permissions for Specific Repositories**, select a repository in the project and specify this role's access to the repository as a whole.

   - To block users with this role from seeing this repository at all, select **No Access**.
   - To allow users with this role to see everything in the repository but not commit to it, select **View Only**.
   - To allow users with this role to commit to any path in the repository, select **View and Commit**.

6. If you want to control access to a specific path within the repository for users with this role, select **Path-based Permissions** (PBP).

   See Who can Access Source Code? and Wildcard-based access control and path-based permissions (PBP) in TeamForge for more information.

   1. Click **Add**.
   2. Specify a path in the repository.

      > **TIP:** The path does not have to exist when you specify it. You can set a permission for a path that may be created later.

   3. Select **No Access**, **View Only**, or **View and Commit** for this path.

      ✓ If none of the available permissions (**View Only**, **View and Commit**, or **Path-based Permissions**) is selected for any repository, and none of the options under **Source Code Permissions** is selected, users with this role do not see the **Source Code** toolbar button.

      ✓ If two paths have different permissions, the permissions on the lower-level path take effect. For example, consider a role that has "No Access" set for the path /branches/version3/users, but has "View and Commit" access to `/branches/version3/users/vijay`.

      ✓ Users with this role can:

      - ✓ Check code in and out of the vijay directory.
      - ✓ Click down through all the directories in that path, including users.

✓ Users with this role cannot check files in and out of the users directory or monitor commits to users.

7. Click **Save**.

> **NOTE:** You can also restrict the information that goes out with commit notification emails. See Who can Access Source Code?

Control HTML Headers in Hand-coded Project Pages

When a HTML page that you created in Microsoft Word or Frontpage looks strange, you may be able to fix it by suppressing HTML head content.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Settings** section, select **Show custom web page** and then select **Preserve HTML head content**.
3. Click **Save**.

Pages created in Microsoft Word or Frontpage should render correctly.

Allow Anonymous Subversion Checkouts

To grant anonymous checkout access while restricting write access to a Subversion repository, set the project’s default access permissions to public, provide Source Code View permission to all users, and limit other permissions to specific user classes.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **Permissions**, then the *DEFAULT ACCESS PERMISSIONS* tab.
3. Click **Edit**.
4. Choose the **Public** project access option on the **Edit Default Access Permissions** page and click **Next**.
5. Under **Application Permissions** on the **Edit Default Access Application Permissions** page, choose **Allow all Site Users and Guests** from the drop-down for Source Code View permission.

> **NOTE:** You can give all users checkout access to all repositories or a specific repository.

6. For other application permissions, choose a user class based on your access requirement.

Users can now check out from a repository without entering a password.

Show or Hide an Application

To help users focus on the relevant parts of your project, choose which applications they can see in the Project Home menu.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Admin Menu** page, click **Tools**.

3. Select the applications that you want to be visible to users in your project, and click **Save**.

Only the application buttons you have set to **VISIBLE** appear in the Project Home menu of the project page.

- Removing buttons from this menu does not remove the application from the site. Users with the appropriate permissions can still get to the hidden applications by other means, such as clicking a link in an email. The hidden application button will not appear.

- When an application button is set to **Visible**, but a given user does not have permission to use that application, that user still cannot see the application button. For example, if a user has a role that does not permit access to any source code repositories, that user does not see the **Source Code** toolbar button, even if the button is enabled for the project.

- If the Tasks tool is hidden for a project, the option to run reports on the Task tool is also hidden.

- If you create a project template from a project that has hidden applications, any project you create from that template will have the same applications hidden.

Limit User Posts to Discussions by Email

To help reduce the risk of spam or other mischief, you may need to limit the users who can post to your project’s discussion forums by email.

To leverage the advantages of community collaboration, you should keep your forums as open as you can. However, some projects require tighter control over who can participate in discussions. TeamForge enables you to balance openness against privacy along a spectrum of choices.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. Click **Discussion Settings**.

3. Set the value of **EMAIL POSTING** to one of these choices (listed from most restrictive to least restrictive).

This option...	has this effect
Allow only forum admins	Only users with the "discussion admin" permission can post by email.
Users with roles & permissions	Only registered users with the "topic post" permission can post by email. This is the default. (For discussions marked as private, this is the least restrictive setting possible.)
All logged in users	All users who are registered on the site can post by email.
Allow known email addresses only	Users can post by email only if they have explicitly been added to the monitoring list. (This can include people who are not registered site users.)
All site users and guests	Anyone can post by email.

If you select a value that's more liberal than the value your site administrator has set for the site as a whole, the site administrator's setting rules. For example, suppose the site administrator has provided that only users with the appropriate role ("Users with roles & permissions") can post by email. If your project requires extra security, you can choose to accept email only from forum administrators. However, you cannot accept email posts from a less restrictive category of users, such as "All logged in users."

Creating tags and tagging items such as documents, artifacts and so on can aid in classification, marking ownership of work items, marking items as milestones, releases and requirements, and so on. Project Administrators can set up tags to be used by project members in a project. However, project members with CREATE/EDIT permissions can create tags, if required.

Tags, once set up, can be used for tagging items such as Documents, Tracker Artifacts, and so on. To start with, TeamForge 17.1 supports tagging for Documents. Tagging will be extended for other objects in due course.

> **NOTE:** While project members with CREATE/EDIT permissions can create new tags in a project, only project administrators can delete tags.

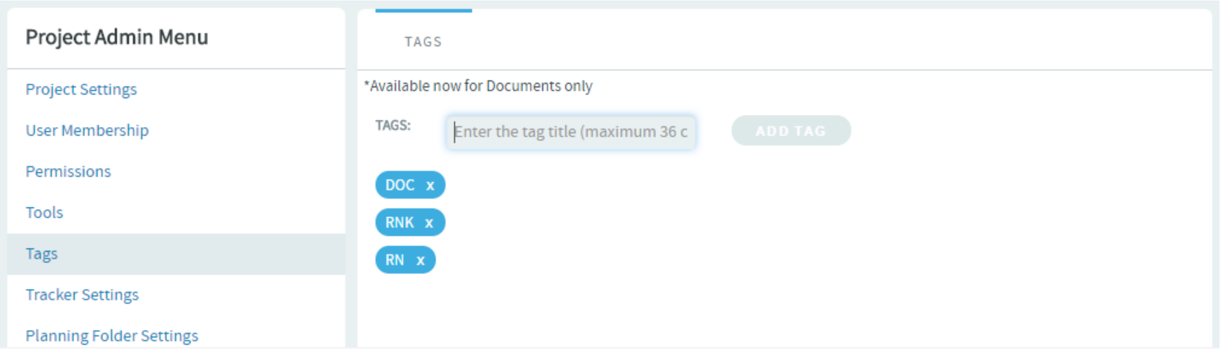
To set up tags,

1. Select a project from **My Workspace** menu.

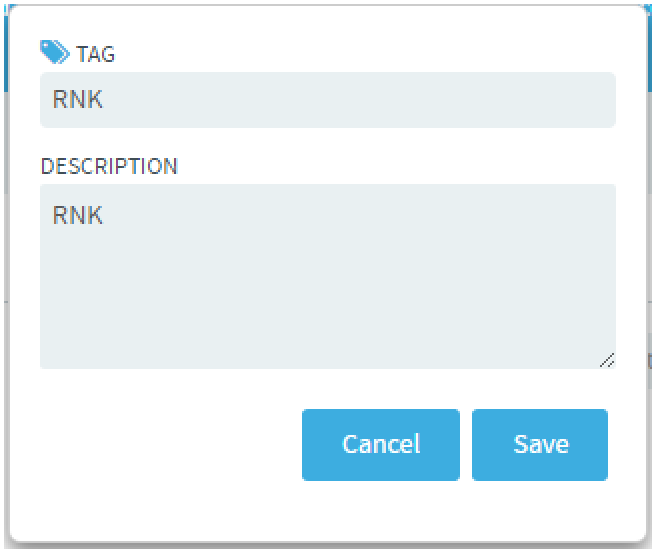
2. Select **Project HOme > Tags**.

3. Type a tag title and click **Add Tag**.

> **NOTE:** A tag can be of 36 characters in length and can contain both alphanumeric and special characters.



4. Click a tag to edit the tag's title and description, edit the tag and click **Save**.



5. To delete a tag, click the "X" mark of the tag. A confirmation message appears. Click **OK** to delete the tag.

> **WARNING:** When you delete a tag, all associations between the specific tag and TeamForge objects are removed. Exercise caution before deleting tags.

Regularly test and measure the features your team produces, involving real users as much as possible.

Accurately representing the needs and responses of real life users is a key part of the product owner’s role.

1. Schedule acceptance meetings throughout the iteration, either at regular intervals or whenever a given user story is completed. Use the latest working build of the product, and exercise the functionality live.

2. Test the functionality implemented matches what is specified in the user story. Also review the user interface and user document associated with the user store if any.

3. Enter any issues in the artifact’s **Comments** field.

4. Update the user store artifact to reflect the outcome.

    - If a user story is accepted, change the artifact status to “Accepted” or the equivalent.

    - If a user story is not accepted, change the artifact status back to “In Development” or the equivalent, for further work. You can review it again in another acceptance meeting.

> **NOTE:** Sometimes a feature turns out not to meet the user’s needs even though it has been implemented as specified. This is normal. It is not uncommon for the user’s needs to change after the user story has been captured, or for the user research process to miss one or more important details. Just note the discrepancy in the user story and route the artifact back into the appropriate part of the process.

To get your software into users' hands, upload your build to the Project Build Library.

Get the PBL Upload Client

You must download the `pbl.py` script to transfer files into the **Project Build Library**.

The PBL upload client is free, open-source, and freely modifiable and distributable.

Download the PBL upload client from http://cubit.open.collab.net/pbl/.

> **NOTE:** The API operations are fully documented, for users who might want to develop their own PBL upload client.

Upload a File

To upload a file, run the `pbl.py` upload command.

In this example, we upload a file named `Release.zip` from our local machine into the public area of the project **myproject**, in the directory `/foo/bar/baz/`.

NOTE: If any part of the requested path does not already exist, `pbl.py` creates the intermediate directories.

Run the `pbl.py` upload command like this, substituting the correct values for your situation.

```
pbl.py upload --api-user=username --api-key=713cdf90-2549-1350-80c3-2d0bcf9a16
97 --api-url http://$external_host/cubit_api/1 --project=myproject -t pub -r /
foo/bar/baz -d "This is the description." /home/Release.zip
```

TIP: Wildcards are accepted in the filename argument. If the file argument is a directory, or a wildcard which includes one or more directories, `pbl.py` recursively uploads all the subdirectories underneath the parent. All the files in the recursive upload get the same description.

Once this operation has completed, you can download this file from https://$external_host/myproject/pub/foo/bar/baz/Release.zip.

For other options of `pbl.py` scrip, run

```
pbl.py help upload
```

Change the Description of a File

You can change the description associated with a file or directory without changing the file itself or the md5 checksum of the file.

In this example, you have already authenticated and saved your user name and key credentials in your home directory.

Run the `pbl.py changedesc` command like this:

```
pbl.py changedesc -l http://$external_host/cubit_api/1 --project=myproject -t
pub -r /foo/bar/baz/Release.zip -d "This is the new description"
```

Move a File

With the `pbl.py move` command, you can move files or directories within a project, or even between projects.

The syntax for this command is a bit different than the rest of the commands, because the other commands only operate on one project or file or directory at a time, and that is not the case the the move operation.

To move a file, run the `pbl.py move` command with these options. In the simplest case, we move a file, or a directory and all its contents, from one name to another.

Commands	Description
--srcproj projname	The name of the project the source file is located in.
--destproj projname	The name of the project to move the file to. If left blank, defaults to value of --srcproj.
--srcpath path	The path to the file or directory to move.
destpath path	The destination path for the file or directory specified in --srcpath. Two important things to note about this option: • If you specify a path which does not exist, that path will be automatically created for you as part of the move. • If the --destpath parameter ends with a slash ("/"), the destination will be assumed to be a directory. If it does not end with a slash, the destination will be assumed to be a file. An example of this behavior is below. This is approximately how the UNIX "mv" command behaves.
--srctype {pub|priv}	The visibility type of the source file, either "pub" or "priv".
--desttype {pub|priv}	The visibility type of the destination file, either "pub" or "priv".
--force	If the destination file exists, the --force option must be used to replace it.

NOTE: Because destpath does not end with a slash – `/foo/bar/baz/Release_old.zip` – the last component of the path is interpreted as a `file named 'Release_old.zip'.`

```
pbl.py move -l http://$external_host/TeamForge Lab Management_api/1 --srcprj=m
yproject --srctype=pub --srcpath=/foo/bar/baz/Release.zip --destpath=/foo/bar/
baz/Release_old.zip
```

To move a file from one project to another, and also change it from public to private, run the command like this.

NOTE: Because destpath ends with a slash – `/foo/bar/baz/archive/` – the last component of the path is interpreted as a `directory named 'archive'.`

```
pbl.py move -l http://$external_host/TeamForge Lab Management_api/1 --srcprj=m
yproject --destproj=myproject_archive --srctype=pub --desttype=priv --srcpath=
/foo/bar/baz/Release.zip --destpath=/foo/bar/baz/archive/
```

To enable project members to allocate hosts from a public cloud in your Lab Management site, you must turn on a setting to allow the inclusion of public clouds in your project.

1. In **Administration > Projects**, click **Edit** for your project.

2. Turn on the **Allow Public Clouds** setting. If you want to be able to control the hosts your project members can select from, turn this setting OFF. In this case, your project members can only select systems from a cloud your project explicitly owns. Your project’s **Allowed Clouds** page lists all the available clouds from which project members can allocate hosts.

> **NOTE:** If you think that your project needs more systems than those available from a cloud, you need to ask the cloud administrator for your site to increase this number.

The Lab Management Cloud plugin enables Hudson and Jenkins to automatically create slaves from Lab Management clouds.

Install the Lab Management Cloud Plugin for Hudson and Jenkins

Get the plugin from **openCollabNet** and upload it using the **Plugin Manager**.

1. Download the `labmanagement.hpi` file from **openCollabNet**. (Please check back later for availability.)

2. In the Hudson or Jenkins Plugin Manager page, click the *Advanced* tab.

3. In the Upload Plugin section, browse to the location where you saved the `.hpi` file and click **Upload**.

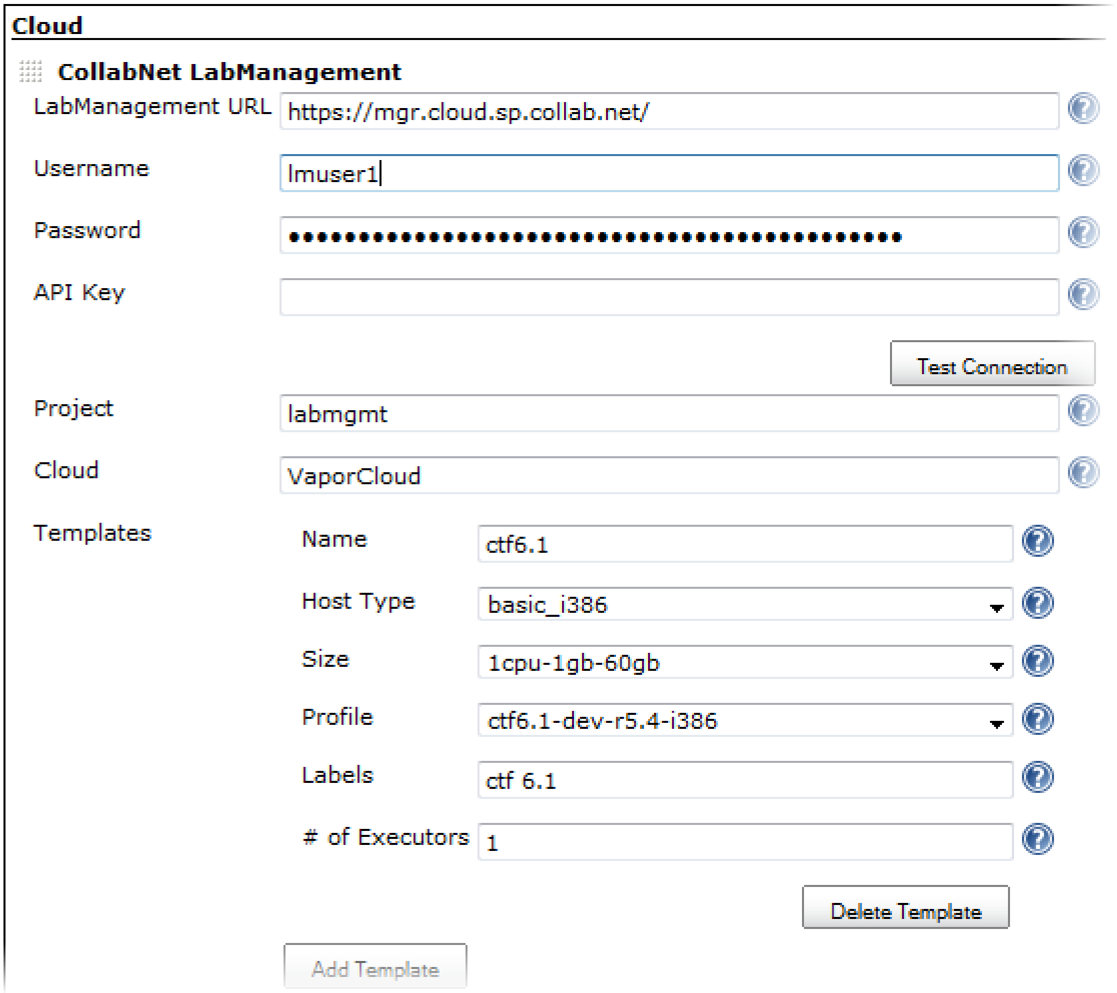
In the Plugin Manager’s *Installed* tab, you should see the Lab Management Cloud plugin enabled.

Configure the Lab Management Cloud Plugin for Hudson and Jenkins

When you add a new cloud, you set up templates specifying details such as the host type and profile for the nodes that will be provisioned from this cloud.

1. In the Cloud section of the Hudson or Jenkins configuration page, click **Add a new cloud** and select **CollabNet Lab Management**.

   Lab Management options are displayed. Here’s an example:



Cloud

CollabNet LabManagement

LabManagement URL: https://mgr.cloud.sp.collab.net/

Username: lmuser1

Password: ••

API Key:

Project: labmgmt

Cloud: VaporCloud

Templates

Name: ctf6.1

Host Type: basic_i386

Size: 1cpu-1gb-60gb

Profile: ctf6.1-dev-r5.4-i386

Labels: ctf 6.1

# of Executors: 1

2. Enter the URL of the Lab Management Manager node. For example, https://mgr.cloud.sp.collab.net/.

3. Provide the user name and password. This user must have access to the Lab Management project where new hosts will be allocated.

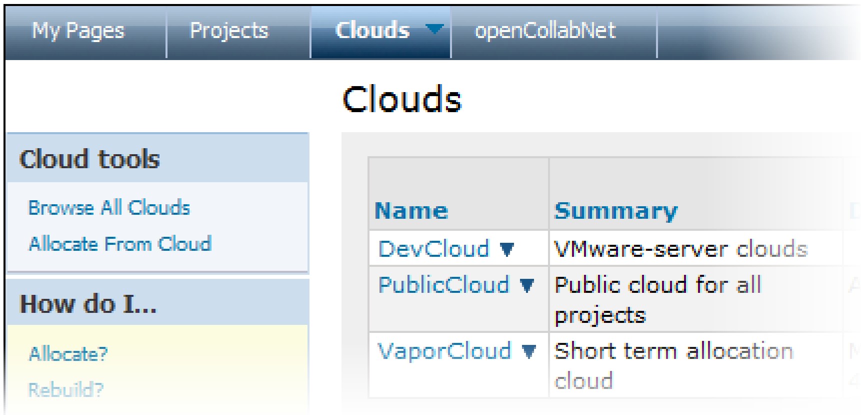
4. Enter the user’s API key for the Lab Management web service.

> **TIP:** Copy and paste the key from the user’s Lab Management home page.

5. Click **Test Connection** to make sure that the options you provided are valid.

6. Specify the Lab Management project where new hosts will be added.

7. Select a Cloud from which new hosts will be allocated. You can find the list of valid cloud names in the Lab Management Manager interface. For example:



Clouds

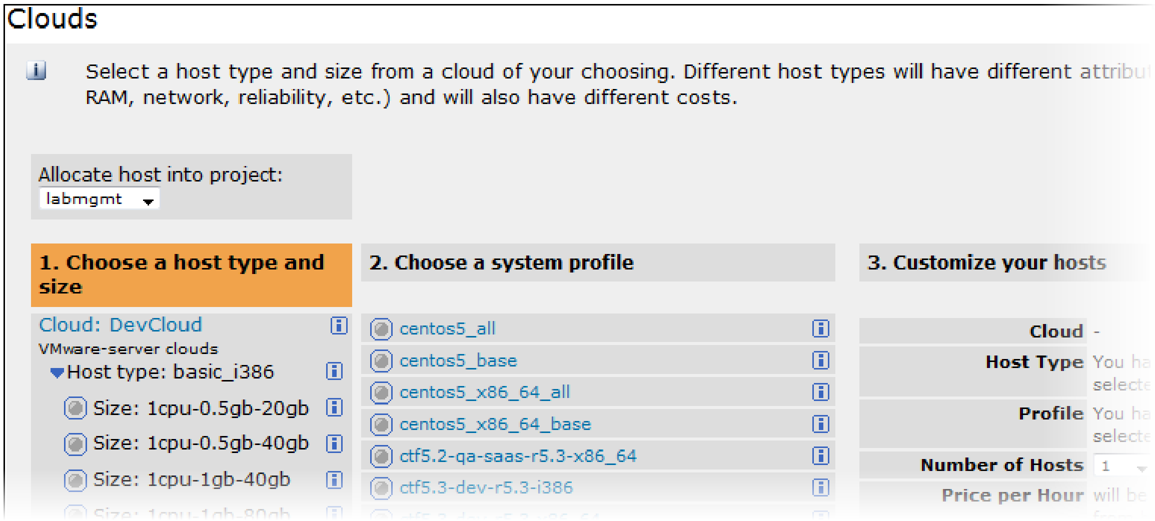
Name	Summary
DevCloud	VMware-server clouds
PublicCloud	Public cloud for all projects
VaporCloud	Short term allocation cloud

8. Click **Add Template** to create a template that Hudson will start.

   1. Specify a name that characterizes the template. The **name** is used to identify the template and is displayed in various parts of the Hudson interface.

   2. For **Host Type**, select the type of hardware you want for the node.

   3. For **Size**, select the amount of resources – memory, CPU, and disk – assigned to this template. You can find the exact amount of resources for each size in the Lab Management Manager's **Clouds > Allocate From Cloud** page. For example:



Clouds

Select a host type and size from a cloud of your choosing. Different host types will have different attribu RAM, network, reliability, etc.) and will also have different costs.

Allocate host into project: labmgmt

Cloud: DevCloud
VMware-server clouds
Host type: basic_i386
Size: 1cpu-0.5gb-20gb
Size: 1cpu-0.5gb-40gb
Size: 1cpu-1gb-40gb
Size: 1cpu-1gb-80gb

centos5_all
centos5_base
centos5_x86_64_all
centos5_x86_64_base
ctf5.2-qa-saas-r5.3-x86_64
ctf5.3-dev-r5.3-i386

Cloud -
Host Type You ha selecte
Profile You ha selecte
Number of Hosts 1
Price per Hour will be

   4. For **Profile**, select the Lab Management profile to be used for the template.

5. Provide a list of **labels** separated by white spaces.

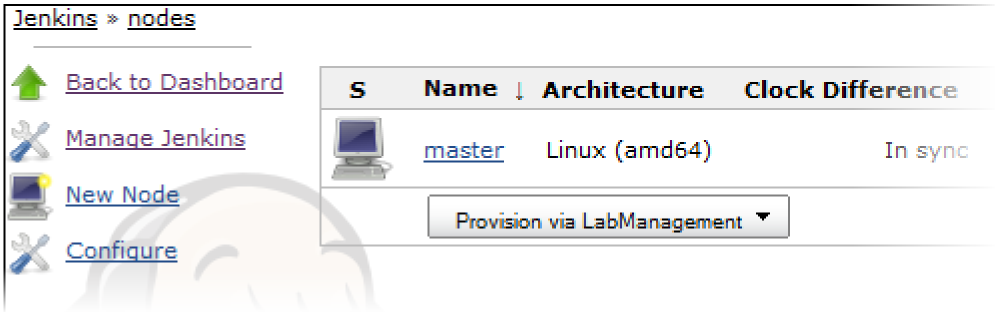
> **NOTE:** The names and labels you provide for the templates are used to specify which jobs will use this cloud.

6. Enter a value for **# of Executors**. This controls the number of concurrent builds that Hudson or Jenkins can perform. So the value affects the overall system load that might be incurred. A good value to start with is the number of processors on your system.

   When using Hudson or Jenkins in the master/slave mode, setting this value to 0 would prevent the master from doing any building on its own. Slaves may not have zero executors, but may be temporarily disabled using the button on the slave's status page.

9. Click **Save**.

In the **Manage Nodes** page, you'll see an option to provision a node from Lab Management.



S	Name ↓	Architecture	Clock Difference
	master	Linux (amd64)	In sync

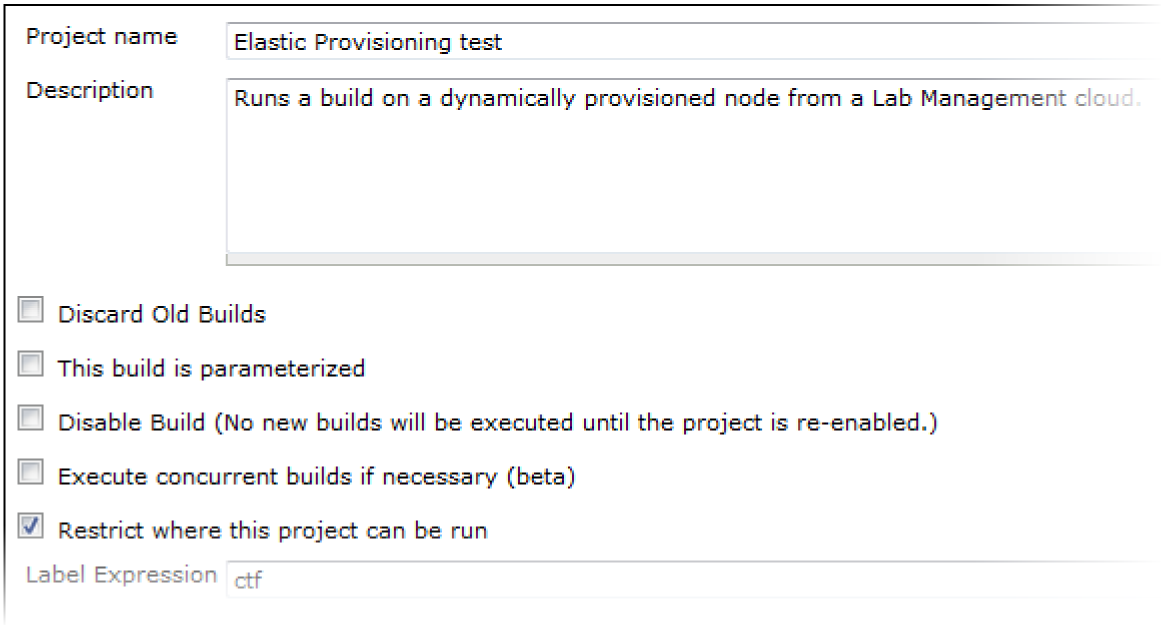
Run a Hudson or Jenkins Job on a Host Provisioned from Lab Management Cloud

When you run a job on a host provisioned from a Lab Management cloud, use a template you defined earlier.

1. In the Hudson or Jenkins configuration page for the job, provide a name and description.

2. Select the **Restrict where this project can be run** option to tie the job to the Lab Management cloud.

3. Enter a label expression. To always run this job on a specific node, just specify its name. However, when several nodes could be available and you don't want to tie the job to a specific one, enter an expression based on the name and label values of a Lab Management cloud template you configured earlier.



Project name	Elastic Provisioning test
Description	Runs a build on a dynamically provisioned node from a Lab Management cloud.

□ Discard Old Builds

□ This build is parameterized

□ Disable Build (No new builds will be executed until the project is re-enabled.)

□ Execute concurrent builds if necessary (beta)

☑ Restrict where this project can be run

Label Expression	ctf

4. Specify any other options you want and click **Save**.

When a node is no longer used, it will be brought down and released.

To collaborate with others on a CollabNet TeamForge site, start by getting a user account.

The process of joining a TeamForge site varies according to your site's setup. Here's what you should consider before you join a TeamForge site:

- Is user self-registration allowed on your site? User self-registration is, by default, disabled in TeamForge. However, TeamForge site administrators may choose to enable user self-registration. Such self-registered user accounts are later approved by the site administrator. For more information, see DISABLE_USER_SELF_CREATION and APPROVE_NEW_USER_ACCOUNTS.
- How is your TeamForge site authenticating users? TeamForge supports user authentication both against its internal database and against other external authentication services such as LDAP, OAuth, and SAML. The account creation procedure varies according to the authentication setup.

  Refer to the relevant instructions in this topic that suit your site's setup.

> **NOTE:** You need a license to use TeamForge. Your site administrator may have already assigned you a license. If you are self-registering, you'll be asked to choose the type of license you need when you create your account. For more information about TeamForge license, see TeamForge License.

Sites that Use TeamForge's Internal Database Authentication

User self-registration is, by default, disabled in TeamForge. In such cases, user accounts can only be created by TeamForge administrators and users would get an email notification with a link to log on to TeamForge.

Follow these steps if your site uses TeamForge's internal database for authentication and if user self-registration is enabled.

1. Click **Create an Account** in the **New Users** section of the TeamForge home page.
2. On the **Create New Account** page, enter a username for your account.

   Your user name must meet these criteria:

   - User name is case sensitive.
   - Minimum number of characters as specified in the site-options.conf file.
   - No spaces.
   - Should have at least one letter.
   - The first character is a letter.
3. Enter and confirm a password.

4. Fill in the rest of the fields and click **Create**.

   Your user account is now created, pending approval by a TeamForge site administrator. Once approved, you will get an email notification with a link to log on to TeamForge.

5. Follow the link in the email to log on to TeamForge.

Sites that Use LDAP Authentication

Follow these steps if your TeamForge site uses LDAP authentication.

1. In the **Log Into TeamForge** section of the TeamForge home page, enter your corporate LDAP username and password.

   > **TIP:** In most cases, your username and password are the username and password with which you log in to your corporate network.

2. Click **Log In**.
3. On the **Create Account** page, re-enter your LDAP password.
4. Enter your full name and email address and click **Create**.

   > **NOTE:** The TeamForge Administrator receives an email notification to approve the new account that you've created. Once approved, you will get an email notification with a link to log on to TeamForge.

5. Follow the link in the email to log on to TeamForge.

Sites that Use SAML/SAML+LDAP Authentication

Follow these steps if your TeamForge site uses SAML or SAML+LDAP authentication.

1. In the **Log In to TeamForge** section of the TeamForge home page, click **Log In**.
2. On the third-party Identity Provider's (IdP) login page, enter the username and password provided by your third-party IdP.
3. Click **Sign In**.
4. On the **Create New Account** page, enter the password and other user details.

> **NOTE:** Your user name and email address fields cannot be edited on the **Create New Account** page.

5. Click **Create**.

> **NOTE:** The TeamForge Administrator receives an email notification to approve the new account that you've created. Once approved, you will get an email notification with a link to log on to TeamForge.

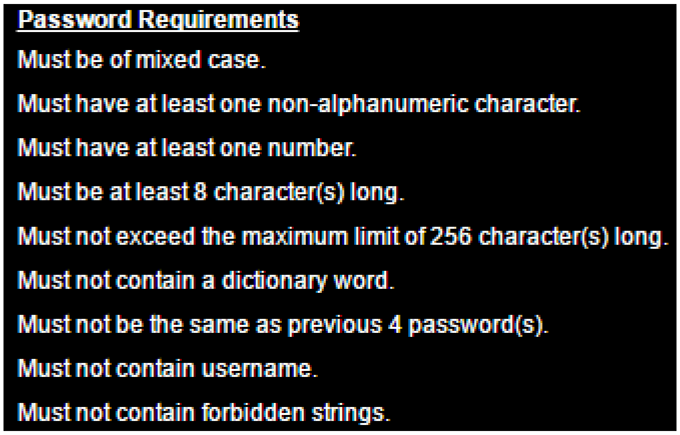
6. Follow the link in the email to log on to TeamForge.

To protect the security of your TeamForge account, change the password regularly.

1. Select **My Settings** from the **My Page** menu.
2. In the **User Details** section, click **Change Password**.
3. In the **My Workspace / Change Password** page, enter your current password.

4. Enter your new password and confirm it.

   The password requirements depend on the security policies enforced on your site. Here's an example:



Password Requirements

Must be of mixed case.

Must have at least one non-alphanumeric character.

Must have at least one number.

Must be at least 8 character(s) long.

Must not exceed the maximum limit of 256 character(s) long.

Must not contain a dictionary word.

Must not be the same as previous 4 password(s).

Must not contain username.

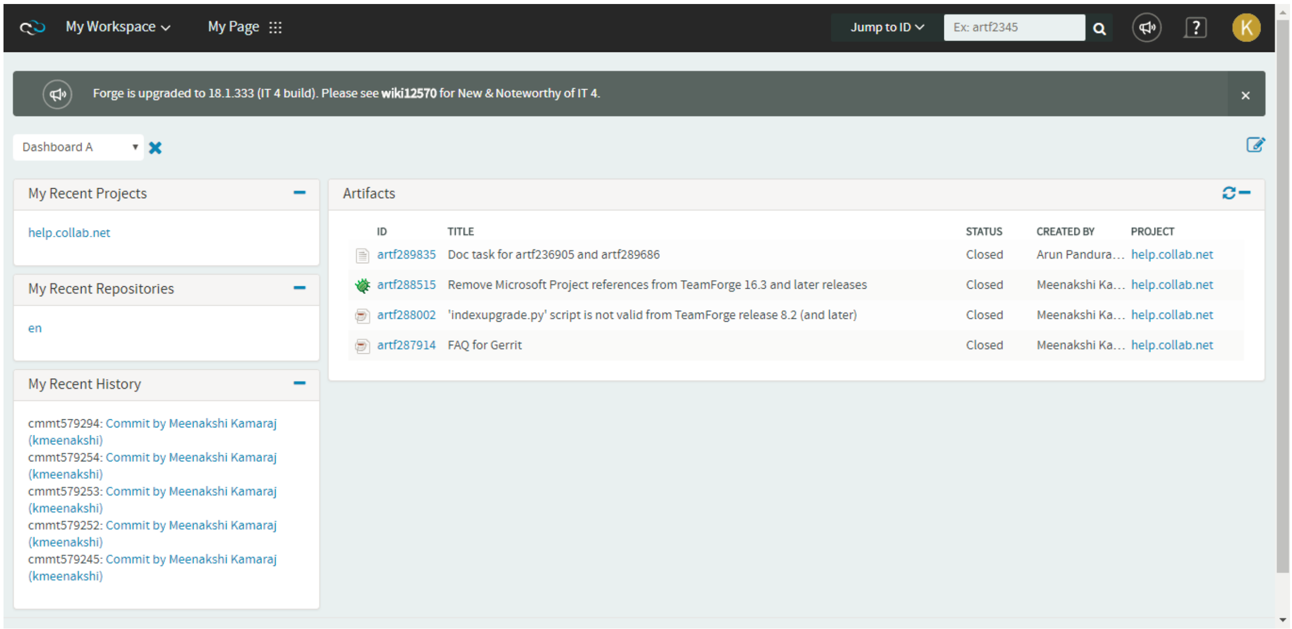
Must not contain forbidden strings.

5. Click **Update**.

   You will see a message that your password was changed successfully.

Your My Workspace is a personal workspace that offers a handful of configurable widgets such as My Recent Projects, My Recent Repositories, Git Code Reviews, Project News and so on. You can use these widgets to view recent projects, recent repositories, recent commits, items assigned to you (TeamForge artifacts and document reviews), Git code reviews, project news, reports and more.
After logging into TeamForge, you are taken to your **My Workspace**.

You can create one or more dashboards to categorize and view data in many different views. For instance, you may decide to create three dashboards: Dashboard A, B and C. With these three dashboards, you may choose to configure Dashboard A and B to view events from projects A and B respectively and configure Dashboard C to view just the code reviews.



Forge is upgraded to 18.1.333 (IT 4 build). Please see **wiki12570** for New & Noteworthy of IT 4.

Dashboard A

My Recent Projects

help.collab.net

My Recent Repositories

en

My Recent History

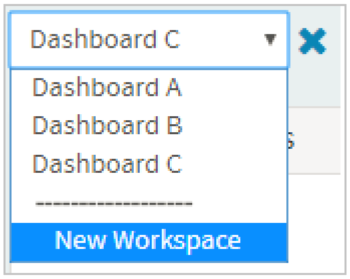
cmmt579294: Commit by Meenakshi Kamaraj (kmeenakshi)
cmmt579254: Commit by Meenakshi Kamaraj (kmeenakshi)
cmmt579253: Commit by Meenakshi Kamaraj (kmeenakshi)
cmmt579252: Commit by Meenakshi Kamaraj (kmeenakshi)
cmmt579245: Commit by Meenakshi Kamaraj (kmeenakshi)

Artifacts

ID	TITLE	STATUS	CREATED BY	PROJECT
artf289835	Doc task for artf236905 and artf289686	Closed	Arun Pandura…	help.collab.net
artf288515	Remove Microsoft Project references from TeamForge 16.3 and later releases	Closed	Meenakshi Ka…	help.collab.net
artf288002	'indexupgrade.py' script is not valid from TeamForge release 8.2 (and later)	Closed	Meenakshi Ka…	help.collab.net
artf287914	FAQ for Gerrit	Closed	Meenakshi Ka…	help.collab.net

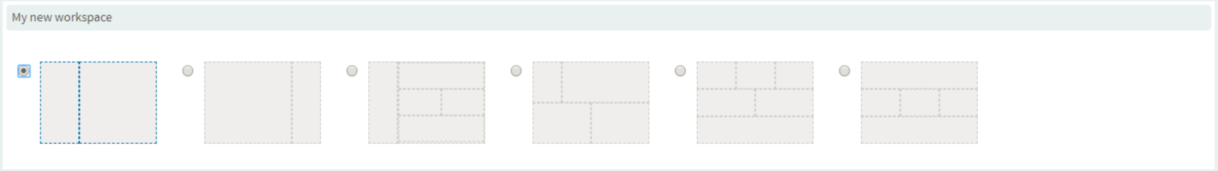
Create and Configure Dashboards: Step by Step

1. To create a new dashboard, click **New Workspace** from the workspace drop-down on your dashboard.



NOTE: you can create as many dashboards as required.

2. Enter a name for the dashboard that you're going choose from the dashboard layouts provided.



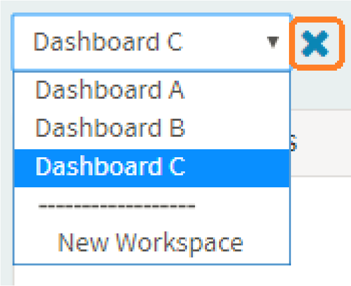
My new workspace

3. Click **Add widget** and select a widget.

> **NOTE:** You can add multiple instances of the same widget to your dashboard. For example, you can add two instances of the **Artifacts** widget, one each to show the **Open** and **Closed** artifacts assigned to you.

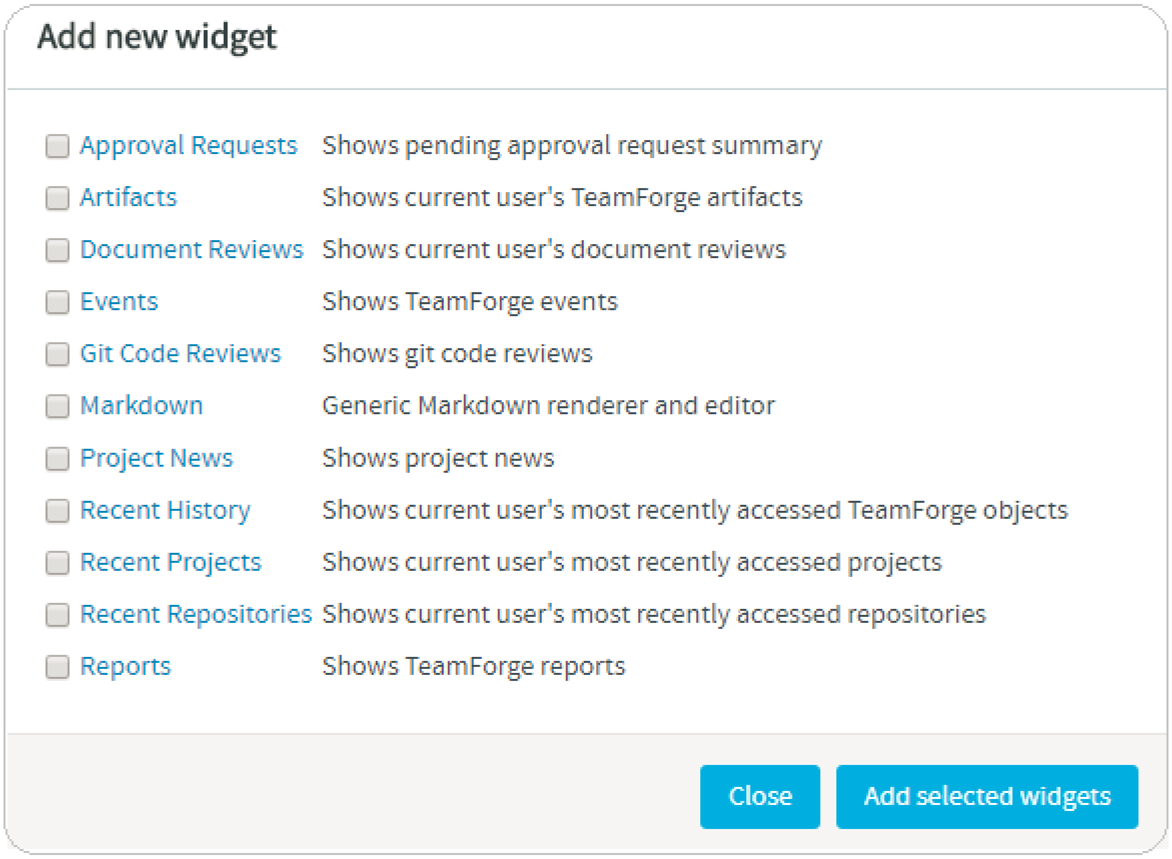
4. All widgets have criteria or content that controls what information they display. These settings can be configured by clicking the **Edit widget configuration** icon ⚙ on the widget after adding them to the My Workspace.
5. Arrange widgets by clicking the **Change widget location** icon ✥ and dragging the widget to the place in your layout where you want to see it.
6. Use the Collapse widget — and Remove widget ✖ icons to collapse and remove widgets.
7. Repeat steps 4 through 7 to add and configure more widgets.
8. When you are done, click **Save**.

You can select a dashboard from the drop-down list and click **X** next to the dashboard drop-down list to delete the dashboard.



My Workspace Widgets

Widgets, which provide the functionality you want to expose on your dashboard.



Add new widget

□ Approval Requests	Shows pending approval request summary
□ Artifacts	Shows current user's TeamForge artifacts
□ Document Reviews	Shows current user's document reviews
□ Events	Shows TeamForge events
□ Git Code Reviews	Shows git code reviews
□ Markdown	Generic Markdown renderer and editor
□ Project News	Shows project news
□ Recent History	Shows current user's most recently accessed TeamForge objects
□ Recent Projects	Shows current user's most recently accessed projects
□ Recent Repositories	Shows current user's most recently accessed repositories
□ Reports	Shows TeamForge reports

A variety of widgets have been provided, which include the following:

- Approval Requests: Shows a summary of pending approval requests
- Artifacts: Shows artifacts assigned to the user
- Document Reviews: Shows document reviews assigned to the user
- Events: Shows TeamForge events
- Git Code Reviews: Shows Git Code Reviews
- Markdown: Generic Markdown renderer and editor
- Project News: Shows project news
- Recent History: Shows objects most recently accessed by the user
- Recent Projects: Shows projects most recently accessed by the user
- Recent Repositories: Shows repositories most recently accessed by the user
- Reports: Shows TeamForge reports

You can select the desired widgets and click the **Add selected widgets** to add them to your **My Workspace** dashboard.

The following filters of the **Artifacts** widget are now multi-select filters that let you further narrow your filter scope: `Project`, `Assigned To`, `Created By`, `Status` and `Priority`. For example, you can now configure the **Artifacts** widget to filter and show artifacts from a select list of multiple projects, multiple users, multiple priorities and so on.

To help get started on collaborating with other project members, you can provide some information about yourself.

This information appears whenever someone clicks your name anywhere on the site. For example, when an artifact is assigned to you, your name appears as a link that other users can click.

1. Click **My Settings** from the **My Page** menu.
2. In the **User Details** section, click **Edit** and provide some information to help potential coworkers get to know you.
3. Click **Choose File** to upload a 100-by-100-pixel picture of yourself, or of something that suggests who you are.
4. Under **Detail**, provide a summary of your interests, skills, or other characteristics.
5. Select the language you prefer to use. You can choose to receive emails from your TeamForge site in English, Chinese, Japanese, or Korean.

   > **NOTE:** If your browser is set to use one of the supported languages, you should already be seeing TeamForge in that language in your web browser.

6. If necessary, review and change your password, official name, and email address.

   This is the email address where you receive alerts about changes to items you are involved with, such as a discussion forum you are monitoring or a code commit associated with an artifact you created.

7. Click **Update**.

User Preferences

1. Select the **User Preferences** tab.
2. Select your notification pereference for monitored items. Select one of the following options from the ** NOTIFICATIONS ON MONITORED ITEMS** drop-down list:
   - Email per Change
   - Daily Digest Email
   - Don’t Send Email
3. Select the **INCLUDE MY OWN UPDATES IN MY NOTIFICATIONS** check box to have notification emails sent for your own updates.
4. Select the **NOTIFY ASSOCIATION AND DEPENDENCY UPDATES** check box to have notification emails sent when associations and dependencies are updated.
5. Select an encoding format from the **FILE ENCODING FOR EXPORT** drop-down list.

6. Click **Save**.

Authorization Keys

Select the **Authorization Keys** tab, enter any authorization keys you may have (ssh authorization keys for example) and click **Update**.

Roles

Select the **Roles** tab, select one of the options from the **View** drop-down list. You can select `Roles Created For a Project`, `Roles Inherited From a Parent Project` or `Site-wide roles` to view the corresponding roles assigned to you. See [Which role is assigned to me?][faqs.html#which-role-is-assigned-to-me]

User Group Membership

Select the **User Group Membership** tab and view the list of user groups you are a member of.

To create, store, and share work on a CollabNet TeamForge site, first join a project.

1. Log on to TeamForge.
2. From your **My Page** menu, click **Projects**.

   Now you're looking at the projects of which you are already a member, if any.
3. Click the **All Projects** tab to see all TeamForge projects, based on each project's access settings.
4. Select the project you want to join and click **Request Membership**.
5. On the **Join Project** page, explain why you want to join the project in the **REQUEST COMMENT** text box, and click **Submit**.

   Your request is submitted to the project administrator for approval. You will receive an email notification when your request is either approved or denied.

   > **TIP:** You can also request project membership by clicking **Join this Project** on the home page of the project you want to join.

When you want to search for TeamForge objects such as tracker artifacts or documents, you can quickly search using a unique identifier or a keyword or you can perform an advanced search.

TIP: While searching with an ID, you can select an object category from the drop-down list and search only within the selected object category. You can also select **Advanced Search** from the drop-down list to perform an advanced search.

Full-text Search in TeamForge

Here is some detailed information to help you make the most of TeamForge's full-text search capabilities powered by the Apache Lucene full-text search engine.

An Overview of Full-text Search

Here is some detailed information about the TeamForge's full-text search.

Searchable Items in TeamForge

The following items are searchable in TeamForge.

- Discussion forums, posts and topics
- Documents and document folders
- File releases, packages, and FRS files
- News posts
- Visible project pages
- Projects
- Source code files and commits
- Tasks and task folders
- Trackers and tracker artifacts
- Users
- Wiki pages
- Integrated applications

Integrated applications

Integrated applications may or may not have search capabilities. Refer to the integrated application's documentation to know more about its search features and to appreciate how the search features of the integrated application and TeamForge differ.

Attached files

You can search the contents of attached files. For more information about supported document formats, click here.

Notes

You may witness some delay for the TeamForge objects to appear in the search results and the extent of delay depends on the load on the server.

When you search, the contents of all the search-able fields (of an object) are collectively searched for matches. For instance, when you search for an artifact, the contents of the title, description and comment fields, put together, are searched. As an example, the search entry CollabNet AND TeamForge returns an artifact if the content of its title, description and comment, put together, has both the words “CollabNet” and “TeamForge”. In other words, the word “CollabNet” could be in the artifact title and the word “TeamForge” could be in the artifact description (and not necessarily present on the same field).

If you search with multiple words, items containing any of the words in the search string are returned. For more information, see Multiple Terms Search. On the other hand, if you want to find items where the words, say CollabNet and TeamForge, both appear, type CollabNet AND TeamForge. For more information, see Boolean Operators.

The TeamForge searches for full words. Use Wildcard Searches for partial word searches.

Search terms are case-insensitive. For example, if you search using the keyword `collabnet`, pages that contain COLLABNET, CollabNet and collabnet are all returned.

TeamForge Full-text Search Guidelines

Here is some guidelines to help you create effective searches.

Single Term Search

Single-term search looks for all search results that match the search text. For example, a search entry of `doc` only returns search results of “doc”.

Multiple Terms Search

Multiple-terms search looks for all search results that match any of the words in the search text. For example, a search entry of `document plan` returns search results of “document”, “plan”, and “document plan”.

Search by Phrase

A group of words surrounded by double quotes, such as `"product requirements"`, return only search results containing the entire phrase.

Boolean Operators

Terms and phrases can be combined with Boolean operators for more complex searches. Boolean operators must be in upper case. Use:

- `OR` between two terms returns search results containing either of the terms. This is the default operator used if no other operator is specified.
- `AND` between two terms returns only search results containing both of the terms.
- The `+` operator before a term makes the term required. Only search results containing the terms are returned.
- The `-` or `NOT` operator before a term returns only search results that do not contain the term. The character “-“ represents the Boolean operator AND NOT.

TIP: You can group Boolean searches using parentheses. For example, (`doc` OR `test`) AND `plan` returns search results containing “doc plan” and “test plan”.

Wildcard Searches

To look for search results with a single character replaced, use the ? symbol. For example, to look for search results with “text” or “test”, enter `te?t`.

To look for search results with more than one character replaced, use the * symbol. For example, to look for search results such as “content” or “contest” or “continuous” or “control”, enter `cont*`.

NOTE: You can use wildcard symbols in the middle or at the end of a search, but not as the first character of a search keyword.

Fuzzy Searches

To look for search results with spelling similar to the search term entered, use the ~ symbol as the last character of the search keyword. For example, to look for search results with spelling similar to “roam”, enter `roam~`. This returns search results such as “roam” and “roams”.

Special Characters

If you have any of the following special characters in your search text, you must escape them by enclosing the entire phrase in double quotes. `+ - & || ! ( ) { } [ ] ^ " ~ * ? : \`

For example, to look for search results containing the hyphenated term “product-development”, enter `product-development`.

The special character “+” represents the Boolean operator AND. The special character “-“ represents the Boolean operator AND NOT.

Regular Expression Search with Forward Slashes

Lucene 4 supports regular expression searches matching a pattern between forward slashes “/”. For example, to look for search results containing the words “moat” or “boat”, use the search string `/[mb]oat/`.

If you are specifically looking for search results containing a forward slash “/” character, you must backslash-escape or quote-escape the forward slash character. For example, to look for search results containing `<opt/collabnet>`, use the search string `<opt\/collabnet>`.

Excluded Words

The following words are considered stop words and are not search-able on their own: `a, an, and, are, as, at, be, but, by, for, if, in, into, is, it, no, not, of, on, or, s, such, that, the, their, then, there, these, they, this, to, was, will, with`.

Range Searches

You can do a range-bound search using the TO operator. For example, the search entry, `[artf1100 TO artf1200]`, returns items containing values between artf1100 and artf1200, including artf1100 and artf1200. To exclude the upper and lower bounds from the search results, use curly brackets {} instead of square brackets [].

Jump to ID Search

1. Log on to TeamForge. If you are not logged on, you can search only projects and items that have been designated public.
2. If you know the unique identifier of an object and want to quickly go to the object, type the unique identifier in the **Jump to ID** text box and click the search icon.
   The default quick search option is **Jump to ID**.
3. If you want to do a keyword search of a specific object type (such as documents or discussions), type the keyword in the text box, select an object type from the drop-down list and click the search icon. The following table lists the search-able object types you can select from the drop-down list.

Searchable object types	Description
Discussions	Select this option to search in discussion forums.

Searchable object types	Description
Documents	Select this option to search for documents.
File Releases	Select this option to search in file releases.
News	Select this option to search project news.
Project Pages (Visible)	Select this option to search project pages.
Projects	Select this option to search projects.
Source Code	Select this option to search the source code. For more information, see How to search for Source Code?.
Tasks	Select this option to search for tasks.
Trackers	Select this option to search for tracker artifacts.
Users	Select this option to search for users.
Wiki	Select this option to search in Wiki pages.

Advanced Search

The Advanced Search function lets you search globally on all the projects or on specific projects of interest. You can also scope your search to one or more components such as Documents, Discussions and so on using the Advanced Search.

1. Click **Advanced Search** from the **Jump to ID** menu (drop-down list).
2. On the **Search Criteria** page, enter the keywords to search for.
3. Select one or more components such as **Discussions**, **Documents** and so on from the **IN** list.
4. Select one or more projects listed in the **IN PROJECTS** list. You can also select **All Projects**.
5. Select the **Search Attachments** check box and the **Search Comments** check box if you want to search attachments and comments respectively.
   Attachments refer to tracker artifact attachments. Comments refer to tracker artifact comments and task comments.
6. Select one of the two options, **Search Active Versions Only** or **Search All Versions**, to specify whether you want to search active document versions only or all document versions respectively. Searching only active document versions allows you to eliminate search results for outdated documents.
7. Click **Search**.

Your search results are organized by TeamForge application. The search score indicates the relevance of each result to your search criteria. You can see only those items that your project membership and permissions allow you to see.

A Tracker is a collection of related artifacts that describe work to be done or issues to be resolved. Every project should have one or more trackers. When you start a tracker, you decide which fields will be used, who will use them, and how they will use them.

What is a Tracker?

A tracker is a collection of records that follow the development of a unit of work from conception through to completion. You can create a tracker to manage almost any kind of work that your project calls for.

In each TeamForge project, you can create any number of trackers. Each tracker tracks a different type of data.

For each tracker, you can define values for status, category, and other default fields. You can create your own fields to capture additional data that is specific to your project or organization. You can also create tracker workflows to specify the criteria necessary for users to change tracker status values.

Individual tracker entries are referred to as tracker artifacts. The role-based access control system enables you to control which project members are allowed to view, create, and edit tracker artifacts. Within a project or across the projects, a tracker can be cloned along with the workflow.

Summary information about the number and status of project tracker artifacts is provided on each project's **Tracker Summary** page. The Project Dashboard also provides an at-a-glance overview of the status of each project member's projects, including information on the number and status of the tasks and tracker artifacts in each project, and project overrun and underrun statistics.

Create a Tracker

Create a tracker whenever you need to report and track bugs, feature requests, support requests, or any other type of issue where ownership, status, and activity must be managed.

Individual tracker entries are referred to as tracker artifacts. A tracker is a set of tracker items with a common purpose, such as bug reports, feature requests, or tasks.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. Click **Create**.
4. On the **Create Tracker** page, provide a name and description for the tracker.

   > **TIP:** Descriptions help users learn how best to provide the information you want from them. To maximize your chances of getting useful data, make your description as informative as you can.

5. Select an icon that suggests the type of work the tracker is handling. This icon will appear with any artifact in this tracker, wherever it is viewed on the site. For example, if someone brings an artifact from

this tracker into a planning folder, users of the planning folder can glance at the artifact's icon to see where it comes from.

6. Select the relevant unit from the **DISPLAY EFFORT IN** field. The units displayed here are configured based on the size of the artifacts in the tracker. Eg. Select the unit as **HOURS** for a tracker of small defects, **DAYS** for a tracker of tasks, and **WEEKS** for a tracker of epics.

    > **NOTE:** Configure the units at the project level and not at the planning folder level.

7. Select **INCLUDE FOREIGN CHILDREN** to include points and efforts from children artifacts across the projects in TeamForge.

    > **NOTE:** In a parent artifact, enabling CALCULATE POINTS field sums and rolls up the points from all its children artifacts within the project. In this total, if you want to include children artifacts from other projects across TeamForge, have the INCLUDE FOREIGN CHILDREN option enabled.

8. Click **Create**. The new tracker appears at the bottom of your list of trackers.
9. If necessary, drag the tracker to a place in your tracker list that makes sense. The order you set here controls the order of every tracker list in your project.
10. You'll probably need some custom fields to capture information that's specific to your project. See Create Custom Trcaker Fields.
11. To speed up the team's work, you may want to set up some rules for automatically reassigning artifacts when their contents change. See Create a Tracker Workflow.

Create Custom Tracker Fields

To track data that is not captured by the default set of fields, create new fields that fit your project's purposes.

You can create the following user-defined fields in each tracker:

- Up to 30 text entry fields.
- Up to 30 date fields.
- Up to 30 single-select fields.
- An unlimited number of multiple-select fields.

> **CAUTION:** Creating a large number of user fields and multiple-select fields may affect site's performance.

Create a Text Field

To let users type in data, create a text entry field.

A tracker can have up to 30 text fields.

1. Click **Project Admin** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. Click the tracker to which you want to add a text field.
4. On the *TRACKER FIELDS* tab, click **Add Field**.
5. On the **Create Field** page, provide a name for the field.
6. Configure the shape of the field with the **Field Width** and **Field Height** fields.
7. To help users enter the right text values, select **Use Text Validation** and supply a regular expression that describes the appropriate values. This can help reduce errors and keep your team's data as meaningful as it can be. For more detailed instructions, see Validate Text Entries in a Tracker Artifact.
8. Click **Save Field**. The new field is created.

Create a "Select" Field

To let users choose values from a list that you define, create a "Select" field.

You can create up to 30 single-select fields and an unlimited number of multiple-select fields in a tracker.

CAUTION: Creating a large number of multiple-select and user fields may affect TeamForge's performance.

1. Click **Project Admin** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of existing trackers, click the tracker to which you want to add a "Select" field.
4. On the *TRACKER FIELDS* tab, click **Add Field**.
5. On the **Create Field** page, provide a name for the field.
6. Use the **Input Type** menu to specify whether users will be able to select one value or more than one. If you're going to make this a required field, pick one of the values to be the default value. This value is applied to existing artifacts and artifacts that are moved from another tracker.
7. Decide whether users *must* choose a value.
    - Required fields automatically appear on the **Submit Artifact** page.

> **NOTE:** If you make the field required, you must specify a default value. If you make a **User** field required, specify one or more default users. If you make a **Date** field required, the default is 'today'.

- For optional fields, select **DISPLAY ON SUBMIT** if you want the field to appear when a user first creates an artifact.
- To prevent the field from being used at all, select **DISABLED**. (By default, new fields are enabled.)

8. Use the **Values** section of the **Create Field** page to add more values for the user to choose from.
9. Keep adding values until you have the list of options you want, then click **Save Field**.

Create a People-picker Field

To let users choose other users from a list, create a “people picker” field.

The **Assigned to** field is a people-picker field that is present in every tracker. An artifact can be assigned to only one user at a time. You can give yourself more flexibility by adding any number of customer people-picker fields.

For example, your QA team may want to assign a person to track progress on an artifact while it is assigned to a developer. You might create a people-picker called “QA monitor” to specify who that person should be.

In the people-picker fields you create, users can select multiple users.

> **CAUTION:** Creating a huge number of user fields can slow down the site’s performance.

1. Click **Project Admin** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of existing trackers, click the tracker to which you want to add a people-picker field.
4. On the *TRACKER FIELDS* tab, click **Add Field**.
5. On the **Create Field** page, provide a name for the field.
6. On the **INPUT TYPE** menu, select *Select User(s)*.
7. In the **DEFAULT FILTER** field, choose whether the list of people available in your new field will include members of this project or everyone who is registered on the site.
8. Configure the size of the field with the **FIELD WIDTH** field.
9. Click **Save Field**. The new field is created.

Organize Tracker Fields

Most tracker artifacts ask the user for a lot of information. You can arrange the input fields in columns and rows to make it easier for users to find the fields they need.

1. Click **Project Admin** from the **Project Home** menu.
2. Click **Tracker Settings**.

3. From the list of current trackers, click the tracker whose fields you want to organize. Click the *TRACKER FIELDS* tab if it isn’s already showing.
4. If some fields seem to be logically connected to each other, create a section to bring them together.
    1. Click **Add Separator** and select **Section**. Give the section a short but descriptive label.
    2. In the list of fields, drag your new “Section Separator” row to a position that makes sense.
    3. Drag the appropriate fields under the Section Separator that you just created.
5. Within a section, arrange fields logically into columns.
    1. Click **Add Separator** and select **Column**. Give the column a short but descriptive label.
    2. In the list of fields, drag your new “Column Separator” to a position that makes sense.
    3. Drag the appropriate fields under the Column Separator that you just created.
    4. Create as many columns as you need. Drag a column separator above another column separator to move it to the left in the artifact entry form. Drag it below to move it to the right.
6. Within a column, group fields into rows if appropriate.
    1. Click **Add Separator** and select **Row**. Give the row a short but descriptive label.
    2. In the list of fields, drag your new “Row Separator” to a position that makes sense, then drag the appropriate fields under the Row Separator.

> **NOTE:** You can have rows and columns without sections.

Enable or Disable Tracker Fields

If a tracker field is disabled, it does not appear on the Artifact page. Most fields can be disabled.

Disabled fields are accessible only to tracker administrators on the *TRACKER FIELDS* tab.

You can enable or disable any field that is user-defined or configurable. However, you can’t disable all configurable fields. For example, the **Status**, **Priority**, **Category**, and **Planning Folder** fields can’t be disabled.

> **NOTE:** If your goal is to prevent users from entering data into a field when submitting an artifact, but still displayes the field on the **Edit Artifact** and **Tracker Search** pages, don’ disable the field. Instead, clear the **DISPLAY ON SUBMIT** option on the **Edit Tracker Field** page.

1. Click **Project Admin** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of current trackers, click the tracker you want to configure.
4. On the *TRACKER FIELDS* tab, select the fields you want to enable or disable.
    - Click **Disable** to remove them from the **Artifact** page.
    - Click **Enable** to allow them to be configured and displayed.

> **NOTE:** Data in disabled fields is still searchable, but disabled fields do not appear as inputs on the **Search** pages.

Configure Required Fields for a Tracker

If a field is set as required, users cannot create artifacts without completing it. Most tracker fields can be required or optional.

Each tracker can have its own required and optional fields. Required fields are marked with a blue asterisk (*) on the **Create Artifact** page.

> **NOTE:** When you make a field required, any field whose values depend on that fields values is also required. See Help Users Select Options in a Tracker Artifact for more information about dependednt field values.

1. Click **Project Admin** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of current trackers, click the one you want to configure.
4. On the *Tracker Fields* tab, click the name of the field you want to set as required or optional. By default, only the **Title**, **Description**, and **Status** fields are set as required.
5. On the **Edit Field** page, select or clear the **Required** option to make a field required or optional. Required fields automatically show up on the **Create Artifact** page.

   > **NOTE:** System-defined fields and the **Status** field are always required.

6. For optional fields, select or clear the **DISPLAY ON CREATE** option. This specifies whether the field will appear on the **Create Artifact** page.
7. Click **Save Field**.

Configure Tracker “Select” Field Values

To help users provide meaningful informatin, supply them with useful field values to choose from in the input fields in the tracker entry form.

> **TIP:** Once a tracker has been created, you may create one or more user-defined single-select or multiple-select fields, add predefined values to the fields, remove values, if required, enable or disable fields, and change the default values for fields.

1. Click **Project Admin** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of existing trackers, click the name of the tracker that you want to configure.
4. On the *TRACKER FIELDS* tab, click the name of the field whose values you want to edit.
5. On the **Edit Field** page, set up the field values you want users to see when they create a tracker artifact.
    - To define a new value, click **Add**.
    - To rename a value, edit the existing text. If you rename a value, the value is renamed in all existing artifacts.
    - To remove a value, check the box and click **Delete**. If you delete a value, the value is changed to **None** in all existing artifacts.
    - Select **DEFAULT VALUE** to set which option will be chosen if the user makes no selection. When you move a tracker artifact from one tracker to another, the default field value is the value that comes along.
    - When you edit the values of the **Status** field, you are also asked to describe what each value's status means, as shown in the **Values** section of the **Edit Field** page. This status meaning is used in Advanced Search to define which values are returned when searching for open or closed artifacts.
    - As always, when you create a new tracker, the default value for the '**Priority**' field is set as '4 - Low'. However, you can change the default value by editing this configurable single select field, '**Priority**'. You cannot delete or disable the **Priority** tracker field.
    - When you change the tracker fields, the values in the existing artifacts remain unchanged.
6. Click **Move Up** or **Move Down** to order the list the way want it.
7. Click **Save Field**.

All values are now available in the selection menu for the field.

Configure Tracker Units

You can estimate the value of efforts meaningfully in the form of units using the TeamForge tracker.

You can set any unit as the default for a tracker or planning folder and create any number of units. The default base unit is 'Hours' which you can rename, but not delete. You can also toggle between the burndown chart and effort values for any unit.

> **NOTE:** You cannot enter decimal values in the **Conversion** field. Deleting a unit will cause all the associated artifacts to express effort in the form of the base unit.

1. Click **Project Admin** from the **Project Home** menu.
2. Click **Tracker Settings**.

3. On the *UNITS* tab, click **Add**.
4. On the **Add Unit** page, enter the **UNIT NAME** and **CONVERSION** value for the unit.
5. Click **Save**.

Configure Default Tracker Columns

When you're looking at the artifacts list in a tracker, a planning folder or a team, you can select the columns you want to see, either for this session or permanently.

You set your column preferences for each tracker, planning folder or team independently. If your project administrator has set default columns for the entire project, your individual column choices override those settings.

1. Click **Trackers** from the **Project Home** menu.
2. Select a tracker, planning folder or team and click **COLUMNS > Configure**.
   - If you've already saved a column configuration, click it and skip the rest of these steps.
   - To go back to the default column configuration, click **System (default)** and skip the rest of these steps.
   - To set up a new configuration, click **Configure**.
3. Choose your columns.
   1. Move the columns you want from **Available Columns** to **Selected Columns**. **Artifact ID: Title, Priority** and **Status** are required columns.

      > **NOTE:** Selecting more columns can increase the time required to load the listing page.

   2. Remove any columns you don't need from **Selected Columns**.
   3. Use the move up and move down arrows to change the display order of the columns.
4. Apply your choices to your view of the tracker.
   - To use this arrangement this time only, click **Apply**. The next time you log in, you'll start with the default view again.
   - To save your column layout for repeated use, click **Apply and Save**, then give your arrangement a name. The next time you log in, you'll see the column arrangement you just selected. (If you've sorted the records in your view that sort order is saved too.)

   > **TIP:** If you are editing a column configuration that already exists, you can rename it by saving it under a new name.

5. To make the same set of columns appear every time you come to this tracker, planning folder or team, click **COLUMNS > Save** and from **Save Column Configuration** page, select **Make this my default view**.

6. To make the same set of columns appear for every user the first time they see any tracker, planning folder or team in the project, click **COLUMNS > Save** and from **Save Column Configuration** page, select **Make this the default view for all project members**.

The project default configuration you set is now the default configuration for all project members, unless they have created their own personal default column configuration.

Create a Tracker Workflow

To channel project member's work on tracker items, set up rules for how a tracker item can move forward.

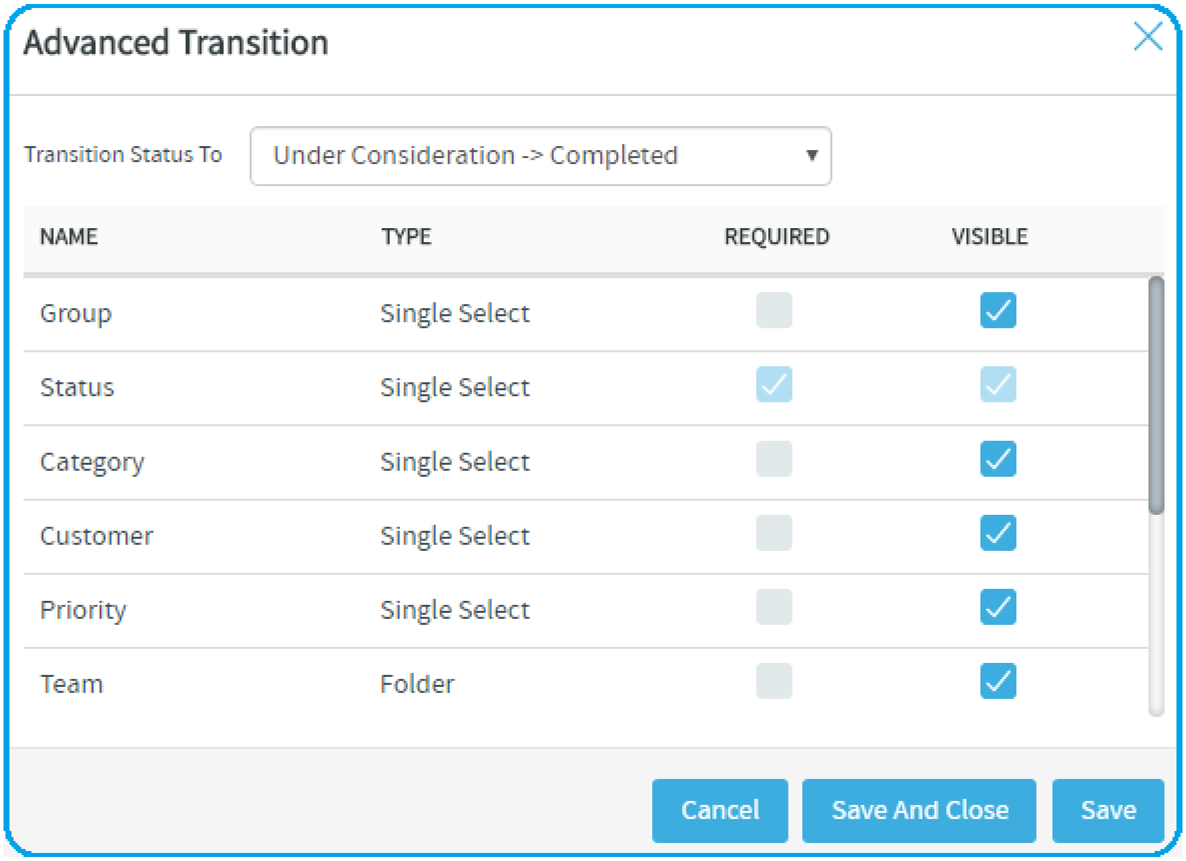
Before creating a tracker workflow, see that these criteria are met:

- You have a tracker to work with.
- The tracker has a set of statuses defined, such as “In progress” and “Ready for QA”.
- Roles exist, and you can assign project members to them.

A workflow is a sequence of changes from one status to another. You can define status transitions for any combination of tracker statuses in the tracker.

1. Click **Project Admin** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of existing trackers, select a tracker.
4. Click the *WORKFLOW* tab. The **Workflow** page lists all of your status values and the tracker workflow that you have configured.
5. On the **Workflow** page, click the status value for which you want to create a workflow.
6. On the **Edit Field Transition** page, select a status value from the **Create Transition to Status** drop-down menu.
7. Click **Add**. A new workflow is added. The **Any** workflow is changed to **Remaining Statuses**.
8. In the **ROLES** section, specify which users can make this changes. For example, only users with the QA Engineer role are allowed to change artifacts from **Open** to **Cannot Reproduce**.
9. Click the *Advanced Transition* link.

> **NOTE:** For any new unsaved transition, an alert is shown asking you to save the transition so that you can it in the **Transition Status To** drop-down list. Click **OK** to configure already saved transitions or click **Cancel** and then click **Save and View** in the **Edit Field Transition** page to save the unsaved transitions.



Advanced Transition

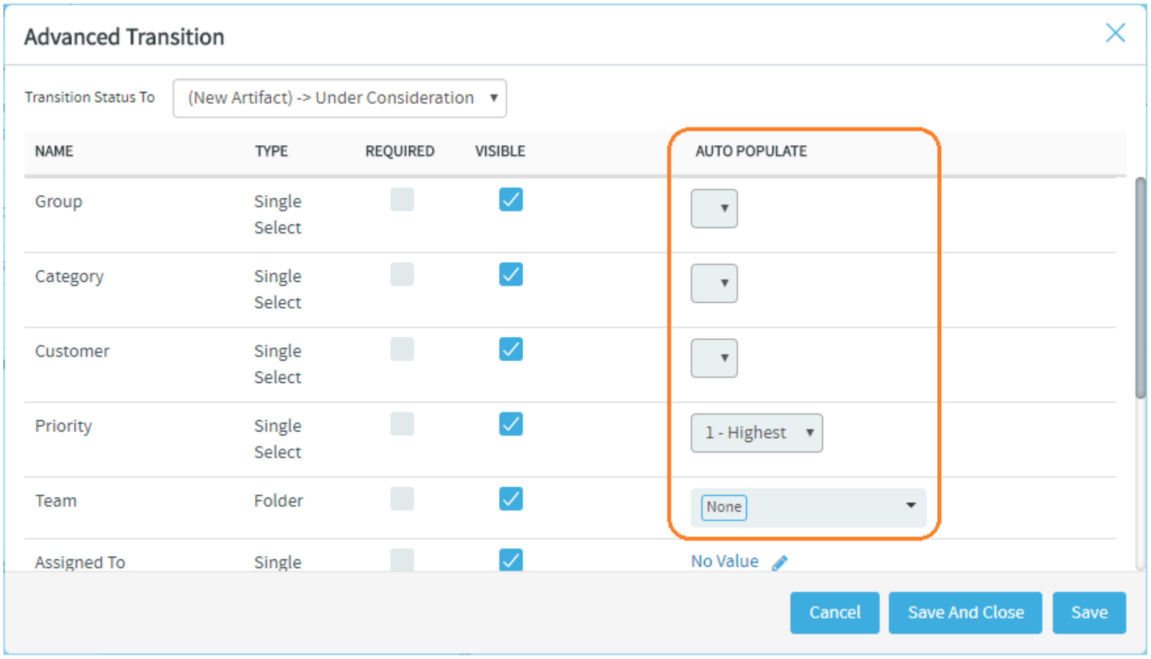
Transition Status To: Under Consideration -> Completed

NAME	TYPE	REQUIRED	VISIBLE
Group	Single Select	□	☑
Status	Single Select	☑	☑
Category	Single Select	□	☑
Customer	Single Select	□	☑
Priority	Single Select	□	☑
Team	Folder	□	☑

1. Select the transitions workflow for which you want to apply Advanced Transition settings from the **Transition Status To** drop-down list.
2. Select the **REQUIRED** check box against the fields for which the user *must* provide values. For example, the user must assign the tracker item to someone and enter a comment.

> **NOTE:** Fields whose values depend on a required parent field are automatically required. See Help Users Select Options in a Tracker Artifact for more information on parent and child fields.

3. Select or unselect the **Visible** check box for showing or hiding fields respectively for the selected status transition.
4. Select the values on the **AUTO POPULATE** column for the fields, which you want to get populated during the selected workflow transition.



Transition Status To: (New Artifact) -> Under Consideration

NAME	TYPE	REQUIRED	VISIBLE	AUTO POPULATE
Group	Single Select	□	☑	▾
Category	Single Select	□	☑	▾
Customer	Single Select	□	☑	▾
Priority	Single Select	□	☑	1 - Highest ▾
Team	Folder	□	☑	None ▾
Assigned To	Single	□	☑	No Value

Points to note:

- **Required** fields are always visible.
- Advanced Transition rules are applied when you create or edit artifacts in a tracker and only when edit artifacts in Planning, Task, and Kanban boards.
- Hidden field values, if updated via SOAP/REST APIs, are ignored.

10. Click **Save**.

The workflow is now saved. When a user submits or edits the status of a tracker artifact, he or she sees only the options that are allowed by the workflow.

Change a Tracker

A tracker's real-world uses often outgrow the name or description you gave it when you created it. When that happens, it is a good idea to update the tracker to reflect its changing role.

1. Click **Project Admin** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of existing trackers, click the tracker you want to edit, and click **Edit**.
4. On the **Edit Tracker** page, provide a new name or description for the tracker, and update the icon.
5. Update the units from **DISPLAY EFFORT IN** and click **Save**.

> **NOTE:** These units are configured in the **Units** page at the project level, and not at the planning folder level.

6. Select **INCLUDE FOREIGN CHILDREN** to include points and efforts from children artifacts across the projects in TeamForge.

> **NOTE:** In a parent artifact, enabling **CALCULATE POINTS** field sums and rolls up the points from all its children artifacts within the project. In this total, if you want to include children artifacts from other projects across TeamForge, have the **INCLUDE FOREIGN CHILDREN** option enabled.

7. If necessary, drag the tracker to a place in your tracker list that makes sense. The order you set here controls the order of every tracker list in your project.

Clone a Tracker

To save efforts in duplicating a tracker within the project and across the projects, choose cloning options available in TeamForge.

You can clone a tracker within a project along with the workflow. This can be done across the projects as well. If you are the Project Admin for the source and destination project, you can create new roles that are available in the destination project. However, the permissions associated with the role are not copied from the source project. A tracker admin cannot create new roles while cloning a tracker across projects.

1. Click **Project Admin** from the **Project Home** menu.
2. From the **Project Admin Menu**, click **Tracker Settings**.
3. Select **Clone External Tracker** from the drop-down list available in **Clone** button to clone a tracker from the source project.

> **NOTE:** To clone a tracker within the project, you can select the tracker from the **Tracker Settings** page and click **Clone Tracker**.

4. On the **Cloning Tracker** page, name the new tracker and enter a suitable description.
5. Enter **SOURCE TRACKER ID** of the tracker available in the source project and click **Next**. The **Cloning External Tracker** page appears.
6. On the **Cloning External Tracker** page, name the new tracker and enter a suitable description.
7. Click **Create**. The cloned tracker appears at the bottom of your list of trackers.

Auto Assign Tracker Artifacts

You can configure the tracker to automatically assign newly submitted artifacts to specific project members.

You can assign artifacts to individuals based on the values in the **Category** or *Release** field.

1. Click **Project Admin** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of existing trackers, click the tracker you want to configure.
4. On the *AUTO ASSIGNMENT* tab, select **Auto Assign By Category** or **Auto Assign By Release**.
5. For each category or release value, choose a project member from the **AUTOMATICALLY ASSIGN TO** drop-down menu. This menu contains all project members with the tracker edit permission. Click the **Search** icon to display a list of project members to whom you can auto-assign artifacts. Choose `None` if you do not want artifacts with a specific value automatically assigned.
6. Click **Save**.

Whenever a new artifact is submitted with a value in the relevant field, the assignee receives an email notification and the artifact appears on the assignee's **My Page**.

Help Users Select Options in a Tracker Artifact

You can help users cope with complex information by guiding them to eligible values in single-select fields.

Any tracker that manages real-world information will quickly become very complex. Users can be confused by a proliferation of "Select" fields. Confusion can lead to inconsistent data, which makes your job harder.

You can help relieve the complexity by showing users their eligible options in a given field based on values they have already selected in another field. You can create overlapping sequences of dependent fields, with as many levels as you need.

This simplifies things for the user, but for the tracker administrator it can quickly get complicated. So let's look at an example.

1. Click **Project Admin** from the **Project Home** menu.
2. Click **Tracker Settings** and create a tracker. For this example, let's call it the *Lunch Planning* tracker.
3. Create a single-select field and call it **Lunch type**. (For this example, we'll ignore the built-in fields, such as **Status** and **Assigned to**. We're just working with fields that you create.)

4. Create some values for the **Lunch type** field. Let's call them Buffet, Picnic, and Banquet.

   Each type of lunch will make sense in some kinds of locations and not others. For example, you would not normally plan a banquet lunch in a park. We are now going to make it easy for users to avoid making such a mistake.

5. Create a single-select field called **Location type**, with Lunch type as the parent field, and give it some plausible values.
    1. Start by adding an option called Beach. In the **Parent Values** column, choose Picnic, because that's the kind of lunch you would have at the beach. (The **Parent Values** column lists all the values in the parent field you selected.)
    2. Add a `Restaurant`. In the **Parent Values** column, hold down the **Ctrl** key and choose `Banquet` and `Buffet`, because either of those could be held at a restaurant.
6. Create a single-select field called **Location**, with `Location Type` as the parent field, and give it some values to choose from.
    1. Start with an option called `Happy Food Restaurant`. In the **Parent Values** column, select `Restaurant`, because that's the type of location that Happy Food Restaurant is.
    2. Add another option, `Hanalei Cove`. Under **Parent Values**, select `Beach`.
7. Save your work and go to the tracker whose settings you have been editing. Try selecting from the interdependent values you have just created.

Observe that the value you choose in the **Lunch Type** field controls which values are available to you in the **Location Type** field, and that selecting a value in **Location Type** in turn controls the values that appear in the **Location** field.

- When a user selects `Banquet` for a lunch type, they can select `Restaurant` but not `Beach` in the **Location Type** field. You will have less error correction to do, and users will avoid confusion.
- When a user selects Picnic for their lunch type, the **Location Type** field offers only `Park` and `Beach`. Now you will not have to go through and clean up after users who mistakenly choose to plan a picnic at Happy Food Restaurant, and the doorman at Happy Food Restaurant will not have to turn away users who mistakenly show up with picnic baskets.

> **IMPORTANT:** If you are used to defining your own Tracker fields, the ability to make field values depend on other values may change how your trackers work in ways you didn't anticipate. Keep these points in mind:

- Linking fields in this way doesn't modify existing data, but when users later modify fields that are linked, they will have to adhere to the relationships you set here.
- If a field has a parent, and that parent field also has a parent, the top-most parent field must have at least one value.
- When a field has a parent field that is required, the child field's default value is set to None. If that required parent field is deselected, the child field no longer has to be required, but Required remains the default.
- If you require a specific field value before an artifact can be placed in a given status, that field's children are subject to the same requirements. See Create a Tracker Workflow for more about controlling what status an artifact can be in.

- If you delete a field that contains values that another field's values depend on, the dependent field becomes a standard single-select field on its own.
- When you cut and paste an artifact from one tracker to another, only those field values that also exist in the new tracker come along with the artifact. If those values aren't valid under the dependency rules of the new tracker, they are still brought along.

Validate Text Entries in a Tracker Artifact

You can help users contribute useful information by testing their text entries against rules you configure.

Text fields can be error-prone because they invite free-form input. You can help users provide usable information by automatically rejecting values that don't match the needs of the tracker.

This simplifies things for the user, but for the tracker administrator it can be complicated. So let's look at an example.

> **NOTE:** If your goal is to require users to enter some value, whatever the value is, don't use text field validation. Select the Required option instead.

1. Click **Project Admin** from the **Project Home** menu.
2. Click **Tracker Settings** and create a tracker. For this example, let's call it the *Bugs* tracker. It will be used to record entomological specimens in a collection.
3. Create a text field and call it **Legs**. This is for users to record the number of legs each specimen displays.
4. Select **USE FIELD VALIDATION** and supply a validation rule that requires the user to enter a number. For example, if a given insect has six legs, you'll want the user to enter the numeral **6**, and not a string such as "six" or "several."

   Try this regular expression: `\d{1,3}`

   This rule requires the user to enter a number with one, two or three digits. Now, a user who means to record a centipede with 100 legs but enters 1000 by mistake will not be able to save the artifact until the error is corrected.
5. Enter a sample string to test your regular expression. Any part of the sample string that matches your regular expression appears under **Match Results**. If nothing appears, rework your regular expression until you get a match.
6. Create another text field and call it **Location**. This is where users will record the geographical spot where they collected the bug.

7. Select **Use Text Validation** and supply a validation rule that requires the user to enter a pair of geographical coordinates. For example, if a given insect was found outside CollabNet's California headquarters, you'll want the user to enter a string like `37.674689,-122.384652`, and not something like "Brisbane" or "Out on the lawn."

   Try this regular expression: `[-]?[0-9]*[.]{0,1}[0-9]{0,4}`

   This rule requires the user to enter two numbers, separated by a comma, in the general format of a pair of mapping coordinates.

   > **NOTE:** This particular regular expression does not guarantee that the coordinates are valid, just that they look like coordinates.

8. Save your work. In the tracker whose settings you have been editing, try entering a number greater than 999 in **Legs**, or a street address in **Location** The red **X** next to the field indicates that the text entry is incorrect. A green check indicates that the value meets the requirements.

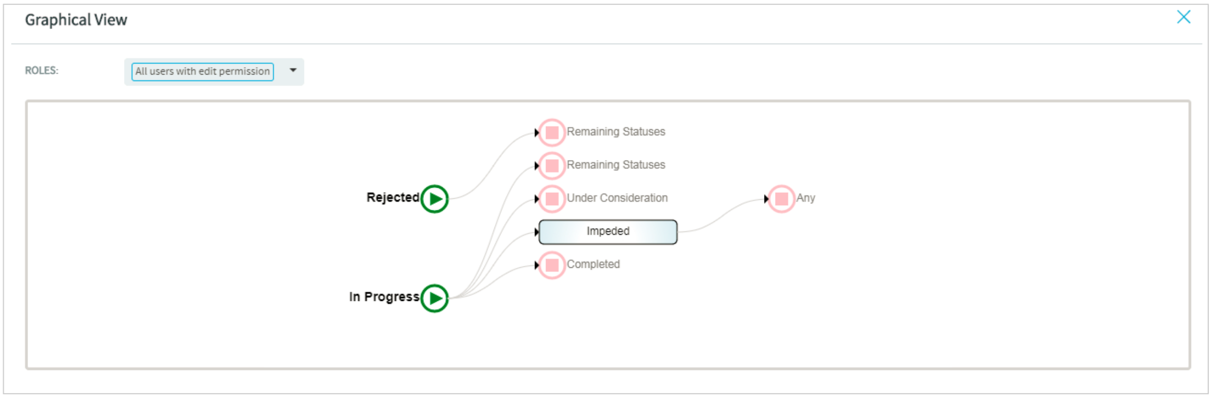
Notice that any field in which you are validating text entries is identified by **Text Entry (with Field Validation)** when listed on the *TRACKER FIELDS* tab.

Graphical Workflow Viewer for Trackers

You can now easily understand and interpret the tracker workflows meant for a specific user role with the help of a graphical workflow viewer.

The graphical representation of any workflow shows what the user with a specific role can do. However, the required fields, hidden fields, and auto populate fields set in the workflow are not shown in the graphical representation. To view the graphical representation of a workflow for a specific role:

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of existing trackers, select a tracker.
4. Click the **WORKFLOW** tab.
5. On the **Workflow** page, click the **Graphic View** button. The following screen with the graphical view of the workflows for the selected role is shown.



This workflow is read-only and non-editable. The **Roles** drop-down list contains the user roles (as seen in the tabular view) for the which workflows have been configured, in addition to the default roles: `All users with create permission` and `All users with edit permission`.

You can create a tracker artifact whenever you need to report and track a bug, feature request, support request, or other type of issue. You can also create a tracker artifact without logging into TeamForge just by sending an email to the tracker.

> **NOTE:** For more information on Trackers, see Set up Trackers.

Create a Tracker Artifact

Individual tracker entries are referred to as tracker artifacts, or just artifacts.

1. In any tracker, planning folder or teams view, click **Create Artifact** and select the tracker in which you want to create your artifact. By default, your new artifact is created in the tracker, planning folder or team you are currently looking at.
2. Answer the questions posed by the required fields.

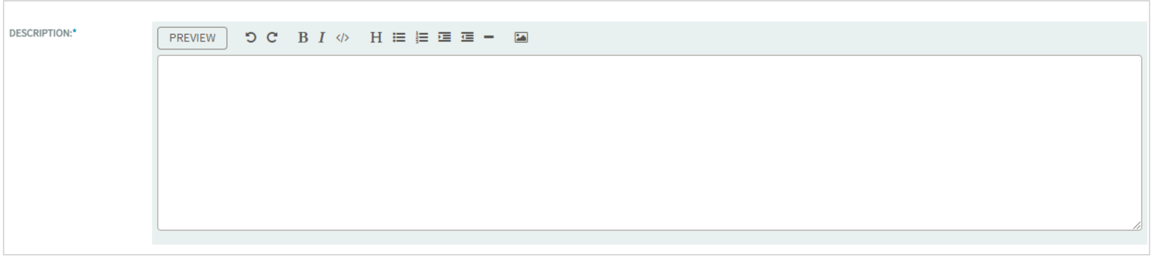
   > **NOTE:** Different trackers will have different combinations of fields to fill out, depending on what the tracker administrator has set up.

   1. Provide a **Title** and **Description** that summarize the issue or work item in a few words.

      > **TIP:** Descriptions help users learn how best to provide the informaiton you want from them. To maximize your chances of getting useful data, make your description as information as you can.

> **NOTE:** A new Markdown editor has been introduced for the description field. Now, you can change the format of the content of the description and make its look and feel better than ever. With this editor, you can preview your content, undo and redo the changes, set the bold and italics font styles, add a codeblock, include headers, add bulleted and numbered lists, choose to indent or outdent a paragraph, add horizontal rule and images.

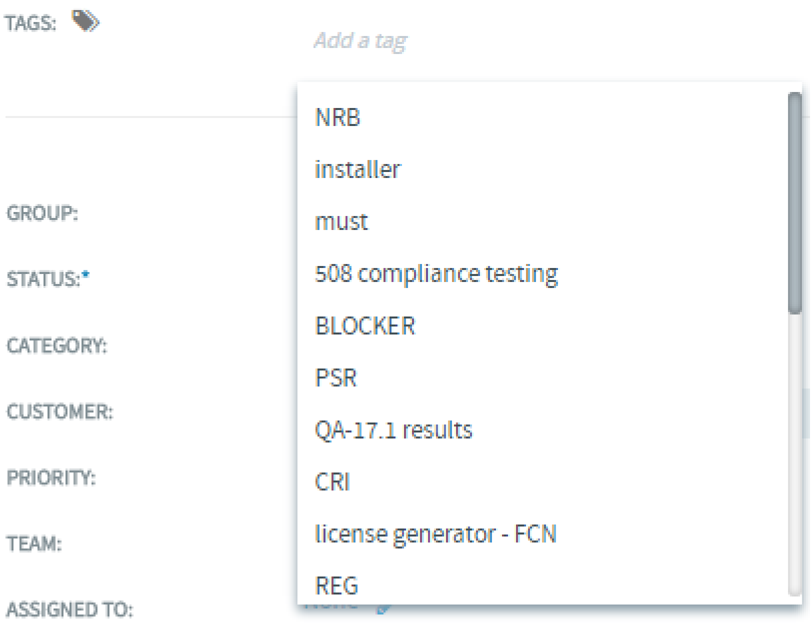
TeamForge uses Showdown—a bidirectional Markdown to HTML to Markdown converter written in Javascript. For more information, see the official Showdown Documentation. Here's an abridged version of the Markdown syntax documentation.



Artifacts support @mentions: Artifact description and comments now support @mentions and users called out via @mentions are added to the monitoring list. Include usernames with "@" as prefix (for example, @mphippard) to add users to the monitoring list.

> **NOTE:** Users called out via @mentions must have *Artifact View* permission to be added to the monitoring list.

2. Now project members with CREATE/EDIT permissions can also create new tags or add existing tags, if required from **Create Artifact** page. Tags creation from **Create Artifact** page enables you to create tags on the go and overrides the limitation of creating tags only from the **Tags** page. However, you cannot rename or delete a tag from **Create Artifact** page. Click the **Add tag** button next to get the list of tags mapped to your project. You can add up to a maximum of 10 tags to any artifact and a message is displayed if your try to add more than 10 tags. If the entered tag name is not available already. a context menu **Create a new tag** shows up for you to create a new tag with the desired tag name.

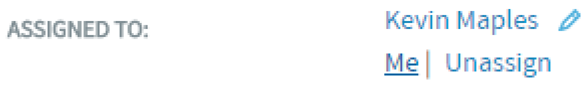


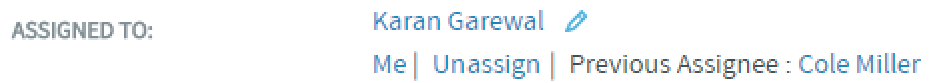
> **NOTE:** Wherever the tag widget is not applicable, the associated tags are displayed as read only tags. For example in **View Artifact** page.

3. For **Priority**, select a value that expresses the important or urgency of the work you are describing.
4. Assign the artifact to a team by selecting a name from the **Team** list.
5. If you want to assign it to a specific project member, choose a name from the **Assigned To** list. This list displays the names of all the project members, irrespective of the team you may have selected in the previous step.

> **NOTE:** If you project administrator has configured the tracker to automatically assign artifacts to project members, you can skip this step.

Reassigning artifacts can now be done in no time. Use the links under the **Assigned To** field to quickly unassign the artifact to “None”, and reassign the artifact to yourself or to the previous assignee. You can also click the “Re-assign” icon to search and reassign the artifact to any other user.





ASSIGNED TO: Karan Garewal

Me | Unassign | Previous Assignee : Cole Miller

    6. Select the planning folder that the work belongs to from the **Planning Folder** list.

3. Record any other information that may be appropriate. For example, if your project is using a Scrum-based methodology, your project manager may have provided a **Points** field to track estimates of relative effort.

4. Add a file attachment, if appropriate.

   > **NOTE:** When creating or editing an artifact, you can drag and drop any number of files with their overall size not exceeding 25 MB in the Attachments field. You can also select multiple files using the Browse button. Make sure that you add only files of restricted file types. For more information on attaching restricted files types, see New Features in TeamForge 17.8.

5. Save your changes.

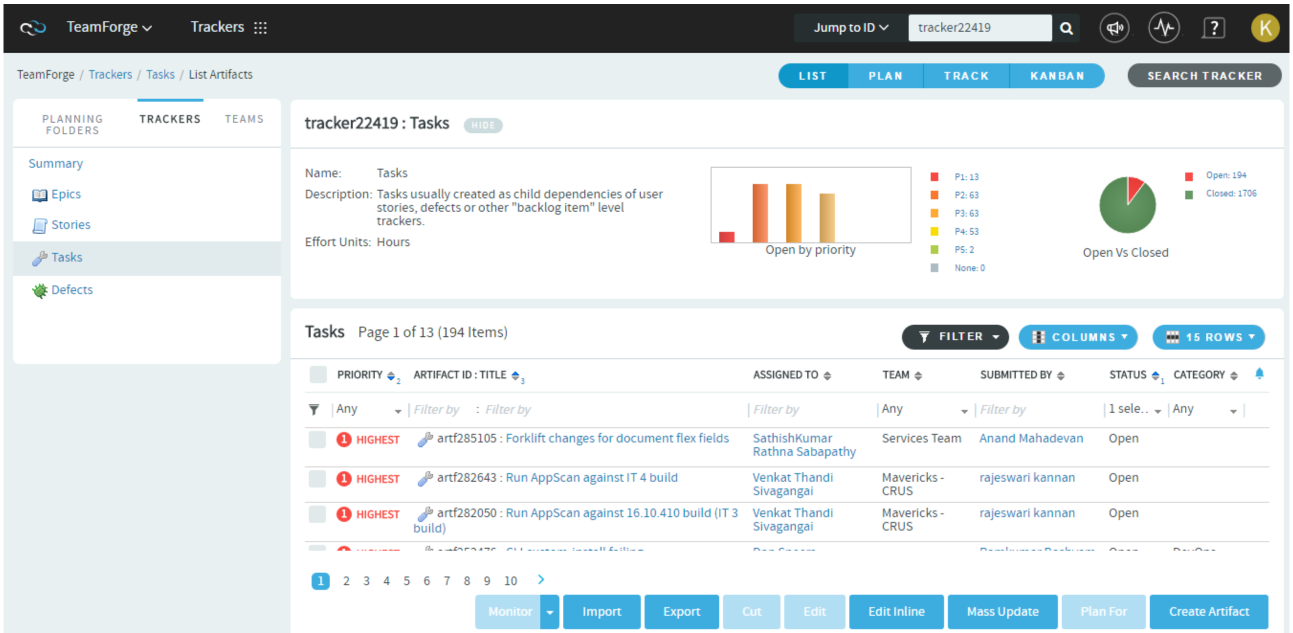
> **NOTE:** From 17.8 release, TeamForge is configured to send HTML emails to users assigned to and users monitoring the artifact you create or update. For more details, see HTML EMails.

Create a Tracker Artifact by Email

You do not have to be logged into TeamForge to submit a tracker artifact using email, but you must have the tracker submit permission for the tracker to which you are submitting.

Send an email message to `<tracker id>@<TeamForge server>`.

> **TIP:** You can find the tracker ID on the **List Artifacts** page.



tracker22419 : Tasks

Name: Tasks
Description: Tasks usually created as child dependencies of user stories, defects or other "backlog item" level trackers.
Effort Units: Hours

Tasks Page 1 of 13 (194 Items)

PRIORITY	ARTIFACT ID : TITLE	ASSIGNED TO	TEAM	SUBMITTED BY	STATUS	CATEGORY
Any	Filter by : Filter by	Filter by	Any	Filter by	1 sele..	Any
HIGHEST	artf285105 : Forklift changes for document flex fields	SathishKumar Rathna Sabapathy	Services Team	Anand Mahadevan	Open	
HIGHEST	artf282643 : Run AppScan against IT 4 build	Venkat Thandi Sivagangai	Mavericks - CRUS	rajeswari kannan	Open	
HIGHEST	artf282050 : Run AppScan against 16.10.410 build (IT 3 build)	Venkat Thandi Sivagangai	Mavericks - CRUS	rajeswari kannan	Open	

TeamForge maps your email to the tracker record like this:

Email field	Tracker field
To	Tracker email address
Subject	Artifact title
Body	Artifact description
Attachments	Attachments

Artifacts support @mentions: Artifact description and comments now support @mentions and users called out via @mentions are added to the monitoring list. Include usernames with "@" as prefix (for example, @mphippard) to add users to the monitoring list.

NOTE: Users called out via @mentions must have *Artifact View* permission to be added to the monitoring list.

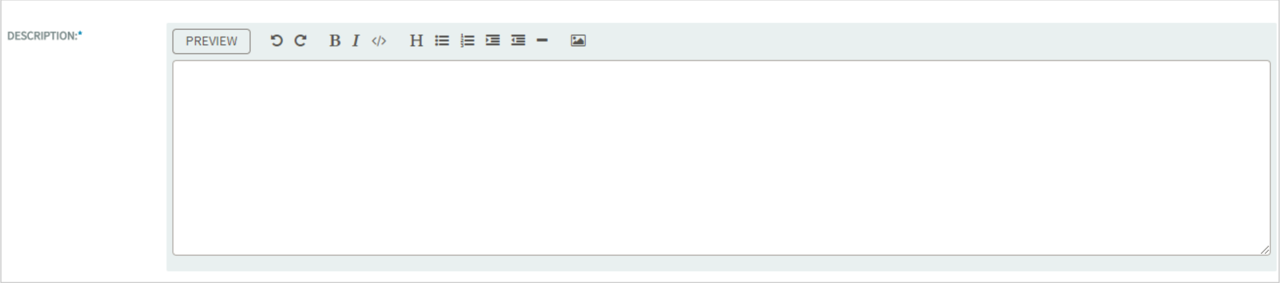
Updating the information in tracker artifacts is one important way that project members can work together effectively.

Support for @mentions

Artifact **Comments** and **Description** fields support @mentions and users called out via @mentions are added to the monitoring list. Usernames with the @ prefix (for example, @mphippard) are automatically added to the monitoring list. However, users called out via @mentions must have **Artifact View** permission to be added to the monitoring list.

Support for Markdown

The **Description** and **Comments** field supports Markdown. With the Markdown editor, you can change the format your text for better readability, preview your changes, undo and redo the changes especially while drafting the description or comments. Formatting options include: mark your text with bold and italics font styles, add a codeblock, include headers, add bulletted and numbered lists, indent a paragraph, add horizontal rule and images.



TeamForge uses Showdown—a bidirectional Markdown to HTML to Markdown converter written in Javascript. For more information, see the official Showdown Documentation. Here’s an abridged version of the Markdown syntax documentation.

Edit a Tracker Artifact

When work has been done on a tracker item, or more information is needed, the project member to whom the item is assigned should update the item’s status accordingly. Comments from other project members help the artifact’s owner decide how to handle the work.

For example, when the work defined in the tracker item is completed, change its status from **Open** to **Fixed**.

Your tracker administrator may have set up work flow rules that constrain your ability to do certain kinds of updates. For example, an administrator may have specified that only users with the “QA Engineer” role can change an artifact’s status from **Open** to **Closed**.

A project manager might also change an artifact’s priority, return it to the submitter for additional clarification, or assign it to a project member for resolution or action.

Generally, the project member to whom the tracker artifact is assigned will update the status. Any project member with the appropriate permissions can add comments.

> **TIP:** Each comment in an artifact or task has a unique ID with its own URL. To link directly to a particular comment, copy that comment’s URL and paste it into an email, a project page, or another comment. For example, to point to the third comment in artifact 12345, write `artf12345#3` in your comment. (If the artifact or task you are linking to is on a different site, give the complete URL, like this: `http://mysite.com/sf/go/task1234#3`.

1. Click **Trackers** from the **Project Home** menu.
2. On the list of project trackers, click the tracker you want to update.

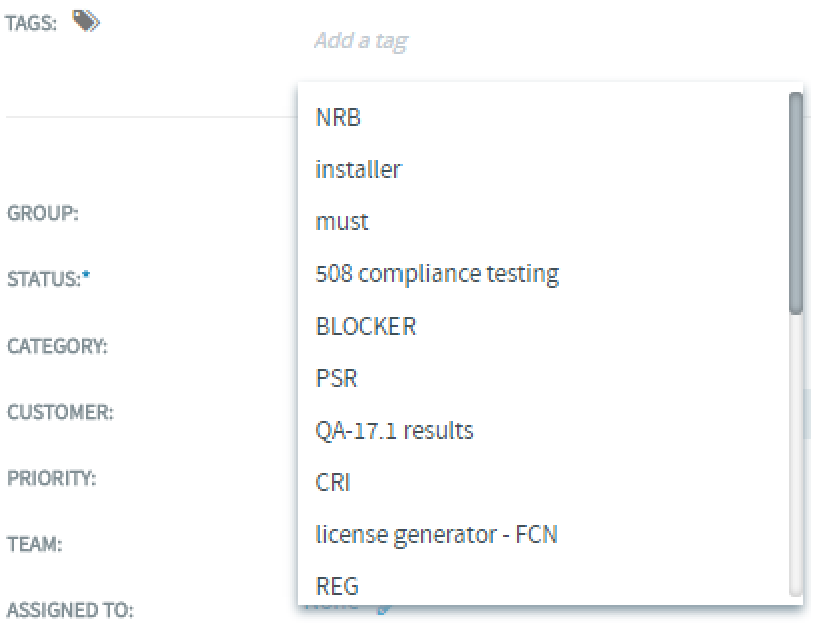
3. On the **List Artifacts** page, select the tracker artifact you want to update.
4. On the **View Artifact** page, click **Edit**.
5. On the **Edit Artifact** page, make your changes.
    1. Select a tracker type from the **Tracker** drop-down list to change the artifact's tracker type.
    2. Edit the artifact's title and description.
    3. Click **Save**.

6. On the **View Artifact** page, change any other fields if required, and click **Update**.

    Project members with **CREATE/EDIT** permissions can create new tags or use existing tags, if required, from the **Edit Artifact** page.

    Create tags on the go from the **Edit Artifact** page. However, you cannot rename or delete a tag from the **Edit Artifact** page.

    Click **Add Tag** next to the **Tags** field to get the list of tags mapped to your project. You can add up to a maximum of 10 tags to any artifact and a message is displayed if you try to add more than 10 tags. Type a tag name and if the tag name is not exitsting already, a contextual **Create a new tag** link shows up for you to create a new tag.



> **NOTE:** Wherever the tag widget is not applicable, the associated tags are displayed as read only tags. For example in **View Artifact** page.

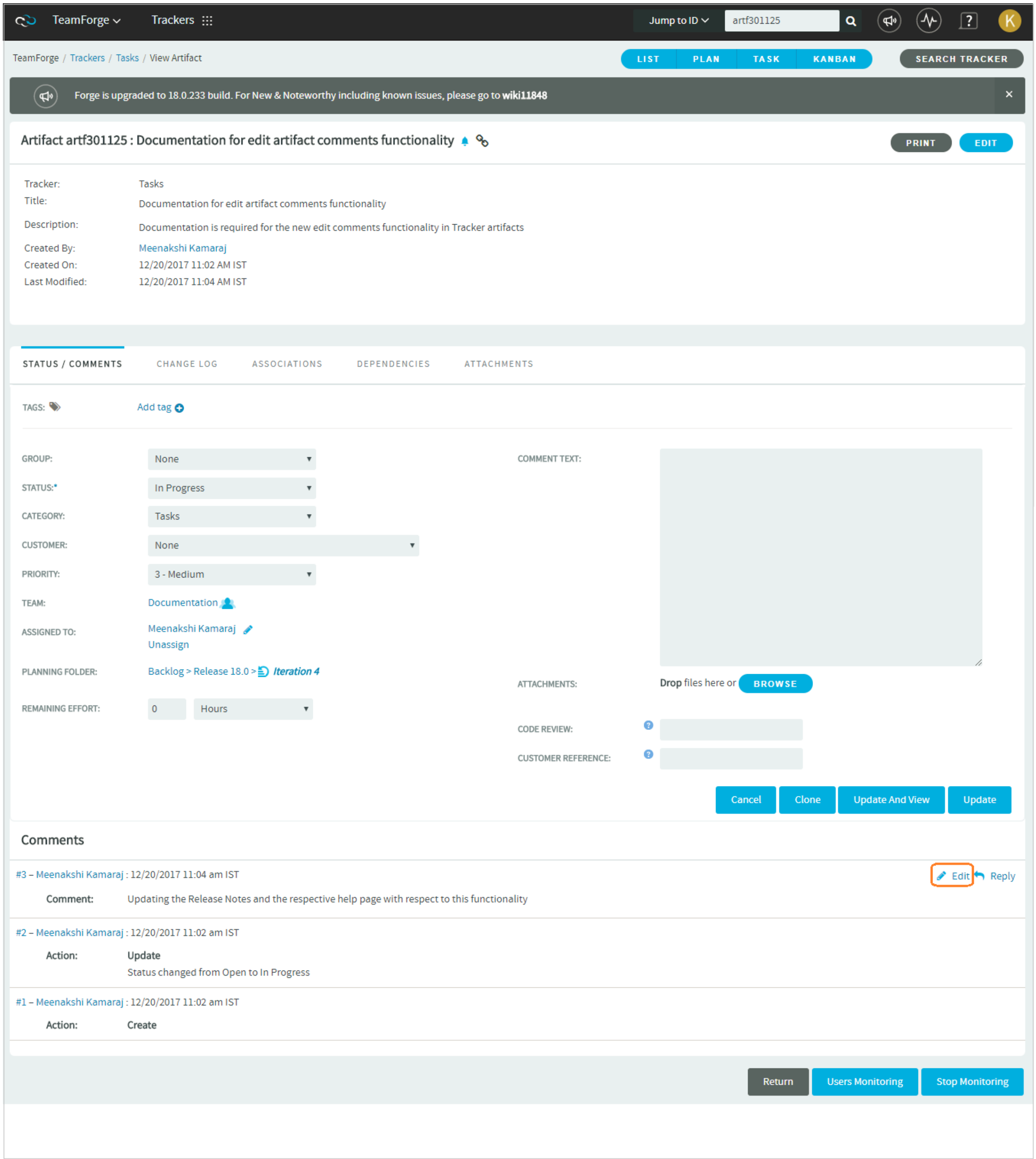
> **NOTE:** From 17.8 release, TeamForge is configured to send HTML emails to users assigned to and users monitoring the artifact you create or update. For more details, see HTML EMails.

> **NOTE:** When creating or updating an artifact, you can drag and drop any number of files with their overall size not exceeding 25 MB in the **Attachments** field. You can also select multiple files using the **Browse** button. Make sure that you add only files of restricted file types. For more information on attaching restricted files types, see PROHIBITED FILE TYPES.

All updates to the artifact are recorded in the **Comments** section of the **Status/Comments** tab.

Edit Your Comments

Have you got something wrong like a typo error when you added comments to a tracker artifact? Don’t worry !!! You can now edit your comments and make the corrections. You can see an **Edit** link for every comment that you have added. Click the **Edit** link.



Forge is upgraded to 18.0.233 build. For New & Noteworthy including known issues, please go to **wiki11848**

Artifact artf301125 : Documentation for edit artifact comments functionality

Tracker:	Tasks
Title:	Documentation for edit artifact comments functionality
Description:	Documentation is required for the new edit comments functionality in Tracker artifacts
Created By:	Meenakshi Kamaraj
Created On:	12/20/2017 11:02 AM IST
Last Modified:	12/20/2017 11:04 AM IST

TAGS:	Add tag
GROUP:	None
STATUS:*	In Progress
CATEGORY:	Tasks
CUSTOMER:	None
PRIORITY:	3 - Medium
TEAM:	Documentation
ASSIGNED TO:	Meenakshi Kamaraj Unassign
PLANNING FOLDER:	Backlog > Release 18.0 > *Iteration 4*
REMAINING EFFORT:	0 Hours
COMMENT TEXT:	
ATTACHMENTS:	**Drop** files here or BROWSE
CODE REVIEW:	
CUSTOMER REFERENCE:	

Comments

#3 – Meenakshi Kamaraj : 12/20/2017 11:04 am IST

Comment: Updating the Release Notes and the respective help page with respect to this functionality

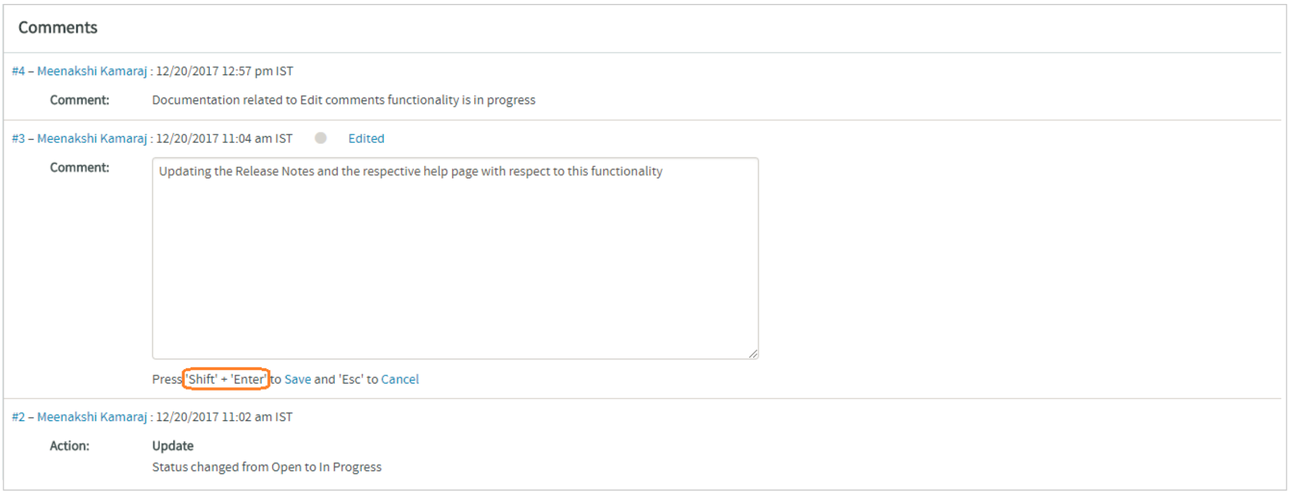
#2 – Meenakshi Kamaraj : 12/20/2017 11:02 am IST

Action: **Update**
Status changed from Open to In Progress

#1 – Meenakshi Kamaraj : 12/20/2017 11:02 am IST

Action: **Create**

Change the comments and press **Shift+Enter** or click **Save** to save the changes.



Comments

#4 – Meenakshi Kamaraj : 12/20/2017 12:57 pm IST

Comment: Documentation related to Edit comments functionality is in progress

#3 – Meenakshi Kamaraj : 12/20/2017 11:04 am IST ● Edited

Comment: Updating the Release Notes and the respective help page with respect to this functionality

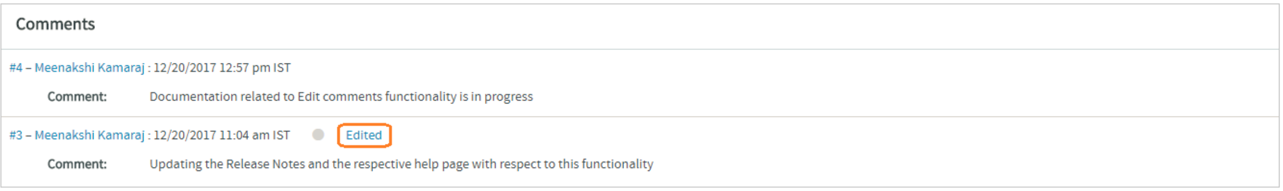
Press 'Shift' + 'Enter' to Save and 'Esc' to Cancel

#2 – Meenakshi Kamaraj : 12/20/2017 11:02 am IST

Action: Update
Status changed from Open to In Progress

You can edit the same comment any number of times.

All your edited comments have the status *Edited* to indicate that the comment was edited already. To see the history of the edits, click the *Edited* link.



Comments

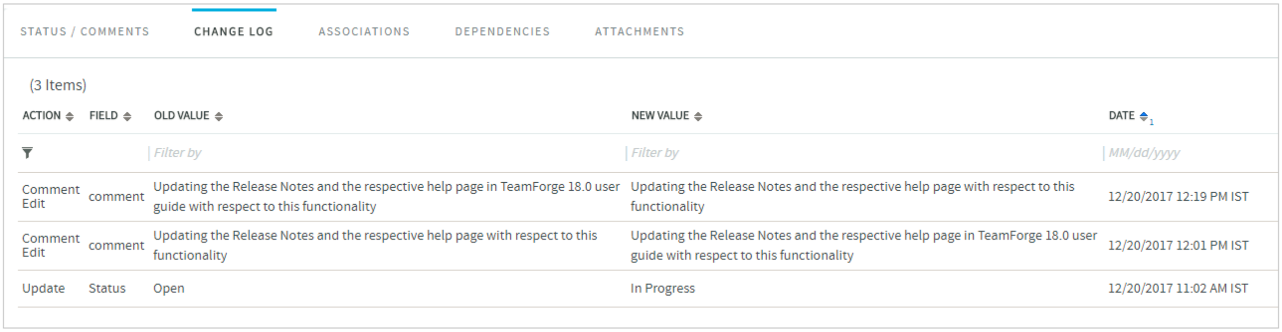
#4 – Meenakshi Kamaraj : 12/20/2017 12:57 pm IST

Comment: Documentation related to Edit comments functionality is in progress

#3 – Meenakshi Kamaraj : 12/20/2017 11:04 am IST ● Edited

Comment: Updating the Release Notes and the respective help page with respect to this functionality

The edit history of your comments and the comments added/edited by other users are shown on the **Change Log** section. The **Change Log** includes the all the changes with respect to the same comment.



(3 Items)

ACTION	FIELD	OLD VALUE	NEW VALUE	DATE
		Filter by	Filter by	MM/dd/yyyy
Comment Edit	comment	Updating the Release Notes and the respective help page in TeamForge 18.0 user guide with respect to this functionality	Updating the Release Notes and the respective help page with respect to this functionality	12/20/2017 12:19 PM IST
Comment Edit	comment	Updating the Release Notes and the respective help page with respect to this functionality	Updating the Release Notes and the respective help page in TeamForge 18.0 user guide with respect to this functionality	12/20/2017 12:01 PM IST
Update	Status	Open	In Progress	12/20/2017 11:02 AM IST

Whenever you edit your comments, email notifications are sent to the users monitoring the artifact.

Edit a Tracker Artifact by Email

To comment on a tracker artifact when you are not logged into TeamForge, send an email to the tracker, or reply to an automatic update email about the artifact.

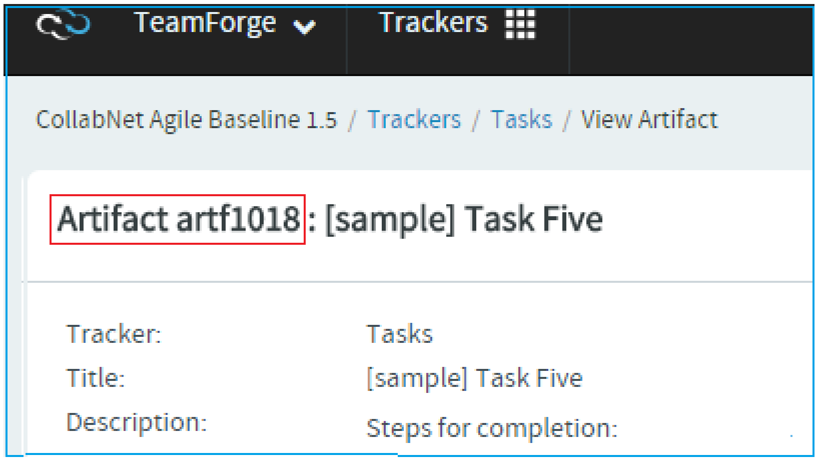
You can also add a comment or an attachment to a tracker artifact that you are monitoring by responding to the monitoring email notification.

You can also use email to add an attachment. But to edit any other fields, you must make the changes in TeamForge.

You do not have to be logged into TeamForge to edit a tracker artifact using email, but you must have the tracker edit permission for the tracker contaning the artifact you want to edit.

Send an email message to `<artifact id>@<CollabNet TeamForge server>`.

TIP: You can find the artifact ID on the **Artifact Details** page.


TeamForge
Trackers
CollabNet Agile Baseline 1.5 / Trackers / Tasks / View Artifact

Artifact artf1018: [sample] Task Five

Tracker: Tasks
Title: [sample] Task Five
Description: Steps for completion:

TeamForge maps your email to the tracker record like this:

Email field	Tracker field
To	Tracker email address
Subject	Artifact title
Body	Artifact description
Attachments	Attachments

Edit Multiple Artifacts (Mass Update)

When you have a large number of artifacts to update (for example, all the artifacts in a tracker, or a filtered list of artifacts in a planning folder), you can edit all the artifacts at once.

WARNING: Exercise caution while mass-updating multi-select fields. Mass-updating multi-select fields (such as Tags and multi-select lists) overrides any existing values with the new values you select.

NOTE: When you update two or more artifacts at a time, each user who is monitoring any of the changed artifacts gets a single email describing all the updates.

1. Click **Trackers** from the **Project Home** menu.
2. Go to the tracker or planning folder that contains the tracker artifacts you want to edit.

   ✓ In a tracker, you can use the filter to help you find the desired artifacts.

   ✓ You can select the **Planning Folder** tab and select the planning folder that contains the artifacts (cross-tracker artifacts) you want to edit.

   ✓ You can only see common fields when you select artifacts from more than one tracker for mass update.
3. Select the artifacts you want to edit, and click **Mass Update**.
4. For artifacts in a tracker, choose which ones you want to update.
   - **Selected**: Updates only the artifacts that you selected.
   - **Filtered Set**: Updates all artifacts returned by your filter, or all artifacts in the tracker if you did not apply a filter.

   TIP: Choose **Filtered Set** when the artifacts span multiple pages and you want to select them all.
5. Make your changes and click **Update**.

IMPORTANT: Some fields in your tracker may have values that depend on values in other fields, or use validation rules to ensure correct content. If your mass update operation breaks any such dependencies, you must fix the errors before running the mass update.

Edit Multiple Artifacts Inline

When you have a list of tracker or planning folder artifacts to update, you can edit all the artifacts inline at once.

1. Click **Trackers** from the **Project Home** menu.
2. Select the **Trackers**, **Planning Folders** or **Teams** tab, select the tracker, planning folder or team that contains the artifacts you want to edit inline.

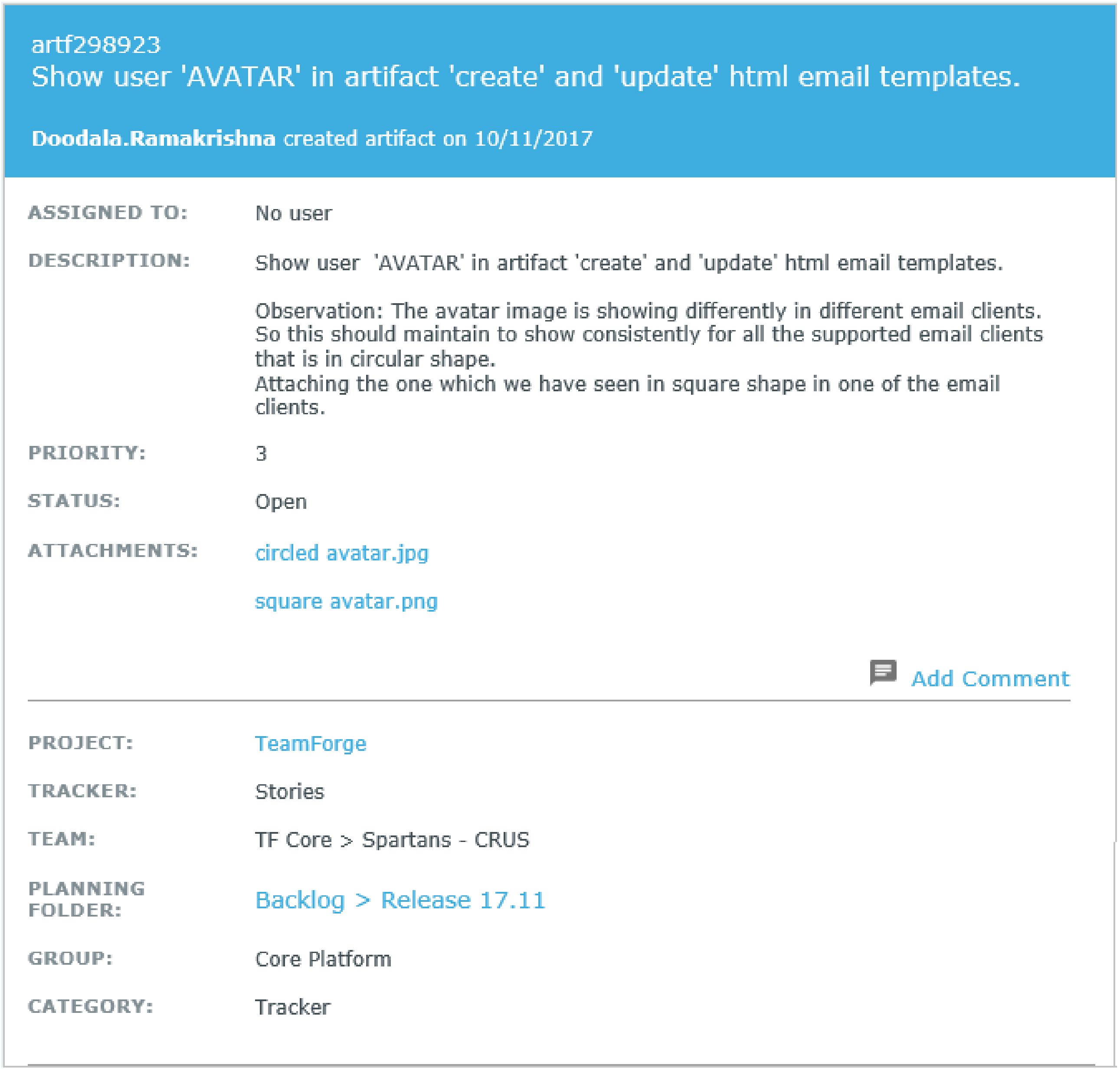
3. In the **List Artifacts** page, click **Edit Inline**. To help you identify editable columns, all non-editable columns are disabled. You can see a hand symbol when you hover the mouse pointer over the editable columns.
4. Click a field in a column and edit the selected artifact. For example;
    - Clicking a field in the **Assigned To** column lets you edit the person assigned to the specific artifact.
    - Clicking a field in the **Planned For** column displays the **Planning Folder** dialog box to let you change the planning folder for the specific artifact.
5. When you are done, click **Save**.

From its 17.8 release, when you create or update an artifact, TeamForge sends HTML emails to users assigned to and users monitoring that artifact.

HTML emails are formatted emails that look like a newsletter that you receive from a web service. These emails are embellished with colors, graphics, table columns and links. In this way, HTML emails enhance the look and feel of the emails and override the simple and plain features of Text emails that only include text.

By default, the HTML email configured in TeamForge contains the artifact details such as artifact id, artifact title, description, assigned to, customer, priority, status, attachments, and so on. Details of the fields with null values are not shown in the email.

When you create an artifact, TeamForge sends an email that looks like:



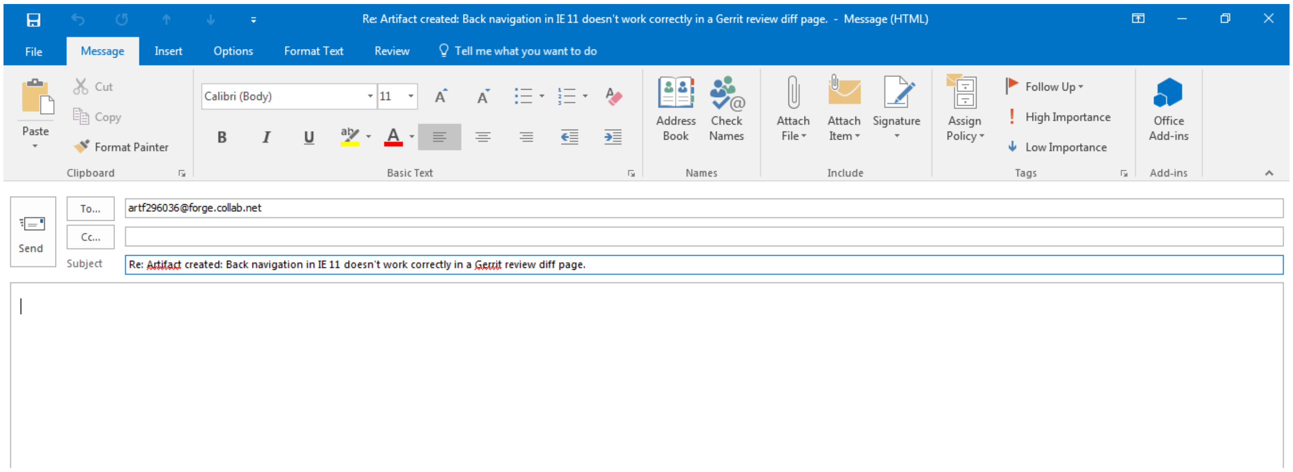
artf298923
Show user 'AVATAR' in artifact 'create' and 'update' html email templates.

Doodala.Ramakrishna created artifact on 10/11/2017

ASSIGNED TO:	No user
DESCRIPTION:	Show user 'AVATAR' in artifact 'create' and 'update' html email templates. Observation: The avatar image is showing differently in different email clients. So this should maintain to show consistently for all the supported email clients that is in circular shape. Attaching the one which we have seen in square shape in one of the email clients.
PRIORITY:	3
STATUS:	Open
ATTACHMENTS:	circled avatar.jpg square avatar.png

PROJECT:	TeamForge
TRACKER:	Stories
TEAM:	TF Core > Spartans - CRUS
PLANNING FOLDER:	Backlog > Release 17.11
GROUP:	Core Platform
CATEGORY:	Tracker

The sections of the above email that are highlighted in red denote that they are links. You can click them to go to the respective destination page to which the link takes you through. For example, clicking the Artifact id and title of the artifact shown on the banner of the email takes you to the artifact details page. If you click the **Add Comment** link, you will get the following screen. This is typically your mail client in which you can enter your comments and send it. You will notice that the To and the Subject fields are autofilled already. You just need to provide your comments and hit the **Send** button. The comment that you add here gets appended to list of existing comments in the artifact details page.

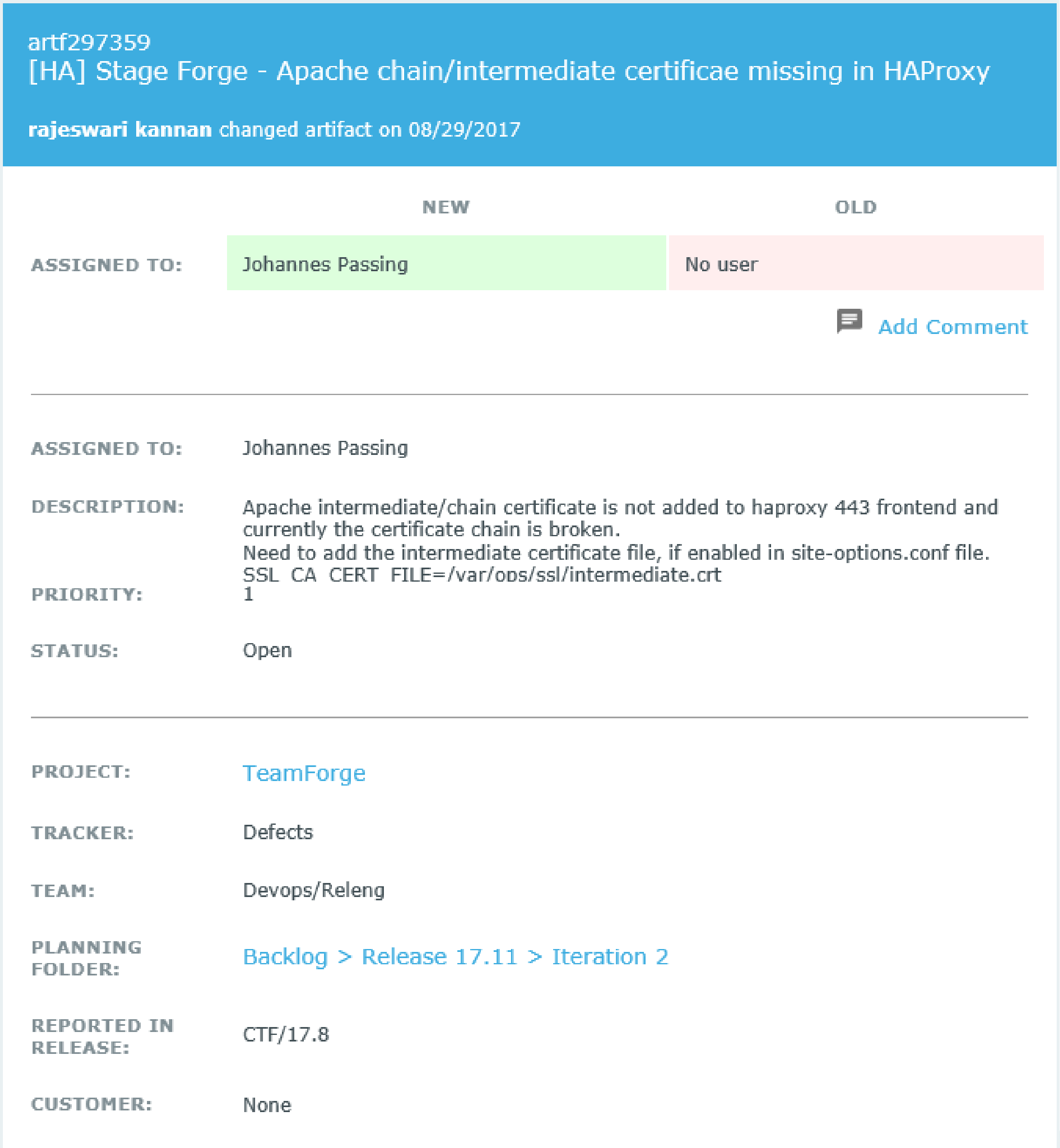


Re: Artifact created: Back navigation in IE 11 doesn't work correctly in a Gerrit review diff page. - Message (HTML)

To... artf296036@forge.collab.net

Subject: Re: Artifact created: Back navigation in IE 11 doesn't work correctly in a Gerrit review diff page.

The email sent when an artifact is updated contains the new and old values for fields such as Assigned To, Status, Priority, Planning Folder and so on. When you update an artifact, TeamForge sends an email that looks like:



artf297359
[HA] Stage Forge - Apache chain/intermediate certificae missing in HAProxy

rajeswari kannan changed artifact on 08/29/2017

	NEW	OLD
ASSIGNED TO:	Johannes Passing	No user

Add Comment

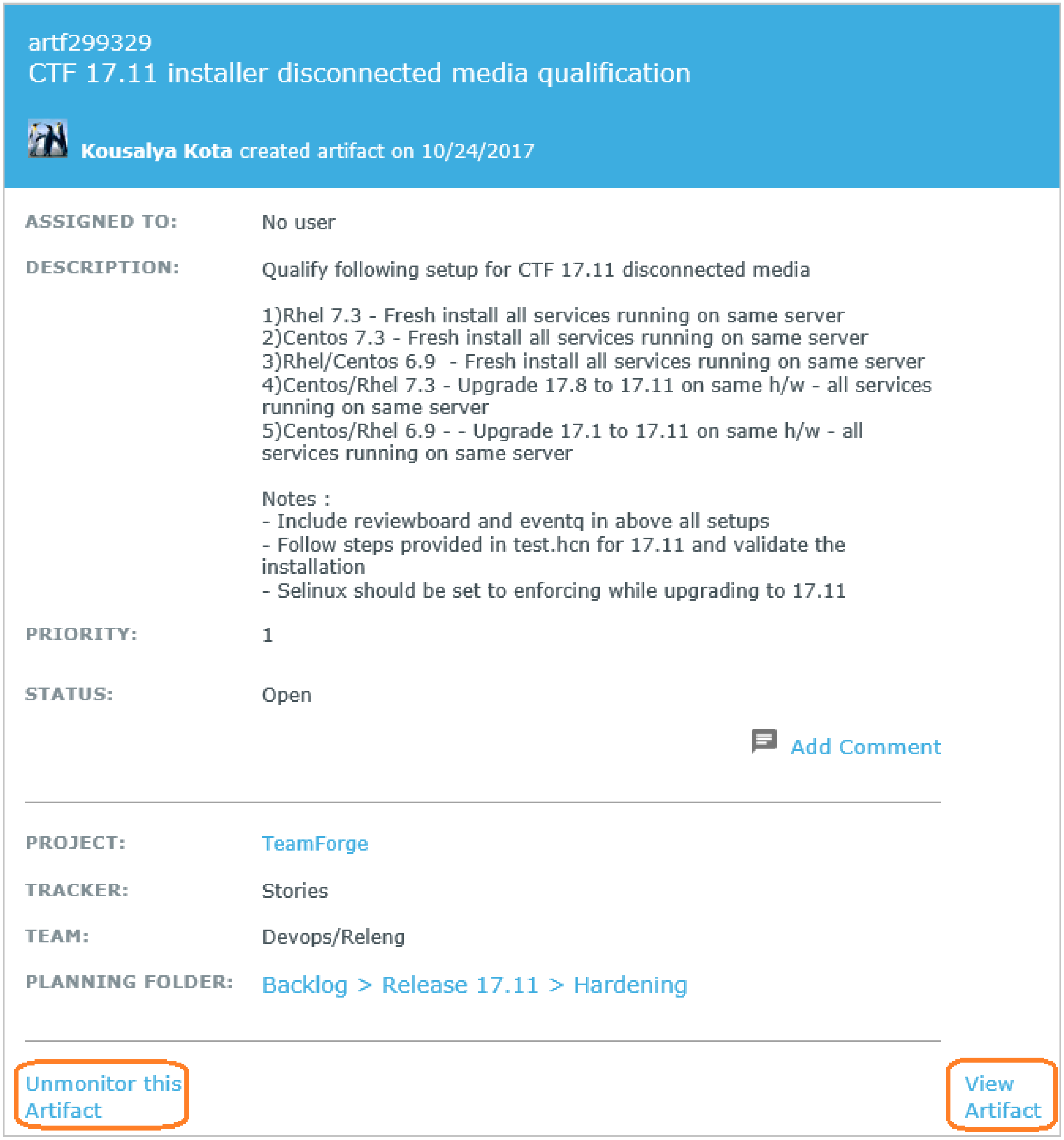
ASSIGNED TO:	Johannes Passing
DESCRIPTION:	Apache intermediate/chain certificate is not added to haproxy 443 frontend and currently the certificate chain is broken. Need to add the intermediate certificate file, if enabled in site-options.conf file. SSL CA CERT FILE=/var/ops/ssl/intermediate.crt
PRIORITY:	1
STATUS:	Open

PROJECT:	TeamForge
TRACKER:	Defects
TEAM:	Devops/Releng
PLANNING FOLDER:	Backlog > Release 17.11 > Iteration 2
REPORTED IN RELEASE:	CTF/17.8
CUSTOMER:	None

NOTE: Outlook for Windows, Outlook for Mac, and Office 365 Web Client are the email clients that support the HTML email format.

You can now be able to unmonitor any artifact right from the HTML emails that you receive whenever you create or update any artifact.

The following screen shows the email that you receive as a monitoring user when an artifact is created.



artf299329
CTF 17.11 installer disconnected media qualification

Kousalya Kota created artifact on 10/24/2017

ASSIGNED TO: No user

DESCRIPTION: Qualify following setup for CTF 17.11 disconnected media

1)Rhel 7.3 - Fresh install all services running on same server
2)Centos 7.3 - Fresh install all services running on same server
3)Rhel/Centos 6.9 - Fresh install all services running on same server
4)Centos/Rhel 7.3 - Upgrade 17.8 to 17.11 on same h/w - all services running on same server
5)Centos/Rhel 6.9 - - Upgrade 17.1 to 17.11 on same h/w - all services running on same server

Notes :
- Include reviewboard and eventq in above all setups
- Follow steps provided in test.hcn for 17.11 and validate the installation
- Selinux should be set to enforcing while upgrading to 17.11

PRIORITY: 1

STATUS: Open

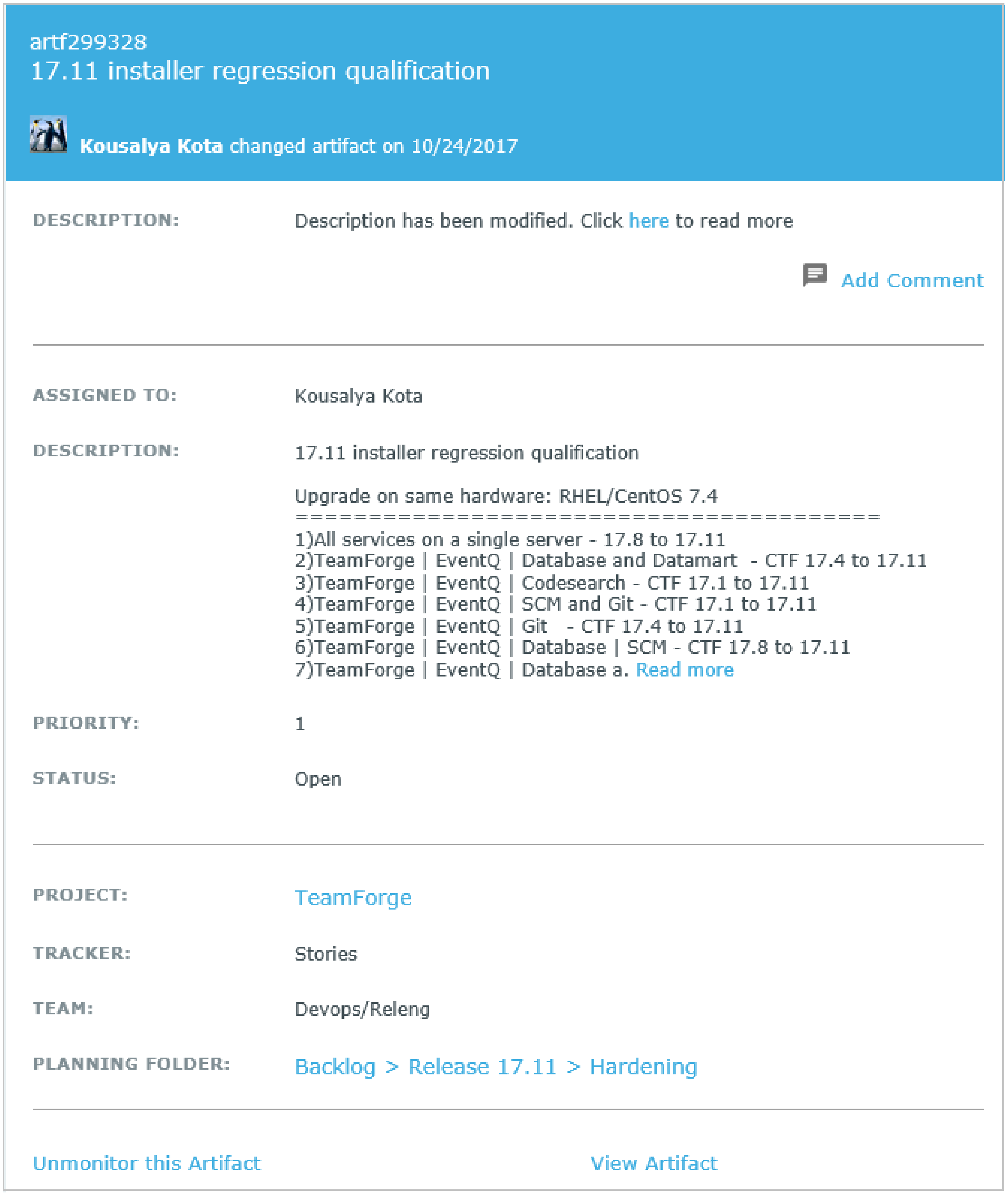
PROJECT: TeamForge

TRACKER: Stories

TEAM: Devops/Releng

PLANNING FOLDER: Backlog > Release 17.11 > Hardening

The following screen shows the email that you receive as a monitoring user when an artifact is updated.



artf299328
17.11 installer regression qualification

Kousalya Kota changed artifact on 10/24/2017

DESCRIPTION: Description has been modified. Click here to read more

ASSIGNED TO: Kousalya Kota

DESCRIPTION: 17.11 installer regression qualification

Upgrade on same hardware: RHEL/CentOS 7.4
===
1)All services on a single server - 17.8 to 17.11
2)TeamForge | EventQ | Database and Datamart - CTF 17.4 to 17.11
3)TeamForge | EventQ | Codesearch - CTF 17.1 to 17.11
4)TeamForge | EventQ | SCM and Git - CTF 17.1 to 17.11
5)TeamForge | EventQ | Git - CTF 17.4 to 17.11
6)TeamForge | EventQ | Database | SCM - CTF 17.8 to 17.11
7)TeamForge | EventQ | Database a. Read more

PRIORITY: 1

STATUS: Open

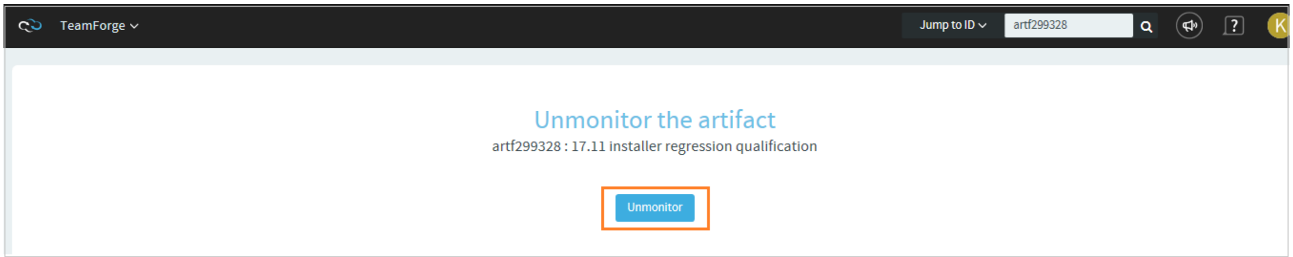
PROJECT: TeamForge

TRACKER: Stories

TEAM: Devops/Releng

PLANNING FOLDER: Backlog > Release 17.11 > Hardening

After you click the **Unmonitor this Artifact** link, you will be taken to the page as follows.



Unmonitor the artifact

artf299328 : 17.11 installer regression qualification

Click **Unmonitor** on this page after which a success message is displayed.

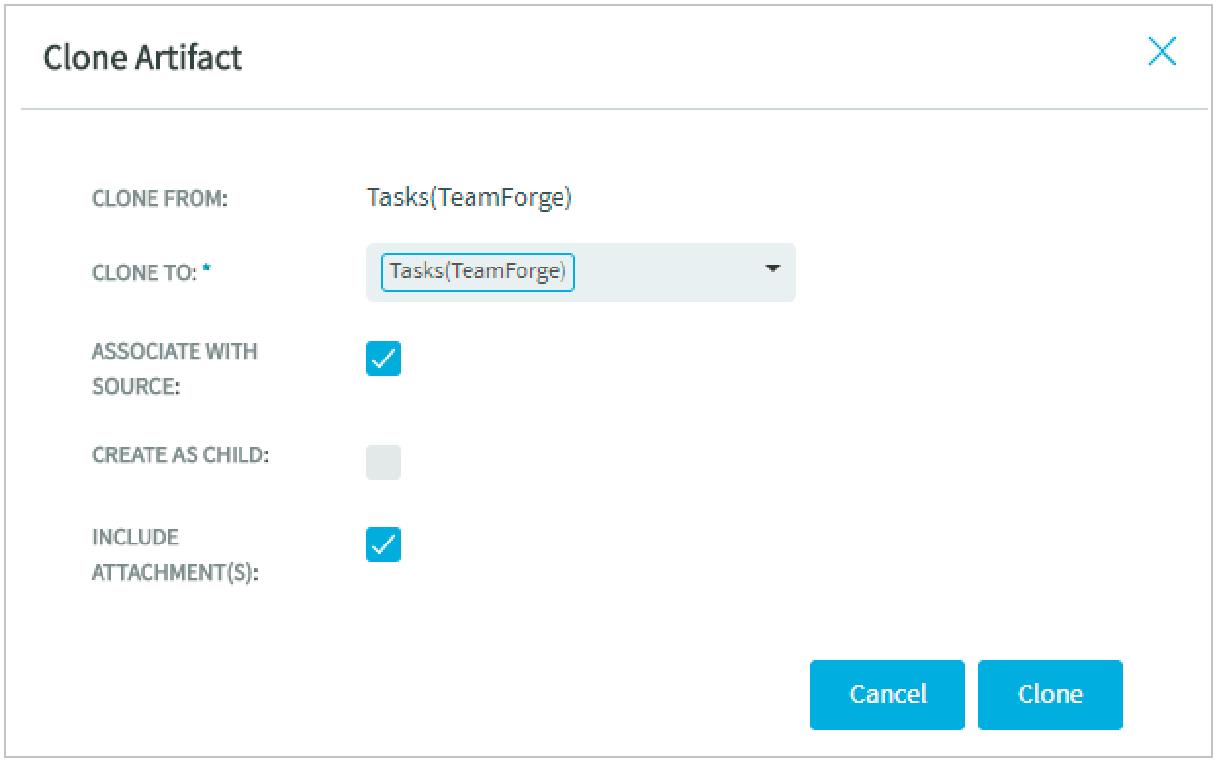
That's all ! You are not monitoring the artifact anymore.

In addition to this, a **View Artifact** link is also included at the right bottom of the email. You can click this link to go to the **View Artifact page** of the artifact.

Save effort in duplicating a tracker artifact within the project and across projects by cloning artifacts in TeamForge.

You can now clone an artifact from one tracker to another within the project or across projects. For system defined mandatory fields, artifact data are copied from the source tracker to the target tracker as long as the fields are available on both the source and target trackers. Values for any conflicting fields are set to “None” otherwise.

1. Click **My Workspace** and select a project.
2. Select **Project Home** > **Trackers** and select the tracker that contains the artifact to be cloned.
3. Open the tracker you want to clone. The **View Artifact** page appears.
4. Click **Clone**.



Clone Artifact

CLONE FROM: Tasks(TeamForge)

CLONE TO: * Tasks(TeamForge)

ASSOCIATE WITH SOURCE: ☑

CREATE AS CHILD: ☐

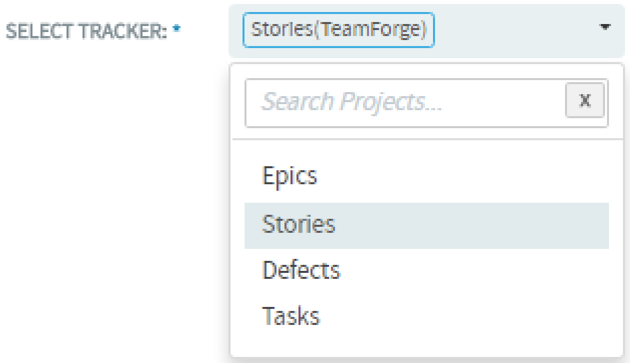
INCLUDE ATTACHMENT(S): ☑

The **Clone Artifact** dialog box appears.

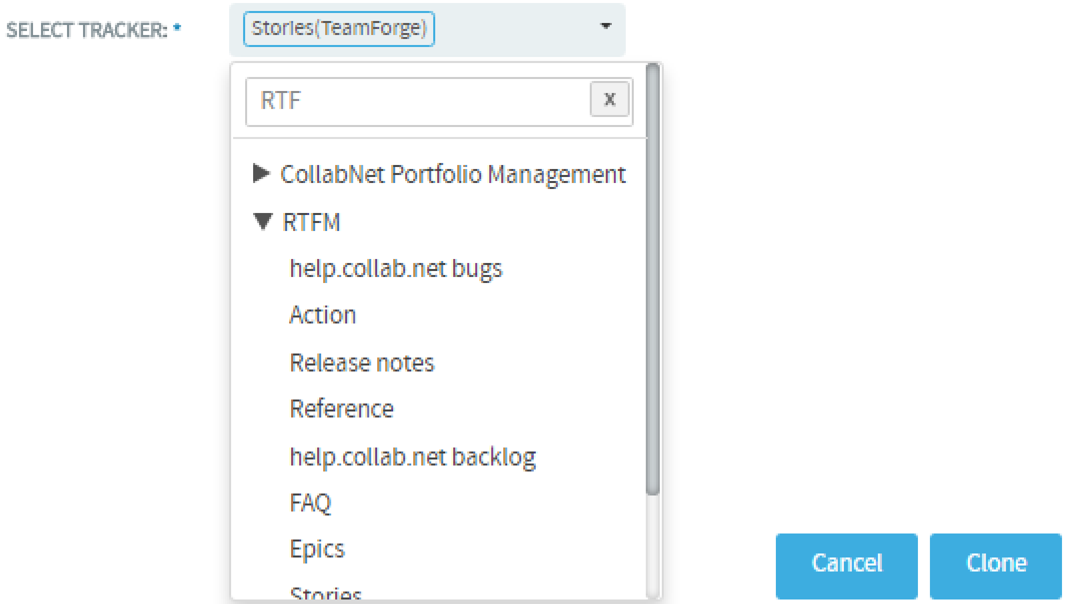
5. Select the **Associate With Source Artifact** check box (selected by default) to create an association with the source artifact.
6. Select the **Include Attachments** check box to include attachments from the source artifact.
7. Select the **Create as child** check box to make the artifact being cloned as the child of the source or the original artifact from which it is being cloned.

8. Click **Select Tracker** drop-down list and select the target tracker. You can either select a tracker from within the project in context or search and select a tracker from a different project. Use the **Search Projects** text box to find the project that has the target tracker and select the tracker.

Selecting a project from within the same project



Selecting a cross project tracker where you want to clone the artifact



9. Click **Clone**. The **Submit Artifact** page appears. For system defined mandatory fields, artifact data are copied from the source tracker to the target trackers as long as the fields are available on both the source and target trackers. Values for any conflicting fields are set to “None” otherwise.
10. Update the fields, if required, and click **Save** or **Save and View**. The artifact is cloned and saved in the target tracker.

As work proceeds on a tracker artifact, its focus may change. If this happens, it may be appropriate to move the artifact into a different tracker, or to a tracker in a different project.

To move a tracker artifact between projects, you must have either the tracker administration permission or tracker `View`, `Submit`, `Edit`, and `Delete` permissions in both the source and destination projects.

You can move one or more tracker artifacts in the same operation.

1. Click **Trackers** from the **Project Home** menu.
2. On the list of project trackers, click the title of the tracker containing the tracker artifact that you want to move.
3. Select the tracker artifact (or artifacts) that you want to move, and click **Cut**. CollabNet TeamForge removes the tracker artifacts and places them on the clipboard.
4. Go to the tracker into which you want to paste the tracker artifacts.

   > **NOTE:** To move the tracker artifacts into another project, first find the destination project using your **Projects** menu.

5. Click **Paste**.
   - If all artifact assignees have the appropriate permissions in the destination tracker, the tracker artifacts are now moved.
   - If not, the **Paste Artifacts** window shows you the number of tracker artifacts that cannot be reassinged automatically.
6. Choose one of the following methods to reassign any unassigned tracker artifacts.
   - **Reassign Artifacts to**: Assigns all unassigned tracker artifacts to the project member you select.
   - **Myself**: Reassigns all unassigned tracker artifacts to you.
7. Click **Next**.

The tracker artifacts are now moved to the destination tracker. The details of the move are recorded in the **Change Log** tab of the **Artifact Details** page.

- If the project members to whom the tracker artifacts are assigned have the appropriate permissions in the destination project, the tracker artifacts are automatically reassinged to the same project members. If not, you are offered several options for reassigning them.
- If both trackers use the same user-defined fields and field values, values in these fields are retained.
- If each tracker uses different user-defined fields or field values:
  - Values are set to the default value, if one was specified by the tracker administrator. Default values are mandatory for required fields.
  - Values are set to **None**, if no default value was specified by the tracker administrator.

When a large number of tracker artifacts makes it difficult to find the one you want, filter and sort the list to narrow down the possibilities.

Configure Tracker Columns

When you're looking at the artifacts list in a tracker, a planning folder or a team, you can select the columns you want to see, either for this session or permanently.

You set your column preferences for each tracker, planning folder or team independently. If your project administrator has set default columns for the entire project, your individual column choices override those settings.

1. Click **Trackers** from the **Project Home** menu.
2. Select a tracker, planning folder or team and click **COLUMNS > Configure**.
   - If you've already saved a column configuration, click it and skip the rest of these steps.
   - To go back to the default column configuration, click **System (default)** and skip the rest of these steps.
   - To set up a new configuration, click **Configure**.
3. Choose your columns.
   1. Move the columns you want from **Available Columns** to **Selected Columns**. **Artifact ID: Title, Priority** and **Status** are required columns.

      > **NOTE:** Selecting more columns can increase the time required to load the listing page.

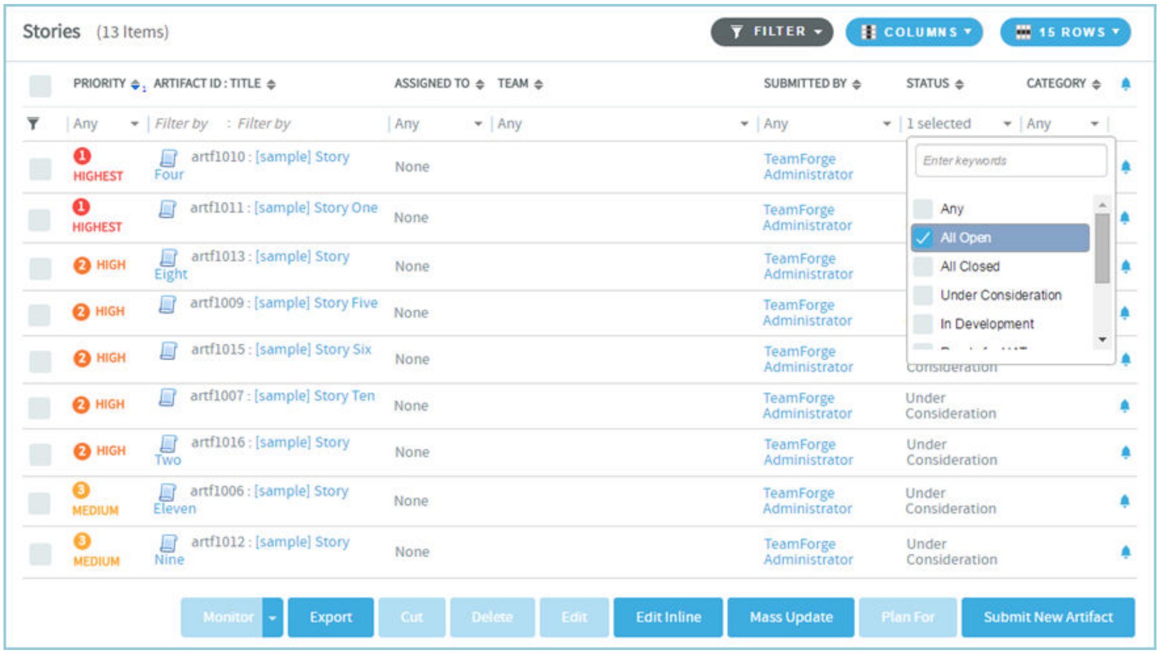
   2. Remove any columns you don't need from **Selected Columns**.
   3. Use the move up and move down arrows to change the display order of the columns.
4. Apply your choices to your view of the tracker.
   - To use this arrangement this time only, click **Apply**. The next time you log in, you'll start with the default view again.
   - To save your column layout for repeated use, click **Apply and Save**, then give your arrangement a name. The next time you log in, you'll see the column arrangement you just selected. (If you've sorted the records in your view that sort order is saved too.)

     > **TIP:** If you are editing a column configuration that already exists, you can rename it by saving it under a new name.

5. To make the same set of columns appear every time you come to this tracker, planning foler or team, click **COLUMNS > Save** and from **Save Column Configuration** page, select **Make this my default view**.

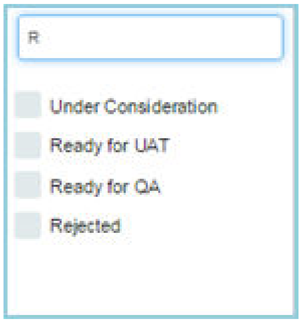
Tracker List Artifacts View

1. Click **Trackers** from the **Project Home** menu.
2. On the **Tracker Summary** page, click the title of the tracker in which you want to look at artifacts.
3. Specify the filter criteria in one or more filter fields (at the top of each column) and clic **FILTER**.
    - You can find a filter field at the top of each column in most of the tables in the TeamForge application.
    - The filter field could be a text box or a drop-down list with multi-select check boxes.



PRIORITY	ARTIFACT ID : TITLE	ASSIGNED TO	TEAM	SUBMITTED BY	STATUS	CATEGORY
Any	Filter by : Filter by	Any	Any	Any	1 selected	Any
1 HIGHEST	artf1010 : [sample] Story Four	None		TeamForge Administrator		
1 HIGHEST	artf1011 : [sample] Story One	None		TeamForge Administrator		
2 HIGH	artf1013 : [sample] Story Eight	None		TeamForge Administrator		
2 HIGH	artf1009 : [sample] Story Five	None		TeamForge Administrator		
2 HIGH	artf1015 : [sample] Story Six	None		TeamForge Administrator	Consideration	
2 HIGH	artf1007 : [sample] Story Ten	None		TeamForge Administrator	Under Consideration	
2 HIGH	artf1016 : [sample] Story Two	None		TeamForge Administrator	Under Consideration	
3 MEDIUM	artf1006 : [sample] Story Eleven	None		TeamForge Administrator	Under Consideration	
3 MEDIUM	artf1012 : [sample] Story Nine	None		TeamForge Administrator	Under Consideration	

    - You can type your filter criteria in the text boxes. The search text is case-insensitive.
    - You can also select the filter values from on e or more drop-down lists. By default, you can only select up to 10 filter values in a drop-down list. However, you can set a value that suits your requirement for the `FILTER_DROPDOWN_MAX_SELECTION` token in the `site-options.conf` file to increase or decrease the count.
    - **Filter-as-you-type**: You can find the **Enter keywords** text box in all filter drop-down lists. As you type your filter keyword, instant search results are shown in the drop-down list. For example, in the following illustration, typing “R” instanstly shows all statuses having the alphabet “R”. The search text is case-insensitive.



   - Some search filters may not appear if your site administrator has not enabled them.

4. After filtering, if you want to save a filter for future use:
   1. Click **FILTER** and select **Save** from the drop-down list. The **Save Filter As** window appears.
   2. Type a name for the filter in the **FILTER NAME** text box.
   3. Click **Save**. The filter is saved. You can view and select the saved filters at a later point in time by clicking the **Filter** drop-down list.

   > **NOTE:** You can save filters only in specific contexts. This feature may not be available in all the tables where you can filter table list items.

5. To delete save filters:
   1. Click **FILTER** and select **Delete** from the drop-down list. The **Select Filters To Be Deleted** window appears.
   2. Select one or more filters to delete.

   > **TIP:** Press and hold the **Ctrl** key to select more than one filter.

   3. Click **Delete**. A message such as `2 tracker filter(s) have been deleted successfully.` is displayed if the process was successful.
6. After filtering, if you want to clear the filters, click **FILTER** and select **Clear** from the drop-down list.
7. Use the up-down arrow at the top of any column to sort your list by that column.
   - Your primary sort column is identified by a superscript **1** next to the up-down arrow, and your secondary and third-level sort columns, if any, are likewise marked.
   - Click the up-down arrow again to reverse the sort order.
   - You cannot sort the list by the following fields (columns)—`Reported in Release`, `Fixed in Release`, `Tags`, multi-select flex fields and user flex fields.

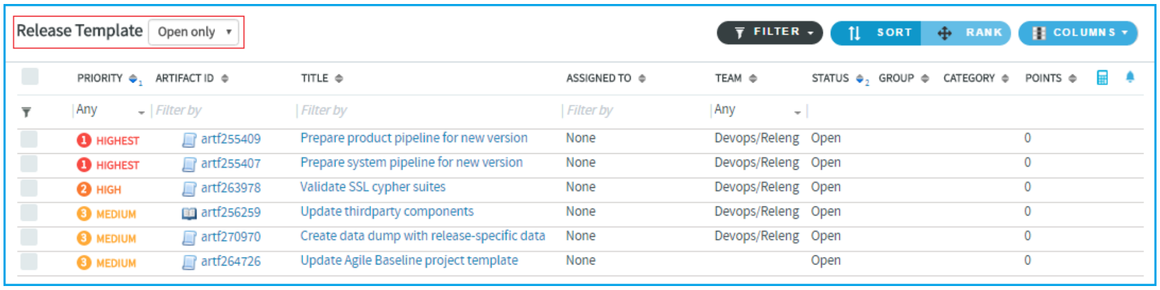
Planning Folder List Artifacts View

✓ In the Planning Folder list view, you can filter only by **Priority**, **Artifact ID**, **Title**, **Assigned To** and **Team** columns.
✓ Though the artifacts are listed in a tree view (parent artifact with its child artifacts), the filter is applicable only for the parent artifacts and not their children.
✓ The filter is available only in the Sort mode and not in the Rank mode.
✓ The filter that you set is retained even after you navigate to other pages and return to this page.

1. Click **Trackers** from the **Project Home** menu.
2. Click **Planning Folders**.
3. On the Summary page, click the planning folder in which you want to look at artifacts.
4. Specify the filter criteria in one or more filter fields (at the top of the filterable columns) and click **FILTER**.
   - The filter field could be a text box or a drop-down list with multi-select check boxes.
   - You can type your filter criteria in the text boxes. The search text is case-insensitive.
   - You can also select the filter values from on e or more drop-down lists. By default, you can only select up to 10 filter values in a drop-down list. However, you can set a value that suits your requirement for the `FILTER_DROPDOWN_MAX_SELECTION` token in the `site-options.conf` file to increase or decrease the count.
   - **Filter-as-you-type**: You can find the **Enter keywords** text box in all filter drop-down lists. As you type your filter keyword, instant search results are shown in the drop-down list. For example, in the following illustration, typing “R” instanstly shows all statuses having the alphabet “R”. The search text is case-insensitive.
5. After filtering, if you want to clear the filters, click **FILTER** and select **Clear** from the drop-down list.
6. Use the up-down arrow at the top of any column to sort your list by that column.
   - Your primary sort column is identified by a superscript **1** next to the up-down arrow, and your secondary and third-level sort columns, if any, are likewise marked.
   - Click the up-down arrow again to reverse the sort order.
   - You cannot sort the list by the following fields (columns)—`Reported in Release`, `Fixed in Release`, `Tags`, multi-select flex fields and user flex fields.

   Filter by artifact status

   In addition to the column-wise filter, you can also filter and view artifacts based on their status alone using the status drop-down list.



	PRIORITY	ARTIFACT ID	TITLE	ASSIGNED TO	TEAM	STATUS	GROUP	CATEGORY	POINTS
	Any	Filter by	Filter by	Filter by	Any				
	1 HIGHEST	artf255409	Prepare product pipeline for new version	None	Devops/Releng	Open			0
	1 HIGHEST	artf255407	Prepare system pipeline for new version	None	Devops/Releng	Open			0
	2 HIGH	artf263978	Validate SSL cypher suites	None	Devops/Releng	Open			0
	3 MEDIUM	artf256259	Update thirdparty components	None	Devops/Releng	Open			0
	3 MEDIUM	artf270970	Create data dump with release-specific data	None	Devops/Releng	Open			0
	3 MEDIUM	artf264726	Update Agile Baseline project template	None		Open			0

7. Select **All**, **Open only** or **Closed** from the drop-down list and all the relevant artifacts (including child artifacts) are displayed. For example, select **Open only** and all the artifacts with associated status values such as ‘Under Construction’, ‘Open’ and ‘In Progress’ are displayed.

> **NOTE:** The statuses **All**, **Open only** and **Closed** are associated with user-defined status values while configuring a tracker field on the **Edit Tracker Field** page (**Project Admin > Tracker Settings**). For more information, see Configure Tracker **Select** Field Values.

Teams List Artifacts View

- In the Teams list view, you can filter only by **Priority**, **Artifact ID**, **Assigned To** and **Planned For** columns.
- The filter that you set is retained even after you navigate to other pages and return to this page.

> **NOTE:** The Sort and Rank modes are not available on the Teams list view.

Filter by columns

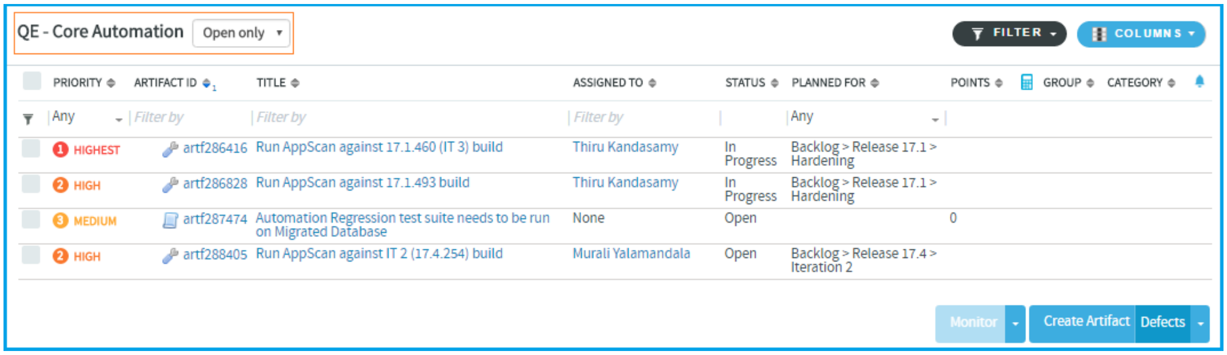
1. Click **Trackers** from the **Project Home** menu.
2. Click **Teams**.
3. On the Summary page, click the planning folder in which you want to look at artifacts.
4. Specify the filter criteria in one or more filter fields (at the top of the filterable columns) and click **FILTER**.
    - The filter field could be a text box or a drop-down list with multi-select check boxes.
    - You can type your filter criteria in the text boxes. The search text is case-insensitive.
    - You can also select the filter values from on e or more drop-down lists. By default, you can only select up to 10 filter values in a drop-down list. However, you can set a value that suits your requirement for the `FILTER_DROPDOWN_MAX_SELECTION` token in the `site-options.conf` file to increase or decrease the count.
    - **Filter-as-you-type**: You can find the **Enter keywords** text box in all filter drop-down lists. As you type your filter keyword, instant search results are shown in the drop-down list. For example, in

the following illustration, typing “R” instanstly shows all statuses having the alphabet “R”. The search text is case-insensitive.

5. After filtering, if you want to clear the filters, click **FILTER** and select **Clear** from the drop-down list.
6. Use the up-down arrow at the top of any column to sort your list by that column.
   - Your primary sort column is identified by a superscript **1** next to the up-down arrow, and your secondary and third-level sort columns, if any, are likewise marked.
   - Click the up-down arrow again to reverse the sort order.
   - You cannot sort the list by the following fields (columns)—`Reported in Release`, `Fixed in Release`, `Tags`, multi-select flex fields and user flex fields.

Filter by artifact status

In addition to the column-wise filter, you can also filter and view artifacts based on their status alone using the status drop-down list.



QE - Core Automation Open only

PRIORITY	ARTIFACT ID	TITLE	ASSIGNED TO	STATUS	PLANNED FOR	POINTS	GROUP	CATEGORY
Any	Filter by	Filter by	Filter by		Any			
1 HIGHEST	artf286416	Run AppScan against 17.1.460 (IT 3) build	Thiru Kandasamy	In Progress	Backlog > Release 17.1 > Hardening			
2 HIGH	artf286828	Run AppScan against 17.1.493 build	Thiru Kandasamy	In Progress	Backlog > Release 17.1 > Hardening			
3 MEDIUM	artf287474	Automation Regression test suite needs to be run on Migrated Database	None	Open		0		
2 HIGH	artf288405	Run AppScan against IT 2 (17.4.254) build	Murali Yalamandala	Open	Backlog > Release 17.4 > Iteration 2			

7. Select **All**, **Open only** or **Closed** from the drop-down list and all the relevant artifacts (including child artifacts) are displayed. For example, select ** only** and all the artifacts with associated status values such as ‘Under Construction’, ‘Open’ and ‘In Progress’ are displayed.

> **NOTE:** The statuses **All**, **Open only** and **Closed** are associated with user-defined status values while configuring a tracker field on the **Edit Tracker Field** page (**Project Admin > Tracker Settings**). For more information, see Configure Tracker **Select** Field Values.

If the filter returns too many results or not enough, try the search facility. The Tracker contains a comprehensive search system that enables you to find a specific artifact or set of artifacts quickly.

Search for Tracker Artifacts

You can find tracker artifacts by supplying a keyword, an artifact ID, a date range or some other value. You can also search comments, attachments, and user-defined fields.

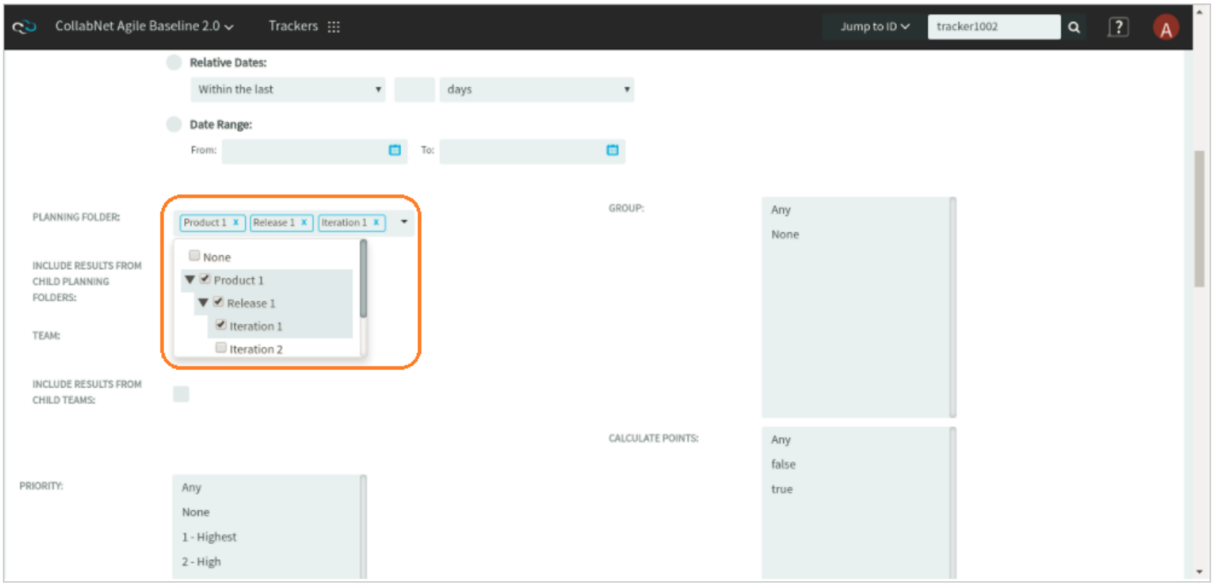
1. Click **Trackers** from the **Project Home** menu.
2. Set the scope of your search.
    - To search all the trackers in your project, click **Search Trackers**.
    - To search a specific tracker, click the tracker you want to search, then click **Search Tracker** in that tracker's list view.
3. In the *Tracker Search Criteria* section, enter the keywords to search for.

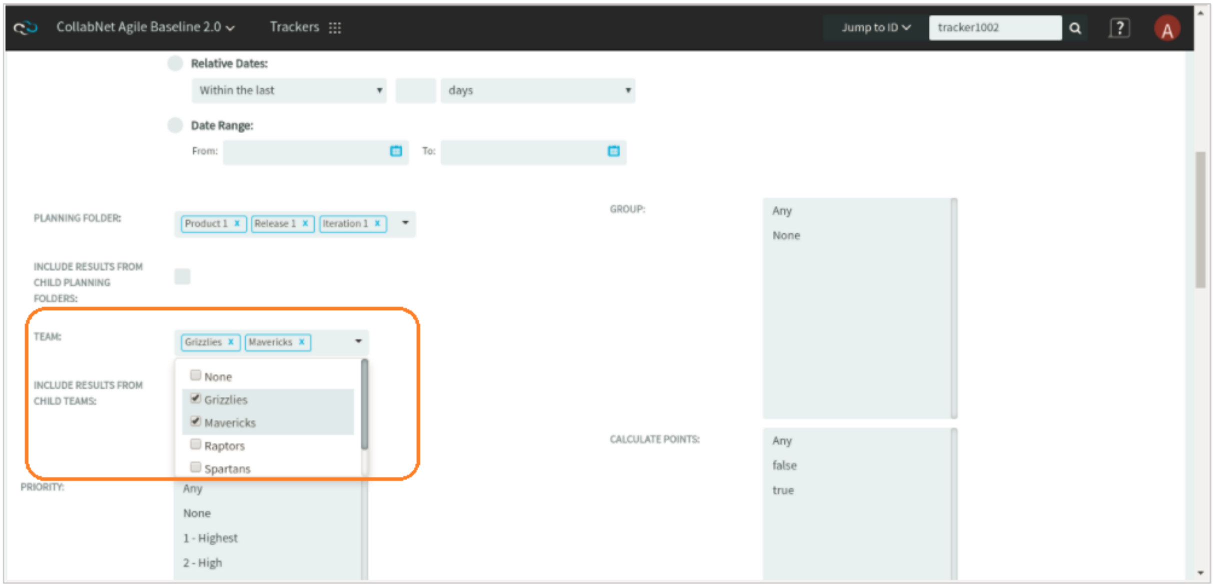
> **NOTE:** Wildcards are allowed.

4. Select the elements of the artifact to search. For example, to search for a keyword in the title and description fields, select the **Title** and **Description** check boxes. If you want to search all the text fields (including **Title**, **Description** and user-defined text fields), select the **All Text Fields** check box.

5. To search for a keyword in the comments field, select **Include Comments**.

    When searching the tracker, you can now select one or more planning folders or teams from the respective fields on the **Search Tracker** page.





○ Relative Dates:
Within the last ▾ | | days ▾

○ Date Range:
From: | To:

PLANNING FOLDER: Product 1 × Release 1 × Iteration 1 × ▾

INCLUDE RESULTS FROM CHILD PLANNING FOLDERS: □

TEAM: Grizzlies × Mavericks × ▾

□ None
☑ Grizzlies
☑ Mavericks
□ Raptors
□ Spartans

INCLUDE RESULTS FROM CHILD TEAMS:

PRIORITY: Any, None, 1 - Highest, 2 - High

GROUP: Any, None

CALCULATE POINTS: Any, false, true

6. If you know the artifact’s ID, enter that.
7. Click the calendar icon to select dates, if appropriate. You can also specify relative dates such as “Within the last 7 days”.
8. Use as many of the remaining tracker search criteria as seems useful.
9. Click **Search**.

All tracker artifacts matching your search criteria are displayed.

> **NOTE:** If your search included multiple trackers. the icon next to each artifact in your search results can help you identify which tracker that artifact belongs to.

Find your Own Artifacts

To narrow your scope, try searching only for artifacts assigned to or submitted by you.

1. Click **Trackers** from the **Project Home** menu.
2. On the list of project trackers, click the title of a tracker.
3. In the tracker artifact list view, click **Search Tracker**.
4. In the *Tracker Search Criteria* section, select the **Use running search** options in the **Assigned To** or **Submitted By** fields.
5. Click **Search**.

All tracker artifacts matching your search criteria are displayed.

Save a Search for Tracker Artifacts

To reuse a search that you devised for tracker artifacts, save the search criteria.

To be able to save your search, you must first select a tracker, and create and run your search.

1. In the *Tracker Search Results* section, click **Save Search from Results**.
2. In the **Save Search As** page, enter a descriptive name for the search, and click **Save**.

The search appears under **My Saved Tracker Searches**. The name of the search, the tracker for which it was run, the criteria used, and a system-assigned ID are displayed.

Share a Saved Tracker Search

To enable other users to use a search that you have devised, save it as a shared search.

To share your search with other users, you must be a tracker administrator.

1. In the *Tracker Search Results* section, click **Save Search from Results**.
2. In the **Save Search As** page, enter a descriptive name for the search.
3. Select the **Shared Search** option.

The search appears under **Shared Tracker Searches**. The name of the search, the tracker for which it was run, the criteria used, the user who created the search, and a system-assigned ID are displayed.

Run a Saved Search

To find artifacts in the current tracker or a different one in the project, run a saved search in the appropriate context.

1. Click **Trackers** from the **Project Home** menu.
2. On the list of project trackers, click the titile of a tracker.
3. In the tracker artifact list view, click **Search Tracker**.
4. Select the **Show My Searches Across Trackers** and **Show Shared Searches Across Trackers** options.
5. Expand the *My Saved Tracker Searches* and *Shared Tracker Searches* sections. The saved searches for all trackers in the project are displayed.
6. To run a search for the current tracker, click the **Search** or **Search Here** link below the search name in the required search. All artifacts in the current tracker, that match the search criteria, are displayed.
7. To run a search for a different tracker, click the **Search There** link below the tracker name in the required search. All artifacts that match the search criteria in the selected tracker, are displayed.

Refine a Saved Search

Refine your saved searches and keep them updated.

To be able to refine (edit) a previously saved search, you must first run the search and click **Refine Search**.

1. Click **Trackers** from the **Project Home** menu.
2. Click **Search Trackers**.
3. Click **My Saved Tracker Searches**.
4. Click the **Search** link under the saved search (in **Saved Search Name** column) you want to refine. The saved search is run and results appear.
5. Click **Refine Search**. The Tracker Search Criteria for the saved search shows up.
6. Modify the search criteria and click **Save Search**. The *Save Search As* dialog box appears.
7. Type a name and click **Save**. You may save the search with the same name, particularly if your have already shared the search with others. Otherwise, you can save the search with a new name too.

The search is saved with your new search criteria.

Remove a Saved Search

To remove a saved search for tracker artifacts, delete it from the list of personal or shared searches.

> **NOTE:** Only tracker administrators can remove a shared search.

1. Click **Trackers** from the **Project Home** menu.
2. On the list of project trackers, click the title of the tracker in which you want to delete a saved search.
3. In the tracker artifact list view, click **Search Tracker**.
4. Expand *My Saved Tracker Searches* or *Shared Tracker Searches* so that the available searches are displayed.
5. Select a search and click **Delete**.
6. Accept the confirmation message.

The search is removed from the table where you deleted it.

You can connect tracker artifacts with other CollabNet TeamForge items such as documents or topics. Creating associations between items enables you to define relationships, track dependencies, and enforce work flow rules.

Associate a Tracker Artifact with a Document, Task, Integrated Application Object, or Forum

When a tracker artifact is related to other TeamForge items, such as tasks, documents, integrated application objects, or discussions, you can connect the tracker artifact to the other item by creating an association.

Creating associations between items helps you to define relationships, track dependencies, and enforce work flow rules.

1. Click **Trackers** from the **Project Home** menu.
2. On the list of project trackers, find the tracker artifact with which you want to create an association. Use the filter if needed.
3. Click the artifact title.
4. On the **View Artifact** page, click the *Associations* tab. The list of existing associations appears.
5. Click **Add**.
6. In the **Add Association Wizard** window, select the items with which you want to associate the artifact:
    - **ENTER ITEM ID** - If you know the item's ID, you can enter it directly.

      > **NOTE:** To associate an object in an integrated application from within TeamForge, use the `[<prefix_objectid>]` format. Successful associations appear hyperlinked. Each integrated application displays its prefix on moving the mouse over the application name in the tool bar.

    - **ADD FROM RECENTLY VIEWED** - Select one of the last ten items you looked at during this session.
    - **ADD FROM RECENTLY EDITED** - Select one of the last ten items you changed.
7. Click **Next**.
8. You may add a comment in the **ASSOCIATION COMMENT** text box.
9. Save your work.
    - Click **Finish and Add Another** to add additional associations.
    - Click **Finish** to return to **Details** page.
10. Click the *Associations* tab to view a graphical representation of all the associated items. Through the Association Viewer, you can choose to view associations in the form of a list or flip over to the Trace view to explore the layers of associations (including parent/child dependencies) laid out in a timeline. You can scroll across the Trace view by dragging the mouse over the association layer or use the 'Previous' and 'Next' arrows to view all the objects as events in a timeline.

While the *Associations* tab shows the count of the total number of associations, you can only view the most recent 500 associations when you click the *Associations* tab in case the artifact has more than 500 associations. You can, however, browse through the Association Viewer to view older associations.

You can click on each node on the graphical association viewer to filter and display the associated items in the table below the association viewer thus matching the number of associations provided on the selected node. For example, if the node that you select for filtering is having two associations on it, the table displays the two associated items as a result of the filtering process.

Associate a Tracker Artifact with a File Release

To track the source and resolution of a bug or a feature request, associate its tracker artifact with the file release in which it was discovered and fixed.

Tracker artifacts associated with file releases are also displayed separately, providing a simply way to track all issues that were discovered in or fixed in a specific file release.

> **TIP:** You can also add associations from the tracker artifact's **View Artifact** page.

1. On the **Project Administration** page, enable both the **REPORTED IN RELEASE** and **FIXED IN RELEASE** fields.

   - **REPORTED IN RELEASE** - When submitting a new artifact, the user can choose from a drop-down list of all releases in the project to identify the release in which the issue was discovered.
   - **FIXED IN RELEASE** - After the issue is fixed, the user can choose from a drop-down list of all releases in the project to identify the release in which the issue was fixed.

   > **TIP:** If either of these fields says `Unknown`, the artifact you are working on may be associated with a file release that you don't have permission to view. You can leave that as it is or change it to a file release that you do have permission to view.

2. When submitting a new artifact, choose the release in which the issue was discovered from the **REPORTED IN RELEASE** drop-down list. The drop-down list shows all releases in the project, except those that are in **pending** status.
3. After the issue is fixed, update the artifact by choosing the release in which the issue was resolved from the **FIXED IN RELEASE** drop-down list. The drop-down list shows all releases in the project, except those that are in **pending** status.

The associated tracker artifacts are displayed on the **View Release** page, in the **REPORTED TRACKER ARTIFACTS** and **FIXED TRACKER ARTIFACTS** sections.

Associate a Tracker Artifact with a Code Commit

When checking in files to your SCM repository, you can create links to one or more tracker artifacts or other CollabNet TeamForge items.

Associations track the links between code and the bugs, feature requests, or other tracker artifacts that the code addresses. You can also associate code commits with other TeamForge items, such as tasks or documents.

A project administrator can make associations mandatory for all code commits. When this is made mandatory, the following additional rules pertaining to code commit can also be set:

- Code commits can be performed only for open artifacts.
- To perform a code commit, the committer must be the owner of the specific artifact.

> **NOTE:** Once you enforce the above rules, validations are strictly enforced for commits against tracker artifacts only. In case you commit against any other TeamForge object, for example a wiki or a document, mere existence of the object ID ensures successful commit and association and no validations are performed against the status of the object or who it is assigned to.

You can create tracker artifact associations from whatever interface you normally use to check code into your SCM repository. You do not have to log into TeamForge.

Use the same syntax for commits to CVS and Subversion repositories.

When making a code commit, add the associate command in the commit message like this: `[<item id>]`, such as the TeamForge tracker artifact ID or task ID.

- TeamForge item IDs are always letters followed by four or more numbers, such as `task1029` or `artf10011`.
- To associate a commit with multiple TeamForge items, separate the item IDs with commas.
- All associations are displayed in the *Associations* tab of the **Commit Details** page.
- The **Comment** section lists the comments made with each commit.

> **NOTE:** To associate an object in an integrated application, use the `[<prefix_objectid>]` format. Each integrated application displays its prefix or moving the mouse over the application name.

> **NOTE:** When an association is added to or removed from TeamForge objects such as tracker artifacts, tasks, documents, discussions, and file releases, a notification mail is sent to users monitoring these objects. A option is provided at site level and user level to make sure whether the notification mail has to be sent or not. For more information on this, see Configure your Site’s Settings.

NOTE: To remind yourself of the details of the association later, look in the *Change Log* tab of the associated **View Artifact** page.

To organize your work with tracker artifacts, you can make an artifact a child or a parent of another artifact.

Make a Tracker Artifact depend on another Artifact

- A child artifact can have only one parent.
- A parent artifact can have any number of children.
- A parent artifact cannot be closed if a child is open.

NOTE: `Open` or `Closed` in this context, refers to a type of status, not the status itself. A tracker administrator can specify a group of statuses such as `Deferred`, `Fixed`, `Rejected`, as equivalent to `Closed`, while `In progress` and `Under consideration` might be specified as `Open` statuses. For example, when you look at the artifact summaries at the top of any tracker list page, you are seeing a summary of the status type, not the status, of the artifacts.

1. On the artifact page, click the *DEPENDENCIES* tab.
2. Choose or create the related artifact.
   - If the parent artifact already exists, click **Choose Child** or **Choose Parent**. In the pop-up, type in the artifact ID or choose from the list of your recently edited artifacts.
   - If the related artifact does not exist yet, click **Create Child in** or **Cread Parent in** and select the tracker that the new related artifact will belong to. Fill in the form the same way as you would for submitting an unrelated artifact.

   NOTE: If **Choose Parent** or **Create New Parent** is not visible, the artifact already has a parent artifact. An artifact can only have one parent artifact.
3. Click **Next**.
4. Write a comment that describes the relationship, if appropriate, and click **Finish**.

   NOTE: When a dependency is added to or removed from a tracker artifact, a notification mail is sent to users monitoring the artifact. An option is provided at site level and user level to make sure whether the notification mail has to be sent or not. For more information on this, see Configure your Site's Settings.

The parent-child relationship between the artifacts is established.

- To cancel a parent dependency, click **Remove**.
- To cancel a child dependency, select the child artifact and click **Remove**.

> **NOTE:** Task dependencies and tracker artifact dependencies are different from each other.

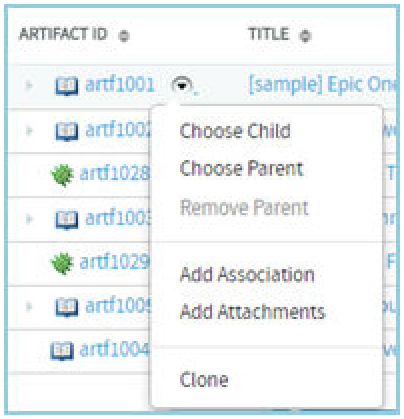
✓ For tracker artifacts, an artifact with dependencies (a “parent” artifact) can’t be considered closed unless all of its dependent artifacts (“children”) are closed.
✓ For tasks, a dependency means one task can’t start until another task is completed.

Context Menu to create Dependencies, Associations or Add Attachments

Use the context menu available in planning folder list view to quickly choose a “parent” or “child”, remove a “parent”, add associations, add attachments or clone artifacts.

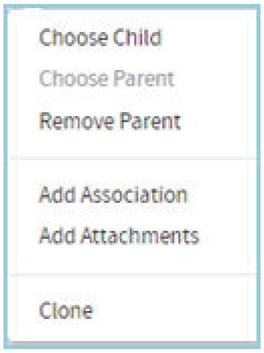
In the planning folder list view, you can see a down arrow icon next to the artifact ID (as shown in the following screen)when you move your mouse pointer over artifact rows.

Down arrow icon

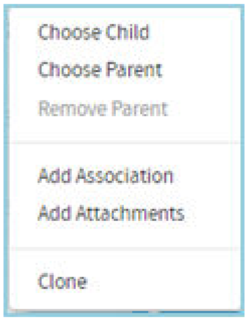


A context-sensitive menu pops up on clicking the down arrow icon.

Context menu if an artifact has a parent



Context menu if an artifact has no parent



To quickly choose a parent or child, add associations and attachments, or remove a parent:

1. Select a planning folder on the **List Trackers**, **Planning Folder** and **Teams** page. The planning folder **List Artifacts** page appears.
2. Move your mouse over an artifact's ID. A down arrow icon appears.
3. Click the down arrow icon to see the context menu. Depending on the context, you can do one of the following tasks.
   - Choose a Child
   - Choose a Parent
   - Remove a Parent
   - Add Associations
   - Add Attachments
   - Clone an Artifact

Choose a Child

1. Click **Choose Child** from the context menu. The **Selection Children…** window appears.

2. Select the **Enter Artifact ID** option and type an artifact ID. This artifact becomes the child.

> **TIP:** You can select the **Add from Recently Edited** option and choose a child from the list of recently edited artifacts.

3. Click **Next**.
4. You mad add a comment in the **DEPENDENCY COMMENT** text box.
5. Click **Finish**.

> **NOTE:** You may also click **Finish and Add Another** to continue adding more children for the same artifact.

Choose a Parent

1. Click **Choose Parent** from the context menu. The **Selecting Parent…** window appears.
2. Select the **Enter Artifact ID** option and type an artifact ID. This artifact becomes the parent.

> **TIP:**

3. Click **Next**.
4. You may add a comment in the **DEPENDENCY COMMENT** text box.
5. Click **Finish**.

Remove a Parent

1. Click **Remove Parent** from the context menu. A confirmation message appears as: `Do you want to remove the dependency?`
2. Click **OK**.

Add Associations

1. Click **Add Association** from the context menu. The **Add Association Wizard** appears.
2. Select the **Enter Item ID** option and type an artifact ID for creating an association.

> **TIP:** You can select the **Add from Recently Edited** option and select an artifact from the list of recently edited artifacts.

3. Click **Next**.
4. You may add a comment in the **Association Comment** text box.
5. Click **Finish**.

> **NOTE:** You may also click **Finish and Add Another** to continue associating more artifacts.

Add Attachments

1. Click **Add Attachments** from the context menu. The **Add Attachments** window appears.
2. Type a comment for the attachments in the **COMMENT TEXT** box.
3. Click **Choose File**.
4. Browse and select the file you want to attach.
5. Click the **Attach another file** link to add more attachments. Repeat this step for adding more attachments.
6. Click **Add** to attach the selected files to the artifact.

Clone an Artifact

1. Click **Clone** from the context menu. The **Clone Artifact** window appears.
2. Provide a name and description for cloning the artifact.
3. Click **Clone** to clone the artifact.

You can now import artifacts into TeamForge using the Excel/CSV tracker import function. Data from both Excel and CSV files can be imported.

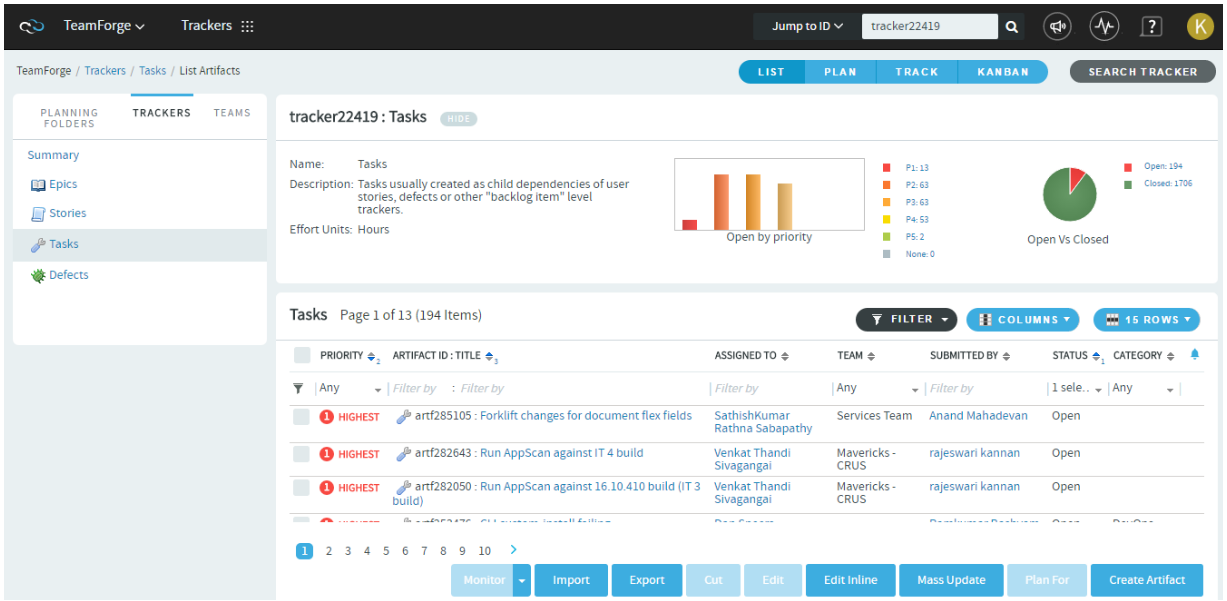
✓ You must have Site Administrator permission or Tracker Submit and Edit permission to import artifacts from Excel/CSV files.

✓ The Excel/CSV template can be downloaded from the Import Artifact window.

✓ If you want to create new artifacts, the following tracker fields are obsolutely required while importing data from Excel/CSV files: Artifact ID, Title, Priority, Status and Description. However, for updating an existing artifact, a valid “Artifact ID” is all that’s required along with the data for other fields, which you may want to update.

✓ The tracker import function supports upto 500 rows of data in Excel/CSV files. However, you can configure the number of artifacts that can be mass-imported. For more information, see Configure Your Site’s Settings.

1. Click **Trackers** from the **Project Home** menu.
2. Select a tracker from the *TRACKERS* tab.
3. Click **Import**.



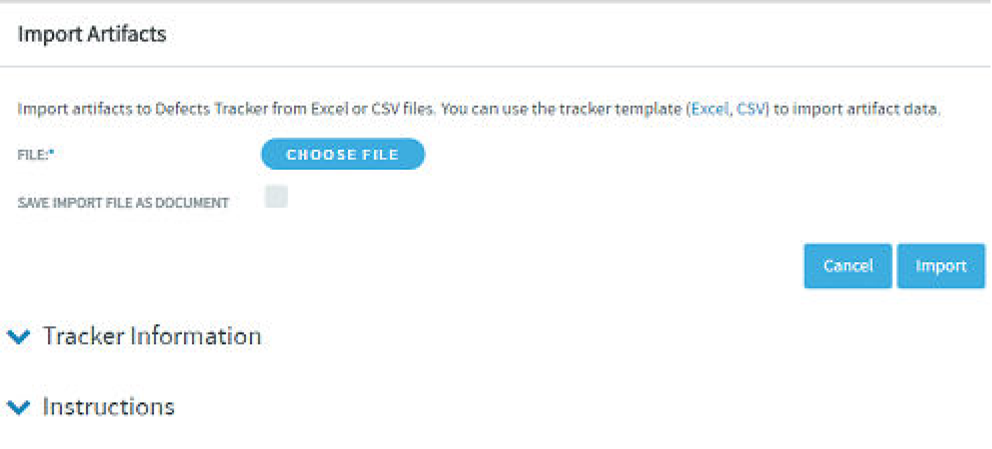
tracker22419 : Tasks

Name: Tasks
Description: Tasks usually created as child dependencies of user stories, defects or other "backlog item" level trackers.
Effort Units: Hours

Tasks Page 1 of 13 (194 Items)

PRIORITY	ARTIFACT ID : TITLE	ASSIGNED TO	TEAM	SUBMITTED BY	STATUS	CATEGORY
Any	Filter by : Filter by	Filter by	Any	Filter by	1 sele..	Any
HIGHEST	artf285105 : Forklift changes for document flex fields	SathishKumar Rathna Sabapathy	Services Team	Anand Mahadevan	Open	
HIGHEST	artf282643 : Run AppScan against IT 4 build	Venkat Thandi Sivagangai	Mavericks - CRUS	rajeswari kannan	Open	
HIGHEST	artf282050 : Run AppScan against 16.10.410 build (IT 3 build)	Venkat Thandi Sivagangai	Mavericks - CRUS	rajeswari kannan	Open	

4. Click **CHOOSE FILE**. Browse and select the Excel/CSV file that contains the artifacts to be imported.



Import Artifacts

Import artifacts to Defects Tracker from Excel or CSV files. You can use the tracker template (Excel, CSV) to import artifact data.

FILE:*

SAVE IMPORT FILE AS DOCUMENT □

Tracker Information

Instructions

TIP: You can select the **SAVE IMPORT FILE AS DOCUMENT** check box if you want to have the imported Excel/CSV file stored in **DOCUMENTS** for future use.

5. Click **Import**.

Limitations

It's worth considering the following points while importing artifacts from Excel/CSV files:

- The Excel/CSV tracker import function allows importing data for any existing tracker fields even if the fields are disabled in the tracker.

- The **Import** button is not being localized on Chinese and Korean locales.
- The import instructions on the **Import Artifacts** page is not being localized on Chinese, Japanese and Korean locales.
- The Estimated Effort, Actual Effort, Remaining Effort, and Points fields can take “0” or any positive integer as values.
- Leave the “Comment text” field empty in the Excel/CSV files if you intent to create new artifacts.
- While it is expected that the “Priority” for an artifact can range from 0 to 5, the Excel/CSV import function allows any value for the “Priority” field to be imported. Make sure your Excel/CSV input file consists of “Priority” values ranging from 0 and 5.

To use the contents of artifacts from a tracker or a planning folder in other applications, export them to a .csv, .xml, .xlsx or tab-delimited file.

Sometimes, it can be useful to look at the status of a project by sharing a spreadsheet using Microsoft Excel or Google Spreadsheets.

> **TIP:** You can also export tracker artifacts from a tracker report or a set of search results.

1. Click **Trackers** from the **Project Home** menu.
2. Click a tracker or planning folder. In a tracker:
   - You can export a subset of the artifacts in the list view by setting up a filter. See Tracker List Artifacts View.
   - You can select specific artifacts and export only those checked items. The current sort order in the list view is retained in the exported list of artifacts as well.
3. Click **Export**.
4. In the **Export Data** window, select an export format that you can import into the other application. For example, to use the data in a spreadsheet program, select .csv.
5. Move the fields you are interested in from the **AVAILABLE COLUMNS** list to the **SELECTED COLUMNS** list, then click **Export**.

   > **NOTE:** When you export tracker artifacts directly from a tracker, you can choose from all the fields in the tracker. When you export from a planning folder, only the standard fields provided by TeamForge can be exported. For more information on what you can do with different kinds of fields, see What fields can I use in a tracker?

6. Artifacts are exported and the file is downloaded automatically.

   > **TIP:** Starting from TeamForge 16.10, while exporting tracker artifacts, artifacts are exported to one of the file formats such as Excel, CSV, Tab-delimited or XML (as selected by the user) and the file

> is downloaded automatically. To improve performance, the file download link, which you would have used in the past, is no longer available.

To plan how and when an artifact is to be addressed, assign it to a planning folder.

A planning folder can contain artifacts from many different trackers. When you assign an artifact to a planning folder, the artifact is still in the tracker where it was created.

1. Click **Trackers** from the **Project Home** menu.
2. Find the tracker that contains the artifact you want to assign, and select the artifact. You can select any number of artifacts at once.
3. Click **Plan for** and select a planning folder.

The planning folder you assigned is now shown in the **Planned For** column on the **Planning Folder Summary** page.

You can always reassign an artifact to another planning folder as conditions change. Any effort data you have provided is recalculated automatically.

> **NOTE:** When a tracker is disabled, artifacts from that tracker do not contribute to the effort totals calculated for any planning folder they are in.

> **TIP:** When you are working with a single artifact, you might prefer another way to assign it to a planning folder which is to open the artifact you want to assign and look for the **Planning Folder** field. Select a planning folder for the artifact and save the artifact.

A planning folder is a way to organize work into feasible chunks and monitor its progress. As a project admin or as a user with the appropriate permissions, create and populate all the planning folders you need to capture the work you are planning.

When you've thought through your plan, express it in one or more planning folders.

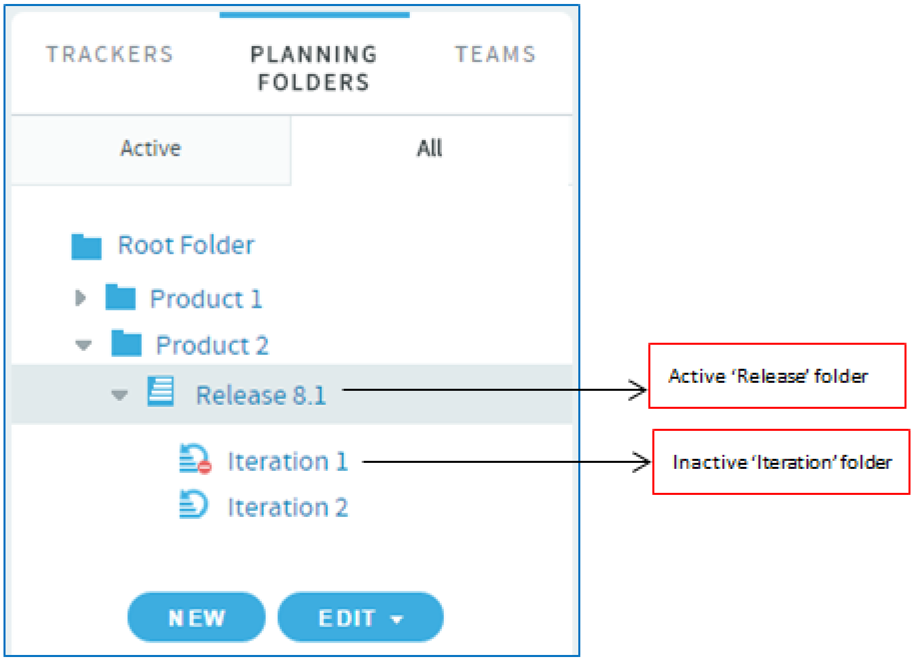
> **TIP:** It often makes sense to set up planning folders *after* you have outlined and analyzed the features you plan to deliver. See Define the Scope of Your Project.

For your agile projects, you have the option to create planning folders specific to iterations and releases, or you can create generic planning folders which you may customize later. When you've thought about the general categories the work falls into, you are ready to create planning folders that reflect those ideas.

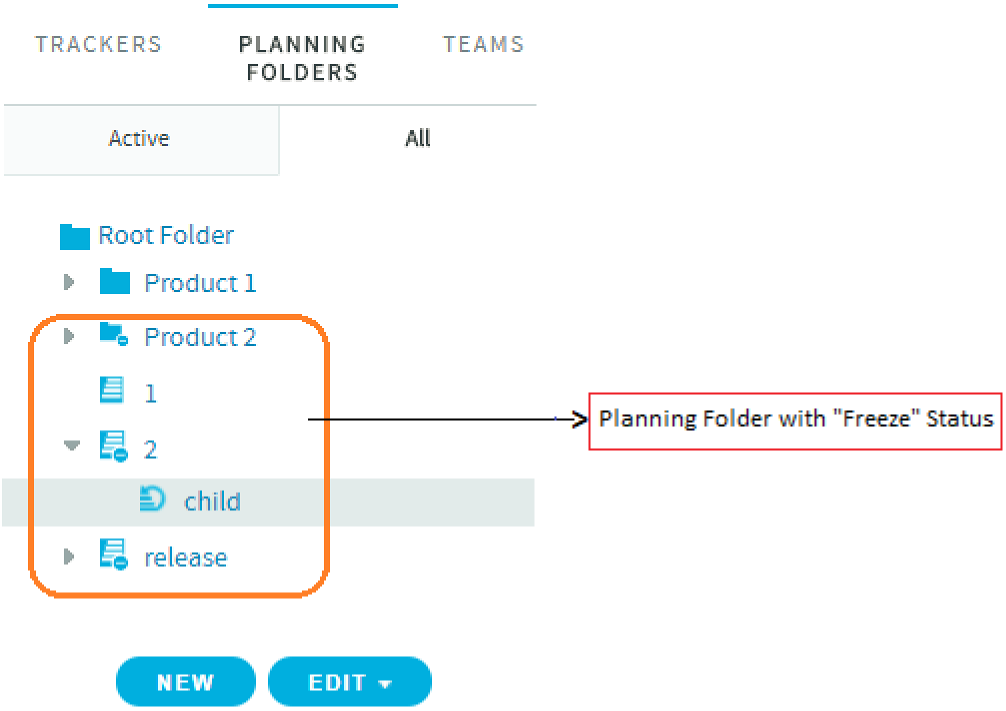
A planning folder can represent:

- A set of tasks, such as Iteration 3, or "Initial infrastruture development".
- A period of time, such as "April", or "Q2-2010".
- A phase of development, such as "Testing" or "Deployment".
- A component of the product, such as "Chapter 12" or "Rear stabilizer".

Navigate to **Project Home > Trackers** page and click the *All* tab under the *Planning Folders* section to see the statuses of the planning folders.



Planning folder with "Freeze" status (introduced in TeamForge 18.1):



When you’ve set up your planning folders, you have four views available to work with them:

- **LIST**: The list view.
- **PLAN**: The planning board view.
- **TASK**: The task board view.
- **KANBAN**: The kanban board view.

Related Links

- What is a planning folder?
- Manage Statuses for a Planning Folder

Ranking in Planning Folders

The ranking logic has been revamped in TeamForge 17.11 to improve the overall Planning Folder performance. This new ranking logic is applied to artifacts inside planning folders while you upgrade to TeamForge 17.11, due to which data migration can take longer than usual while upgrading to TeamForge 17.11. The time taken for data migration is proportional to the number of artifacts available inside individual planning folders. You can use the following queries to find out the total number of artifacts and the the total number of artifacts that are inside planning folders.

- The following query can get you the total number of artifacts:

```
select count(id) as total_artifacts from artifact;
```

- The following query can get you the total number of artifacts that are inside planning folders:

```
select count(a.id) as artifacts_with_pf from artifact a, item i where a.id = i
.id and i.planning_folder_id like 'plan%' and i.is_deleted <> '1';
```

CollabNet's Performance Lab test results show that it takes 11 minutes approximately to migrate (apply the new ranking logic) a site that with 245K artifacts in total, of which around 46K artifacts were placed inside planning folders.

> **NOTE:** With this new light weight ranking function, the hidden URL, which was used by project administrators to reset ranks for a given project, is no longer supported.

As a project admin or as a user with the appropriate permissions, create and populate all the planning folders you need to capture the work you are planning.

1. Click **Trackers** from the **Project Home** menu.
2. Under **PLANNING FOLDERS**, click the folder in which you want to create a new planning folder. Click **NEW**. An appropriate planning folder creation page is displayed.
3. Select the type of planning folder you want to create: **Release**, **Iteration** or **Folder**.

   While "Release" and "Iteration" are specific planning folder types, "Folder" is your standard, generic planning folder. Though having multiple planning folder types help you logically organize your planning folders, there's no hierarchical restrictions on these different planning folder types. For example, as a user, you have the liberty to create a "Release" planning folder as a child of an "Iteration" planning folder, which may not be the ideal use case in an Agile project.

4. Enter a brief and descriptive name for your planning folder. For example:
   - In an agile project, depending on your requirement, you can create two or more iteration folders called "Iteration 1", "Iteration 2" and son on within the Release planning folder.
   - In a phased, waterfall-style project, you might name your first planning folder "Design", the next "Build", and so on.

   > **TIP:** Don't worry if you don't have anything of interest to put into your planning folders yet. The parallel process of filling out the feature tree will provide plenty of material for this.

> **NOTE:** This field is mandatory.

5. Use the **DESCRIPTION** to briefly signal the kind of work that will be contained in this planning folder. Include enough information to help people get up to speed quickly when they join your project. But save most of the detail for the individual tracker artifacts, where project members will spend most of their time.
6. For **STATUS**, (available since TeamForge 5.4), select a value that communicates where the planning folder is in its life cycle. For a new planning folder, you'll probably want to select `Not started`.

> **TIP:** You can create more values to choose from in your Project Settings.

7. If you have set up a file release in the File Releases tool to deliver the work you are tracking here, you can identify it in the **FILE RELEASE** field (available since TeamForge 5.4). Any artifact you add to this planning folder will also appear in the Planned Tracker Artifacts tab when you look at that file release.

> **NOTE:** If you are creating an Iteration folder, this field displays the default file release value as its parent folder level (**Release** folder) which you can modify.

8. For **CAPACITY**, (available since TeamForge 5.4), provide a number that expresses how much work your team thinks it can do in the period represented by this planning folder. This field is available only for "Iteration" and "Folder" types.

> **TIP:** When you create a new planning folder, it can be a good idea to set its capacity to zero until you get a feel for the artifacts and resources you are working with.

9. For **POINTS CAPACITY**, (available since TeamForge 7.1), provide a number that represents the amount of work (expressed as number of story points) that you think can be handled in that specific planning folder. This field is available only for "Release" and "Folder" types.

> **NOTE:** To assist project managers gauge the planning folder's points capacity, when a release or standard planning folder is selected in one of the planning board swim lanes, this information is displayed as a ration of planned number of story points versus the actual number of story points. This informaiton does not apply to Iteration folders and therefore does not appear on the planning board swimlanes when you select an Iteration planning folder. For more information, see Use the Planning Board.

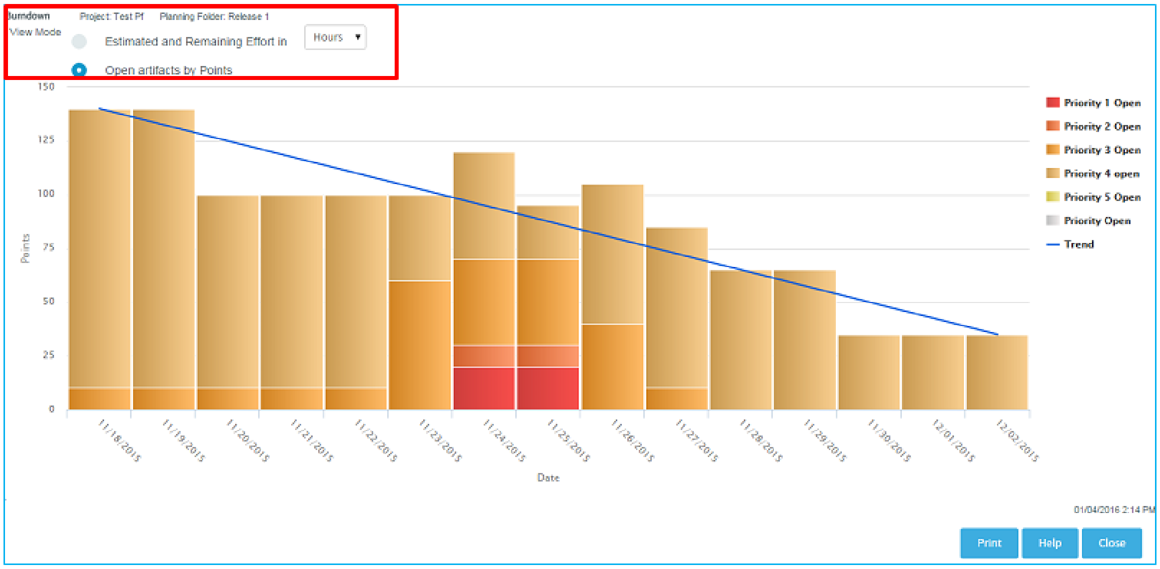
10. Select or enter a start and end date for the work that this planning folder will contain. You can change these dates at any time. However, it is important to specify dates even if they are not firm, because useful visual aids and reports depend on them.

> **NOTE:** The start and end dates of the child planning folder should be within that of its parent folder. These dates are mandatory for an “Iteration” planning folder because the burndown chart for Iteration planning folders is enabled only when you specify the start and end dates of an Iteration planning folder. Also, these dates are not mandatory for a Release planning folder because the burndown chart of a Release planning folder takes into account the start and end dates of its child planning folders (Iterations) and not its own

11. Select the unit that is suitable for the planning folder from the **DISPLAY EFFORT IN** field. Example: If the planning folder represents a sprint, then select the unit as **Hours** and if it represents a collection of sprints, then select the unit as **Points**.

> **NOTE:** Units are configured at the project level and not at the planning folder level.

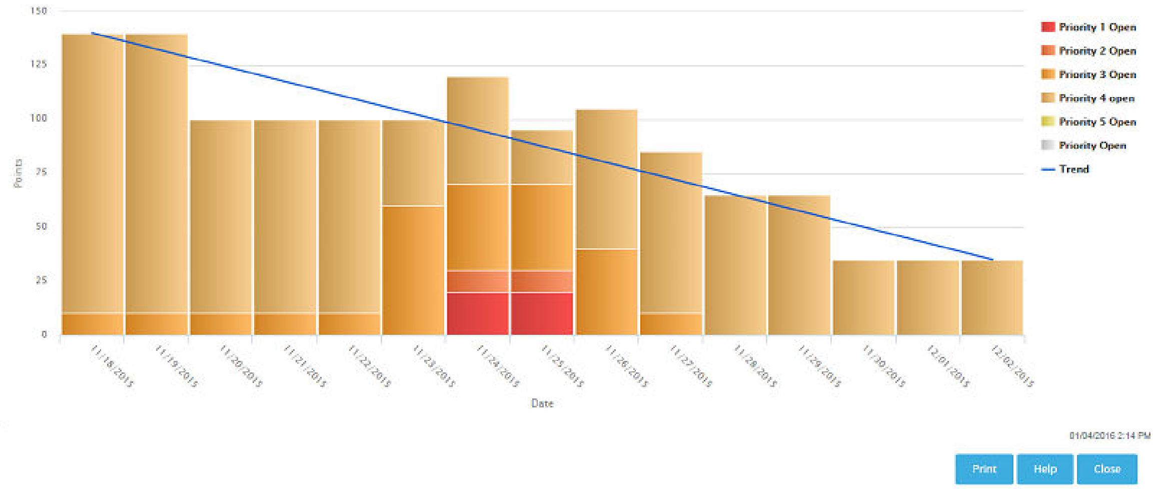
12. Select either **Effort** or **Points** from the **BURNDOWN VIEW BY** drop-down list to set the View Mode in the burndown charts (in the planning folder *List Artifacts* page). This field is available only for “Folder” type. Depending on the planning folder type, the burndown chart is displayed.
    - Burndown chart for the standard planning folder type which includes the View Mode:



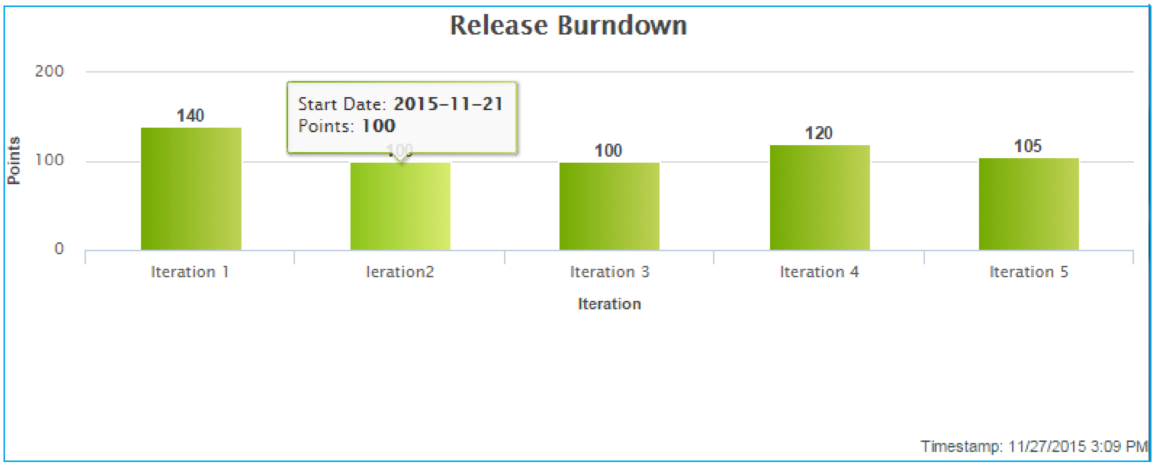
Burndown Project: Test Pf Planning Folder: Release 1

View Mode
○ Estimated and Remaining Effort in Hours ▾
● Open artifacts by Points

    - Burndown chart for “Iteration” planning folder type:



- Iteration-wise burndown chart for a “Release” planning folder type:



13. Click **Create**. The planning folder is created.

A planning folder is a dynamic representation of a changing situation. From time to time, you'll want to update its name, description and other parameters to reflect changes in the underlying work.

1. Click **TRACKERS** from the **Project Home** menu.
2. In the planning folder tree, click the planning folder you want to update.
3. Use the **EDIT** menu to make your changes.

When the sequence of work in your project changes, it's a good idea to reorder the planning folders so that they accurately reflect the real order or work.

For example, it’s not unusual for product owners, who must constantly weigh the changing business value of features, to ask for one of user stories to be put aside so that another set can be delivered first. When this happens, a project manager may want to move the corresponding planning folders so that their relative positions visually confirm their relationship.

TIP: You may also want to set the planning folder's status to a value that communicates the change in the order of work.

By default, planning folders are reorderd according to when they were created. New planning folders are added to the end of the folder tree.

1. Click **TRACKERS** from the **Project Home** menu.
2. In the folder tree, click the folder containing the subfolders that you want to reorder.
3. Choose **Reorder Subfolders** from the **EDIT** menu.
4. Organize the subfolders:
    - To sort them alphabetically, click Alphabetize icon.
    - To reorder a specific folder, select it by clicking the title, then click move up or move down arrow until the folder is where you want it.

You can move artifacts in a planning folder up and down to reflect the order in which they should be addressed.

A set of user stories often includes dependency relationships that require that one user story be addressed before another. For example, suppose a backend database modification must be in place before a user interface component can be created to access it. Both user stories may have the same degree of priority, but one has to be done before the other.

One way to manage this is by ranking the user stories in your planning view so that they reflect the sequence you want. When you do this, team members have a better chance of selecting work efficiently and reducing the need for rework.

NOTE: Rank is not the same as priority. Two artifacts may have the same priority but different ranks, or sequency positions.

1. If you are not an administrator on this project, ask your project administrator or site administrator to assign you sequencing permission.
2. Click **TRACKERS** from the **Project Home** menu.
3. In the folder tree, open the planning folder containing the artifacts that you want to reorder.
4. Click **RANK**. You can now drag and drop the artifacts in your planning folder into whatever relative positions you want.

To go back to the sort order you had before (for example, sorted by priority), click **SORT**.

A planning folder's position may reflect scheduling changes in relation to the work contained in other planning folders.

For example, when plans change and the work represented in a later planning folder moves ahead of an earlier planning folder, move the folders around in the hierarchy to reflect the new relationship.

When you move a document folder, any documents and subfolders that it contains are also moved to the destination folder. Before promoting a planning folder to a higher level in your hierarchy of folder, make sure its member artifacts (and its member’s parent artifacts, if any) are all assigned to the same planning folder, or to a planning folder that belongs to the same folder hierarchy.

1. Click **TRACKERS** from the **Project Home** menu.
2. In the planning folder tree, find the foler that you want to move. Click the title to select it.
3. Choose **Cut** from the **EDIT** menu.
4. In the folder tree, find the folder into which you want to move the selected folder. Click the title of the folder to select it.

   > **NOTE:** You can move a folder into the root folder or into any other folder, as long as no member artifact ends up having a parent belonging to a higher-level folder in the same tree.

5. Choose **Paste** option from the **EDIT** menu. The folder is moved to the destination folder.

To use the contents of artifacts from a planning folder in other applications, export them to a CSV, XML, XSLX formats or tab-delimited file.

For example, sometimes it can be useful to look at the status of a project by sharing a spreadsheet using Microsoft Excel or Google Spreadsheets.

1. Click **TRACKERS** from the **Project Home** menu.
2. On the **List Trackers, Planning Folders and Teams** page, click **PLANNING FOLDER**.
3. On the **Summary** page, click the desired planning folder.
4. From the **List Artifacts** page, click **SORT** or **RANK** as required. You can also select specific set of artifacts to be exported.

> **NOTE:** When you export artifacts in the rank mode, the parent artifacts are listed along with the children as ranked in the **List Artifacts** page. Similarly, the sort order in the list view is retained in the exported list as well.

5. Click **Export**.
6. In the **Export Data** window, select an export format that you can import into the other application. For example, to use the data in a spreadsheet program, select `CSV`.
7. Move the fields you are interested in from the **AVAILABLE COLUMNS** list to the **SELECTED COLUMNS** list, then click **Export** to complete the process.

To help team members understand how to work with a planning folder, create statuses for it that correspond to the planning folder's life cycle.

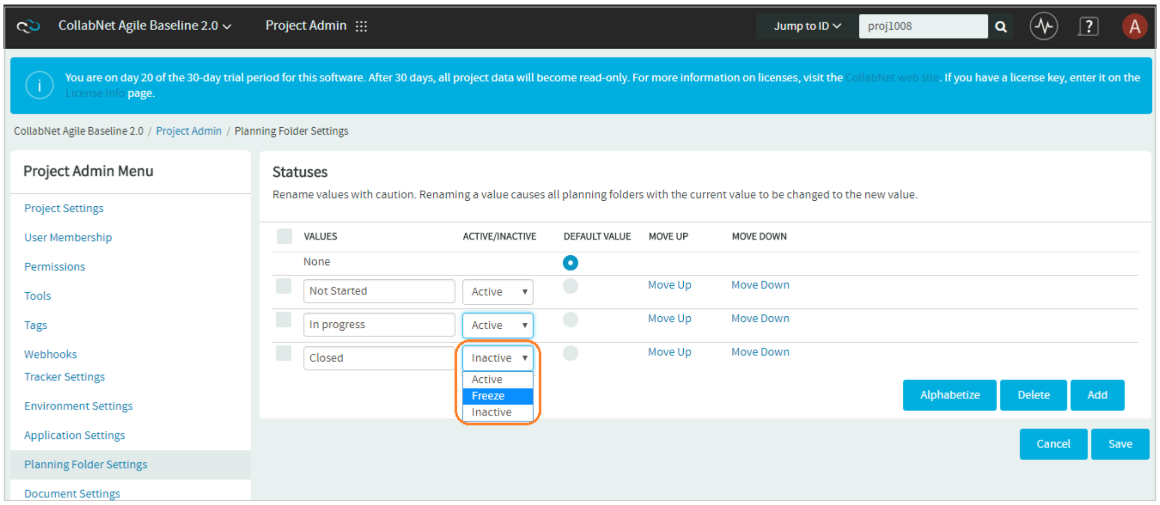
When you have created a planning folder status, you can apply it to any planning folder in your project.

For example, at a given moment you might have one planning folder in “Under development” status, while another is in “Finished” status and another is in “Preliminary scoping” status.

> **TIP:** You may also want to move the planning folder to a relative position in the planning folder list that communicates where it stands in the order of work.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Planning Folder Settings**.
3. Under **Statuses**, click **Add**.
4. Configure your new planning folder status.
    1. Give the status an unambiguous name that other will find easy to understand.
    2. Specify whether the status counts as active or inactive or freeze. Project members may want to exclude inactive planning folders from their view, to avoid overload.

        > **NOTE:** *Freeze* is the new status added to the **Active/Inactive** drop-down list in TeamForge 18.1. If you have selected the status of a planning folder to *Freeze*, you cannot be able to move any artifact into or out of this planning folder. However, you can update any artifact within this planning folder.



You are on day 20 of the 30-day trial period for this software. After 30 days, all project data will become read-only. For more information on licenses, visit the CollabNet web site. If you have a license key, enter it on the License Info page.

CollabNet Agile Baseline 2.0 / Project Admin / Planning Folder Settings

Statuses

Rename values with caution. Renaming a value causes all planning folders with the current value to be changed to the new value.

	VALUES	ACTIVE/INACTIVE	DEFAULT VALUE	MOVE UP	MOVE DOWN
	None		●		
□	Not Started	Active	○	Move Up	Move Down
□	In progress	Active	○	Move Up	Move Down
□	Closed	Inactive Active Freeze Inactive	○	Move Up	Move Down

3. Select which value is the default value for new planning folders and planning folders that originated in earlier TeamForge versions.

4. Arrange your planning folder statuses in the order that makes sense for you.

The Team feature, as the name implies, enables you to create logical groups of team members to carry out project activities more efficiently in an agile environment. Using the Team list view, you can create, edit, delete and view teams at the project level.

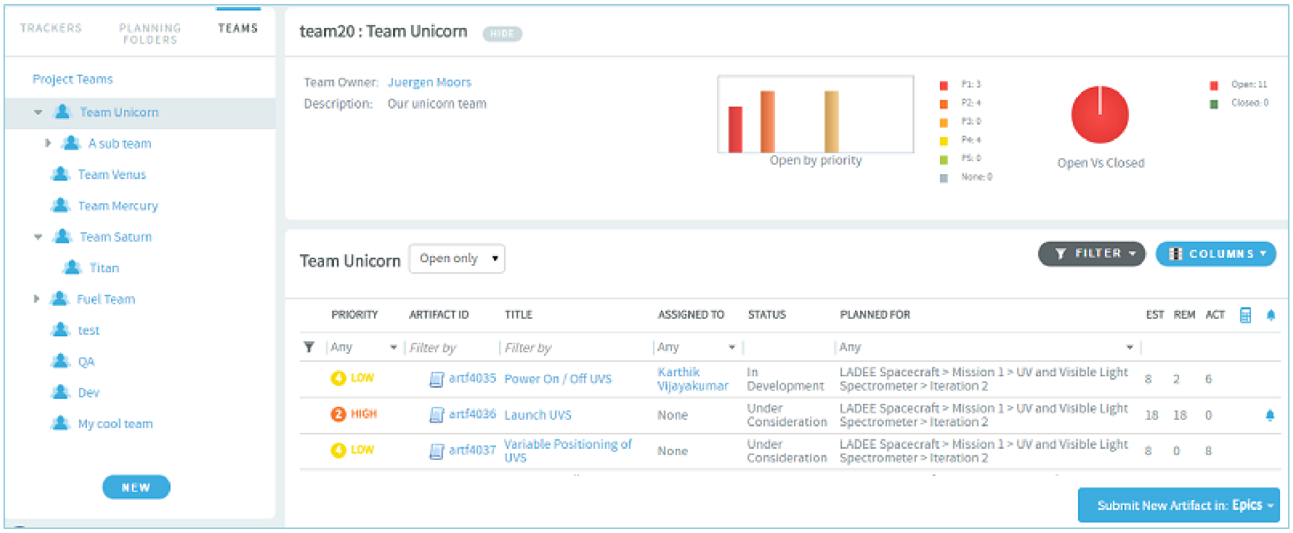
Teams - An Overview

Using Teams, project activities can be planned, tracked, collaborated, reported and executed in a more organized and structed manner.

This features facilitates effective communication among the team members who need to be aware of the changes and latest updates occurring in their projects and act accordingly. For example, using the team's view of backlogs, any project impediment can be communicated to the team and resolved quickly.

The Team list view consists of a tree structure of all the teams in a project, a summary and a filter table of the selected team. The filter table allows you to assign artifacts to a specific team. You can configure your team filter table and filter work items depending on the information you want. The filter table lists only the artifacts for the selected team. If you have selected a parent team, the filter table does not include the artifacts of the subteams (child teams).

> **IMPORTANT:** To access the Team feature, you must have the Tracker 'view' permission.



Team Owner: Juergen Moors
Description: Our unicorn team

Team Unicorn

PRIORITY	ARTIFACT ID	TITLE	ASSIGNED TO	STATUS	PLANNED FOR	EST	REM	ACT
Any	Filter by	Filter by	Any		Any			
4 LOW	artf4035	Power On / Off UVS	Karthik Vijayakumar	In Development	LADEE Spacecraft > Mission 1 > UV and Visible Light Spectrometer > Iteration 2	8	2	6
2 HIGH	artf4036	Launch UVS	None	Under Consideration	LADEE Spacecraft > Mission 1 > UV and Visible Light Spectrometer > Iteration 2	18	18	0
4 LOW	artf4037	Variable Positioning of UVS	None	Under Consideration	LADEE Spacecraft > Mission 1 > UV and Visible Light Spectrometer > Iteration 2	8	0	8

Team Roles and Permissions

Other than the Tracker 'view' permission, there are no role permissions that you need to set specifically for Team on the **Permissions** page of **Project Admin**. Depending upon your project role, you can view the appropriate icons on the Team list view and perform Team related tasks.

The following table lists the various roles and their specific permissions associated with Team:

- **Project Administrator** - You can create, edit, delete, and view teams.
- **Team Owner** - Any team member can be designated as a team owner. You can be a team owner for more than one team. As a team owner, you can only edit and view your team details.
- **Team Member** - Any project member can be added to a team. As a team member, you are only allowed to view the team members.

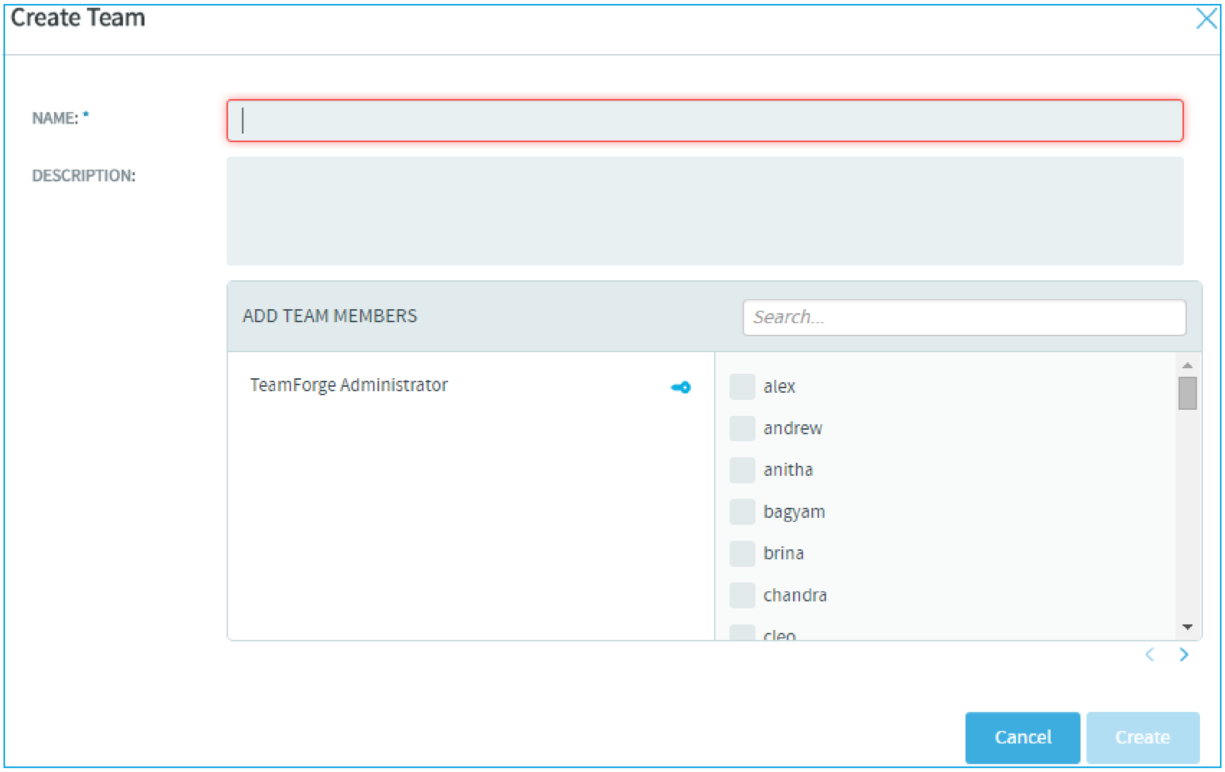
Permissions	**Roles**		
	Team member	**Team owner**	**Project admin**
Create a Team	✗	✗	✓
Delete a Team	✗	✗	✓
Add team members while creating a team	✗	✗	✓
Designate team owner while creating a team	✗	✗	✓
Edit your team details	✗	✓	✓
Add / delete team members while editing your team(s)	✗	✓	✓
Designate a team member as the team owner while editing your team	✗	✓	✓
View a team tree	✓	✓	✓
Assign artifacts to any team / team member	✓	✓	✓

Create a Parent Team

As a project administrator, you can create a parent team at the root level - **Project Teams**.

1. Click **TRACKERS** from the **Project Home** menu.
2. On the left pane of the default list view, click **TEAMS**. All the teams in your project are displayed on the pane.
3. Select **Project Teams** and click **NEW** to create a new team. Alternatively, you can hover the mouse on **Project Teams** and click the **Create Team** (⊕) button.

   The **Create Team** window is displayed.



4. Give a name and description to the new team

    - The team name is a required field. You are allowed to enter a maximum of 64 characters including spaces.
    - No two teams on the same level in the Team tree hierarchy can have the same name within a project; however, a team can have the same name as that of a deleted team.

5. Add team members by selecting the users from the list displayed. Alternatively, you can search for specific users by typing the relevant alphabets and selecting from the list of matching names.

    - The user who creates a team becomes the team owner automatically.
    - A team member can be a part of more than one team.

6. Designate a team member as the team owner by hovering the mouse on the specific name and clicking the **Team Owner** (🔑) button.

7. Click **Create**. A new team is created.

Create a Sub Team

A team can have many sub teams.

1. On the Team tree view, select the team for which you want to create a sub team and click **NEW** or hover your mouse on the specific team and click the **Create Sub Team** (⊕) button.

   The **Create Sub Team** window is displayed.

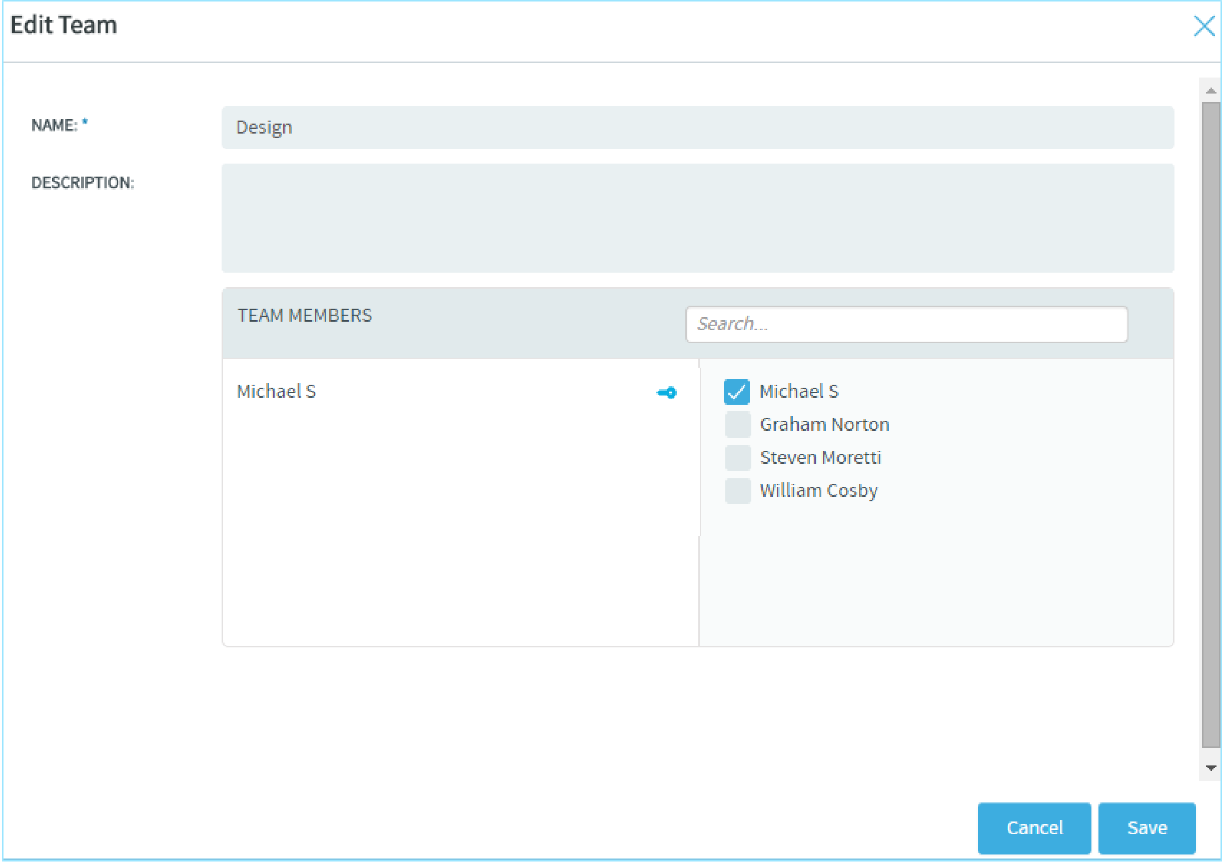
2. Provide all the required information, and click **Create**. A sub team is created.

Edit a Team

As a project administrator, you can edit any team in your project. Similarly, if you are a team owner, you can edit the team(s) for which you are the owner.

1. On the Team tree, hover your mouse on the team you want to edit and click the **Edit Team** (✎) button.

   The **Edit Team** window is displayed.



Edit Team

NAME: * Design

DESCRIPTION:

TEAM MEMBERS

Michael S

☑ Michael S
□ Graham Norton
□ Steven Moretti
□ William Cosby

2. Make the required changes:
   - Edit the team name and description.
   - To add more team members, select the names from the list displayed.
   - To delete team members, hover the mouse on the specific name and click the **Remove Member** (⛔) button.
   - To designate another team member as the team owner, hover the mouse on the specific name and click the **Team Owner** (🔑) button.

     > **NOTE:** As a team owner, you may decide to designate a different user as your team owner. In that case, once you update your changes, you will only be a team member and you will no longer have the permission to edit your team.

3. Click **Save**.

Delete a Team

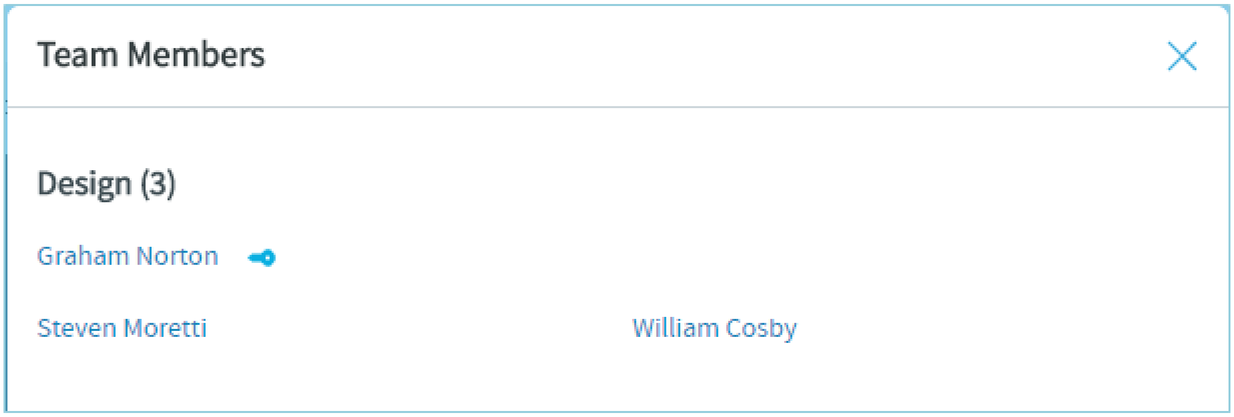
As a project administrator, you can delete any team from your project. If a team has sub teams (child teams), you can delete the parent team only after deleting all its child teams.

To delete a team, on the Team tree, hover your mouse on the team you want to delete and click the **Delete Team** (🗑) button.

View Team Members

On the Team tree, hover your mouse on the team whose members you want to view and click the **View Team Members** (👁) button.

The **Team Members** window is displayed.



Team Members

Design (3)

Graham Norton

Steven Moretti

William Cosby

The Planning Board is an important tool for your TeamForge project's agile planning activities. It enables you to plan and monitor the features that are required in each sprint (or iteration), and assign them from the product backlog to specific sprints.

When you've set up your planning folders, you have four views available to work with them:

- **LIST**: The list view.
- **PLAN**: The planning board view.
- **TASK**: The task board view.
- **KANBAN**: The kanban board view.

The planning board view complements the list view. While the latter offers you capabilities to accomplish various actions such as create, edit, and delete artifacts, planning folders and teams, the former offers product owners (or similar users) the ability to view, rank and move artifacts across the three planning folders (swimlanes) in a physical board-like user interface. In the Planning Board, planning folders are represented as swimlanes. In each swimlane, the tracker artifacts for the selected planning folders are represented as cards. You can also have a team's view of artifacts (backlogs and tasks), which is a swimlane representation of artifact cards for the selected team in the selected planning folder.

TeamForge user roles and permissions that are in place for planning folders apply to all the four views.

Use the Planning Board

You start populating the Planning Board by selecting a planning folder for each swin lane.

The drop-down list for each swim lane displays the hierarchy of active planning folders for the all the sprints and scrum teams that are under the project's root planning folder. By default, inactive planning folders are not shown in the swim lane drop-down list. You can organize the Planning Board the way you want by populating the swim lanes with planning folders of interest to you: for example, you may select the planning folder corresponding to the product backlog in the leftmost swim lane and various teams working on the release in the other two swim lanes, or a different sprint in each swim lange. Your selections are remembered for the current session.

In the Planning Board, artifact cards are displayed in ranked sequence within each swim lane. The tasks that you can accomplish while you work with the Planning Board include:

- Adding artifacts (quick add) in select planning folders with minimum required artifact information.
- Editing artifacts.
- Moving artifact cards within a swim lane and ranking them in select planning folders.
- Reassigning (move) artifacts from one planning folder to the other.

1. Click **TRACKERS** from the **Project Home** menu.

2. In the **List Trackers, Planning Folders and Teams** page, click **PLAN**. A Planning Board for the current project context is displayed.

3. For each swim lane in the Planning Board, select a planning folder from the drop-down list. The tracker artifacts contained by the planning folders are displayed as story cards within the corresponding swim lanes.

   - If a selected planning folder has subfolders, the artifacts within the subfolders are not displayed.

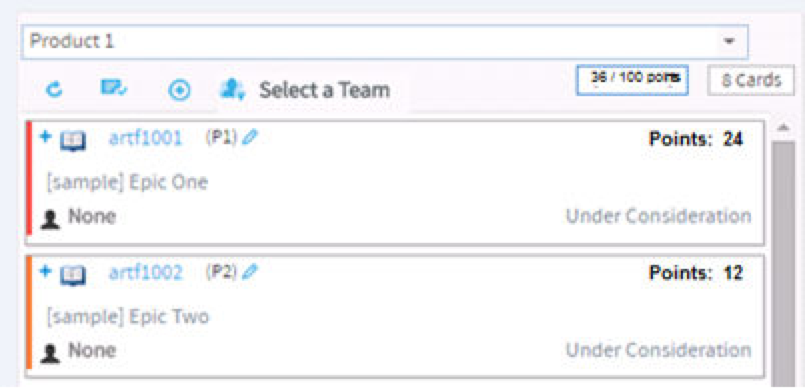
   - Cards have a color-coded bar to visually identify the priority. In addition, the open and closed card's background is color-coded to uniquely identify the status.

   - For each card, the artifact ID, title, priority, status, assigned to and points (story points) are displayed. For an artifact, the estimated effort (in hours) in shown only if its points=0. Relevant tooltips appear when you hover your mouse over these data elements.

   Assistive information such as the number of cards in the selected planning folder and planning folder points capacity are displayed at the top of the swim lane.

   > **NOTE:** The Planning folder points capacity is not displayed for `Iteration` folder types.

   For example, in the following screenshot, the planning folder points capacity is shown as `36/100 Pts`. Which means, you planned for 100 points, whereas the current points capacity (the sum of all open and closed artifacts) is 36 points. In other words, you have room for taking up more work in that particular planning folder.



Product 1

Select a Team

36 / 100 points

8 Cards

artf1001 (P1) Points: 24

[sample] Epic One

None Under Consideration

artf1002 (P2) Points: 12

[sample] Epic Two

None Under Consideration

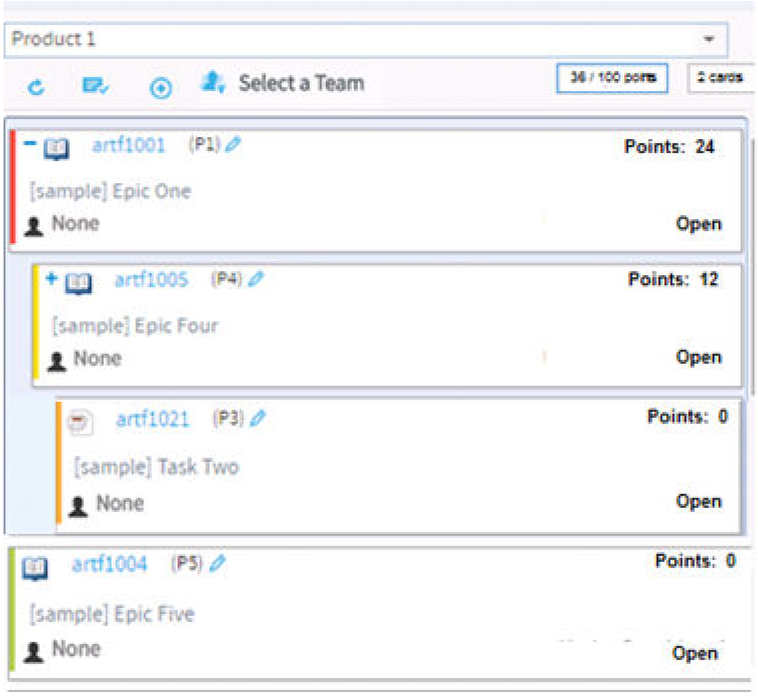
   You can also see 2 cards at the top of the swim lane, which is the count of the number of cards in the selected planning folder (by default, only open cards are counted).

Click the **Show/hide closed artifacts** (✓) toggle button to see the count of both open and closed artifacts.

> **NOTE:** You must have set the `Points Capacity` while creating planning folders to be able to see the current points capacity against the planned points capacity. For more information, see Create a Planning Folder

.

4. Click the swin lane refresh (↻) button to refresh individual swim lanes.
5. To know more about an artifact, click the artifact ID link. The **View Artifact** page is displayed.
6. By default, closed artifacts are not shown in swim lanes. Click the Show/hide (✓) toggle button to show or hide closed artifacts in swim lanes.
7. To expand an artifact that has child artifacts, click the "**+**" symbol in the artifact card. The first, second and third-level child artifacts are displayed. When you expand an artifact, the parent and child artifacts are visually outline by means of colored boxes as shown in the following screenshot.



In the Planning Board, artifact cards are displayed in ranked sequence within each swim lane. Using Planning Board, you can quickly add new cards (quick add), edit cards, move cards within a swimlane to rank them and move cards between swim lanes to reassign them to other planning folders.

Add Artifacts Quickly

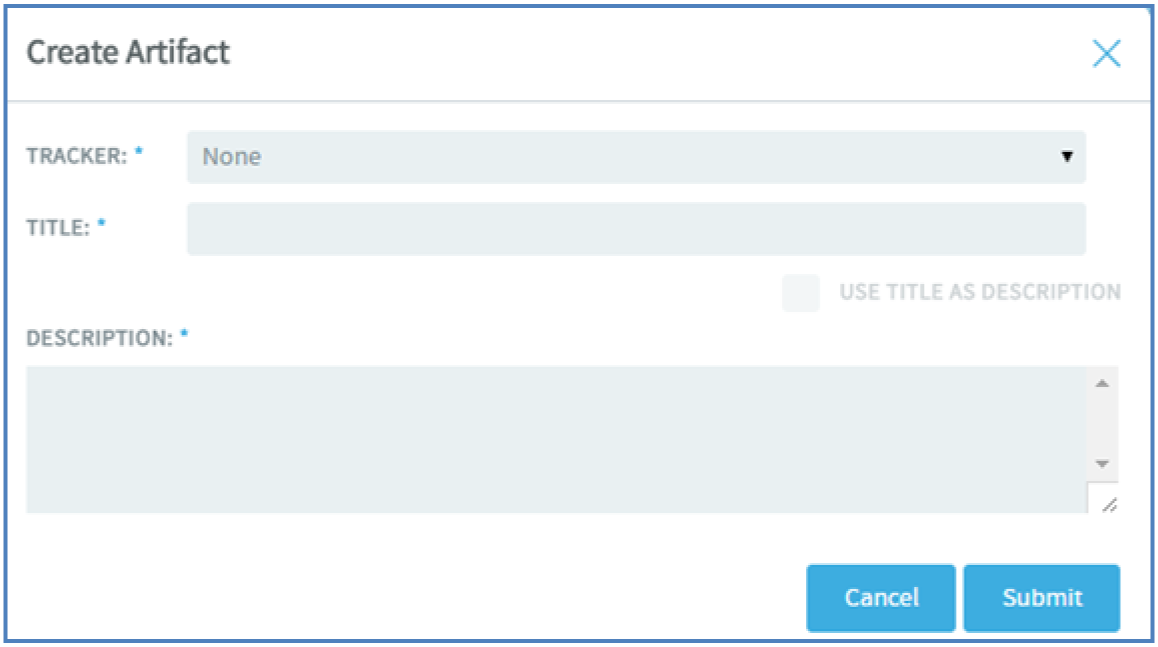
You can add artifacts (quick add) in select planning folders with minimum required artifact information.

> **IMPORTANT:** While you quickly add artifacts with data for just three fields such as the tracker type, title and description, the artifacts are, however, saved with default values for other required fields, which you may choose to update later. You cannot add artifacts from the Planning Board if a workflow is configured for the meta status change from `New` to `Open`.

1. Select a planning folder in one of the swim lanes. The quick add (⊕) button at the top of the swim lane is enabled.

   > **NOTE:** To be able to add artifacts in Planning Board, you need to have the required permissions.

2. Click the quick add (⊕) button. The **Create Artifact** window appears.



Create Artifact

TRACKER: * None

TITLE: *

□ USE TITLE AS DESCRIPTION

DESCRIPTION: *

3. Select a tracker type from the **TRACKER** drop-down list.

4. Type a title and description. You can select the **USE TITLE AS A DESCRIPTION** check box to copy the title you type to the **DESCRIPTION** text box.

   Artifacts support @mentions: Artifact description and comments now support @mentions and users called out via @mentions are added to the monitoring list. Include usernames with “@” as prefix (for example, @mphippard) to add users to the monitoring list.

   > **NOTE:** Users called out via @mentions must have `Artifact View` permission to be added to the monitoring list.

5. Click **Submit**. The artifact is added as the bottommost artifact in the swim lane (planning folder). You may choose to rank the artifact (if you have rank permission) or update the artifact with more meta data later. To update the artifact, click the artifact ID link.

Rank a Card

If you have the requisite permission, you can drag and drop a card about or below the other cards within a swim lane to position it in the order you want the artifact addressed.

> **NOTE:** The positioning of cards is retained when you switch between the list and board views (planning, task and kanbar). For related information on ranking artifacts in the list view, see Reorder the Contents of a Planning Folder.

Reassign (move) a Card

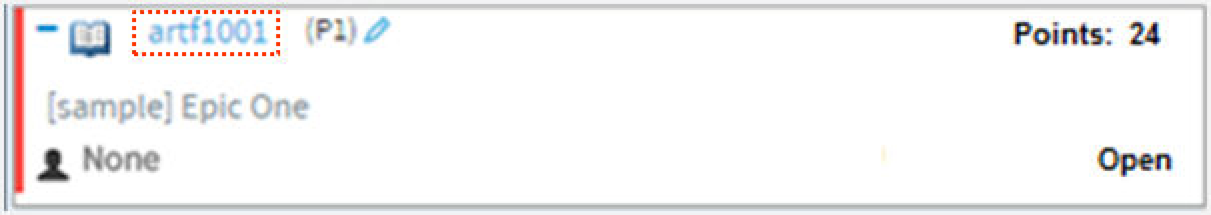
In the course of planning your release, you may want to assign an artifact from the product backlog to a specific sprint or team, or move an artifact from one sprint to another. To do this, just drag and drop the artifact card from one swim lane to the other.

> **NOTE:** When you drop the card, its rank is relative to its position in the new swim lane. However, if a child artifact is moved to a swim lane (planning folder) where its parent is already present, the child artifact is file under its parent and its rank is relative to the parent’s position.

To be able to move cards between swim lanes, you need to have the permission to update planning folders and rank artifact cards.

Edit an Artifact using the Artifact ID Link

1. Click the artifact ID link.



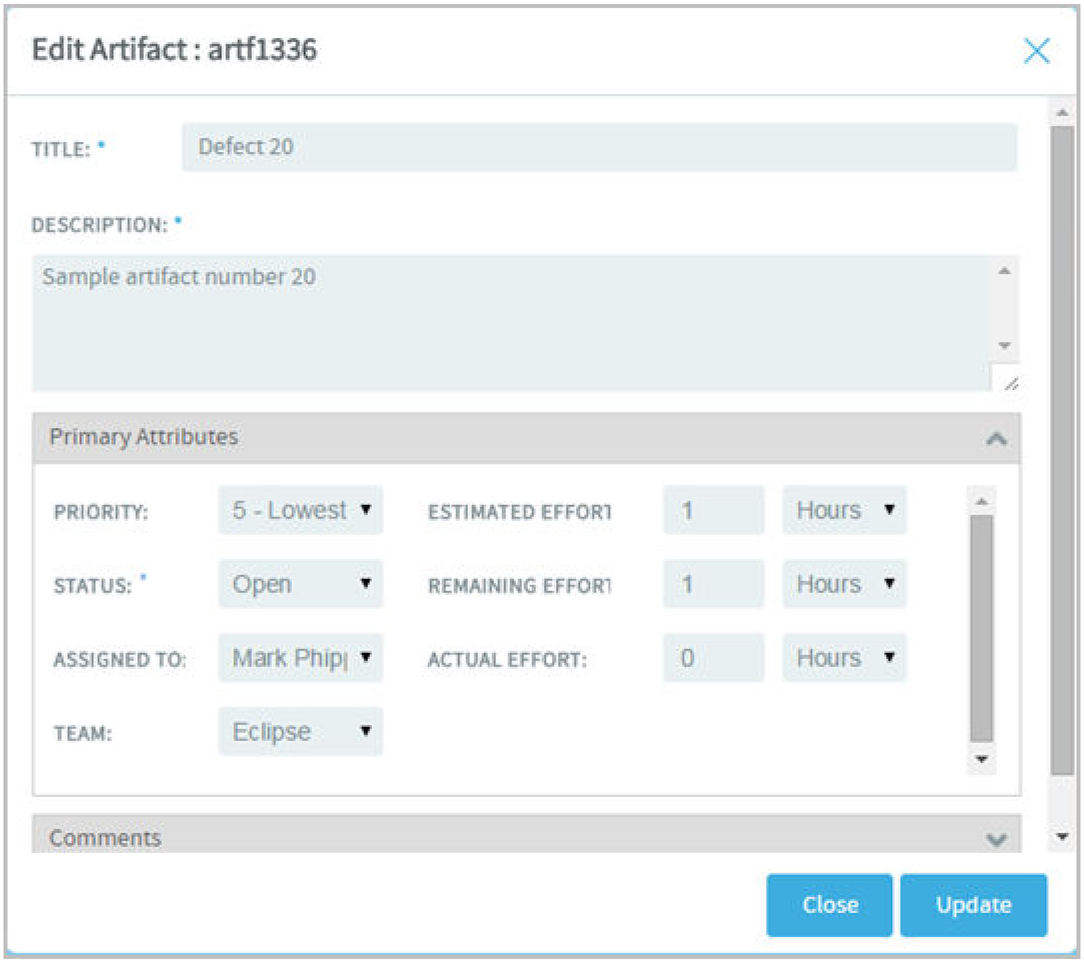
The **View Artifact** page appears.

2. Make the necessary changes and click **Update** to update the artifact and return to the Planning Board.

To quickly edit an artifact:

3. Click the edit artifact (✎) button in a card. The **Edit Artifact** window appears.

> **NOTE:** To be able to edit artifacts, you need to have the required permissions.



Edit Artifact : artf1336

TITLE: * Defect 20

DESCRIPTION: * Sample artifact number 20

Primary Attributes

PRIORITY: 5 - Lowest

STATUS: * Open

ASSIGNED TO: Mark Phip|

TEAM: Eclipse

ESTIMATED EFFORT 1 Hours

REMAINING EFFORT 1 Hours

ACTUAL EFFORT: 0 Hours

4. You can edit the following in the **Edit Artifact** window:

   - **Artifact title**
   - **Artifact description**
   - **Primary and secondary attributes**

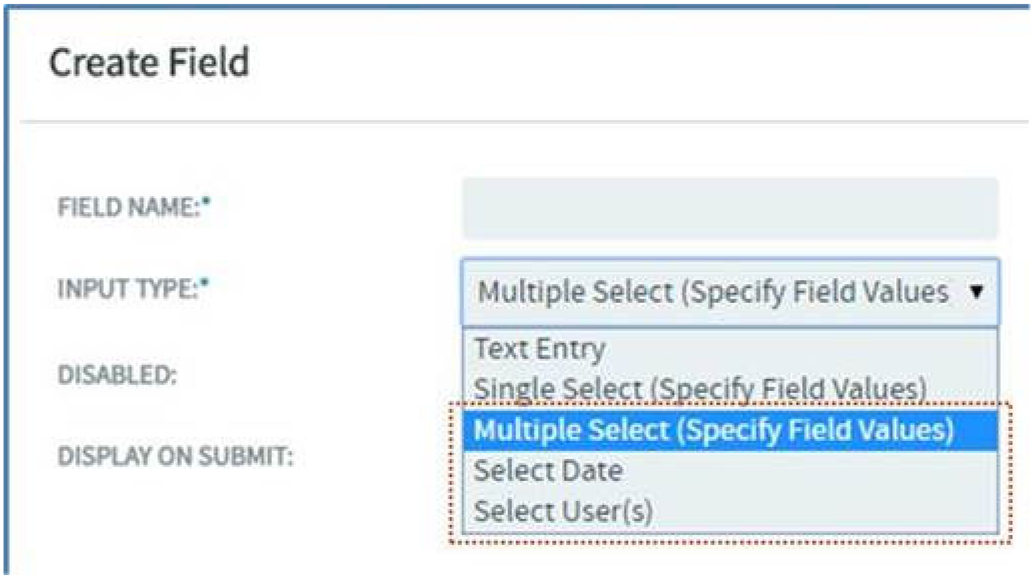
     You have two expandable frames, **Primary Attributes** and **Secondary Attributes**, that list fields you can edit. Primary attributes include Priority, Status, Assigned To, Team, Estimated Effort, Remaining Effort, Actual Effort and Points. The expandable frames and the fields listed in them depend on your application configuration that you set in Tracker Settings. For example:

     - Configurable fields which are enabled. For more information, see Enable or Disable Tracker Fields.

- User-defined fields (flex fields) with the field type Text Entry and Single Select and set as **Required** (Tracker Settings > Tracker Administration > Create Field) or workflow configured. For more information, see Create Custom Tracker Fields and Create a Tracker Workflow.

> **NOTE:** You cannot change the status of an artifact in the Planning Board view, if `Assigned To` is set as a mandatory field for status changes per your tracker workflow settings (Tracker Settings). However, you can edit the status of an artifact in the List view with the same `Assigned To` workflow constraint.

By default, the following field types which you select while creating a user-defined field (Tracker Settings), are non-editable from the planning board's **Edit Artifact** window: Multiple select, Date picker and User picker.


Create Field

FIELD NAME:*

INPUT TYPE:* Multiple Select (Specify Field Values

DISABLED:

DISPLAY ON SUBMIT:

Text Entry
Single Select (Specify Field Values)
Multiple Select (Specify Field Values)
Select Date
Select User(s)

- **Comments**

  In this expandable frame, in addition to adding comments, you can view the last five comments for an artifact.

Artifacts support @mentions: Artifact description and comments now support @mentions and users called out via @mentions are added to the monitoring list. Include usernames with “@” as prefix (for example, @mphippard) to add users to the monitoring list.

> **NOTE:** Users called out via @mentions must have `Artifact View` permission to be added to the monitoring list.

5. After editing an artifact, click **Update**.

Task board is an important tool in the Agile process. It helps the team to focus on the work at hand in the current sprint and feed progress data back into the system.
The Task board is a not a sprint-planning tool; it is more of a sprint-tracking tool. Use the Planning Board for planning the stories that will be dealt-with in the sprint. During the sprint, team members can use the Task Board to break down the stories into tasks and track them to completion.

For more information, see What are Planning, Task, and Kanban Boards?

TeamForge project administrators can configure one Task Board per project. Once that is done, project members can use the Task Board.

- During Task Board configuration:
  - Select one or more backlog trackers.

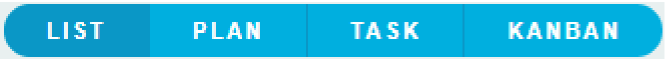
    > **NOTE:** Backlog trackers are tracker items such as epics, stories, defects and so on, for which tasks can be created.

  - Select a task tracker. You can select only one task tracker for a project.
  - Select at least two and up to seven statuses. The number of task swimlanes in the Task Board is equal to the number of statuses you select.
  - Select the unit of effort for backlog and task trackers.

  For more information, see What are Planning, Task and Kanban boards?

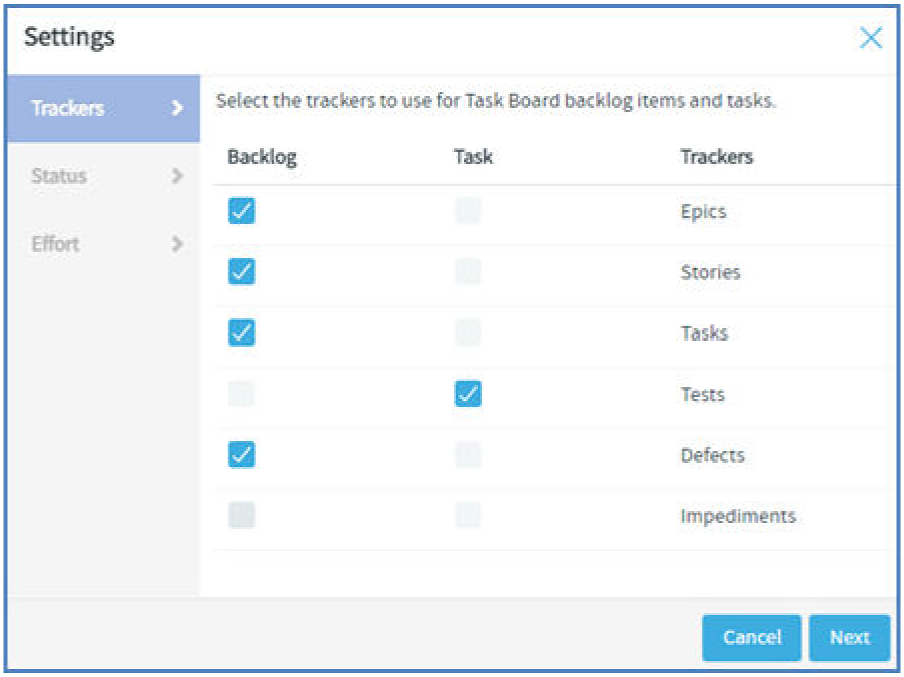
1. Click **TRACKERS** from the **Project Home** menu.

2. In the **List Trackers, Planning Folders and Teams** page, click **Task**.



3. Click the Task Board configuration icon (⚙). The **Settings** window appears.

4. Select one or more backlog trackers and a task tracker.

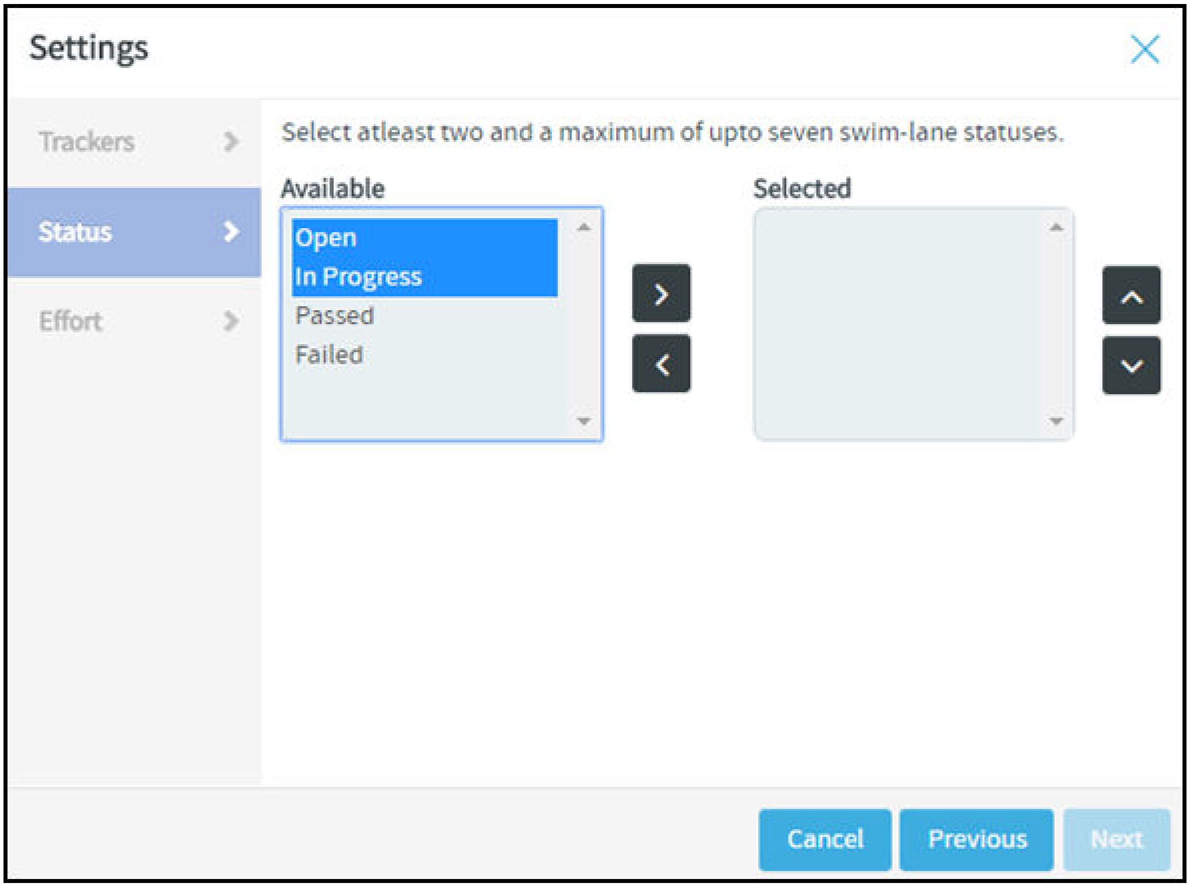


Select the trackers to use for Task Board backlog items and tasks.

Backlog	Task	Trackers
☑	☐	Epics
☑	☐	Stories
☑	☐	Tasks
☐	☑	Tests
☑	☐	Defects
☐	☐	Impediments

NOTE: You can always revisit your backlog and task tracker settings for Task Board and change them if required at any future point in time.

5. Click **Next**.
6. Select at least two and up to seven statuses from the **Available** list box. Press and hold the **Ctrl** key to select multiple items.
7. Click the forward arrow to add the selected statuses to the **Selected** list box.



✓ You can always revisit your status settings for Task Board and change them if required at any future point in time.

✓ Use the forward and backward arrows to move statuses to and fro the **Available** and **Selected** list boxes.

✓ Use the up or down arrows to move statuses up or down within the **Selected** list box.

8. Click **Next**.
9. Select the task effort field and backlog item size field.

> **NOTE:** If the effort field is disabled for the backlog and task trackers, it is not possible to select the task effort and backlog item size fields and the value is set to 'None'.

10. Click **Finish**. The following success message appears: "Task Board preferences saved successfully".

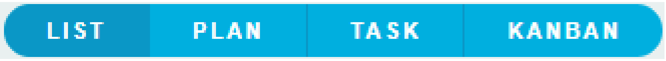
You have now successfully configured the Task Board for your project. Project members can start using the Task Board to manage their tasks.

During a sprint, TeamForge project members can use the Task Board to view tasks, create tasks for backlog items, edit tasks and drag and drop tasks across swimlanes as they progress.

With the Task Board, you can:

- View backlog items of a selected planning folder in the **Backlog Items** swimlane, which is the leftmost swimlane of the task board.
- View tasks (belonging to backlog items) pinned to swimlanes based on the status.
- Edit backlog items and tasks by clicking the artifact ID link.
- Move tasks (cards) from one swimlane to the other as tasks progress.
- Add new tasks for backlog items.

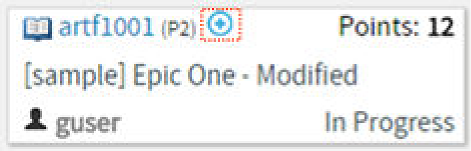
1. Click **TRACKERS** from the **Project Home** menu.
2. In the **List Trackers, Planning Folders and Teams** page, click **TASK**.



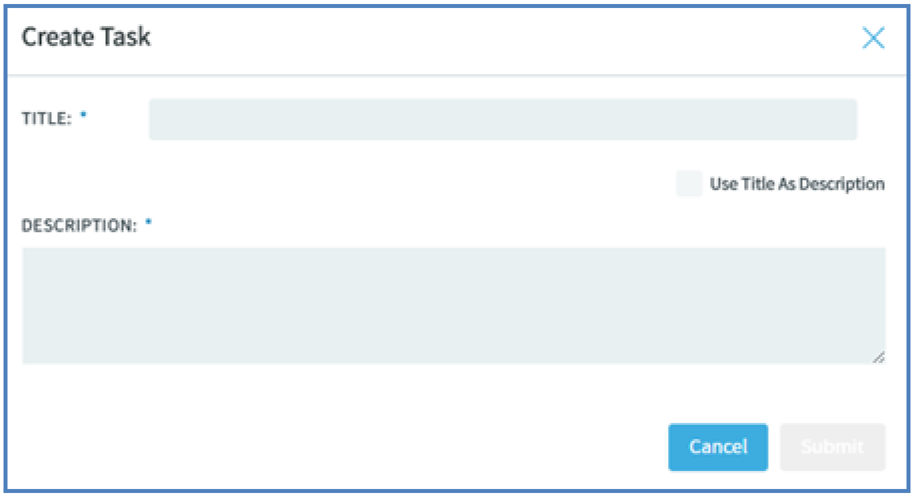
   A Task Board for the current project context is displayed.
3. Select a planning folder from the drop-down list. The Task Board is populated with backlog items and their tasks.

   - If a selected planning folder has subfolders, the artifacts within the subfolders are not displayed.
   - Cards have a color-coded bar to visually identify the priority. In addition, the open and closed cards' background is color-coded to uniquely identify the status.
   - For each card, the artifact ID, title, priority, assigned to and effort (in terms of 'Points' for backlog items and 'Hours' for tasks) are displayed. Status information is shown on backlog cards only. Relevant tooltips appear when you hover your mouse over these data elements. You can also hover your mouse over the tracker icon in backlog and task cards to know the tracker name.

   Add New Tasks
4. Click the **+** icon.



The **Create Task** window appears.

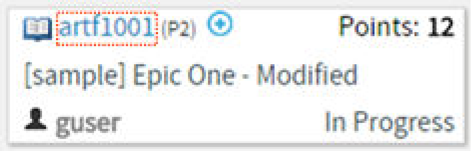


5. Type a title and description and click **Submit**.

> **TIP:** To use the title you type as the description, select the `Use Title as Description’ check box.

To edit a backlog item or task using the task ID link:

6. On the **List Trackers, Planning Folders and Teams** page, select the planning folder.
7. From the listed backlog items and tasks, click the artifact ID link you want to edit.

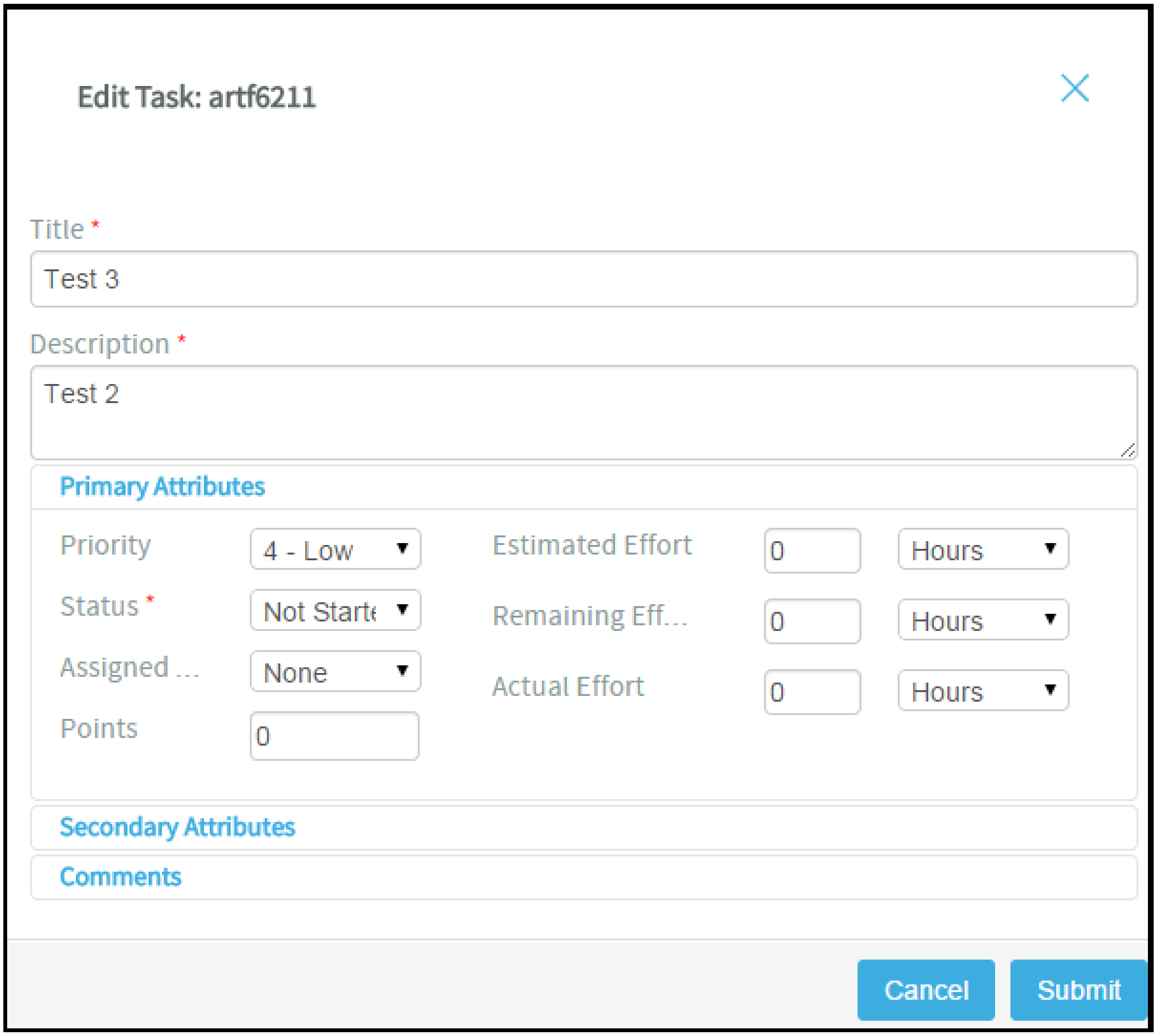


The **View Artifact** page appears.

8. Modify the backlog item or task and click **Update** to update the artifact and return to the Task Board.

To quickly edit a task

9. Click the edit artifact ( ) button in a task. The **Edit Task** window appears.



Edit Task: artf6211

Title * Test 3

Description * Test 2

Priority 4 - Low

Status * Not Starte

Assigned ... None

Points 0

Estimated Effort 0 Hours

Remaining Eff... 0 Hours

Actual Effort 0 Hours

10. You can edit the following on the **Edit Task** window:

- **Artifact title**
- **Artifact description**
- **Primary and secondary attributes**

You have two expandable frames, **Primary Attributes** and **Secondary Attributes**, that list fields you can edit. Primary attributes include Priority, Status, Assigned To, Estimated Effort, Remaining Effort, Actual Effort, Points and Team. The expandable frames and the fields listed in them depend on your application configuration that you set in Tracker Settings. For example:

- Configurable fields which are enabled. For more information, see Enable or Disable Fields.
- User-defined fields (flex fields) with the field type Text Entry and Single Select and set as **Required** (Tracker Settings > Tracker Administration > Create Field). For more information, see Create Custom Tracker Fields.
- Fields which are workflow configured. For more information, see Create a Tracker Workflow.

By default, the following field types, which you select while creating a user-defined field (Project Admin > Tracker Settings), are non-editable from the task board's Edit Task window: Multiple select (Multiple Select), Date picker (Select Date) and User picker (Select User(s)).

- **Comments**

  In this expandable frame, in addition to adding comments, you can view the last five comments for an artifact.

11. After editing an artifact, click **Update**.

Move tasks across swimlanes

12. To move tasks across swimlanes, drag a task card from the source swimlane and drop it on the destination swimlane.

> **NOTE:** You need to have task tracker 'edit' permission to move cards across swimlanes.

View both 'open' and 'closed' backlog items

13. Click the **Show/hide closed artifacts** (✓) toggle button to view both 'open' and 'closed' backlog items.

> **NOTE:** When you select a planning folder, the Task Board shows only 'open' backlog items by default.

Refresh the Task Board

14. Click the **Refresh** (⟳) button to refresh the Task Board.

Auto assign task

If a task is not assigned to anyone, that is, if it is assigned to **None**, then it can be assigned to the user who has logged in using the **Auto Assign Task to Me** feature. This check box appears only if:

- The Task Board is configured.
- The user who logged in has the edit permission to the task tracker.
- The project is not locked.

15. Select the **Auto Assign Task to Me** check box and drag the artifact from one swimlane to another (from one status to another) and the artifact is automatically assigned to you.

Position artifact cards

16. If you have the requisite permission, you can drag and drop a card above or below the other cards within a swimlane to position it in the order you want the artifact addressed.

> **NOTE:** The positioning of cards is retained when you switch between the list and board views (planning, task and kanban). For related information on ranking artifacts in the list view, see Reorder the Contents of a Planning Folder.

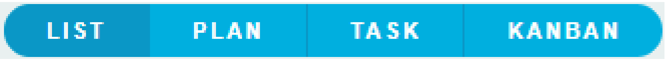
The Kanban Board is a project management tool, which gives you a snapshot of the work items, in which states they are, how they are progressing and if there is any bottleneck to be cleared for a smooth delivery of the product on time.

As a project administrator, you can configure your Kanban Board based on your project requirements.

Create Kanban Board states for the different stages of your project development process, specify workflow constraints for each state and map them to your tracker statuses.

1. Click **TRACKERS** from the **Project Home** menu.

2. In the **List Trackers, Planning Folders and Teams** page, click **Kanban**.

   The following message is displayed prompting you to configure a Kanban Board for the current project.

   ```
   Kanban Board is not configured for the project.
   ```

3. Click the Kanban Board configuration icon (⚙) to open the **Settings** window.

   > **NOTE:** Use the configuration icon when you create a Kanban Board for the first time or when you want to modify the settings of your current Kanban Board.

   Alternatively, you can use **Manage Boards** (☰) to create a new board.

   For more information about its usage, see Use Kanban Board.

4. Give a name to the new Kanban Board.

5. Select the trackers whose artifacts you want to view on your Kanban Board. (These are the trackers you created for your project. For more information on how to create a tracker, see [Create a tracker][trackers-createatracker].)

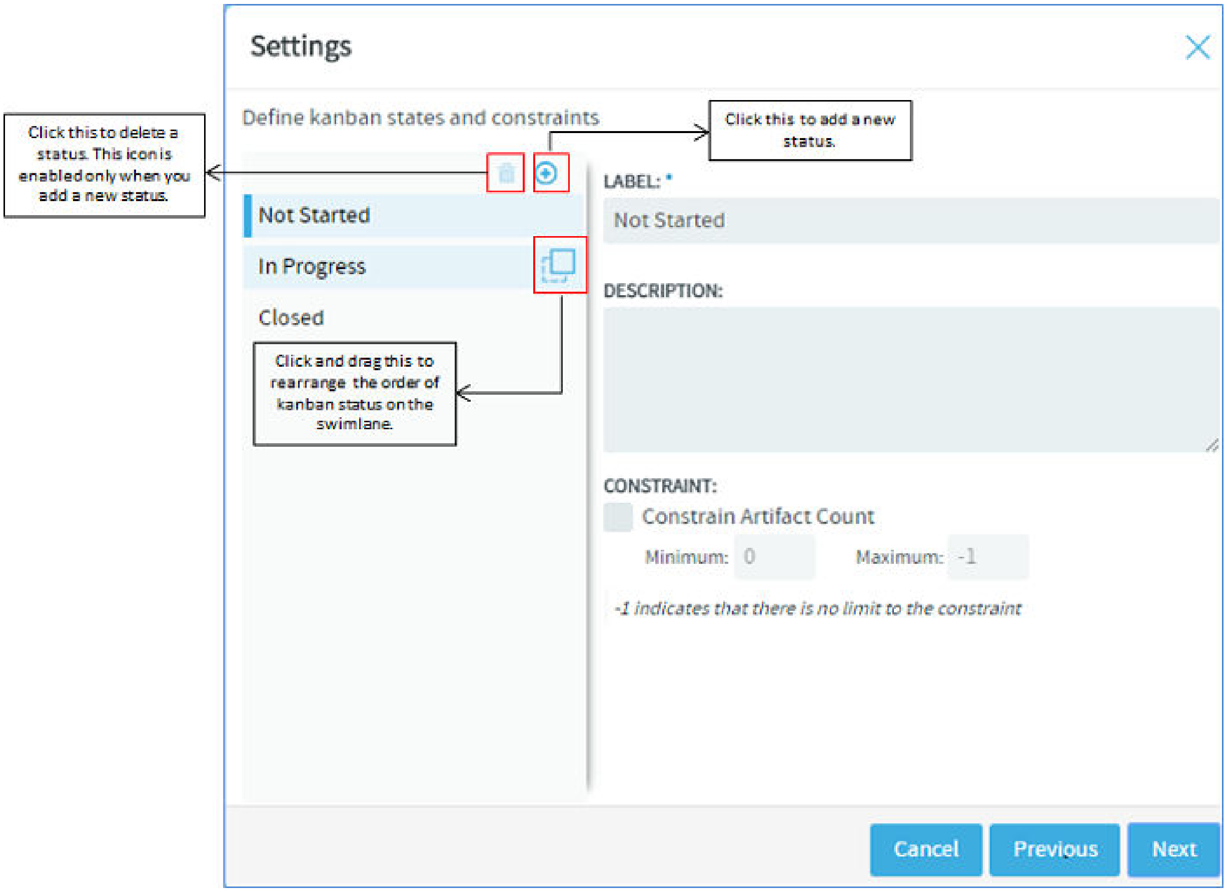
6. Click **Next**.

   Kanban States and Constraints

   A kanban state is the status of a work item in a value stream (swimlanes). The constraints or the limits that you specify here dictates the behaviour of the board. These are checkpoints which eventually help you track the progress of a work item, identify bottlenecks and fix them. The default kanban statuses are 'Not Started', 'In Progress' and 'Closed'.

   You can create your own kanban statuses, delete any state including the default one and rearrange the order in which they would appear as swimlanes on your Kanban Board.

   > **NOTE:** If there are only three kanban statuses, the `Delete Status` icon is disabled disallowing you to delete any more because you must maintain a minimum of three statuses.


Settings
Define kanban states and constraints
Click this to delete a status. This icon is enabled only when you add a new status.
Click this to add a new status.
Not Started
In Progress
Closed
Click and drag this to rearrange the order of kanban status on the swimlane.
LABEL: *
Not Started
DESCRIPTION:
CONSTRAINT:
Constrain Artifact Count
Minimum: 0
Maximum: -1
-1 indicates that there is no limit to the constraint
Cancel
Previous
Next

7. Create and configure kanban statuses.

    1. Click the **Add Status** icon. You can have a maximum of 20 statuses.

    2. Give a label and description to the new status.

        NOTE: The kanban status label can have a maximum of 64 characters and the description, a maximum of 256 characters. Alpha numeric and special characters are allowed.

    3. Select the **Constrain Artifact Count** check box to specify the minimum and maximum workflow constraints, that is, specify the minimum and maximum number of work items (artifact cards) that ought to be present in each status. You may specify these values based on the number of work items in the queue versus the number of resources available.

        Remember the following when you set the minimum or maximum value to -1.

Constraint Name	Logic	Example
Minimum value = -1	This is treated as zero. Therefore, though the maximum value should be greater than the minimum value, the following value range is allowed: Minimum = -1; Maximum = 0 and above	Let us assume that you are in the last iteration of your release. As a project manager, you would not want to see any artifact in the 'Impeded' status. So you set the minimum to -1 and maximum to 0, which translates to zero artifacts. When this constraint is applied, your Kanban Board flags a violation, if artifacts show up in the 'Impeded' status, thereby drawing your attention to address the issue immediately.
Maximum value = -1	This indicates there is no limit to the maximum value.	You may choose to set -1 as the maximum constraint for the 'Closed' status because you would want to see as many closed artifacts as possible.

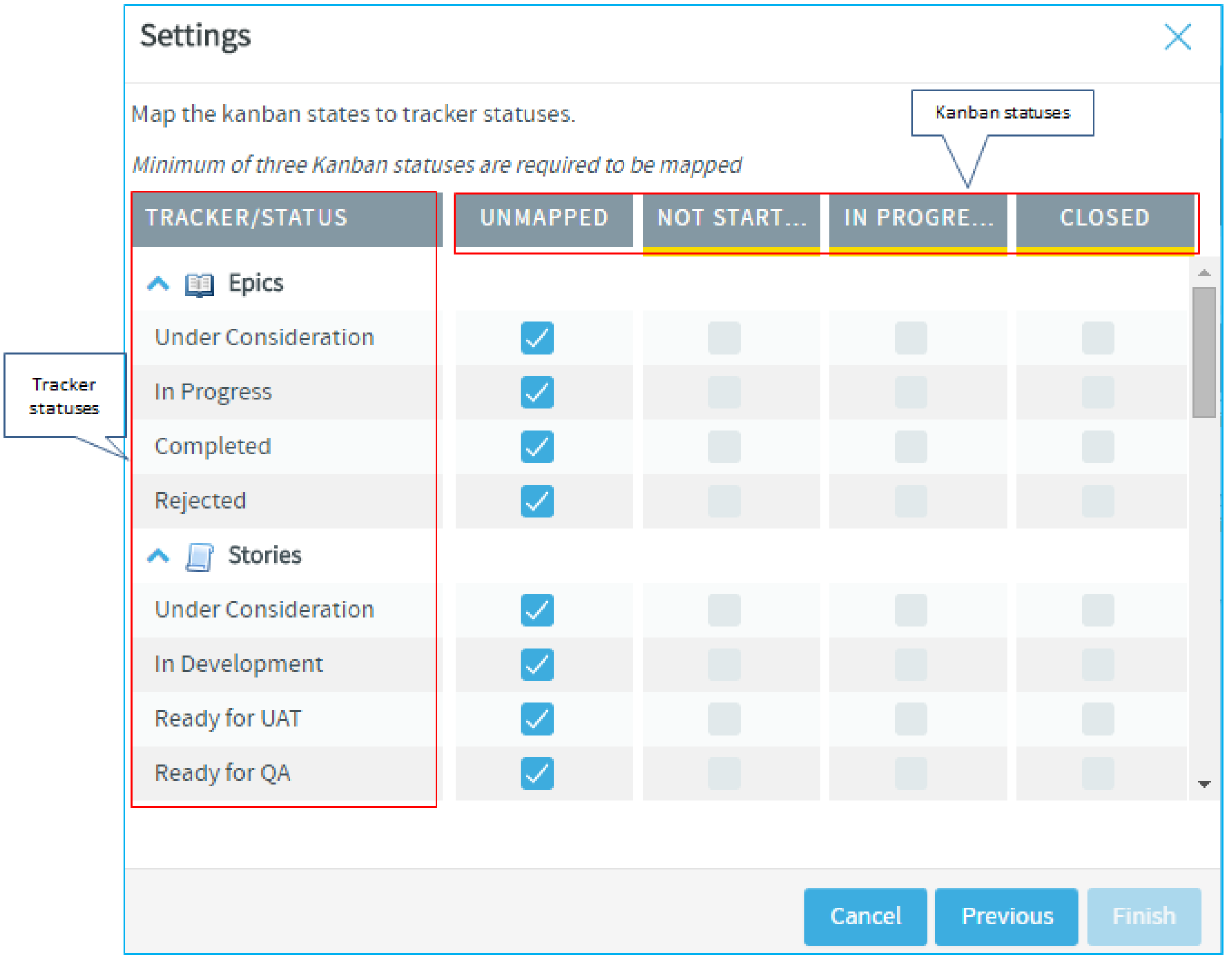
> **IMPORTANT:** These constraints are applicable only at the planning folder level and not at the team level because the Kanban Board is expected to give an overall visibility to the work items for an iteration or release within a planning folder and not just within a specific team.

Map Kanban States with Tracker Status

Next, make a logical mapping of the kanban states with the tracker statuses to view those tracker artifacts on your Kanban Board. For example, to view defects, you must first make a meaningful mapping of the kanban states with the defect tracker status(es). The mapped kanban states appear as swimlanes on the Kanban Board.

The trackers (you had selected in Step 5) and their statuses appear on the left. All the kanban statuses appear at the top. The tracker statuses are unmapped by default.

- You need to map a minimum of 3 kanban statuses.
- You can map a single kanban status to one or more tracker statuses, but you cannot map a single tracker status to more than one kanban status.



Settings

Map the kanban states to tracker statuses.

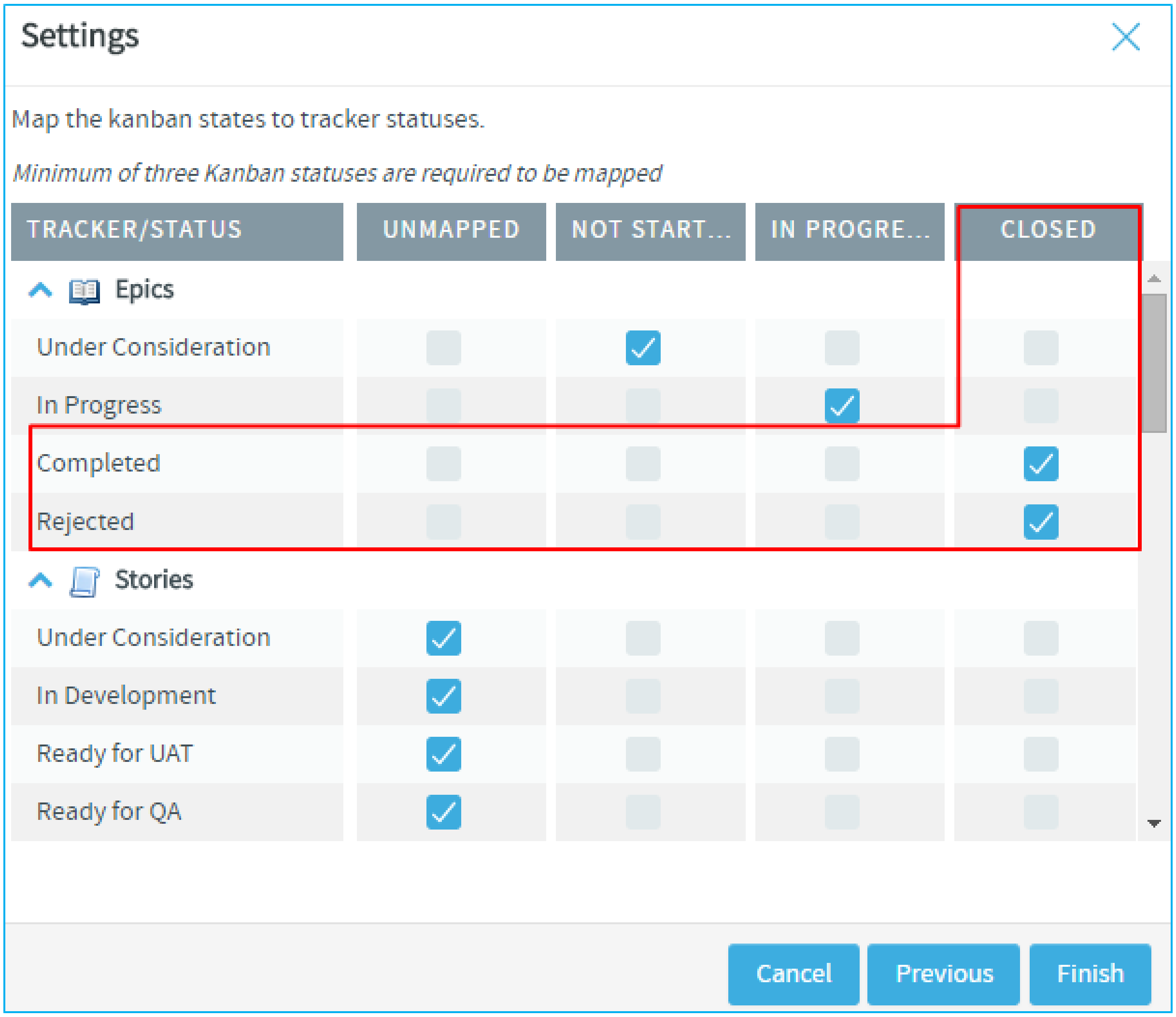
Minimum of three Kanban statuses are required to be mapped

TRACKER/STATUS	UNMAPPED	NOT START...	IN PROGRE...	CLOSED
Epics				
Under Consideration	☑	□	□	□
In Progress	☑	□	□	□
Completed	☑	□	□	□
Rejected	☑	□	□	□
Stories				
Under Consideration	☑	□	□	□
In Development	☑	□	□	□
Ready for UAT	☑	□	□	□
Ready for QA	☑	□	□	□

Kanban statuses

Tracker statuses

8. Click the appropriate check box. For example, for an Epic tracker, map ‘Completed’ and ‘Rejected’ tracker statuses with the Kanban statuses ‘Closed’ by selecting the appropriate check boxes as shown in the following screenshot.



Settings

Map the kanban states to tracker statuses.

Minimum of three Kanban statuses are required to be mapped

TRACKER/STATUS	UNMAPPED	NOT START...	IN PROGRE...	CLOSED
Epics				
Under Consideration	□	☑	□	□
In Progress	□	□	☑	□
Completed	□	□	□	☑
Rejected	□	□	□	☑
Stories				
Under Consideration	☑	□	□	□
In Development	☑	□	□	□
Ready for UAT	☑	□	□	□
Ready for QA	☑	□	□	□

9. Click **Finish** to complete the process.

Manage Boards

You can create and manage multiple kanban boards per your requirement. As a project administrator, you may want to create multiple boards at the iteration level, at each unit level, (Dev, QA and so on), at the hierarchical level or at each project team level, depending upon the needs of the users. For example, to get the big picture and make executive decisions you may want to create a kanban board at the management level; at the same time, at a project member level, you can create a board to focus on the immediate tasks to be completed.

10. Use the various icons available in **Manage Boards** to maintain multiple kanban boards in a project.

To do this...	Click this icon
To use **Manage Boards**	☰
To create a new board	⊕
To edit a board	✎
To delete a board	🗑
To set a default board	☑

NOTE: You cannot delete the default Kanban Board. When you delete your current active board, the default one shows up automatically.

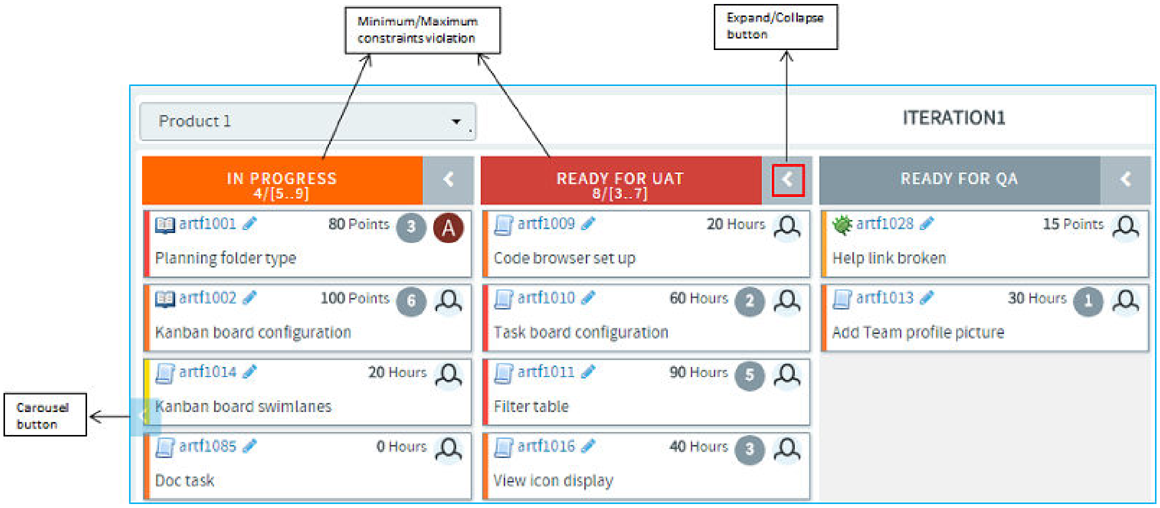
After the kanban board is configured, you can use it to view, plan and track work items for the selected planning folder or the selected team within that planning folder.

1. Click **TRACKERS** from the **Project Home** menu.
2. In the **List Trackers, Planning Folders and Teams** page, click **Kanban**.

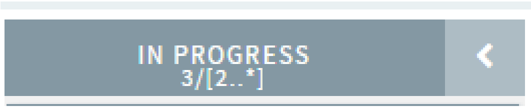
3. Select the required kanban board from **Manage Boards** (available only for project administrators). If you are a project member, select your kanban board from the **Select Board** list. The selected kanban board for the current project context is displayed.
4. Select the planning folder for which you want to view the status of the artifacts. If your project contains teams, the **Select a Team** drop-down list appears. From this drop-down, select **All artifacts** or if you want to view artifacts specific to a team within the selected planning folder, select the relevant team. Depending upon your selection, artifacts pertaining to the mapped trackers are displayed in the appropriate swimlane.

   - You can view a maximum of 5 swimlanes at a time on your Kanban Board. If there are more, use the carousel scroll to slide across the swimlanes. You can expand or collapse each swimlane. When there are no results to display in a swimlane, the **Expand/Collapse** button does not appear.
   - Based on the configuration values, if the constraints are violated, the relevant status headers are highlighted appropriately. A violation of minimum constraint is highlighted in orange indicating that resources are underutilized whereas that of maximum constraint is highlighted in red indicating

resources being overloaded and bottlenecks to be fixed. For more information on how a kanban board is configured, see Set up Kanban Board.



- Each swimlane header displays the status label, the total number of artifacts within the selected planning folder, and the minimum and maximum constraints that were configured for the kanban state.



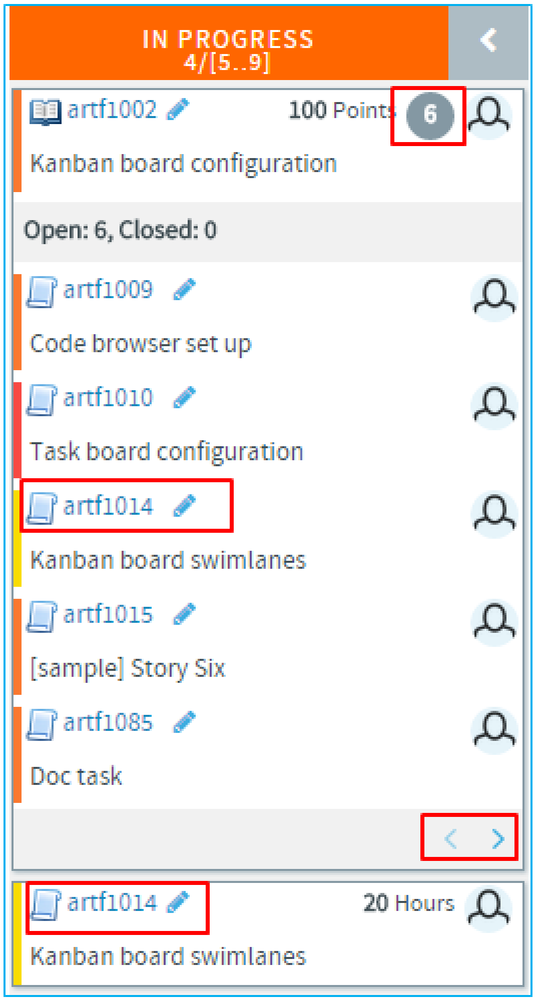
In the above scenario, for the ‘In Progress’ status:

  - ‘3’ is the total number of artifacts in the ‘In Progress’ status within the selected planning folder.
  - ‘2’ indicates the minimum constraint and ‘*’ indicates the maximum constraint ‘-1’. (-1 as the minimum constraint translates to 0 whereas -1 as maximum indicates there is no limit and so represented by an *.)

    > **IMPORTANT:** These values and constraint validation are planning folder specific and not team specific. For example, in the above scenario, the total number of artifacts is the overall count of artifacts which are in the ‘In Progress’ status within the selected planning folder and not within the selected team.

- Each artifact card displays its points (story points) or estimated effort if these fields are enabled in your tracker settings. If both are enabled, only the estimated effort is displayed hovering the mouse on which the artifact's points is displayed. Similarly, if all the effort fields (estimated, remaining and actuals) are enabled, you can view them by hovering the mouse on the estimated effort.

- Cards have a color-coded bar to visually identify the priority. In addition, the open and closed cards' background is color-coded to uniquely identify the status.
- A parent artifact card displays the count of its child artifacts (**Show/Hide child artifacts** button). Click this button to view or hide the child artifacts. If there are more than 5 child artifacts, when you expand the parent artifact, you will see pagination arrows ('Previous'/Next') at the bottom of the parent artifact card.



- If the tracker types of both the parent and child artifacts have been mapped with the kanban statuses, the child artifacts appear within their parent artifact card and also as an individual card.

In the above scenario, Epic (parent artifact) and Story (child artifact) trackers have been mapped to the kanban status ‘In Progress’. Story ‘artf1014’ is the child artifact of the epic ‘artf1002’. So you will see ‘artf1014’ within the parent artifact card as well as outside of it as an individual artifact card.

- When a child artifact is closed, a ‘Closed’ tag appears next to the child artifact ID within the parent artifact card.

Edit artifacts

Use Kanban Board to edit an artifact using the Edit icon on the artifact card or move artifacts from one status to another appropriately. Based on the status changes you make, the swimlane headers get updated appropriately.

Remember:
You can edit an artifact only if you have the tracker edit permission. Otherwise, you can only view it.

Artifacts support @mentions: Artifact description and comments now support @mentions and users called out via @mentions are added to the monitoring list. Include usernames with “@” as prefix (for example, @mphippard) to add users to the monitoring list.

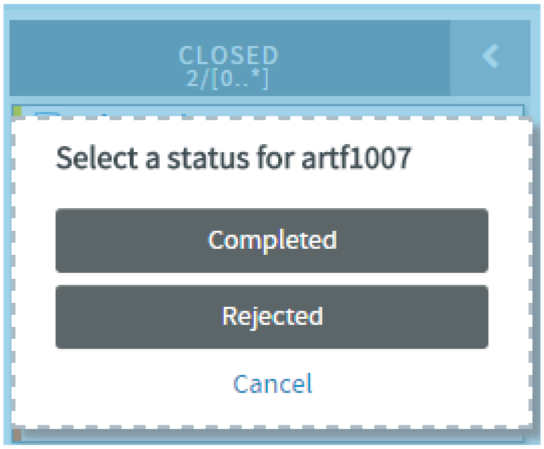
> **NOTE:** Users called out via @mentions must have `Artifact View` permission to be added to the monitoring list.

5. When you drag the artifacts from one swimlane to another, note the following validations:

    - The changes you see in swimlane headers do not apply to any specific team but takes into account the total count of artifacts within the selected planning folder.
    - You cannot move an artifact card to a status that is not mapped to that particular tracker type. For example, a ‘Ready for QA’ kanban status may not have been mapped to any of the epic tracker statuses. So when you attempt to move an epic to that status, you will get an `Invalid status configuration` error.
    - You can move a parent artifact to the ‘Closed’ status only if all its child artifacts are closed. However, a child artifact is independent of its parent with regard to the status change, that is, when you move a parent artifact from one status to another, the status of the child may still remain unchanged. For example, an epic may have many stories, the status of some may change

from ‘Not Started’ to ‘In Progress’ whereas some may not. So when the status of an epic as one single unit may change, it need not apply to all of its child artifacts.

- When your move violates a constraint (minimum or maximum), a warning is displayed; but you can still move the artifacts.
- If a kanban status is mapped to more than one tracker status, when you move an artifact, you have an option to choose a status as shown in the following screen shot.



Select a status for artf1007

Create artifacts and child artifacts

Using the Create Artifact icon on the top right of your kanban board, you can quickly create an artifact. The Create Artifact icon appears only when you configure a kanban board and select a planning folder.

Similarly, using the Create Child Artifact icon available on the artifact card, you can quickly create a child artifact for any artifact card displayed on Kanban Board.

> **NOTE:** The Create Child Artifact icon does not appear on closed artifact cards.

You can create an artifact or a child artifact only if you have the required tracker permission.

Artifacts support @mentions: Artifact description and comments now support @mentions and users called out via @mentions are added to the monitoring list. Include usernames with “@” as prefix (for example, @mphippard) to add users to the monitoring list.

> **NOTE:** Users called out via @mentions must have `Artifact View` permission to be added to the monitoring list.

6. Click the required icon: Create Artifact or Create Child Artifact (⊕).
7. Enter the required information in the relevant window and click **Submit**.
   - Only trackers configured for the kanban board appear in the Trackers drop-down list.
   - While you quickly add artifacts with data for just three fields such as the tracker type, title and description, the artifacts are, however, saved with default values for other required fields, which you may choose to update later. If the default state of the selected tracker is not mapped for the newly created artifact or child artifact, the artifact card does not show up on the kanban board.

Show / Hide closed cards

8. When you have a large volume of closed cards in a planning folder, you can restrict the number of closed cards you want to view by toggling between the 'Show all closed artifacts' icon (📅).

   Or the 'Hide artifacts older than 60 days' icon using which you can hide closed cards older than 60 days. This is the default option (📅).

> **NOTE:** The Show/Hide toggle icon appears only on configured kanban boards. Your selection is saved for the subsequent sessions as well.

Position artifact cards

9. If you have the requisite permission, you can drag and drop a card above or below the other cards within a swim lane to position it in the order you want the artifact addressed.

> **NOTE:** The positioning of cards is retained when you switch between the list and board views (planning, task and kanban). For related information on ranking artifacts in the list view, see Reorder the Contents of a Planning Folder.

The Team view in Planning Board provides an option to filter artifacts for a specific team within the selected planning folder.

You can select the same planning folder in all three swimlanes. This enables an efficient and faster way of planning a sprint simultaneously for three different teams residing in the same planning folder. You can also select the same planning folder and same team in all three swimlanes for easy handling of artifact cards.

View Artifact Cards

1. Click **TRACKERS** from the **Project Home** menu.
2. In the **List Trackers, Planning Folders and Teams** page, click **PLAN**. A Planning Board for the current project context is displayed.
3. Select a planning folder from the drop-down list.

   > **NOTE:** You can select the same planning folder in all three swimlanes.

4. View the artifact cards based on your requirement:
   - Select a team from the **Select a Team** drop-down list to view artifacts of a specific team.
   - Select **None** to view unassigned artifacts.
   - Select **All artifacts** to view all the artifact cards residing in the selected planning folder.

   > **NOTE:** If there are no teams created for a project, then the `Select a Team` drop-down list is not displayed at all.

Assign Artifacts to a Team

You can select an unassigned artifact from one swimlane and drag it to another where you have a team selected.

> **NOTE:** Whenever you move an artifact card from one swimlane to another, all its child artifacts (if any) are also moved along with it.

The Team view in Task Board provides an option to filter backlog items for a specific team within the selected planning folder.

This team’s view of artifacts applies only to backlog items. When you filter and view the backlog items for a specific team, the tasks (if any) within the filtered backlog items are also displayed in the relevant status swimlanes. However, you cannot filter tasks alone for a team.

You can move the artifact cards between swimlanes. If the artifacts are workflow configured, appropriate error messages are displayed when a workflow rule is broken.

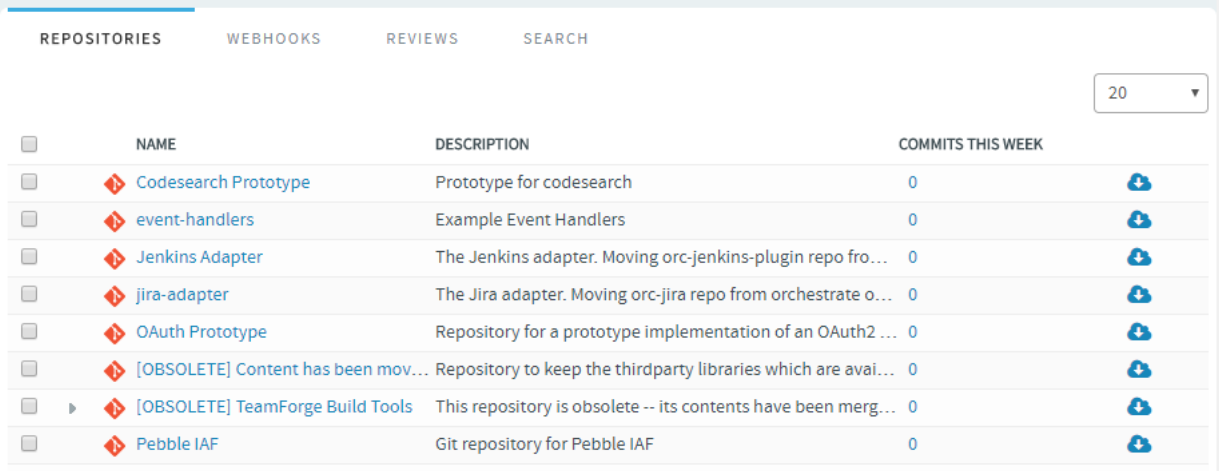
1. Click **TRACKERS** from the **Project Home** menu.
2. In the **List Trackers, Planning Folders and Teams** page, click **TASK**. A Task Board for the current project context is displayed.
3. Select a planning folder from the drop-down list. All the backlog items and their tasks in the selected planning folder are displayed.
4. Select a team from the **Select a Team** drop-down list.

   > **NOTE:** If there are no teams created for a project, then the `Select a Team` drop-down list is not displayed at all.

   All the backlog items and their tasks in the selected team are displayed.

For Subversion and Git repositories, you have the option to use the TeamForge code browser which is turned on by default while integrating the source code server.

1. Click **SOURCE CODE** from the **Project Home** menu.
2. Select the **Repositories** tab.



□	NAME	DESCRIPTION	COMMITS THIS WEEK
□	Codesearch Prototype	Prototype for codesearch	0
□	event-handlers	Example Event Handlers	0
□	Jenkins Adapter	The Jenkins adapter. Moving orc-jenkins-plugin repo fro…	0
□	jira-adapter	The Jira adapter. Moving orc-jira repo from orchestrate o…	0
□	OAuth Prototype	Repository for a prototype implementation of an OAuth2 …	0
□	[OBSOLETE] Content has been mov…	Repository to keep the thirdparty libraries which are avai…	0
□	[OBSOLETE] TeamForge Build Tools	This repository is obsolete -- its contents have been merg…	0
□	Pebble IAF	Git repository for Pebble IAF	0

3. From the list of project repositories, select the repository you want to look at.

   Click the name of a Subversion or a Git repository in which you want to view code. On the top right of code browser, you can select the branch/tag (for Git) or specify the revision (for SVN) you want to browse.

The View Tab

This tab allows you to do the following:

- Browse through the folder hierarchy of the repository and view the content of specific files. For any folder or file you are viewing within a branch (Git) or revision (SVN), you can obtain the commit information pertaining to its last update.

- While viewing a single specific commit or a file, you can see the paths that were modified in that commit, the associations including JIRA such as builds, code reviews and so on (from EventQ events) for the specific commit and the difference between files in that commit.

> **IMPORTANT:** To view the associations, you must have installed EventQ and must have RBAC (role-based access control) permission to use the “EventQ READ” or “Reporting API” of TeamForge EventQ. If either of these requirements is not met, this section will not show up at all.

- While viewing a folder, if there is a file named `readme`, `readme.txt` or `readme.md`, that file will automatically be rendered beneath the list of files in the folder. If the file contains markdown formatting, it will be rendered as rich text.

The `Changes` Tab

This tab lets you view all of the commits that touched a specific path you are browsing within a branch or revision. Click a commit to view its details.

The `Graph` Tab

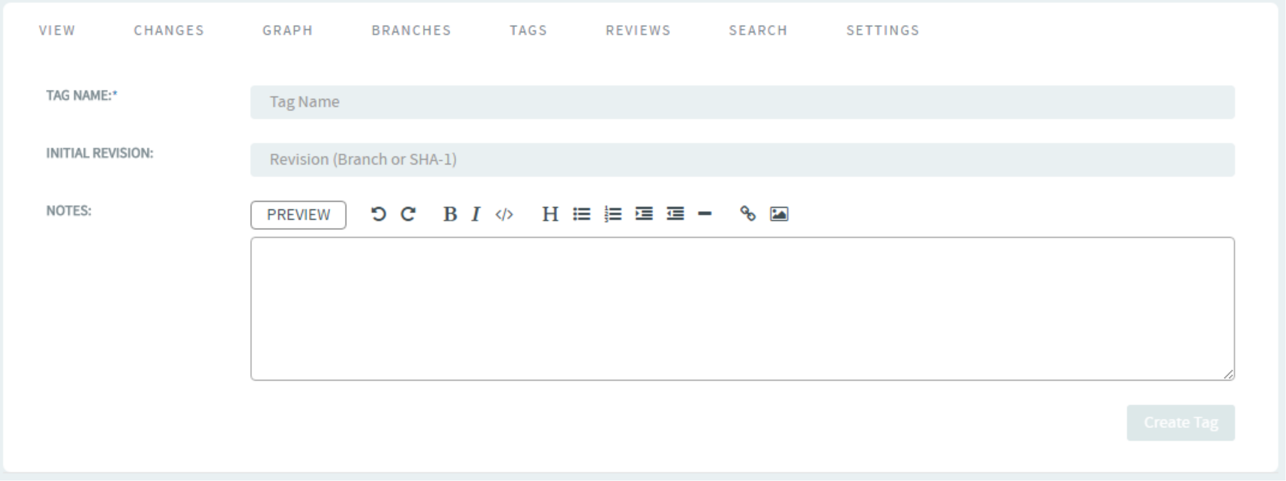
This tab provides a graphical representation of the changes made including branching and merging of repositories.

The `Branches` Tab (for Git)

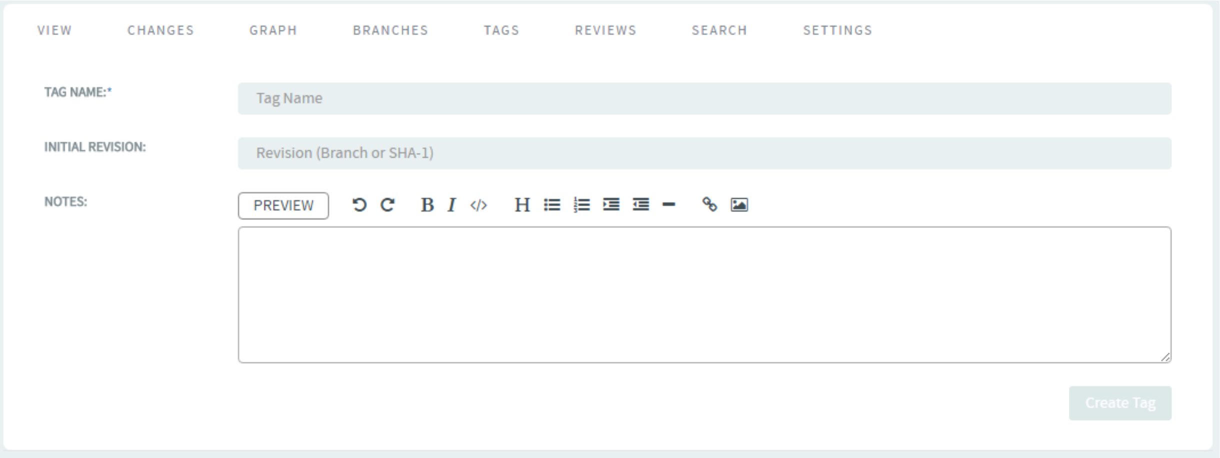
This allows you to see all of the branches in the repository in their relation to the default (master) one. Using **Compare Branch**, you can see the commits in the branch that do not exist in the default branch.

The `Tags` Tab

This tab lets you create Git tags and tag specific points in history as being important. Typically, you can use this functionality to mark release points (v1.0 and so on) with an option to add Release Notes for the tagged version. Once you create a tag, you can use it to download source code as a zip/tar file and view the tag information in *Changes and Graph* tabs.



1. To create Git tags and tag specific points in history as being important (to mark release points, for example, v1.0, and so on), select the *TAGS* tab and click **Create Tag**.
2. Type a tag name and revision number and add a Release Notes for the tag. Click **Create Tag**.

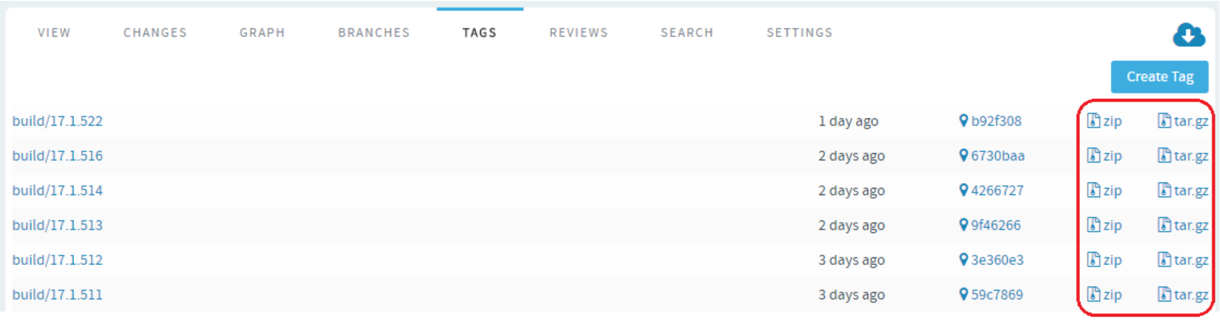


TAG NAME:* Tag Name

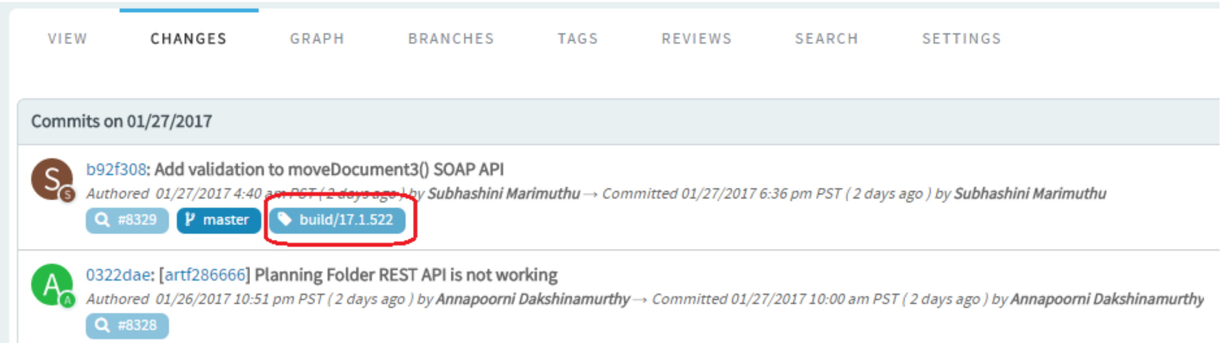
INITIAL REVISION: Revision (Branch or SHA-1)

NOTES:

Once you create a tag, you can use it download source code as a zip/tar file and view the tag information in *Changes* and *Graphs* tabs.



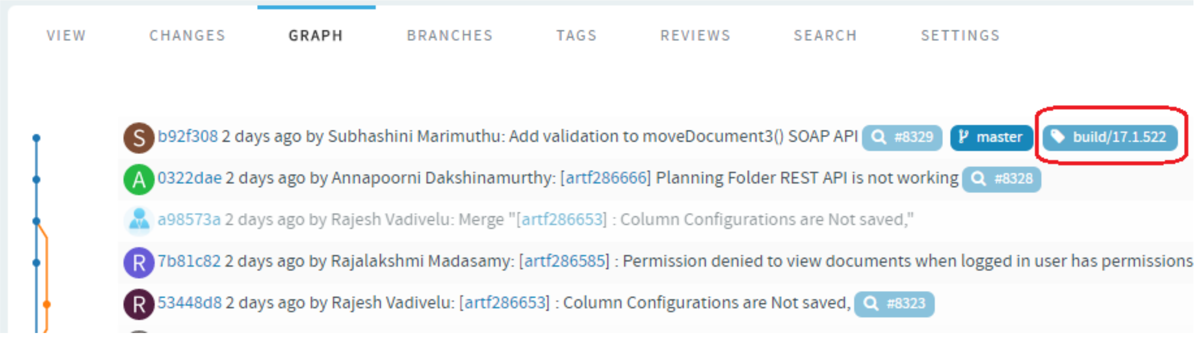
build/17.1.522	1 day ago	b92f308	zip	tar.gz
build/17.1.516	2 days ago	6730baa	zip	tar.gz
build/17.1.514	2 days ago	4266727	zip	tar.gz
build/17.1.513	2 days ago	9f46266	zip	tar.gz
build/17.1.512	3 days ago	3e360e3	zip	tar.gz
build/17.1.511	3 days ago	59c7869	zip	tar.gz



Commits on 01/27/2017

b92f308: Add validation to moveDocument3() SOAP API
Authored 01/27/2017 4:40 am PST (2 days ago) by ***Subhashini Marimuthu*** *→ Committed 01/27/2017 6:36 pm PST (2 days ago) by* ***Subhashini Marimuthu***
#8329 master build/17.1.522

0322dae: [artf286666] Planning Folder REST API is not working
Authored 01/26/2017 10:51 pm PST (2 days ago) by ***Annapoorni Dakshinamurthy*** *→ Committed 01/27/2017 10:00 am PST (2 days ago) by* ***Annapoorni Dakshinamurthy***
#8328



b92f308 2 days ago by Subhashini Marimuthu: Add validation to moveDocument3() SOAP API #8329 master build/17.1.522

0322dae 2 days ago by Annapoorni Dakshinamurthy: [artf286666] Planning Folder REST API is not working #8328

a98573a 2 days ago by Rajesh Vadivelu: Merge "[artf286653] : Column Configurations are Not saved,"

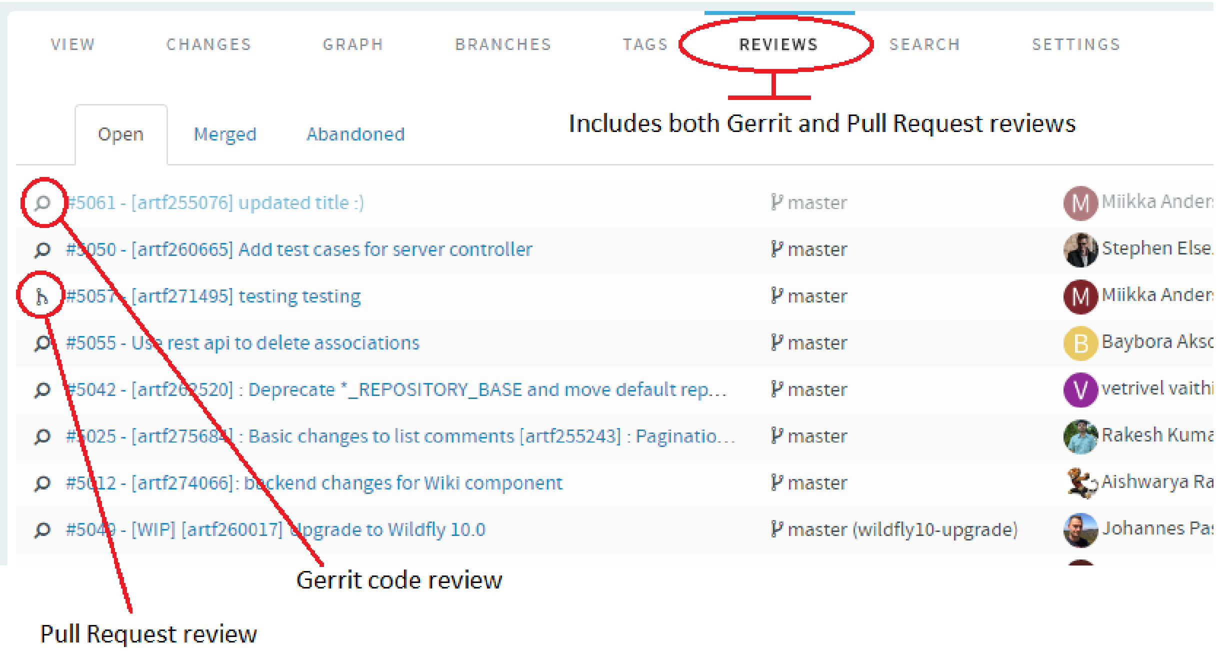
7b81c82 2 days ago by Rajalakshmi Madasamy: [artf286585] : Permission denied to view documents when logged in user has permissions

53448d8 2 days ago by Rajesh Vadivelu: [artf286653] : Column Configurations are Not saved, #8323

The `Reviews` Tab

This tab lists all the Open, Merged, and Abandoned reviews, both Pull Requests and Gerrit single-commit reviews. Pull requests allow developers to collaborate with each other on a code change before merging it into another branch on a GIT repository. You can access this tab only when the repository owner has enabled this feature. For more information, see Pull Request: Step by Step.

- **Support for Both Pull Requests and Single-commit Gerrit Reviews**: Supports all types of code review policies, which include Pull Requests and single commit Gerrit Reviews.



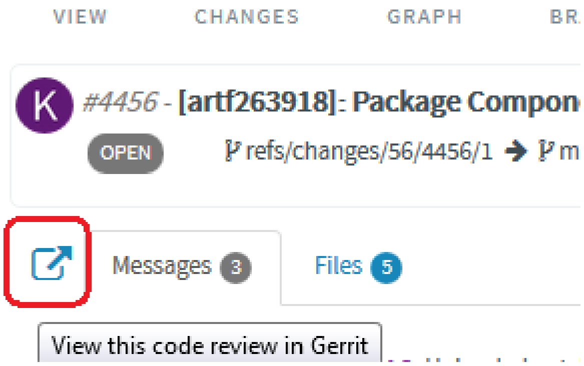
Includes both Gerrit and Pull Request reviews

#5061 - [artf255076] updated title :) master Miikka Ander
#5050 - [artf260665] Add test cases for server controller master Stephen Else
#5057 - [artf271495] testing testing master Miikka Ander
#5055 - Use rest api to delete associations master Baybora Aksc
#5042 - [artf262520] : Deprecate *_REPOSITORY_BASE and move default rep... master vetrivel vaithi
#5025 - [artf275684] : Basic changes to list comments [artf255243] : Paginatio... master Rakesh Kuma
#5012 - [artf274066]: backend changes for Wiki component master Aishwarya Ra
#5049 - [WIP] [artf260017] Upgrade to Wildfly 10.0 master (wildfly10-upgrade) Johannes Pa

Gerrit code review

Pull Request review

- **Auto Refresh When a Pull Request Changes**: When a pull request changes, the page is automatically refreshed to reflect the changes.
- **Comments to Support @mentions**: Inline comments are parsed for @mentions and users called out via @mentions are added as reviewers.
- **Open Your Gerrit Reviews in Gerrit's User Interface**: A new button has been added to let you open your Gerrit Reviews in Gerrit's user interface.

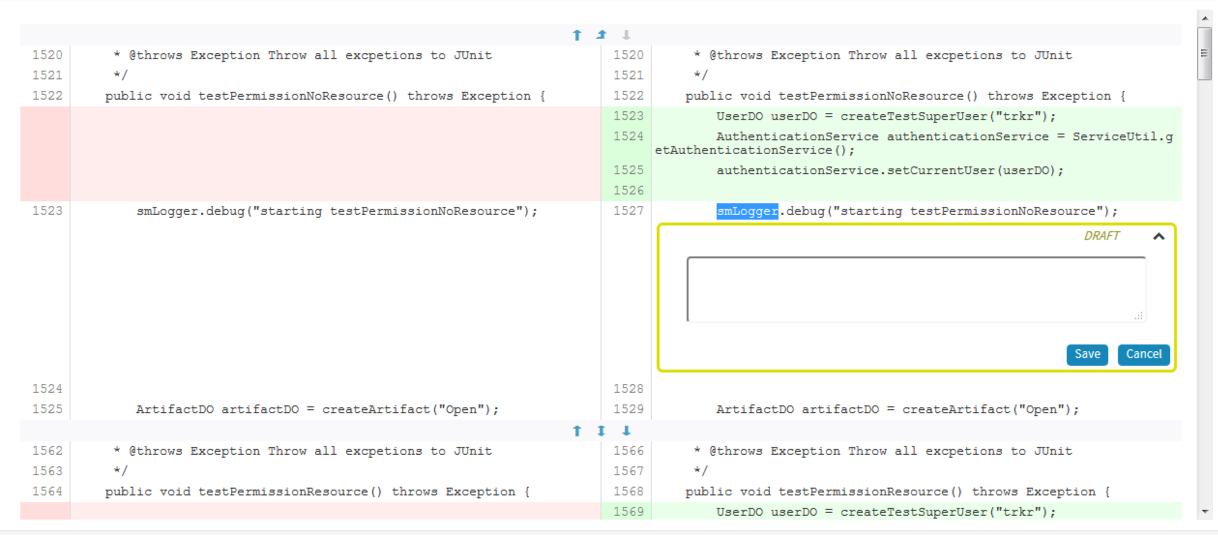


#4456 - [artf263918]: Package Compon

refs/changes/56/4456/1 → m

View this code review in Gerrit

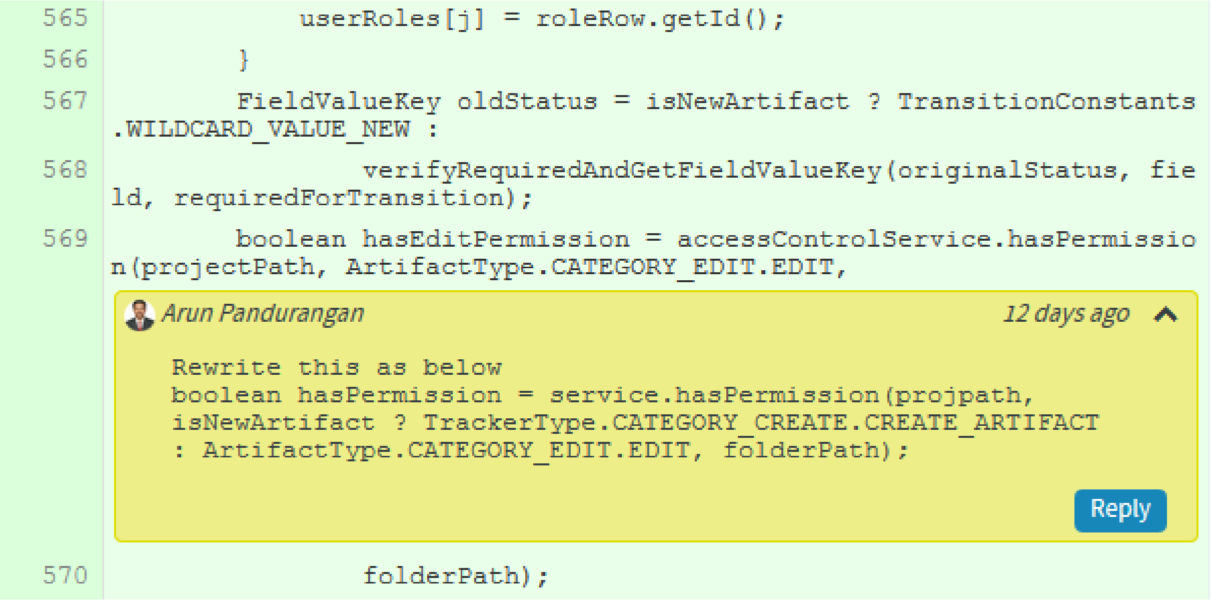
- **Code Commenting**: During code reviews, you can now add line comments in context while looking at the files in diff view. You can double-click to block a line/text and add a comment.



```
     * @throws Exception Throw all excpetions to JUnit
     */
    public void testPermissionNoResource() throws Exception {
        smLogger.debug("starting testPermissionNoResource");

        ArtifactDO artifactDO = createArtifact("Open");

     * @throws Exception Throw all excpetions to JUnit
     */
    public void testPermissionResource() throws Exception {
```

```
     * @throws Exception Throw all excpetions to JUnit
     */
    public void testPermissionNoResource() throws Exception {
        UserDO userDO = createTestSuperUser("trkr");
        AuthenticationService authenticationService = ServiceUtil.g
etAuthenticationService();
        authenticationService.setCurrentUser(userDO);

        smLogger.debug("starting testPermissionNoResource");

        ArtifactDO artifactDO = createArtifact("Open");

     * @throws Exception Throw all excpetions to JUnit
     */
    public void testPermissionResource() throws Exception {
        UserDO userDO = createTestSuperUser("trkr");
```

You can also reply to line comments.



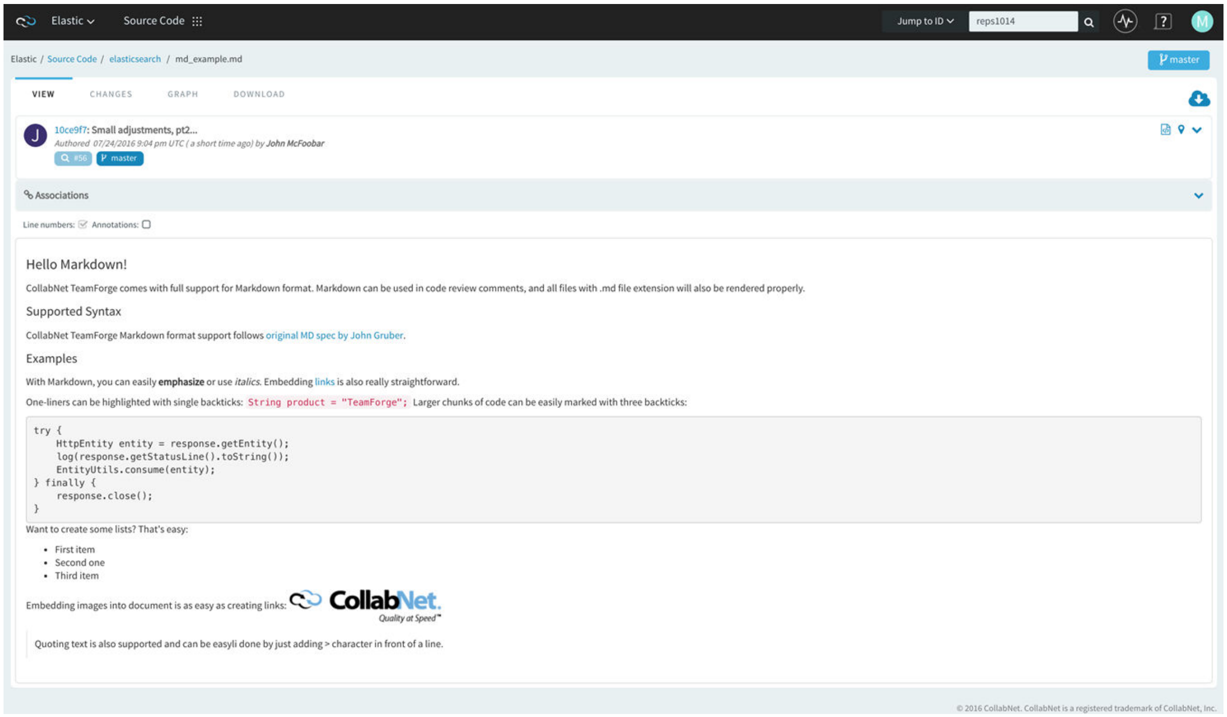
```
            userRoles[j] = roleRow.getId();
        }
        FieldValueKey oldStatus = isNewArtifact ? TransitionConstants
.WILDCARD_VALUE_NEW :
            verifyRequiredAndGetFieldValueKey(originalStatus, fie
ld, requiredForTransition);
        boolean hasEditPermission = accessControlService.hasPermissio
n(projectPath, ArtifactType.CATEGORY_EDIT.EDIT,
```

Arun Pandurangan 12 days ago

```
Rewrite this as below
boolean hasPermission = service.hasPermission(projpath,
isNewArtifact ? TrackerType.CATEGORY_CREATE.CREATE_ARTIFACT
: ArtifactType.CATEGORY_EDIT.EDIT, folderPath);
```

```
            folderPath);
```

- **Ability to Diff the Change Against the Base or a Previous Patch Set**: As part of the Gerrit review workflow, you now have the ability to diff the change against the Base or a previous Patch Set.
- **Markdown Support**: Markdown support for all .MD files: Render Markdown files when viewed through Code Browser.



Elastic / Source Code / elasticsearch / md_example.md

10ce9f7: Small adjustments, pt2...
Authored 07/24/2016 9:04 pm UTC (a short time ago) by John McFoobar

Hello Markdown!

CollabNet TeamForge comes with full support for Markdown format. Markdown can be used in code review comments, and all files with .md file extension will also be rendered properly.

Supported Syntax

CollabNet TeamForge Markdown format support follows original MD spec by John Gruber.

Examples

With Markdown, you can easily **emphasize** or use *italics*. Embedding links is also really straightforward.

One-liners can be highlighted with single backticks: `String product = "TeamForge";` Larger chunks of code can be easily marked with three backticks:

```
try {
    HttpEntity entity = response.getEntity();
    log(response.getStatusLine().toString());
    EntityUtils.consume(entity);
} finally {
    response.close();
}
```

Want to create some lists? That's easy:

- First item
- Second one
- Third item

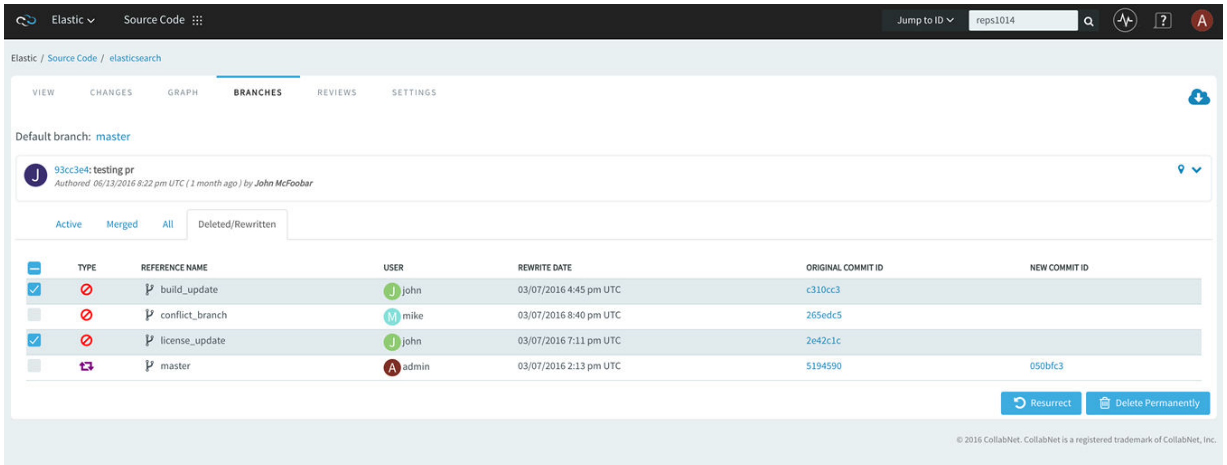
Embedding images into document is as easy as creating links: CollabNet. Quality at Speed™

> Quoting text is also supported and can be easyli done by just adding > character in front of a line.

© 2016 CollabNet. CollabNet is a registered trademark of CollabNet, Inc.

TeamForge uses Showdown—a bidirectional Markdown to HTML to Markdown converter written in Javascript. For more information, see the official Showdown Documentation. Here's an abridged version of the Markdown syntax documentation.

- **Mass Delete/Resurrect Options**: Mass delete/resurrect options in *History Protect* tab:



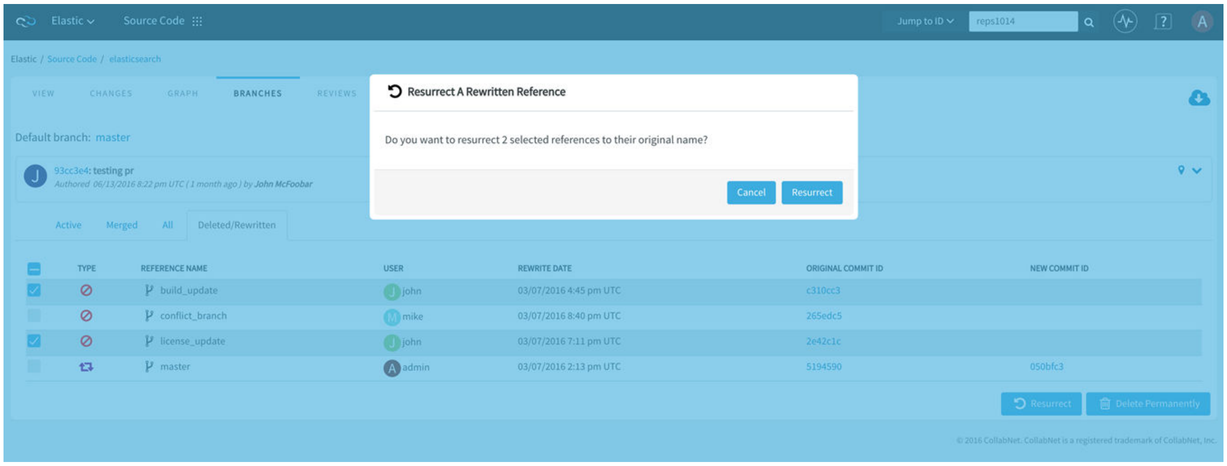
Elastic / Source Code / elasticsearch

Default branch: master

93cc3e4: testing pr
Authored 06/13/2016 8:22 pm UTC (1 month ago) by John McFoobar

☐	TYPE	REFERENCE NAME	USER	REWRITE DATE	ORIGINAL COMMIT ID	NEW COMMIT ID
☑	⊘	build_update	john	03/07/2016 4:45 pm UTC	c310cc3	
☐	⊘	conflict_branch	mike	03/07/2016 8:40 pm UTC	265edc5	
☑	⊘	license_update	john	03/07/2016 7:11 pm UTC	2e42c1c	
☐	⇄	master	admin	03/07/2016 2:13 pm UTC	5194590	050bfc3

© 2016 CollabNet. CollabNet is a registered trademark of CollabNet, Inc.



Resurrect A Rewritten Reference

Do you want to resurrect 2 selected references to their original name?

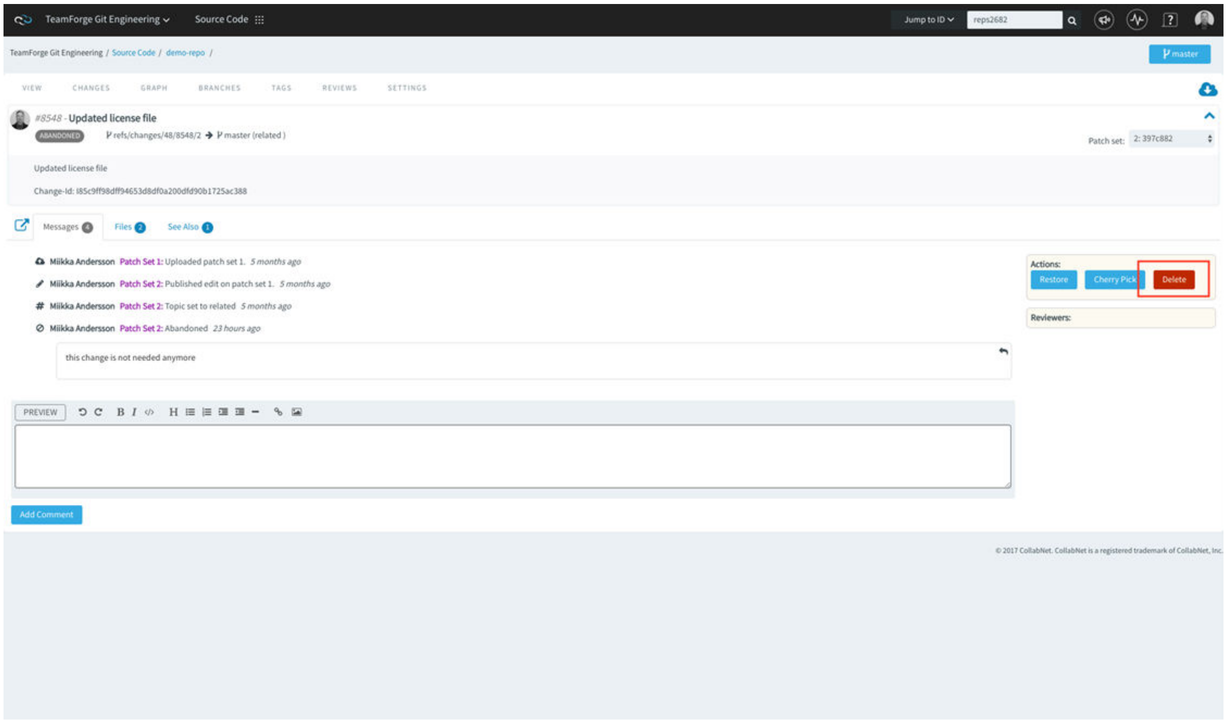
Default branch: master

93cc3e4: testing pr
Authored 06/13/2016 8:22 pm UTC (1 month ago) by John McFoobar

☐	TYPE	REFERENCE NAME	USER	REWRITE DATE	ORIGINAL COMMIT ID	NEW COMMIT ID
☑	⊘	build_update	john	03/07/2016 4:45 pm UTC	c310cc3	
☐	⊘	conflict_branch	mike	03/07/2016 8:40 pm UTC	265edc5	
☑	⊘	license_update	john	03/07/2016 7:11 pm UTC	2e42c1c	
☐		master	admin	03/07/2016 2:13 pm UTC	5194590	050bfc3

© 2016 CollabNet. CollabNet is a registered trademark of CollabNet, Inc.

- **Delete abadoned reviews**: To delete an abandoned review, open the review in the code browser and click **Delete** from *Actions*.



#8548 - Updated license file
ABANDONED refs/changes/48/8548/2 → master (related)
Patch set: 2: 397c882

Updated license file

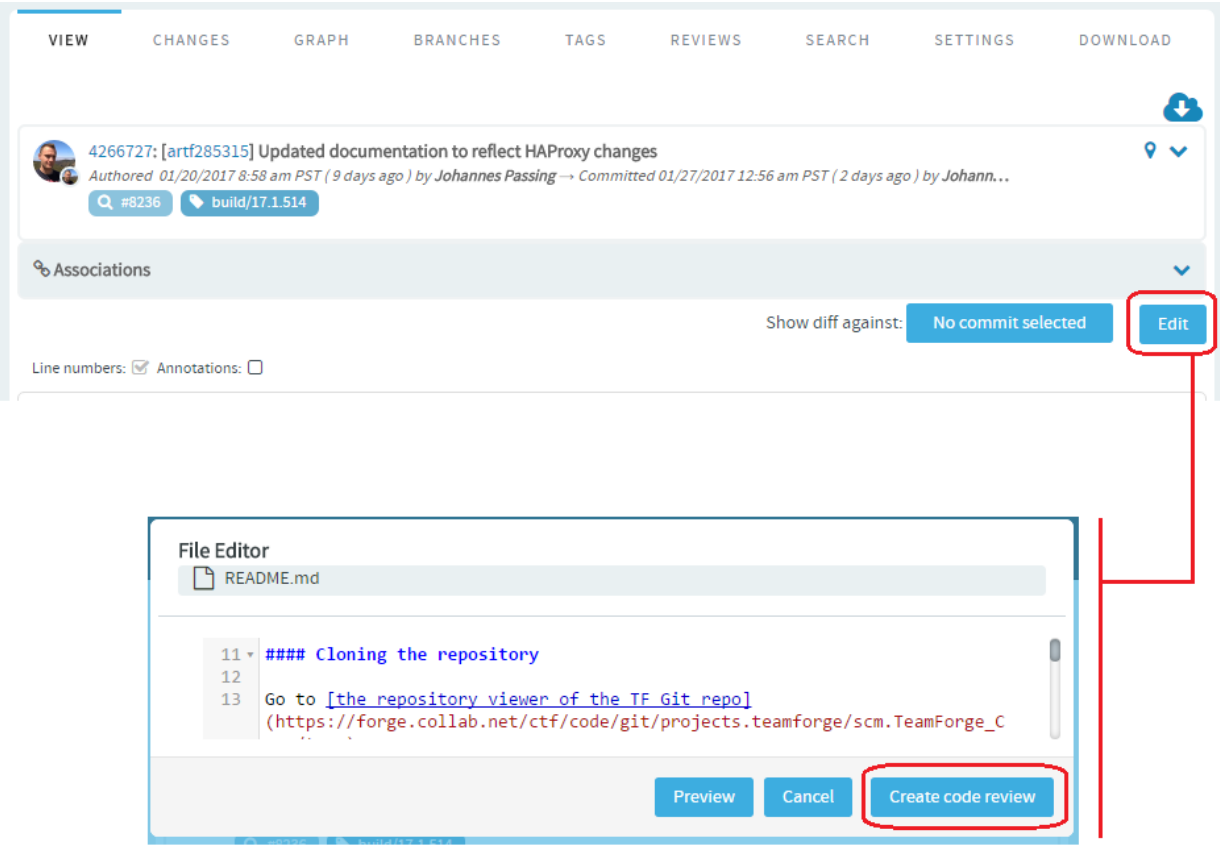
Change-Id: I85c9ff98dff94653d8df0a200dfd90b1725ac388

Miikka Andersson Patch Set 1: Uploaded patch set 1. *5 months ago*
Miikka Andersson Patch Set 2: Published edit on patch set 1. *5 months ago*
Miikka Andersson Patch Set 2: Topic set to related *5 months ago*
Miikka Andersson Patch Set 2: Abandoned *23 hours ago*

this change is not needed anymore

© 2017 CollabNet. CollabNet is a registered trademark of CollabNet, Inc.

- **Inline Editing of Files**: Quick changes to files, if required only to few files, can be done using the inline edit feature from within the code browser without having to clone an entire repository. Browse the repository, locate and open the file in the *View* tab, click **Edit** to open the file in the *File Editor*, make your changes, **Create code review** and **Publish** your changes for review.

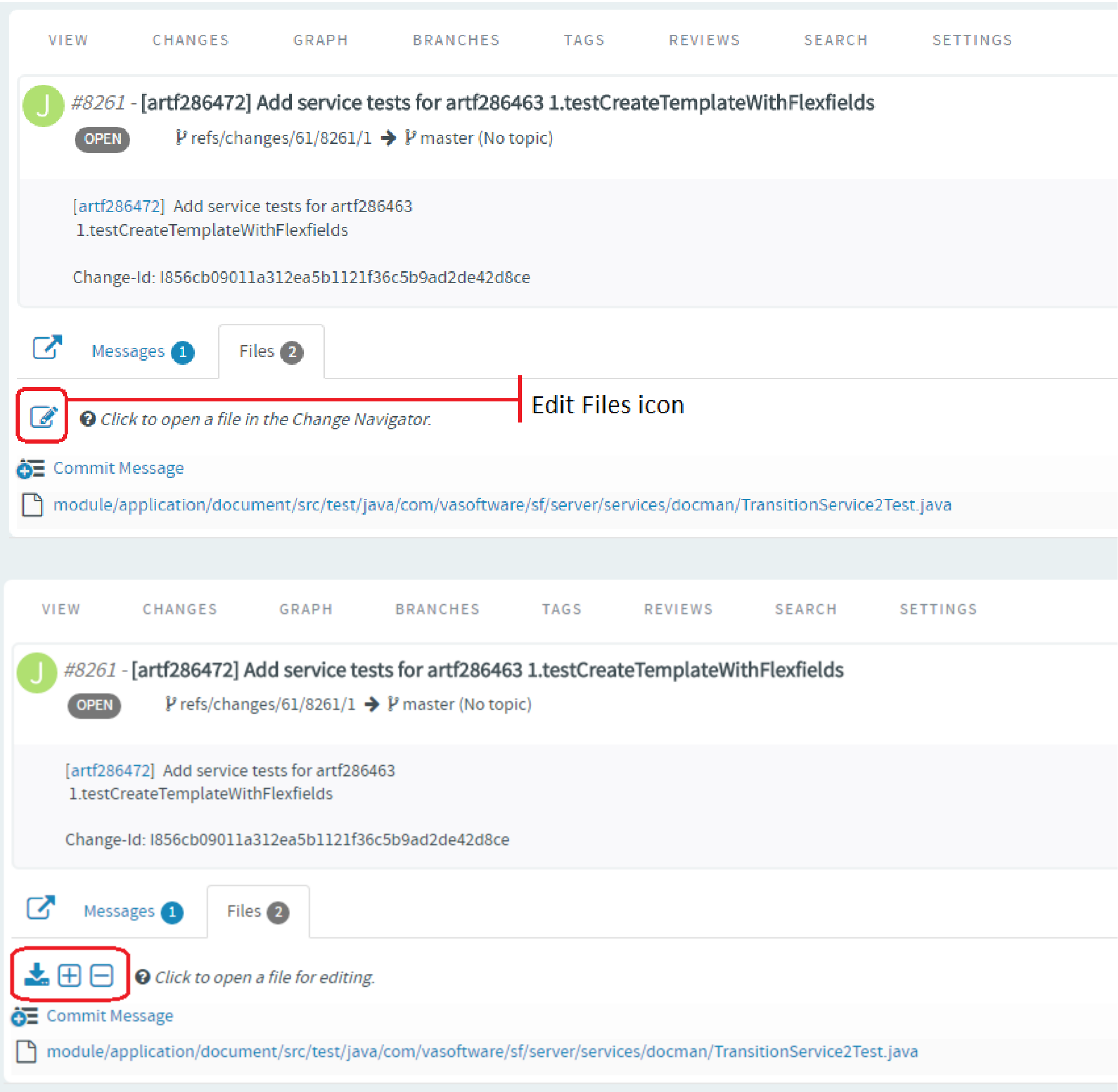


4266727: [artf285315] Updated documentation to reflect HAProxy changes

Authored 01/20/2017 8:58 am PST (9 days ago) by Johannes Passing → Committed 01/27/2017 12:56 am PST (2 days ago) by Johann...

```
#### Cloning the repository

Go to [the repository viewer of the TF Git repo]
(https://forge.collab.net/ctf/code/git/projects.teamforge/scm.TeamForge_C
```

You can add new files to a review and delete files from a review by click the **Edit Files** icon and then the “+” and “-“ icons respectively.



#8261 - [artf286472] Add service tests for artf286463 1.testCreateTemplateWithFlexfields

refs/changes/61/8261/1 → master (No topic)

[artf286472] Add service tests for artf286463 1.testCreateTemplateWithFlexfields

Change-Id: I856cb09011a312ea5b1121f36c5b9ad2de42d8ce

Click to open a file in the Change Navigator.

module/application/document/src/test/java/com/vasoftware/sf/server/services/docman/TransitionService2Test.java

#8261 - [artf286472] Add service tests for artf286463 1.testCreateTemplateWithFlexfields

refs/changes/61/8261/1 → master (No topic)

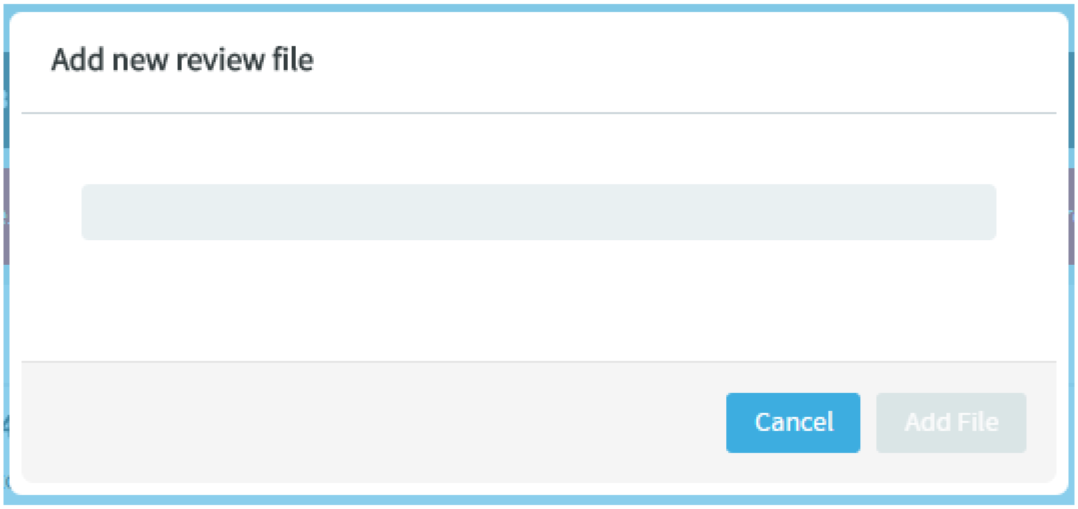
[artf286472] Add service tests for artf286463 1.testCreateTemplateWithFlexfields

Change-Id: I856cb09011a312ea5b1121f36c5b9ad2de42d8ce

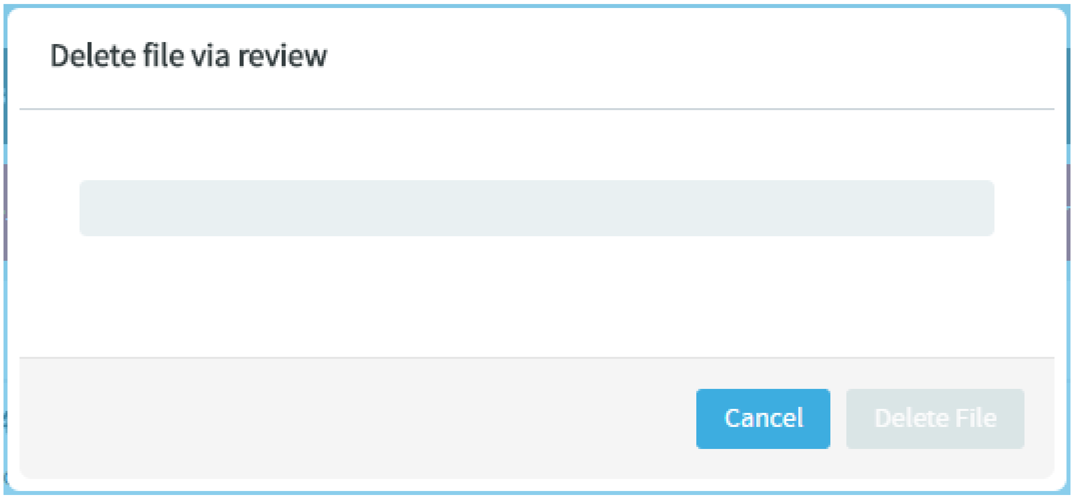
Click to open a file for editing.

module/application/document/src/test/java/com/vasoftware/sf/server/services/docman/TransitionService2Test.java

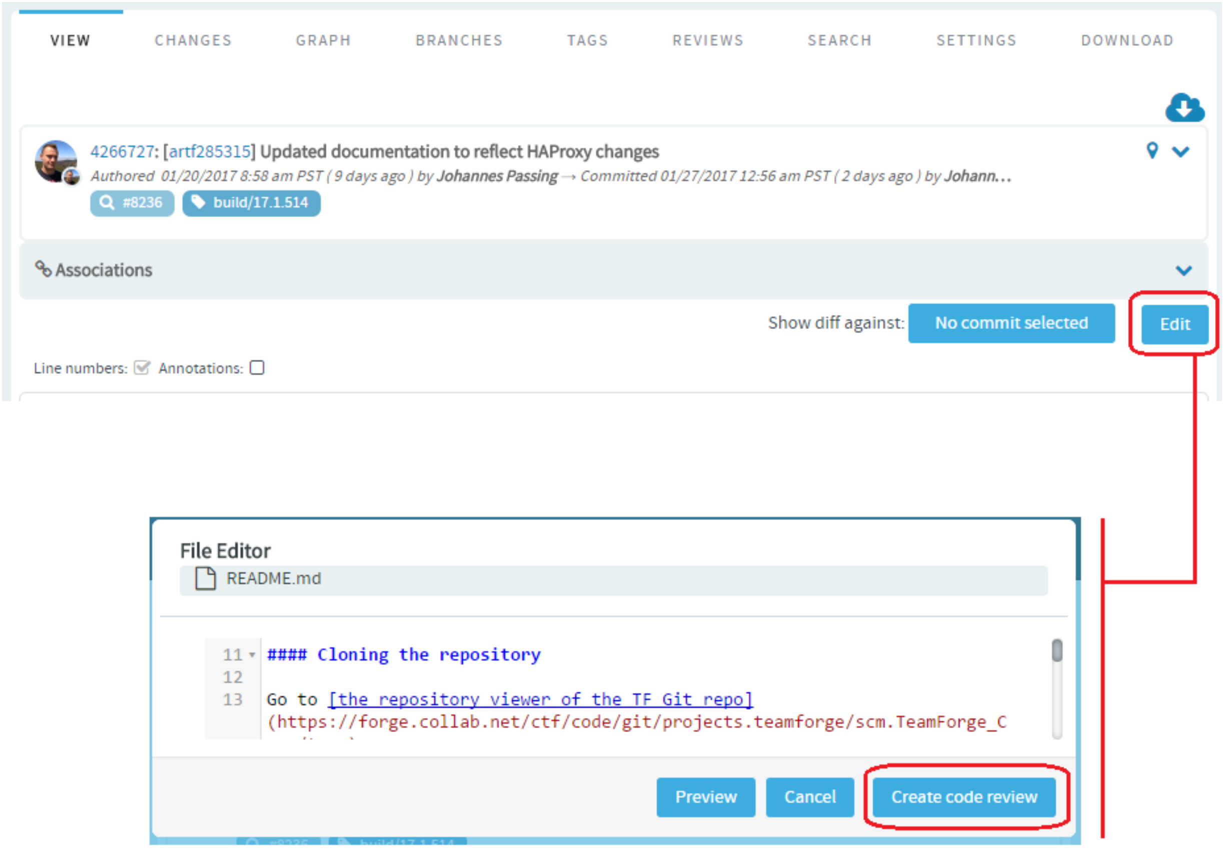
Type the name of the file to see results matching the file name, select a file and click **Add File**.



Type the name of the file you want to delete to see results matching the file name, select the file and click **Delete File**.



Click the **Complete File Edits** icon.

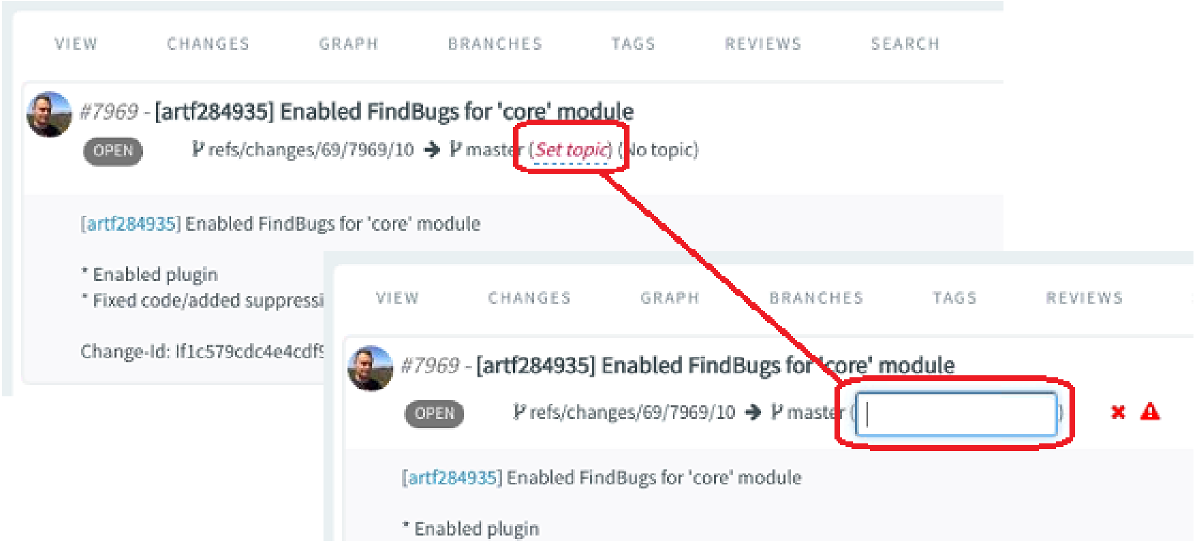


4266727: [artf285315] Updated documentation to reflect HAProxy changes
Authored 01/20/2017 8:58 am PST (9 days ago) by Johannes Passing → Committed 01/27/2017 12:56 am PST (2 days ago) by Johann...
#8236 build/17.1.514

File Editor
README.md

```
#### Cloning the repository

Go to [the repository viewer of the TF Git repo]
(https://forge.collab.net/ctf/code/git/projects.teamforge/scm.TeamForge_C
```

- **Submit Whole Topic**: You can now bundle related changes (code reviews) by topic and submit the whole topic for review instead of just submitting changes one-by-one. Just open a review, click the **Set Topic** link and enter the topic name.



#7969 - [artf284935] Enabled FindBugs for 'core' module
refs/changes/69/7969/10 → master

[artf284935] Enabled FindBugs for 'core' module

* Enabled plugin
* Fixed code/added suppressi

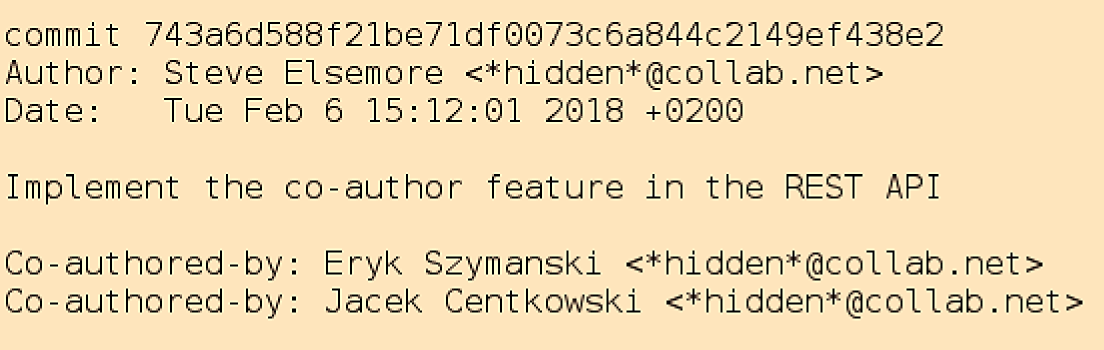
Change-Id: If1c579cdc4e4cdf

#7969 - [artf284935] Enabled FindBugs for 'core' module
refs/changes/69/7969/10 → master

[artf284935] Enabled FindBugs for 'core' module

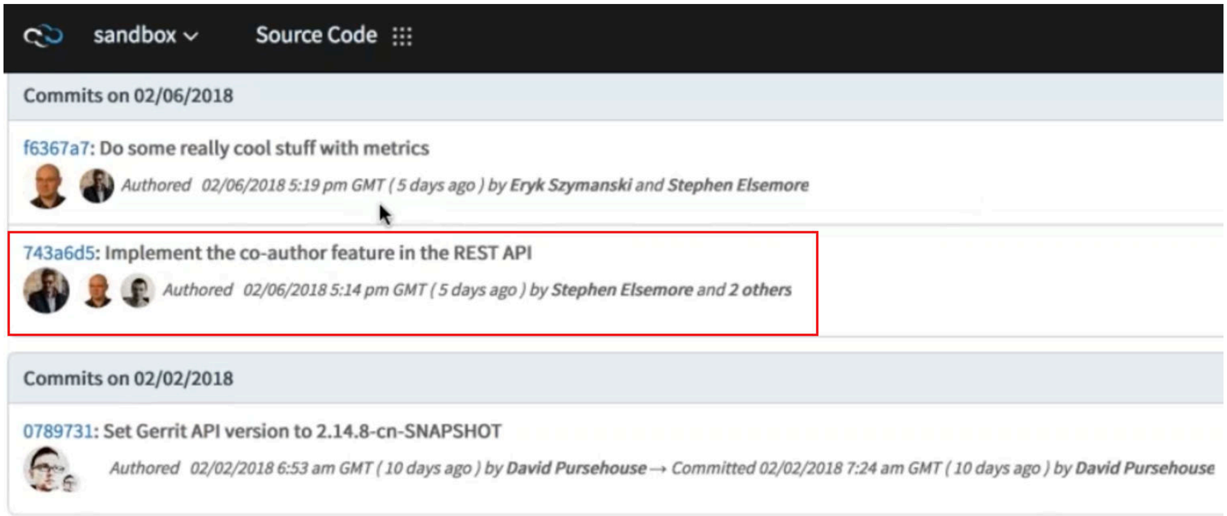
* Enabled plugin

- **Add Coauthors in Commit Message**: The coauthor name is also included as part of the author avatar and **Authored by** information on the changes list, whenever a change is done to the “Co-authored-by” footer text. More information can be seen in change details view.



```
commit 743a6d588f21be71df0073c6a844c2149ef438e2
Author: Steve Elsemore <*hidden*@collab.net>
Date:   Tue Feb 6 15:12:01 2018 +0200

Implement the co-author feature in the REST API

Co-authored-by: Eryk Szymanski <*hidden*@collab.net>
Co-authored-by: Jacek Centkowski <*hidden*@collab.net>
```

Add coauthors in the Commit Message



Commits on 02/06/2018

f6367a7: Do some really cool stuff with metrics
Authored 02/06/2018 5:19 pm GMT (5 days ago) by Eryk Szymanski and Stephen Elsemore

743a6d5: Implement the co-author feature in the REST API
Authored 02/06/2018 5:14 pm GMT (5 days ago) by Stephen Elsemore and 2 others

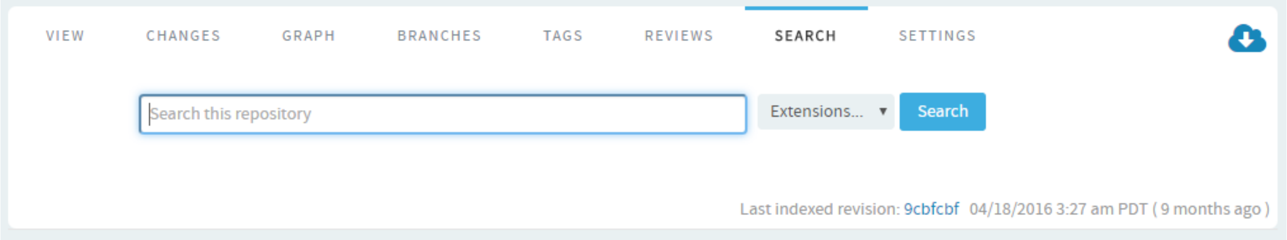
Commits on 02/02/2018

0789731: Set Gerrit API version to 2.14.8-cn-SNAPSHOT
Authored 02/02/2018 6:53 am GMT (10 days ago) by David Pursehouse → Committed 02/02/2018 7:24 am GMT (10 days ago) by David Pursehouse

List of coauthors

The Search Tab

This tab lets you search for code via TeamForge Code Search powered by Elasticsearch. You can search all files in a repository or narrow your scope to specific file types such as C, C++, C# and so on. Type your search keyword, select a file extension (optional) and click **Search**. For more information, see Search Code.



The `Settings` tab

This tab lets you configure the repository settings.

Related Links

Gerrit Code Review Policies

Each project can have one or more source code repositories. Before you can create a source code repository, a site administrator must first add one or more SCM servers to the CollabNet TeamForge environment.

1. Click **Source Code** in the project navigation bar.

2. In the list of the project repositories, click **Create Repository**.

3. On the **Create Repository** page, enter the directory name for the repository.

   > **NOTE:** For CVS repositories, the directory name is the name of the directory relative to the CVS server's repository root directory. A UNIX group by the same name is also created to enforce permissions.

4. Enter a name and description for the repository. If you plan to use an SCM server that requires approval for new repositories, use the **Description** field to provide your reason for asking to create this repository.

5. Choose the server on which you want to create the repository.

   > **NOTE:** The menu contains all of the SCM servers that the CollabNet TeamForge administrators have added to the CollabNet TeamForge environment.

6. If you want to require that each code commit be associated with an artifact (or a task or some other work item), select **Association Required on Commit**.

7. For security reasons, you may want to restrict email notifications to the essential information. If so, select **Hide Details in Monitoring Messages**.

8. Click **Save**.

Your request for a new repository is submitted. You will receive an email notification when your repository is created or if your request for a new repository is denied.

- If the SCM server that you chose does not require approval for new repositories, the repository is created.
- If the SCM server that you chose requires approval for new repositories, a CollabNet TeamForge administrator must approve your repository before it is created.

> **NOTE:** For information about implications of TeamForge EventQ integration on SCM commits and associations, see SCM Commits in TeamForge with EventQ Integration.

Related Links

Gerrit Code Review Policies

Include custom object IDs in your commit messages and have them automatically converted to hyperlinks. This is possible if you set up custom object ID mapping for the repository.
Custom object ID mapping is a process in which you define a combination of regular expression and link URL that is used to dynamically create hyperlinks of custom object IDs used in commit messages. For example, you can define a custom object ID mapping to automatically linkify objects of an external application such as JIRA tickets.

1. Log on to TeamForge and select a project from the **My Workspace** menu.

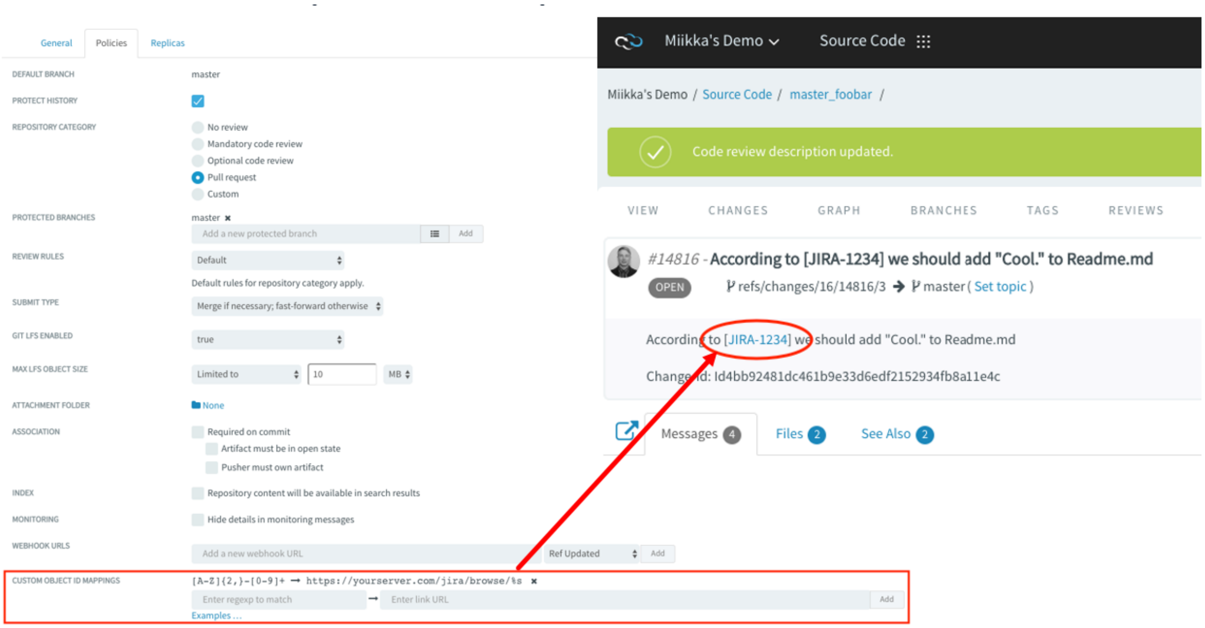
2. Click **SOURCE CODE** from the **Project Home** menu.

3. Select a repository and select the **SETTINGS** tab.

4. Select the **Policies** tab.

5. Type the regular expression and link URL in the **CUSTOM OBJECT ID MAPPING** field and click **Add**.

6. Repeat step 5 to add more such custom object ID mapping, if required.



General | Policies | Replicas

DEFAULT BRANCH: master

PROTECT HISTORY: ☑

REPOSITORY CATEGORY: ○ No review ○ Mandatory code review ○ Optional code review ● Pull request ○ Custom

PROTECTED BRANCHES: master ✖

REVIEW RULES: Default

Default rules for repository category apply.

SUBMIT TYPE: Merge if necessary; fast-forward otherwise

GIT LFS ENABLED: true

MAX LFS OBJECT SIZE: Limited to 10 MB

ATTACHMENT FOLDER: None

ASSOCIATION: □ Required on commit □ Artifact must be in open state □ Pusher must own artifact

INDEX: □ Repository content will be available in search results

MONITORING: □ Hide details in monitoring messages

WEBHOOK URLS: Ref Updated

CUSTOM OBJECT ID MAPPINGS: `[A-Z]{2,}-[0-9]+ → https://yourserver.com/jira/browse/%s` ✖

Miikka's Demo / Source Code / master_foobar /

Code review description updated.

#14816 - According to [JIRA-1234] we should add "Cool." to Readme.md

OPEN refs/changes/16/14816/3 → master (Set topic)

According to [JIRA-1234] we should add "Cool." to Readme.md

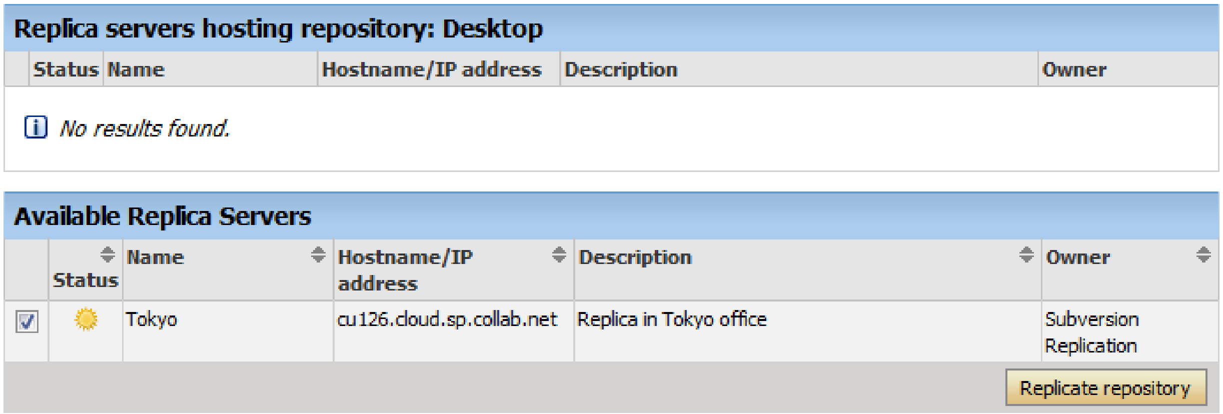
Change-Id: Id4bb92481dc461b9e33d6edf2152934fb8a11e4c

When a Subversion Edge replica has been successfully registered with a TeamForge SCM integration server, it is available to project administrators in projects using that server to house repositories. To replicate a Subversion repository, you need to add it to a replica server.

Before you can replicate a Subversion repository, an administrator must first add one or more replica servers. This involves converting a Subversion Edge server, and then approving the replica in TeamForge.

1. Click **SOURCE CODE** from the **Project Home** menu.
2. In the list of project repositories, select the one you want to replicate and click **Edit**.

   The **Edit Repository** page lists the available replica servers. Here's an example:



Replica servers hosting repository: Desktop

Status	Name	Hostname/IP address	Description	Owner
No results found.				

Available Replica Servers

	Status	Name	Hostname/IP address	Description	Owner
☑	☀	Tokyo	cu126.cloud.sp.collab.net	Replica in Tokyo office	Subversion Replication

If you don't see any available replica server listed here, it may be because none were created for this Subversion server, or there are pending replicas which haven't yet been approved by a TeamForge administrator.

3. In the Available Replica Servers section, select a replica server and click **Replicate Repository**. You will see it in the list of replica servers hosting the repository.

    - To see details such as the hostname and the user managing the replica repository, and recent replication command history, click the **Status** icon for the repository.

    - For the Subversion checkout command, see the project's **Source Code** page.

    > **NOTE:** The TeamForge account used while setting up the replica determines what's available to be checked out from a replica repository. This account - the **Managed By** user in the **Repository Replica Details** page - could have been provided total access to the master repository, or restricted access using path-based permissions.

Related Links

- Replicate a Git Repository

To stay up to date with code development on a project, browse the code commits made to each repository integrated into your TeamForge site.

For each code commit, you can view the list of files that were checked in, the version history of each file, and any associations with other TeamForge items, such as tracker artifacts or tasks.

> **NOTE:** You can see only those paths in the repository that the repository administrator has given you access to.

1. Click **SOURCE CODE** from the **Project Home** menu.

2. From the list of project repositories, select the repository you want to look at, then click **View Commits**.

3. If you have internal code browser disabled (see Integrate a Source Code Server): The **Commits** section of the **Repository Details** page lists all the code commits in the repository. By default, it shows the commits made over the preceding seven days.

    a. Specify the filter criteria in **Commit Name**, **Committed By** or **Committed On** date ranges and click **FILTER**.

b. After filtering, if you want to clear the filters, click **FILTER** and select **Clear** from the drop-down list.

c. To view the details of a commit, click its title. The **Files** section of the **Commit Details** page lists all files that were checked in with the code commit, including the version number of each file and the last operation that was performed, such as modified, deleted, moved, copied, or added.

- To view the file information, click the file name.
- To view the latest version of the file, click the file version.

d. To look at other items related to this commit, click the **ASSOCIATIONS** tab.

4. If you have the internal code browser enabled (see Integrate a Source Code Server): The **Changes** tab lists all the commits in the repository sorted by date.

   - Use the **Expand all** toggle button to expand or hide commit log messages for all the commits or selectively show or hide the commit log for the commit you are interested in.
   - You can also **browse repository from a specific commit** you are interested in.

Create associations between code commits and other CollabNet TeamForge items, such as tracker artifacts or documents, to help define relationships, track dependencies, and enforce workflow rules.

For example:

- Associate a code commit with the bugs, feature requests, or other tracker artifacts that the code addresses.
- Associate a code commit with the task requiring its completion.
- Associate a code commit with an object in an integrated application.
- Associate a code commit with a requirements document.

> **NOTE:** When you commit files to your source code repository, a source code commit notification mail is sent to users who are monitoring that source code repository. An option is provided at site level and user level to make sure whether the notification mail has to be sent or not. For more information on this, see Configure your Site's Settings.

Associate Code with Other Items While Committing

When you commit files to your source code repository, use the commit comment to quickly link your commit with one or more tracker artifacts or other TeamForge items.

Associations track the links between code and the bugs, feature requests, or other tracker artifacts that the code addresses. You also associate code commits with other TeamForge items, such as tasks or documents.

A project administrator can make associations mandatory for all code commits. When this is made mandatory, the following additional rules pertaining to code commit can also be set:

- Code commits can be performed only for open artifacts.
- To perform a code commit, the committer must be the owner of the specific artifact.

> **NOTE:** Once you enforce the above rules, validations are strictly enforced for commits against tracker artifacts only. In case you commit against any other TeamForge object, for example a wiki or a document, mere existence of the object ID ensures successful commit and association and no validations are performed against the status of the object or who it is assigned to.

You can create tracker artifact associations from whatever interface you normally use to check code into your SCM repository. You do not have to log into TeamForge.

Use the same syntax for commits to CVS and Subversion repositories.

When making a code commit, add the associate command in the commit message like this: [<item id>], such as the TeamForge tracker artifact ID or task ID.

- TeamForge item IDs are always letters followed by four or more numbers, such as `task1029` or `artf10011`.
- To associate a commit with multiple TeamForge items, separate the item IDs with commas.
- All associations are displayed in the *ASSOCIATIONS* tab of the **Commit Details** page.
- The **Comment** section lists the comments made with each commit.

> **NOTE:** To associate an object in an integrated application, use the [<prefix_objectid>] format. Each integrated application displays its prefix on moving the mouse over the application name.

> **NOTE:** When you commit files to your source code repository, a source code commit notification mail is sent to users who are monitoring that source code repository. An option is provided at site level and user level to make sure whether the notification mail has to be sent or not. For more information on this, see Configure your Site’s Settings.

> **TIP:** To remind yourself of the details of the association later, look in the *CHANGE LOG* tab of the associated **VIEW ARTIFACT** page.

Create Associations with Code That is Already Committed

At any time after a code commit is made, you can associate the code commit with other CollabNet TeamForge items, such as tasks, integrated application objects, or documents.

1. Click **SOURCE CODE** from the **Project Home** menu.
2. On the list of project repositories, select the repository containing the code commit with which you want to create an association.
3. Click **View Commits**.
4. On the **Repository Details** page, click the name of the commit with which you want to create an association.
5. On the **Commit Details** page, click the *ASSOCIATIONS* tab.
6. On the list of existing associations, click **Add**.
7. In the **Add Association Wizard** window, select the items with which you want to associate the artifact:

    - **ENTER ITEM ID** - If you know the item's ID, you can enter it directly.

      ✓ To associate an object in an integrated application from with TeamForge, use the [<prefix_objectid>] format.

      ✓ Each integrated application displays its prefix on moving the mouse over the application name in the tool bar.
    - **ADD FROM RECENTLY VIEWED** - Select one of the last ten items you looked at during this session.
    - **ADD FROM RECENTLY EDITED** - Select one of the last ten items you changed.
8. Click **Next**.
9. You may add a comment in the **ASSOCIATION COMMENT** text box.
10. Save your work.
    - Click **Finish and Add Another** to add additional associations.
    - Click **Finish** to return to the **Details** page.

> **NOTE:** When you commit files to your source code repository, a source code commit notification mail is sent to users who are monitoring that source code repository. An option is provided at site level and user level to make sure whether the notification mail has to be sent or not. For more information on this, see Configure your Site's Settings.

11. Click the *Associations* tab to view a graphical representation of all the associated items. Through the Association Viewer, you can choose to view associations in the form of a list or flip over to the Trace

view to explore the layers of associations (including parent/child dependencies) laid out in a timeline. You can scroll across the Trace view by dragging the mouse over the association layer or use the 'Previous' and 'Next' arrows to view all the objects as events in a timeline.

While the *Associations* tab shows the count of the total number of associations, you can only view the most recent 500 associations when you click the *Associations* tab in case the artifact has more than 500 associations. You can, however, browse through the Association Viewer to view older associations.

You can click on each node on the graphical association viewer to filter and display the associated items in the table below the association viewer thus matching the number of associations provided on the selected node. For example, if the node that you select for filtering is having two associations on it, the table displays the two associated items as a result of the filtering process.

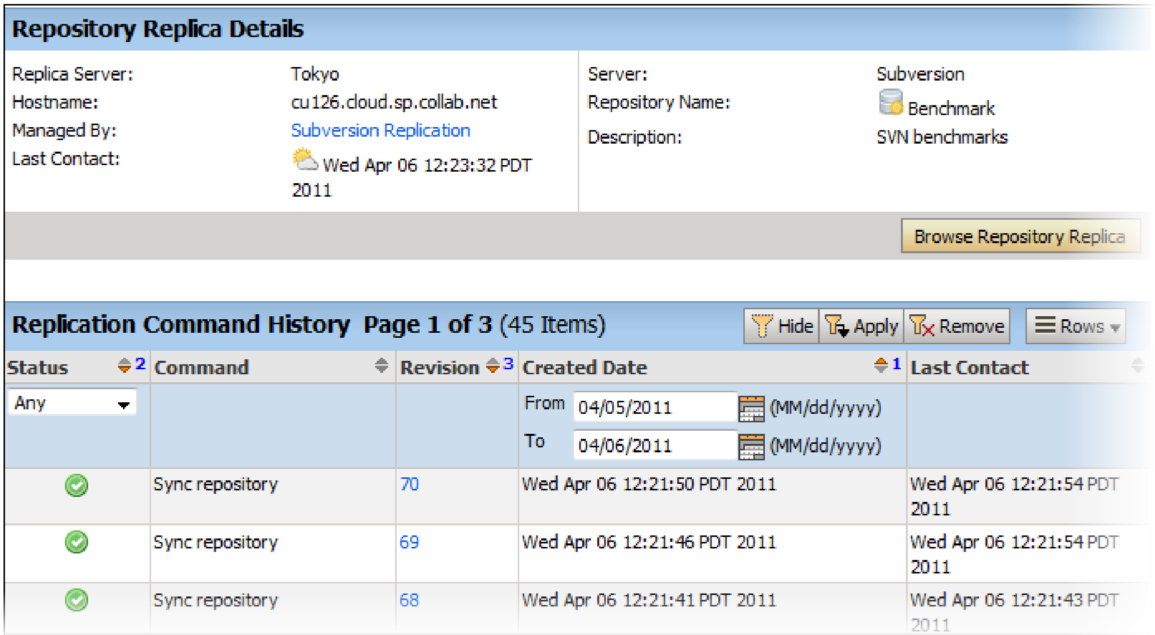
Recent command history for a replica server or a specific repository allows you to check for errors and see whether there are pending commands.

- Maybe a repository revision is not showing up. You can check for errors to know it's not just because the repository is still synchronizing.
- You can also check if there are commands waiting in the queue – you'd be able to see whether the repository is truly in sync.

- To check the command history for a replica repository, follow these steps:

  - Click **SOURCE CODE** from the **Project Home** menu.

  - In the list of repositories, click the icon for the one you're interested in. You'll see repository details and command history. Here's an example:



Repository Replica Details

Replica Server:	Tokyo	Server:	Subversion
Hostname:	cu126.cloud.sp.collab.net	Repository Name:	Benchmark
Managed By:	Subversion Replication	Description:	SVN benchmarks
Last Contact:	Wed Apr 06 12:23:32 PDT 2011		

Replication Command History Page 1 of 3 (45 Items)

Status 2	Command	Revision 3	Created Date 1	Last Contact
Any			From 04/05/2011 (MM/dd/yyyy) To 04/06/2011 (MM/dd/yyyy)	
✓	Sync repository	70	Wed Apr 06 12:21:50 PDT 2011	Wed Apr 06 12:21:54 PDT 2011
✓	Sync repository	69	Wed Apr 06 12:21:46 PDT 2011	Wed Apr 06 12:21:54 PDT 2011
✓	Sync repository	68	Wed Apr 06 12:21:41 PDT 2011	Wed Apr 06 12:21:43 PDT 2011

TIP: You can also check command history by clicking on the **Status** icon in the **Edit Repository** page.

- You need to be a TeamForge administrator to check the command history across a replica server.
  - On the site administration navigation bar, click **INTEGRATIONS**.
  - On the **SCM INTEGRATIONS** page, click the name of the replica server you’re interested in. The **Edit System** page displays the command history.

You can use the checkout command to check out the code from Subversion or GIT repository.

Check out Subversion Code Anonymously

When you want to experiment with the code, you can do an anonymous checkout from the Subversion repository. The checkout command uses a unique system-created user called “guest” and works without authentication.

To make anonymous checkout possible, the project administrator must set public and repository view permission to “All Users.” The checkout command differs based on whether a user is logged in or not.

- When you are not logged in, use this command to check out:

```
svn checkout --username guest <domain>/svn/repo URL name
```

- When you are logged in, use this command to check out:

```
svn checkout --username <logged in_username><domain>/svn/repo URL name
```

Check out GIT Code

Generally, GIT repositories can be accessed by more than one protocol. To support this, a **Protocol** drop-down list is available on the source code repositories page. This feature applies only to GIT repositories and so selecting a protocol determines the check out command for a GIT repository. This also gives the user an option to override the default protocol which is set while configuring the GIT integration server.

If you don't want to enter your password each time you access a CVS repository, you can create a set of SSH authorization keys and use the public-key method for automatic authentication.

Secure shell (SSH) can use public-key cryptography to confirm your identity, authenticate you to the remote host, and encrypt data transmission. On many TeamForge sites, access to CVS repositories is authenticated using SSH.

These instructions will help you only if all the following are true:

- You are using CVS.
- The CVSROOT (or “cvs -d”) info given you by your admin (or the CTF “Repositories” list) does not include `:pserver:`
- Your admin has set up SSH access for you.

Check with your site administrator if you are not sure about any of these conditions.

1. Generate an SSH key pair according to the instructions in the documentation for your SSH client. (Each SSH client provides its own mechanism for key pair generation.)
2. Log on to TeamForge.
3. On your **My Workspace** page, select **MY SETTINGS** from the **My Page** menu.
4. On your **User Details** page, click **AUTHORIZATION KEYS** tab.
5. On the **Authorization Keys** page, copy the public key from the `.pub` file in your SSH installation directory and paste it into the **Authorized Keys** field.
6. Click **Update**.

Your SSH public key is now saved. When you log into a CVS repository on your TeamForge site, SSH automatically checks your private key against this public key and authenticates you. You do not have to enter a password.

> **NOTE:** You can store multiple SSH public keys in TeamForge, but only one for each client machine.

Browse TeamForge to find the code you want to work on, then check out the code.

You can view the contents of each file in a repository, plus additional information about each file such as revision history, comments, date and time of submission, and branch and tag information. You can also view differences (diffs) between any two files. You can also search for code in a repository using TeamForge Code Search.

> **NOTE:** You can see only those paths in the repository that the repository administrator has given you access to.

> **NOTE:** If you're getting code from a Subversion replica repository, the TeamForge account used while setting up the replica determines what's available to be checked out. This account could have been provided total access to the master repository, or restricted access using path-based permissions.

1. Click **SOURCE CODE** from the **Project Home** menu.
2. On the list of project repositories, click the name of the repository in which you want to view code. For each file, the revision number, time since check-in, author, and last log entry appear in the **Repository Browser**.
    - To view a file, or to view the diffs between two files, click the file name.
    - To view a specific version of the file, click **Download**.
    - To view the differences between two files, do either of these:
        - Click **[select for diffs]** next to each of the two files that you want to compare.
        - Enter the file revision numbers in the Diffs between boxes at the bottom of the page.
3. If you need to diff files, choose a display from the **Type of Diff** menu, then click **Get Diffs**. The differences between the two files are displayed.
4. Use your source control client to check out the code to your local machine.

Internal Code Browser

For Subversion and Git repositories, you have the option to use the TeamForge code browser which is turned on by default while integrating the source code server. Fore more information, see Integrate a source code server.

On the list of project repositories, click the name of a Subversion or a Git repository in which you want to view code. On the top right of code browser, you can select the branch/tag (for Git) or specify the revision (for SVN) you want to browse.

- **View:** This tab allows you to do the following:
  - Browse through the folder hierarchy of the repository and view the content of specific files. For any folder or file you are viewing within a branch (Git) or revision (SVN), you can obtain the commit information pertaining to its last update.
  - While viewing a single specific commit or a file, you can see the paths that were modified in that commit, the associations including JIRA such as builds, code reviews and so on (from EventQ events) for the specific commit and the differences between files in that commit.

    > **IMPORTANT:** To view the associations, you must have installed EventQ and must have RBAC (role-based access control) permissions to use the “EventQ READ” or “Reporting API” of TeamForge EventQ. If either of these requirements is not met, this section will not show up at all.

  - While viewing a folder, if there is a file named `readme`, `readme.txt` or `readme.md` that file will automatically be rendered beneath the list of files in the folder. If the file contains markdown formatting, it will be rendered as rich text.

```
* **Changes:** This tab lets you view all of the commits that touched a specifi
c path you are browsing within a branch or revision. Click a commit to view it
s details.

* **Graph:** This tab provides a graphical representation of the changes made
including branching and merging of repositories.

* **Branches (for Git):** This allows you to see all of the branches in the re
pository in their relation to the default (master) one. Using **Compare** Bran
ch you can see the commits in the branch that do not exist in the default bran
ch.

* **Tags:** This tab lets you create Git tags and tag specific points in histor
y as being important. Typically you can use this functionality to mark release
```

points (v1.0, and so on) with an option to add Release Notes for the tagged revision. Once you create a tag, you can use it to download source code as a zip/tar file and view the tag information in Changes and Graph tabs.

  <figure><img class="docimage" src="images/171_gittags01.png" alt="" ↗↙figure>

  <figure><img class="docimage" src="images/171_tags01.png" alt="" ↗↙figure>

  <figure><img class="docimage" src="images/171_tags02.png" alt="" ↗↙figure>

* **Reviews:** This tab lists all the Open, Merged and Abandoned reviews, both Pull Requests and Gerrit single-commit reviews. Pull requests allow developers to collaborate with each other on a code change before merging it into another branch on a GIT repository. You can access this tab only when the repository owner has enabled this feature. For more information, see [Pull Request Step-by-Step][pullrequest.html#pullrequeststepbystep].

  * **Support for both Pull Requests and single-commit Gerrit Reviews:** Supports all types of code review policies, which include Pull Requests and single commit Gerrit Reviews.
    <figure><img class="docimage" src="images/171_scm_01.png" alt="" ↗↙figure>

  * **Auto refresh when a Pull Request changes:** When a pull request changes, the page is automatically refreshed to reflect the changes.
  * **Comments to support @mentions:** Inline comments are parsed for @mentions and users called out via @mentions are added as reviewers.
  * **Open your Gerrit Reviews in Gerrit's user interface:** A new button has been added to let you open your Gerrit Reviews in Gerrit's user interface.
    <figure><img class="docimage" src="images/167_scm02.png" alt="" ↗↙figure>

  * **Code commenting:** During code reviews, you can now add line comments in context while looking at the files in diff view. You can double-click to block a line/text and add a comment.
    <figure><img class="docimage" src="images/167_commenting01.png" alt="" ↗↙figure>

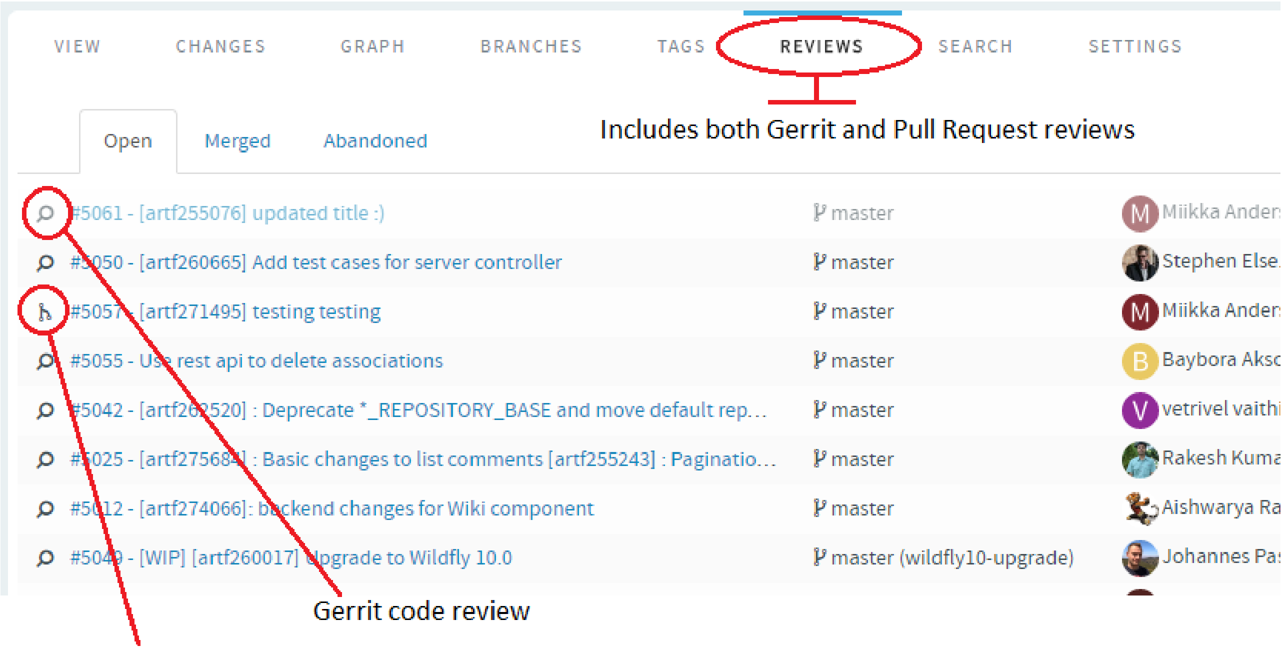
    You can also reply to line comments.
    <figure><img class="docimage" src="images/167_commenting02.png" alt="" ↗↙figure>

  * **Ability to diff the change against the Base or a previous Patch Set:** As part of the Gerrit review workflow, you now have the ability to diff the change against the Base or a previous Patch Set.
  * **Markdown support:** Markdown support for all .MD files: Render Markdown files when viewed through Code Browser.

```
    <figure><img class="docimage" src="images/167_md.png" alt=""  ↗↙figure>


    TeamForge uses Showdown—a bidirectional Markdown to HTML to Markdown conve
rter written in Javascript. For more information, see the official [Showdown Do
cumentation](https://github.com/showdownjs/showdown/wiki). Here's an abridged
version of the [Markdown syntax documentation](https://sourceforge.net/p/teamf
orge/wiki/markdown_syntax/).
  * **Mass delete/resurrect options:** Mass delete/resurrect options in Histor
y Protect tab:
    <figure><img class="docimage" src="images/167massdeleteresurrect.png" alt="
"  ↗↙figure>

    <figure><img class="docimage" src="images/167_massdeleteresurrect01.png" al
t=""  ↗↙figure>


  * **Inline editing of files:** Quick changes to files, if required only to few
 files, can be done using the inline edit feature from within the code browser
without having to clone an entire repository. Browse the repository, locate an
d open the file in the View tab, click **Edit** to open the file in the File Edi
tor, make your changes, **Create code review** and **Publish** your changes fo
r review.
    <figure><img class="docimage" src="images/171_inlinedit01.png" alt=""  ↗↙fig
ure>


  * **Submit whole topic:** You can now bundle related changes (code reviews)
by topic and submit the whole topic for review instead of just submitting chan
ges one-by-one. Just open a review, click the **Set Topic** link and enter the
 topic name.
    <figure><img class="docimage" src="images/171_submitwholetopic.png" alt=""
 ↗↙figure>



* **Search:** This tab lets you search for code via TeamForge Code Search powe
red by Elasticsearch. You can search all files in a repository or narrow your s
cope to specific file types such as C, C++, C# and so on. Type your search keywo
rd, select a file extension (optional) and click Search. For more information,
see [Search Code][searchcode].
  <figure><img class="docimage" src="images/codesearch01.png" alt=""  ↗↙figure>


* **Settings:** This tab lets you configure the repository settings.
* **Download:** This tab allows you to download a copy of the required file.
```

TeamForge provides a unified code review experience as it supports both Pull Request and Gerrit single-commit reviews.

Git repositories are hosted and service in TeamForge via Gerrit. Gerrit is most widely known for providing powerful code review features. While Gerrit includes a powerful code review feature, the way it works and the workflow is different from the Pull Request style that was introduced by GitHub. A Google search of “Gerrit Pull Request” will yield a bounty of passionate viewpoints on these differences. Hashing through the pros and cons would just add another result to that search. Instead, just know that with TeamForge, you are free to use either style of code review methodology, even on the same Git repository.



Includes both Gerrit and Pull Request reviews

#5061 - [artf255076] updated title :)	master	Miikka Ander
#5050 - [artf260665] Add test cases for server controller	master	Stephen Else
#5057 - [artf271495] testing testing	master	Miikka Ander
#5055 - Use rest api to delete associations	master	Baybora Aksc
#5042 - [artf262520] : Deprecate *_REPOSITORY_BASE and move default rep...	master	vetrivel vaithi
#5025 - [artf275684] : Basic changes to list comments [artf255243] : Paginatio...	master	Rakesh Kuma
#5012 - [artf274066]: backend changes for Wiki component	master	Aishwarya Ra
#5049 - [WIP] [artf260017] upgrade to Wildfly 10.0	master (wildfly10-upgrade)	Johannes Pa

Gerrit code review

Pull Request review

While TeamForge supports both pull request and single-commit Gerrit reviews, this topic focuses more on the Pull Request type reviews.

Pull Request Configuration

In order to use pull requests in your Git repository, there is some configuration that must be done first. This is to set up proper permissions in your repository so that your policies are being followed.

Open the repository in the code browser, click the *Settings* tab, thenclick *Policies* tab. This is where you configure the pull request-based code review policy. For more information, see Configure Pull Request for Repositories: Step by Step.

Repository Category and **Protected Branches**: A new category named “Pull requests” has been added. What this category does is set up the repository permissions so that users can create and push to feature branches but require pull requests to certain “protected branches”. Once you change the repository category to “Pull request”, the **Protected Branches** field shows up. This will be the list of branches that you will be merging your pull requests into. Typically, this would be the “master” branch but you may also have various “release” branches that you would like to protect. Users will not be able to directly push changes to these protected branches. Instead, the user will create a feature branch with their changes in it, and then create a pull request when they are ready for their changes to be reviewed and merged to the protected branch.

Review Rules: These review rules govern the requirements for a given change to be eligible to be merged. There are four new policies available:

- **No Approval Required**: This is similar to the policy on sites like GitHub. This basically means anyone with the proper TeamForge permission can accept and merge any pull request. This means you probably favor "social" policies and trust that your reviewers will do the right thing. Pull requests become a tool to aid with code review and it is still possible for users to use the voting tools in the review to communicate their feedback but the votes on a review do not prevent the review from being merged.

- **Code Review Required**: With this policy, the voting tools begin to matter. The change cannot be merged until it has a net positive vote total, not counting the owner of the review. In other words, if two users give a thumps up and one a thumbs down, then it will be eligible to be merged – assuming the owner of the review is not one of the two thumbs up. The owner can vote, but their votes do not count towards the total.

- **CI Required**: The assumption here is that the relevant votes are being cast by a "bot" or "process" such as a Jenkins CI job. There is no UI in the pull request provided to cast these votes, it will be done via API or by using the Gerrit UI. The pull request shows a check mark when a positive Verified vote has been cast and an X when a negative vote has been cast. With this policy, the change cannot be merged unless there is at least one positive verified vote and no negative votes. Users can still provide thumbs up and down votes but they do not control whether or not the change is eligible to be merged. Of course, the person that decides to merge the change can still factor in the code review votes and comments.

- **Code Review and CI Required**: This is obviously just a combination of the two previous policies. So a positive Verified vote is necessary, with no negative verify votes, and a total positive Code Review vote is required.

- **Default**: The final policy is to just use the Gerrit code review default policy. This requires a +2 code review vote and a +1 verified vote and there cannot be a -2 code review vote as that acts as a veto. Users that are not familiar with Gerrit tend to find these voting rules confusing. For example, two +1 votes does not equal a +2 vote and your permssions determine what votes you are able to cast. We do not recommend you use this policy if you are using pull requests, but it is an available option and might be desired if you are already using Gerrit reviews and do not want to change the voting rules.

Pull Request Workflow Walkthrough

The primary difference between the pull request workflow and the normal Gerrit change-based model is the use of branches. In the pull request model, the assumption is that work will being on a feature branch and you create a pull request when you are ready to start receiving feedback on the branch. This could mean the work is ready to be merged, but it could also mean that you just want to get feedback from the CI system or initial feedback from code review. Once all feedback and review is complete and the pull request is eligible to be merged, then the request can be merged and the feature branch deleted.

Create feature branch (locally)

If working in a small team, you might want to create the feature branch on the server or create one locally and push to the server right away. For now, we will assume that ther is just a single developer. Typically, it is best to just begin the process by creating a feature branch locally. It is a good idea to fetch all changes from the server before beginning this process:

```
$ git checkout master && git pull origin master
$ git checkout -b feature_branch
```

Give your feature branch a meaningful name.

Commit to feature branch and push to server

The next steps are of course to just do your work and commit changes locally. Before you have pushed the changes to the server, it is OK to do things like squashing your commits or rebase your branch on master, but once you have pushed your branch to the server, you should no longer do this. The first time you push to the server, you will need to set the upstream branch to the name you want to give your feature branch on the server.

```
git push --set-upstream origin feature_branch
```

Create pull request

When you are ready to merge the change, or at least to start getting feedback, you should create a pull request, add reviewers and have reviewers share feedback on your changes. See Create Pull Request: Step by Step for more information.

Pull requests are implemented as merge commits between your feature branch and the target branch. The pull request subject and description will combine to form the commit message for the merge commit. You can also provide a Markdown summary of the change that will be captured as the first comment on the pull request. If no summary is provided, then the commit message will serve as the first comment. If you have automated CI configured, then it will typically run as soon as the pull request is created and whenever it is updated. So you could also create a pull request early in your process so that you can benefit from the feedback of your CI system. If youare posting a pull request that is not ready to be merged, it is a good convention to follow to cast a Thumbs Down vote in the pull request to signify this to potential reviewers.

- **Reviewers**: Reviewers are anyone that you potentially want to provide feedback on the change. Adding a user does not require the user to review the change, it just notifies th euser of its existence and see it in lists and filters of reviews they are assigned to. Likewise, users that have not been added to the review are still free to cast votes on the review.

- **Voting**: Tools are provided to cast votes on the review. If you are using one of the four code review policies provided with TeamForge, you will see simple Thumbs Up/Down buttons to cast your vote.

Commit and Push More Changes

As you continue to work on your feature branch, you can just commit changes to your local branch and push them to the server. If you are working in a team on the same branch, then you will need to fetch and rebase changes made by other team members to the feature branch.

```
$ git add/commit etc.
$ git push
```

Update Feature Branch and Pull Request

If new changes are pushed to the feature branch, the pull request must be updated to include the new changes. This involves updating the merge commit to recognize the new HEAD of the feature branch. To update the pull request, you simply need to open it in the Web UI. It will then update itself as needed.

> **NOTE:** When a pull request is updated, all existing votes will be reset and need to be cast again based on the new review.

Resolve Conflicts

When working in a feature branch, it is not uncomment for conflicts to arise between your feature branch and the target. When this happens, you will not be able to merge the pull request and you will see a warning of the conflict in the web UI. To resolve the conflicts, you must fetch and merge the changes from the target branch into your feature branch and resolve and commit the conflict resolution. Then push the result back to your feature branch on the server and update the pull request.

```
$ git fetch $ git merge origin/master
$ git add/commit etc. $ git push
```

Of course, you can also rebase and force push to update your feature branch as long as you understand the ramifications of this when collaborating with a team that is sharing the same branch.

Merge Pull Request

Once a pull request is eligible to be merged, meaning there are no merge conflicts with the target and all voting requirements have been satisfied, the **Merge** button will be enabled in the web UI. Anyone who can see this button has the permissions to merge the pull request and just needs to click the button to merge it. See Merge pull requests: Step by Step for more information. This ends the life cycle for this pull request and if the user has the necessary permissions, they will also be given the ability to delete the feature branch from the server.

It is possible to continue to use the feature branch and create new pull requests to merge subsequent changes, but this is not recommended. It is generally a good idea to delete feature branches once they have been merged.

Pull Request: Step-by-Step

Pull requests allow developers to collaborate with each other on a code change begore merging it into another branch on a Git repository.

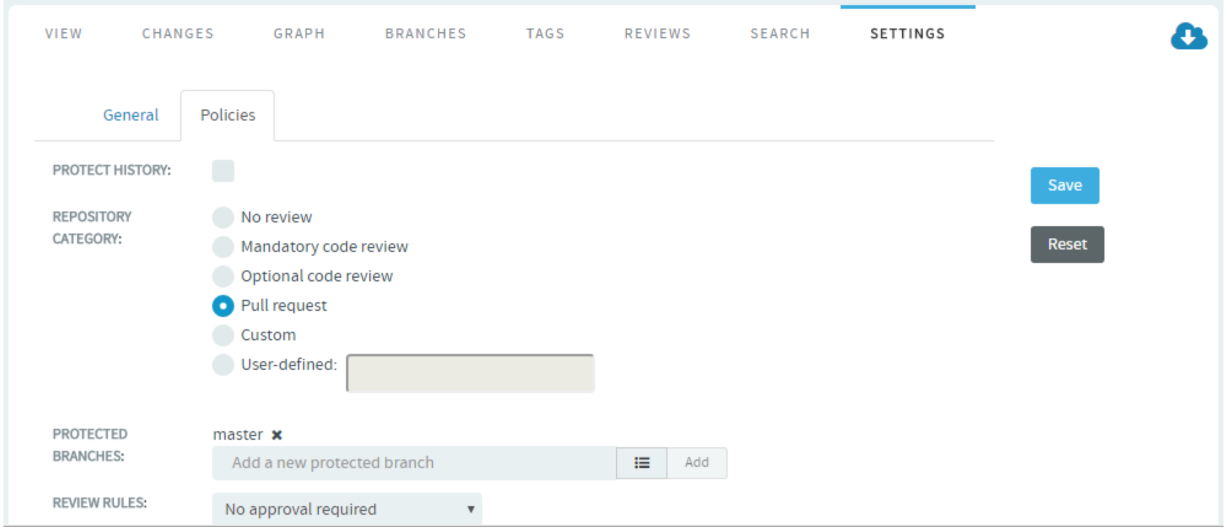
Pull request is a fully integrated solution of the code browser component. It supports all the basic functionality such as creating, viewing, updating, abandoning, rebasing and merging of pull requests. Using a pull request, you notify others about a feature or fix change that needs attention.

> **IMPORTANT:** You can access the pull request feature only when the repository owner enables the feature and sets the code review policy on the **Settings > Policies** tab.

Configure “Pull Request” for Repositories

In order to use pull requests in your Git repositry, you need to set up proper permissions in your repository so that your policies are being followed. Configure the repository from the *Settings* tab.

1. Click **SOURCE CODE** from the **Project Home** menu.
2. Browse and open the Git repositoy in the code browser.
3. Select **Settings > Policies**.
4. Select **Pull request** for **Repository Category**. The **Protected Branches** and **Review Rules** fields show up.



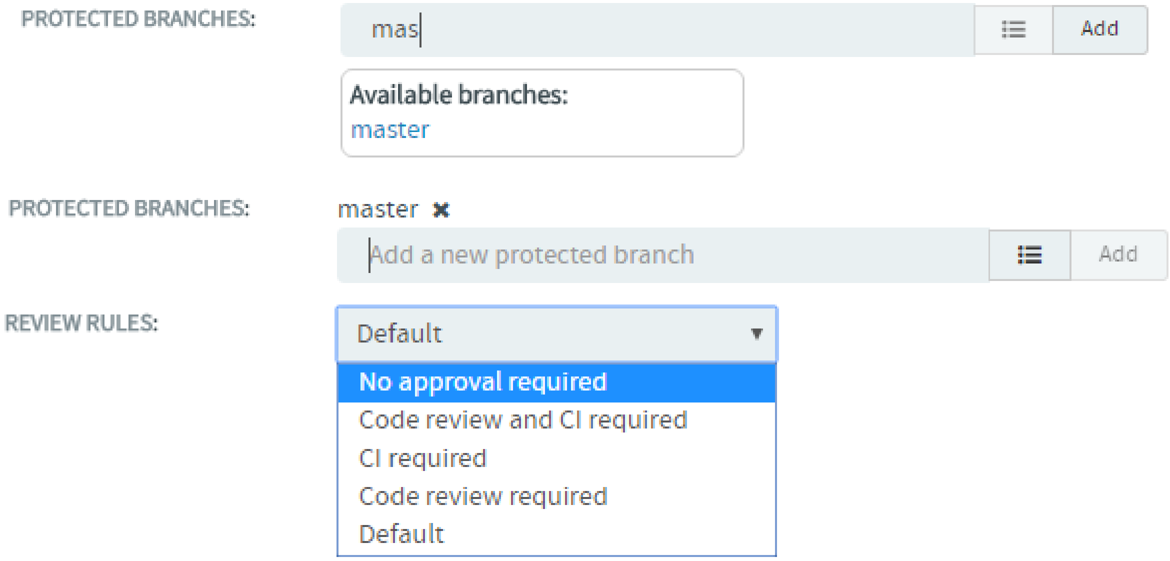
PROTECT HISTORY: □

REPOSITORY CATEGORY:
○ No review
○ Mandatory code review
○ Optional code review
● Pull request
○ Custom
○ User-defined:

PROTECTED BRANCHES: master ✖
Add a new protected branch | Add

REVIEW RULES: No approval required ▾

Add one or more protected branches. Type the branch name, select the branch and click **Add**.



5. Click **Save**.

Create a Pull Request

When you are ready to merge the change, or at least to start getting feedback, you should create a pull request. You can do this easily from the **Branches** tab by clicking on the **Create** button for your feature branch.

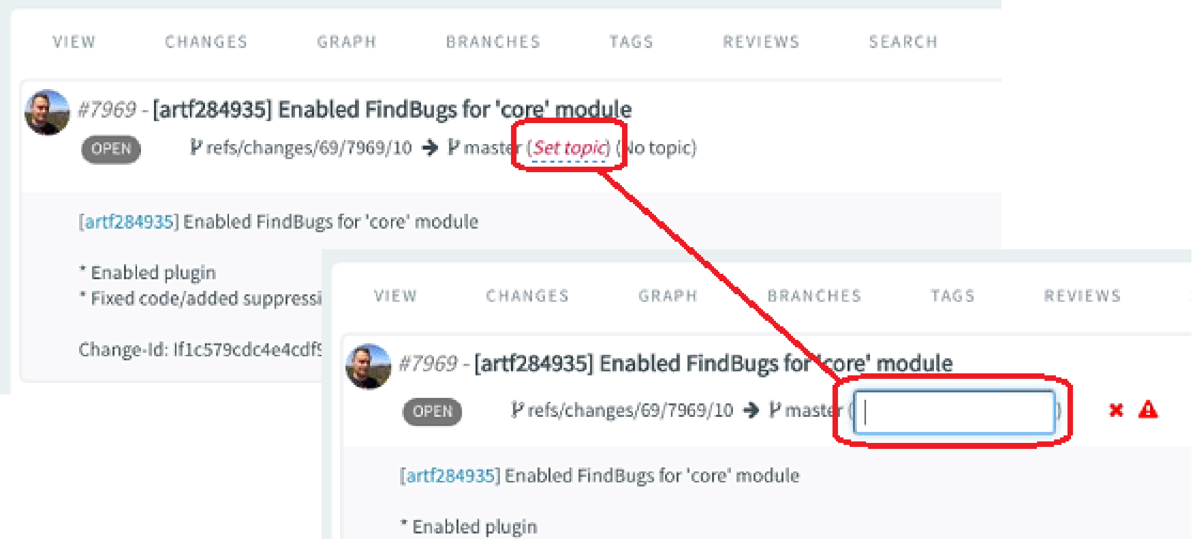
1. Go to the **Branches** tab on a Git repository page.
2. Click **Create**.
3. Select the source branch which is wanted to be merged.
4. Select the target branch to which you want the changes to be merged.
5. Give an appropriate subject line and description that will be used as a commit message for a merged pull request.

   Optionally, you can provide a summary of the pull request. This supports markdown formatting.

   The **Commits** tab at the bottom of the page displays the list of commits made in the selected source branch. The **Files** tab shows the difference between the source and target branch.

6. Click **Create Pull Request**.

7. **Submit whole topic**: You can now bundle related changes (code reviews) by topic and submit the whole topic for review instead of just submitting changes one-by-one. Just open a review, click the **Set Topic** link and enter the topic name.



#7969 - [artf284935] Enabled FindBugs for 'core' module

refs/changes/69/7969/10 → master (Set topic) (No topic)

[artf284935] Enabled FindBugs for 'core' module

* Enabled plugin
* Fixed code/added suppressi

Change-Id: If1c579cdc4e4cdf5

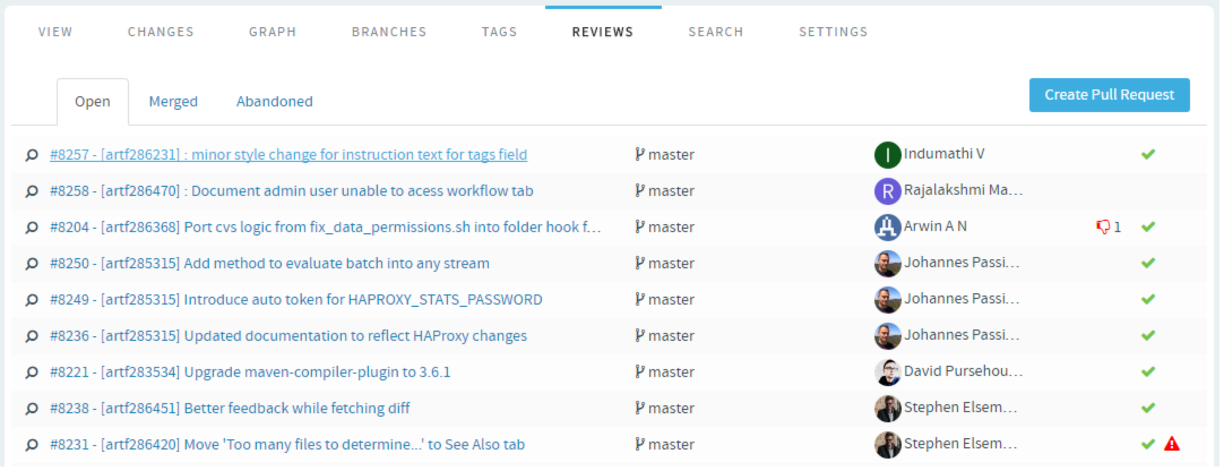
#7969 - [artf284935] Enabled FindBugs for 'core' module

refs/changes/69/7969/10 → master ( )

[artf284935] Enabled FindBugs for 'core' module

* Enabled plugin

8. As a reviewer, click the pull request that you want to review from the list of open pull requests on the **Reviews** tab.

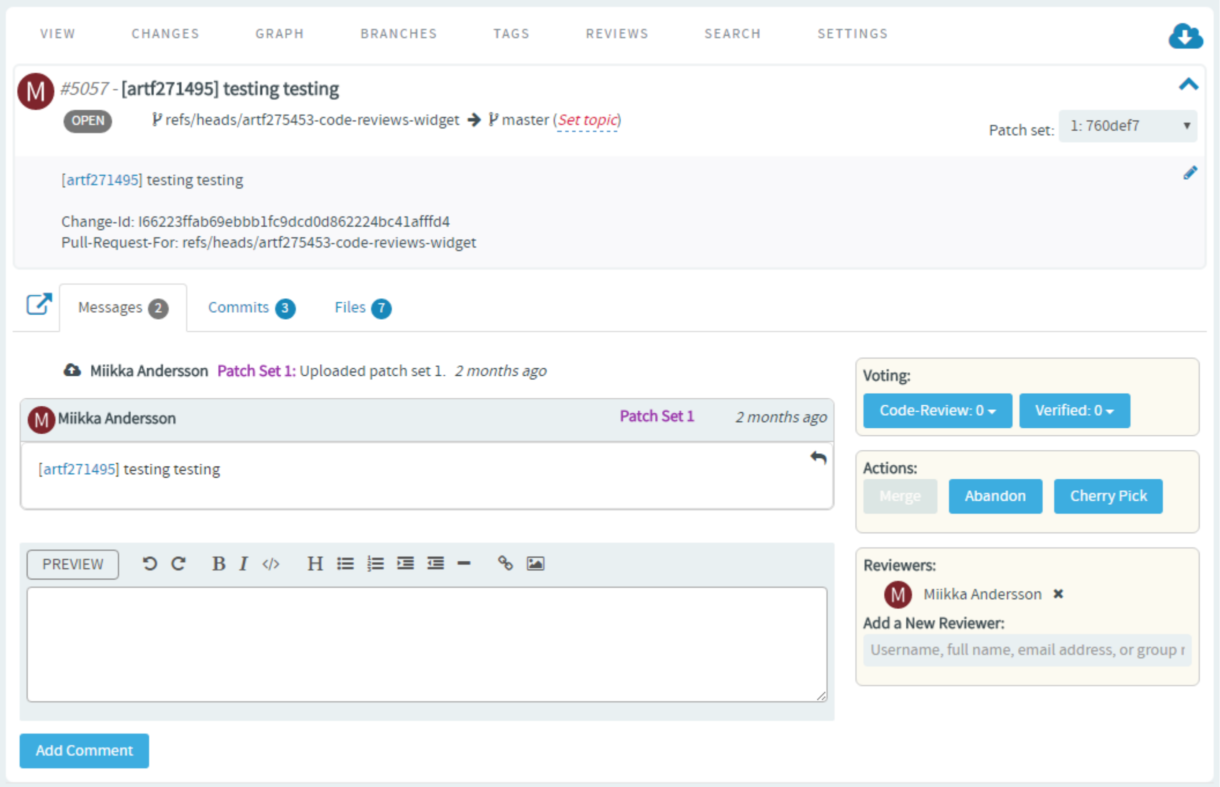


#8257 - [artf286231] : minor style change for instruction text for tags field	master	Indumathi V	✓
#8258 - [artf286470] : Document admin user unable to acess workflow tab	master	Rajalakshmi Ma...	
#8204 - [artf286368] Port cvs logic from fix_data_permissions.sh into folder hook f...	master	Arwin A N	1 ✓
#8250 - [artf285315] Add method to evaluate batch into any stream	master	Johannes Passi...	✓
#8249 - [artf285315] Introduce auto token for HAPROXY_STATS_PASSWORD	master	Johannes Passi...	✓
#8236 - [artf285315] Updated documentation to reflect HAProxy changes	master	Johannes Passi...	✓
#8221 - [artf283534] Upgrade maven-compiler-plugin to 3.6.1	master	David Pursehou...	✓
#8238 - [artf286451] Better feedback while fetching diff	master	Stephen Elsem...	✓
#8231 - [artf286420] Move 'Too many files to determine...' to See Also tab	master	Stephen Elsem...	✓ ⚠

Review a Pull Request

In addition to the requested reviewers, anyone with access to the repository and who wishes to comment on the pull request can review and post their comments on the pull request details page.

1. On the pull request details page, you can switch between three views: *Messages*, *Commits* and *Files*. Click the *Commits* tab to view the list of commits. Click the *Flies* tab to review the code changes made in each file. You have the option to view the difference between the source and target branches.



#5057 - [artf271495] testing testing

OPEN refs/heads/artf275453-code-reviews-widget → master (Set topic)

Patch set: 1: 760def7

[artf271495] testing testing

Change-Id: I66223ffab69ebbb1fc9dcd0d862224bc41afffd4
Pull-Request-For: refs/heads/artf275453-code-reviews-widget

Miikka Andersson Patch Set 1: Uploaded patch set 1. 2 months ago

Miikka Andersson Patch Set 1 2 months ago

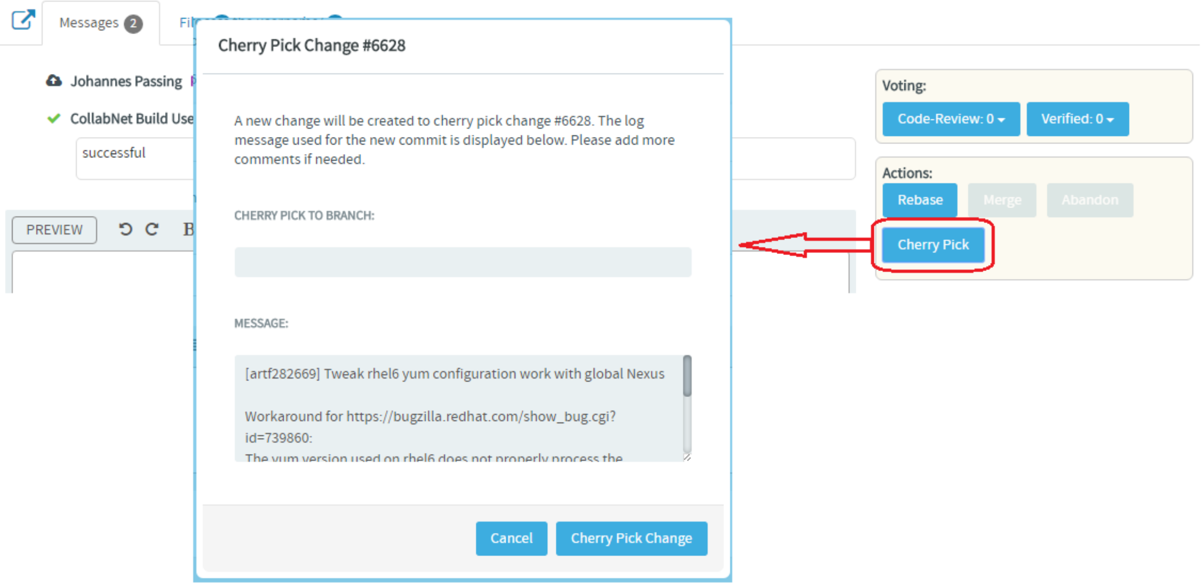
[artf271495] testing testing

Reviewers:
Miikka Andersson
Add a New Reviewer:
Username, full name, email address, or group r

2. Once you have reviewed the changes, on the *Messages* tab, enter your comments and give an appropriate voting as well.

> **NOTE:** The message section supports markdown formatting with a preview option.

3. **Cherry Pick: Apply the changes introduced by existing commits**: You can also cherry pick and apply changes introduced by existing commits to another branch. For example, you can now use this Cherry Pick function in TeamForge’s native code browser to apply a commit in master to a release branch.



Cherry Pick Change #6628

A new change will be created to cherry pick change #6628. The log message used for the new commit is displayed below. Please add more comments if needed.

CHERRY PICK TO BRANCH:

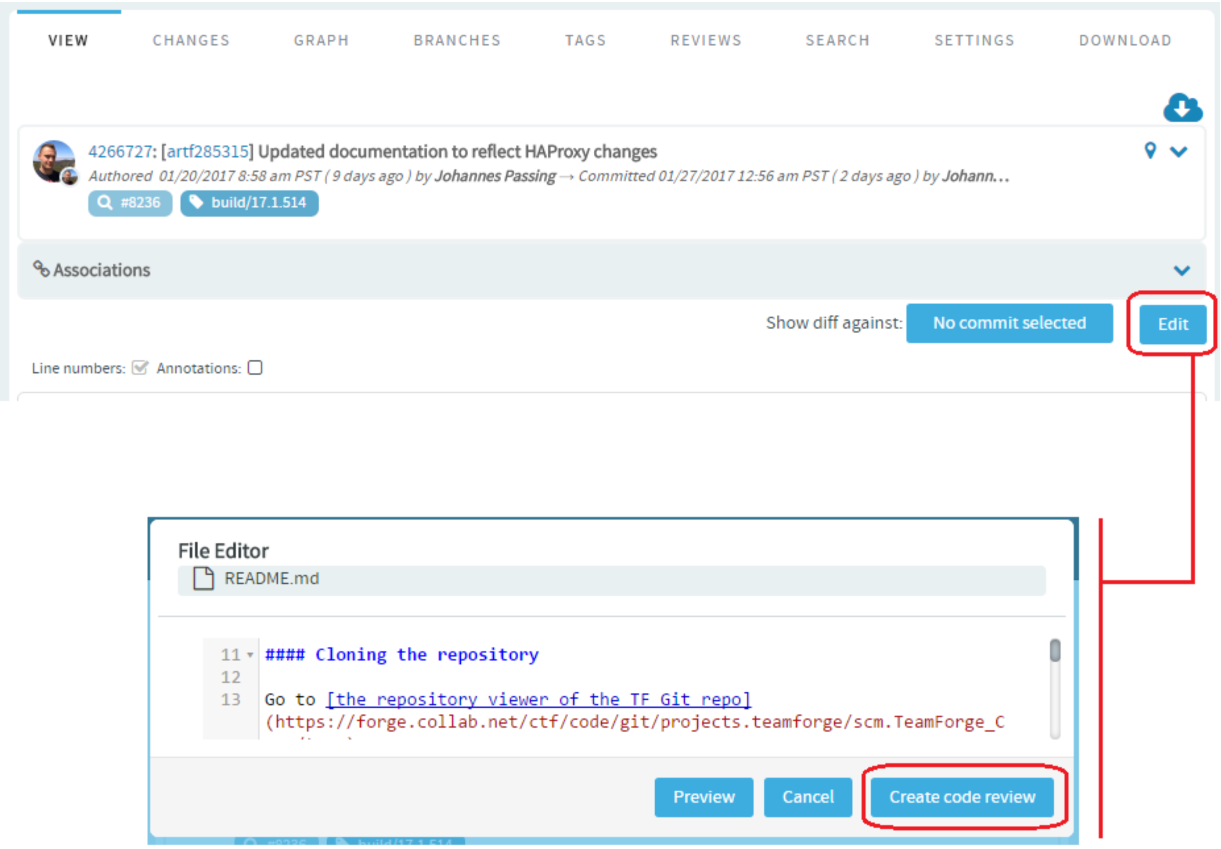
MESSAGE:

[artf282669] Tweak rhel6 yum configuration work with global Nexus

Workaround for https://bugzilla.redhat.com/show_bug.cgi?id=739860:
The yum version used on rhel6 does not properly process the

Inline Editing of Files

1. Quick changes to files, if required only to few files, can be done using the inline edit feature from within the code browser without having to clone an entire repository. Browse the repository, locate and open the file in the **View** tab, click **Edit** to open the file in the **File Editor**, make your changes, **Create code review** and **Publish** your changes for review.



4266727: [artf285315] Updated documentation to reflect HAProxy changes

Authored 01/20/2017 8:58 am PST (9 days ago) by Johannes Passing → Committed 01/27/2017 12:56 am PST (2 days ago) by Johann...

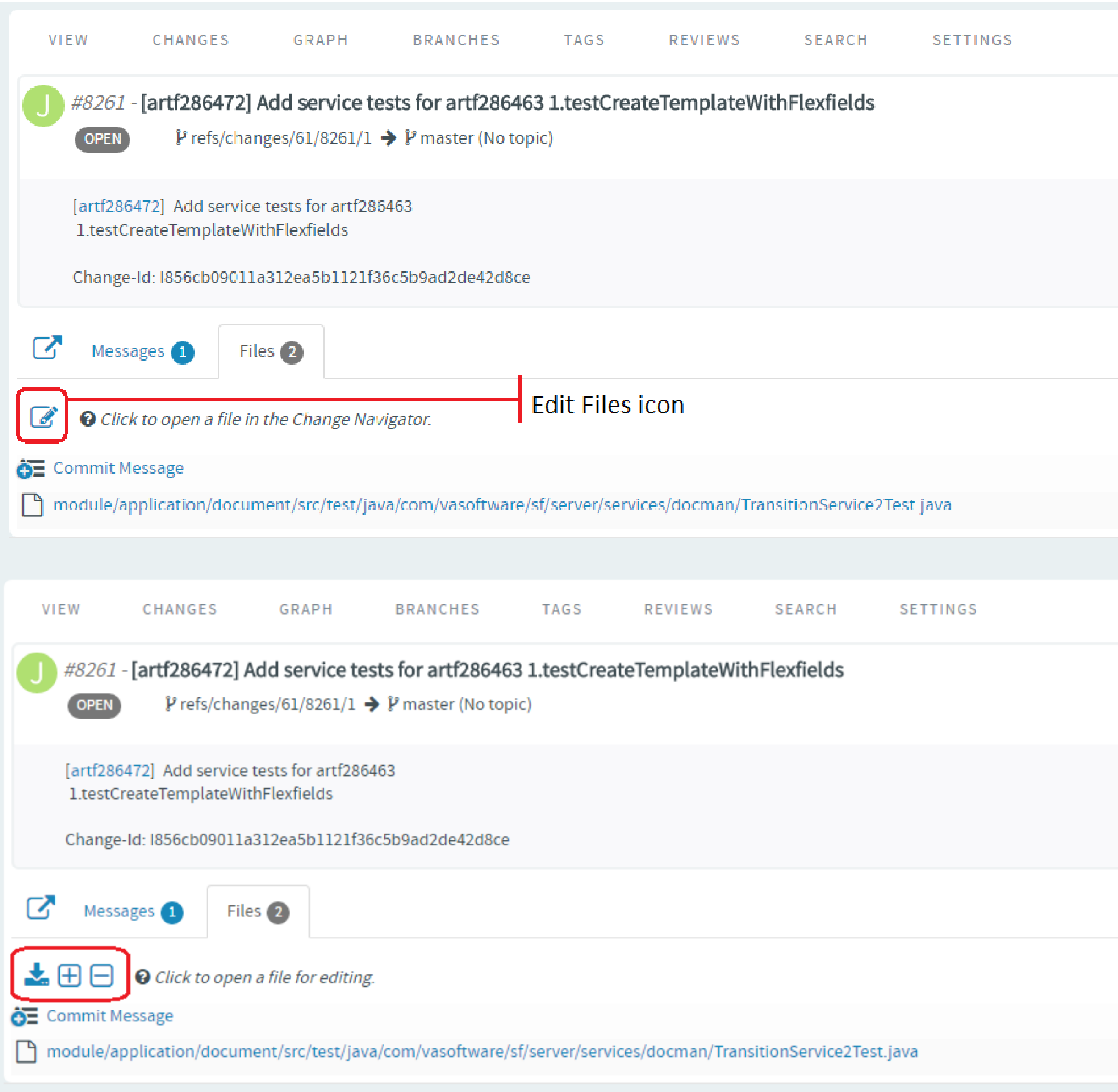
Line numbers: ☑ Annotations: ☐

File Editor

README.md

```
#### Cloning the repository

Go to [the repository viewer of the TF Git repo]
(https://forge.collab.net/ctf/code/git/projects.teamforge/scm.TeamForge_C
```

2. You can also add new files to a review and delete files from a review by clicking the **Edit Files** icon and then the “+” and “-“ icons respectively.



#8261 - [artf286472] Add service tests for artf286463 1.testCreateTemplateWithFlexfields

refs/changes/61/8261/1 → master (No topic)

[artf286472] Add service tests for artf286463
1.testCreateTemplateWithFlexfields

Change-Id: I856cb09011a312ea5b1121f36c5b9ad2de42d8ce

Click to open a file in the Change Navigator.

module/application/document/src/test/java/com/vasoftware/sf/server/services/docman/TransitionService2Test.java

#8261 - [artf286472] Add service tests for artf286463 1.testCreateTemplateWithFlexfields

refs/changes/61/8261/1 → master (No topic)

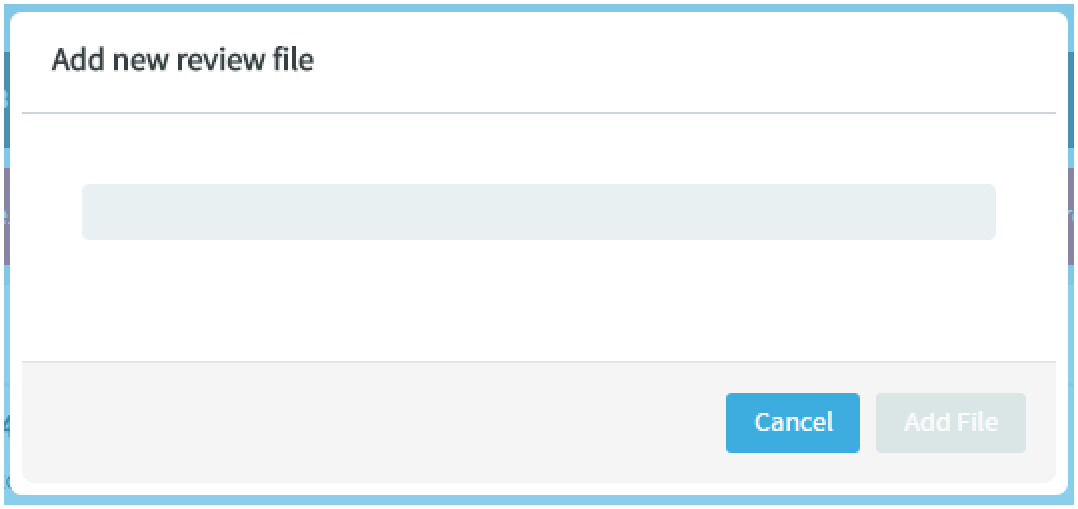
[artf286472] Add service tests for artf286463
1.testCreateTemplateWithFlexfields

Change-Id: I856cb09011a312ea5b1121f36c5b9ad2de42d8ce

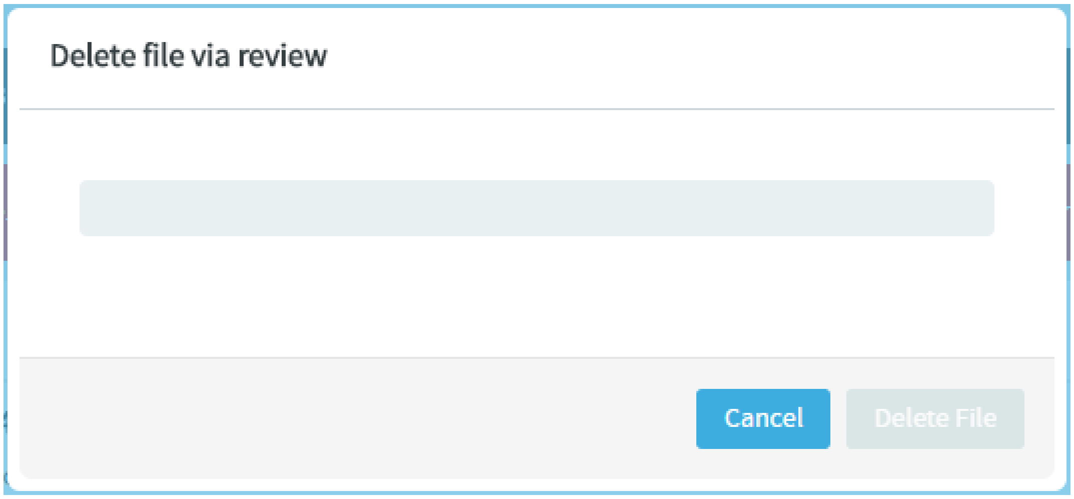
Click to open a file for editing.

module/application/document/src/test/java/com/vasoftware/sf/server/services/docman/TransitionService2Test.java

Type the name of the file to see results matching the file name, select a file and click **Add File**.



Type the name of the file you want to delete to see results matching the file name, select the file and click **Delete File**.



3. Click the **Complete File Edits** icon.

Merge (close) a Pull Request

Once the pull request is reviewed, it is ready to be merged, that is the **Merge** button on the pull request details page is enabled only if the pull request satisfies all the repository specific qualification criteria. For example, it is possible to merge pull requests even without any voting if "no voing" has been defined as gating criteria. Also, it might require both voting AND acceptance by Continuous Integration; basically it totally depends on the gating criteria of the repository in question.

> **NOTE:** These criteria are set by the repository owner in **Settings > Policies** tab for the **Pull Request** Repository Category.

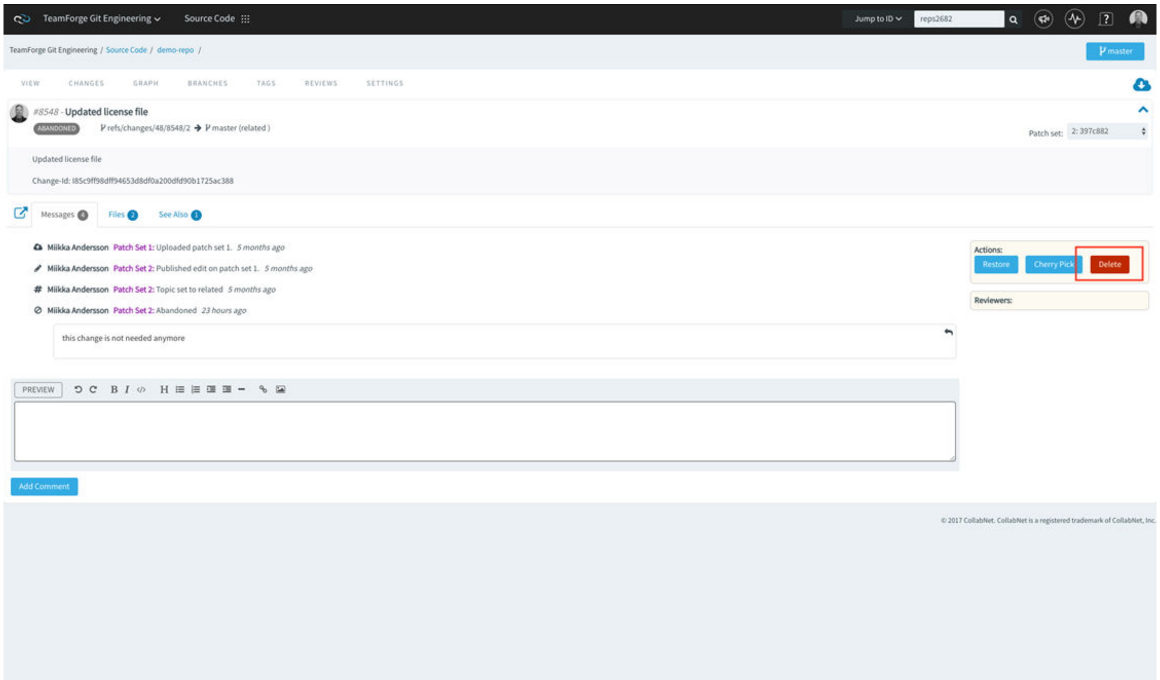
1. Click the **Merge** button to merge the source branch into the target branch.
   - If the source branch is not updated with the latest changes, a merge conflict is detected prompting you to rebase your request.
   - Click the **Rebase** button. Once rebased, the pull request has to be revalidated after which you can merge the pull request into the target branch.
2. The newly-merged pull request is added to the list of merged pull requests.

   > **NOTE:** Once merged, on the **Graph** view, you can see that the merged branch has been added to the graph. A link to the pull request is also provided.

View Pull Requests

1. Click the *Reviews* tab on a GIT repository page. The pull request details page displays the list of *Open*, *Merged* and *Abandoned* pull requests under appropriate tabs.

   - For each pull request, the author, title, number of thumbs up/down, and the time elapsed since the pull request was created, are shown.
   - On the *Abandoned* pull request details page, you have the option to restore or delete an abandoned pull request. To restore or delete an abandoned review, open the review in the code browser and click **Restore** or **Delete** respectively from *Actions*.



#8548 - Updated license file

refs/changes/48/8548/2 → master (related)

Updated license file

Change-Id: I85c9ff98dff94653d8df0a200dfd90b1725ac388

Miikka Andersson Patch Set 1: Uploaded patch set 1. *5 months ago*

Miikka Andersson Patch Set 2: Published edit on patch set 1. *5 months ago*

Miikka Andersson Patch Set 2: Topic set to related *5 months ago*

Miikka Andersson Patch Set 2: Abandoned *23 hours ago*

this change is not needed anymore

Related Links

Gerrit Code Review Policies

Integration with Black Duck Code Sight (BDCS) is no longer supported in TeamForge 17.1 and later. TeamForge 17.1 (and later) is equipped with its own code search function powered by Elasticsearch. You can do away with BDCS integration while upgrading to TeamForg 17.1 (and later) and set up TeamForge Code Search, which is now one of the integral services of TeamForge. This section discussions the features of TeamForge Code Search and what it takes to set up Code Search on your site.

TeamForge Code Search powered by Elasticsearch

TeamForge's Code Search feature uses Elasticsearch as its engine for the indexing and retrieval of documents. Elasticsearch, used in TeamForge, has no customizations other than what is provides via its engine and API. For more information about Elasticsearch, refer to its documentation.

TeamForge Code Search (Elasticsearch): How it works?

Elasticsearch provides the engine for the indexing and retrieval of documents. TeamForge is equipped with its own indexes that extract source files from the SCM repositories and send them to the Elasticsearch engine for indexing. These indexers run as an hourly cron job. When the job starts, it calls an interal TeamForge API that returns the list of repositories to be indexed on the specified SCM integration server. The indexer then processes the repositories, one at a time, starting with the repository that was most recently committed to. For each repository, it determines if a full indexing is needed or just an incremental indexing is sufficient to catch up since the last indexing operation. It then goes down the list of files and processes them for indexing. Only files that need to be indexed are extracted. For example, large binary files are not extracted as they would be indexed anyaway. As the indexers crawl a repository, it would skip any file that is greater than 1 MB. To index a file it has to be UTF8 text.Files are scanned for file encoding and converted to UTF8 if required. Binary files are skipped by the indexers as would be files that cannot be converted to UTF8 format. All files that are skipped are logged along with the reason.

The engine in Elasticsearch is configured to convert the source to lower case and tokenize it. A Camel case filter is used so that common programming elements are broken into individual tokens based on camel case and other common separators used in function names etc. A set of common programming terms (if, then, else, return, exit etc.) are not indexed by default.

How the data is indexed does effect the search results as partial matches are not generally returned. For example, consider a function name in the source code such as: TeamForgeHelper. This would be indexed so that a search for any of these terms would find a match: Team, Forge, Helper, TeamForgeHelper. Nothing else would match, including “TeamForge” or “Tea”.

Search results are returned to users with TeamForge RBAC applied to it at a repository level. In other words, a user must have permission to view the entire repository to be able to search it. This works with SVN Path-based permissions (PBP), but only if PBPs are being used to restrict write access. If PBP’s are used to deny the user view access to certain paths in the repository, then TeamForge Code Search denies the user the ability to search the repository.

Considerations while upgrading to TeamForge 17.1 (or later)

Consider the following points while upgrading from TeamForge 16.10 (or earlier) to TeamForge 17.1 (or later) versions.

- Code Search is now an integral part of TeamForge, which is installed by default during TeamForge installation
- You can install TeamForge Code Search either on the TeamForge Application Server or on a separate SCM integration Server. It is recommended to install TeamForge Code Search on a separate server if your site’s indexing needs are considerable high (large number of repositories, for example).

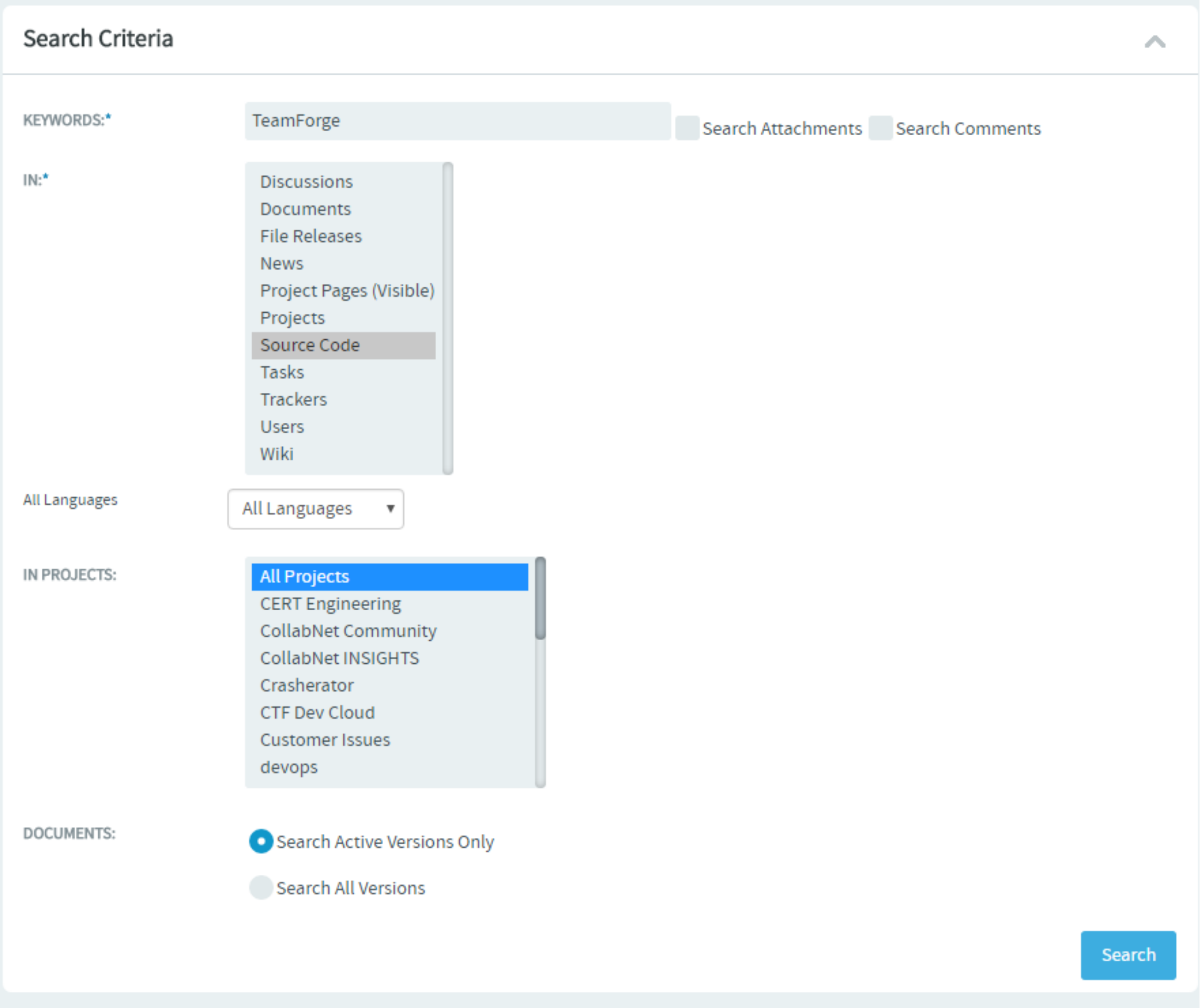
- TeamForge Code Search works with GIT and SVN repositories. TeamForge Code Search has no support for CVS repositories.
- Installation of Elasticsearch is determined by adding the “codesearch” identifier to the SERVICES token of either the TeamForge Application Server or the SCM Integration Server (if Code Search runs on a separate server). Refer to the `site-options.conf` configuration section of TeamForge installation/ upgrade instructions for more information.
- Elasticsearch needs 2GB of JVM heap size by default on a TeamForge site. You must have adequate RAM to accomodate the JVM heap requirements of Elasticsearch in addition to the JVM heap requirements of other components such as Jboss, integrated applications, and so on.
- If required, you can increase the JVM heap size for Elasticsearch. Set the [ELASTICSEARCH_JAVA_OPTS][siteoptiontokens.html#elasticsearch] token with the Elasticsearch JVM heap size required for your site.
- Elasticsearch stores every document, which for code search is each source file that is indexed, and it has its index itself as well. As a rule of thumb, the index for a repository would be roughly the same size as the working copy for that repository but in practice it will likely be smaller. Consider that all binary files and all files greater than 1 MB are not indexed at all. So, obviously any repository that has large working copy due to these types of files will have a much smaller index. By default, Elasticsearch compresses the original files using LZ4 compression. It also offers a “best_compression” codec that compresses the originals using DEFLATE algorithm. In TeamForge, indexes have been updated to use the maximum compression.
- The Elasticsearch data and logs are stored in */opt/collabnet/teamforge* folder. Log rotation, startup and so on are all managed the same way as other TeamForge services.
- As Black Duck Code Sight (BDCS) is not supported:
  - remove all BDCS tokens from your `site-options.conf` file while upgrading to TeamForge 17.1 or later. TeamForge create runtime fails otherwise. See Site Options Change Log for a list of obsolete `site-options.conf` tokens.
  - there is no migraton support for existing BDCS indexes. All the repositories should be indexed afresh post upgrade to TeamForge 17.1 (or later). The list of repositories to index, remains the same though. After upgrading to 17.1 or later, whatever repositories were being indexed by BDCS are indexed by the new Code Search without any user intervention. However, if you had customized indexing for one or more repositories using the BCDS administration UI, such information will have been migrated and would need to be done again, but that can now be done via the TeamForge UI.

How to search for Source Code?

Searching repositories for code snippets can be done via the “Jump to ID” search or via the Advanced Search.

Just select **Source Code** from the **Jump to ID** drop-down list, type the search keyword and press enter to search the repositories of the project in context. You can also select **Advanced Search** from the **Jump to ID** drop-down list and do an advanced searc for the required code.

1. Click **Advanced Search** from the **Jump to ID** drop-down list.
2. Type the search keyword.
3. Select **Source Code** from the list of components.
4. Set the scope of search: Select `All Languages` (default) or one of the programming languages such as C, C++, C# and so on from the **All Languages** drop-down list.
5. Select either *All Projects* or a selected few projects (select one or more projects) from the **IN PROJECTS** list.



Search Criteria

KEYWORDS:* TeamForge □ Search Attachments □ Search Comments

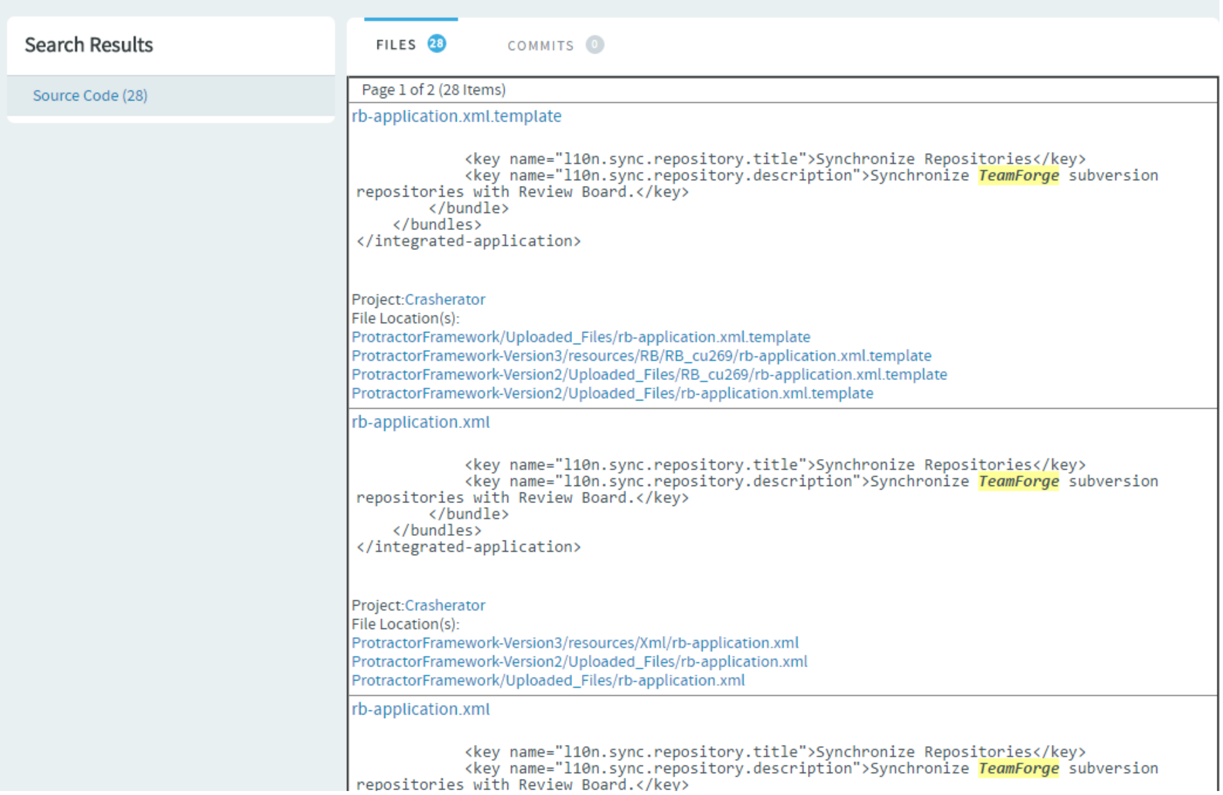
IN:* Discussions, Documents, File Releases, News, Project Pages (Visible), Projects, **Source Code** (selected), Tasks, Trackers, Users, Wiki

All Languages: All Languages ▾

IN PROJECTS: **All Projects** (selected), CERT Engineering, CollabNet Community, CollabNet INSIGHTS, Crasherator, CTF Dev Cloud, Customer Issues, devops

DOCUMENTS: ● Search Active Versions Only ○ Search All Versions

6. Click **Search**. The search results are displayed in *Files* and *Commits* tabs.



Page 1 of 2 (28 Items)

rb-application.xml.template

```
            <key name="l10n.sync.repository.title">Synchronize Repositories</key>
            <key name="l10n.sync.repository.description">Synchronize TeamForge subversion
repositories with Review Board.</key>
        </bundle>
    </bundles>
</integrated-application>
```

Project:Crasherator
File Location(s):
ProtractorFramework/Uploaded_Files/rb-application.xml.template
ProtractorFramework-Version3/resources/RB/RB_cu269/rb-application.xml.template
ProtractorFramework-Version2/Uploaded_Files/RB_cu269/rb-application.xml.template
ProtractorFramework-Version2/Uploaded_Files/rb-application.xml.template

rb-application.xml

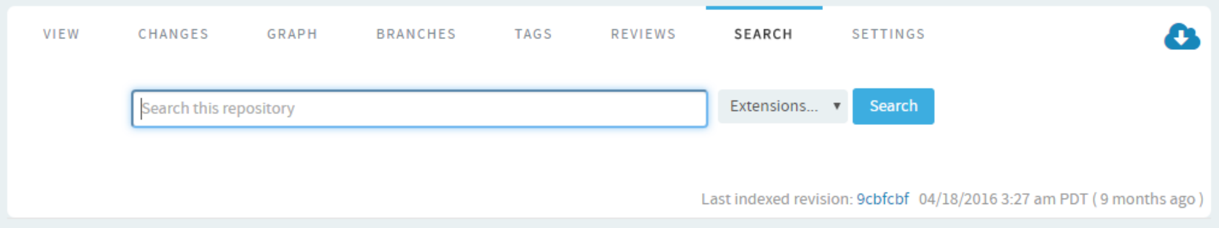
```
            <key name="l10n.sync.repository.title">Synchronize Repositories</key>
            <key name="l10n.sync.repository.description">Synchronize TeamForge subversion
repositories with Review Board.</key>
        </bundle>
    </bundles>
</integrated-application>
```

Project:Crasherator
File Location(s):
ProtractorFramework-Version3/resources/Xml/rb-application.xml
ProtractorFramework-Version2/Uploaded_Files/rb-application.xml
ProtractorFramework/Uploaded_Files/rb-application.xml

rb-application.xml

```
            <key name="l10n.sync.repository.title">Synchronize Repositories</key>
            <key name="l10n.sync.repository.description">Synchronize TeamForge subversion
repositories with Review Board.</key>
```

7. In case you want to search for code in a specific repository, select **Project Home > Source Code**, select a repository you want to search, click **Browser Repository**, and select the *Search* tab. You can search all files in a repository or narrow your scope to specific file types such as C, C++, C# and so on.



Last indexed revision: 9cbfcbf 04/18/2016 3:27 am PDT (9 months ago)

8. Type your search keyword, select a file extension (optional) and click **Search**. The search results are displayed in *Files* and *Commits* tabs.

When you delete a repository, a request is submitted to the administrator for approval.

You need to have the required permission to delete SCM repositories.

1. Click **SOURCE CODE** from the **Project Home** menu.

2. In the list of the repositories, select the repository you want to delete and click **Delete**. The following confirmation message appears: `All SCM data in this repository will be lost. Are you sure you want to delete this repository?`

3. Click **OK** to delete.

Your request for deleting a repository is submitted. You will receive an email notification when your repository is deleted or if your request for deleting a repository is denied.

- If the SCM server that you chose does not require approval for deleting repositories, the repository is deleted right away.
- If the SCM server that you chose requires approval for deleting repositories, a CollabNet TeamForge administrator must approve your request to delete a repository before it is deleted.

NOTE: By default, repository delete requests on unmanaged CVS servers require approval, because the SCM administrator must manually delete the repository.

The document management system is a centralized repository for creating, storing, and managing information about a project. Project members with the Document Admin permission can create, edit and administer documents and document folders. In addition to the Document Admin permission, individual permissions to create, edit and delete documents, and document folders can also be set so that project members who do not have the Document Admin permission can still perform these tasks.

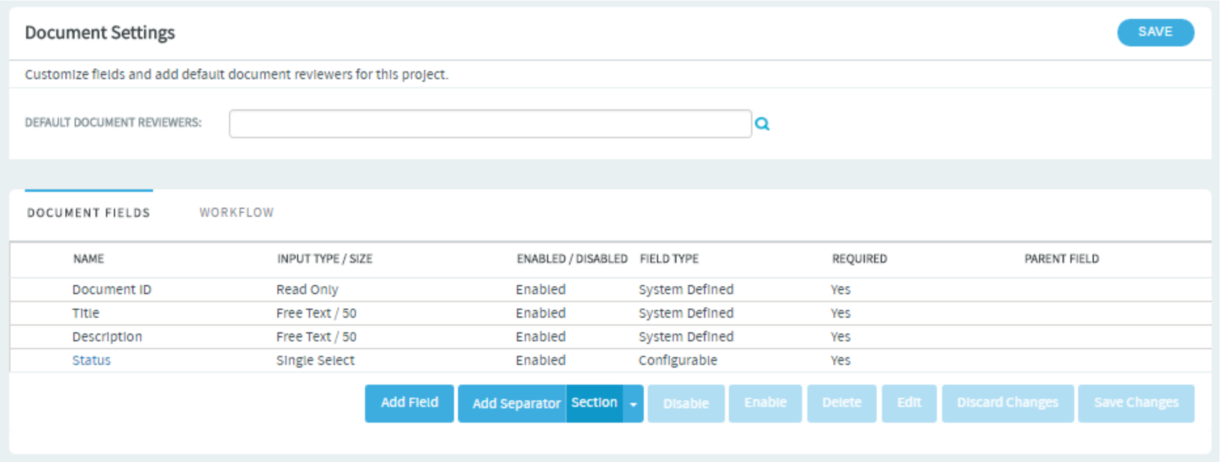
Document Settings

You can configure the document settings in TeamForge 17.1 and later versions. At the project level, document administrators can set up document fields, document workflow and mandatory document reviewers.

In addition to the system defined fields Document ID, Title and Description, a new configurable single select field, **Status** has been added in TeamForge 17.1. In addition to the default values, `DRAFT` and `FINAL`, you can now add custom statuses for documents.

You can also select one or more TeamForge users, for example project administrators and scrum masters, as mandatory document reviewers. This ensures that all project documents are reviewed by them when document reviews are initiated.

1. To configure document settings for a project, select the project from **My Workspace** menu and select **Projects > Project Admin > Document Settings**.



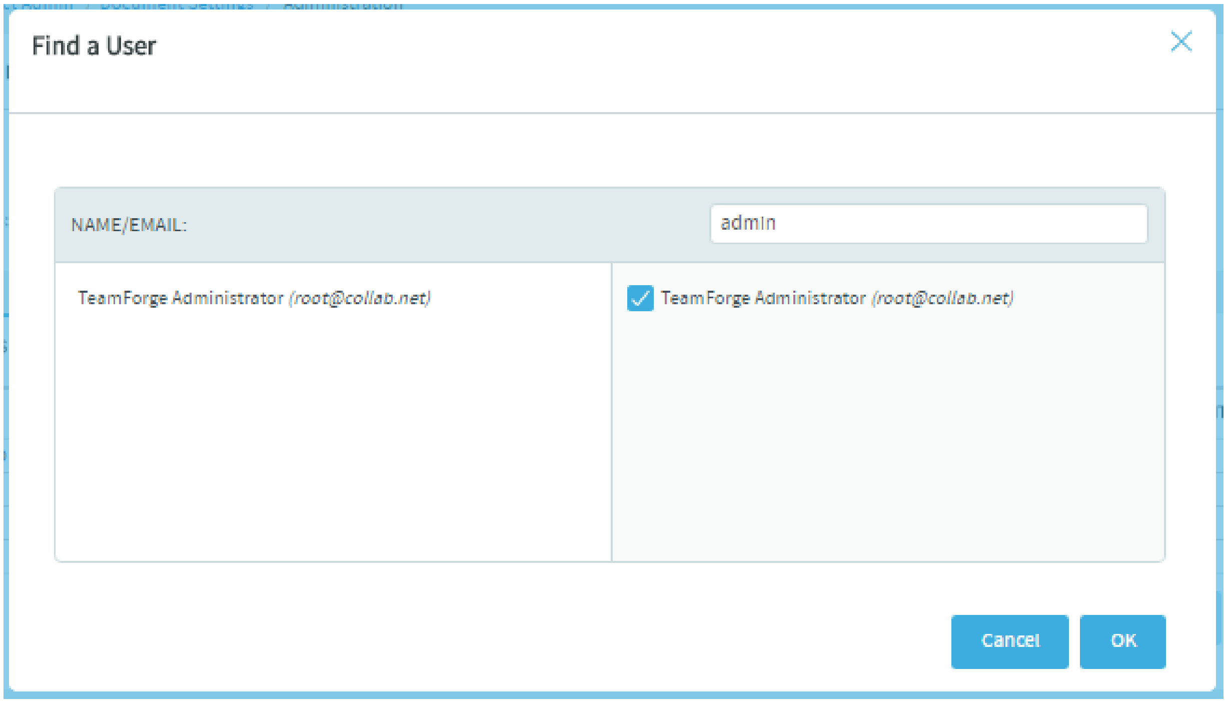
Document Settings

Customize fields and add default document reviewers for this project.

DEFAULT DOCUMENT REVIEWERS:

NAME	INPUT TYPE / SIZE	ENABLED / DISABLED	FIELD TYPE	REQUIRED	PARENT FIELD
Document ID	Read Only	Enabled	System Defined	Yes	
Title	Free Text / 50	Enabled	System Defined	Yes	
Description	Free Text / 50	Enabled	System Defined	Yes	
Status	Single Select	Enabled	Configurable	Yes	

2. Add one or more default document reviewers. Click the **Default Document Reviewers** field's search icon. Search and select users from the **Find a User** dialog box and click **OK**.

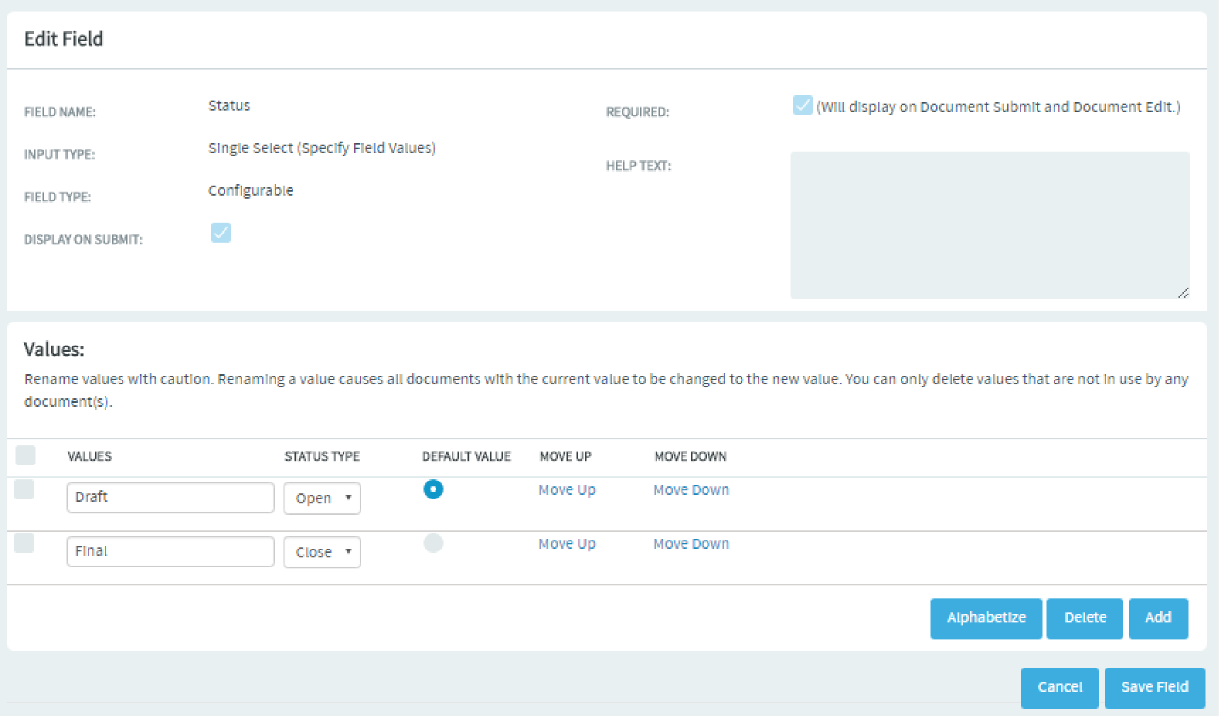


TeamForge Administrator *(root@collab.net)*

☑ TeamForge Administrator *(root@collab.net)*

3. Set up custom dcoument statuses. In addition to the default document statuses, `DRAFT` and `FINAL`, you can add one or more custom document statuses.

   a. Select the **DOCUMENTS FIELDS** tab and click the **Status** field. The **Edit Field** page appears.



Edit Field

FIELD NAME: Status

INPUT TYPE: Single Select (Specify Field Values)

FIELD TYPE: Configurable

DISPLAY ON SUBMIT: ☑

REQUIRED: ☑ (Will display on Document Submit and Document Edit.)

HELP TEXT:

Values:

Rename values with caution. Renaming a value causes all documents with the current value to be changed to the new value. You can only delete values that are not in use by any document(s).

□	VALUES	STATUS TYPE	DEFAULT VALUE	MOVE UP	MOVE DOWN
□	Draft	Open	●	Move Up	Move Down
□	Final	Close	○	Move Up	Move Down

b. Click **Add**. A new row is added in the **Values** section.

c. Type the name of the new document status, select a status type (open or close) from the drop-down list and select the **Default Value** option if you want to make this the default status when a document is created. You can move the row up or down by clicking the **Move Up** and **Move Down** links, if required.

d. When done, click **Save Field**. A confirmation message appears.

e. Click **OK** to save the new status.

f. Repeat the about steps to add more custom document statuses.

4. Add custom document fields (flex fields), if required. For more information, see Create Your Own Document Fields.

Create Your Own Document Fields

To track data that is not captured by the default set of fields, create new fields that fit your project's purposes.

You can create the following user-defined fields for your documents:

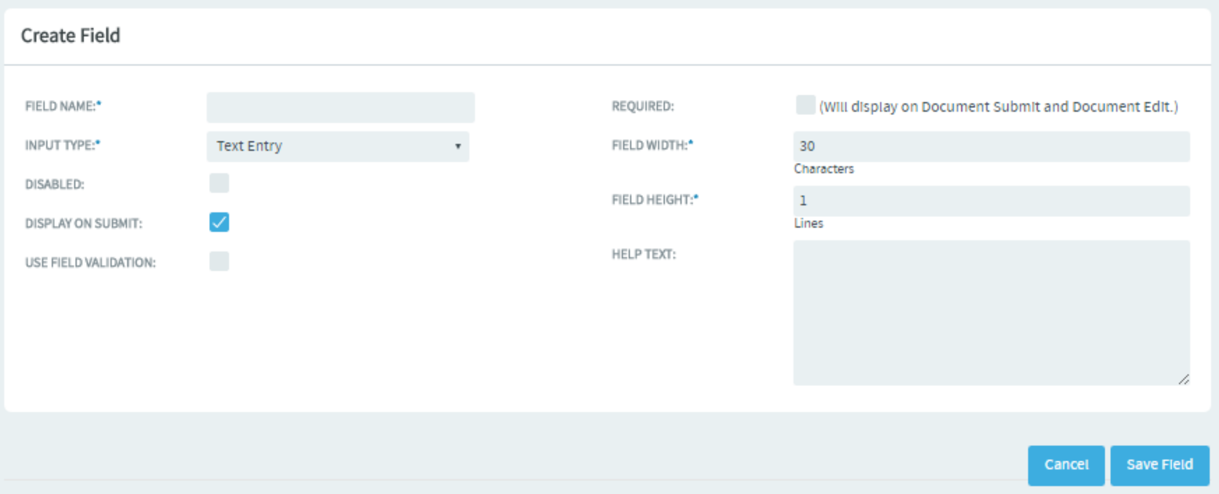
- Up to 30 text entry fields.

- Up to 30 date fields.
- Up to 30 single-select fields.
- Up to 30 multiple-select fields.

Create a Text Field

To let users type in date, create a text entry field. You can have up to 30 text fields.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Document Settings**.
3. On the **DOCUMENT FIELDS** tab, click **Add Field**. The **Create Field** page appears.



Create Field

FIELD NAME:*

INPUT TYPE:* Text Entry

DISABLED: □

DISPLAY ON SUBMIT: ☑

USE FIELD VALIDATION: □

REQUIRED: □ (Will display on Document Submit and Document Edit.)

FIELD WIDTH:* 30 Characters

FIELD HEIGHT:* 1 Lines

HELP TEXT:

4. On the **Create Field** page, provide a name for the field.
5. Configure the shape of the field with the **Field Width** and **Field Height** fields.
6. Select **Text Entry** from the **Input Type** drop-down list.
7. To help users enter the right text values, select **Use Text Validation** and supply a regular expression that describes the appropriate values. This can help reduce errors and keep your team's data as meaningful as it can be. For more detailed instructions, see Validate Text Field Values.
8. Click **Save Field**. The new field is created.

Validate Text Field Values

You can help users contribute useful information by testing their text entries against rules you configure. Text fields can be error-prone because they invite free-form input. You can help users provide usable information by automatically rejecting values that don't match the needs of the document.

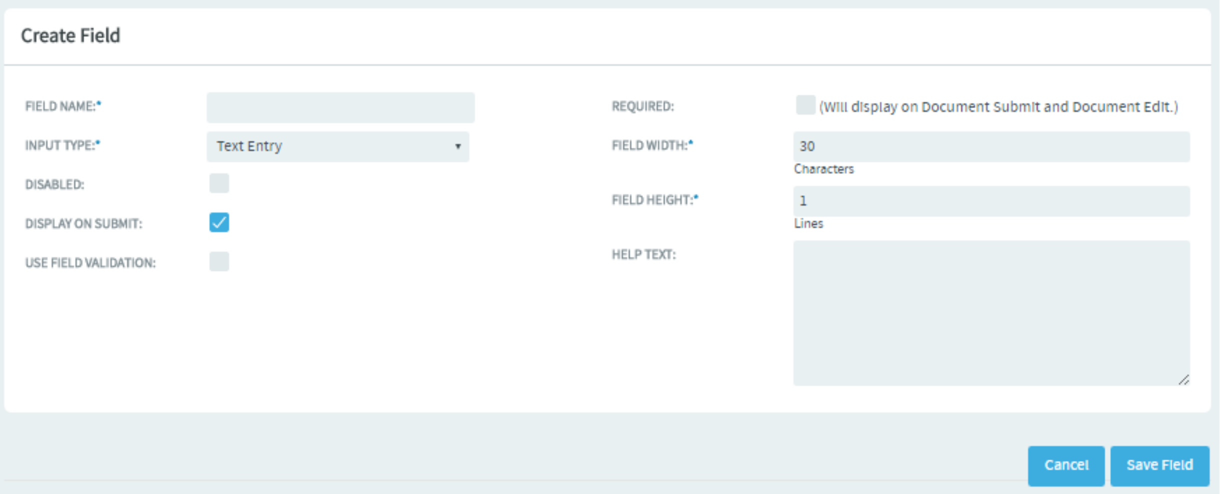
This simplifies things for the user, but for the document administrator it can be complicated. So let's look at an example.

> **NOTE:** If your goal is to require users to enter `some` value, whatever the value is, don't use text field validation. Select the `Required` check box instead.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. Click **Document Settings**.

3. On the **DOCUMENT FIELDS** tab, click **Add Field**. The **Create Field** page appears.



Create Field

FIELD NAME:*

INPUT TYPE:* Text Entry

DISABLED: □

DISPLAY ON SUBMIT: ☑

USE FIELD VALIDATION: □

REQUIRED: □ (Will display on Document Submit and Document Edit.)

FIELD WIDTH:* 30 Characters

FIELD HEIGHT:* 1 Lines

HELP TEXT:

4. On the **Create Field** page, provide a name for the field. For example, name the text field **Legs**. This is for users to record the number of legs each specimen displays.

5. Configure the shape of the field with the **Field Width** and **Field Height** fields.

6. Select **Text Entry** from the **Input Type** drop-down list.

7. Select the **USE FIELD VALIDATION** check box and supply a validation rule that requires the user to enter a number. For example, if a given insect has six legs, you'll want the user to enter the numeral **6**, and not a string such as "six" or "several".

Try this regular expression: `\d{1, 3}`

This rule requires the user to enter a number with one, two or three digits. Now, a user who means to record a centipede with 100 legs but enters 1000 by mistake will not be able to save the document until the error is corrected.

8. Enter a sample string to test your regular expression. Any part of the sample string that matches your regular expression appears under **Match Results**. If nothing appears, rework your regular expression until you get a match.

9. Create another text field and call it **Location**. This is where users will record the geographical spot where they collected the bug.

10. Select the **USE FIELD VALIDATION** check box and supply a validation rule that requires the user to enter a pair of geographical coordinates. For example, if a given insect was found outside CollabNet's California headquarters, you'll want the user to enter a string like 37.674689,-122.384652, and not something like "Brisbane" or "Out on the lawn".

    Try this regular expression: `[-]?[0-9]*[.]{0,1}[0-9]{0,4}`

    This rule requires the user to enter two numbers, separated by a comma, in the general format of a pair of mapping coordinates.

    > **NOTE:** This particular regular expression does not guarantee that the coordinates are valid, just that they look like coordinates.

11. Save your work. In the document whose settings you have been editing, try entering a number greater that 999 in **Legs**, or a street address in **Location**. The red **X** next to the field indicates that the text entry is incorrect. A green check indicates that the value meets the requirements.

    > **NOTE:** Any field in which you are validating text entries is identified by 'Text Entry (with Field Validation)' when listed on the DOCUMENT FIELDS tab.

Create a "Select" Field

To let users choose values from a list that you define, create a "Select" field.

You can create up to 30 single-select and 30 multiple-select fields for documents.

1. Click **PROJECT ADMIN** from the **Project Admin** menu.
2. Click **Document Settings**.
3. On the **DOCUMENT FIELDS** tab, click **Add Field**. The **Create Field** page appears.
4. On the **Create Field** page, provide a name for the field.
5. Use the **Input Type** menu to specify whether users will be able to select one value or more than one. If you're going to make this a required field, pick one of the values to be the default value. This value is applied to existing documents and documents that are moved from another project.
6. Decide whether users *must* choose a value.

- Required fields automatically appear on the **Submit Artifact** page.

  > **NOTE:** If you make the field required, you must specify a default value. If you make a `User` field required, specify one or more default users. If you make a `Date` field required, the default is 'today'.

- For optional fields, select **DISPLAY ON SUBMIT** if you want the field to appear when a user first creates a document.
- To prevent the field from being used at all, select **DISABLED**. (By default, new fields are enabled.)

1. Use the **Values** section of the **Create Field** page to add more values for the user to choose from
2. Keep adding values until you have the list of options you want, then click **Save Field**.

Create a "People-picker" Field

To let users choose other users from a list, create a "People-picker" field.

For example, you may want to create a "Document Owner" field that can be used to identify the user who owns the document. You might create a people-picker field called "Document Owner" to specify who that person should be.

In people-picker fields you create, users can select multiple users.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Document Settings**.

3. On the **DOCUMENT FIELDS** tab, click **Add Field**. The **Create Field** page appears.
4. On the **Create Field** page, provide a name for the field.
5. On the **INPUT TYPE** menu, select `Select User(s)`.
6. In the **DEFAULT FILTER** field, choose whether the list of people available in your new field will include members of this project or everyone who is registered on the site.
7. Configure the size of the field with the **FIELD WIDTH** field.
8. Click **Save Field**. The new field is created.

Create a Document Workflow

To channel project members' work on documents, set up rules for how a document can move forward.

> **NOTE:** A workflow is a sequence of changes from one status to another. You can define status transitions for any combination of document statuses in Document Settings.

Before creating a document workflow, see that these criteria are met:

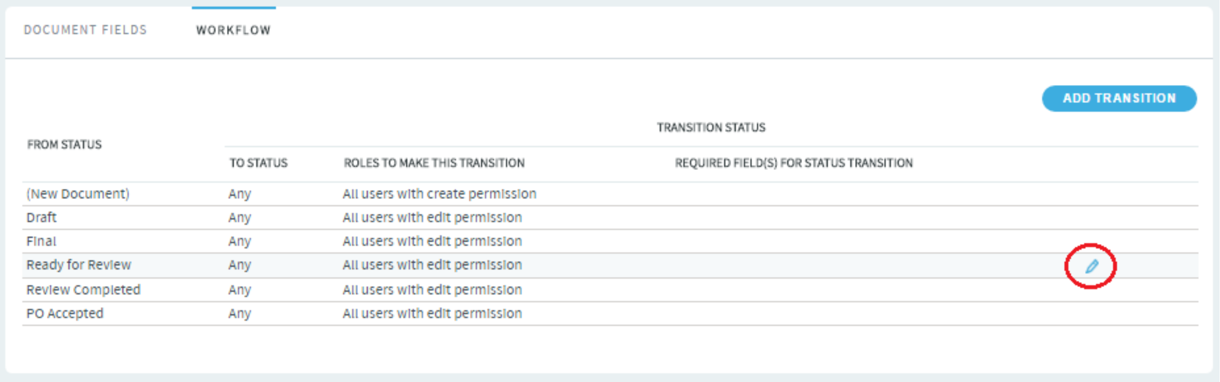
✓ You have configured a set of statuses, such as "Draft", "Ready for Review", "Review In Progress" and so on.

✓ Roles exist, and you can assign project members to them.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Document Settings**.
3. Click the **WORKFLOW** tab.

   > **NOTE:** The **WORKFLOW** tab lists the transition rules for all the document statuses that you have configured already. You can also add new transition rules, if required.

4. Move your mouse over the status rows to view the edit transition icon.



FROM STATUS	TRANSITION STATUS		
	TO STATUS	ROLES TO MAKE THIS TRANSITION	REQUIRED FIELD(S) FOR STATUS TRANSITION
(New Document)	Any	All users with create permission	
Draft	Any	All users with edit permission	
Final	Any	All users with edit permission	
Ready for Review	Any	All users with edit permission	
Review Completed	Any	All users with edit permission	
PO Accepted	Any	All users with edit permission	

5. Select one or more roles that can make the status transition and the **Required Field for Status Transition** from the drop-down lists.

6. Click **Save**.

7. To add a new transition rule, click **Add Transition**. A new row is added for the workflow.

8. Select the **From Status** and **To Status** from drop-down lists, select the **Select Roles** option and select one or more roles that can make the status transition, select the **Required Fields for Status Transition**, and click **Save**.

The workflow is now saved. When a user submits or edits the status of a document, he or she sees only the options that are allowed by the workflow.

Require Documents to be Associated with Artifacts

To help reduce the problem of `orphan` documents, require users who create a document anywhere on the site to associate the document with an artifact.

`Orphan` documents are documents that are abandoned because they are not connected to any tracked activity.

1. Open the `conf/site-options.conf` file in a text editor.
2. Change the value of the `sf.requireAssociationOnDocumentCreate` variable to `true`.
3. Change the value of the `sf.allowedAssociationTypeOnDocumentCreate` variable to `[TrackerArtifact]`.
4. If you want to prevent users from associating documents with closed artifacts, change the value of the `sf.requireArtifactToBeOpenOnDocumentAssociation` variable to `true`.
5. Save `conf/site-options.conf`.

6. Recreate the runtime environment.

   ```
   teamforge deploy
   ```

You can find and view a document by navigating to it, by searching for it, or by viewing it from a URL or unique identifier. You can also open a document from TeamForge directly into Microsoft Office applications.

What is a Document in TeamForge?

For TeamForge purposes, a document is any file that you track with a document artifact.

Any project involves some number of documents that have to be written, edited, reviewed and published. You can use a document artifact to facilitate and standardize the work you do with documents. TeamForge automatically maintains the version history of each document, including the person who posted each version, its status, and any version comments.

If you are the author, use the Review feature to track the process of getting others to evaluate your document.

To make sure everyone who needs to know about a document knows about it, associate it with a task, a code commit, or another kind of artifact.

Create a Document

You can create a document by uploading a file or by entering the text directly into TeamForge.

You can also create a document in TeamForge. TeamForge automatically detects the file extension for many common file types.

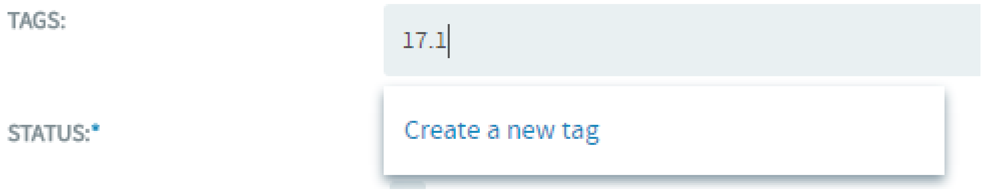
> **NOTE:** To create a document, you must have the Document Admin permission, or ‘Create/View’ permission for document folders or documents.

> **NOTE:** From 17.11 release, TeamForge is configured to send HTML emails to users assigned to and users monitoring the document that create. For more information, see HTML Emails for Documents.

> **NOTE:** You can now be able to unmonitor the documents by clicking the link **Unmonitor the Document** in the emails that you receive when you create documents. For more information, see Unmonitor Documents via HTML emails.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find the folder in which you want to create the document, and click **Add Document**. You can create a document in the root document folder or any other document folder.

3. On the **Create Document** page, enter a name and description for the document.
4. You can tag a document if you want to. Select one or more tags from the **TAGS** field. Your Project Administrator must have set up tags for you to add tags to your documents. For more information, see Set up Tags. You can also create new tags if you have Document CREATE/EDIT permissions. To create new tags, just type a new tag title and click **Create a new tag** link.



5. Identify the status of the document.
   - **Draft** - The document is not yet completed.
   - **Final** - The document is finished.
6. If you want to prevent others from editing or downloading the document, select appropriate check boxes in the LOCK DOCUMENT section.
   - To lock a document for editing, select **Prevent others from editing**.
   - To lock a document for downloads, select **Prevent others from downloading**. The download lock prevents other users from downloading the document. You must have enalbed the edit lock to make this selection.
   - To enable the Document Admin to edit the document, select **Enable Document Admin to edit**. You can choose to allow the Document Admin to unlock and edit a document after enabling the edit lock.

✓ By default, all the locks are disabled. You must have the edit lock enabled to select the download lock. Selecting the edit lock automatically enables the Document Admin to edit the document.

✓ To choose locks in the **LOCK DOCUMENT** section, you must have the Document Edit permission.

7. Select the document creation method.
   - To upload a document, select **Upload From File** and browse to the desired file.
   - To enter the content as text, select **Create Online**.

- To upload a document from a specific URL, select **Enter Document URL** and provide the URL.

> **NOTE:** You cannot prevent others from editing or downloading the document if you choose this URL option. Edit and download locks are not effective in this case.

8. Write any comments you may have about this version of the document.
9. Save your changes:
    - Click **Save** to save the document and return to the document list view.
    - Click **Save and View Details** to save the document and display the document details.

Configure Default Document Columns

When you're looking at the documents list, you can select the columns you want to see, either for this session or permanently.

You set your column preferences for the documents in each project independently. If your project administrator has set default columns for the entire project, your individual column choices override those settings.

1. Click **Documents** from the **Project Home** menu.
2. On the documents list page, click **COLUMNS > Configure**.
    - If you've already saved a column configuration, click it and skip the rest of these steps.
    - To go back to the default column configuration, click **System (default)** and skip the rest of these steps.
    - To set up a new configuration, click **Configure**.
3. Choose your columns.

    > **NOTE:** Only the user-defined fields get listed in the **Available Columns** field.

    1. Move the columns you want from **Available Columns** to **Selected Columns**.

        > **NOTE:** Selecting more columns can increase the time required to load the listing page.

    2. Remove any columns you don't need from **Selected Columns**.
    3. Use the move up and move down arrows to change the display order of the columns.
4. Apply your choices to your view of the documents.

- To use this arrangement this time only, click **Apply**. The next time you log in, you'll start with the default view again.
- To save your column layout for repeated use, click **Apply and Save**, then give your arrangement a name. The next time you log in, you'll see the column arrangement you just selected. (If you've sorted the records in your view that sort order is saved too.)

> **TIP:** If you are editing a column configuration that already exists, you can rename it by saving it under a new name.

5. To make the same set of columns appear every time you come to the documents page, click **COLUMNS > Save** and from **Save Mapping** page, select the checkbox **Default**.

The Document Details Page

Each document in TeamForge has a **Document Details** page where you can find information about the document.

Contents

The **Document Details** page has these components:

- **Document Details** - View details about the document such as the name, description, file type, and size.
- **Status** - View the document status, including whether it is locked for editing.
- **Created By** - Identify and contact the author of each document version.
- **Versions** - View the version history of the document.
- **Change Log** - View the change log of a selected document.
- **Associations** - View any items associated with the document.
- **Version Comment** - Explanation of changes from one version of the document to another.
- **Monitor** - Begin or stop monitoring the document.

Edit a Document

When you are called on to help develop a document, you can edit the document online or by posting a new version.

To edit a document that is currently locked for editing, the user must have the Document Edit permission. To download a document that is currently locked for downloading, the user must have the Document View permission. A Document Administrator that have the option, **Allow Document Admin to edit** enabled in the **LOCK DOCUMENT** section or a Site Admin can also remove the edit and download locks.

Update a Document

To update a document, you can upload a new document version or change the document’s name, description, or lock status.

You can also update a document in TeamForge directly from one of the following Microsoft Office applications: Word, Excel, Project and PowerPoint.

> **NOTE:** From 17.11 release, TeamForge is configured to send HTML emails to users assigned to and users monitoring the document that you update. For more information, see HTML Emails for Documents.

> **NOTE:** You can now be able to unmonitor the documents by clicking the link **Unmonitor the Document** in the emails that you receive when you update documents. For more information, see Unmonitor Documents via HTML Emails.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document free, find the folder containing the document that you want to edit.
3. Select the document you want to edit by clicking the document name.
4. On the **Document Details** page, click **EDIT/UPDATE**. You can see this button at the top right corner of the page only if you have Document Edit Permission.
5. On the **Edit Document** page, make your changes.
   - Update the name or description by entering new text in the appropriate field.
   - Change the status value, if appropriate.
   - To set locks, select appropriate check boxes in the **LOCK DOCUMENT** section.
   - Update a binary document by selecting **Upload From File**, then browsing for the updated file.
   - Update a text document by selecting **Create Online**, then editing the document text.
6. Enter a version comment to describe the changes made, if appropriate.
7. Click **Save**.

The document is now updated.

Change the Active Document Version

You can specify which version of a document is the active version at any time.

TeamForge stores every document version that you or any other user posts. When you open a document from the document list view, you open the active document version. When you open a document's **Document Details** page, the information you see is about the active document version.

The **Versions** section of the **Document Details** page lists all document versions and indicates which one is currently active. By default, the most recently posted document version is the active version.

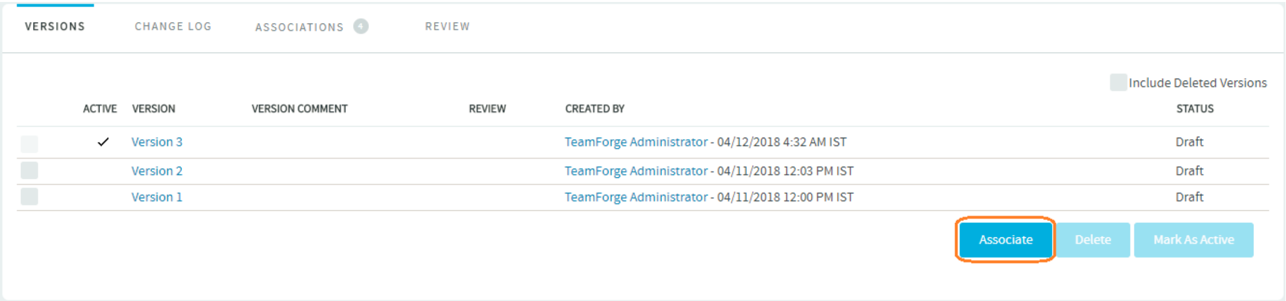
1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find the folder containing the document.
3. Select the document by clicking its name.

   > **TIP:** On the **Document Details** page, in the **Versions** section, the active document is indicate with a check in the **Active** column.

4. Select the version that you want to make the active version, and click **Mark As Active**.

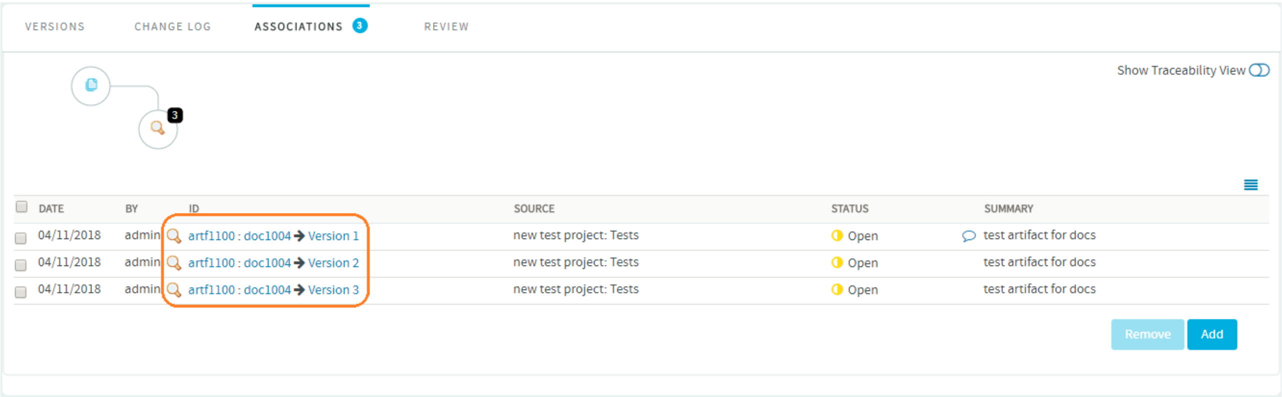
The selected document version is now the active version.

From TeamForge 18.1, you can associate TeamForge objects such as artifacts, documents, and so on with the desired version of a document. Select the desired document version and click the **Associate** button on the **Versions** tab in the **Document Details** page to select and associate a TeamForge object with this document version.



	ACTIVE	VERSION	VERSION COMMENT	REVIEW	CREATED BY	STATUS
□	✓	Version 3			TeamForge Administrator - 04/12/2018 4:32 AM IST	Draft
□		Version 2			TeamForge Administrator - 04/11/2018 12:03 PM IST	Draft
□		Version 1			TeamForge Administrator - 04/11/2018 12:00 PM IST	Draft

> **NOTE:** If you haven't selected any document version, the current active version of the document gets associated with the object.



☐	DATE	BY	ID	SOURCE	STATUS	SUMMARY
☐	04/11/2018	admin	artf1100 : doc1004 → Version 1	new test project: Tests	◐ Open	test artifact for docs
☐	04/11/2018	admin	artf1100 : doc1004 → Version 2	new test project: Tests	◐ Open	test artifact for docs
☐	04/11/2018	admin	artf1100 : doc1004 → Version 3	new test project: Tests	◐ Open	test artifact for docs

Find and View a Document

You can find and view a document by navigating to it, by searching for it, or by viewing it from a URL or unique identifier. You can also open a document from TeamForge directly into Microsoft Office applications.

Navigate to a Document

If you know where a document is posted, or would like to browse to see what documents are available, you can find a document by navigation.

You can get to documents from your project home page or by searching the project's documents page.

1. Choose a way to find documents:

    - Look for a **Documents** window on your project home page. If your project manager has added a document component, you will find all your project documents there (see Create a Project Page Component to know how to add a "Documents" component to the project home page).

    - If there is no Documents component, click **Documents** in the project navigation bar. The **List Documents** page appears.

2. Select a documents folder.

If a documents folder contains sub folders, you can expand the document tree by clicking the arrow to the left of the folder name.

1. To view or save a document, click the file type icon to the left of the document name. Use the filter options to locate the document you want. Documents with file types that can be viewed in a browser, such as HTML and text, are displayed in a TeamForge window. For other file types, use the appropriate viewing software.

2. To view additional information about the document, click the document name.

Search for a Document

Search for a document by its title, its content, the project or folder in which it is located, its file type, its creation date, or its status.

If you don’t know which project to search, or want to search for documents and other types of items simultaneously, use global search.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Click **SEARCH DOCUMENTS**.
3. On the **Document Search Criteria** page, enter the desired search criteria.

   > **TIP:** You can use wildcards.

   - To search by location: If you know the document is located in the current folder, check the **Current Folder** in the **SEARCH CATEGORY** field. Otherwise, the default **Root Category** searches all document folders in the project.
   - To search by name or description, enter the text in the **SEARCH KEYWORDS** field and select **Search Name and Description**.
   - To search by document content, enter the text in the **SEARCH KEYWORDS** field and **Search Within Document**.
   - To search by name, enter the text of the document name in the **DOCUMENT NAME** field.
   - To search all document versions, select **SEARCH ALL VERSIONS**.
   - To search by date range, enter the start and end dates for the search. Click the Calendar icon to select dates from a calendar.
   - To search by author, enter the user name in the **CREATED OR EDITED BY** field. Click the search icon to display a list of project members.
   - To search by document status, select one or more document statuses from the **Status** list box.
4. Click **Search**.

TeamForge lists the documents matching your search criteria.

Lock a Document

While you are working on a document, you can lock it to prevent others from editing or downloading it and then posting new versions to TeamForge.

You can lock a document when you create it or at any time afterward. When a document is locked, any user with the Document View permission can view or download the document. You can edit or download a locked document, if you are Site Admin or a Document Administrator that have the option, **Allow Document Admin to edit** in the **LOCK DOCUMENT** section.

1. On the **Document Details** page, confirm that the lock status is currently ‘Unlocked’.
2. Click **EDIT/UPDATE**. You can see this button at the top right corner of the page only if you have Document Edit permission.
3. On the **Edit Document** page, select appropriate check boxes in the **LOCK DOCUMENT** section.
4. Click **Save**. The document is now locked. It is presented by the Padlock icon in the document list view.

Unlock a Document

To work with a document that someone else has locked, you must unlock the document.

You can unlock a document that you have locked before. If you want to unlock a document that was locked by someone else, you must be a Site Admin or a Document Administrator that has the option, **Allow Document Admin to edit** enabled in the **LOCK DOCUMENT** section.

1. On the **Document Details** page, confirm that the document is currently locked.
2. Click **EDIT/UPDATE**. You can see this button at the top right corner of the page only if you have Document Edit permission.
3. On the **Edit Document** page, clear the check boxes in the **LOCK DOCUMENT** section.
4. Click **Save**.

The document is now unlocked. All project members with the Document Edit permission can now update the document.

Start a Document Review

There are a few things you have to do to get the most out of a document review.

To get a document reviewed, select the appropriate reviewers, set a due date, and explain to the reviewers what you need from them.

Before starting a document review, you must create the document in TeamForge.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. In the document tree, click the title of the document you want input on.
3. On the **Document Details** page, click the **REVIEW** tab, then click **Start a Review**.

   > **NOTE:** If a review is already in process, **Start a Review** does not appear, because you can only have one review at a time.

4. On the **Start a Review Cycle** page, give the review a descriptive name.
5. Enter a due date for reviewers. Click the Calendar icon to select dates.

   > **TIP:** Comments are due at 12:00 midnight on the date selected. Therefore, if you want all document reviews to be completed by end of day Tuesday, select Wednesday as the review end date.

6. Select the required and optional reviewers. Click the Search icon to pick from a list of project members whose access permissions enable them to see this document.
7. Write a message to your reviewers describing the document and the type of input you need from them. Your reviewers will get this message in an automatic email that shows you as the sender.

   > **TIP:** If you add yourself to the reviewers list, you will get a copy of this email for furture reference.

8. Click **Save**.

The document is now submitted for review.

- All reviewers receive an email with the details of the review, your message, and a link to the **Review Details** page where they can download the document and post their responses.

- The document status is changed to **Review** and remains so until the review is closed.

> **TIP:** To help keep track of the development of your document, associate a tracker artifact with your document review.

Read Review Responses

After starting a document review, you receive email notifications whenever a reviewer posts review responses.

You can read the review responses from the email or from the **Review Details** page. The **Review Details** page is the starting point for managing all document review activities.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find the folder containing the desired document.
3. Click the title of the document.
4. On the **Document Details** page, click the **REVIEW** tab. A summary of all open and closed reviews appears.
5. Click the title of the desired review. Review comments are listed in the bottom section of the page.

You can also reach the **Review Details** page by clicking the link provided in any related email notification or by clicking the title of the review in the **My Document Reviews** section of your **My Page**.

Edit Review Details

At any time after a review is started, you can change the review name, the required and optional reviewers, and the due date.

When you edit review details, affected reviewers get an email notification with the details of the change.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find the folder containing the desired document.
3. Click the title of the document.
4. On the **Document Details** page, click the **REVIEW** tab. A summary of all open and closed reviews appears.

5. Click the title of the desired review. The **Document Review** page appears.

6. Click **EDIT**.

7. On the **Edit Review Cycle** page, make your changes.

    - Click the Calendar icon to select dates from a calendar.

    - Click the Search icon to display a list of project members from whom you can select.

8. Click **Save**.

- When you add reviewers, new reviewers receive an email with the details of the review.

- When you remove reviewers, removed reviewers receive an email telling them that their reviews are no longer necessary.

- When you change the due date outstanding reviewers receive an email with the new due date.

Send a Reminder Email

To help project members keep track of their review obligations, you can send a reminder email to reviewers who have not yet submitted their responses.

1. On the **Document Review** page, find the outstanding required and optional reviewers.

2. Click **SEND REMINDER EMAIL**.

3. On the **Send Reminder Email** page, write a message to include in the email.

4. Choose whether to send the document as an attachment to the email.

> **NOTE:** The URL to the document is included in the email regardless of whether the document is attached.

5. Click **Send**.

The email is sent to all reviewers who have not yet submitted responses.

Close a Document Review

After a document review is completed, close the review.

Reviews are never closed automatically. You must close any outstanding reviews on a document before you can start a new one on the same document.

> **NOTE:** You can close a review even if the due date has not yet passed.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find the folder containing the desired document.
3. Click the title of the document.
4. On the **Document Details** page, click the **REVIEW** tab. A summary of all open and closed reviews appears.
5. Click the title of the desired review you want to close.
6. On the **Document Review** page, click **CLOSE REVIEW**.
7. On the **Close Review Cycle** page, change the document status to **Final** or **Draft**.
8. Click **OK**.

The review is now closed. Any reviewers with outstanding reviews receive email notifications that the review is closed.No additional review responses can be submitted.

Review a Document

To contribute to a review, you look at the document, consider responses from other reviewers, and post your own responses.

When you are identified as a reviewer of a document, two things happen:

- You receive an email about the review. The details include the due date, whether you are a required or optional reviewer, and any message text entered by the review submitter. The email also includes a link to submit your response.

- A new entry in the **Documents Awaiting Review** section of your **My Page** provides a summary of the review, including the review due date, the project in which the document is posted, and the person who is requesting your review.

1. Go to the **Document Review** page by clicking the link provided.
    - Click the link in the email notification you received with the review details.
    - Click the title of the document in the **Documents Awaiting Review** section of your **My Page**.
2. On the **Document Review** page, in the **Submit a Response** section in the center of the page, enter your response.
3. If appropriate, use the **Browse** button to attach a marked-up copy of the document, or another attachment.
4. Click **Submit**.

Your response is now submitted.

- The review submitter receives an email notification containing your response.
- Your response comments appear in the **Responses** section of the **Document Review** page.

Associate a Document with Other Items

When a document is related to other CollabNet TeamForge items such as tasks, tracker artifacts, file releases, or news items, you can connect the document to the other item by creating an association.

Creating associations between items enabling you to define relationships, track dependencies, and enforce workflow rules. Some example users for document associations include:

- Associate a requirements document with supporting feature requests or bugs in the tracker.
- Associate a document, or a document review, with the task that requires it as a deliverable.
- Associate a master document such as a product plan, with other supporting documents such as a test plan or beta plan.
- Associate a product announcement document with a file release.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find the folder containing the document to which you want to add an association. Select the document by checking the box, then click **Details**.
3. On the **Document Details** page, click the **ASSOCIATIONS** tab.
4. Click **Add**.

5. In the **Add Association Wizard** select the items with which you want to associate the artifact:
    - **ENTER ITEM ID** - If you know the item’s ID, you can enter it directly.

      ✓ To associate an object in an integration application from within TeamForge, use the [<prefix_objectid>] format. Successful associations appear hyperlinked.

      ✓ Each integrated application displays its prefix on moving the mouse over the application name in the tool bar.
    - **ADD FROM RECENTLY VIEWED** - Select one of the last ten items you looked at during this session.
    - **ADD FROM RECENTLY EDITED** - Select one of the last ten items you changed.
6. Click **Next**.
7. You may add a comment in the **ASSOCIATION COMMENT** text box.
8. Save your work.
    - Click **Finish and Add Another** to add additional associations.
    - Click **Finish** to return to the **Details** page.

      ✓ When an association is added to or removed from TeamForge objects such as tracker artifacts, tasks, documents, discussions, and file releases, a notification mail is sent to users monitoring these objects.

      ✓ An option is provided at site level and user level to make sure whether the notification mail has to be sent or not. For more information on this, see Configure your Site’s Settings.
9. Click the **ASSOCIATIONS** tab to view a graphical representation of all the associated items. Through the Association Viewer, you can choose to view associations in the form of a list or flip over to the Trace view to explore the layers of associations (including parent/child dependencies) laid out in a timeline. You can scroll across the Trace view by dragging the mouse over the association layer or use the ‘Previous’ or ‘Next’ arrows to view all the objects as events in a timeline.

While the **ASSOCIATIONS** tab shows the count of the total number of associations, you can only view the most recent 500 associations when you click the **ASSOCIATIONS** tab in case the artifact has more than 500 associations. You can, however, browse through the Association Viewer to view older associations.

You can click on each node on the graphical association viewer to filter and display the associated items in the table below the association viewer thus matching the number of associations provided on the selected node. For example, if the node that you select for filtering is having two associations on it, the table displays the two associated items as a result of the filtering process.

Project Members with the Document Admin permission can copy and move documents. They can also create and administer documents. However, they can delete documents, only if they have the specific delete permission.

Copy a Document

If a given document belongs logically to two or more document folders, keeping a copy in each folder can help users locate it.

Copying a document can also be useful if you want to use an existing documents as a template for creating a new document. When making a copy of a document, reference to the source document is maintained in the document title and the **Associations** tab.

> **NOTE:** Each copy must be maintained separately. Changes made to one copy are not propagated to other copies.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find and select the folder containing the document that you want to copy.
3. Select the document you want to copy, then click **Copy**. The **Copy Files** dialog box appears.
4. Select the project from the **Select Project** drop-down list and select the document foler where you want to copy the selected document.
5. Click **Copy**.

The document copy is now placed in the selected folder, with the “Copy of ”. The reference to the source document is also maintained in the **Associations** section of the **Document Details** page.

Move a Document

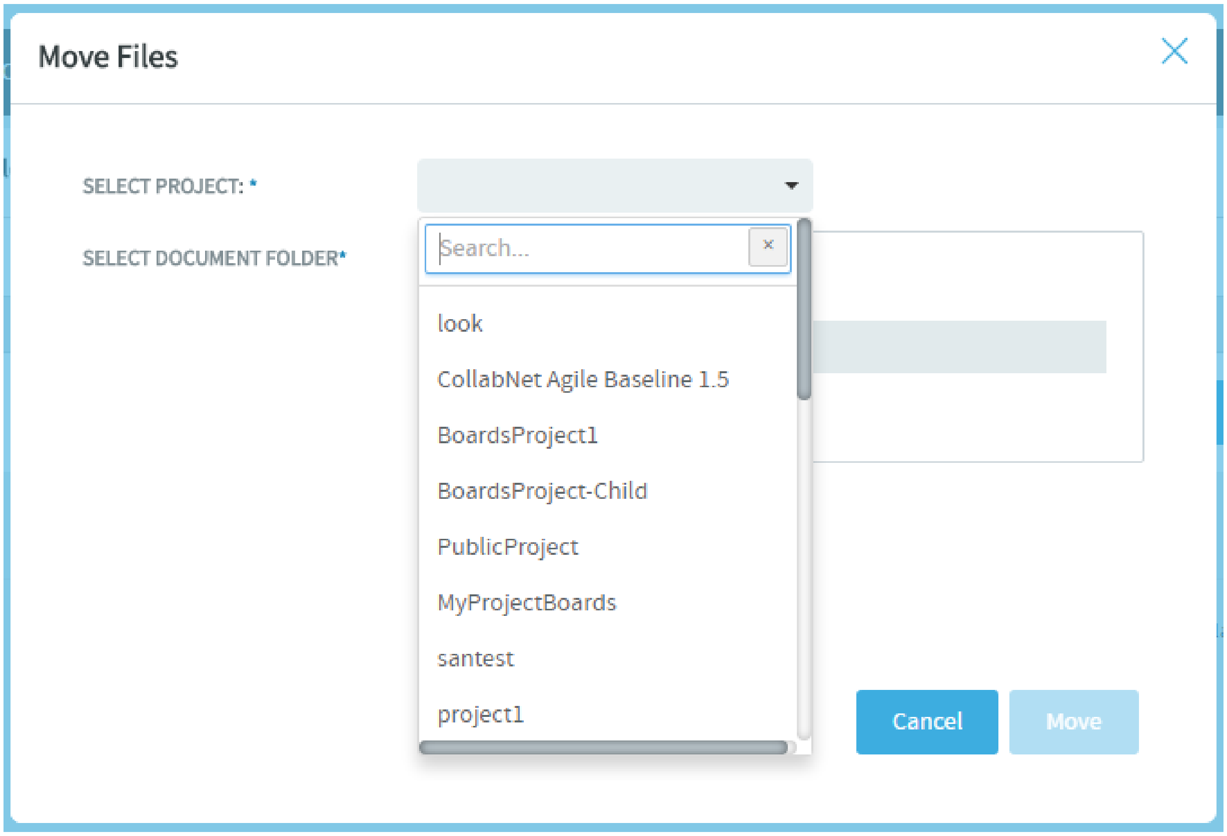
You can move it to another document folder within the project or to a document folder in another project at any time. This allows you to reorganize documents as needed.

You can move one or more documents in the same operation. You cannot move a document while it has an open review.

> **NOTE:** To move a document between projects, you must have document administration permission in both the source and destination projects.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find and select the folder containing the document that you want to move.
3. Select the document you want to move, then click **Move**. The **Move Files** dialog box appears.

4. Select the project from the **Select Project** drop-down list and select the document folder where you want to move the selected document.



5. Click **Move**.

The document is now moved to the selected folder.

Delete a Document

If you no longer need a document, you can delete it. Deleting a document deletes all the document's versions, plus the document's details and review history.

> **NOTE:** To delete documents, you must specifically have the delete permission ('Delete/View') for document folders or documents. This is irrespective of whether you have the Document Admin permission or not.

1. Click **DOCUMENTS** from the **Project Home** menu.

2. Using the document tree, find the folder containing the document that you want to delete.
3. Select the document you want to delete.
4. Click **Delete**.

The document is deleted.

Delete Document Versions

Document versions can grow in number over time. You may choose to delete one or more versions of a document if you no longer need them. This saves a lot of disk space and maintenance overheads.

You can delete a document's version only if:

- you have the delete permission ('Delete/View') for documents. This is irrespective of whether you have the Document Admin permission or not.
- the document has more than one version available.
- the selected document version is not marked as an active version.

CAUTION: Deleting a document's version permanently removes the version and cannot be undone. Exercise caution while deleting document versions.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find the folder containing the document that you want to delete.
3. Click the document's name link to see its versions.
4. Select one or more document versions you want to delete from the **VERSIONS** tab and click **Delete**. A confirmation message is displayed.
5. Click **OK** to delete.

The document versions are deleted.

Project members with the Document Admin permission can create, edit, move and reorder document folders. Similarly, project members with the relevant individual permission can create, edit, delete or view document folders.

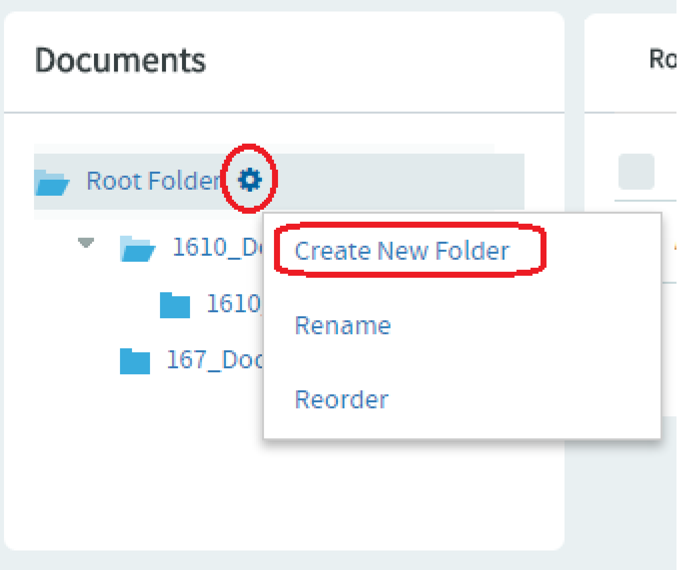
Create a Document Folder

Document folders, like file directories, allow you to organize your documents in a logical, hierarchical manner using labels that are meaningful to you and your organization.

You can create folders in the root folder or as subfolders in any other folder. There is no limit to the number of folders you can create.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find and hover your mouse over the folder where you want a new document folder created. For example, hover your mouse over the **Root Folder**.

   A settings icon shows up when you hover your mouse over a folder.
3. Click the settings icon and select **Create New Folder** from the menu.

   The **Create Folder** dialog box appears.
4. In the **Create Folder** dialog box, enter a name for the folder.
5. Click **Save**.

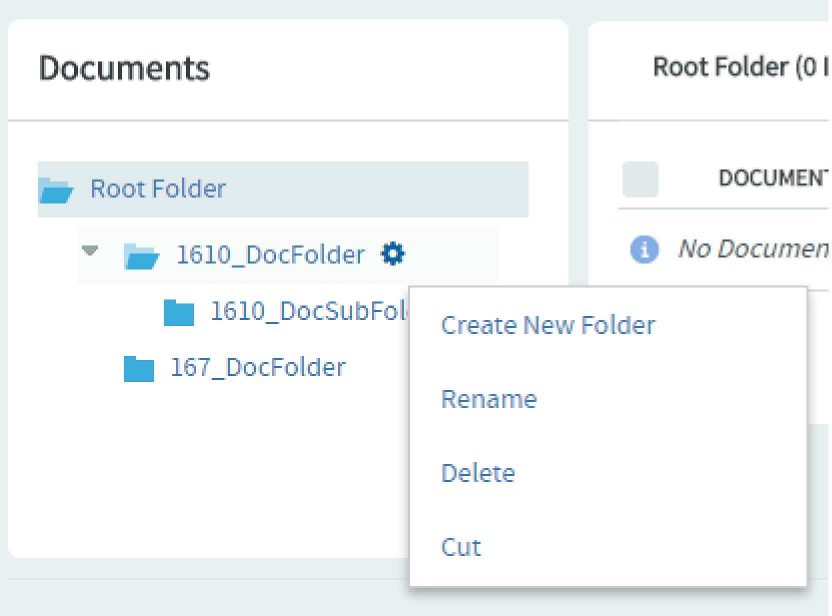
The folder is created.

Rename a Document Folder

When the function of a document folder has changed, it’s a good idea to rename the folder.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find and hover your moust over the folder which you want to be renamed.

   A settings icon shows when you hover your mouse over a folder.
3. Click the settings icon and select **Rename Folder** from the menu.

   The **Edit Folder** dialog box appears.
4. In the **Edit Folder** dialog box, enter a new name for the folder.
5. Click **Save**.

The folder is renamed.

Move a Document Folder

You can move a document folder within a project. You can move a folder in the following ways:

- From the root folder to a subfolder.

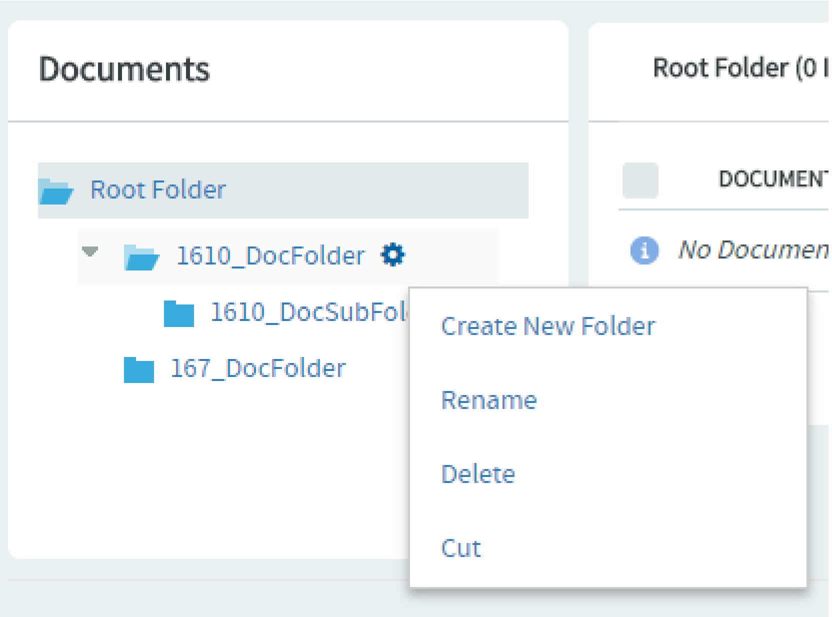
- From a subfolder to the root folder.
- From a subfolder to another subfolder.

When you move a document folder, any documents and subfolders that it contains are also moved to the destination folder.

Project members monitoring the destination folder will continue to receive monitoring notifications after a folder is moved. Project members not monitoring the destination folder will not receive further monitoring notifications.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find and hover your mouse over the folder that you want to move.

   A settings icon shows up when you hover your mouse over a folder.
3. Click the settings icon and select **Cut** from the menu.

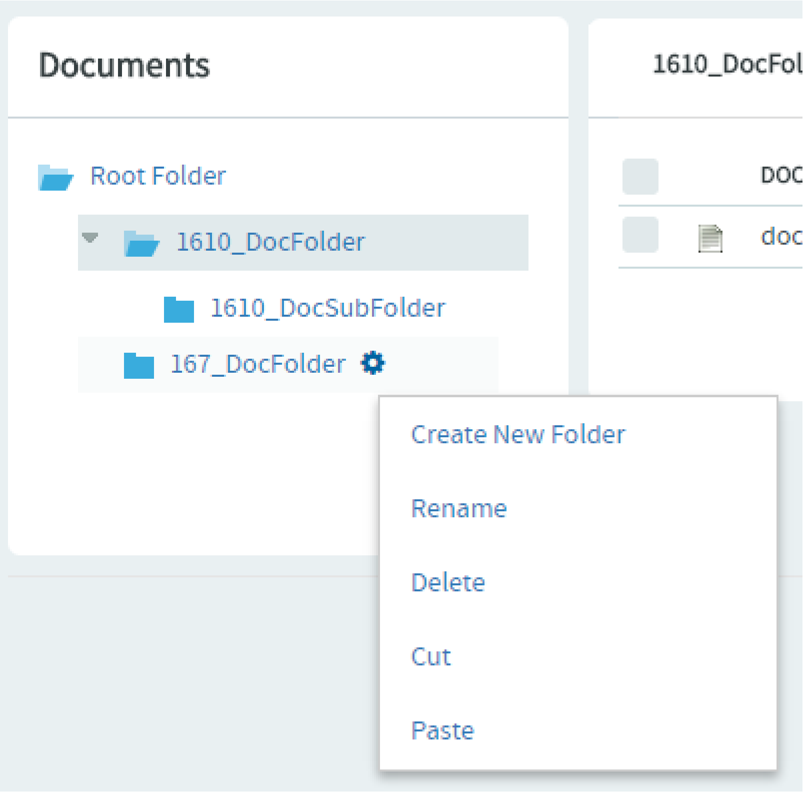


4. Using the document tree, find and hover your mouse over the folder into which you want to move the selected folder.

   A settings icon shows up when you hover your mouse over a folder.

   NOTE: You can move a folder into the root folder or into any other folder.

5. Click the settings icon and select **Paste** from the menu.



The folder is moved to the destination folder.

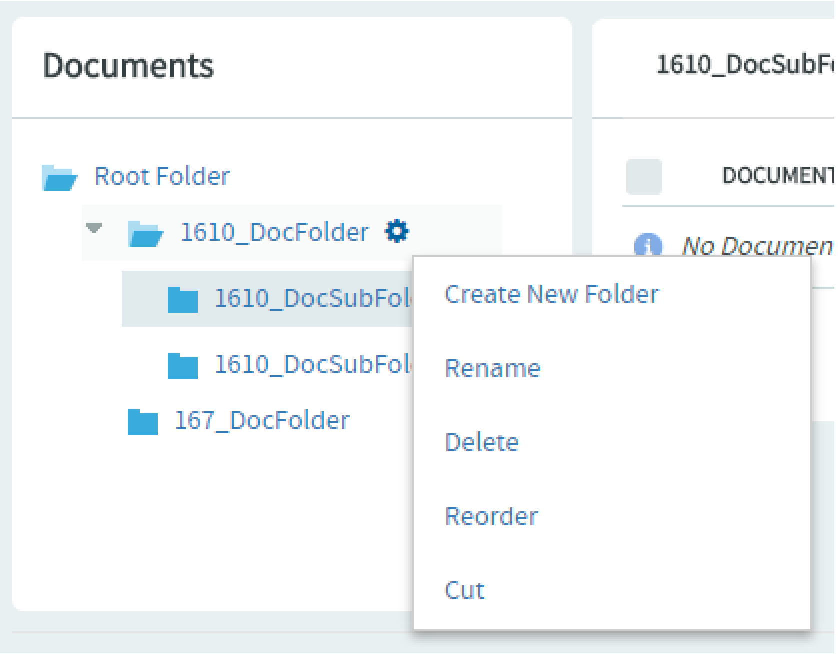
Reorder Document Folders

To change the order in which document folders are displayed, you can alphabetize them or reorder them arbitrarily.

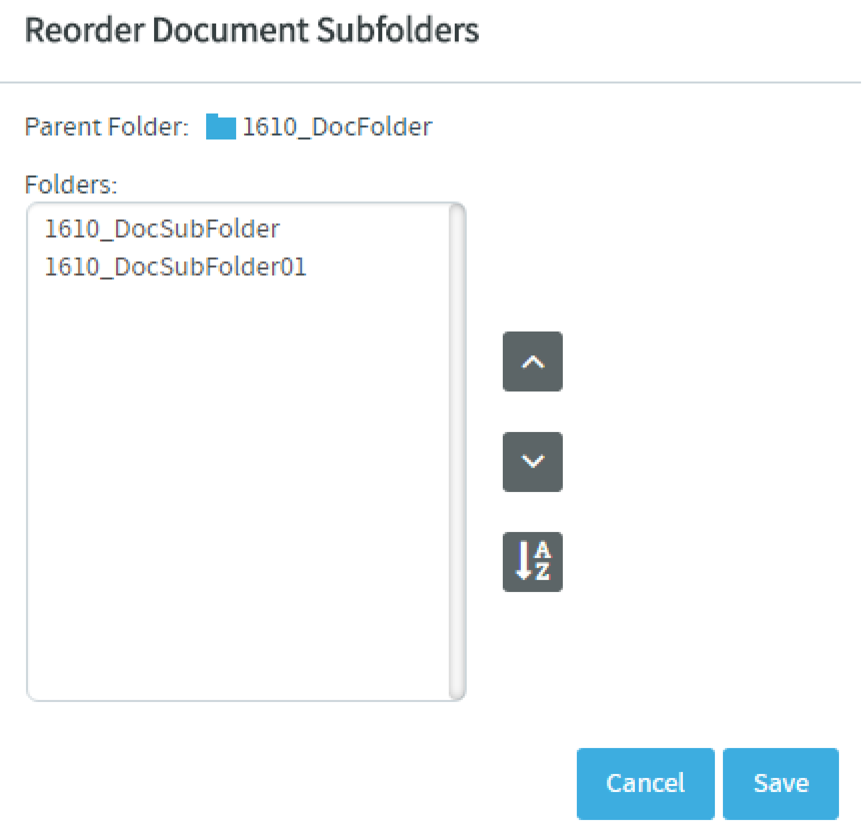
By default, document folders are ordered as they were created. New document folders are added to the end of the document tree.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find and hover your mouse over the foler containing the subfolders that you want to reorder.

   A settings icon shows up when you hover your mouse over a folder.
3. Click the settings icon and select **Reorder** from the menu.



The **Reorder Document Subfolders** dialog box appears.



Reorder Document Subfolders

Parent Folder: 1610_DocFolder

Folders:

1610_DocSubFolder
1610_DocSubFolder01

4. Organize the subfolders:
    - To sort them alphabetically, click alphabetize icon.
    - To reorder a specific folder, select it by clicking the title, then click **Move Up** or **Move Down** until the foler is where you want it.
5. Click **Save**.

Delete a Document Folder

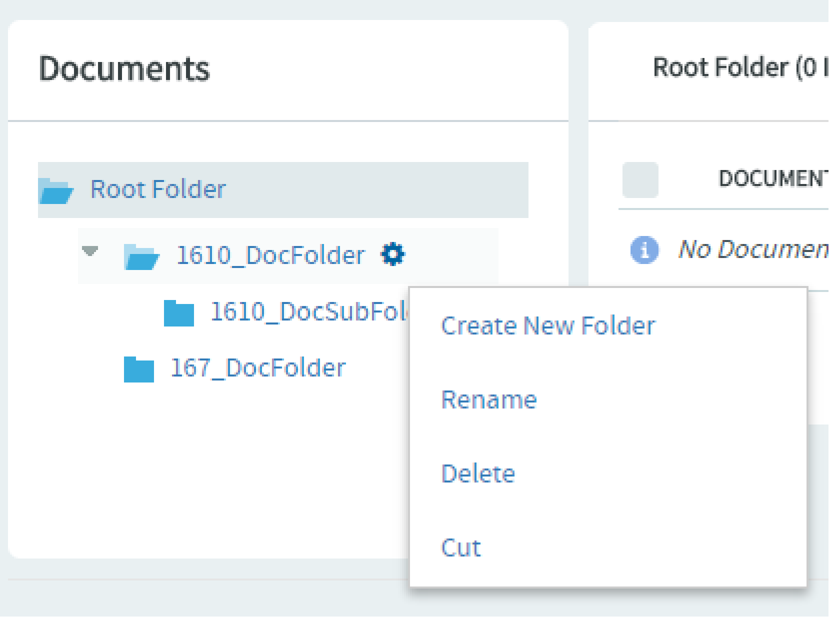
If you no longer need a document folder, it’s a good idea to delete it. Deleting a document folder deletes all of the documents within it.

Deleting a document folder deletes all of the documents within it.

IMPORTANT: Delete a document folder only if you are sure that you no longer need any of the documents within it. Move any documents that you want to save. Even if you have the Document Admin permission, you cannot delete a document folder unless you specifically have the delete permission.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find and hover your moust over the folder you want to delete.

   A settings icon shows up when you hover your mouse over a folder.
3. Click the settings icon and select **Delete** from the menu.

   A confirmation message appears.
4. Click **OK** to delete the document folder.

   ATTENTION: If the folder contains one ore more documents that you do not have permission to delete, you cannot delete the folder.

Work with Your Documents from Microsoft Office Applications

You can use the **File** menu of any of the Microsoft Office applications such as Microsoft Word, Microsoft Excel, Microsoft Project, and Microsoft PowerPoint to work with documents that are stored on your TeamForge site.

Open a TeamForge Document in Microsoft Office

You can get to your Microsoft Word, Excel, Project, and PowerPoint documents in CollabNet TeamForge even if the CollabNet TeamForge web interface is not open.

1. Open your Microsoft Office application.
2. From the **File** menu, choose **Open from CollabNet TeamForge**.
3. In the **Login** window, provide the following information:
    - Your TeamForge user name and password.
    - The URL of your organization's TeamForge installation.

    > **NOTE:** Do not enter the `http://` or `https://` before the server name.

4. Click **Log In**. The **Selection Dialog** window shows all of the projects, document folders, and documents that you have permission to view.
5. Select the document you want to open.

The document opens in your Microsoft Office application.

Save a Document to TeamForge

You can store your Microsoft Office documents in CollabNet TeamForge even if the CollabNet TeamForge web interface is not open.

CollabNet TeamForge works with Microsoft Word, Excel, Project, and PowerPoint.

1. Open your Microsoft application.
2. Open the document you want to save.

3. From the **File** menu, choose **Save to CollabNet TeamForge**.
4. In the **Log in** window, enter the following information:
    - Your TeamForge user name and password.
    - The URL of your organization's TeamForge installation.

    > **NOTE:** Do not enter the `http://` or `https://` before the server name.

5. Click **Log In**. The *Selection Dialog* window displays all of the projects and document folders in which you have permission to create documents.
6. Select the document folder in which you want to save the document.

    > **NOTE:** Do not click on an existing document. This will prompt you to update that document.

7. In the **Document Properties** window, clear **Use Filename**, then enter a name in the **CollabNet TeamForge Document Name** field. By default, the file name is used as the document name.
8. Write a description for the document.
9. Choose whether the document is in draft or final status.
10. To lock the document and prevent others from editing it, select **Lock Document**.
11. Write a version comment, if appropriate.
12. Click **OK**.

The document is now saved in CollabNet TeamForge. You can see the document by going to the CollabNet TeamForge document folder that you selected.

> **NOTE:** After you save a document to CollabNet TeamForge, furture saves will also save in CollabNet TeamForge. If you want to save a document to a local drive, choose **Save As** from your **File** menu.

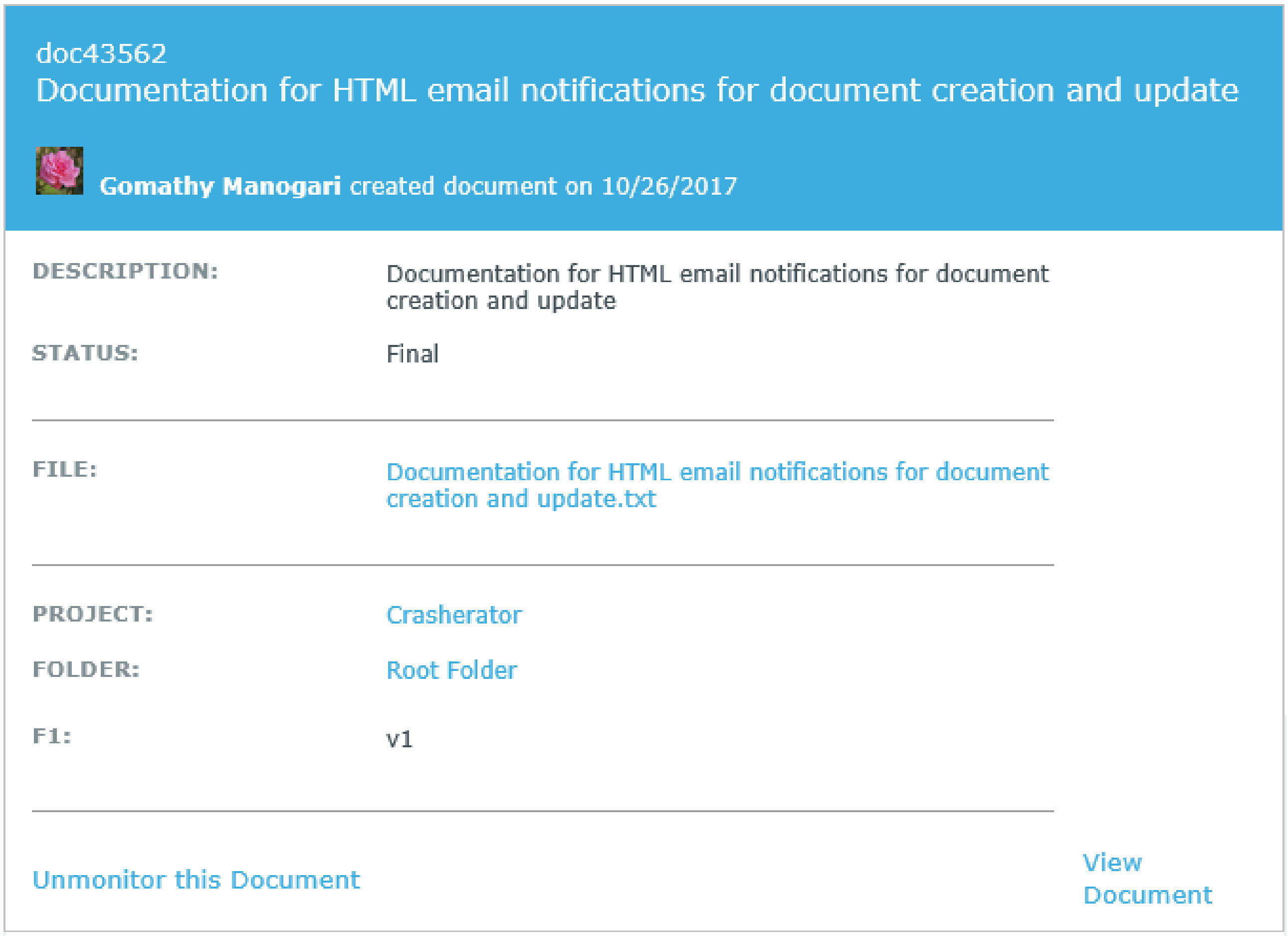
HTML Emails for Documents

Starting from TeamForge 17.11, when you create or update a document, TeamForge sends HTML emails to users monitoring that document or the document folder.

HTML emails are formatted emails that look like a newsletter that you receive from a web service. These emails are embellished with colors, graphics, table colums, and links. HTML emails have a better look and feel and override the simple and plain features of plain text emails.

By default, the HTML email configured for documents in TeamForge contains the document details such as document id, document title, description, status, file (document name with link), project, folder and so on. Details of the fields with null values are not shown in the email.

Sample HTML email when a document is created:



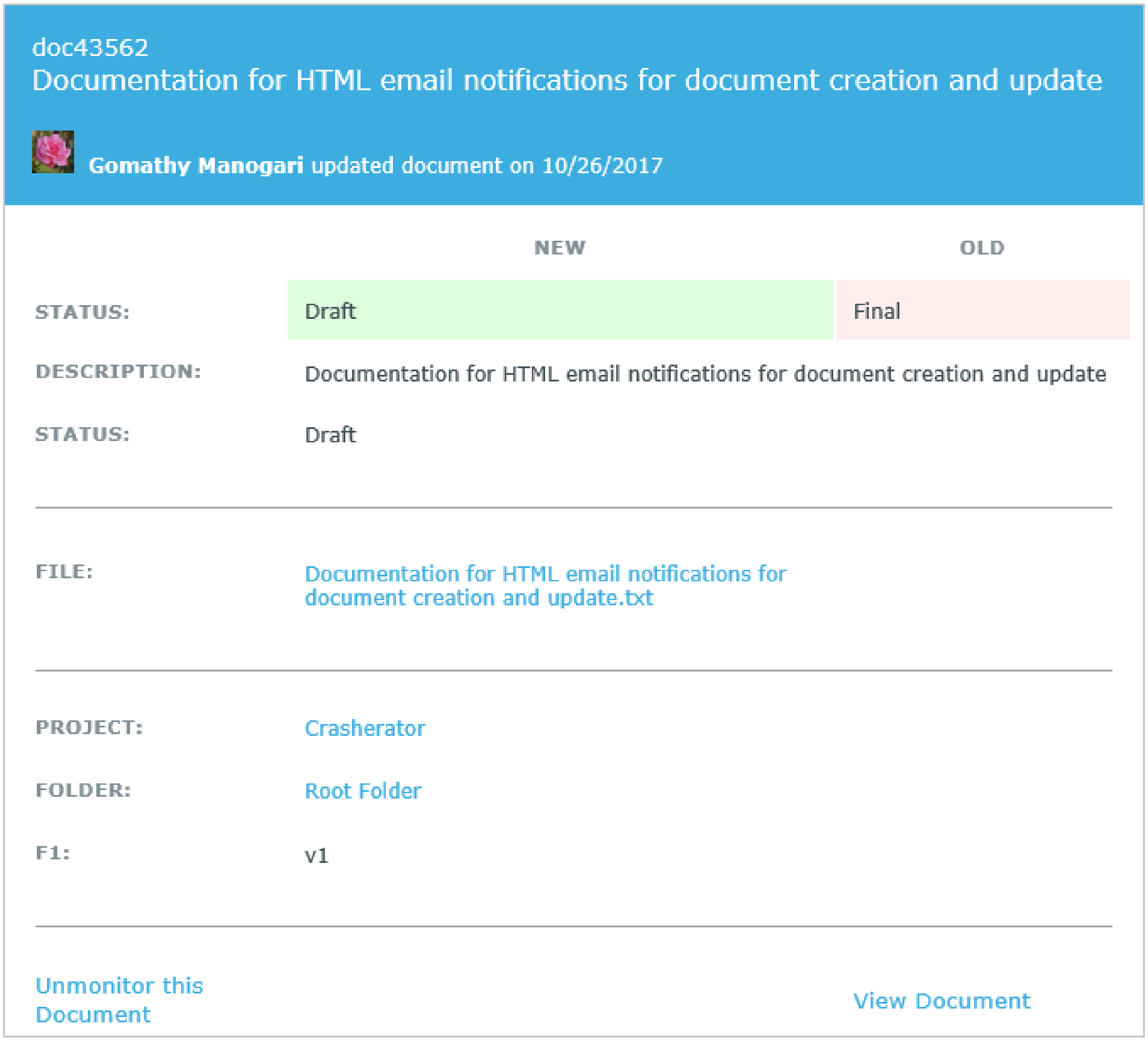
doc43562
Documentation for HTML email notifications for document creation and update

Gomathy Manogari created document on 10/26/2017

DESCRIPTION:	Documentation for HTML email notifications for document creation and update
STATUS:	Final
FILE:	Documentation for HTML email notifications for document creation and update.txt
PROJECT:	Crasherator
FOLDER:	Root Folder
F1:	v1

The email sent when a document is updated contains the new and old values for fields such as linked file name, description, Tags, status and so on.

Sample HTML email when a document is updated:



doc43562
Documentation for HTML email notifications for document creation and update

Gomathy Manogari updated document on 10/26/2017

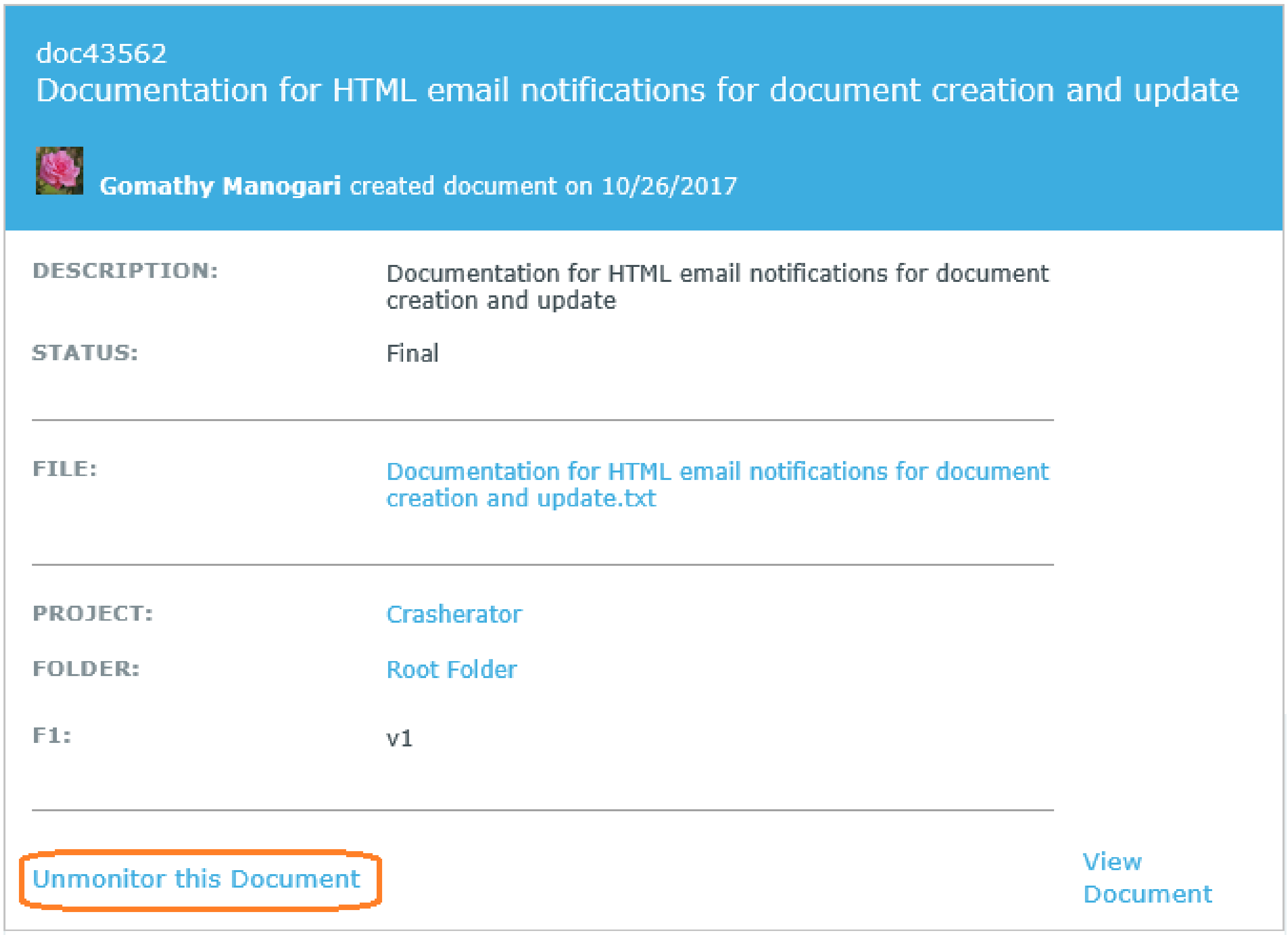
	NEW	OLD
STATUS:	Draft	Final
DESCRIPTION:	Documentation for HTML email notifications for document creation and update	
STATUS:	Draft	
FILE:	Documentation for HTML email notifications for document creation and update.txt	
PROJECT:	Crasherator	
FOLDER:	Root Folder	
F1:	v1	

NOTE: TeamForge's HTML emails are supported by Outlook for Windows, Outlook for Mac, and Office 365 web clients.

Unmonitor Documents via HTML Emails

You can now unmonitor a document by simply clicking the “Unmonitor this Document” link available in the document create/update email notifications.

The following screen shows the email that you receive as a monitoring user when a document is created.

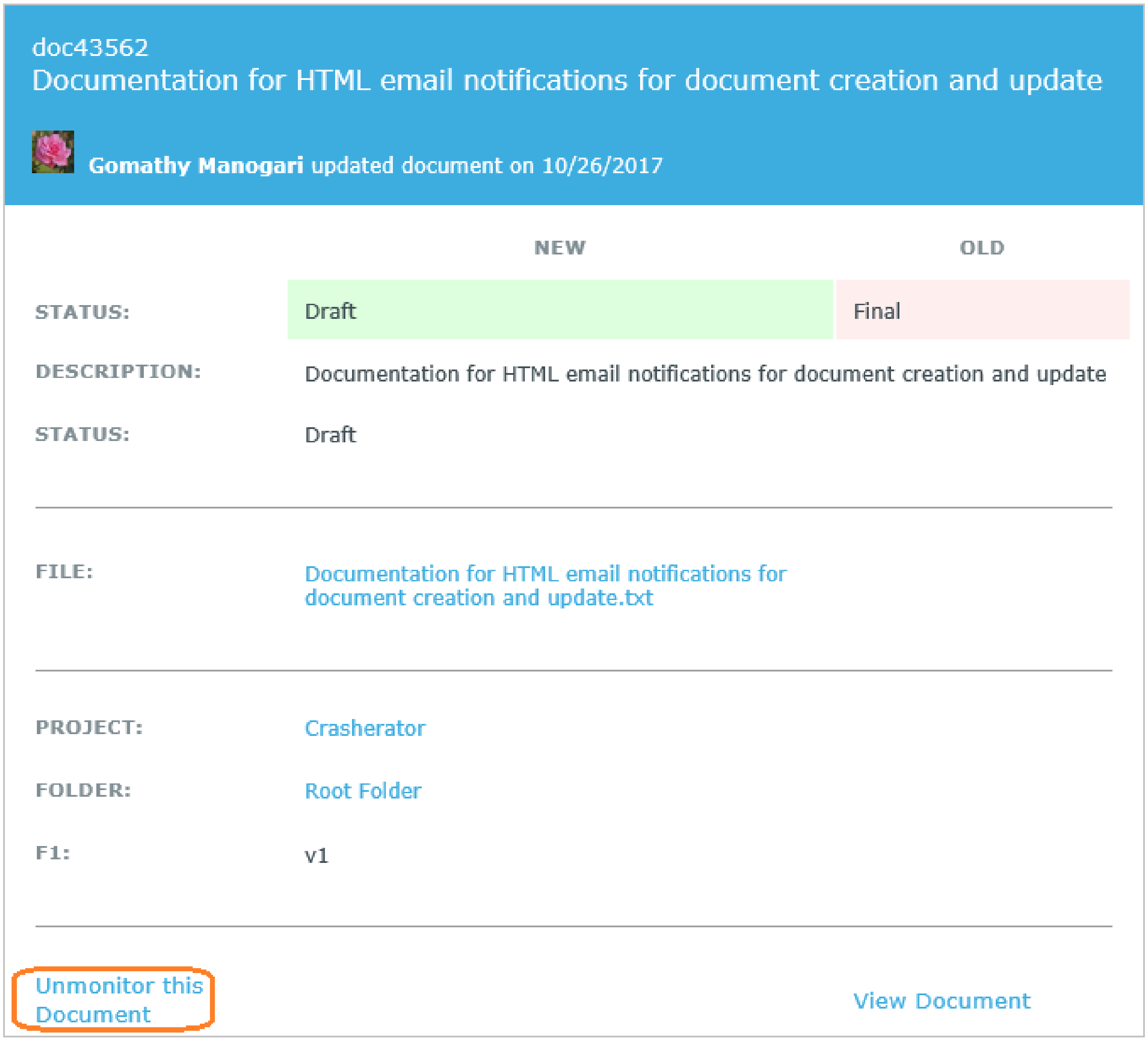


doc43562
Documentation for HTML email notifications for document creation and update

Gomathy Manogari created document on 10/26/2017

DESCRIPTION:	Documentation for HTML email notifications for document creation and update
STATUS:	Final
FILE:	Documentation for HTML email notifications for document creation and update.txt
PROJECT:	Crasherator
FOLDER:	Root Folder
F1:	v1

The following screen shows the email that you receive as a monitoring user when a document is updated.

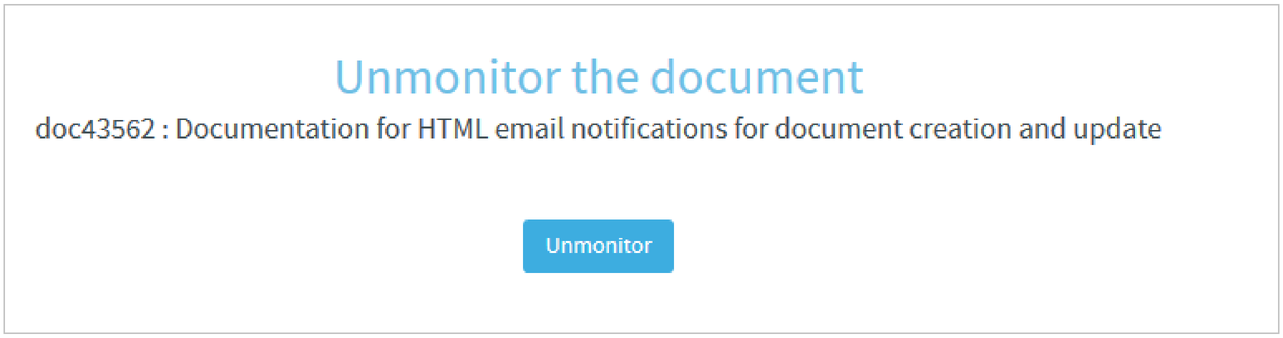


doc43562
Documentation for HTML email notifications for document creation and update

Gomathy Manogari updated document on 10/26/2017

	NEW	OLD
STATUS:	Draft	Final
DESCRIPTION:	Documentation for HTML email notifications for document creation and update	
STATUS:	Draft	
FILE:	Documentation for HTML email notifications for document creation and update.txt	
PROJECT:	Crasherator	
FOLDER:	Root Folder	
F1:	v1	

Clicking the **Unmonitor this Document** link shows the following dialog box:



Unmonitor the document

doc43562 : Documentation for HTML email notifications for document creation and update

Click **Unmonitor**, after which a success message is displayed.

In addition to this, a **View Document** link has been added at the bottom right of the HTML emails. Click this link to go the document details page of the document.

You can communicate with project members via discussion forums. Discussions provide workspaces where project members can discuss project-related topics online or by email. Forum administrators create forums and do what is needed to keep them on track, such as editing or moderating forum posts.

What is a Discussion Forum?

TeamForge discussions provide workspaces where project members can work together online or by email.

Discussion forums and mailing lists are closely integrated. Forum administrators can choose to enable a mailing list for each project forum. A mailing list extends the discussion forum functionality to allow project members to post messages to the forum using email.

Discussion forums can be public or private, depending on the forum's objective and desired level of access into the forum. Private discussion forums restrict anyone without specific access permissions from viewing or posting into the forum.

In a moderated discussion, messages from anyone except "trusted" users are screened by a moderator before they are posted. Messages posted by trusted users do not require the moderator's approval.

It's a good idea to use a discussion forum instead of direct personal email whenever privacy or security concerns don't prevent it, even when the communication only involves two project members.

- Members not directly involved often can make unexpected contributions if they are aware of the discussion.
- TeamForge archives all news items, forum posts, and mailing list communications, so you can go back and find valuable information later.

✓ Encourage project members to work together by creating discussion forums to which project members with the appropriate permissions can post messages.

✓ Discussion forums can also function as mailing lists.

✓ You can choose to make a discussion forum either public or private.

✓ You can be a forum or mailing list administrator without being a project administrator. Ask your project administrator or site administrator to grant you forum administration permissions.

✓ Forum administrators can enable or disable forum moderation and add or remove moderators and trusted users. Any project member with forum post permissions can be a moderator.

✓ Forum administrators can also make forums work like mailing lists.

✓ Guest users can monitor a forum if email monitoring is set to **Allow all site users and guests**.

✓ Additionally, guest users can email-post or subscribe to a forum if the mailing list is enabled and **Email Posting** is set to **Allow all site users and guests** from the discussion settings.

> **TIP:** Who can post by email to a discussion forum is controlled by the **Email Posting** options set for the forum. It does not depend on the permissions set for users of the Web forum.

Create/Rename/Edit a Discussion Forum

Create a Discussion Forum

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, click **Create**.
3. On the **Create Discussion Forum** page, enter a title and description for the forum.
4. To make it a private discussion forum, set the **TYPE** to **Private**.
   Private discussion forums restrict anyone without specific access permissions from posting to the forum. For example, you may want to restrict a preliminary planning discussion to your project's core team before sharing it more widely.
5. If you want the forum to work as a mailing list, select **Enable Mailing List**.
   1. Provide a name for the mailing list in the **EMAIL ADDRESS** field.
      The mailing list name must be unique within a project.
   2. Set who can post to the forum via emails.
      Choose either **User with Roles and Permissions** (default) or **Allow only forum admins** from the **Email Posting** drop-down list.
   3. Set who can subscribe to monitoring via emails.
      Choose either **User with Roles and Permissions** (default) or **Allow only forum admins** from the **Email Monitoring** drop-down list. Selecting **Allow only forum admins** for **Email Monitoring** will not restrict users with `Discussion-View` permission from getting monitoring emails in case they choose to monitor the forum via the web UI.
   4. Choose how replies to posts are handled by setting the **REPLY BEHAVIOR**.

      > **TIP:** Many users are accustomed to having their replies go automatically to the whole list. Others are used to having replies go just to the original sender. You should check with your users to see what makes more sense for a particular mailing list.
      >
      > When **REPLY BEHAVIOR** is set to `To the list`, email replies are sent to the list as a whole, not to the individual post. This may be a change from what some users are used to.

    5. Specify a prefix for the subject lines of messages from this list. This can help users sort their incoming messages, if they are subscribed to multiple lists.
    6. You may limit the size of emails (including attachments, if any). Enter the size (in MB) in the **MESSAGESIZE** field.
    7. Under **FOOTER TEXT**, provide any information you want to show up at the bottom of each email that subscribers receive. For example, you may want to offer useful web locations or email addresses.
6. To make this a moderated forum, select **ENABLE MODERATION**.
7. Click the **Search** icon to add moderators.
   A moderated discussion must have at least one moderator. If your project includes members of a parent project, you can select those members too.
8. Click the **Search** icon to add trusted users. Posts by trusted users do not need moderator approval.
9. Click **Save**.

If you set your forum to work as a mailing list, all project members monitoring the forum will receive notifications whenever a new topic or message is posted.

Rename or Edit a Discussion Forum

To help keep a forum or mailing list focused, try updating its title.

As a forum administrator, you can enable/disable mailing list or moderation features or just update their settings for the discussion forum.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, select the forum you want to change, and click **Edit**.
3. On the **Edit Discussion Forum** page, make your changes.
4. Click **Save**.

> **NOTE:** All project members monitoring the forum receive notifications of the update.

Subscribe to a Discussion Forum or Mailing List

When you monitor a discussion forum, you are notified of contributions to the forum by email. Monitoring a forum is the same thing as subscribing to a mailing list.

A bell icon in the **Monitoring** column indicates that you are subscribed to a forum.

- If you prefer to do it by email:

  - To subscribe in message by message type subscription, send an email to `<Email address name in Mailing list>-<project name>-subscribe@<domain name>`
  - To subscribe in digest type, send an email to `<Email address name in Mailing list>-<project name>-digest-subscribe@<domain name>`
  - You can also change the subscription type from message by message to digest and vice versa.
- If you prefer to do it through the web interface:
  - Click **Discussions** from the **Project Home** menu.
  - Click **Monitor**. Set the notification frequency using **Monitoring Preference** as it suits you.

Subscribe Others to a Discussion Forum or Mailing List

You can add other users to a discussion.

If you are a forum administrator, you can also add users to the discussion as a group.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. Select the forums that you want to add users to.
3. In the **Monitor** list, click **Users Monitoring Selected**.
4. On the **USERS** tab, Click **Add**.
5. On the **Find a User** window, move the users you want into the **USERS TO ADD** column.

> **TIP:** To add users as a group (assuming you are a forum administrator), do the same operation on the **USER GROUPS** tab.

Create a Discussion Forum Topic

Create a new forum topic to begin discussion of a new subject.

A topic starts a message thread to which other users can reply. A forum can have any number of topics.

A forum topic is similar to an email, in that you can use it communicate with other people subscribed to the forum, as if it were an email list. If the forum owner has enabled the forum to work as a mailing list, then you can post to the forum by email as well.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, click the title of the forum in which you want to create a topic.
3. On the **Topic Summary** page, click **Create**.

4. On the **Create a Topic** page, describe the topic in the **Subject** field.
5. Type the message in the Message field. After the topic is created, other users can reply to this message.
6. If you want the message sent by email to people who are not members of the forum, add their email addresses to the **Other Recipients** field. If there is more than one, put commas, semicolons or spaces between them.

> **NOTE:** In a moderated discussion forum, addresses in the **Other Recipients** field get your message only after the moderator approves the message.

7. To add an attachment to the topic, click **CHOOSE FILE** and select the file.
8. Click **Save**.

> **NOTE:** If this discussion forum is moderated, the topic is held until a moderator approves or rejects it. (Except if it is from a trusted user, these messages don't require moderation.)

Reply to a Discussion Forum Message

You can post a message in any topic in any forum you have access to. You can also post a message in reply to another message.

If the forum is moderated, you must have posting permission. Contact the forum moderator.

Tip: If you are getting your forum messages delivered as email, you can reply to a post by email too.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, find the topic in which you want to post a message.
3. In the section containing the message to which you want to respond, do one of the following:
    - Click **Quote** to quote the original message in your response message.
    - Click **Reply** to omit the original message from your response message.
4. Write the message.
5. To add an attachment to the message, click **CHOOSE FILE** and select the file.
6. Click **Save**.

The forum message is posted. Other project members can reply to it using the same process.

Moderate Discussion Forum Posts

A message to a moderated discussion forum is held until a moderator acts on it. (Except if it is from a trusted user. These messages don’t require moderation.)

As a moderator, you get an email when a message is awaiting moderation. The email contains the URL where you can approve or reject the message.

Add or Modify Moderators

As a forum administrator, you can add or remove forum moderators.

If a forum is moderated, it must have at least one moderator.

When you designate a forum moderator, you also become a moderator yourself.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, select the forum for which you want to add or modify the moderators.
3. On the **Topic Summary** page, click **Edit**.
4. On the **Edit Discussion Forum** page, add or modify the forum moderators.

> **NOTE:** The existing moderators are listed.

5. Click the **Search** icon to add or remove forum moderators. You can select one or more moderators.

> **NOTE:** You can select the inherited project members also from the list.

6. To add moderators, select the required users, click **Add** and click **OK**.
7. To remove moderators, select the required users, click **Remove** and click **OK**.
8. On the **Edit Discussion Forum** page, click **Save**.

Moderate a Discussion Forum by Email

If your discussion forum is also a mailing list, you can approve or reject the post by email.

See the notification email for your options.

Option	Description
To accept the message	Send an email to <postId>-accept@<domain>, or just click **Reply**.

Option	Description
To accept the message and add the sender to the “Trusted Users” list	Send an email to <postId>-allow@<domain>, or click **Reply-to-All**.
To reject the message	Send an email to <postId>-reject@<domain>.
To add your comments to the message	Include your comments in the **Start Comment** and **End Comment** blocks in your response email. Your comments will appear in the approved post.

Approve a Forum Post

If a post is appropriate, you can add it to the forum by approving it.

If your discussion forum is also a mailing list, you can approve or reject the post by email. See the options in the notification email.

If the sender consistently contributes useful and appropriate input, you can save the moderation time by designating that user as a `trusted` user. Posts from trusted users don’t have to be moderated.

1. On **My Page**, click the **ITEMS AWAITING MY APPROVAL** tab. The number of posts awaiting approval are displayed against the project names.
2. To view forum details, click the hyperlinked **Number of Posts** for your project.
3. Click the **Number of posts awaiting approval** link on the **Forum Summary** page.

> **NOTE:** The Forum Summary page displays all the forum names and the corresponding number of posts awaiting approval, if any.

4. Select either of the three approval techniques in the **Posts Awaiting Approval** tab.

   As the moderator of the post, you will be able to view the topic title in the **All Topics** tab; and post title in the **Posts Awaiting Approval** tab on the **Topic Summary** page.

> **NOTE:** In the **All Topics** tab, the hourglass icons differentiate the topics that contain posts awaiting approval. To view posts nested within a topic, you can use the hyperlinked topic title.

Option	Description
To approve posts and senders individually	Select the post and click **Approve** or **Approve and Trust**.
To approve multiple posts and senders at once	Select all the posts to be approved and click **Approve** or **Approve and Trust** below the post details.

Option	Description
To view the post details and approve	Click the post title and click **Approve** or **Approve and Trust** on the **Review Post Awaiting Approval** page after reading the details.

Reject a Forum Post

If a proposed message is not appropriate or does not contribute to the goals of your discussion forum, you can reject it.

If your discussion forum is also a mailing list, you can approve or reject the post by email. See the options in the notification email.

You can reject posts with or without comments or reasons for rejection.

1. On **My Page**, go to **ITEMS AWAITING MY APPROVAL** tab.
2. On the **Forum Summary** page, click **Number of posts awaiting approval**. Topics that contain posts awaiting approval have an hourglass icon.
3. On the **Posts Awaiting Approval** tab, choose your method of rejection.

Option	Description
To reject the posts individually	Select the each post, then click **Reject** at the end of the post.
To bulk-reject the posts	Select all the posts you want to reject, then click **Reject** below the post details table.
To view the post details and reject	Click the hyperlinked post title, then click **Reject** on the **Review Post Awaiting Approval** page.
To explain your rejection with a comment	Click **Reject With Comment** instead of **Reject**.

NOTE: The rejection comment is posted to the message sender.

Rejected messages are deleted from the posts awaiting approval list and the message senders are notified by email.

Stop Moderating a Forum

If a moderated discussion forum does not require posts to be moderated anymore, moderation can be disabled.

To change a moderated discussion forum to unmoderated discussion forum in CollabNet TeamForge, select the forum and turn off its moderation feature.

- Users with forum admin permissions only can enable/disable moderation.

- On disabling moderation on a moderated forum, the posts awaiting approval are automatically approved.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, select the moderated forum that does not require moderation anymore.
3. On the **Topic Summary** page, click **Edit**.
4. On the **Edit Discussion Forum** page, to disable moderation, de-select the **ENABLE MODERATION** check box. The following message is displayed:

   ```
   Any post awaiting moderation will be approved automatically.
   ```

5. Click **Save** to turn off the moderation. To keep moderation enabled, select **Cancel**.

   The moderated discussion forum changes to an unmoderated discussion forum and any posts sent to the forum will be displayed without any restrictions.

Post to a Forum by Email

If the forum also works as a mailing list, you can create a forum topic by sending an email message to the forum. You can also reply to a post by replying to the email.

If the forum is moderated, you must have posting permission. Contact the forum moderator.

The forum’s email address, if it has one, appears on the Topic Summary page.

> **TIP:** If a forum is not set up to work as a mailing list, you can still get posts by email when you monitor the forum. See Monitor many items.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, click the forum in which you want to create a topic.
3. On the **Topic Summary** page, find the email address in the **Mailing List** field. Send your email message to that address. CollabNet TeamForge maps your email to the forum topic like this:

Email Field	Forum Topic Field
To	Forum email address
Subject	Title of forum topic
Body	Text of forum topic
Attachments	Attachments

Emails from the forum can be read in RTF (Rich Text Format) or HTML format. The format of the message is delivered as an attachment to the post. Embedded attachments, such as text or images, are also delivered.

In a moderated discussion forum, if you add other addresses in the `cc:` field of your email, those addresses get your email only after the moderator approves the message.

Search for Posts

Search for a post by using keywords, specifying the forum it belongs to, selecting the sender of the post or entering relative date or date range.

You can search for posts either across all forums or within specific forums.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, click **Search Posts**.
3. On the **Post Search Criteria** page, enter the desired search criteria.

   > **TIP:** You can use wildcards.

   - To search by subject, body or attachments, enter the text in the **Search Text** field and select **Subject**, **Body** and/or **Attachments** options.
   - To search by forum name, select the forum in the **Forum** field.

     > **NOTE:** When none of the forum options are selected, the system searches for matching posts across all forums.

   - To search by post-sender, select the user name in the **Posted By** field.

     > **NOTE:** You can select **User running search** option to display your posts.

   - To search by time span, specify relative dates such as “Within the last 7 days”.
   - To search by date range, enter the start and end dates for the search. Click the Calender icon to select dates from a calendar.
4. Click **Search**.

All posts matching your search criteria are displayed in the *Search Results* page.

Associate Forum Messages with Other Items

When a forum message concerns some other CollabNet TeamForge items, such as a document, a tracker artifact, a file release, a code commit, or a task, link the message to the item under discussion by creating an association.

Creating association between items enables you to define relationships, track dependencies, and enforce workflow rules.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, find the forum message with which you want to create an association. Any existing associations are displayed in the **Associations** section.
3. Click **Associate**.
4. In the **Add Association Wizard** window, select the items with which you want to associate the artifact:
    - **ENTER ITEM ID** - If you know the item's ID, you can enter it directly.

      ✓ To associate an object in an integrated application from within TeamForge, use the `[<prefix_objectid>]` format. Successful associations appear hyperlinked.

      ✓ Each integrated application displays its prefix on moving the mouse over the application name in the tool bar.
    - **ADD FROM RECENTLY VIEWED** - Select one of the last ten items you looked at during this session.
    - **ADD FROM RECENTLY EDITED** - Select one of the last ten items you changed.
5. Click **Next**.
6. You may add a comment in the **ASSOCIATION COMMENT** text box.
7. Save your work.
    - Click **Finish and Add Another** to add additional associations.
    - Click **Finish** to return to the **Details** page.

   ✓ When an association is added to or removed from TeamForge objects such as tracker artifacts, tasks, documents, discussions, and file releases, a notification mail is sent to users monitoring these objects.

   ✓ An option is provided at site level and user level to make sure whether the notification mail has to be sent or not. For more information on this, see Configure your Site's Settings.

Delete Forum Messages and Topics

Delete a Forum Message

When a message in a forum is off topic or potentially harmful, you may want to delete it.

Before deleting a forum message, consider leaving it in place so that future users can consult it if they need to. Consider this even if the message does not seem very useful right now.

{include important.html content=”You cannot delete the topic starter’s original message without deleting the entire topic.” %}

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, find the forum message that you want to delete.
3. Click **Delete** and confirm that you want to delete the forum message.

The forum message is deleted.

Delete a Forum Topic

If you no longer want a forum topic in your project, you can delete it.

> **WARNING:** Deleting a forum topic deletes all of the forum messages in the topic. Delete a forum topic only if you are sure that you no longer need any of the forum messages in it.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, click the title of the forum containing the topic that you want to delete.
3. On the **Topic Summary** page, select the topic you want to delete.
4. Click **Delete** and confirm that you want to delete the topic.

The forum topic is deleted.

The TeamForge Wiki allows you to create an unlimited number of Wiki pages in each TeamForge project. A Wiki page is a tool for managing project information as unstructured, linkable content.

What is a Wiki Component?

A wiki component lets you link to an existing wiki page from a project page.

Start a Wiki

To start communicating with other project members via Wiki, create a new page in your project's Wiki.

Every TeamForge project starts with a blank Wiki. You do not need to create a Wiki before you can begin adding content. After a Wiki is started, any user with the appropriate permissions can add or edit content. However, you cannot delete all of the content to start over with a new, blank Wiki.

1. Click **Wiki** from the **Project Home** page.
2. On the Wiki home page, click **Edit**.
3. On the **Edit Wiki** page, write your Wiki text. Wiki content is a combination of plain text, markup for font elements such as bold or italics, headers, bulleted and numbered lists, and links.
4. Customize your Wiki entry with any of these optional steps:
    1. Use the buttons at the top of the text area to add Wiki markup to your text in the **WYSIWYG Editor** mode. You can also enter Wiki syntax directly into the text area in the **Plain Editor** mode. For an explanation of Wiki syntax, click Syntax Reference.
    2. To insert a link to another TeamForge item, just type the item id. You do not need additional Wiki syntax to create the link.
    3. To change the size of the display window, drag the arrow available at the bottom right of the window.
    4. To attach an external file to a wiki page, click **CHOOSE FILE**, then browse for the desired file.
    5. To add a version comments, write it in the **Version Comment** field.
5. Click **Preview Changes** to see how your Wiki content will look. You can make further edit from the Previewing Home page before saving your changes.
6. Click **Update** to save your changes.

Add Wiki Content

When the information you want to share with other project members does not file neatly into a tracker comment or a document review, use a Wiki page for a more free-form communication flow.

After a Wiki is created, any user with the Wiki create, edit, or view permission can add or edit Wiki content. You can also add associations or multiple attachments, or create additional Wiki pages.

NOTE: You can now add multiple attachments in a Wiki page without updating the Wiki each time to include another attachment. If attachments are not required, they can be deleted before updating the Wiki page.

1. Click **Wiki** from the **Project Home** menu. Any existing Wiki content appears.
2. For more information about this Wiki, click **View Details**.
3. On the Wiki home page, click **Edit**.
4. On the **Edit Wiki** page, make your changes or additions to the Wiki content.
   - If you prefer to use buttons to do your text formatting, the way you would with a word processor, click [NAME OF BUTTON].
   - If your tastes run more to typing in your wiki formatting, see Wiki Syntax for the choices available.

   TIP: You can use a variety of preconfigured queries to generate up-to-date content for your Wiki page. For more preconfigured Wiki content, see Wiki Syntax.

5. In the **Version Comment** box, note the reason for your change. It is options, but advisable, to get in the habit of recording a version comment. If your project manager has made it mandatory, then you must record a version comment before you can save your changes.
6. Click **Preview Changes** to see how your Wiki content will look. YOu can make further edits on the **Previewing** page before saving your changes.
7. Click **Update** to save your changes.

Create a New Wiki page

A TeamForge site can have any number of Wiki pages. All Wiki pages are linked, and their relationships are traced on the Back Links tab of each Wiki page.

1. Click **Wiki** from the **Project Home** page.
2. In the large text field, insert the title of your new page between square brackets, like this:

   ```
   Please post comments about [test results] here.
   ```

   Then click **Update**. The text between the square brackers becomes a link on your Wiki page.

   NOTE: You can also type the new link in CamelCase (each word starts with an uppercase letter, no spaces) and skip the square brackets.

3. Click the link to open the new Wiki page. The new page is created. The title of the new page is the same text as the link, rendered in CamelCase.
4. On the **Create Wiki** page, write the content you need, then click **Save**.

> **TIP:** You can use a variety of preconfigured queries to generate up-to-date content for your Wiki page. For more preconfigured Wiki content, see Wiki Syntax.

The new Wiki page is created. The back link to the page from which it was created appears on the *Back Links* tab of the **Show/Hide Details** section.

Search a Wiki

Use the wiki search page to find content in a project wiki.

1. Click **Wiki** from the **Project Home** page.
2. On the **Wiki home** page, click **Search Wiki Pages**.
3. Try one of the predefined searches. These can save you time by running some of the most widely used content searches with a single click:
   - List the wiki pages that have changed in the last 15 days.
   - List the wiki pages that no other wiki page links to.
   - List all the pages in this project’s wiki.
4. If you need to narrow your search beyond the predefined searches, enter some search terms under *Wiki Pages Search Criteria*. Wildcards are allowed.
   - To search active wiki content, enter the search keywords in the **SEARCH TEXT** field.
   - To search active and inactive wiki page versions, select **Search All Versions**. (By default, searches are performed on active wiki page versions only.)
   - To search wiki attachments, select **Include Attachments**.
   - If you know approximately when the wiki content was created or last edited, enter the start and end dates for the search and click **Search**. Click the calendar icon to select dates from a calendar.
   - To search by author, click the Search icon in the **CREATED OR EDITED BY** field that displays a list of project members.
5. Click **Search**.

A list of wiki pages matching your search criteria appears.

View a Wiki page as HTML

One way to resolve messy or incorrect formatting on a wiki page is by converting the page to HTML.

1. Click **Wiki** from the **Project Home** page.
2. Find your wiki page by navigating or searching.
3. Click the **View HTML** button.

The wiki uses **HTML Tidy** to display the cleanest HTML it can manage. View the page source to see the results.

View a Wiki page as a PDF

When you want to send a wiki page to someone outside the project, it can be handy to convert it to a PDF document.

1. Click **Wiki** from the **Project Home** page.
2. Find your wiki page by navigating or searching.
3. Click the **View PDF** button.

Depending on how your browser is set up, the resulting PDF document appears or the browser offers to download it for you.

To help keep track of changes to a Wiki page, require users to record a comment about their changes when they save the page.

1. On any project page, click **PROJECT ADMIN** from the **My Page** menu.
2. On the **Project Settings** page, click **Wiki Settings**.
3. Select **REQUIRES VERSION COMMENT ON EDIT**.

Now users who edit a page in your project's Wiki must leave a note describing their changes.

You can use special wiki text markup to do a wide variety of cool and useful things on TeamForge wiki pages.

For a quick introduction, see the documentation for JSPwiki, which is the wiki engine that drives the TeamForge wiki tool.

Text Effects on Wiki Pages

Wiki markup is great for making text look the way you need it to look.

> **NOTE:** These tools are for use only when you are editing a wiki page in text mode. If you try to use them in WYSIWYG mode, they are displayed just the way you typed them in, which is not what you want.

Syntax	Effect	Details
- - - -	Creates a horizontal rule.	
\ \	Creates a line break.	
!!!text	Creates a level 1 (large) header.	
!!text	Creates a level 2 (medium) header.	
!text	Creates a level 3 (small) header.	
’ ‘text’ ‘	Creates italic text. (That’s two single quotes on each side.)	
__text__	Creates bold text. (That’s two underscores on each side.)	
{{text}}	Creates monospaced text.	
*text	Creates a bulleted list item.	
#text	Creates a numbered list item.	
;term:ex	Creates a definition for the word “term’“with the explanation “ex.”	
{{{text}}}	Creates pre-formatted text.	
%%(<css-style>)<your text>%%	Defines a CSS style command.	%%(font-size: 150%; color: red;) Hello, world!%%
Blank line	Starts a new paragraph.	

Bring TeamForge Data into Wiki Pages

Use this markup format to bring in data from elsewhere on your TeamForge site.

> **NOTE:** These are for use only when you are editing a wiki page in text mode. If you try to use them in WYSIWYG mode, they are displayed just the way you typed them in, which is not what you want.

Syntax	Effect	Details
[{INSERT ExcelToHTMLPlugin src='c:\somesheet.xls'}] or [{INSERT ExcelToHTMLPlugin border='1' src='\\the_server\somesheet.xls'}]	Reads a Microsoft Excel file and displays it as an HTML table.	Parameters: • src: URL / Attachment file name • srcsheet: Sheet name • height: height attribute for the html table • width: width attribute for the html table • border: border attribute for the html table More at http://www.ecyrd.com/JSPWiki/wiki/ExcelToHTMLPlugin

Text Navigation Tools for Wiki Pages

You can use wiki syntax to help readers get around.

> **NOTE:** To use these tools, copy and paste the sample syntax into your Wiki page in “Plain Editor” mode, then customize it appropriately.

Syntax	Effect	Details
%%insert-toc%%	Creates a table of contents consisting of the header text on the page.	
[link]	Creates a link to a new Wiki page called "link."	If the link is a complete URL, a link to the URL is created. If the link points to a .gif, .jpg, or .png image, the image is rendered directly in the page.
[title|link]	Creates a link to a new Wiki page called "link" with the text "title" displayed for the URL.	If the link is a complete URL, a link to the URL is created. If the link points to a .gif, .jpg, or .png image, the image is rendered directly on the page with "title" as ALT text.
~TestText	Disables link creation for a CamelCase word.	CamelCase words are two or more uppercase words with no spaces. By default, a CamelCase word automatically creates a link to a new Wiki page.
[[link]	Creates the text "[link]."	
[{IFramePlugin url='http://open.collab.net/' width='100%' height='500' border='1' scrolling='yes' align='center'}]	Embeds an iframe into a wiki page.	• attachment: Attachment path, e.g. 'IFramePlugin.jar(info)' • url: A URL, e.g 'http://www.google.com' • align: Align the iFrame to left/center/right • border: Whether there is a border or not • width: Width of the iFrame • height: Height of the iFrame • marginwidth: Margin width of the iFrame • marginheight: Margin height of the iFrame • scrolling: Whether the iFrame can be scrolled or not See http://www.ecyrd.com/JSPWiki/wiki/iFramePlugin
Tab Completion <Keyword+Tab>	You can type a keyword and hit the Tab key (under the Tab Completion mode). The	

	editor will fill in with a sample template for the specific markup represented by the keyword.	• link:Inserts a sample link. • h1:Inserts level 1 heading sample. • h2:Inserts level 2 heading sample. • h3:Inserts level 3 heading sample. • bold:Inserts a bold text sample. • italic:Inserts an italics text sample. • mono:Inserts a mono text sample. • mono:Inserts a mono text sample. • sup:Inserts a superscript sample. • sub:Inserts a subscript sample. • strike:Inserts a strike through text sample. • br:Inserts a line break. • hr:Inserts a horizontal line. • pre:Inserts a pre-formatted text sample. • code:Inserts a code block sample. • dl:Inserts a definition list block sample. • toc:Inserts the Table of Contents plugin syntax. • tab:Inserts a sample tabbed section block syntax. • table:Inserts a sample table syntax. • img:Inserts a sample image plugin syntax. • quote:Inserts a sample quoted text block. • sign:Inserts the user's signature.

Attachments for Wiki Pages

You can use wiki markup to bring in information from external sources.

NOTE: These tools are for use only when you are editing a wiki page in text mode. If you try to use them in WYSIWYG mode, they are displayed just the way you typed them in, which is not what you want.

Syntax	Effect	Details
[WikiPageName/ attachmentName]	Embeds an attachment in the page.	If the attachment is a .gif, .jpg, or .png image file, the attachment will be embedded in the page; otherwise, the name of the attachment will display as a downloadable link. After adding attachments, the exact syntax for including the current page's attachments is shown next to each attachment's name in the Attachments section of the Edit Wiki page. You can use the same syntax to embed attachments from other wiki pages in the same project.
[{InsertAttachment page='WikiPage/attachment'}]	Inserts the contents of an attachment (text file) into a page.	If the attachment is a text file, the content of the text file is inserted into the page. For more information, click here. • This markup is for inserting content of a text file. • Files with the following extensions are allowed as attachment: .txt, .html, .xml, .cpp and .java. • In addition to image files, inserting files with the following extensions is not supported by this markup: .doc, .xls, .pdf, .zip, .jar.
[{Mediaplayer src='fileName.wmv'}]	Embeds a Windows Media Player or Quicktime Player on a wiki page.	• src: Media URL / Attachment file name • playertype: "mediaplayer" / "quicktime" • width, height: Dimension of the embedded media displayed • movieheight, moviewidth: Dimension of the display screen • caption: Caption to be displayed below the media player • control: Displays Control bar. mediaplayer: 1 (Show) / 0 (Hide); quicktime: true (Show) / false (Hide) • autostart: Play automatically. mediaplayer: 1 (Auto) / 0 (Manual, Click to play); quicktime: true (Auto) / false (Manual, Click to play) • autorewind: Automatically rewinds when play ends. mediaplayer: 1 (Auto Rewind) / 0 (Play once); quicktime: true (Auto Rewind) / false (Play once) • playcount: Number of times the movie will play. 0 represents always play.

		See http://www.ecyrd.com/JSPWiki/wiki/MediaPlayerPlugin
[{INSERT ExcelToHTMLPlugin src='WikiPage\somesheet.xls'}] or [{INSERT ExcelToHTMLPlugin border='1' src='\ \the_server\somesheet.xls'}]	Reads a Microsoft Excel file and displays it as an HTML table.	Parameters: • src: URL / Attachment file name • srcsheet: Sheet name • height: height attribute for the html table • width: width attribute for the html table More at http://www.ecyrd.com/JSPWiki/wiki/ExcelToHTMLPlugin
[{arnaud.Flash src='yourAttachedFlash.swf'}]	Embeds a Flash Player on your wiki page.	• width='n' • height='n' • controls='true|false' • play='true|false' • loop='true|false' • parameters='param1=value1, &param2=value2' See http://www.ecyrd.com/JSPWiki/wiki/AFlashPlugin
[{IFramePlugin url='http:// open.collab.net/' width='100%' height='500' border='1' scrolling='yes' align='center'}]	Embeds an iframe into a wiki page.	• attachment: Attachment path, e.g. 'IFramePlugin.jar(info)' • url: A URL, e.g 'http://www.google.com' • align: Align the iFrame to left/center/right • border: Whether there is a border or not • width: Width of the iFrame • height: Height of the iFrame • marginwidth: Margin width of the iFrame • marginheight: Margin height of the iFrame • scrolling: Whether the iFrame can be scrolled or not

		See http://www.ecyrd.com/JSPWiki/wiki/iFramePlugin

Tables on Wiki Pages

You can use wiki markup to organize information in tables on a wiki page.

> **NOTE:** These tools are for use only when you are editing a wiki page in text mode. If you try to use them in WYSIWYG mode, they are displayed just the way you typed them in, which is not what you want.

Syntax	Effect	Details
[{Table <table-parameters> || Table Header Example || More… | Table Data Example | More… }]	Inserts a table on the wiki page. See this page for the markup for table elements.	• **rowNumber**: *<integer>* - Row number starts counting at this value; default = 0 (used in conjunction with '#' syntax) • **style**: *<css-style>* Add formatting to the table, e.g. style:'border=2px solid black;' • **dataStyle**: *<css-style>* Format all data cells (prefixed by single pipe signs '|') • **headerStyle**: *<css-style>* Format all header cells (prefixed by double pipe signs '||') • **evenRowStyle**: *<css-style>* Format the even rows, e.g. evenRowStyle='background: #ffff00;' • **oddRowStyle**: *<css-style>* Format the odd rows, e.g. oddRowStyle='color: red;'
||head1||head2	Creates a table column with header text "head1" in the first cell and "head2" in the second cell.	

<table>
<tr><td>|col1|col2</td><td>Creates a table row containing the text "col1" in the first cell and "col2" in the second cell.</td><td></td></tr>
<tr><td>|<</td><td>Collapses a cell with the previous cell so it spans multiple columns.</td><td>http://www.ecyrd.com/JSPWiki/wiki/TablePlugin</td></tr>
<tr><td>||<</td><td>Collapses a header cell with the previous header cell so it spans multiple columns.</td><td>http://www.ecyrd.com/JSPWiki/wiki/TablePlugin</td></tr>
<tr><td>|^</td><td>Collapses a cell with the cell above so that it spans multiple rows.</td><td>http://www.ecyrd.com/JSPWiki/wiki/TablePlugin</td></tr>
<tr><td>||^</td><td>Collapses a header cell with the header cell above so that it spans multiple rows.</td><td>http://www.ecyrd.com/JSPWiki/wiki/TablePlugin</td></tr>
<tr><td>(<css-style>)</td><td>Inside a table cell, adds specific formatting to a cell.</td><td>http://www.ecyrd.com/JSPWiki/wiki/TablePlugin</td></tr>
<tr><td>#</td><td>Inside a table cell, displays the current row number with auto row numbering.</td><td>http://www.ecyrd.com/JSPWiki/wiki/TablePlugin</td></tr>
<tr><td>[{INSERT ExcelToHTMLPlugin src='WikiPage\somesheet.xls'}] or [{INSERT ExcelToHTMLPlugin border='1' src='\\the_server\somesheet.xls'}]</td><td>Reads a Microsoft Excel file that is attached to a wiki page, and displays it as an HTML table.</td><td>Parameters:
• src: If the Excel file is attached to the current wiki page, this is the attachment file name. If it is attached to some other wiki page, this is the URL of the attachment.
• srcsheet: Sheet name
• height: height attribute for the html table
• width: width attribute for the html table
• border: border attribute for the html table
• cellpadding: cellpadding attribute for the html table
• cellspacing: cellspacing attribute for the html table
• background: background attribute for the html table. Attachment file name is accepted as value
• backgroundcolor: backgroundcolor attribute for the html table</td></tr>
</table>

		• keepformat: Formating specified in the excel sheet is applied for the html table. **NOTE:** The complete formating from excel sheet is not applied to the html table, for example, font, font size, etc are not applied and background color, foreground color, etc are applied. yes/no is accepted as value • headercolor: Foreground color for the header (eg: #dcdcdc) • headerbackgroundcolor: Background color for the header (eg: #adadad) • evenrowcolor: Foreground color for the even rows (eg: #adadad) • evenrowbackgroundcolor: Background color for the even rows (eg: #adadad) • oddrowcolor: Foreground color for the odd rows (eg: #adadad) • oddrowbackgroundcolor: Background color for the odd rows (eg: #adadad) • tableclass: Style class name for the HTML table • headerclass: Style class name for header • evenrowclass: Style class name for the even rows • oddrowclass: Style class name for the even rows • stylesheet: Stylesheet for the table. Attachment file name is accepted as value See href="http://www.ecyrd.com/JSPWiki/wiki/ExcelToHTMLPlugin"
[{TableOfContents }]\\	Creates a table of contents consisting of the header text on the page.	

Wiki Plugins

You can use pre-defined wiki plugins to format your current page.

Plugin Name	Details
Section Headings Template	Inserts a page template that includes a table of contents and several section headings. Typically, this can be used in a new or blank page.
Sortable Table	Inserts a new table that can be sorted when you click on the column headers.
Zebra Table	Inserts a new table that has alternating background colors for each row.
Table of Contents Plugin	This plugin automatically generates table of contents that provides links to all the headings on your page.
Insert Page Plugin	This plugin will insert a copy of another page into the current page. You must specify the name of the page to insert.
Current Time Plugin	This plugin displays the current date and local time of the server when the page is viewed.
Insert Attachment	The insert attachment plugin allows you to insert the contents of an attachment into a page.
Media Player	This is an embedded player in your wiki page, supporting Windows Media Player and Apple QuickTime.
Insert Table	Additional table support including multi-line table editing, cell merging, and automatic row numbering.
Flash Player	An embedded flash player for the wiki page.
Insert Excel	Allows you to insert a Microsoft Excel file as a table.
Insert iFrame	Allows embedding attachments, external urls, and files (relative to the docbase).
Code to HTML	Allows source code syntax to be rendered as a HTML output and supports syntax from 130 different programming languages.
Index Plugin	Displays all the pages in wiki in a alphabetical order.
Recent Changes Plugin	Inserts the latest changes in order.
Referred Pages Plugin	Finds and lists all the pages that are referred to by the current page. The depth parameter allows to display a recursive tree of referred pages.
Referring Pages Plugin	Finds and lists all the pages that refer to the current page.
Undefined Pages Plugin	Lists all the pages that are referred to, but not yet created.
Unused Pages Plugin	Lists all the pages that are not currently referred to by any other page.
ExcelToHTMLPlugin	Provides a HTML view for the Excel spread sheet files.
PDFPlugin	Provides the PDF output for the files.
WikiContentToHTML Plugin	Exports a specific page to HTML.

Regular project news announcements help members stay in touch with events that can affect their work on the project.

What is a News Component?

A news component is a way to get developing information to project members via a project page.

You can use a news component to maintain a journal or blog about your project, to announce milestones met, or to share information about a rapidly changing situation.

There is a single set of news posts for your project. If you put a news component on more than one project page, the same new posts will show on all of them.

Post a News Item

To stay up to date, project can post regular news items on the project home page.

News items are posted and displayed on the respective project home page in which the news items are created and on the CollabNet TeamForge home page.

1. Go to the home page of the project in which you want to post the news item.
    - From within the project, click **Project Home** in the project navigation bar.
    - From anywhere in TeamForge, choose the project from the **Project** menu in the TeamForge navigation bar.
2. In the **Project News** section, click **CREATE NEWS POST**.
3. On the **Create News Post** page, provide a title for the news item.
4. Write the news item in the **BODY**. The news item can be up to 40000 characters long, including spaces.
5. Click **Create**.

The news item shows up on the project home page and TeamForge home page immediately.

Edit a News Item

To keep the project news in sync with the developments, modify the news items as and when required.

1. Find the project home page where you want to update the latest news.
    - From within the project, click **Project Home** in the project navigation bar.
    - From anywhere in TeamForge, choose the project from the **Projects** menu in the TeamForge navigation bar.
2. In the **Project News** section, click **Edit** next to the news items that you want to modify.

3. On the **Edit News Post** page, modify the title and/or the content of the news item.
4. Click **Save**.

The news post appears on the project home page, along with your name and the time at which you modified the post.

Delete a News Item

It’s a good idea to promptly delete a news item that is out of date or otherwise incorrect.

Deleting a news item from a project also deletes it from the TeamForge home page.

1. Find the project home page.
   - From within the project, click **Project Home** in the project navigation bar.
   - From anywhere in TeamForge navigation bar.
2. In the **Project News** section, click **Delete** next to the news item that you want to delete, and confirm that you want to delete the news item.

The news item is deleted.

You can publish the output of your project to selected audiences as packages and releases.

What is a Release?

A release is a group of one or more files that are published as a unit.

Each release can have a maturity level attribute to describe its state of completeness. Maturity levels are predefined and include development build, alpha, beta, and general availability.

The TeamForge file release system enables users to publish files and groups of files to selected audiences. Using role-based access control, administrators can control which project members can access each package or release.

Download a Release

Downloading a release brings all the release’s files to your local machine in a single .zip file.

1. Click **File Releases** from the **Project Home** menu.
2. On the **File Release Summary** page, click the title of the package containing the release you want to download.
3. On the **List Releases** page, click the title of the release.
4. On the **View Release** page, click **Download Selected**.

TeamForge prepares the compressed file in .zip format. You are prompted to open or save the file.

Create a Package

A package is a folder into which one or more related releases are published.

For example, you might create a package to represent a product deliverable or major component. You can then create releases within the package for product builds or other groups of files.

> **NOTE:** A package must exist before you create the releases and individual files that will go into the package.

1. Click **File Releases** from the **Project Home** menu.
2. On the **File Release Summary** page, click **Create**.
3. On the **Create Package** page, provide a title and description for the package.
4. Click **Save**.

The package is created. When you have created a package, you can publish a release into the package.

Create a Release

A release is a group of one or more files that are published as a unit.

IMPORTANT: Before creating a release, you must have a package into which you release will go.

1. Click **File Releases** from the **Project Home** menu.
2. On the **File Release Summary** page, click the title of the package in which you want to create the release.
3. On the **List Releases** page, click **Add**.
4. On the **Create Release** page, provide a name and description for the release.
5. Set the status of the release.
    - **Pending:** Releases with pending status are not visible in the drop-down list displayed when you set **Reported In Release** and **Fixed In Release** fields in an artifact. Use **Pending** status when you have created a release but have not yet finished adding files.
    - **Active:** Releases with active status are visible in the drop-down lists displayed when you set **Reported In Release** and **Fixed In Release** fields in an artifact.
6. Identify the maturity level of this release.
7. Click **Save**.

The release is created. You can begin adding files.

TIP: To facilitate tracking, you may want to match this release to the planning folder that tracks the work that's going into the release. If you do that, the relevant work items will automatically appear on the *Planned Tracker Artifact* tab. See Create a Planning Folder.

Edit a Release

You can edit a file release at any point in time, if required.

1. Click **File Releases** from the **Project Home** menu.
2. On the **File Release Summary** page, click the title of the package which has the release that you want to edit.
3. On the **List Releases** page, click the release that you want to edit.
4. Click **Edit** on the release details page.
5. On the **Edit Release** page, edit the fields as required.
6. Click **Save**.

Add Files to a Release

After you have created a release, you can add one or more files. All files in a release are published as a unit.

Project members can download the entire release in a single .zip file, or download only selected files.

1. Click **File Releases** from the **Project Home** menu.
2. On the **File Release Summary** page, click the title of the package containing the release.
3. On the **List Releases** page, click the title of the release.
4. On the **View Release** page, click **Add**.
5. On the **Add File** page, choose the desired file, then click **Save**. The file is added and you are returmed to the **View Release** page.
6. Repeat the last two steps until all files are added.

After you have added your files, you might wish to change the status of the release from pending to active. This will allow users with appropriate permission to select the release when entering or updating an artifact. You can also change the maturity level if needed.

Update Files in a Release

You can replace an existing file in a release with a new file.

1. Click **File Releases** from the **Project Home** menu.
2. On the **File Release Summary** page, click the title of the package containing the release.
3. On the **List Releases** page, in the **Releases** section, click the title of the release containing the file that you want to update.
4. On the **View Release** page, select the file you want to update, and click **Update**.
5. In the **Update a File** page, go to the new file with which you want to replace the current file.
6. Click **Save**.

This file is now updated.

Delete Files from a Release

If a file is no longer needed, it’s a good idea to delete it from the release.

1. Click **File Releases** from the **Project Home** menu.
2. On the **File Release Summary** page, click the title of the package containing the desired release.
3. On the **List Releases** page, in the **Releases** section, click the title of the release containing the file that you want to delete.
4. On the **View Release** page, choose the file you want to delete, and click **Delete**.

The file is deleted.

Update Release Attributes

As your release matures, you should update its maturity level and status.

1. Click **File Releases** from the **Project Home** menu.
2. On the **File Release Summary** page, click the title of the package containing the release.
3. On the **List Releases** page, in the **Releases** section, click the title of the release containing the file that you want to delete.
4. On the **View Release** page, click **Edit**.
5. On the **Edit Release** page, choose a new status for the release. A release can be in Active or Pending status.
6. Select the maturity level of this release from the **Maturity** field.
7. Click **Save**.

Change a Package’s Name or Description

When the purpose or the audience for a package shifts, you may want to rename the package or describe it differently.

1. Click **File Releases** from the **Project Home** menu.
2. On the **File Release Summary** page, select the package you want to edit, and click **Edit**.
3. On the **Edit Package** page, provide the new name or description.
4. To make it easier for users to download the package, select **Show Download Link in Project List**. This makes the download link show up in the project’s entry in the Project Categories system, if the project has been put in one or more categories. See Categorize a Project.
5. Click **Save**.

Associate a Release with other items

If a file release is related to other TeamForge items, such as documents, tasks, tracker artifacts, integrated application objects, or new items, you can connect the file release to the other item by creating an association.

Creating associations between items helps you define relationships, track dependencies, and enforce workflow rules. Some example uses of file release associations include:

- Associate a release with a requirements document.
- Associate a beta release with a task related to managing the beta program.

- Associate a release with tracker artifacts representing bugs fixed in the release.

> **NOTE:** You can also associate tracker artifacts such as bugs and feature requests with the related file releases. You do this as part of working with the tracker.

1. Click **File Releases** from the **Project Home** menu.
2. On the **File Release Summary** page, click the title of the package containing the release with which you want to create an association.
3. On the **List Releases** page, click the name of the release.
4. On the **View Release** page, click the *Associations* tab and click **Add**.
5. In the **Add Association Wizard** window, select the items with which you want to associate the artifact:
   - **Enter Item ID:** If you know the item's ID, you can enter it directly.

     > **NOTE:** To associate an object in an integrated application from within TeamForge, use the `[<prefix_objectid>]` format. Each integrated application displays its prefix on moving the mouse over the application name in the tool bar.

   - **Add from Recently Viewed:** Select one of the last ten items you looked at during this session.
   - **Add from Recently Edited:** Select one of the last ten items you changed.
6. Click **Next**.
7. You may add a comment in the **Association Comment** text box.
8. Save your work.
   - Click **Finish and Add Another** to add additional associations.
   - Click **FInish** to return to the **Details** page.

     > **NOTE:** When an association is added to or removed from TeamFOrge objects such as tracker artifacts, tasks, documents, discussions, and file releases, a notification mail is sent to users monitoring these objects. An option is provided at site level and user level to make sure whether the notification mail has to be sent or not. For more information on this, see Configure your Site's Settings.

9. Click the *Associations* tab to view a graphical representation of all the associated items. Through the Association Viewer, you can choose to view associations in the form of a list or flip over to the Trace view to explore the layers of associations (including parent/child dependencies) laid out in a timeline. You can scroll across the Trace view by dragging the mouse over the association layer or use the 'Previous' and 'Next' arrows to view all the objects as events in a timeline.

While the *Associations* tab shows the count of the total number of associations, you can only view the most recent 500 associations when you click the *Associations* tab in case the artifact has more than 500 associations. You can, however, browser through the **Association Viewer** to view older associations.

Delete a Package

If you no longer need a package, you can delete it.

Deleting a package deletes all of the releases and files within it.

> **IMPORTANT:** Delete a package only if you are sure that you no longer need any of the releases and files within it.

1. Click **File Releases** from the **Project Home** menu.
2. On the **File Release Summary** page, choose the package that you want to delete.
3. Click **Delete**.

The package and all of the releases and files withi it are now deleted.

Delete a Release

If you no longer need a release, you can delete it. Deleting a release deletes all of the files within it.

> **IMPORTANT:** Delete a release only if you are sure that you no longer need any of the files within it.

1. Click **File Releases** from the **Project Home** menu.
2. On the **File Release Summary** page, click the title of the containing the release that you want to delete.
3. On the **List Releases** page, in the **Releases** section, select the release that you want to delete, and click **Delete**.

The release and all of the files within it are now deleted.

When you monitor an item, such as a task or a document, you are notified by email when that item is updated. You can also monitor a discussion forum or project news by subscribing to its Really Simple Syndication (RSS) feed.

Monitor an Item

To get notified by email whenever an item changes, monitor the item. In TeamForge, an item is something you produce with the site's tools. These are some examples of items:

- Documents
- Tasks
- Tracker artifacts
- Discussion forum topics
- Files in a release

1. Go to the folder containing the item or items that you want to begin monitoring.
2. Select the item or items that you want to begin monitoring.
3. Click the **Monitor** down arrow, then choose **Monitor Selected**.

You are now monitoring all selected items. The monitoring icon 🔔 is displayed in the item list view and the monitored items appear in the monitored items list on your personal **Monitoring** page.

> **NOTE:** When you update two or more artifacts at a time, each user who is monitoring any of the changed artifacts gets a single email describing all the updates.

To stop monitoring an item, select it, then roll your mouse over the **Monitor** down arrow and choose **Stop Monitoring Selected**.

> **NOTE:** You can also stop monitoring any item from the monitored items list on your personal **Monitoring** page.

Monitor an Item for someone else

To alert another user to an item, add that user to the list of people monitoring that item.

After a user is added to a monitored item, the user can configure their own monitoring preferences for the item, or stop monitoring the item.

1. Go to the item to which you want to add a user.
2. Click the item to view the artifact.
3. On the item's **View Artifact** page, click **Users Monitoring**.

4. In the **Users Monitoring This Item** window, click **Add**.
5. From the list of available project members, select the user or users that you want to add to the monitored item.

> **TIP:** Press and hold the **Ctrl** key to select more than one user.

6. Click **Add**.

> **TIP:** You can also click **Add All** to select all users.

7. Click **OK**.

The users are now added to the monitored item.

Monitor Many Items

To get notified about a whole class of items whenever one is created or changed, monitor the foler that contains the items.

In TeamForge, a folder is a container for multiple items. Any container can be considered as a folder, even if it is not explicitly called a folder.

- A document folder contains documents.
- A task folder contains tasks.
- A tracker is a folder that contains tracker artifacts.
- A forum is a folder that contains discussion topics.
- A repository is a folder that contains code commits.
- A package is a folder that contains files.

1. Select the folder that you want to begin monitoring. For example, in a project where you are a member, click **SOURCE CODE** from your **Projects Home** menu and select one of the code repositories in the project.
2. Click the **Monitor** down arrow, then choose **Monitor Current Folder**.

You are now monitoring the folder. The monitoring icon 🔔 is displayed in the item list view and the monitored items appear in the monitored items list under the *MONITORING* menu available in the **My Workspace** page.

> **NOTE:** You do not receive monitoring notifications for changes that you yourself make to an item in a monitored folder.

To stop monitoring a folder, click the **Monitor** down arrow, then choose **Stop Monitoring Folder**.

> **NOTE:** You can also stop monitoring any item from the monitored items list under the *Monitoring* menu available in the **My Workspace** page.

Monitor an Application

To monitor the entire application, such as all trackers or all documents in a project, select it from the **Monitoring** tag available in the **My Workspace** page.

Monitoring an application keeps you updated on all items and folder within the application. Unlike individual items, list of items, and folders, you don't monitor entire applications from within a project.

1. Click **Monitoring** from the **My Page** menu.
2. From the **Edit Monitoring Subscriptions and Preferences** pane, on the **My Workspace** page, choose the project in which you want to monitor an application.
3. Click the *MONITORED APPLICATIONS* tab and select the applications that you want to begin monitoring.
4. Click **Save**.

You are now monitoring the selected applications.

Monitor Discussion Forums as RSS Feed

To get an update in your RSS feed reader each time there is an exchange of ideas in a discussion forum, subscribe to the forum's RSS feed.

In TeamForge, you can subscribe to discussion forums via RSS feeds.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, click the name of the forum that you want to subscribe through RSS feed.
3. On the **Topic Summary** page, click the RSS feed icon .

You can now monitor all topics in the selected discussio forum using your RSS feed reader.

> **NOTE:** You can access the project message via the RSS reader without logging in to TeamForge, so long as you have the discussion view permissions.

Monitor Project News as RSS Feed

To get an update in your RSS feed reader each time something important is announced in your project, subscribe to the project news as RSS feeds.

In TeamForge, you can subscribe to project news for all the projects via RSS feeds but not to the news of any specific project.

On the *NEWS* tab of **My Page**, click the RSS feed icon .

> **NOTE:** You must be a member of a project to view its news.

You can now monitor all project news announcements regarding your projects using your RSS feed reader.

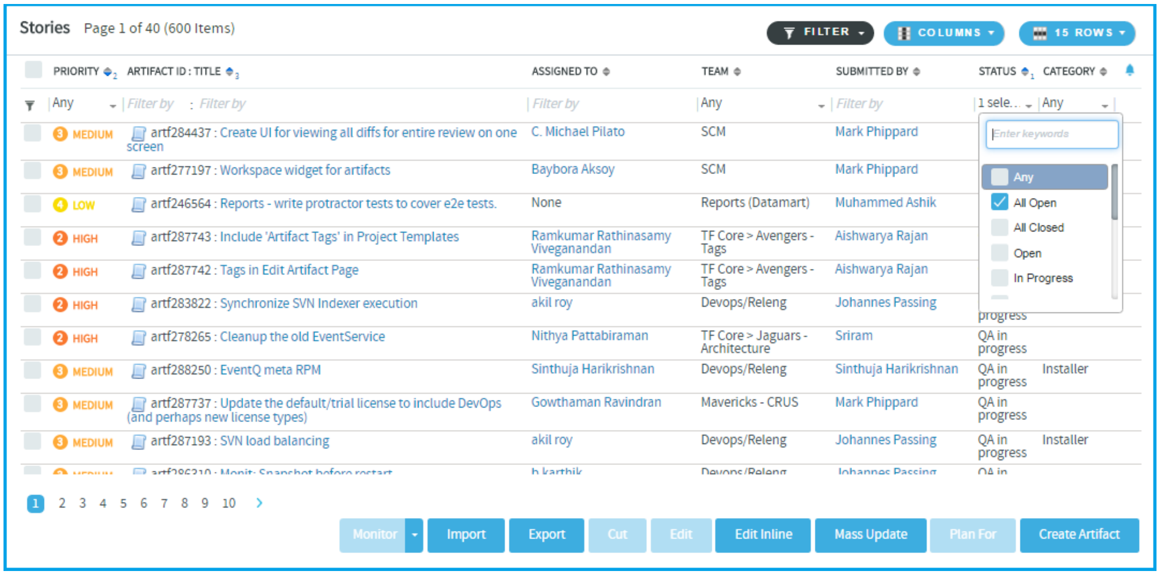
> **NOTE:** You can access the project news via the RSS reader without logging into TeamForge, so long as you have project membership in any project.

View what is being monitored?

All of your monitored items appear under the *MONITORING* tab available in the **My Workspace** page.

From this list, you can view or stop monitoring any item you are currently monitoring. You can also monitor entire applications from this page.

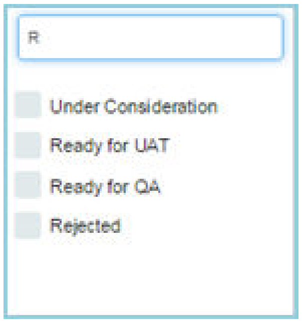
1. Select **MONITORING** from the **My Page** menu. Your personal monitoring page lists all items you are currently monitoring.
2. Specify the filter criteria in one or more filter fields (at the top of each column) and click **FILTER**.
    - You can find a filter field at the top of each column in most of the tables in the TeamForge application.
    - The filter field could be a text box or a drop-down list with multi-select checkboxes.



Stories Page 1 of 40 (600 Items)

PRIORITY	ARTIFACT ID : TITLE	ASSIGNED TO	TEAM	SUBMITTED BY	STATUS	CATEGORY
Any	Filter by : Filter by	Filter by	Any	Filter by	1 sele...	Any
3 MEDIUM	artf284437 : Create UI for viewing all diffs for entire review on one screen	C. Michael Pilato	SCM	Mark Phippard		
3 MEDIUM	artf277197 : Workspace widget for artifacts	Baybora Aksoy	SCM	Mark Phippard		
4 LOW	artf246564 : Reports - write protractor tests to cover e2e tests.	None	Reports (Datamart)	Muhammed Ashik		
2 HIGH	artf287743 : Include 'Artifact Tags' in Project Templates	Ramkumar Rathinasamy Viveganandan	TF Core > Avengers - Tags	Aishwarya Rajan		
2 HIGH	artf287742 : Tags in Edit Artifact Page	Ramkumar Rathinasamy Viveganandan	TF Core > Avengers - Tags	Aishwarya Rajan		
2 HIGH	artf283822 : Synchronize SVN Indexer execution	akil roy	Devops/Releng	Johannes Passing	progress	
2 HIGH	artf278265 : Cleanup the old EventService	Nithya Pattabiraman	TF Core > Jaguars - Architecture	Sriram	QA in progress	
3 MEDIUM	artf288250 : EventQ meta RPM	Sinthuja Harikrishnan	Devops/Releng	Sinthuja Harikrishnan	QA in progress	Installer
3 MEDIUM	artf287737 : Update the default/trial license to include DevOps (and perhaps new license types)	Gowthaman Ravindran	Mavericks - CRUS	Mark Phippard	QA in progress	
3 MEDIUM	artf287193 : SVN load balancing	akil roy	Devops/Releng	Johannes Passing	QA in progress	Installer
MEDIUM	artf286310 : Monit: Snapshot before restart	h karthik	Devops/Releng	Johannes Passing	QA in	

- You can type your filter criteria in the text boxes. The search text is case-insensitive.
- You can also select the filter values from one or more drop-down lists. By default, you can only select up to 10 filter values in a drop-down list. However, you can set a value that suits your requirement for the *FILTER_DROPDOWN_MAX_SELECTION* token in the `site-options.conf` file to increase or decrease the count.
- **Filter-as-you-type**: You can find the **Enter keywords** text box in all filter drop-down lists. As you type your filter keyword, instant search results are shown in the drop-down list. For example, in the following illustration, typing “R” instantly shows all statuses having the alphabet “R”. The search text is case-insensitive.



- Some search filters may not appear if your site administrator has not enabled them.

3. After filtering, if you wnat to clear the filters, click **FILTER** and select **Clear** from the drop-down list.
4. Use the up-down arrow at the top of any column to sort your list by that column.
    - Your primary sort column is identified by a superscript **1** next to the up-down arrow, and your secondary and third-level sort columns, if any, are likewise marked.
    - Click the up-down arrow again to reverse the sort order.

To stop monitoring an item from your personal monitoring page, select the item you want to stop monitoring, then click **Stop Monitoring**.

Check who monitors an item?

To see who is monitoring an item or folder, check the **Users Monitoring This Item** list.

1. Go to the page where the item appears.
2. Click the item to view the artifact.
3. On the item's **View Artifact** page, click **Users Monitoring**.

The **Users Monitoring This Item** window displays a list of all users who are monitoring the item.

Set Frequency for Monitoring Emails

You can set the frequency to check how often you receive monitoring email notifications for all the applications, folders and items you are monitoring.

1. Select **MY SETTINGS** from your **My Page** menu.
2. On your *User Details* section, in the *USER PREFERENCES* tab, choose an email notification preference.
   - **Email Per Change** - Get a separate email notifiaction for each change to a monitored item.
   - **Daily Digest Email** - Get one email notification each day containing a digest of all changes made to monitored items in the preceding twenty-four hours.
   - **Don't Send Email** - Get no email notifications for changes to monitored items. This can be handy when you are on vacation.
3. Click **Save**.

After you have set your global email frequency, you can further customize the frequency of application monitoring emails.

> **NOTE:** When you update two or more artifacts at a time, each user who is monitoring any of the changed artifacts gets a single email describing all the updates.

Set Frequency for Email Notifications on Monitored Applications

You can specify how often you want to receive email notifications about the applications you are monitoring.

To further personalize your monitoring preferences, you can set the frequency of email notifications for each monitored application on a project level. If you don't set your application monitoring email frequency settings, your global settings will get applied.

For example, suppose you are contributing code to the “Widgets” project, but your role in the “Gizmos” project is of a more advisory nature. When you monitor an item in the “Widgets” project, you’ll want more details updates than you will want from items you’ve monitored in the “Gizmos” project.

1. Set the global default email frequency from your **MY SETTINGS** page. See Set Frequency for Monitoring Emails.
2. Click **MONITORING** from the **My Page** menu.
3. From the **Edit Monitoring Subscriptions and Preferences** page, choose the project in which you want to configure monitoring email frequency.
4. On the *EMAIL NOTIFICATION PREFERENCES* tab, specify how often you want to be notified and click **Save**.

Any email notification preferences you set here will override the default preferences that you set on your **MY SETTINGS**page.

NOTE: When you update two or more artifacts at a time, each user who is monitoring any of the changed artifacts gets a single email describing all the updates.

Generate a report to get a snapshot of what is going on in a project. You can generate reports on data stored in both TeamForge's production (operational) database or datamart. Datamart, also known as the Reporting database, is build by extracting, transforming and loading (ETL) TeamForge's production data to a separate database (datamart) at regular intervals.
Important: Unless otherwise stated, you must have datamart enabled on your site to create reports in TeamForge. Note that a few Distribution Reports use data from TeamForge's operational database and a few Activity Reports use data from the EventQ's event data store.

You can specify the time at which the reporting data is refreshed from the production database. By default, the extraction takes place daily at 2:30 a.m. in the TeamForge application server's time zone. See Schedule Data Extraction for Reporting.

You can use reports to display data and group relevant information appropriately and specify intervals at which the datamart extracts TeamForge data from the datamart. For advanced reporting options and datamart information, see Advanced Reporting and Datamart Access.

You can also use external reporting tools to connect to the datamart and generate customized reports. See Datamart Access Using External Tools.

You can now refresh your reports to get the most recent data. You can also see the date and time when the report was last generated. Clicking the **Refresh** icon would fetch the latest available data from the respective data source (Operational DB, Datamart or EventQ).

Reporting Framework

Here's a list of some of the advantages of the new reporting framework:

- Reporting under one umbrella with a central dashboard of reports.
- Cross-project reporting capability.
- Tracker custom defined fields are in datamart and can be used to filter.
- All events are captured via EventQ event datastore.
- You can query and write custom reports.
- All data, including event, associations, and traceability data, can be queried through an elegant API for custom reporting and data extraction.
- High charts-based interactive data visualization charts. You can hover over the charts to see data points, click legends in the chart to toggle specific data point in and out of the chart and so on.
- You can drill one level down on some of the activity reports using column charts to get more clarity on the data points of your interest.
- Categorization of reports.
- Improved usability: Hassle free report creation with new widgets for selecting report criteria such as planning folders, trackers and repositories.

- Ability to save ‘Public’ and ‘Private’ reports: While ‘Public’ reports are visible to all project members with view reports permission, ‘Private’ reports are only for your consumption. You cannot publish ‘Private’ reports in project pages.

Reports–Role Based Access Control

In general, TeamForge site and project administrators, and users with tracker and task view permissions can see the **REPORTS** button on the project navigation bar.

- **Object-based access permissions** - If you are a TeamForge user, your ability to create, edit, preview or view reports depends on whether you have permission to access specific objects such as trackers and repositories in TeamForge. For example, to generate a report on the number of SCM commits in a repository, you must have permission to access the repository. Otherwise, a message such as “*You do not have sufficient permission to perform this operation*” is displayed.

- **Task and Tracker reports (Table reports)** - `Task and Tracker View` permission is required to generate task and tracker reports. You must have `View Activity` permission or object-level permission to view Activity reports such as SCM Commits, Build Activity, Build and Test Activity, Artifact Created and Artifact Closed reports.

- **Context-sensitive access permission** - Reports are shown or hidden from users based on the context. For example, a user must have `View Project Page` permission to view reports published on a project home page.

- **Deleting reports** - Site and project administrators can delete reports. In addition, you can delete your own reports.

Reports Available in TeamForge

Reports in TeamForge are grouped under the following categories.

- Activity Reports
- Agile Reports
- Distribution Reports
- Trend Reports
- Table Reports: Task and Tracker Reports

Activity Reports

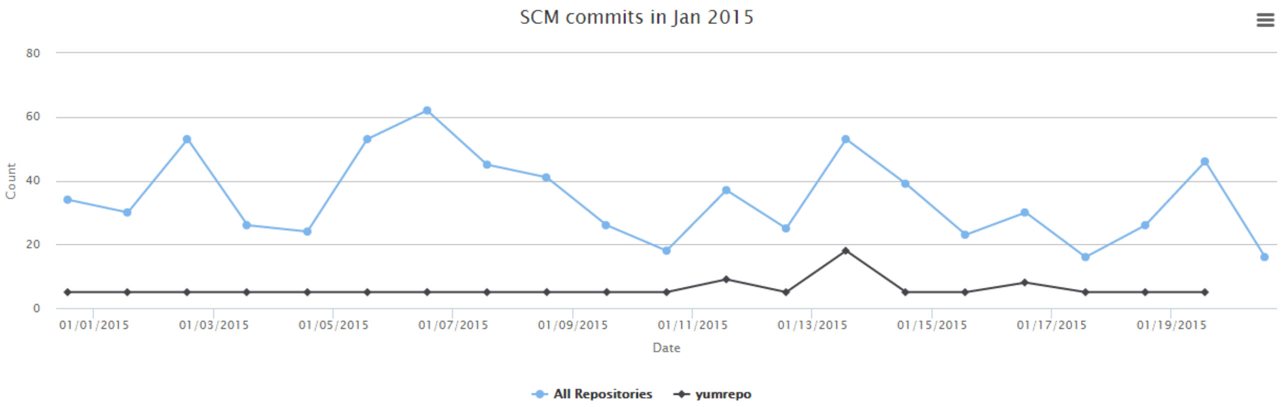
Here’s a list of TeamForge activity reports such as the SCM commits, artifact created and closed reports.

These are reports to track activities such as SCM commits, artifact creation and closure, build and test activities and so on. Some of these reports are powered by data from EventQ’s event data store.

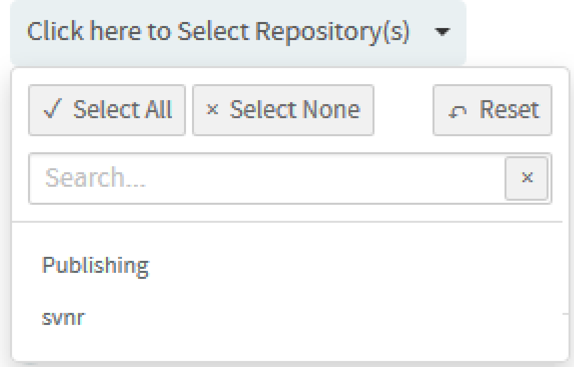
> **NOTE:** You can also drill one level down on some of the activity reports using column charts to get more clarity on the data points of your interest.

SCM Commits

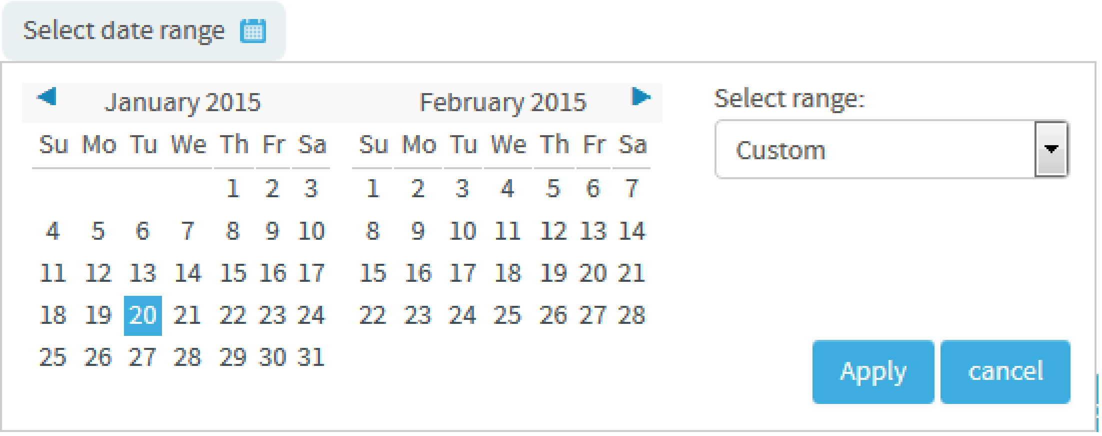
Shows the number of commits in one or more selected repositories in a given time period.



1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.
3. Select **Scm Commits** from **Activity Reports**.
4. Type a report title and description.
5. Select one or more repositories from the **SELECT REPOSITORY(S)** drop-down list.



6. Select date range and click **Apply**. To set the date range, select the start date first and then the end date. Use the month/year navigation arrows to select the month/year you want. If you don't have the exact dates at hand, you can also select one of the time periods from the **Select Range: Custom** drop-down list.



7. Select a chart type such as 'Line' or 'Bar' or 'Column' from the **CHART DISPLAY TYPE** drop-down list.

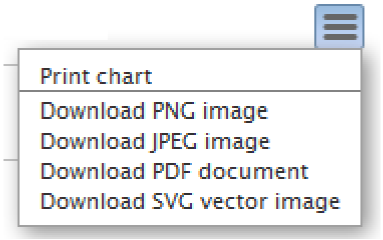
8. Select report visibility: `Public` or `Private`.

9. Click **Preview**.

10. Click **Create**. The report is created and the **View Report** page appears.

Print or download charts

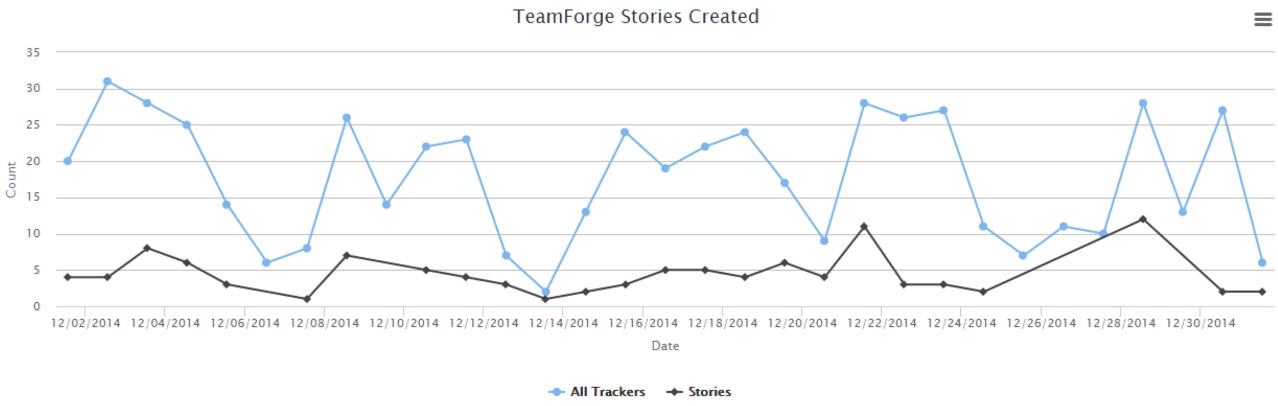
You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



11. Click **Back to Reports List** to go back to the Reports dashboard.

Artifact Created

Shows the numer of artifacts created in one or more selected trackers in a given time period.



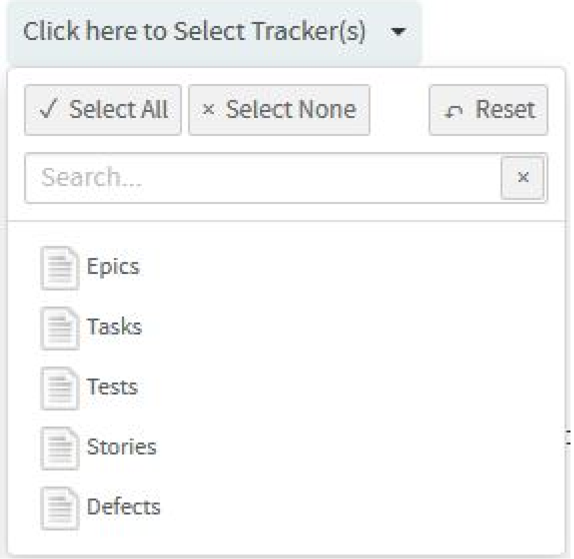
1. Click **REPORTS** from the **Project Home** menu.

2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.

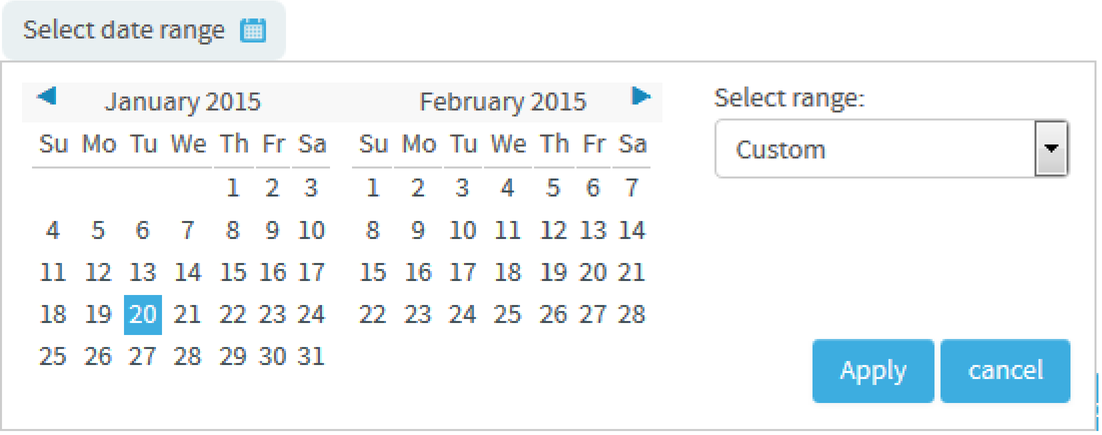
3. Select **Artifact Created** from **Activity Reports**.

4. Type a report title and description.

5. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.



6. Select date range and click **Apply**. To set the date range, select the start date first and then the end date. Use the month/year navigation arrows to select the month/year you want. If you don’t have the exact dates at hand, you can also select one of the time periods from the **Select Range: Custom** drop-down list.



7. Select a chart type such as ‘Line’ or ‘Bar’ or ‘Column’ from the **CHART DISPLAY TYPE** drop-down list.

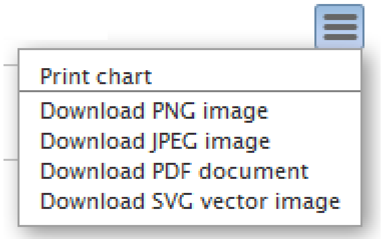
8. Select report visibility: `Public` or `Private`.

9. Click **Preview**.

10. Click **Create**. The report is created and the **View Report** page appears.

Print or download charts

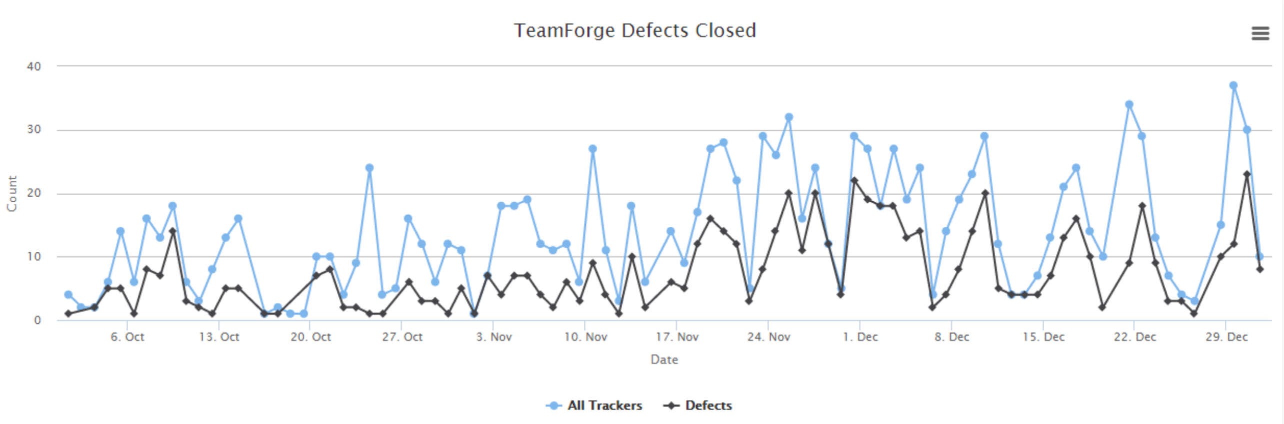
You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



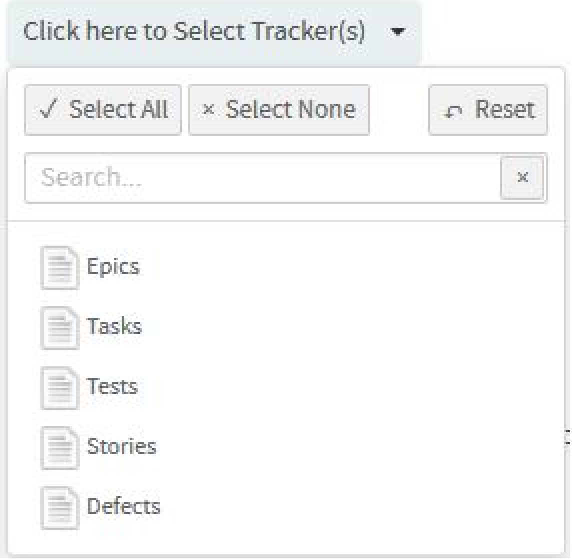
11. Click **Back to Reports List** to go back to the Reports dashboard.

Artifact Closed

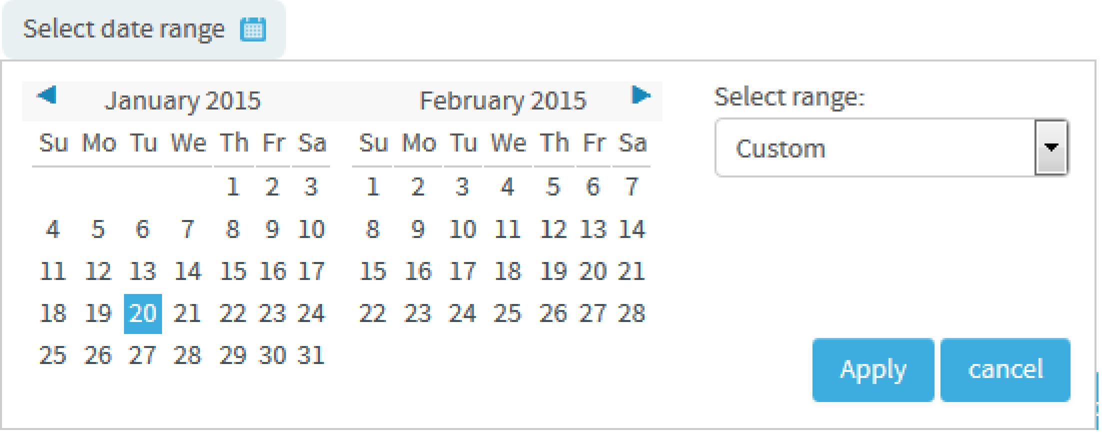
Shows the number of artifacts closed in one or more selected trackers in a given time period.



1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.
3. Select **Artifact Closed** from **Activity Reports**.
4. Type a report title and description.
5. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.



6. Select date range and click **Apply**. To set the date range, select the start date first and then the end date. Use the month/year navigation arrows to select the month/year you want. If you don't have the exact dates at hand, you can also select one of the time periods from the **Select Range: Custom** drop-down list.



Su	Mo	Tu	We	Th	Fr	Sa
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

Su	Mo	Tu	We	Th	Fr	Sa
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28

7. Select a chart type such as 'Line' or 'Bar' or 'Column' from the **CHART DISPLAY TYPE** drop-down list.

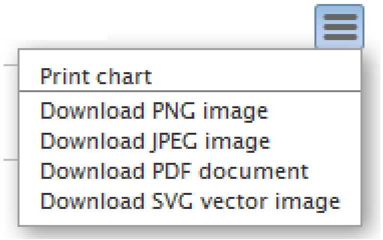
8. Select report visibility: `Public` or `Private`.

9. Click **Preview**.

10. Click **Create**. The report is created and the **View Report** page appears.

Print or download charts

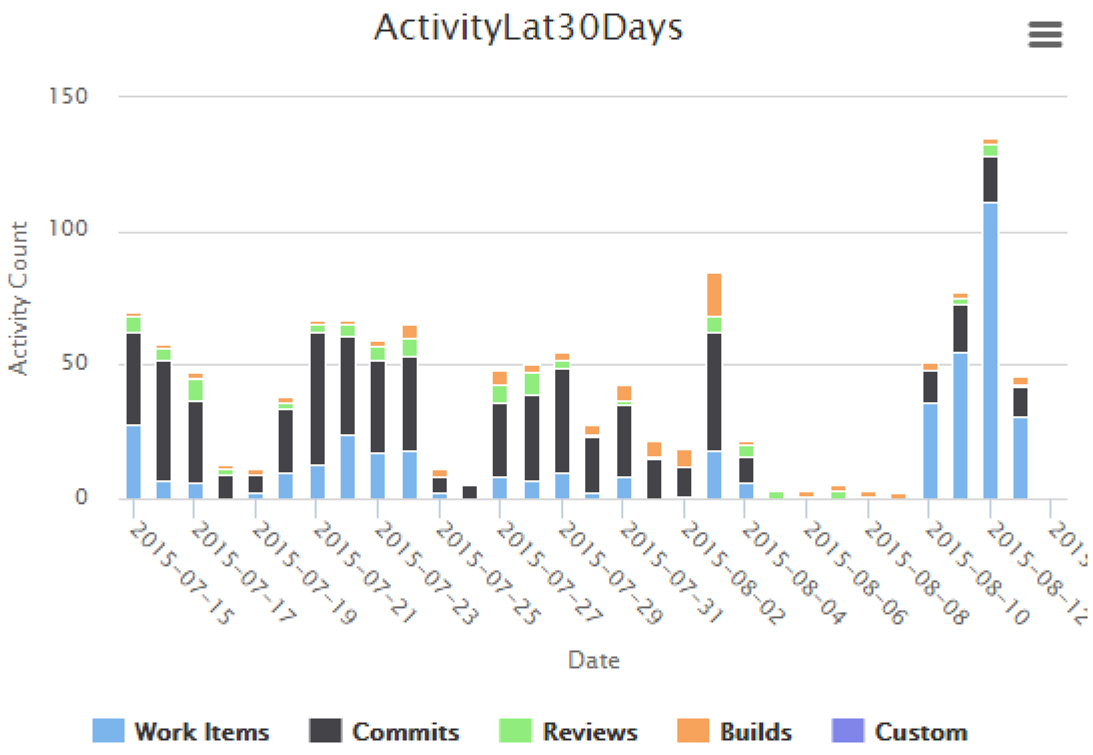
You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



11. Click **Back to Reports List** to go back to the Reports dashboard.

Activity Over Time

Shows the count of various project activities over a period of time such as commits, reviews, builds, and so on. This report is powered by data from the EventQ's event data source.



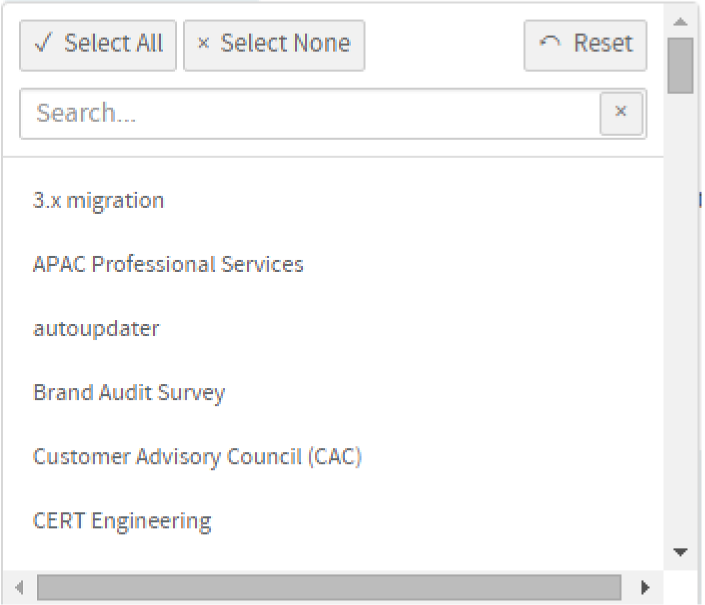
1. Click **REPORTS** from the **Project Home** menu.

2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.

3. Select **Activity Over Time** from **Activity Reports**.

4. Type a report title and description.

5. Select one or more projects from the **PROJECT(S)** drop-down list.



6. Select the number of days from the **LAST N DAYS** drop-down list.

7. The default **Display Type** is *EventsDual*.

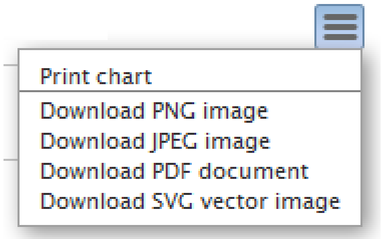
8. Select report visibility: `Public` or `Private`.

9. Click **Preview**.

10. Click **Create**. The report is created and the View Report page appears.

Print or download charts

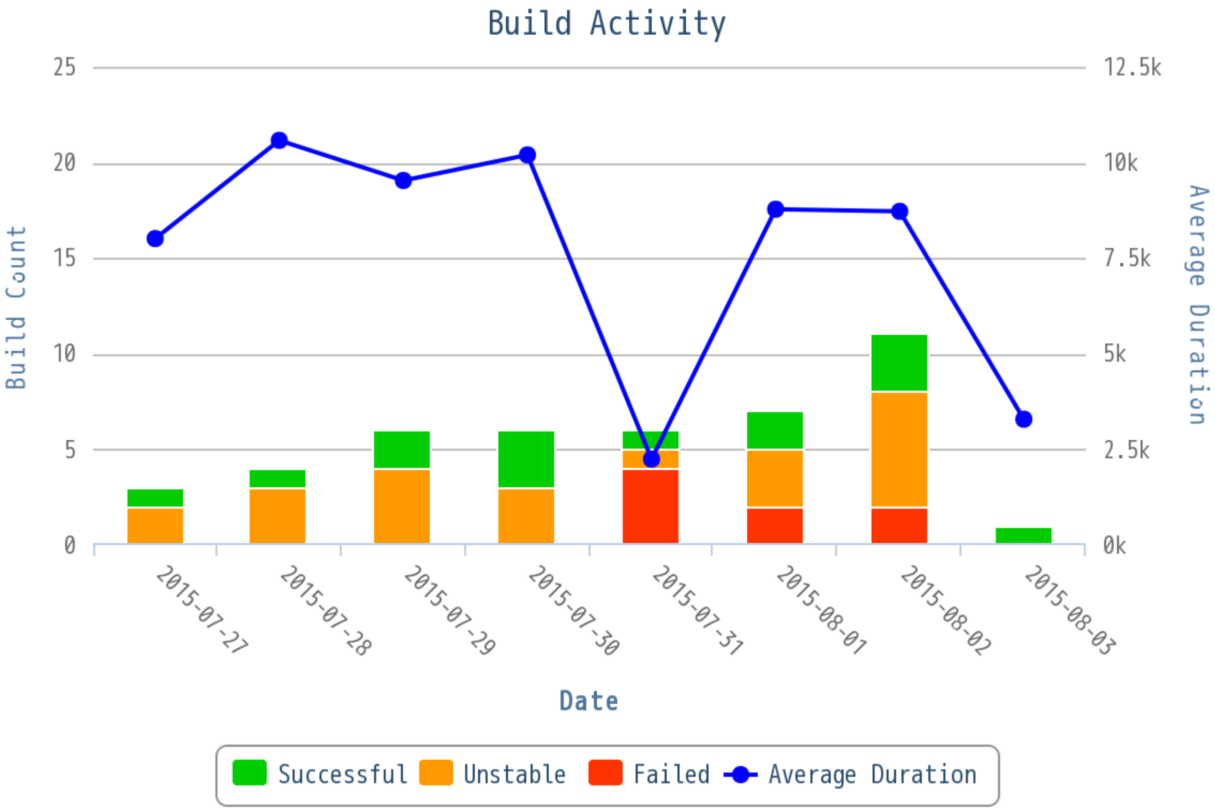
You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



11. Click **Back to Reports List** to go back to the Reports dashboard.

Build Activity

Shows the build activity over a period of time including the count of builds and the average build duration. This report is powered by data from the EventQ's event data store.



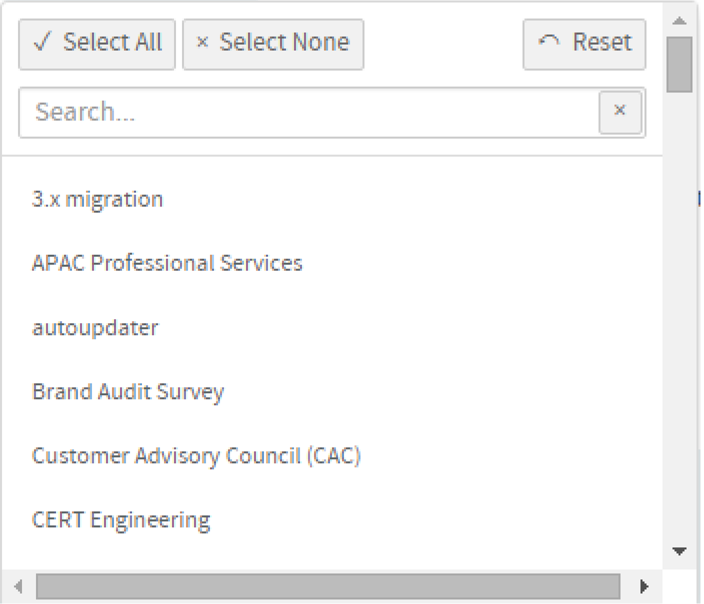
1. Click **REPORTS** from the **Project Home** menu.

2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.

3. Select **Build Activity** from **Activity Reports**.

4. Type a report title and description.

5. Select one or more projects from the **PROJECT(S)** drop-down list.



6. Select the number of days from the **LAST N DAYS** drop-down list.

7. The default **Display Type** is *EventsDual*.

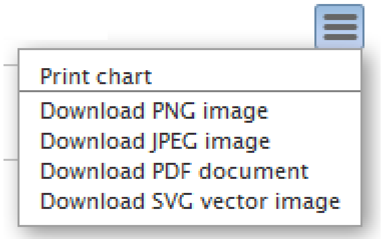
8. Select report visibility: `Public` or `Private`.

9. Click **Preview**.

10. Click **Create**. The report is created and the **View Report** page appears.

Print or download charts

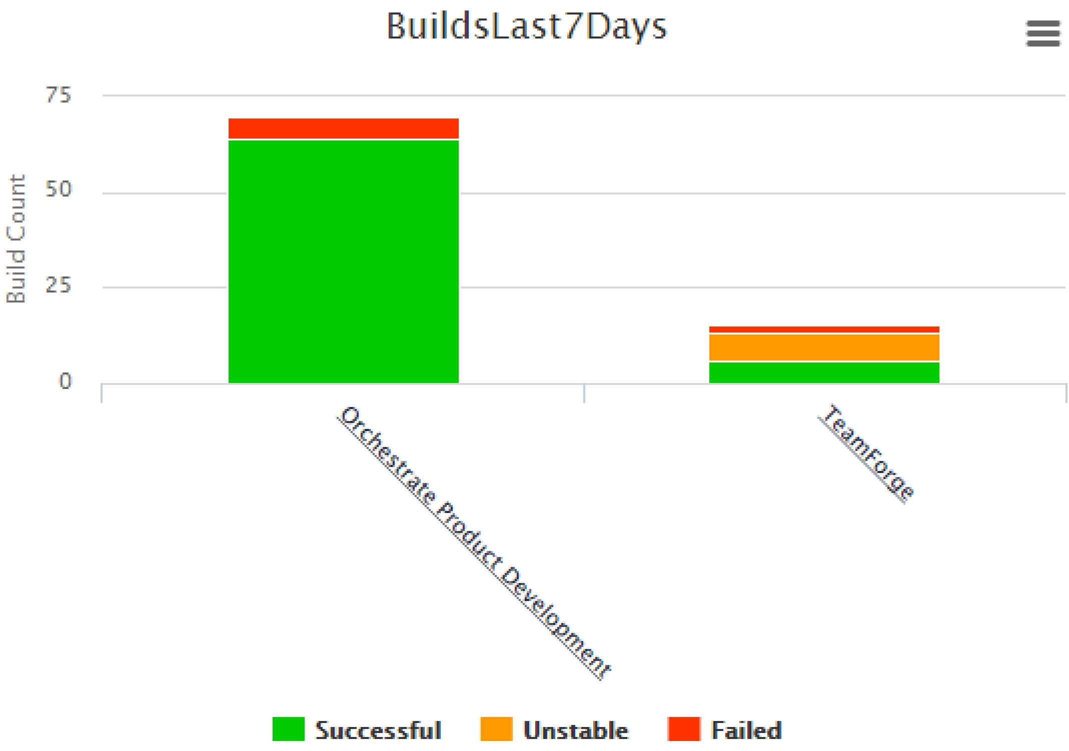
You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



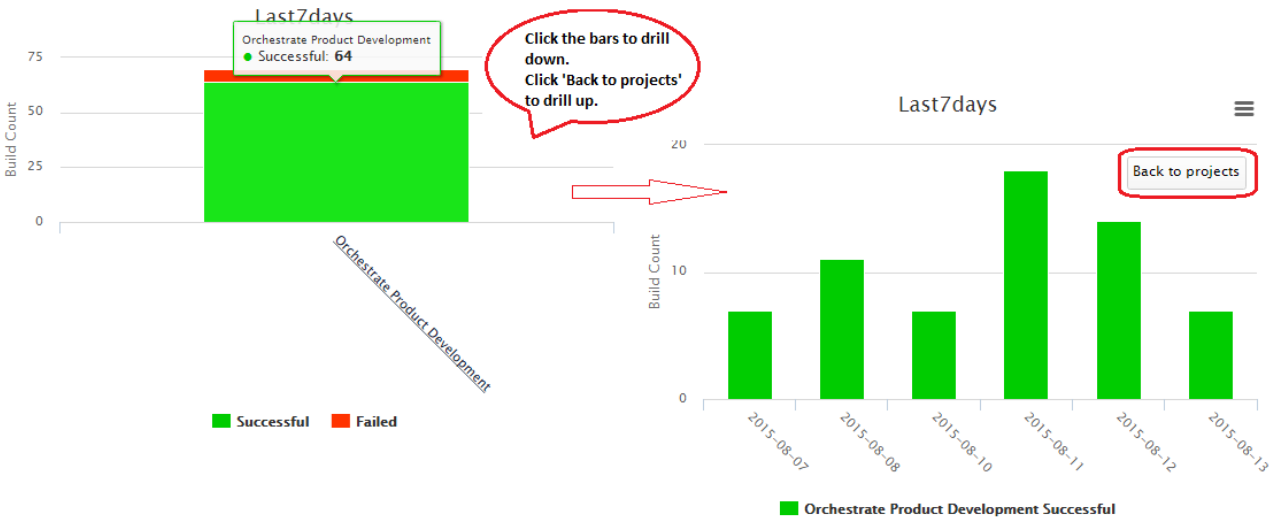
11. Click **Back to Reports List** to go back to the Reports dashboard.

Build Activity by Project

Shows the number of builds over a period of time. This report is powered by data from the EventQ's event data store.



You can drill down this report.



Click the bars to drill down.
Click 'Back to projects' to drill up.

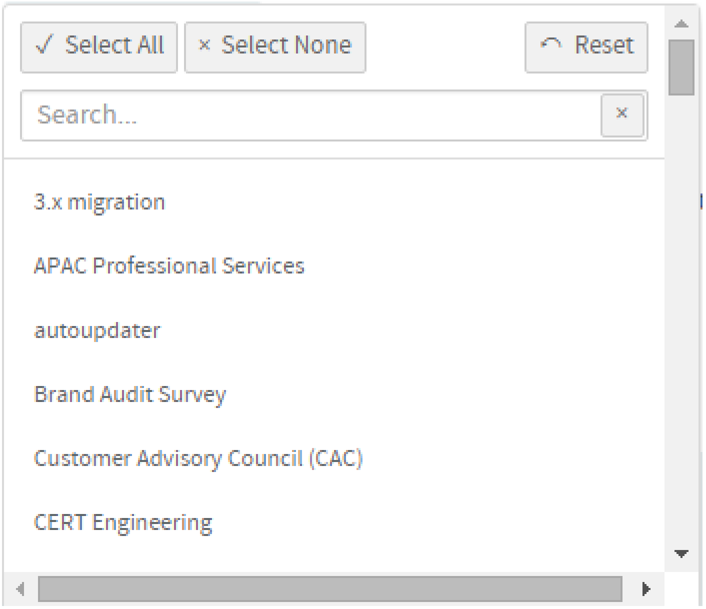
1. Click **REPORTS** from the **Project Home** menu.

2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.

3. Select **Build Activity by Project** from **Activity Reports**.

4. Type a report title and description.

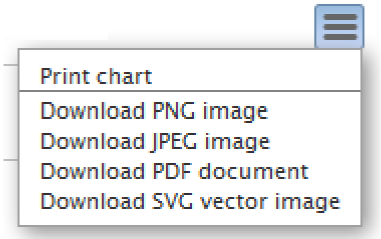
5. Select one or more projects from the **PROJECT(S)** drop-down list.



6. Select the number of days from the **LAST N DAYS** drop-down list.
7. The default **Display Type** is *EventsDrilldown*.
8. Select report visibility: `Public` or `Private`.
9. Click **Preview**.
10. Click **Create**. The report is created and the **View Report** page appears.

    Print or download charts

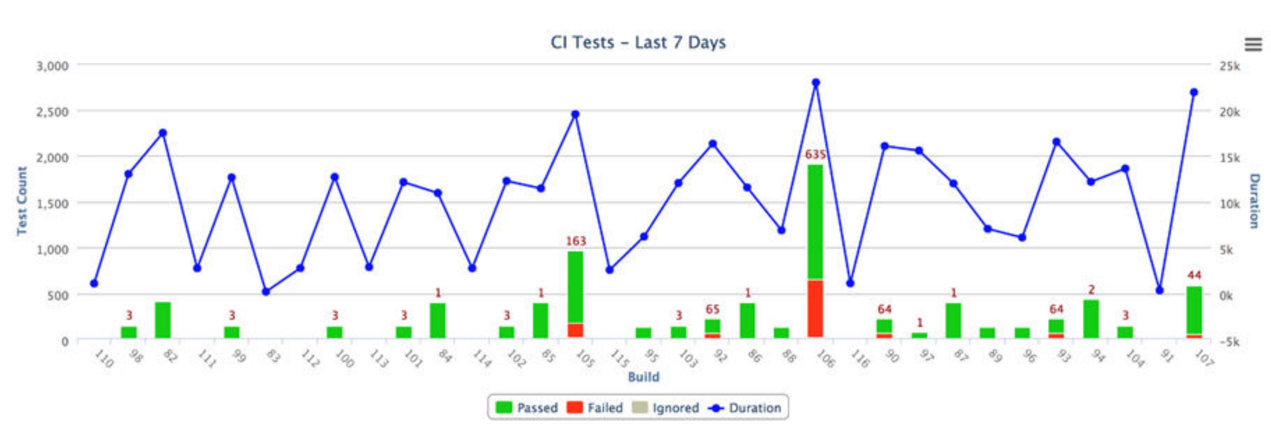
    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



11. Click **Back to Reports** List to go back to the Reports dashboard.

Build and Test Activity

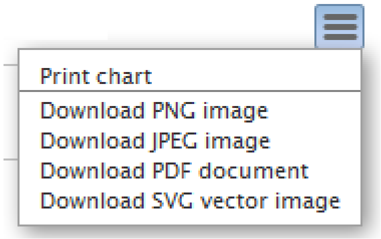
Shows the total number of CI tests—including passed, failed, and ignored—performed for each build over a period of time.Also shows the total test duration for each build. This report is powered by data from the EventQ's event data store.



1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.
3. Select **Build and Test Activity** from **Activity Reports**.
4. Type a report title and description.
5. Select one of the projects from the **PROJECT(S)** drop-down list.
6. Select the number of days from the **LAST N DAYS** drop-down list.
7. The default **Display Type** is *EventsDual*.
8. Select report visibility: `Public` or `Private`.
9. Click **Preview**.
10. Click **Create**. The report is created and the **View Report** page appears.

    Print or download charts

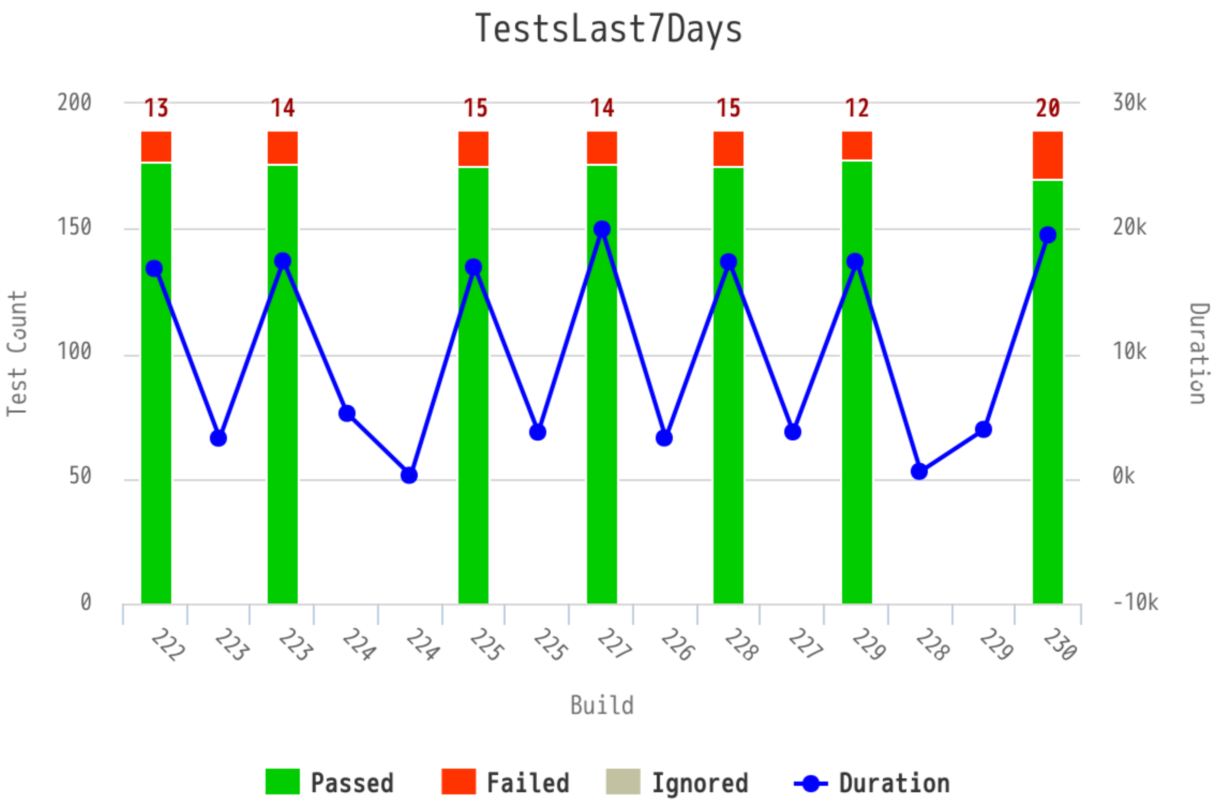
    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.

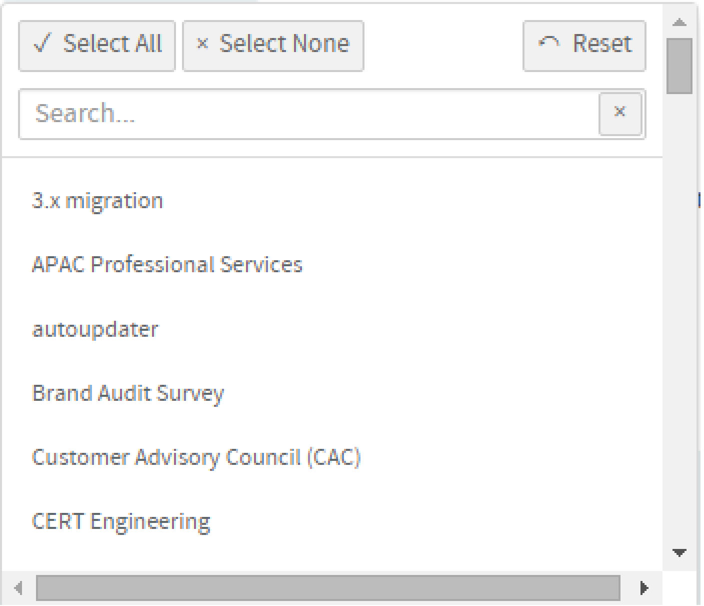
11. Click **Back to Reports List** to go back to the Reports dashboard.

Test Activity

Shows the count of tests over a period of time including the average build duration. This report is powered by data from the EventQ's event data store.



1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.
3. Select **Test Activity** from **Activity Reports**.
4. Type a report title and description.
5. Select one or more projects from the **PROJECT(S)** drop-down list.



6. Select the number of days from the **LAST N DAYS** drop-down list.

7. The default **Display Type** is *EventsDual*.

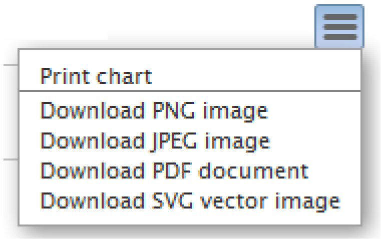
8. Select report visibility: `Public` or `Private`.

9. Click **Preview**.

10. Click **Create**. The report is created and the **View Report** page appears.

    Print or download charts

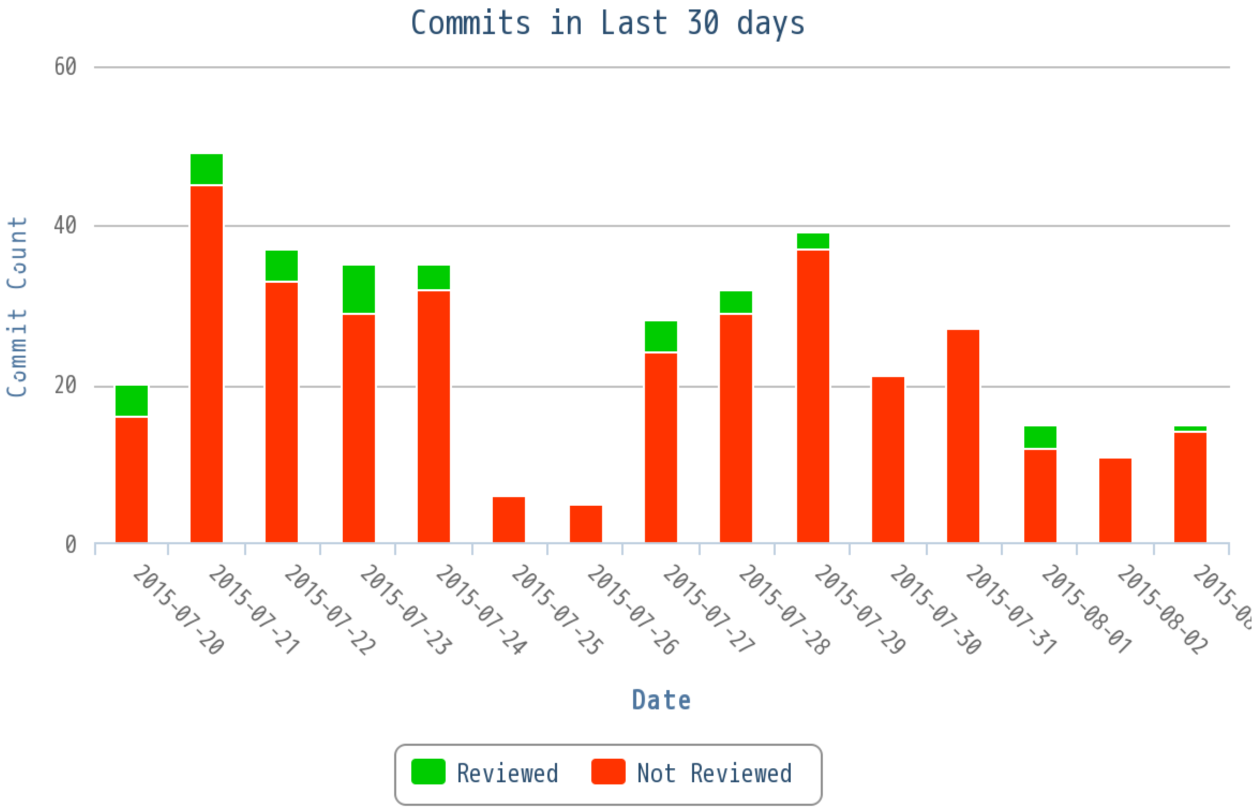
    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



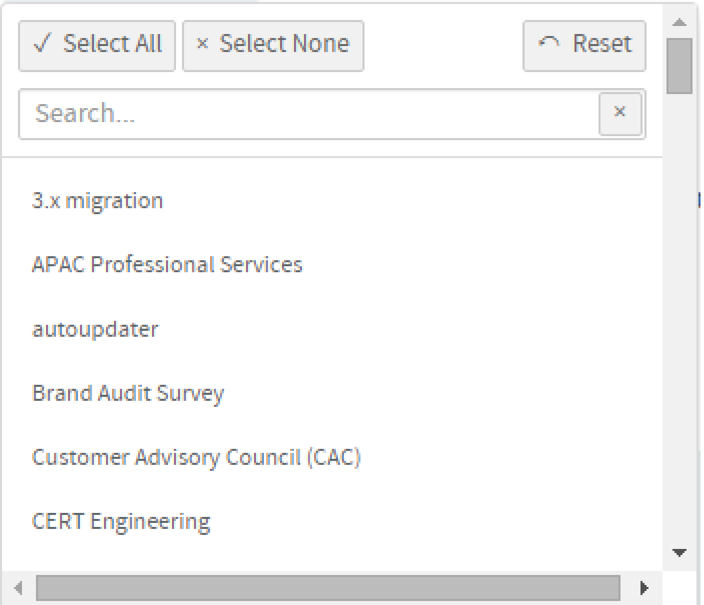
11. Click **Back to Reports List** to go back to the Reports dashboard.

Commit Activity

Shows the count of daily commits over a period of time. You can group this report by the review status, if required. This report is powered by data from the EventQ's event data store.



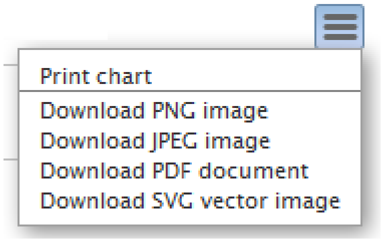
1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.
3. Select **Commit Activity** from **Activity Reports**.
4. Type a report title and description.
5. Select one or more projects from the **PROJECT(S)** drop-down list.



6. Select the number of days from the **LAST N DAYS** drop-down list.
7. By default, the **GROUP BY: Review Status** check box is selected. Clear this check box if you do not want to group the report by review status.
8. The default **Display Type** is *EventsDual*.
9. Select report visibility: `Public` or `Private`.
10. Click **Preview**.
11. Click **Create**. The report is created and the **View Report** page appears.

Print or download charts

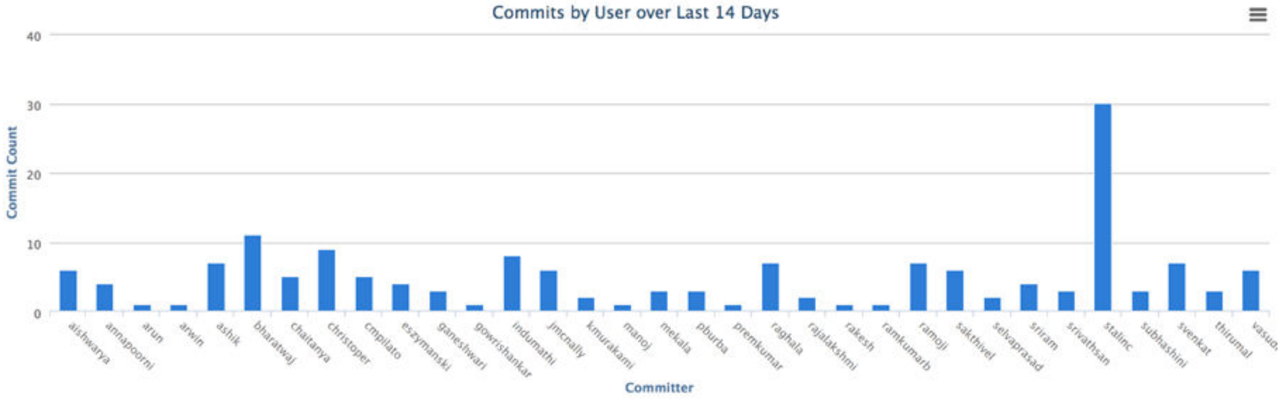
You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



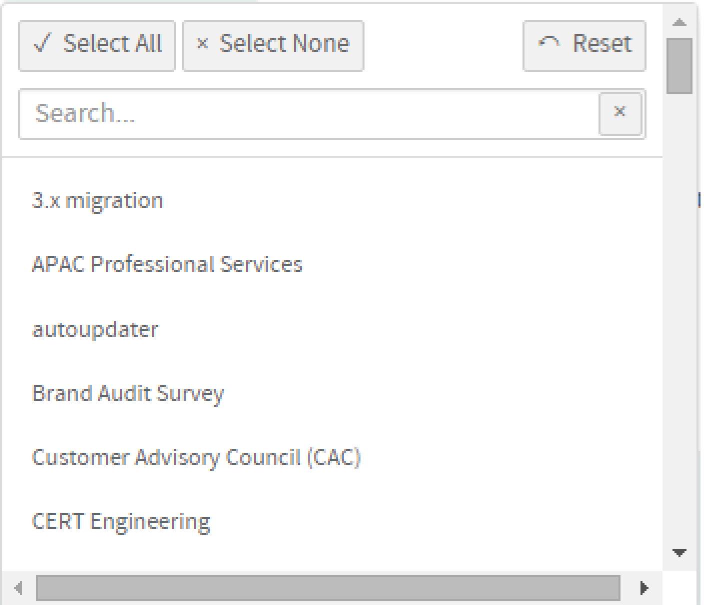
12. Click **Back to Reports List** to go back to the Reports dashboard.

Commits by User

Shows the number of commits by each user over a period of time. This report is powered by data from the EventQ's event data store.



1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.
3. Select **Commits by User** from **Activity Reports**.
4. Type a report title and description.
5. Select one or more projects from the **PROJECT(S)** drop-down list.



3.x migration

APAC Professional Services

autoupdater

Brand Audit Survey

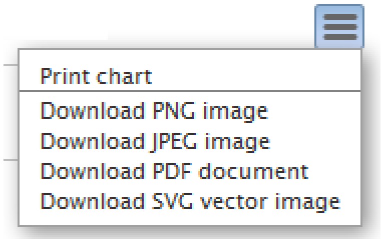
Customer Advisory Council (CAC)

CERT Engineering

6. Select the number of days from the **LAST N DAYS** drop-down list.
7. By default, the **GROUP BY: Review Status** check box is selected. Clear this check box if you do not want to group the report by review status.
8. **EXCLUDE USERS (COMMA-SEPARATED LIST)** - Optionally, you can type a list of comma-separated user names that you want to exclude from the report.
9. The default **Display Type** is *EventsDual*.
10. Select report visibility: `Public` or `Private`.
11. Click **Preview**.
12. Click **Create**. The report is created and the **View Report** page appears.

    Print or download charts

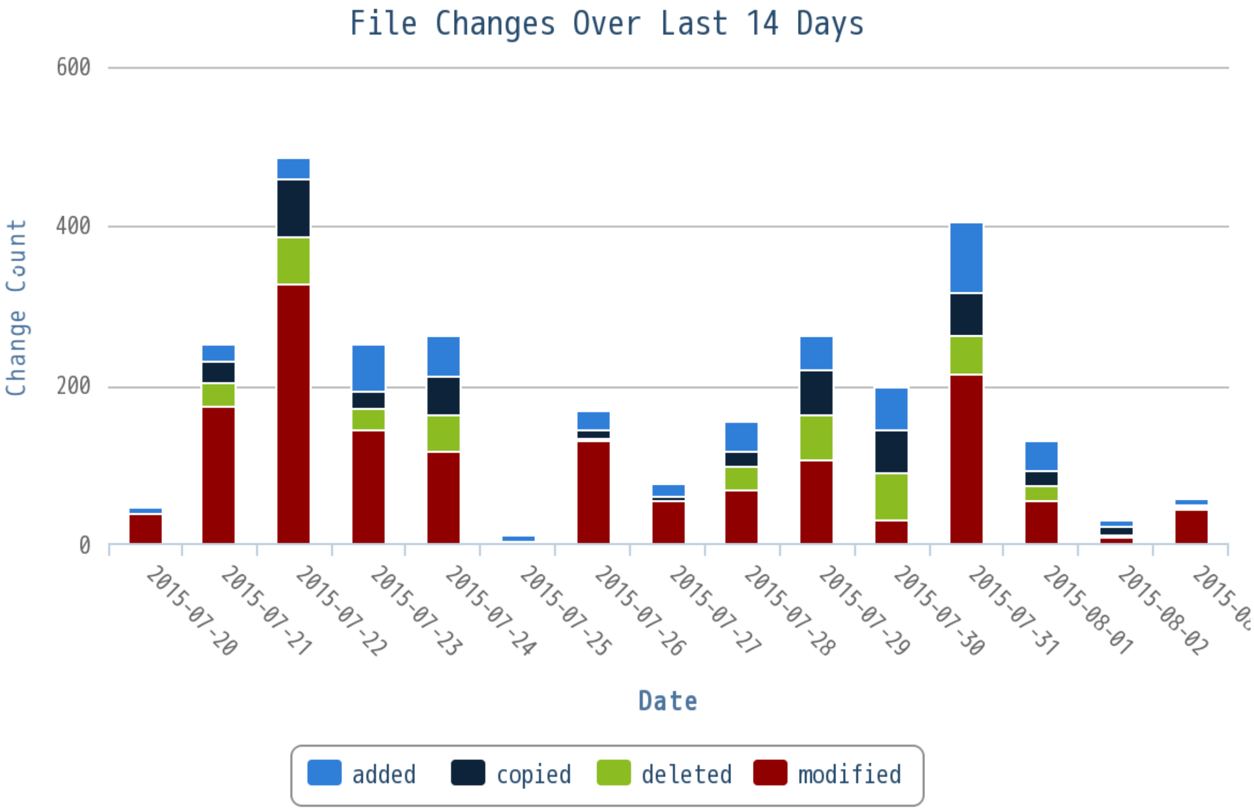
    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



13. Click **Back to Reports List** to go back to the Reports dashboard.

File Changes over Time

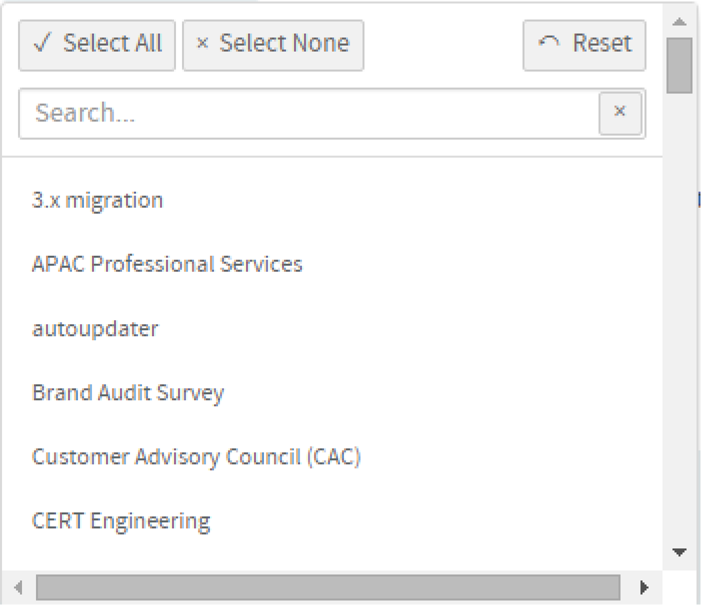
Shows the file change statistics over time, including the total count of changes and the number of add, delete, copy and modify operations. This report is powered by data from the EventQ's event data store.



1. Click **REPORTS** from the **Project Home** menu.

2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.

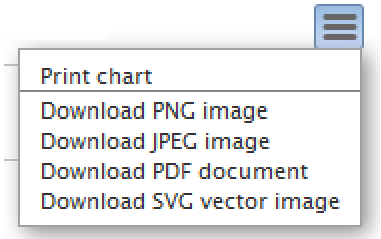
3. Select **File Changes Over Time** from **Activity Reports**.
4. Type a report title and description.
5. Select one or more projects from the **PROJECT(S)** drop-down list.



6. Select the number of days from the **LAST N DAYS** drop-down list.
7. The default **Display Type** is *EventsDual*.
8. Select report visibility: `Public` or `Private`.
9. Click **Preview**.
10. Click **Create**. The report is created and the **View Report** page appears.

Print or download charts

You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



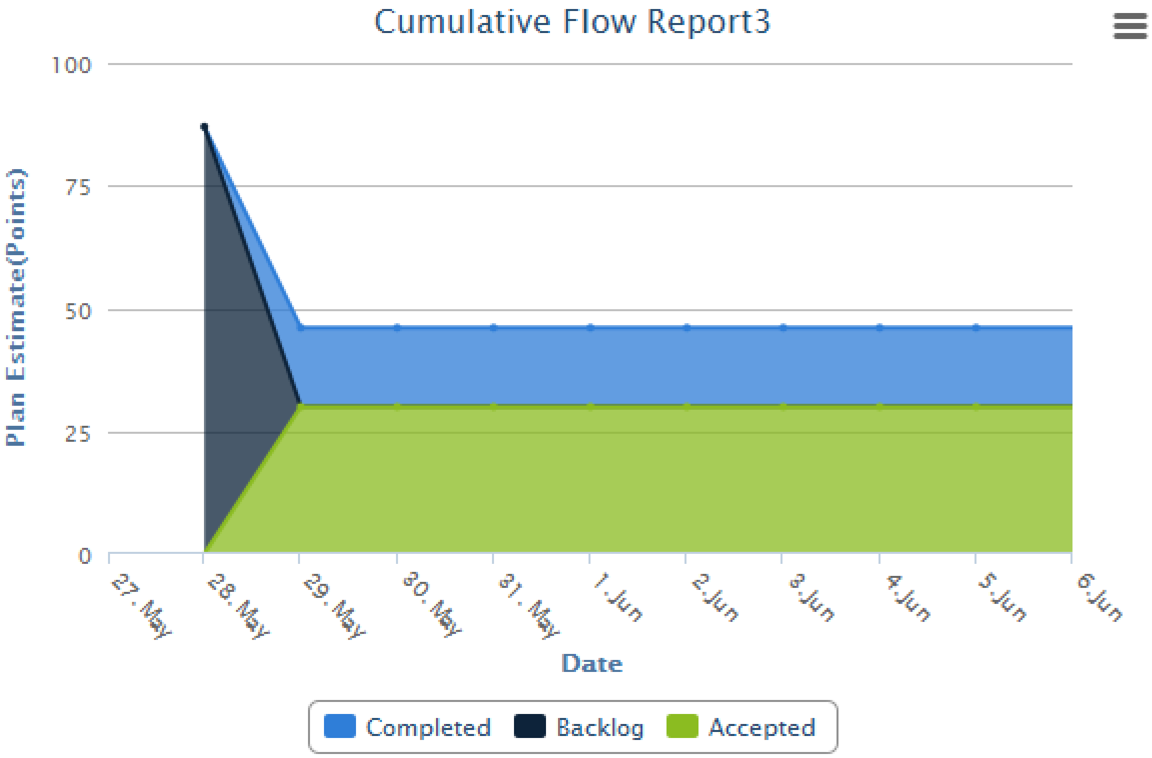
11. Click **Back to Reports List** to go back to the Reports dashboard.

Agile Reports

Here's a list of TeamForge agile reports such as the release burn up and burn down reports.

Cumulative Flow Chart

The cumulative flow chart shows the progress of backlog items by status for a sprint or a release.



- Generate this report to see the rolled up status chart of scheduled work items in a release or a sprint.

- By viewing the rolled up status of backlog items by date, you can forecast whether you are on track or not, adjust the scope if required and identify bottlenecks in your release or sprint.

- You have ‘Date’ in the X axis and ‘Plan Estimate’ in terms of number of points in the Y axis.

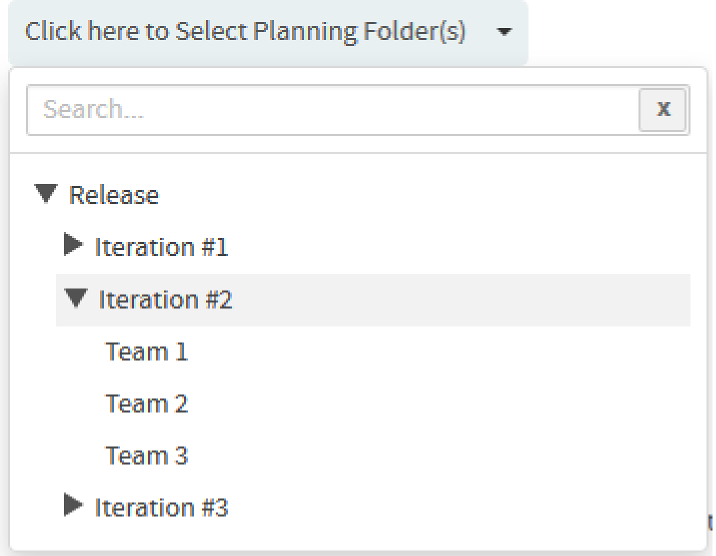
1. Click **REPORTS** from the **Project Home** menu.

2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.

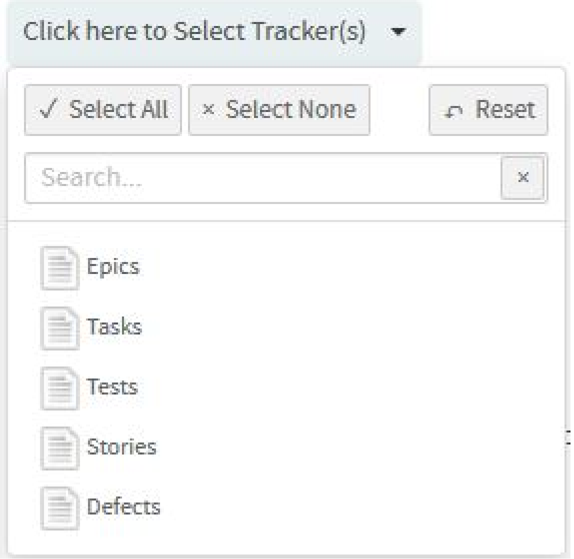
3. Select **Cumulative Flow Chart** from **Agile Reports**.

4. Type a report title and description.

5. Select a planning folder from the **SELECT PLANNING FOLDER(S)** drop-down list.



6. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.



7. Leave the **CHART DISPLAY TYPE** as *Area*, which is the only available chart type for this report.

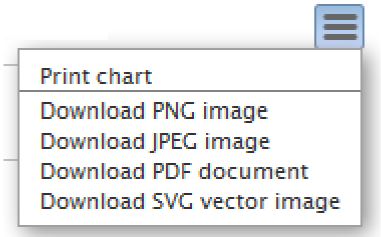
8. Select report visibility: `Public` or `Private`.

9. Click **Preview**.

10. Click **Create**. The report is created and the **View Report** page appears.

    Print or download charts

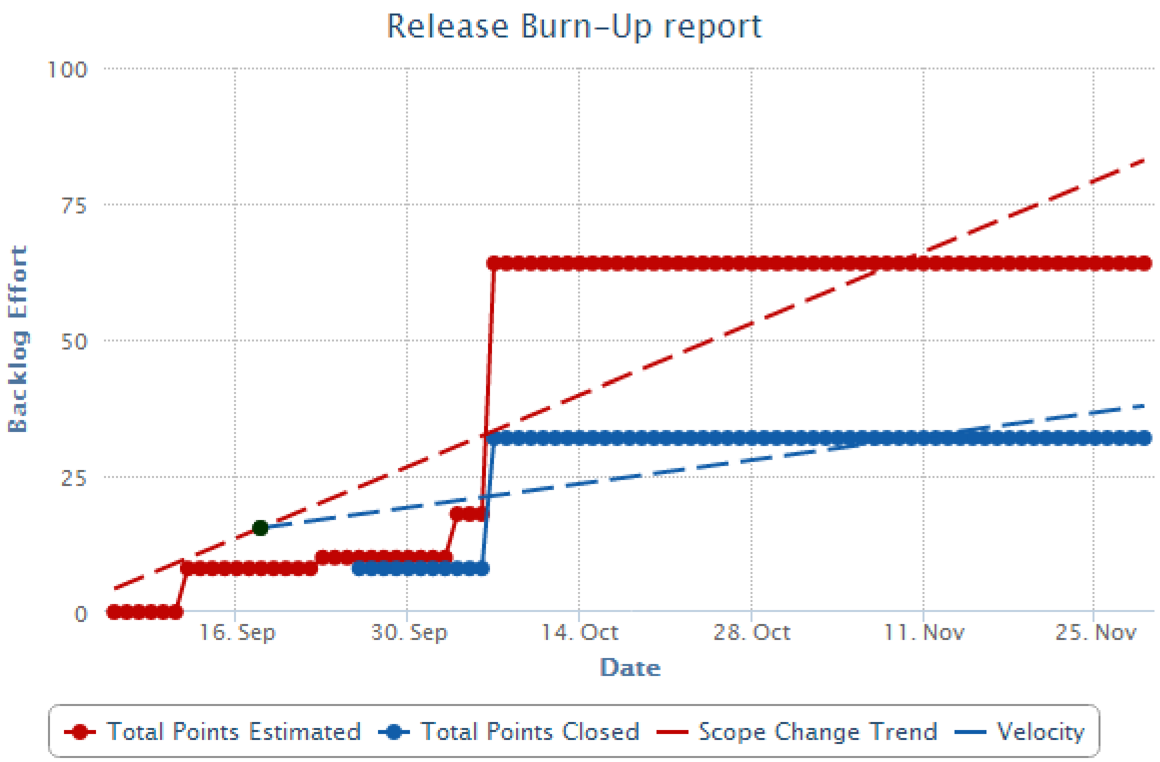
    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



11. Click **Back to Reports List** to go back to the Reports dashboard.

Release Burn Up Chart

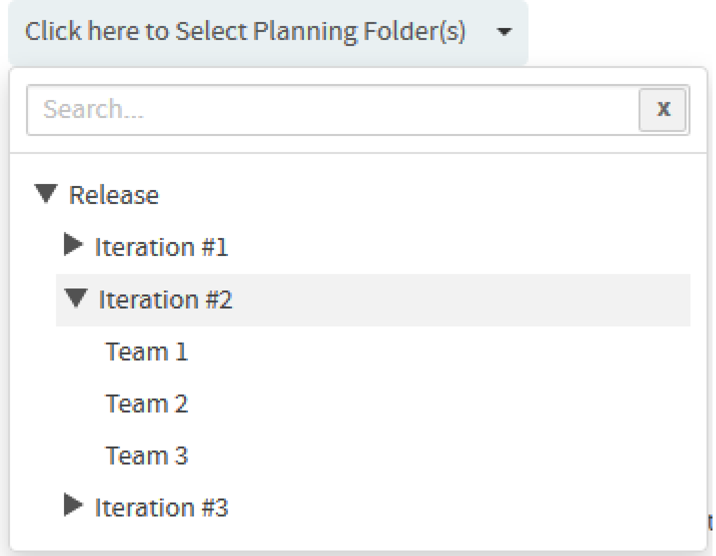
The Release Burn Up Chart shows the work progress to date against the total planned work. This chart shows the total planned work, total work completed to date and the rate of progress (velocity).



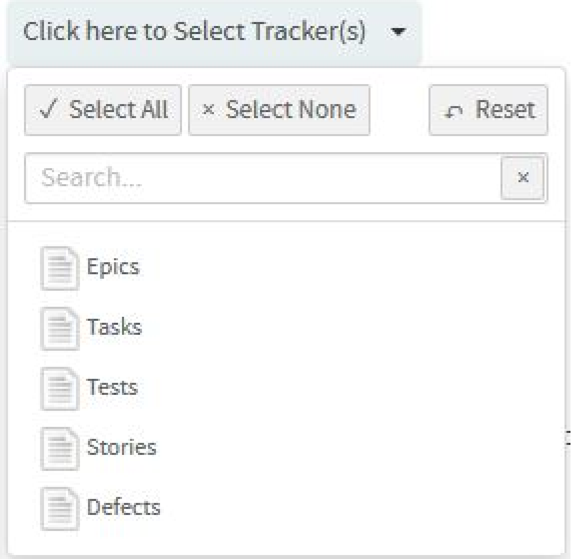
IMPORTANT: You must have the planning folder's start and end dates defined to generate this chart.

This chart includes a trend line to show the planned work scope change between a release's start and end dates (in other words, the start and end dates of the release's planning folder). Optionally, you can also include a trend line in the chart that forecasts when planned work might be completed depending on the average rate of progress (velocity).

1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.
3. Select **Release Burn Up Chart** from **Agile Reports**.
4. Type a report title and description.
5. Select a planning folder from the **SELECT PLANNING FOLDER(S)** drop-down list.



6. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.



7. Optionally, select the **Include Forecast** check box.

8. Select either `Points` or `Hours`.

9. Leave the **CHART DISPLAY TYPE** as `Trendlines`, which is the only available chart type for this report.

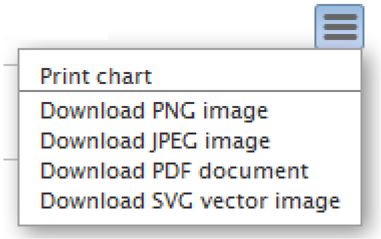
10. Select report visibility: `Public` or `Private`.

11. Click **Preview**.

12. Click **Create**. The report is created and the **View Report** page appears.

    Print or download charts

    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



Print chart
Download PNG image
Download JPEG image
Download PDF document
Download SVG vector image

13. Click **Back to Reports List** to go back to the Reports dashboard.

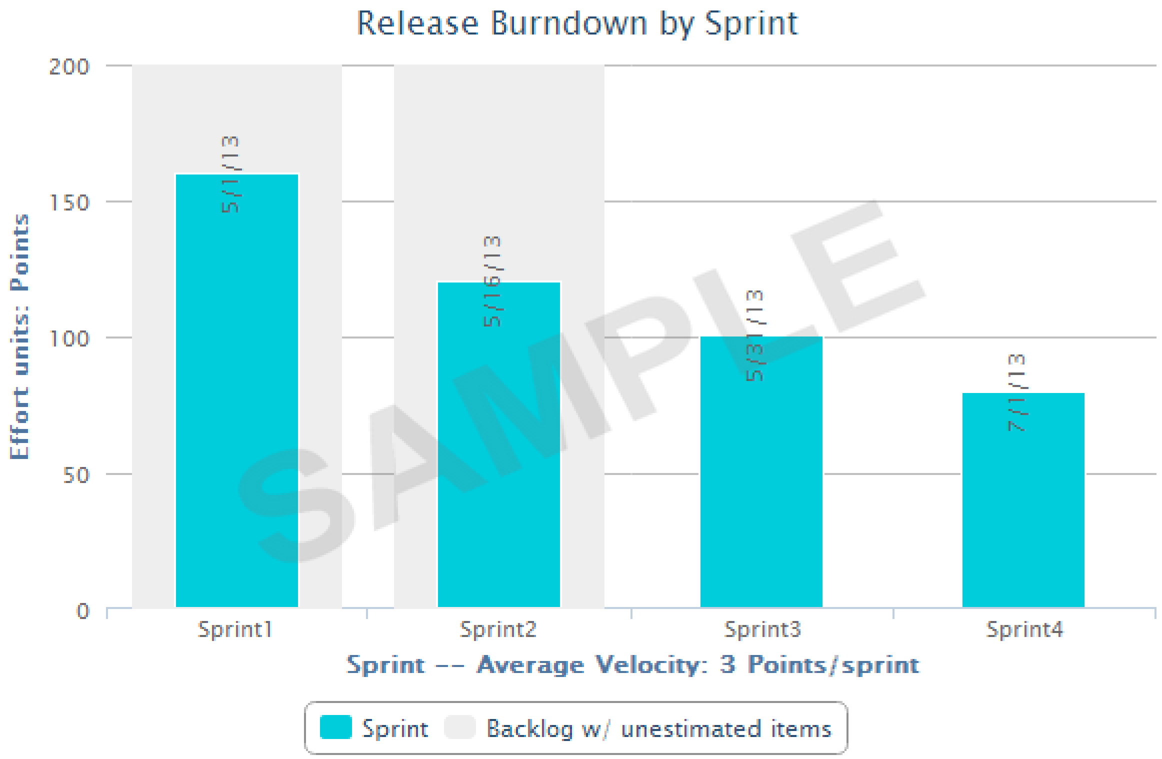
Burn Down Chart

The Burn Down Chart shows your project's progress on a daily basis.

As work gets completed sprint-by-sprint for a release, your backlog tends to decrease. The burn down chart tells the story of how much work is left to be done versus how much time is left. With time along the X axis, the amount of work (backlog) measured in story points is on the Y axis. You can also see the average velocity per sprint in the release burn down chart.In addition to creating burn down charts by release, you can configure this report for a selected sprint planning folder to see its progress on a day-to-day basis.

- The story told in your burn down chart is only as reliable as the underlying data.
- You must have the planning folder's start and end dates defined to generate this chart.
- You can have the **POINTS** field enabled or disabled for a tracker. If a release planning folder consists of both "points-enabled" and "points-disabled" trackers, the release burn down chart shows grey-colored bars for all sprints, meaning, there are work items without points estimation data. This is because, the "points-disabled" tracker work items are considered as unestimated backlog items as they don't have points data. As a workaround, while configuring the release burn down chart, you can include "points-enabled" trackers alone to have the release burn down chart behave as expected.

Things to consider if you are generating burn down chart for a release planning folder



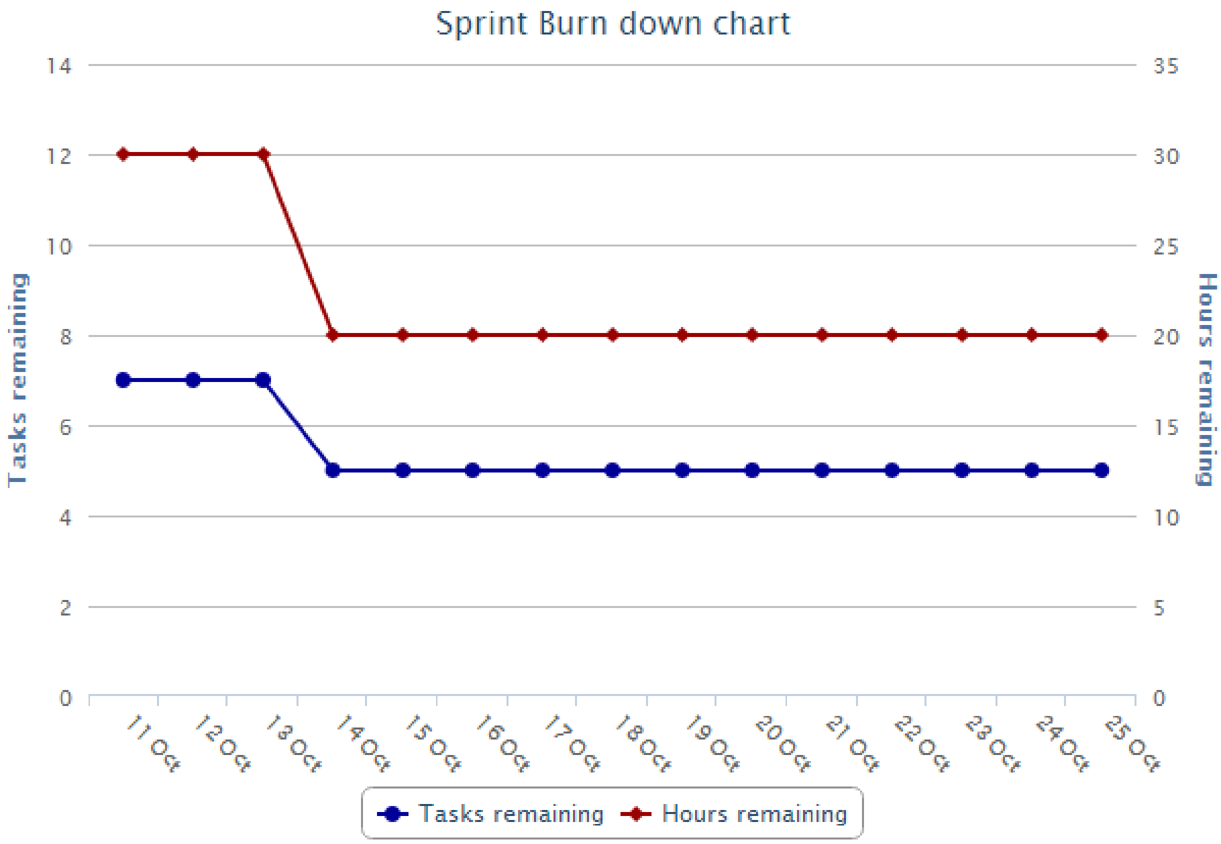
- It is recommended to have all your sprint planning folders organized hierarchically within your release planning folder. Release burn down chart can be generated only if that planning folder has child (sprint) planning folders.

- Within a selected release planning folder, all immediate child planning folders are considered as sprint planning folders.

- The bar chart shows the work left to be done (estimated as points) on the first day of every sprint.

- The release burn down calculation is based on (story) points and no effort data is considered. Meaning, the release burn down will not behave as expected if you have effort data alone and no points data.

Things to consider if you are generating burn down chart for a sprint planning folder



- The sprint burn down calculation is based on remaining effort data and no (story) points data is considered. Meaning, the sprint burn down will not behave as expected if you have points data alone and no remaining effort data.

- You can exclude weekends or specific days from your report if you are creating a burn down chart for a sprint planning folder.

- The sprint burn down chart shows the number of tasks and hours remaining for a selected sprint planning folder.

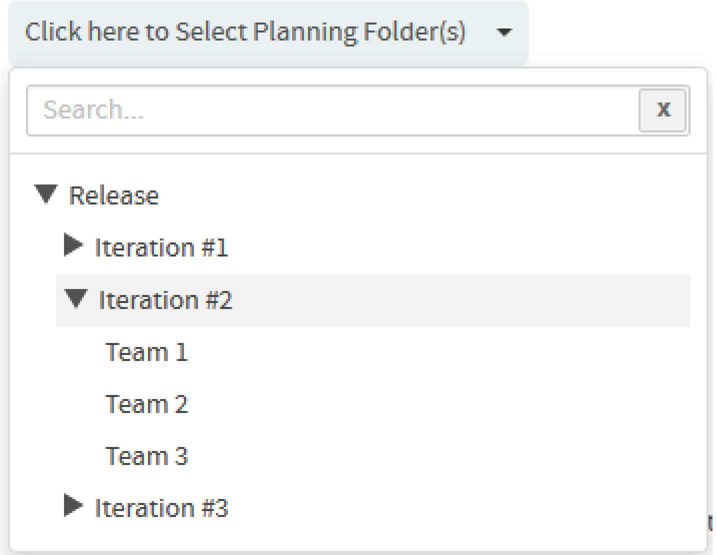
1. Click **REPORTS** from the **Project Home** menu.

2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.

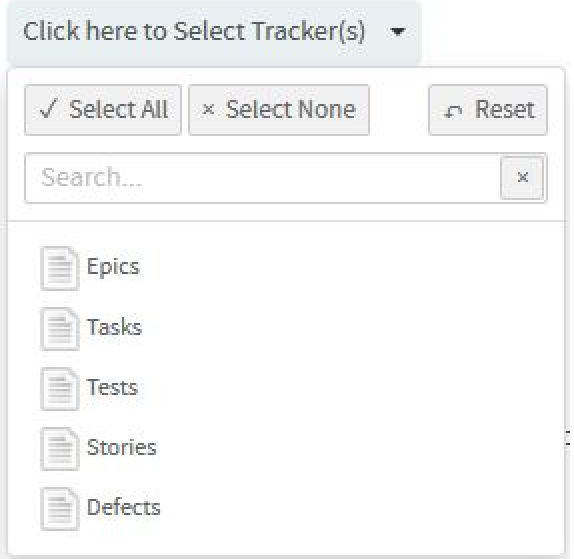
3. Select **Burn Down Chart** from **Agile Reports**.

4. Type a report title and description.

5. Select a planning folder from the **SELECT PLANNING FOLDER(S)** drop-down list.



6. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.



7. Select one of the options, **Release** or **Sprint**, to create burn down chart for either a release planning folder or a sprint planning folder.

> **TIP:** Select the Exclude Weekends check box to exclude weekends from the report.

8. Leave the **CHART DISPLAY TYPE** as `Burndown`, which is the only available chart type for this report.

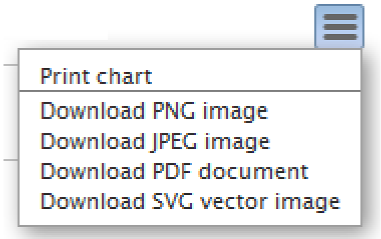
9. Select report visibility: `Public` or `Private`.

10. Click **Preview**.

11. Click **Create**. The report is created and the **View Report** page appears.

    Print or download charts

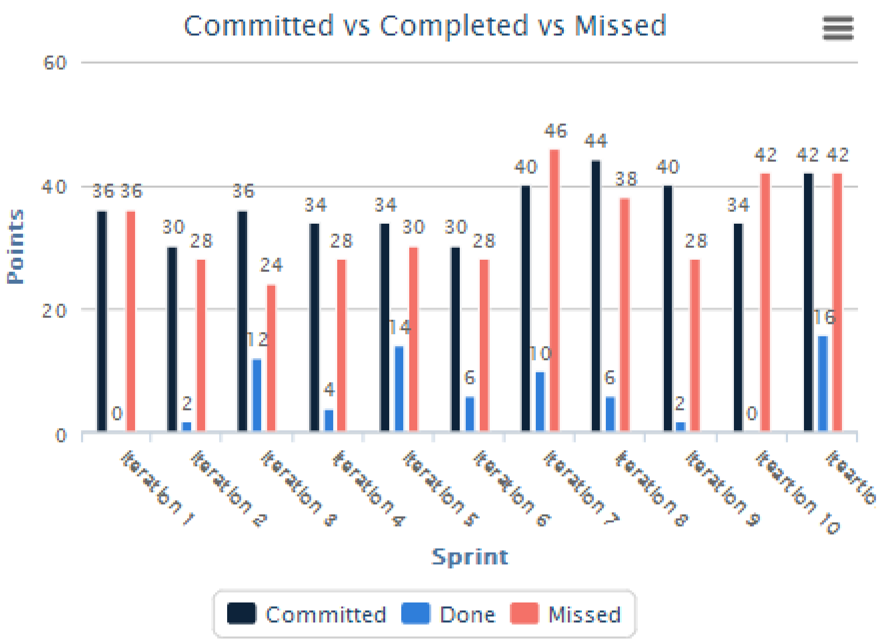
    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



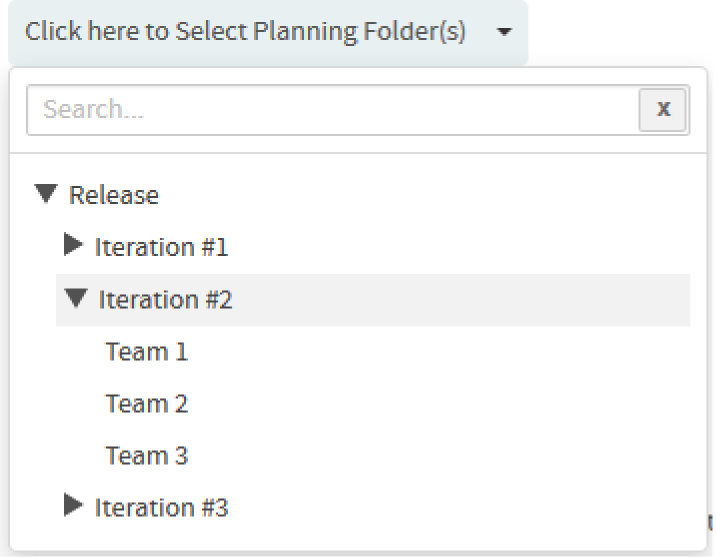
12. Click **Back to Reports List** to go back to the Reports dashboard.

Committed vs Done vs Missed

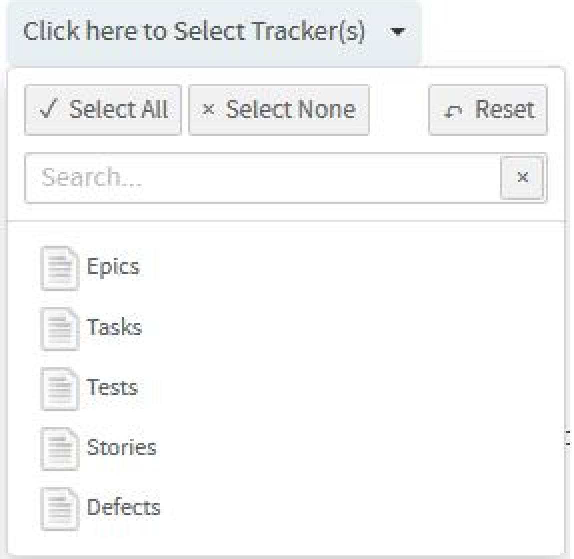
The Committed vs Done vs Missed chart shows a comparison of the amount of work (in terms of story points) committed, completed and missed for a sprint.



1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.
3. Select **Committed vs Done vs Missed** from **Agile Reports**.
4. Type a report title and description.
5. Select a planning folder from the **SELECT PLANNING FOLDER(S)** drop-down list.



6. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.



7. Leave the **CHART DISPLAY TYPE** as *Columndone*, which is the only available chart type for this report.

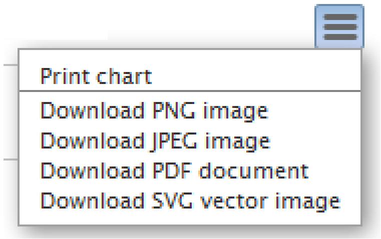
8. Select report visibility: `Public` or `Private`.

9. Click **Preview**.

10. Click **Create**. The report is created and the **View Report** page appears.

Print or download charts

You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



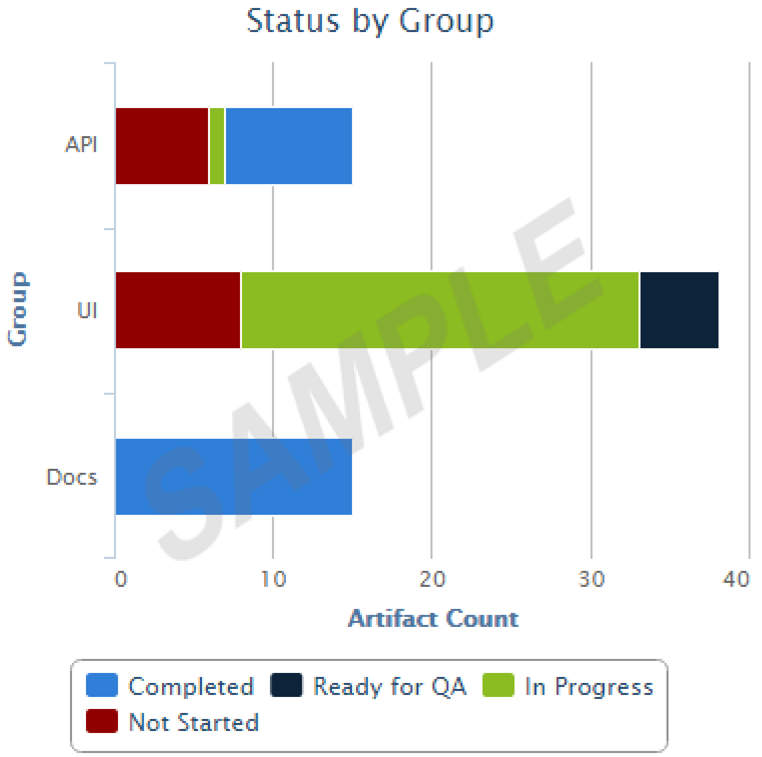
11. Click **Back to Reports List** to go back to the Reports dashboard.

Distribution Reports

Here’s a list of reports to see distribution of work items such as tracker artifacts by parameters such as status, size, effort and so on.

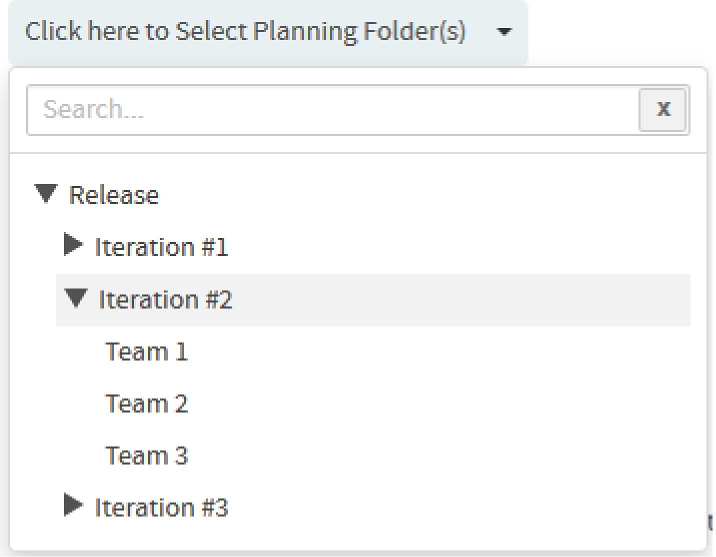
Status Distribution by Area or Group

You can generate this chart for a specific sprint or release planning folder (optionally you can include child planning folders, if any) and take stock of the number of artifacts in different statuses and have them grouped by `Group`, `Category`, `Customer`, `Assigned To`, `Priority` or `Teams`.



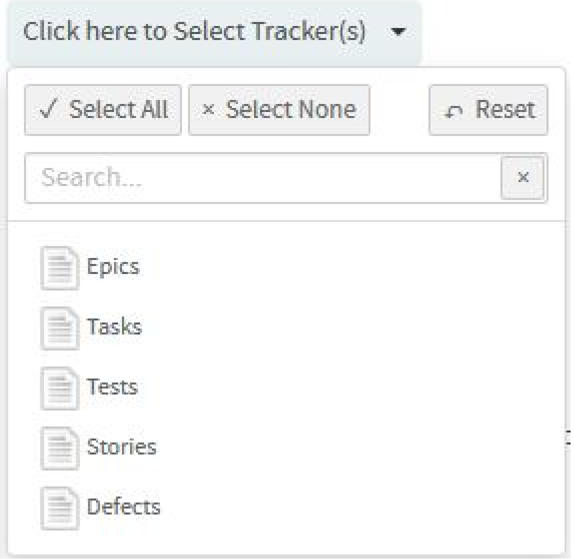
NOTE: Unlike other reports (that use datamart), this report runs on your operational database.

1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.
3. Select **Status Distribution by Area/Group** from **Distribution Reports**.
4. Type a report title and description.
5. Select a planning folder from the **SELECT PLANNING FOLDER(S)** drop-down list.



6. Optionally, select the **Include: Child planning folders** check box to have the child planning folders, if any, included.

7. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.



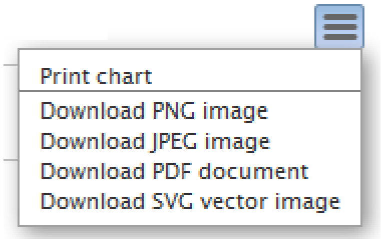
8. Select an option from the **GROUP BY** drop-down list to have the chart grouped by `Group`, `Category`, `Customer`, `Assigned To`, `Priority` or `Teams`.

9. Leave the **CHART DISPLAY TYPE** as *Bar*, which is the only available chart type for this report.

10. Select report visibility: `Public` or `Private`.
11. Click **Preview**.
12. Click **Create**. The report is created and the **View Report** page appears.

    Print or download charts

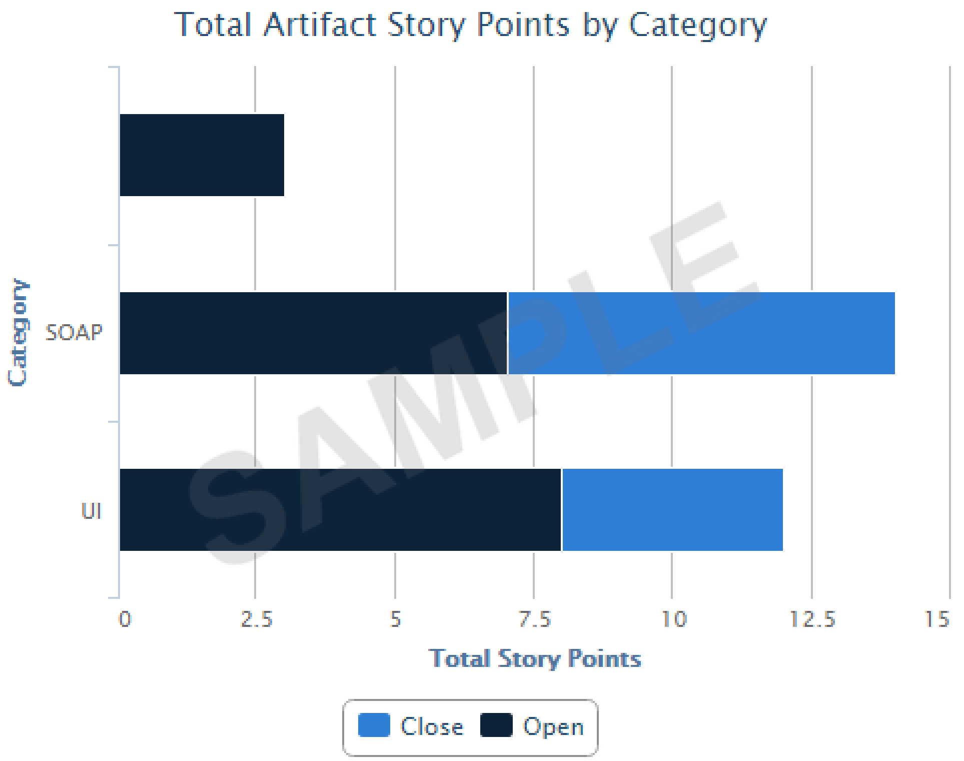
    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



13. Click **Back to Reports List** to go back to the Reports dashboard.

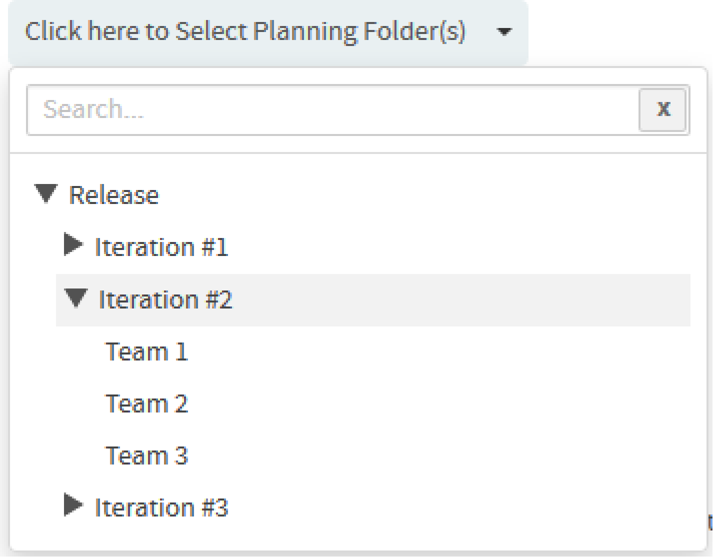
Total Size by Area or Group

You can generate this chart for a specific sprint or release planning folder (optionally you can include child planning folders, if any) and take stock of the total size (as in total of story points or estimated, actual or remaining effort) of artifacts and have them grouped by `Group`, `Category`, `Customer`, `Assigned To`, `Priority`, `Tracker` or `Teams`. You can generate this report for all artifacts or just for open or closed artifacts.

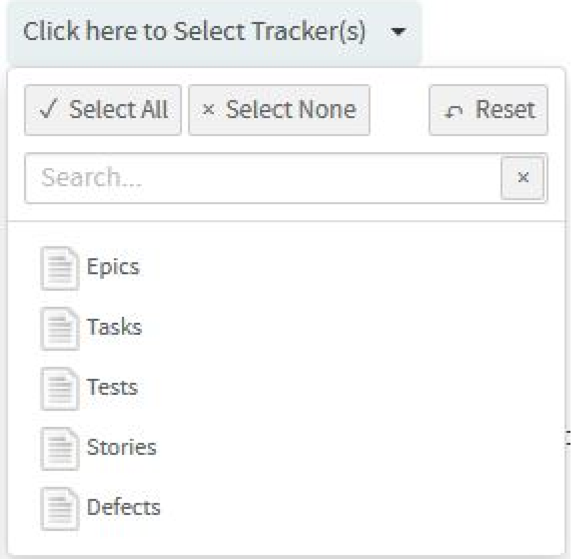


NOTE: Unlike other reports (that use datamart), this report runs on your operational database.

1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.
3. Select **Total Size by Area/Group** from **Distribution Reports**.
4. Type a report title and description.
5. Select a planning folder from the **SELECT PLANNING FOLDER(S)** drop-down list.



6. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.



7. Optionally, select the **Include: Child planning folders** check box to have the child planning folders, if any, included.

8. Select what you want to total from the **TOTAL BY** drop-down list. You can total by `Story Points` or `Estimated Effort` or `Actual Effort` or `Remaining Effort`.

9. Select an option from the **GROUP BY** drop-down list to have the chart grouped by `Group`, `Category`, `Customer`, `Assigned To`, `Priority`, `Tracker` or `Teams`.

10. Select one of the options, Include artifacts: **All** (default) or **Open** or **Closed**, to include all artifacts or just open or closed artifacts respectively.

11. Leave the **CHART DISPLAY TYPE** as *Bar*, which is the only available chart type for this report.

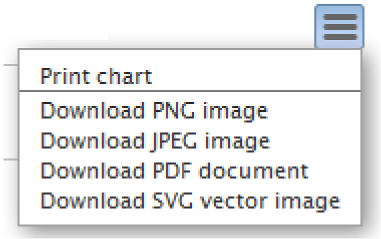
12. Select report visibility: `Public` or `Private`.

13. Click **Preview**.

14. Click **Create**. The report is created and the **View Report** page appears.

    Print or download charts

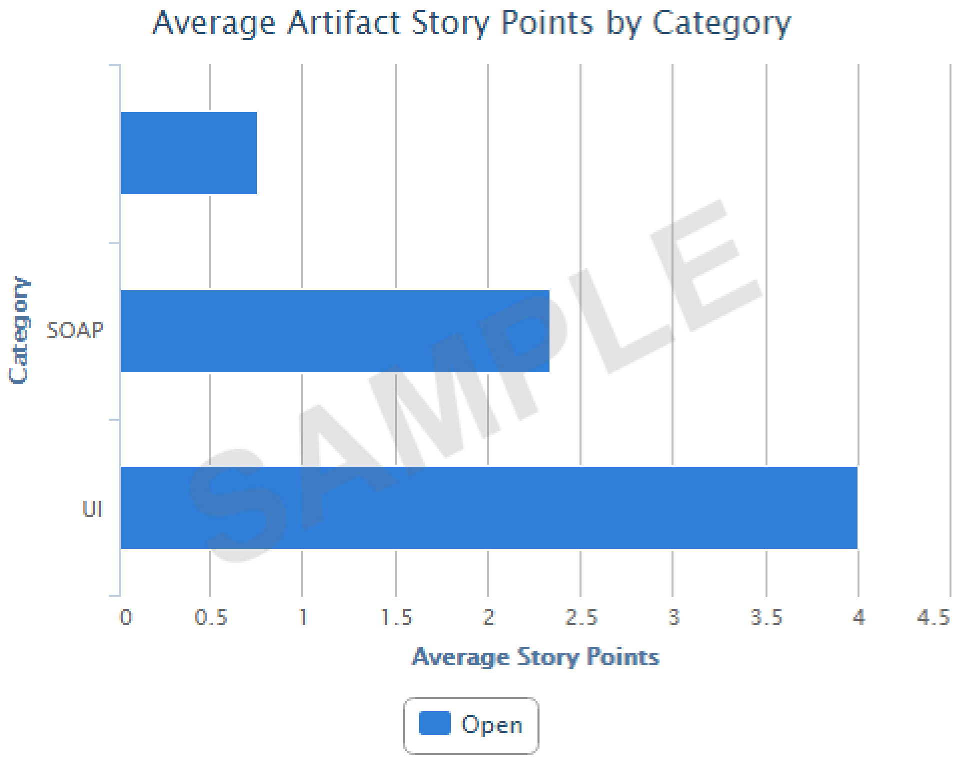
    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



15. Click **Back to Reports List** to go back to the Reports dashboard.

Average Size by Area or Group

You can generate this chart for a specific sprint or release planning folder (optionally you can include child planning folders, if any) and take stock of the average size (as in average of story points or estimated, actual or remaining effort) of artifacts and have them grouped by `Group`, `Category`, `Customer`, `Assigned To`, `Priority`, `Tracker` or `Teams`. You can generate this report for all artifacts or just for open or closed artifacts.



NOTE: Unlike other reports (that use datamart), this report runs on your operational database.

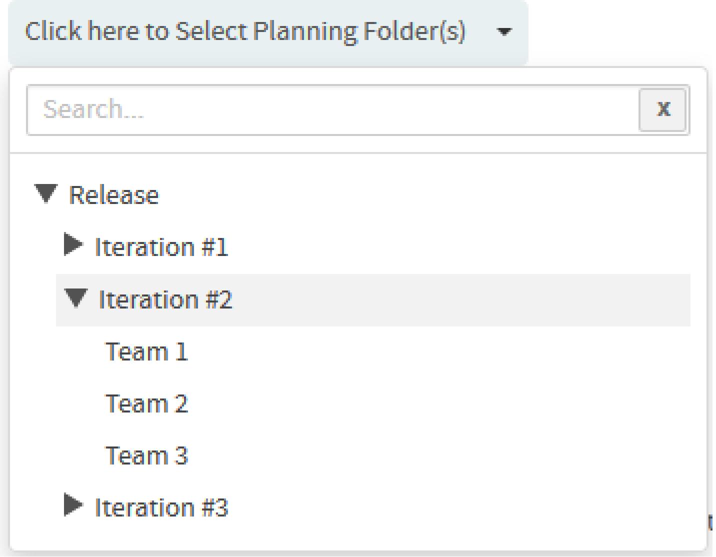
1. Click **REPORTS** from the **Project Home** menu.

2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.

3. Select **Average Size by Area/Group** from **Distribution Reports**.

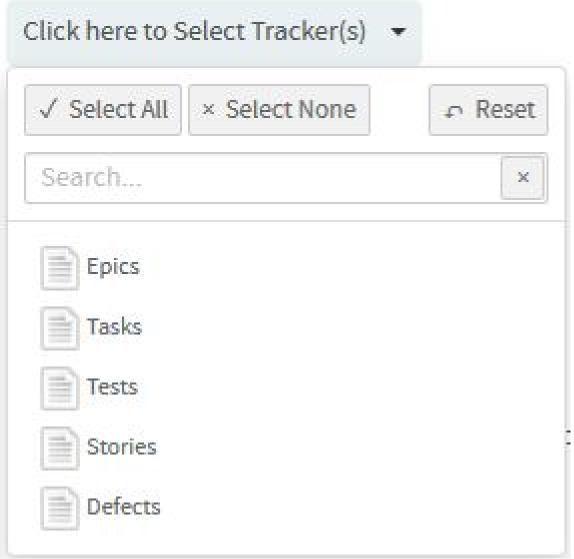
4. Type a report title and description.

5. Select a planning folder from the **SELECT PLANNING FOLDER(S)** drop-down list.



6. Optionally, select the **Include: Child planning folders** check box to have the child planning folders, if any, included.

7. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.



8. Select what you want to average from the **AVERAGE BY** drop-down list. You can average by `Story Points` or `Estimated Effort` or `Actual Effort` or `Remaining Effort`.

9. Select an option from the **GROUP BY** drop-down list to have the chart grouped by `Group`, `Category`, `Customer`, `Assigned To`, `Priority`, `Tracker` or `Teams`.

10. Select one of the options, Include artifacts: **All** (default) or **Open** or **Closed**, to include all artifacts or just open or closed artifacts respectively.

11. Leave the **CHART DISPLAY TYPE** as *Bar*, which is the only available chart type for this report.

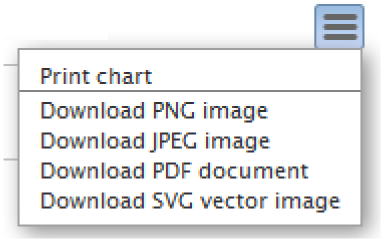
12. Select report visibility: `Public` or `Private`.

13. Click **Preview**.

14. Click **Create**. The report is created and the **View Report** page appears.

    Print or download charts

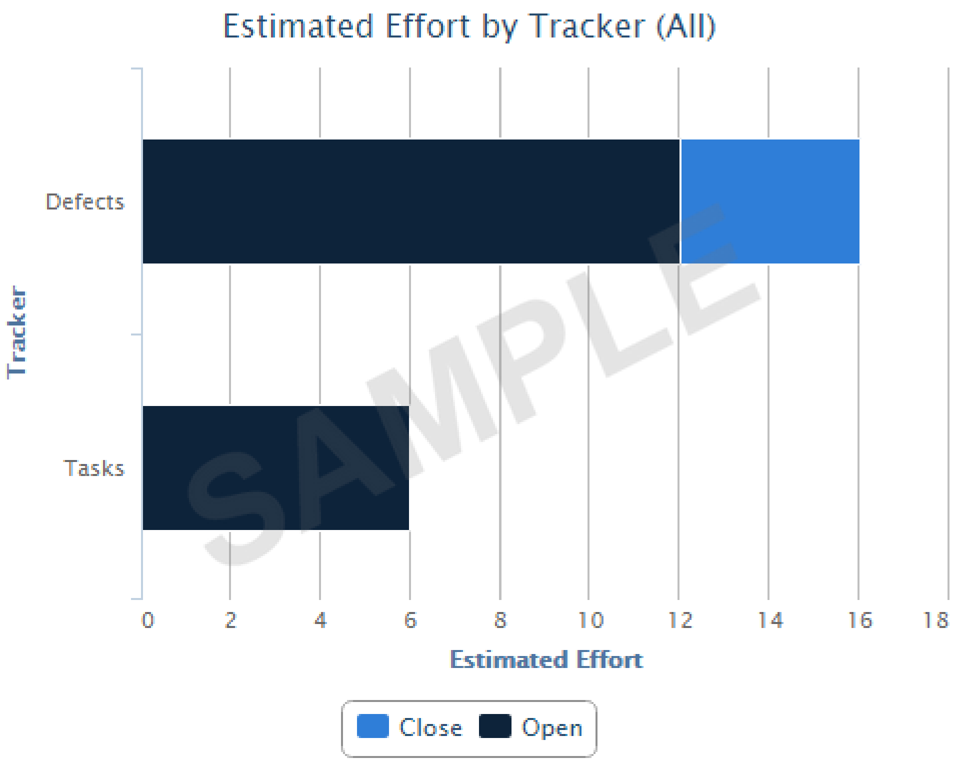
    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



15. Click **Back to Reports List** to go back to the Reports dashboard.

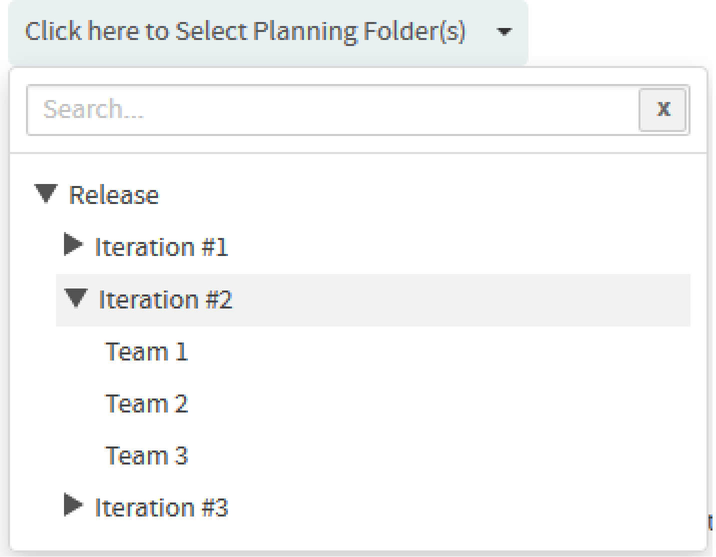
Total Size by Tracker Type

You can generate this chart for a specific planning folder (optionally you can include child planning folders, if any) and take stock of the total size of artifacts belonging to one or more tracker types and have them aggregated by `Story Points` or `Actual Effort` or `Estimated Effort` or `Remaining Effort`. This report is more useful if you want to know the total size by tracker types for a release and so is typically run against a specific release planning folder. You can generate this report for all artifacts or just for open or closed artifacts.



NOTE: Unlike other reports (that use datamart), this report runs on your operational database.

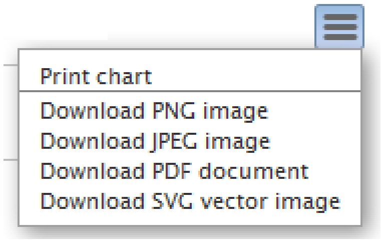
1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.
3. Select **Total Size by Tracker Type** from **Distribution Reports**.
4. Type a report title and description.
5. Select a planning folder from the **SELECT PLANNING FOLDER(S)** drop-down list.



6. Optionally, select the **Include: Child planning folders** check box to have the child planning folders, if any, included.
7. Select what you want to aggregate from the **AGGREGATE BY** drop-down list. You can aggregate by `Story Points` or `Estimated Effort` or `Actual Effort` or `Remaining Effort`.
8. Select one of the options, Include artifacts: **All** (default) or **Open** or **Closed**, to include all artifacts or just open or closed artifacts respectively.
9. Leave the **CHART DISPLAY TYPE** as *Bar*, which is the only available chart type for this report.
10. Select report visibility: `Public` or `Private`.
11. Click **Preview**.
12. Click **Create**. The report is created and the **View Report** page appears.

    Print or download charts

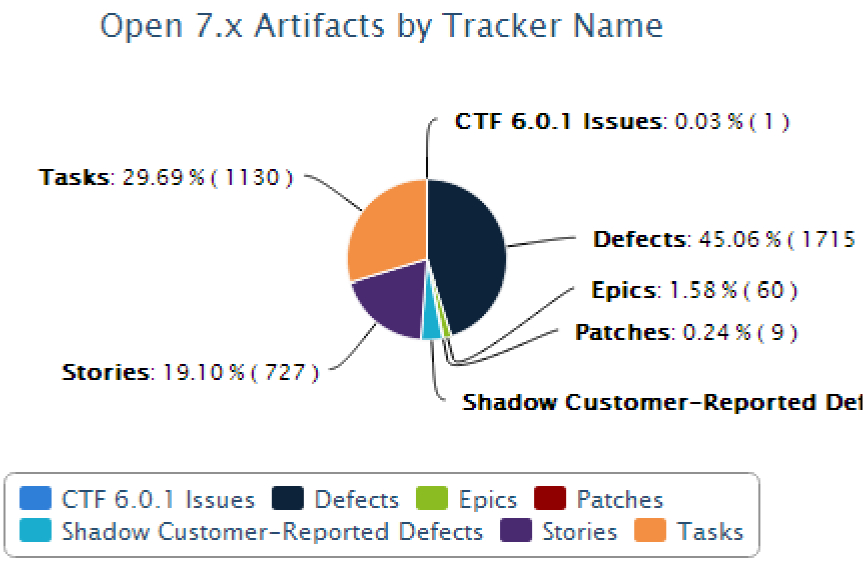
    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



13. Click **Back to Reports List** to go back to the Reports dashboard.

Artifact Distribution Chart (Multiple Trackers)

Using this chart, you can report the status of artifacts belonging to more than one tracker or planning folder. You can also create charts based on a combination of more than one tracker and planning folder.



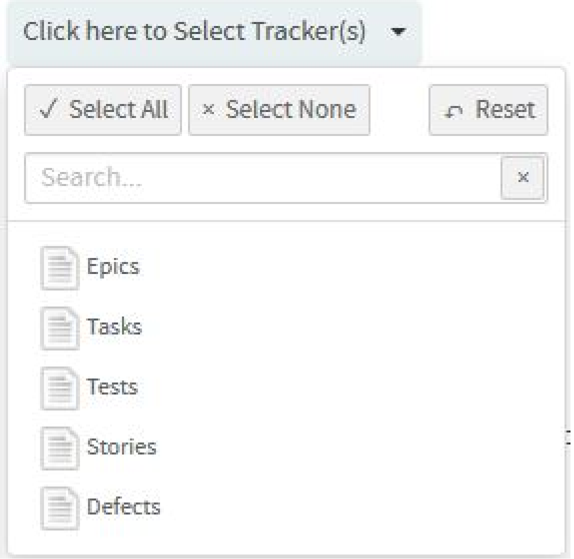
1. Click **REPORTS** from the **Project Home** menu.

2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.

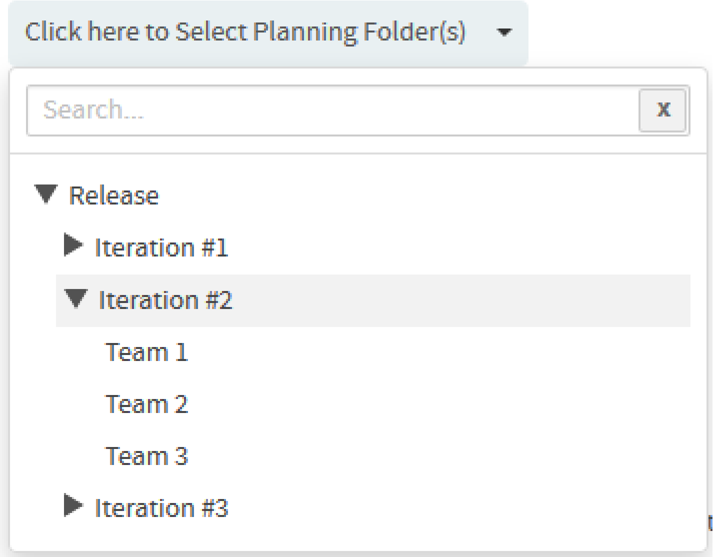
3. Select **Artifact Distribution Chart (Multiple Trackers)** from **Distribution Reports**.

4. Type a report title and description.

5. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.



6. Select a planning folder from the **SELECT PLANNING FOLDER(S)** drop-down list.

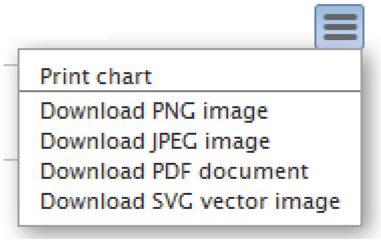


7. Select one of the options from the **DISTRIBUTE BY** drop-down list.
8. Select one of the options from the **OPEN VS CLOSE** drop-down list.
9. Leave the **CHART DISPLAY TYPE** as *Pie*, which is the only available chart type for this report.
10. Select report visibility: `Public` or `Private`.
11. Click **Preview**.

12. Click **Create**. The report is created and the **View Report** page appears.

    Print or download charts

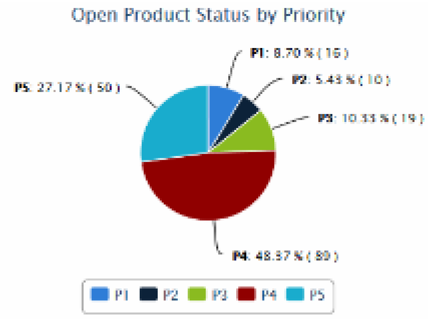
    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



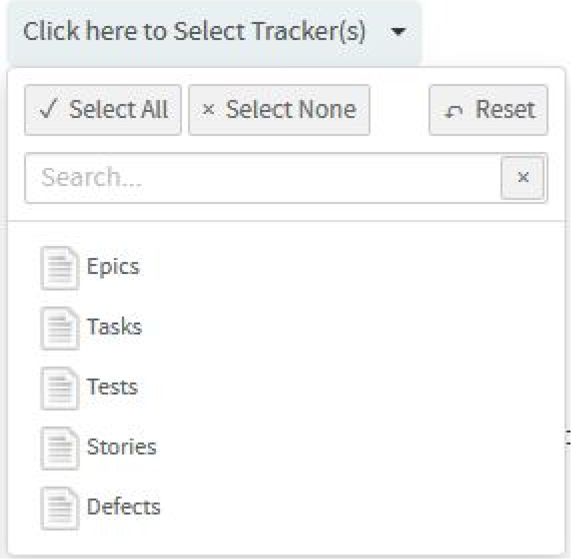
13. Click **Back to Reports List** to go back to the Reports dashboard.

Artifact Distribution Chart (Single Tracker)

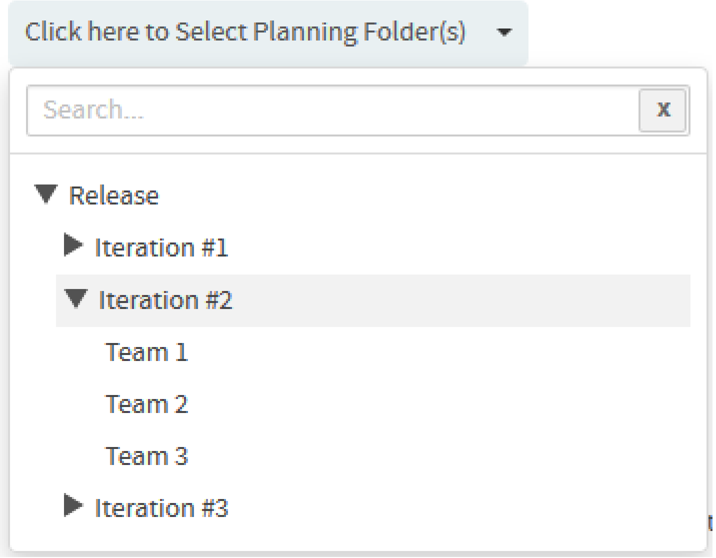
This chart comes in handy if you want to report the status of artifacts belonging to a specific tracker.



1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.
3. Select **Artifact Distribution Chart (Single Tracker)** from **Distribution Reports**.
4. Type a report title and description.
5. Select a tracker from the **SELECT TRACKER(S)** drop-down list.



6. Select one or more planning folders (select the check boxes) from the **SELECT PLANNING FOLDER(S)** drop-down list.



7. Select one of the options from the **DISTRIBUTE BY** drop-down list.

8. Select one of the options from the **OPEN VS CLOSE** drop-down list.

9. Leave the **CHART DISPLAY TYPE** as *Pie*, which is the only available chart type for this report.

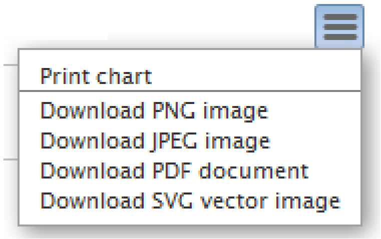
10. Select report visibility: `Public` or `Private`.

11. Click **Preview**.

12. Click **Create**. The report is created and the **View Report** page appears.

    Print or download charts

    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



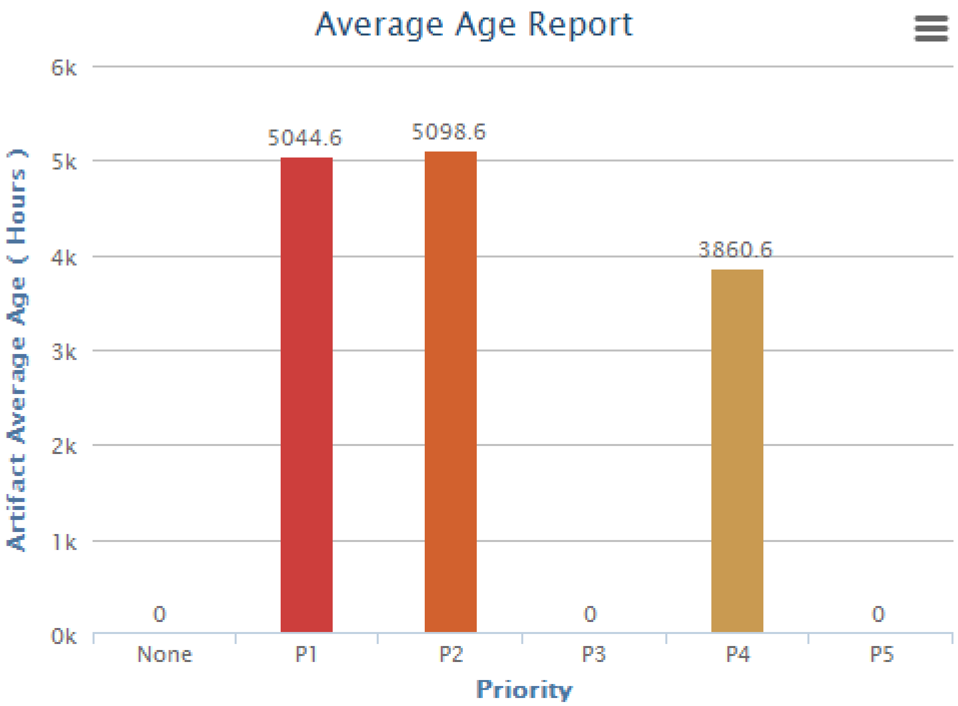
13. Click **Back to Reports List** to go back to the Reports dashboard.

Trend Reports

Here's a list of reports to know some artifact trend data.

Average Age Report

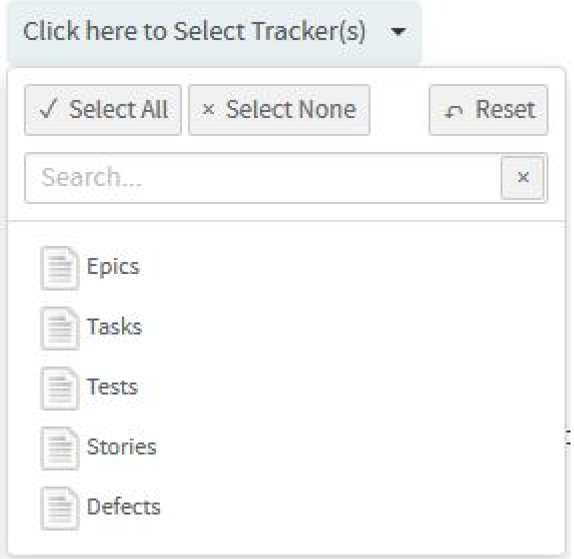
The average age chart lets you know the average age of artifacts in one or more trackers you select.



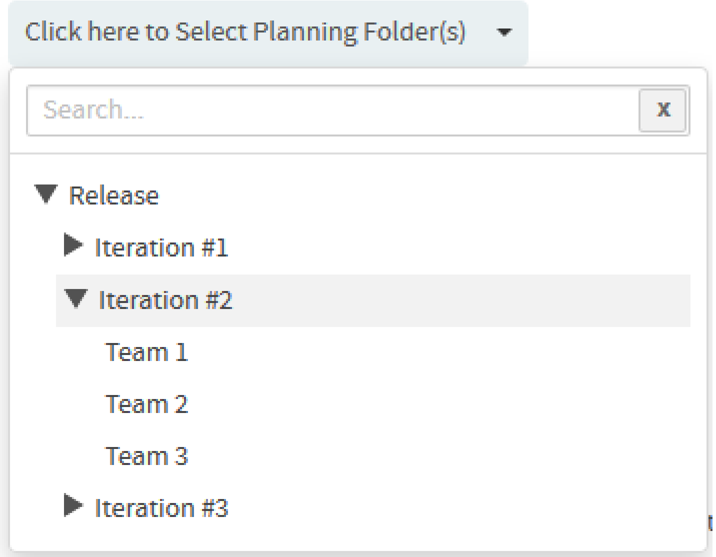
You can:

- Generate this report to know the average age of artifacts in various units of time such as number of hours, days or weeks.
- Group this report either by artifact priority such as P1, P2 and so on or by both artifact priority and category.
- Generate this report for either open or closed artifacts.
- Exclude weekends from being included in the average age calculation.

1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.
3. Select **Average Age Report** from **Trend Reports**.
4. Type a report title and description.
5. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.



6. Select a planning folder from the **SELECT PLANNING FOLDER(S)** drop-down list.



7. Select one of the time units such as hours, days or weeks from the **AVG. TIME IN** drop-down list.
8. Select either **Priority** or **Priority & Category** from the **GROUP BY** drop-down list.
9. If required, select **EXCLUDE Weekends** check box to exclude weekends from being included in the average age calculation.
10. Select either `Open Only` or `Closed` from the **OPEN VS. CLOSE** drop-down list.

11. Leave the **CHART DISPLAY TYPE** as *Stackedcolumns*, which is the only available chart type for this report.

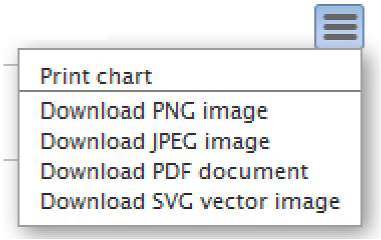
12. Select report visibility: `Public` or `Private`.

13. Click **Preview**.

14. Click **Create**. The report is created and the **View Report** page appears.

    Print or download charts

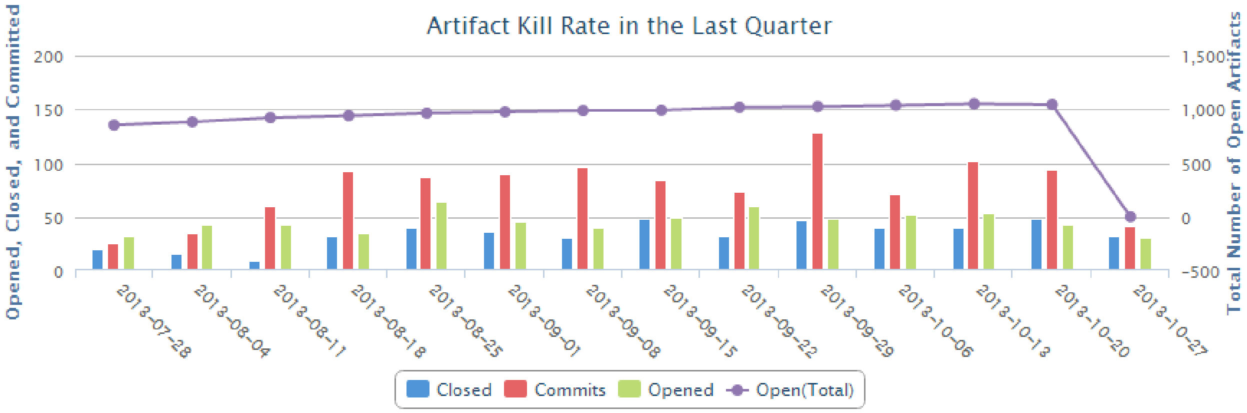
    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.



15. Click **Back to Reports List** to go back to the Reports dashboard.

Artifact Open/Close Chart (Multiple Trackers)

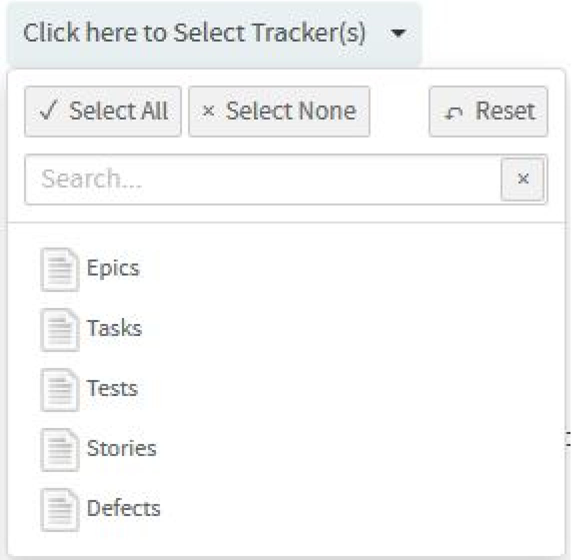
Use this chart to know the number of opened and closed artifacts over a period of time across one or more trackers or planning folders. This chart also helps in knowing the number of associated commits in specific repositories over a period of time. You can also know the total number of open artifacts through this chart.



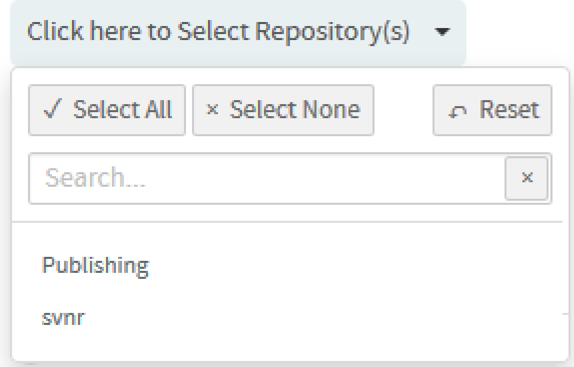
The number of opened and closed artifacts, and the number of associated commits are shown as color-coded bars (scale on left Y coordinate axis) and the total number of open artifacts is shown as a color-coded line (scale on right Y coordinate axis). The X coordinate axis shows the time scale of the chart.

At least one of the reporting parameters such as the **Tracker ID, Planning Folder ID or the Repository ID** is required to generate this chart. The following table lists the various reporting parameters for Artifact - Open/Close Chart (Multiple Trackers).

1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page. The **Select Report Type** page appears.
3. Select **Artifact Open/Close Chart (Multiple Trackers)** from **Trend Reports**.
4. Type a report title and description.
5. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.



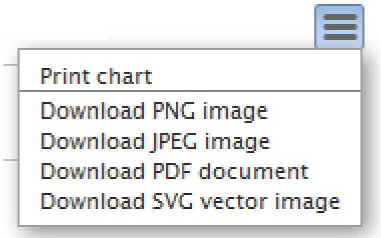
6. Select one or more planning folders (select the check boxes) from the **SELECT PLANNING FOLDER(S)** drop-down list.
7. Select one or more repositories from the **SELECT REPOSITORY(S)** drop-down list.



8. Select one or more artifact priorities. For example, select P1 and P2 to include P1 and P2 artifacts in your chart. Select the **None** to include artifacts that have no priority assigned to them.
9. Select one of the report durations: Last Week/Last Month/Last Quarter/Last Year`.
10. Leave the **CHART DISPLAY TYPE** as *Dualaxeslinecolumn*, which is the only available chart type for this report.
11. Select report visibility: `Public` or `Private`.
12. Click **Preview**.
13. Click **Create**. The report is created and the **View Report** page appears.

    Print or download charts

    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.

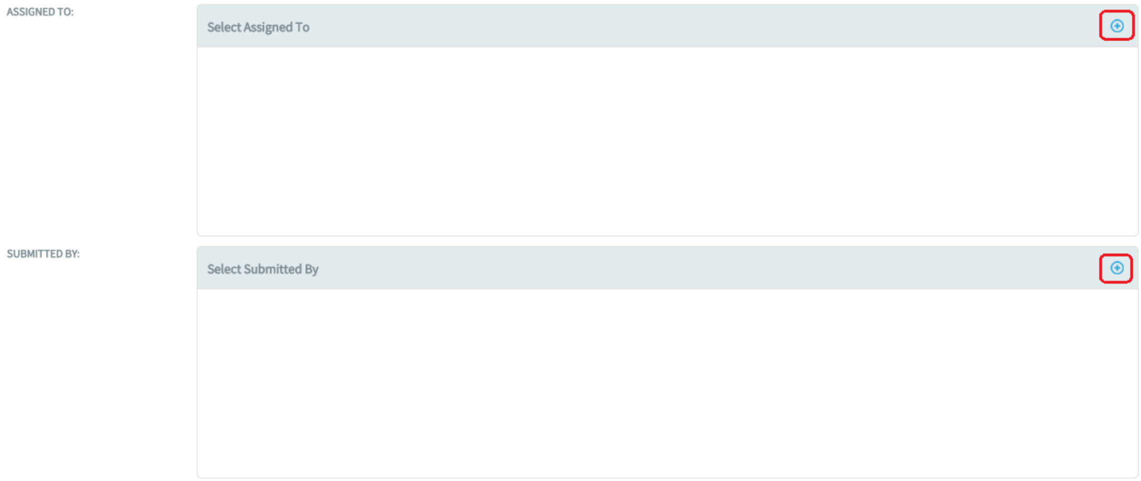


14. Click **Back to Reports List** to go back to the Reports dashboard.

Table Reports: Task and Tracker Reports

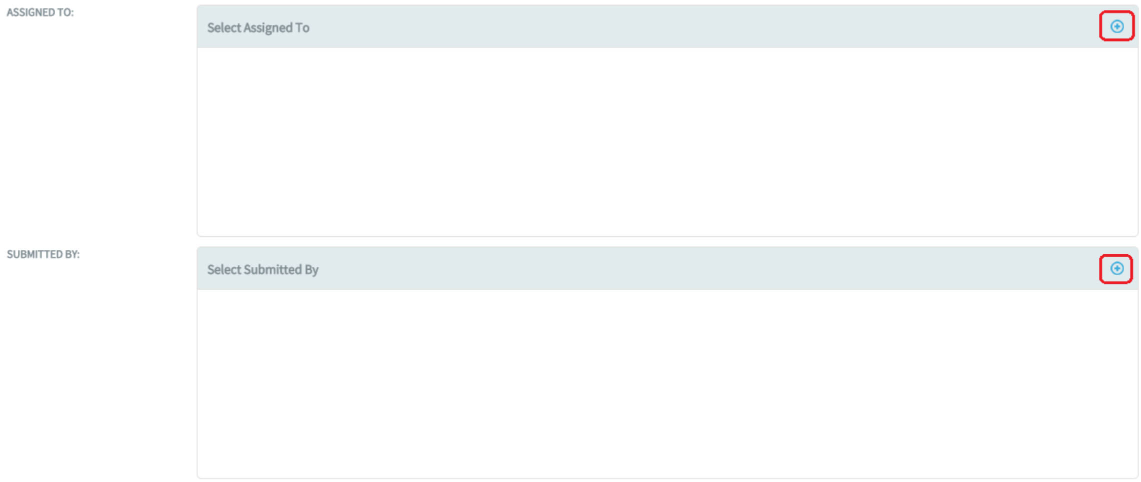
These are reports on trackers and tasks in a tabular format.

- **Task reports** - Task reports display selected summary data about project tasks. You can generate reports on the tasks in a selected project or across multiple projects.
    - Click **REPORTS** from the **Project Home** menu.
    - Click **Create** in the **List Reports** page. The **Select Report Type** page appears.
    - Select **Task** from **Table Reports**.
    - **Title** and **Description** - Type a title and description for your report.
    - **Project(s)** - Select the project or projects from which you want task data to be reported.
    - **Status(s)** - Select the status of the tasks to be included in your report.
    - **Priority(s)**: Select one or more priorities of tasks to be included in your report.
    - Select one or more **Assigned To** or **Submitted By** project members. Click the '+' icon to add members.



-
    - **Start Date** and **End Date** - Select the start and end date ranges of tasks to be included in your report.
    - **Report Field(s)** - Select the fields you want displayed in your report.
    - Select one of the report visibility options: `Public` (default) or `Private`.
    - Click **Create**. The **View Report** page appears. The task report is created.

- **Tracker reports** - Tracker reports give you a summary of the history of tracker artifacts. A tracker report can cover the tracker artifacts in a selected tracker or across all trackers in a project.
    - Click **REPORTS** in the project navigation bar.
    - Click **Create** in the **List Reports** page. The **Select Report Type** page appears.
    - Select **Tracker** from **Table Reports**.
    - To create a tracker report across multiple projects, click **Show All Projects** and select the trackers from the required projects.
    - **Title** and **Description** - Type a title and description for your report.
    - **Select Tracker(s)** - Select the trackers from which you want tracker artifact data to be reported. Group By: Have the report grouped by one of the following: Assigned To, Category, Customer, Group, Priority, Status or Team.
    - **Summary Statistics** - Select one of the following statistics to summarize the report: Count of Artifacts, Sum of Points or Sum of Effort.
    - **Priority(s)** - Select one or more priorities of artifacts to be included in your report.
    - **Select Artifact Maturity(s)** - Select one or all of the following values to have all or new or modified artifacts included in the report respectively: `Any` or `New` or `Edited`.
    - Select date ranges for create date (**Submitted On**), last edited date (**Last Modified**), or closed date (**Closed**).
    - Select one or more **Assigned To** or **Submitted By** project members. Click the '+' icon to add members.



- Select a planning folder form the **Select Planning Folder(s)** drop-down list.

- Select one of the report visibility options: `Public` (default) or `Private`.

- Click **Next**. A list of filters (such as category, status, reported in release, fixed in release and so on) for the selected trackers are shown. Select the filter values from the drop-down lists and click **Create**. Based on the criteria being selected for filtering, it returns the configured rows.

  The **View Report** page appears. The default value of the token **MAX_REPORT_ROWS** is set to '300' in `site-options.conf` file. For trackers with more than 300 artifacts, the tracker report is generated based on the criteria:

  - If you have selected a single tracker having more than 300 artifacts, the tracker report shows only the first 300 artifacts. To see all the artifacts in that tracker, click the **Export** button.

  - If you have selected multiple trackers with more than 300 artifacts in all, then the tracker report shows the artifacts based on the following calculation:

    No. of artifacts in report = **MAX_REPORT_ROWS** ÷ No. of selected trackers

  For example, if you have selected 3 trackers with more than 300 artifacts in all, the first 100 artifacts is displayed in the tracker report for each tracker. You can export the report for each tracker to view all the artifacts in it.

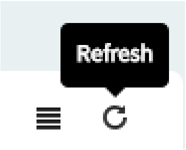
  > **IMPORTANT:** In a tracker report, when you select more than one value from a multi-select, user-defined field as filter and if an artifact is associated with all of the selected values, then

that artifact’s record is duplicated for each of the selected values.

For example, assume that you have a ‘Select User’ multi-select, user-defined field with values ‘User 1’, ‘User 2’ and ‘User 3’ in a tracker report. All these three values are associated with ‘artifact 1001’. Select all three values as filter and generate the tracker report. You will see ‘artifact 1001’ record being duplicated, that is, you will see three individual ‘artifact 1001’ records created for each of the three users.

Run a report to get fresh data on the status of your project.

The report data is regenerated each time the report is run. You can also refresh the data by clicking the ‘Refresh’ icon in the **List Reports** page.



TIP: You can modify the report criteria before generating the report data. For example, you might want to run the same report each week, using all of the same report criteria except the start and end dates.

NOTE: Report results contain only those items that you have permission to view. Other project members with different permissions might get different results when running the same report.

1. Click **REPORTS** from the **Project Home** menu.
2. Click the title of the desired report. The report is generated and displayed.
3. To edit a report, click **REPORTS**, select the report you want to edit (select the check box), click **Edit**, make the changes, and click **Save**.
4. To export report data, click **REPORTS**, select the table report you want to export (click the report title).

   NOTE: You can export only Table reports.

   IMPORTANT: For multi-select user-defined select user field, you will export individual record for each user.

5. On the **Export Data** window, select an export format (.csv, .xml, .xlsx or tab-delimited file) and select the columns you want on the report and click **Export**.

6. To delete reports, click **REPORTS**, select one or more reports you want to delete (select the check boxes) and click **Delete**. A confirmation message is displayed. Click **OK** to delete the selected reports.

You can customize your reports by modifying the parameters in config.ini file. For example, some reports require a particular data source such as datamart and some require a particular license type such as ALM.

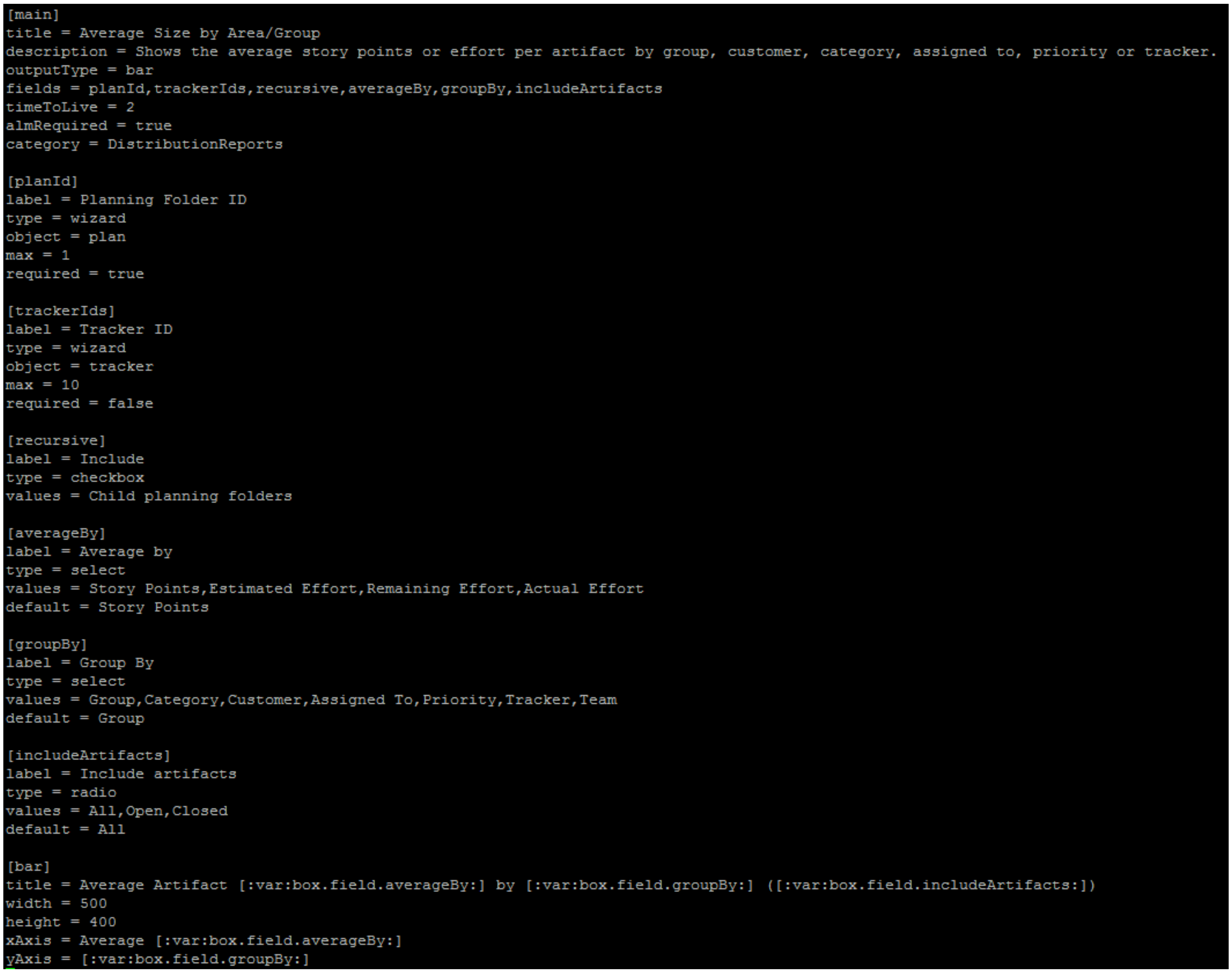
You may want to hide or display reports depending on the availability of their data source or the type of license so that users do not see reports which they cannot run.

For example:

- There are reports which require a data store (datamart or EventQ's events) to be enabled.
- There are many reports that do not apply to SCM users.

Further, you can also change a report's title, description or timeToLive duration of cached data and so on.

You can customize reports by modifying the parameters in `config.ini` file. For example, if you want to customize the "Average Size by Area/Group" report, modify the parameters in `opt/collabnet/teamforge/var/cliserver/app/reports/pkg/averageSizeByArea/config.ini` file. The following illustration shows a sample `config.ini` file.



```
[main]
title = Average Size by Area/Group
description = Shows the average story points or effort per artifact by group, customer, category, assigned to, priority or tracker.
outputType = bar
fields = planId,trackerIds,recursive,averageBy,groupBy,includeArtifacts
timeToLive = 2
almRequired = true
category = DistributionReports

[planId]
label = Planning Folder ID
type = wizard
object = plan
max = 1
required = true

[trackerIds]
label = Tracker ID
type = wizard
object = tracker
max = 10
required = false

[recursive]
label = Include
type = checkbox
values = Child planning folders

[averageBy]
label = Average by
type = select
values = Story Points,Estimated Effort,Remaining Effort,Actual Effort
default = Story Points

[groupBy]
label = Group By
type = select
values = Group,Category,Customer,Assigned To,Priority,Tracker,Team
default = Group

[includeArtifacts]
label = Include artifacts
type = radio
values = All,Open,Closed
default = All

[bar]
title = Average Artifact [:var:box.field.averageBy:] by [:var:box.field.groupBy:] ([:var:box.field.includeArtifacts:])
width = 500
height = 400
xAxis = Average [:var:box.field.averageBy:]
yAxis = [:var:box.field.groupBy:]
```

Here’s a sample list of parameters you can modify:

- **almRequired**
  - If *almRequired*=`true` - The site should be in the ALM mode and the users should have an ALM license to operate the ALM reports.
  - If *almRequired*=`false`- The site can be in either of the modes; both ALM and SCM licensed users can operate these reports.
- **devopsrequired**
  - If *devopsrequired*=`true` - Users should have DevOps license to operate DevOps reports.
- **scmrequired**

    - If *scmrequired*=`true` - Users should have Version Control license to operate SCM reports.

- **eventsRequired**
    - If *eventsRequired*=`true` - EventQ's events data store should be installed. Users can operate these reports irrespective of the site mode and license type.
    - If *eventsRequired*=`false` - Users cannot operate these reports irrespective of the site mode and license type.
- **datamartRequired**
    - If *datamartRequired*=`true` - Datamart should be installed. Users can operate these reports irrespective of the site mode and license type.
    - If *datamartRequired*=`false` - Users cannot operate these reports irrespective of the site mode and license type.
- **category**
    - You can change the category of a report by specifying it in `config.ini`. For example, to change the category of a specific report from 'Agile' to 'Distribution', specify as shown below:

      category=*DistributionReports*
- **timeToLive**
    - If *timeToLive*=`2` - The cached report data invalidated and refreshed every 2 hours.
    - The `timeToLive` parameter sets the duration (in hours) after which the cached report data is invalidated and refreshed from the data source. The default `timeToLive` for reports that use operational database is 2 hours. It is 24 hours for reports that use datamart. You can change the `timeToLive` duration depending on your site's requirements. However, shorter timeToLive duration can impact performance, though.

Using external reporting and OLAP tools, query the datamart directly and generate reports. The database schema diagrams provide the means to create advanced query scripts to extract required information from the datamart.

Accessing the datamart for reporting provides more analytical data than is provided by the TeamForge user interface options. The external tool must connect directly to the datamart and is then granted read-only permission to all data in the datamart.

NOTE: It is recommended that you limit the access to the datamart to a limited set of trusted users.

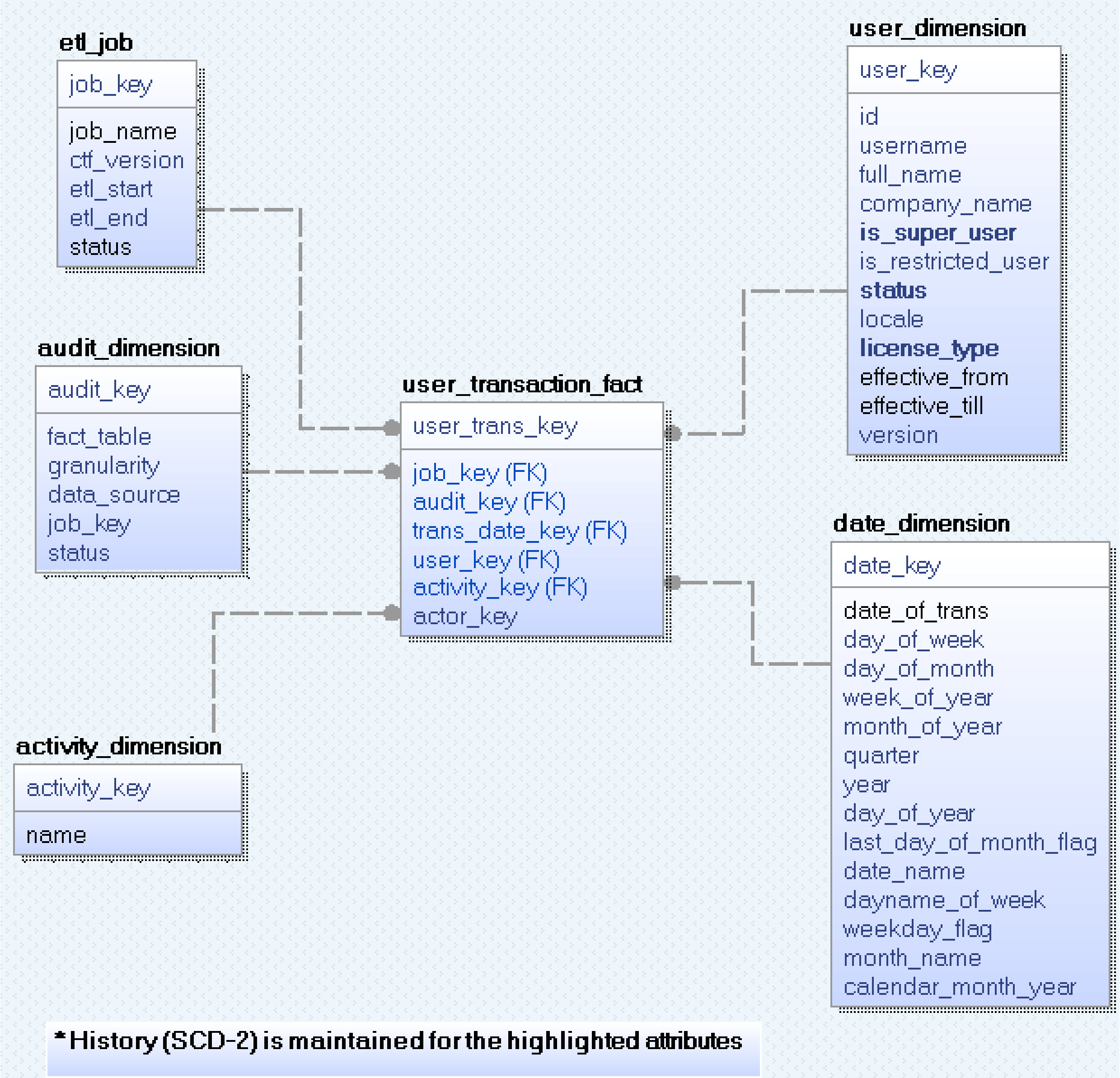
Datamart Schemas

The `Users`, `SCM`, and `Tracker` schemas are documented here for use in queries to the datamart. The datamart uses a “Star Schema” design for tables.

User Schema

Query the user schema to obtain useful information in the user database tables detailed here in a schema diagram.

User schema contains user login information; TeamForge captures one fact row for each user that is logged in during the day.



Description of User Schema

- **etl_job** Used to track the ETL run status. There is one record per ETL run for a job, for example, Tracker ETL or User ETL. `etl_job`has a 1-to-many relationship with `audit_dimension` since a job may update more than one fact table. All report generation queries must “join” the `etl_job` table with the condition `etl_job.status=1`, thereby discarding data from incomplete ETL runs.

- **audit_dimension** Holds metadata about fact table records. There is one record per fact table for an ETL run.
- **date_dimension** Conformed dimension used for all transaction times.
- **user_dimension** Used for string user attributes and is a “slowly changing dimension of type 2 (SCD-2).” `is_super_user`, `status`, and `license_type` are the SCD-2 fields.
- **activity_dimension** Conformed dimension that stores the activity or transaction names for various activities being tracked.
- **user_transaction_fact** A fact-less fact table with user data of “daily” granularity.

Sample Queries

You can obtain useful user information by querying the user database, and further refine the results by using filters on the “date”, “user type” (admin or non), “status”, and “license type” fields. For example:

- Number of users who are logged in, by day, over a period of time:

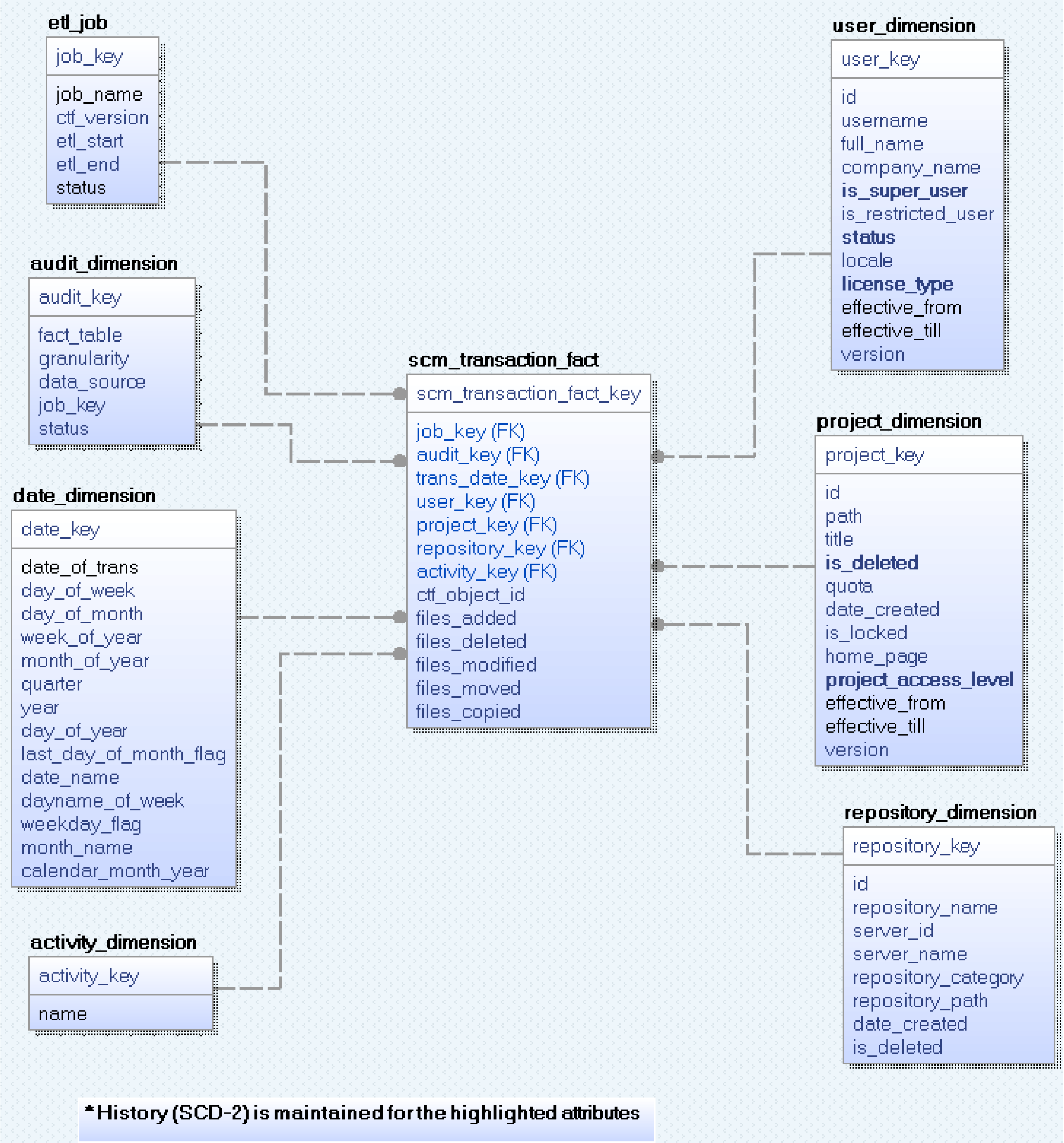
```
SELECT c.date_of_trans as Date, count(distinct(b.id)) as NumUsers

FROM user_transaction_fact a, user_dimension b, date_dimension c, etl_job
d
WHERE a.user_key=b.user_key and a.trans_date_key=c.date_key and a.job_key=
d.job_key
and d.status=1 and c.date_of_trans >= '2012-12-17' and c.date_of_trans <=
'2012-12-21'
GROUP BY c.date_of_trans
```

- List of users who have logged in:

```
SELECT c.date_of_trans as Date, b.username as UserName
FROM user_transaction_fact a, user_dimension b, date_dimension c, etl_job
d
WHERE a.user_key=b.user_key and a.trans_date_key=c.date_key and a.job_key=
d.job_key
and d.status=1 and c.date_of_trans >= '2012-12-17' and c.date_of_trans <=
'2012-12-21'
GROUP BY c.date_of_trans, b.username
```

SCM Schema

Query the SCM schema to obtain useful commit information in the SCM tables detailed here in a schema diagram.



Description of SCM Schema

- **etl_job** Used to track the ETL run status. There is one record per ETL run for a job, for example, Tracker ETL or User ETL. `etl_job` has a 1-to-many relationship with `audit_dimension` since a job may update more than one fact table. All report generation queries must "join" the `etl_job` table with the condition `etl_job.status=1`, thereby discarding data from incomplete ETL runs.

- **audit_dimension** Holds metadata about fact table records. There is one record per fact table for an ETL run.

- **date_dimension** Conformed dimension used for all transaction times.

- **activity_dimension** Conformed dimension that stores the activity or transaction names for various activities being tracked.

- **user_dimension** Used for string user attributes and is a "slowly changing dimension of type 2 (SCD-2)." `is_super_user`, `status`, and `license_type` are the SCD-2 fields.

- **project_dimension** Used for storing project attributes and is a "slowly changing dimension of type 2 (SCD-2)." `is_deleted` and `project_access_level` are the SCD-2 fields.

- **repository_dimension** Used for storing repository attributes and is a "slowly changing dimension of type 1."

- **scm_transaction_fact** The fact table for SCM activities with "transaction" granularity. TeamForge inserts a row in this table for every SCM activity that it processes in a transaction.

  - TeamForge object id , if available.
  - Number of files added, deleted, modified, moved, copied, if applicable.

Sample Queries

You can obtain useful SCM information by querying the SCM database. For example:

- Number of SCM commits, sorted by date:

```
select b.date_of_trans as Date,
         count(a.scm_transaction_fact_key) as NumCommits
         from scm_transaction_fact a, date_dimension b
         where a.trans_date_key=b.date_key
         group by b.date_of_trans
```

- Number of SCM commits, with quarterly trend:

```
select 'Q'||b.quarter as Quarter,
         count(a.scm_transaction_fact_key) as NumCommits
         from scm_transaction_fact a, date_dimension b
         where a.trans_date_key=b.date_key
         group by b.quarter
```

- List of users who made commits.

```
select b.username as UserName,
         count(a.scm_transaction_fact_key) as NumCommits
         from scm_transaction_fact a, user_dimension b
         where a.user_key=b.user_key
         group by b.username
```

- Project-wise commit data:

```
select b.id as ProjectId, b.title as ProjectName,
         count(a.scm_transaction_fact_key) as NumCommits
         from scm_transaction_fact a, project_dimension b
         where a.project_key=b.project_key
         group by b.id, b.title
```

- Commits by date, in a specific project:

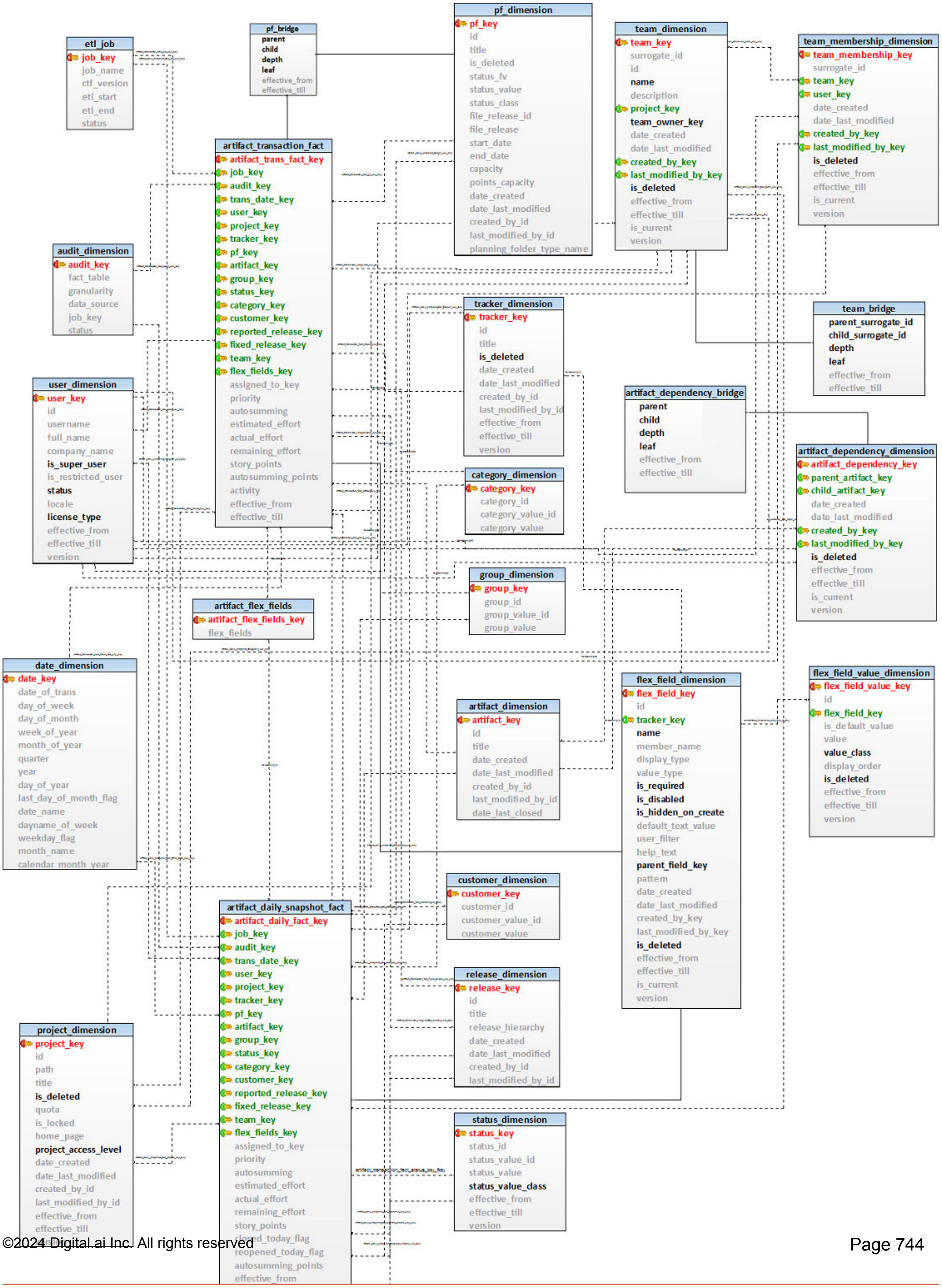
```
select c.date_of_trans as Date, b.id as ProjectId, b.title as ProjectName,

         count(a.scm_transaction_fact_key) as NumCommits
         from scm_transaction_fact a, project_dimension b, date_dimension
c
         where a.project_key=b.project_key and a.trans_date_key=c.date_key
           and b.id='proj1008'
         group by c.date_of_trans, b.id, b.title
```

Tracker Schema

Query the tracker schema to obtain useful information in the tracker database tables detailed here in a schema diagram.

TeamForge constructs the state or field values of the artifact during each update. The schema addresses both activity queries and total count queries.

TeamForge includes information for the activities in fixed fields, default artifact fields, and flex fields. This information includes artifact `create`, `move`, `update` for fixed-value changes, `delete`, `open_to_close`, and `close_to_open`.


etl_job
job_key
job_name
ctf_version
etl_start
etl_end
status
pf_bridge
parent
child
depth
leaf
effective_from
effective_till
pf_dimension
pf_key
id
title
is_deleted
status_fv
status_value
status_class
file_release_id
file_release
start_date
end_date
capacity
points_capacity
date_created
date_last_modified
created_by_id
last_modified_by_id
planning_folder_type_name
team_dimension
team_key
surrogate_id
id
name
description
project_key
team_owner_key
date_created
date_last_modified
created_by_key
last_modified_by_key
is_deleted
effective_from
effective_till
is_current
version
team_membership_dimension
team_membership_key
surrogate_id
team_key
user_key
date_created
date_last_modified
created_by_key
last_modified_by_key
is_deleted
effective_from
effective_till
is_current
version
artifact_transaction_fact
artifact_trans_fact_key
job_key
audit_key
trans_date_key
user_key
project_key
tracker_key
pf_key
artifact_key
group_key
status_key
category_key
customer_key
reported_release_key
fixed_release_key
team_key
flex_fields_key
assigned_to_key
priority
autosumming
estimated_effort
actual_effort
remaining_effort
story_points
autosumming_points
activity
effective_from
effective_till
audit_dimension
audit_key
fact_table
granularity
data_source
job_key
status
tracker_dimension
tracker_key
id
title
is_deleted
date_created
date_last_modified
created_by_id
last_modified_by_id
effective_from
effective_till
version
team_bridge
parent_surrogate_id
child_surrogate_id
depth
leaf
effective_from
effective_till
user_dimension
user_key
id
username
full_name
company_name
is_super_user
is_restricted_user
status
locale
license_type
effective_from
effective_till
version
artifact_dependency_bridge
parent
child
depth
leaf
effective_from
effective_till
category_dimension
category_key
category_id
category_value_id
category_value
artifact_dependency_dimension
artifact_dependency_key
parent_artifact_key
child_artifact_key
date_created
date_last_modified
created_by_key
last_modified_by_key
is_deleted
effective_from
effective_till
is_current
version
group_dimension
group_key
group_id
group_value_id
group_value
artifact_flex_fields
artifact_flex_fields_key
flex_fields
date_dimension
date_key
date_of_trans
day_of_week
day_of_month
week_of_year
month_of_year
quarter
year
day_of_year
last_day_of_month_flag
date_name
dayname_of_week
weekday_flag
month_name
calendar_month_year
flex_field_dimension
flex_field_key
id
tracker_key
name
member_name
display_type
value_type
is_required
is_disabled
is_hidden_on_create
default_text_value
user_filter
help_text
parent_field_key
pattern
date_created
date_last_modified
created_by_key
last_modified_by_key
is_deleted
effective_from
effective_till
is_current
version
flex_field_value_dimension
flex_field_value_key
id
flex_field_key
is_default_value
value
value_class
display_order
is_deleted
effective_from
effective_till
version
artifact_dimension
artifact_key
id
title
date_created
date_last_modified
created_by_id
last_modified_by_id
date_last_closed
customer_dimension
customer_key
customer_id
customer_value_id
customer_value
artifact_daily_snapshot_fact
artifact_daily_fact_key
job_key
audit_key
trans_date_key
user_key
project_key
tracker_key
pf_key
artifact_key
group_key
status_key
category_key
customer_key
reported_release_key
fixed_release_key
team_key
flex_fields_key
assigned_to_key
priority
autosumming
estimated_effort
actual_effort
remaining_effort
story_points
closed_today_flag
reopened_today_flag
autosumming_points
effective_from
release_dimension
release_key
id
title
release_hierarchy
date_created
date_last_modified
created_by_id
last_modified_by_id
project_dimension
project_key
id
path
title
is_deleted
quota
is_locked
home_page
project_access_level
date_created
date_last_modified
created_by_id
last_modified_by_id
effective_from
effective_till
status_dimension
status_key
status_id
status_value_id
status_value
status_value_class
effective_from
effective_till
version
artifact_transaction_fact_status_key

Description of Tracker Schema

- **etl_job** Used to track the ETL run status. There is one record per ETL run for a job, for example, Tracker ETL or User ETL. `etl_job` has a 1-to-many relationship with audit_dimension since a job may update more than one fact table. All report generation queries must “join” the `etl_job` table with the condition `etl_job.status=1`, thereby discarding data from incomplete ETL runs.

- **audit_dimension** Holds metadata about fact table records. There is one record per fact table for an ETL run.

- **date_dimension** Conformed dimension used for all transaction times.

- **user_dimension** Used for string user attributes and is a “slowly changing dimension of type 2 (SCD-2).” `is_super_user`, `status`, and `license_type` are the SCD-2 fields.

- **project_dimension** Used for storing project attributes and is a “slowly changing dimension of type 2 (SCD-2).” `is_deleted` and `project_access_level` are the SCD-2 fields.

- **pf_dimension** The planning folder dimension for storing planning folder attributes. Use this table to generate reports without planning folder hierarchy.

- **pf_bridge** A table for the planning folder hierarchy containing end-of-day status. `pf_bridge` is a “slowly changing dimension of type 2 (SCD-2).” You must limit queries to a given parent planning folder while joining with the `pf_bridge` table.Use `pf_bridge` to generate reports around planning folder hierarchy by joining with parent tables such as `artifact_daily_snapshot_fact` or `artifact_transaction_fact`.

  Joining `pf_bridge` with `artifact_transaction_fact` generates correct results only if end-of-day in queries is set to “12:00 a.m.”. While formulating queries with `pf_bridge`, please note:

  - `parent` and `child` fields — Contain values from `pf_dimension.pf_key;` a depth of 0 indicates self.

  - `leaf` field — Is true if the child is a leaf node, otherwise false. Every planning folder will have an entry here with a depth of 0; every parent planning folder will have entries for all of its children, recursively, up to the leaf node of all branches.

  - `effective_from` and `effective_till` fields — Indicate the period when the parent-child relationship is correct, and can be used in queries to get the hierarchy for a specified time period.

- **tracker_dimension** Used for storing tracker attributes and is a “slowly changing dimension of type 2 (SCD-2).” `is_deleted` is the changing field that act as a filter for reports.
- **artifact_dimension** Used for storing artifact data and is a “slowly changing dimension of type 1.”
- **group_dimension** Used for holding values of the TeamForge tracker field “group”. `group_dimension` has values for all artifacts from all Trackers.
- **status_dimension** Used for holding values of the TeamForge tracker field “status”. The `status_value_class` field represents the meta-status of an artifact and has values “Open” and “Close”. `status_dimension` has values for all artifacts from all Trackers.
- **category_dimension** Used for holding values of the TeamForge tracker field “category”. `category_dimension` has values for all artifacts from all Trackers.
- **customer_dimension** Used for holding values of the TeamForge tracker field “customer”. `customer_dimension` has values for all artifacts from all Trackers.
- **release_dimension** Used for holding values of the TeamForge tracker fields “Reported in Release” and “Fixed in Release”. `release_dimension` has values for all artifacts from all Trackers.
- **artifact_transaction_fact** Every change in fixed or default artifact field values, or to project or tracker artifacts, results in a row inserted to `artifact_transaction_fact`. Changes to flex-field values, adding comments, attachments, and so on do not add a row to the table. The `artifact_dimension.date_last_modified` field has the time in the source tracker at the time of the `ETL run.artifact transaction fact` can be used to generate reports around activities such as create, update, delete and move, and for intra-day reports. `artifact_transaction_fact` has a “transaction” granularity.
- **artifact_daily_snapshot_fact** An aggregate table that holds the daily snapshot data or end-of-day status. `artifact_daily_snapshot_fact` can be used to generate reports for artifact close & re-open counts. It is recommended to use this table for end-of-day reports as it has fewer rows compared to `artifact_transaction_fact`. `artifact_daily_snapshot_fact` has a “daily” granularity.
- **team_dimension** A dimension for holding values of Teams. This is a “slowly changing dimension of type 2 (SCD-2)”. Use this table to generate reports without team hierarchy.
- **team_membership_dimension** This is more or less a factless fact table that holds the changes in terms of Team memberships. This is a “slowly changing dimension of type 2 (SCD-2)”.

- **team_bridge** A table for handling team hierarchy containing end-of-day status. team_bridge is a "slowly changing dimension of type 2 (SCD-2)." You must limit queries to a given parent team while joining with the `team_bridge` table. Use `team_bridge` to generate reports around team hierarchy (ex: a report on all artifacts assigned to Team 1 and its child teams) by joining with parent tables such as `artifact_daily_snapshot_fact` or `artifact_transaction_fact`.

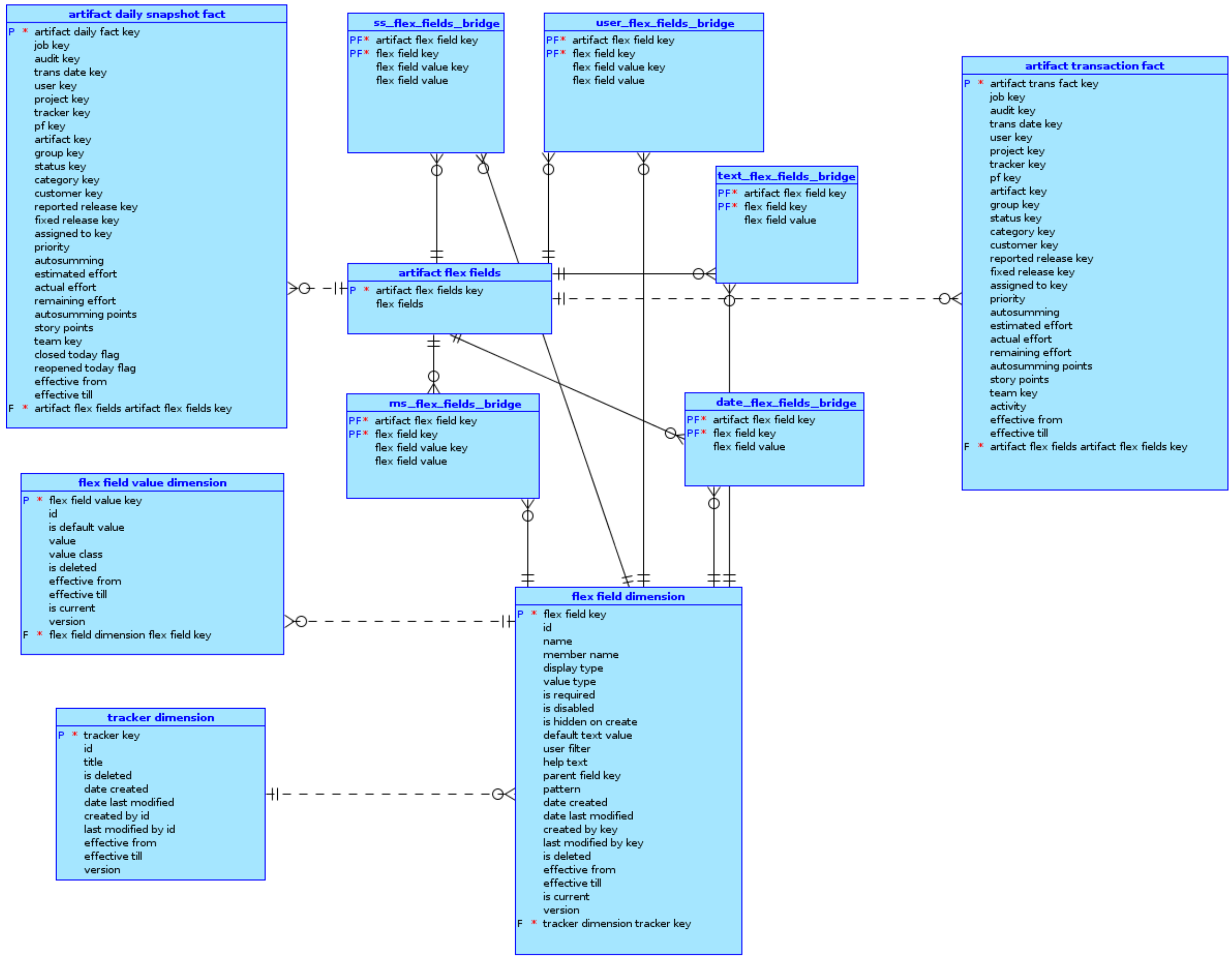
  Joining `team_bridge` with `artifact_transaction_fact` generates correct results only if end-of-day in queries is set to "12:00 a.m.". While formulating queries with `team_bridge`, please note:

  - `parent_surrogate_id` and `child_surrogate_id` fields — Contain values from `team_dimension.surrogate_id;` a depth of 0 indicates self.
  - `leaf` field — Is true if the child is a leaf node, otherwise false. Every team will have an entry here with a depth of 0; every parent team will have entries for all of its children, recursively, up to the leaf node of all branches .
  - `effective_from` and `effective_till` fields — Indicate the period when the parent-child relationship is correct, and can be used in queries to get the hierarchy for a specified time period.

  > **NOTE:** `team_dimension` is a `slowly changing dimension of type 2`. So, make sure that the JOIN clause in queries that join team_bridge and team_dimension includes the date and time as well.

- **flex_field_dimension** Used for storing information about flex fields and is a "slowly changing dimension of type 2 (SCD-2)".
- **flex_field_value_dimension** Used for storing information about flex field values and is a "slowly changing dimension of type-2 (SCD-2)".
- **artifact_dependency_dimension** Used for storing relationship between artifacts and is a "slowly changing dimension of type-2 (SCD-2)".
- **artifact_dependency_bridge** A table for the artifacts hierarchy containing end-of-day status. This is a "slowly changing dimension of type-2 (SCD-2). Use this table to generate reports around artifact hierarchy.
- **artifact_flex_fields** A table to bind the flex field updates on artifacts into a well formed XML, which stores the 'field key', 'value' and 'type' of the flex field.

Schema diagram for XML to non-XML conversion



Description of Schema used for XML to non-XML Conversion

- **flex field dimension** Used for storing information about flex fields and is a “slowly changing dimension of type 2 (SCD-2)”.

- **flex field value dimension** Used for storing information about flex field values and is a “slowly changing dimension of type-2 (SCD-2)”.

- **artifact flex fields** A table to bind the flex field updates on artifacts into a well formed XML, which stores the ‘field key’, ‘value’ and ‘type’ of the flex field.

- **artifact daily snapshot fact** An aggregate table that holds the daily snapshot data or end-of-day status. `artifact_daily_snapshot_fact` can be used to generate reports for artifact close & re-open counts. It is recommended to use this table for end-of-day reports as it has fewer rows compared to `artifact_transaction_fact`. `artifact_daily_snapshot_fact` has a “daily” granularity.

- **artifact transaction fact** Every change in fixed or default artifact field values, or to project or tracker artifacts, results in a row inserted to `artifact_transaction_fact`. Changes to flex-field values, adding comments, attachments, and so on do not add a row to the table. The `artifact_dimension.date_last_modified` field has the time in the source tracker at the time of the ETL run. `artifact transaction fact` can be used to generate reports around activities such as create, update, delete and move, and for intra-day reports. `artifact_transaction_fact` has a “transaction” granularity.
- **tracker dimension** Used for storing tracker attributes and is a “slowly changing dimension of type 2 (SCD-2).” `is_deleted` is the changing field that act as a filter for reports.
- **ss_flex_fields_bridge** A table to hold details about single select flex field assigned to an artifact. To get the details of the assigned artifact, the “ss_flex_fields_bridge” table must be joined with the “artifact_transaction_fact” table and the “artifact_dimension” table.
- **user_flex_fields_bridge** A table to hold details about user flex field assigned to an artifact. To get the details of the assigned artifact, the “user_flex_fields_bridge” table must be joined with the “artifact_transaction_fact” table and the “artifact_dimension” table.
- **text_flex_fields_bridge** A table to hold details about text flex field assigned to an artifact. To get the details of the assigned artifact, the “text_flex_fields_bridge” table must be joined with the “artifact_transaction_fact” table and the “artifact_dimension” table.
- **date_flex_fields_bridge** A table to hold details about date flex field assigned to an artifact. To get the details of the assigned artifact, the “date_flex_fields_bridge” table must be joined with the “artifact_transaction_fact” table and the “artifact_dimension” table.
- **ms_flex_fields_bridge** A table to hold details about multi select flex field assigned to an artifact. To get the details of the assigned artifact, the “ms_flex_fields_bridge” table must be joined with the “artifact_transaction_fact” table and the “artifact_dimension” table.

Sample Queries

You can obtain useful tracker information by querying the tracker database. For example:

- Number of artifacts created in a tracker, sorted by date:

```
SELECT b.date_of_trans, count(a.artifact_key)
          FROM artifact_transaction_fact a, date_dimension b, tracker_dime
nsion c
          WHERE a.trans_date_key=b.date_key and a.tracker_key=c.tracker_ke
y
          and a.activity='Create' and c.title='Tracker-1'
          and b.date_of_trans >= '2011-10-31' and b.date_of_trans <= '2011
-11-07'
          GROUP BY b.date_of_trans
          ORDER BY b.date_of_trans
```

- Number of artifacts created in a tracker, sorted by date and priority:

```
SELECT b.date_of_trans, d.title as Tracker, 'P'||a.priority as Priority,c.
id ArtifactId
         FROM artifact_transaction_fact a, date_dimension b, artifact_dime
nsion c, tracker_dimension d
         WHERE a.trans_date_key=b.date_key and a.artifact_key=c.artifact_k
ey and a. tracker_key=d.tracker_key
         and a.activity='Create' and d.title='Tracker-1'
         and b.date_of_trans >= '2011-10-31' and b.date_of_trans <= '2011-
11-07'
         ORDER BY b.date_of_trans, d.title, Priority
```

- Number of artifacts created on a particular day in a particular planning folder:

```
SELECT b.date_of_trans, c.id ArtifactId, c.title
         FROM artifact_transaction_fact a, date_dimension b, artifact_dime
nsion c, pf_dimension d, pf_bridge e  WHERE a.trans_date_key=b.date_key an
d a.artifact_key=c.artifact_key and a.pf_key=e.child
         and d.pf_key=e.parent and a.activity='Create' and d.title='Pf-1'
         and '2011-10-31' >= e.effective_from and '2011-10-31' < e.effecti
ve_till
         and b.date_of_trans = '2011-10-31'
         ORDER BY b.date_of_trans
```

- Number of closed tracker artifacts, sorted by day:

```
SELECT b.date_of_trans, count(a.artifact_key)
         FROM artifact_daily_snapshot_fact a, date_dimension b, tracker_di
mension c
```

```
         WHERE a.trans_date_key=b.date_key and a.tracker_key=c.tracker_key
 and a.closed_today_flag='true'
         and c.title='Tracker-1' and b.date_of_trans >= '2011-10-31' and b
.date_of_trans <= '2011-11-07'
         GROUP BY b.date_of_trans
         ORDER BY b.date_of_trans
```

- List of artifacts in the “Open” state, sorted by day:

```
SELECT a.date_of_trans, c.id, c.title
         FROM date_dimension a inner join artifact_daily_snapshot_fact b
         on a.date_of_trans >= date(b.effective_from) and a.date_of_trans
< date(b.effective_till)
         inner join artifact_dimension c on b.artifact_key=c.artifact_key
         inner join status_dimension d on b.status_key=d.status_key
         WHERE d.status_value_class='Open' and a.date_of_trans >= '2011-10
-31'
         and a.date_of_trans <= '2011-11-02'
         ORDER BY a.date_of_trans
```

- List of artifacts assigned to a team (including child teams) on a given day:

```
select ad.id as artifact_id
        from artifact_transaction_fact fact
        inner join date_dimension dd on (dd.date_of_trans + 1) between fac
t.effective_from and fact.effective_till
        inner join team_dimension child_td on child_td.team_key = fact.tea
m_key
        inner join team_bridge tb on tb.child_surrogate_id = child_td.surr
ogate_id and
              (dd.date_of_trans + 1) between tb.effective_from and tb.effe
ctive_till
        inner join team_dimension parent_td on tb.parent_surrogate_id = pa
rent_td.surrogate_id and
              (dd.date_of_trans + 1) between parent_td.effective_from and
parent_td.effective_till
        inner join artifact_dimension ad on ad.artifact_key = fact.artifac
t_key
        inner join etl_job ej on ej.job_key = fact.job_key and ej.status =
 1
```

```
        where dd.date_of_trans = '2015-01-01' and parent_td.id = 'team1002
'
        order by artifact_id
```

Datamart Access Using External Tools

Use external tools to directly query the PostgreSQL or Oracle datamarts to access more TeamForge data than is available through options on the TeamForge user interface.

Accessing the Datamart

You can use OLAP or GUI tools to query your datamart directly to procure all the TeamForge data that is relevant to your analysis.

The external tool must be in the same network as the datamart to ensure fast access with a direct connection to the database. On the TeamForge host machine, you must enable `REPORTS_DATABASE_HOST` and `REPORTS_DATABASE_PORT` for remote access.

> **NOTE:** You only have read access; the datamart does not allow writes.

Enabling the Datamart for Access

You must enable the datamart for direct queries and re-start the site. Once enabled, all data in the datamart can be queried using external tools.

Enable TeamForge (PostgreSQL Datamart) for Queries from External Tools

You must enable, then re-start TeamForge so that you can use an external tool to query the datamart directly.

You cannot use external reporting tools to connect directly to a datamart if you have a single-box installation using localhost:SERVICES.

1. Edit the `site-options.conf` file. Set `REPORTS_DB_ACCESS_HOSTS` to the IP address or IP address range (CIDR address) from which the tool will establish connection to the datamart. Specify multiple IPs as a comma-separated list. For example: `REPORTS_DB_ACCESS_HOSTS = 10.0.0.2,10.2.1.0/24`.

> **NOTE:** If the TeamForge site is not within your network and a Network Address Translation (NAT) is configured, then specify the NAT's external facing IP in the token. If you have used an `advanced` installation method to install TeamForge, you must also manually add the IP addresses to the pg_hba.conf file.

2. In `site-options.conf` file, set `REPORTS_DATABASE_READ_ONLY_USER` and `REPORTS_DATABASE_READ_ONLY_PASSWORD` with your user and password.

3. On the TeamForge host machine, enable `REPORTS_DATABASE_HOST` and `REPORTS_DATABASE_PORT` for remote access.

4. Provision services.

   ```
   teamforge provision
   ```

> **NOTE:** TeamForge 17.4 (and later) installer expects the system locale to be LANG=en_US.UTF-8. TeamForge `provision` command fails otherwise.

You can now enable trusted users to establish connections by providing them with values of options `REPORTS_DATABASE_HOST` , `REPORTS_DATABASE_PORT`, `REPORTS_DATABASE_READ_ONLY_USER`, and `REPORTS_DATABASE_READ_ONLY_PASSWORD`.

Enable TeamForge (Oracle Datamart) for Queries from External Tools

You must enable, then restart TeamForge so that you can use an external tool to query the datamart directly.

You can use external reporting tools to connect directly to a datamart only if you do not have a single-box installation using `localhost:SERVICES`.

1. In `site-options.conf` file, set `REPORTS_DATABASE_READ_ONLY_USER` and `REPORTS_DATABASE_READ_ONLY_PASSWORD` with your user and password.

2. On the TeamForge host machine, enable `REPORTS_DATABASE_HOST` and `REPORTS_DATABASE_PORT` for remote access.

3. Provision services.

   ```
   teamforge provision
   ```

> **NOTE:** TeamForge 17.4 (and later) installer expects the system locale to be LANG=en_US.UTF-8. TeamForge `provision` command fails otherwise.

You can now enable trusted users to establish connections by providing them with values of options `REPORTS_DATABASE_HOST`, `REPORTS_DATABASE_PORT`, `REPORTS_DATABASE_READ_ONLY_USER`, and `REPORTS_DATABASE_READ_ONLY_PASSWORD`.

Common Errors While Connecting to PostgreSQL or Oracle Datamarts

You may encounter these errors while configuring your datamart to allow queries from external tools.

PostgreSQL Database Access Error

This error in a PostgreSQL datamart is because TeamForge is not granting access to your IP address.

```
Error: psql: FATAL: no pg_hba.conf entry for host "111.111.111.111", user "tes
t", database "datamart"
```

Solution: Have the TeamForge administrator grant access to your IP address.

PostgreSQL Authentication Error

```
Error: psql: FATAL: password authentication failed for user "test"
```

Solution: Specify the correct user name and password.

PostgreSQL Connection Error

```
Error: psql: could not connect to server: Connection refused. Is the server ru
nning on host "<datamart-host>" and accepting TCP/IP connections on port 5632?
```

Solution:

- Check the host and port numbers; the database must be running on host “datamart-host” and accept TCP/IP connections on port 5632.
- Check if remote access is enabled on TeamForge; the Teamforge administrator can enable the required access.

Oracle Connection Error

This error in an Oracle datamart occurs when some connection parameters are set incorrectly.

```
Error: ORA-12505, TNS:listener does not currently know of SID given in connect
 descriptor.
```

Solution: Check and set your connection parameters appropriately.

Query PostgreSQL Datamart Based on Flex Fields

Create query scripts to query and extract the required information from the PostgreSQL datamart. Make sure these scripts are available to any user (including users with read-only permission) who wants to execute these scripts. This topic lists the functions and sample queries for a few specific use cases.

Important:

✓ Log into TeamForge as a Reporting user and run the queries. Note that this is a one-time process.

✓ Use the following flex field functions by joining the flex_field_dimension, artifact_flex_field and artifact_transaction_fact (or) artifact_daily_snapshot_fact tables as it depends on the XML data and flex field key generated by the platform.

Common script used across all flex field functions

```
    CREATE OR REPLACE FUNCTION array_search(needle anyelement, haystack anyarr
ay)
      RETURNS integer AS
    $BODY$
        SELECT i
          FROM generate_subscripts($2, 1) AS i
         WHERE $2[i] = $1
      ORDER BY i
    $BODY$
      LANGUAGE sql STABLE
      COST 100;
```

Function for Date Flex Fields

Input Arguments

- Flex field XML (derived automatically by joining the `artifact_transaction_fact` (or) `artifact_daily_snapshot_fact` and `artifact_flex_field` tables).

- Flex field key (derived automatically from `flex_field_dimension` table and other tables).

Output Arguments

Date flex field values

```
  CREATE OR REPLACE FUNCTION get_artifact_date_value(artifact_xml xml, field_ke
y integer)
    RETURNS character varying AS
  $BODY$
  DECLARE
  field_value_key varchar(9055) default '';
   BEGIN
   SELECT cast((xpath('/fields/field/@val',artifact_xml)) [(array_search(field_ke
y::text,(xpath('/fields/field/@key',artifact_xml))::text[]))] as varchar) into fi
eld_value_key;
   Return field_value_key;
   END;
  $BODY$
    LANGUAGE plpgsql VOLATILE
    COST 100;
```

Function for Text Flex Fields

Input Arguments

- Flex field XML (derived automatically by joining the `artifact_transaction_fact` (or) `artifact_daily_snapshot_fact` and `artifact_flex_field tables`).
- Flex field key (derived automatically from `flex_field_dimension` table and other tables).

Output Arguments

Text flex field values

```
  CREATE OR REPLACE FUNCTION replacen(artifact_xml1 xml)
    RETURNS text AS
  $BODY$
   select replace ((select replace(artifact_xml1::varchar(3000),'<!','&lt;!'))
,']>',']&gt;')

  $BODY$
    LANGUAGE sql STABLE
    COST 100;


  CREATE OR REPLACE FUNCTION get_artifact_text_value(artifact_xml xml, field_ke
y integer)
```

```
    RETURNS character varying AS
  $BODY$
   DECLARE
    field_value_key varchar(9055) default '';
   BEGIN
   SELECT cast((xpath('/fields/field/@val',replacen(artifact_xml)::xml)) [(array
_search(field_key::text,(xpath('/fields/field/@key',replacen(artifact_xml)::xml))
::text[]))] as varchar) into field_value_key;
   Return field_value_key;
   END;
   $BODY$
     LANGUAGE plpgsql VOLATILE
     COST 100;
```

Function for Single-select Flex Fields

Input Arguments

- Flex field XML (derived automatically by joining the `artifact_transaction_fact` (or) `artifact_daily_snapshot_fact` and `artifact_flex_field` tables).
- Flex field key (derived automatically from `flex_field_dimension` table and other tables).

Output Arguments

Values selected or stored in single-select flex field

```
  CREATE OR REPLACE FUNCTION get_artifact_select_value(artifact_xml xml, field_
key integer)
    RETURNS character varying AS
 $BODY$
  DECLARE
   field_value_key varchar(9055) default '';
  singleSelectValues varchar(9055) default '';
  BEGIN
   SELECT cast((xpath('/fields/field/@val',artifact_xml)) [(array_search(field_ke
y::text,(xpath('/fields/field/@key',artifact_xml))::text[]))] as varchar) into fi
eld_value_key;
   Select value into singleSelectValues  from flex_field_value_dimension where fl
ex_field_value_key::text = (field_value_key::text);

  Return singleSelectValues;
  END;
 $BODY$
   LANGUAGE plpgsql VOLATILE
   COST 100;
```

Function for Multi-select Flex Fields

Input Arguments

- Flex field XML (derived automatically by joining the `artifact_transaction_fact` (or) `artifact_daily_snapshot_fact` and `artifact_flex_field` tables).
- Flex field key (derived automatically from `flex_field_dimension` table and other tables).
- Specific value(s) stored in the field or the keyword 'ALL' for retrieving all values stored in multi-select flex field.
- Logical conditional operator. Possible values are 'ALL' or 'ANY'. The argument value of 'ALL' performs a search equivalent to the 'AND' condition and the value of 'ANY' performs a search equivalent to the 'OR' condition.

Output Arguments

Values selected or stored in multi-select flex field

```
  -----First Execute Inner Function------
-------Inner function start------
CREATE OR REPLACE FUNCTION get_artifact_multiselect_values_any(artifact_xml xm
l, field_key integer, selectedfields text[])
RETURNS text AS
$BODY$
DECLARE
fieldValues text;
multiFieldValues text default '';
splitfield text[];
splitfield2 text[];
loop1 integer DEFAULT 1;
 loop2 integer DEFAULT 1;
arrayLength integer DEFAULT 1;
 arrayLength2 integer DEFAULT 1;
multifield text DEFAULT '';
filteredField text Default '';
BEGIN
 Select cast((xpath('/fields/field/@val',artifact_xml))
[array_search(field_key::text,(xpath('/fields/field/@key',artifact_xml)::text[]))
]
 as varchar(9055)) into fieldValues;
Select array_length(regexp_split_to_array(fieldValues, ','),1) into arrayLength
;
Select regexp_split_to_array(fieldValues, ',') into splitfield;
IF arrayLength IS NOT NULL THEN
        FOR loop1 IN 1..arrayLength LOOP
        Select value into multiFieldValues from flex_field_value_dimension where
 flex_field_value_key = splitfield[loop1]::integer;
        multifield := concat(multifield,',',multiFieldValues);
        EXIT WHEN loop1 > arrayLength;
        END LOOP;


        multifield := rtrim(multifield,'" ');
        multifield := ltrim(multifield,',');

Select array_length(regexp_split_to_array(array_to_string(selectedFields,','),
 ','),1) into arrayLength2;
Select regexp_split_to_array(array_to_string(selectedFields,','), ',') into sp
litfield2;
IF selectedFields = Array['ALL'] THEN
filteredField:=multifield;
Return filteredField;
ELSE
FOR loop2 IN 1..arrayLength2 LOOP
IF   multifield ~ splitfield2[loop2] THEN
      filteredField := concat(splitfield2[loop2],',',filteredField);
END IF;
EXIT WHEN loop2 > arrayLength2;
END LOOP;
END IF;
END IF;
filteredField := rtrim(filteredField,'" ');
filteredField := rtrim(filteredField,',');
Return filteredField;
 END;
$BODY$
LANGUAGE plpgsql VOLATILE
COST 100;
-------Inner function end------


CREATE OR REPLACE FUNCTION get_artifact_multiselect_value(artifact_xml xml, fie
ld_key integer,selectedfields text[],condition character varying)
  RETURNS text as
  $BODY$
  DECLARE
   fieldValues text;
   loop1 integer default 1;

   loop2 integer default 1;
  arrayLength1 integer default 1;
  arrayLength2 integer default 1;
  arrayLength3 integer default 1;
  multiFieldValues text default '';
  multifield text DEFAULT '';
  returnvalues text DEFAULT '';
    i integer default 1;
    j integer default 1;
     v_array1 text[];
     v_array2 text[];
     v_array3 text[];
     results text Default '' ;
     filteredField text Default '';
     count integer default 0;
   BEGIN
       Select cast((xpath('/fields/field/@val',artifact_xml))[array_search(field_
key::text,(xpath('/fields/field/@key',artifact_xml)::text[]))]
       as varchar(9055)) into fieldValues;
       Select array_length(regexp_split_to_array(fieldValues, ','),1) into arra
yLength1;
       Select regexp_split_to_array(fieldValues, ',') into v_array1;
  IF arrayLength1 IS NOT NULL THEN
       FOR  loop1 in 1..arrayLength1 LOOP
       select value into multiFieldValues from  flex_field_value_dimension where
 flex_field_value_key=v_array1[loop1]::integer;
       multifield := concat(multifield,',',multiFieldValues);
       EXIT WHEN loop1 > arrayLength1;
       multifield := rtrim(multifield,'" ');
       multifield := ltrim(multifield,',');
       END LOOP;
       Select array_length(regexp_split_to_array(multifield, ','),1) into array
Length2;
       Select regexp_split_to_array(multifield, ',') into v_array2;
       Select array_length(regexp_split_to_array(array_to_string(selectedfields
,','), ','),1) into arrayLength3;
       Select regexp_split_to_array(array_to_string(selectedfields,','), ',') i
nto v_array3;

             IF (UPPER(condition)=UPPER('ALL')) THEN
                  IF selectedFields = Array['ALL'] THEN
                       results:=multifield;
                  ELSE
          IF (v_array2 is not  null AND v_array3 is not null) THEN
               FOR i in 1..arrayLength3 LOOP
                FOR j in 1..arrayLength2 LOOP
                     IF (v_array3[i]= v_array2[j]) THEN
                   filteredField := concat(v_array3[i],',',filteredField);

                   count=count+1;
                   EXIT;
                     END IF;

               END LOOP;
```

```
                 END LOOP;
          ELSE
             results = false;

                END IF;
                IF (count=arrayLength3) THEN
                results=filteredField;
                ELSE
                results='';
                END IF;
             END IF;
           ELSE

            select get_artifact_multiselect_values_any(artifact_xml,field_key,s
electedfields) into returnvalues;
                IF returnvalues IS NOT NULL THEN
                    results=returnvalues;
                ELSE
                     results='';
                END IF;

           END IF;
  END IF;
  results := rtrim(results,'" ');
  results := rtrim(results,',');
 RETURN(results);
 END;
$BODY$
LANGUAGE plpgsql VOLATILE
COST 100;
```

Function for Multi-user Flex Fields

Input Arguments

- Flex field XML (derived automatically by joining the `artifact_transaction_fact` (or) `artifact_daily_snapshot_fact` and `artifact_flex_field` tables).
- Flex field key (derived automatically from `flex_field_dimension` table and other tables).
- Specific value(s) stored in the field or the keyword ‘ALL’ for retrieving all values stored in multi-select flex field.
- Logical conditional operator. Possible values are ‘ALL’ or ‘ANY’. The argument value of ‘ALL’ performs a search equivalent to the ‘AND’ condition and the value of ‘ANY’ performs a search equivalent to the ‘OR’ condition.

Output Arguments

Values selected or stored in multi-user flex field

```
     -----First Execute Inner Function------

 ------Inner function start---
CREATE OR REPLACE FUNCTION get_artifact_user_values_any(artifact_xml xml, field
_key integer, selectedfields text[])
  RETURNS text AS
$BODY$
DECLARE
fieldValues text;
multiFieldValues text default '';
splitfield text[];
splitfield2 text[];
loop1 integer DEFAULT 1;
 loop2 integer DEFAULT 1;
arrayLength integer DEFAULT 1;
 arrayLength2 integer DEFAULT 1;
multifield text DEFAULT '';
filteredField text Default '';
BEGIN
 Select cast((xpath('/fields/field/@val',artifact_xml))
[array_search(field_key::text,(xpath('/fields/field/@key',artifact_xml)::text[]))
]
 as varchar(9055)) into fieldValues;
Select array_length(regexp_split_to_array(fieldValues, ','),1) into arrayLength
;
Select regexp_split_to_array(fieldValues, ',') into splitfield;
IF arrayLength IS NOT NULL THEN
   FOR loop1 IN 1..arrayLength LOOP
      Select full_name into multiFieldValues from user_dimension where user_ke
y = splitfield[loop1]::integer;
      multifield := concat(multifield,',',multiFieldValues);
      EXIT WHEN loop1 > arrayLength;
   END LOOP;
      multifield := rtrim(multifield,'" ');
      multifield := ltrim(multifield,',');

     Select array_length(regexp_split_to_array(array_to_string(selectedFields,
','), ','),1) into arrayLength2;
     Select regexp_split_to_array(array_to_string(selectedFields,','), ',') in
to splitfield2;
      IF selectedFields = Array['ALL'] THEN
      filteredField:=multifield;
      Return filteredField;
      ELSE
          FOR loop2 IN 1..arrayLength2 LOOP
          IF   multifield ~ splitfield2[loop2] THEN
        filteredField := concat(splitfield2[loop2],',',filteredField);
          END IF;
          EXIT WHEN loop2 > arrayLength2;
          END LOOP;
      END IF;
END IF;
filteredField := rtrim(filteredField,'" ');
filteredField := rtrim(filteredField,',');
Return filteredField;
 END;
$BODY$
  LANGUAGE plpgsql VOLATILE
  COST 100;

 ------Inner Function End-----

create or replace FUNCTION get_artifact_user_value(artifact_xml xml, field_key
integer,selectedfields text[],condition character varying)
   RETURNS text as
  $BODY$
  DECLARE
   fieldValues text;
   loop1 integer default 1;

   loop2 integer default 1;
  arrayLength1 integer default 1;
  arrayLength2 integer default 1;
  arrayLength3 integer default 1;
  multiFieldValues text default '';
  multifield text DEFAULT '';
  returnvalues text DEFAULT '';
    i integer default 1;
    j integer default 1;
     v_array1 text[];
     v_array2 text[];
     v_array3 text[];
     results text Default '' ;
     filteredField text Default '';
     count integer default 0;
   BEGIN
       Select cast((xpath('/fields/field/@val',artifact_xml))[array_search(field_
key::text,(xpath('/fields/field/@key',artifact_xml)::text[]))]
       as varchar(9055)) into fieldValues;
       Select array_length(regexp_split_to_array(fieldValues, ','),1) into arra
yLength1;
       Select regexp_split_to_array(fieldValues, ',') into v_array1;
  IF arrayLength1 IS NOT NULL THEN
       FOR  loop1 in 1..arrayLength1 LOOP
       select full_name into multiFieldValues from  user_dimension where user_
key=v_array1[loop1]::integer;
       multifield := concat(multifield,',',multiFieldValues);
       EXIT WHEN loop1 > arrayLength1;
       multifield := rtrim(multifield,'" ');
       multifield := ltrim(multifield,',');
       END LOOP;
       Select array_length(regexp_split_to_array(multifield, ','),1) into array
Length2;
       Select regexp_split_to_array(multifield, ',') into v_array2;
       Select array_length(regexp_split_to_array(array_to_string(selectedfields
,','), ','),1) into arrayLength3;
       Select regexp_split_to_array(array_to_string(selectedfields,','), ',') i
nto v_array3;

             IF (UPPER(condition)=UPPER('ALL')) THEN
                  IF selectedFields = Array['ALL'] THEN
                       results:=multifield;
                  ELSE
          IF (v_array2 is not  null AND v_array3 is not null) THEN
               FOR i in 1..arrayLength3 LOOP
                FOR j in 1..arrayLength2 LOOP
                     IF (v_array3[i]= v_array2[j]) THEN
                   filteredField := concat(v_array3[i],',',filteredField);

                   count=count+1;
                   EXIT;
                     END IF;

               END LOOP;
                 END LOOP;
          ELSE
             results = false;

                END IF;
                IF (count=arrayLength3) THEN
                results=filteredField;
                ELSE
                results='';
                END IF;
             END IF;
           ELSE

            select get_artifact_user_values_any(artifact_xml,field_key,selectedf
ields) into returnvalues;
                IF returnvalues IS NOT NULL THEN
                    results=returnvalues;
                ELSE
                     results='';
                END IF;

           END IF;
  END IF;
  results := rtrim(results,'" ');
  results := rtrim(results,',');
 RETURN(results);
 END;
$BODY$
LANGUAGE plpgsql VOLATILE
COST 100;
```

Sample Use Cases and Queries

> **NOTE:** Flex field Name, Value, Tracker title and so on that are used in the filtering conditions are case-sensitive. See Case 7 below for building a case-insensitive search using upper() or lower()functions.

Use Case 1: Filter Date and Text Flex Fields

Suppose you want to retrieve data based on the following assumptions:

- Date flex field named ‘CreatedDate’ that has a value of ‘2015-09-02 00:00:00’.
- Text flex field named ‘Text102’ with part of its value containing the pattern ‘soft’.
- Tracker title is ‘Tracker101’.

> **NOTE:** You can change the field name, tracker title and values based on the data in your system. The date passed as input should be of the format `YYYY-MM-DD HH24:MI:SS`.

Sample Query

```
select
  t.Date as Date,
  t.Project as Project,
  t.Tracker as Tracker,
  t.Planing_folder as Planing_folder,
  'P'||t.Priority as Priority,
  count(distinct t.artifact_key)
from

(            select     b.date_of_trans as Date,
        e.title as Project,c.title as Tracker,
        d.title as Planing_folder,
        'P'||a.priority as Priority,
        get_artifact_text_value(ff.flex_fields,ffd.flex_field_key) as TextFlexFiel
d,
        a.artifact_key
from
        artifact_transaction_fact a,
        date_dimension b,
        artifact_flex_fields ff,
        tracker_dimension c,
        pf_dimension d,
        project_dimension e ,
        flex_field_dimension ffd
where
        a.trans_date_key=b.date_key
        and ff.artifact_flex_fields_key = a.flex_fields_key
        and a.tracker_key=c.tracker_key
        and a.pf_key=d.pf_key
        and a.project_key=e.project_key
        and a.tracker_key=ffd.tracker_key
        and e.title='Project SCD-2 Test'
        and ffd.name in ('Text102')
        and c.title in ('Tracker101')
        and get_artifact_text_value(ff.flex_fields,ffd.flex_field_key) like 'soft%
'

union

    select    b.date_of_trans as Date,
        e.title as Project,c.title as Tracker,
        d.title as Planing_folder,
        'P'||a.priority as Priority,
        regexp_split_to_table(get_artifact_date_value(ff.flex_fields, ffd.flex_fie
ld_key),E',') as DateFlexField,
        a.artifact_key

from
        artifact_transaction_fact a,
        date_dimension b,
        artifact_flex_fields ff,
        tracker_dimension c,
        pf_dimension d,
        project_dimension e ,
        flex_field_dimension ffd
where
        a.trans_date_key=b.date_key
        and ff.artifact_flex_fields_key = a.flex_fields_key
        and a.tracker_key=c.tracker_key
        and a.pf_key=d.pf_key
        and a.project_key=e.project_key
        and a.tracker_key=ffd.tracker_key
        and e.title='Project SCD-2 Test'
        and ffd.name in ('CreatedDate')
        and c.title in ('Tracker101')
        and  get_artifact_date_value(ff.flex_fields, ffd.flex_field_key)='2015-09-
02 00:00:00') as t
group by 1,2,3,4,5
order by 1,2,3,4,5;
```

Use Case 2: Filter Single-select Flex Fields

Suppose you want to retrieve data based on the following assumptions:

- Flex field name: 'FSS1'
- Value selected or stored: 'S1'

- Tracker title: 'Tracker2'

> **NOTE:** You can change the field name, tracker title and values based on the data in your system.

Sample Query

```
select
    b.date_of_trans as Date,
    pd.title as Project,
    td.title as Tracker,
    d.title as Planning_folder,
    'P'||a.priority as Priority ,
    get_artifact_select_value(aff.flex_fields,ffd.flex_field_key) as SingleSelectF
lexField,
    count(distinct a.artifact_key) TotalArtifacts
from
    artifact_transaction_fact a,
    date_dimension b,
    artifact_flex_fields aff,
    project_dimension pd ,
    tracker_dimension td,
    pf_dimension d,
    flex_field_dimension ffd
where
    a.trans_date_key=b.date_key
    and a.flex_fields_key=aff.artifact_flex_fields_key
    and a.tracker_key=td.tracker_key
    and a.project_key=pd.project_key
    and td.tracker_key=ffd.tracker_key
    and a.pf_key=d.pf_key
    and pd.title='ProjectTestFunctionSCD-2'
    and td.title in ('Tracker2')
    and ffd.name='FSS1'
    and get_artifact_select_value(aff.flex_fields,ffd.flex_field_key)='S1'
    --and date(a.effective_from)<=date(now()) and date(a.effective_till)>'2012
-04-01'
group by 1,2,3,4,5,6
order by 1,2,3,4,5,6;
```

Use Case 3: Filter Text Flex Fields

Suppose you want to retrieve data based on the following assumptions:

- Flex field name: ‘Text101’
- Value contains: ‘hello’
- Tracker name: ‘Tracker101’

> **NOTE:** You can change the field name, tracker title and values based on the data in your system.

Sample Query

```
select
     b.date_of_trans as Date,
    e.title as Project,
    c.title as Tracker,
    d.title as Planing_folder,
    'P'||a.priority as Priority,
    get_artifact_text_value(ff.flex_fields,ffd.flex_field_key) as TextFlexField,
    count(distinct a.artifact_key) Artifacts
from
    artifact_transaction_fact a,
    date_dimension b,
    artifact_flex_fields ff,
    tracker_dimension c,
    pf_dimension d,
    project_dimension e ,
    flex_field_dimension ffd
where
    a.trans_date_key=b.date_key
    and ff.artifact_flex_fields_key = a.flex_fields_key
    and a.tracker_key=c.tracker_key
    and a.pf_key=d.pf_key
    and a.project_key=e.project_key
    and a.tracker_key=ffd.tracker_key
    and e.title='Project SCD-2 Test'
    and ffd.name = 'Text101'
    and c.title in ('Tracker101')
    and get_artifact_text_value(ff.flex_fields,ffd.flex_field_key) like 'hello%'
    --and date(a.effective_from)<=date(now()) and date(a.effective_till)>'2012
-04-01'
group by 1,2,3,4,5,6
order by 1,2,3,4,5,6;
```

Use Case 4: Filter Date Flex Fields

Suppose you want to retrieve data based on the following assumptions:

- Flex field name: ‘CreatedDate’
- Value: ‘2015-09-02 00:00:00’
- Tracker name: ‘Tracker101’

> **NOTE:** You can change the field name, tracker title and values based on the data in your system.

Sample Query

```
select
    b.date_of_trans as Date,
    e.title as Project,
    c.title as Tracker,
    d.title as Planing_folder,
    'P'||a.priority as Priority,
    regexp_split_to_table(get_artifact_date_value(ff.flex_fields, ffd.flex_field_k
ey),E',') as DateFlexField,
    count(distinct a.artifact_key)

from
    artifact_transaction_fact a,
    date_dimension b,
    artifact_flex_fields ff,
    tracker_dimension c,
    pf_dimension d,
    project_dimension e ,
    flex_field_dimension ffd
where
        a.trans_date_key=b.date_key
        and ff.artifact_flex_fields_key = a.flex_fields_key
        and a.tracker_key=c.tracker_key
        and a.pf_key=d.pf_key
        and a.project_key=e.project_key
        and a.tracker_key=ffd.tracker_key
        and e.title='Project SCD-2 Test'
        and ffd.name in ('CreatedDate')
        and c.title in ('Tracker101')
        and  get_artifact_date_value(ff.flex_fields, ffd.flex_field_key)='2015-09-
02 00:00:00'

group by  1,2,3,4,5,6
order by 1,2,3,4,5,6;
```

Use Case 5: Multi-select Flex Fields

Suppose you want to retrieve data based on the following assumptions:

- Multi-select flex field name: ‘Country’
- Value: ‘Russia,India’
- Tracker name: ‘TrackerN’

- Conditional parameter: ‘ALL’

✓ You can change the field name, tracker title and values based on the data in your system.

✓ Flex field name, value and tracker title that are used in the SQL filter conditions are case sensitive.

✓ If you want to select all the values in User flex field, then pass ‘ALL’ as the conditional parameter.

✓ If you want to select any value, then pass ‘ANY’ as the conditional parameter.

Sample Query

```
select b.date_of_trans as Date, e.title as Project,c.title as Tracker,
d.title as Planing_folder,'P'||a.priority as Priority,
 get_artifact_multiselect_value(ff.flex_fields,ffd.flex_field_key,'{Russia,India}'
,'ALL') as MultiselectFlexField,
count(distinct a.artifact_key) Artifacts
--,a.artifact_key
from
    artifact_transaction_fact a,
    date_dimension b,
    artifact_flex_fields ff,
    tracker_dimension c,
    pf_dimension d,
    project_dimension e ,
    flex_field_dimension ffd
where
    a.trans_date_key=b.date_key
    and ff.artifact_flex_fields_key = a.flex_fields_key
    and a.tracker_key=c.tracker_key
    and a.pf_key=d.pf_key
    and a.project_key=e.project_key
    and a.tracker_key=ffd.tracker_key
    --and e.title='TestMultiFunction'
    and ffd.name = 'Country'
    and c.title in ('TrackerN')
    and get_artifact_multiselect_value(ff.flex_fields,ffd.flex_field_key,'{Russia,
India}','ALL')!=''
    --and date(a.effective_from)<=date(now()) and date(a.effective_till)>'2012
-04-01'
group by 1,2,3,4,5,6
order by 1,2,3,4,5,6;
```

Use Case 6: Filter User Flex Fields

Suppose you want to retrieve data based on the following assumptions:

- User flex field name: ‘Select User’
- Value: ‘user1,user2’
- Tracker name: ‘TrackerN’

- Conditional parameter: ‘ALL’

✓ You can change the field name, tracker title and values based on the data in your system.

Flex field name, value and tracker title that are used in the SQL filter conditions are case sensitive.

✓ If you want to select all the values in User flex field, then pass ‘ALL’ as the conditional parameter.

✓ If you want to select any value, then pass ‘ANY’ as the conditional parameter.

Sample Query

```
select b.date_of_trans as Date, e.title as Project,c.title as Tracker,
d.title as Planing_folder,'P'||a.priority as Priority,
 get_artifact_user_value(ff.flex_fields,ffd.flex_field_key,'{user1,user2}','ALL')
as UserFlexField,
count(distinct a.artifact_key) Artifacts
--,a.artifact_key
from
    artifact_transaction_fact a,
    date_dimension b,
    artifact_flex_fields ff,
    tracker_dimension c,
    pf_dimension d,
    project_dimension e ,
    flex_field_dimension ffd
```

```
where
    a.trans_date_key=b.date_key
    and ff.artifact_flex_fields_key = a.flex_fields_key
    and a.tracker_key=c.tracker_key
    and a.pf_key=d.pf_key
    and a.project_key=e.project_key
    and a.tracker_key=ffd.tracker_key
    --and e.title='TestMultiFunction'
    and ffd.name = 'Select User'
    and c.title in ('TrackerN')
    and get_artifact_user_value(ff.flex_fields,ffd.flex_field_key,'{user1,user2}',
'ALL')!=''
    --and date(a.effective_from)<=date(now()) and date(a.effective_till)>'2012
-04-01'
group by 1,2,3,4,5,6
order by 1,2,3,4,5,6;
```

Use Case 7: Filter Text Flex fields: Case-insensitive Search

Suppose you want to retrieve data based on the following assumptions:

- Flex field name: ‘Text102’
- Value contains: ‘soft’

- Tracker name: ‘Tracker101’

> **NOTE:** You can change the field name, tracker title and values based on the data in your system. The upper() function can be replaced with the lower() function appropriately for case-insensitive search.

Sample Query

```
select
            b.date_of_trans as Date,
        e.title as Project,
        c.title as Tracker,
        d.title as Planing_folder,
        'P'||a.priority as Priority,
        get_artifact_text_value(ff.flex_fields,ffd.flex_field_key) as TextFlexFiel
d,
        count(distinct a.artifact_key) as TotalCounts
from
        artifact_transaction_fact a,
        date_dimension b,
        artifact_flex_fields ff,
        tracker_dimension c,
        pf_dimension d,
        project_dimension e ,
        flex_field_dimension ffd
where
        a.trans_date_key=b.date_key
        and ff.artifact_flex_fields_key = a.flex_fields_key
        and a.tracker_key=c.tracker_key
        and a.pf_key=d.pf_key
        and a.project_key=e.project_key
        and a.tracker_key=ffd.tracker_key
        and UPPER(e.title)=UPPER('Project SCD-2 Test')
        and UPPER(ffd.name)=UPPER('Text102')
        and UPPER(c.title) = UPPER('Tracker101')
        and UPPER(get_artifact_text_value(ff.flex_fields,ffd.flex_field_key)) like
 UPPER('soft%')

group by 1,2,3,4,5,6
order by 1,2,3,4,5,6;
```

Query Oracle Datamart Based on Flex Fields

Create query scripts to query and extract the required information from the Oracle datamart. Make sure these scripts are available to any user (including users with read-only permission) who wants to execute these scripts. This topic lists the functions and sample queries for a few specific use cases.

Important:

✓ Log into TeamForge as a Reporting user and run these queries. Note that this is a one-time process.

✓ Use the following flex field functions by joining the `flex_field_dimension`, `artifact_flex_field` and `artifact_transaction_fact` (or) `artifact_daily_snapshot_fact` tables as it depends on the XML data and flex field key generated by the platform.

Function for Date Flex Fields

Input Arguments

- Flex field XML (derived automatically by joining the `artifact_transaction_fact` (or) `artifact_daily_snapshot_fact` and `artifact_flex_field` tables).
- Flex field key (derived automatically from `flex_field_dimension` table and other tables).

Output Arguments

Date Flex Field Values

```
create or replace FUNCTION get_artifact_date_value( artifact_xml IN XMLTYPE, fi
eld_key IN INTEGER )
   RETURN varchar2
   IS
 flex varchar2(4000);
   BEGIN
      SELECT x.val INTO flex FROM dual ,
      XMLTABLE ('/fields/field[@key=$keyalias]'
          PASSING artifact_xml,field_key as "keyalias"
          COLUMNS val VARCHAR2(4000) PATH '@val') x;
   RETURN(flex);
   END;
```

Function for Text Flex Fields

Input Arguments

- Flex field XML (derived automatically by joining the `artifact_transaction_fact` (or) `artifact_daily_snapshot_fact` and `artifact_flex_field` tables).
- Flex field key (derived automatically from `flex_field_dimension` table and other tables).

Output Arguments

Text Flex Field Values

```
create or replace FUNCTION replacen( artifact_xml1 IN XMLTYPE)
   RETURN varchar2
   IS
   replaceval  varchar2(3000);
   BEGIN
   SELECT REPLACE((REPLACE(cast(artifact_xml1 as varchar2(3000)),'','')),'',''
) into replaceval  from dual;
   RETURN(replaceval);
   END;
create or replace FUNCTION get_artifact_text_value( artifact_xml IN XMLTYPE, fi
eld_key IN integer )
   RETURN varchar2
   IS flex varchar2(3200);
   fields_xml XMLTYPE;
   BEGIN
   SELECT XMLTYPE(replacen(artifact_xml)) INTO fields_xml FROM dual;
      SELECT x.val INTO flex FROM dual ,
      XMLTABLE ('/fields/field[@key=$keyalias]'
          PASSING fields_xml,field_key as "keyalias"
          COLUMNS val VARCHAR2(400) PATH '@val' ) x;
   RETURN(flex);
   END;
```

Function for Single-select Flex Fields

Input Arguments

- Flex field XML (derived automatically by joining the `artifact_transaction_fact` (or) `artifact_daily_snapshot_fact` and `artifact_flex_field` tables).
- Flex field key (derived automatically from `flex_field_dimension` table and other tables).

Output Arguments

Values selected or stored in single-select flex field

```
create or replace FUNCTION get_artifact_select_value( artifact_xml IN XMLTYPE,
 field_key IN INTEGER)
   RETURN VARCHAR2
   IS
    singleSelectValue VARCHAR2(100);
    flex INTEGER;
   BEGIN
      SELECT x.val INTO flex FROM dual ,
      XMLTABLE ('/fields/field[@key=$keyalias]'
          PASSING artifact_xml,field_key as "keyalias"
          COLUMNS val VARCHAR2(30) PATH '@val') x;
   SELECT  z.value INTO singleSelectValue from  FLEX_FIELD_VALUE_DIMENSION z
where z.FLEX_FIELD_VALUE_KEY=flex;
   RETURN(singleSelectValue);
   END;
```

Function for Multi-select Flex Fields

Input Arguments

- Flex field XML (derived automatically by joining the `artifact_transaction_fact` (or) `artifact_daily_snapshot_fact` and `artifact_flex_field` tables).
- Flex field key (derived automatically from `flex_field_dimension` table and other tables).
- Specific value(s) stored in the field or the keyword ‘ALL’ for retrieving all values stored in multi-select flex field.
- Logical conditional operator. Possible values are ‘ALL’ or ‘ANY’. The argument value of ‘ALL’ performs a search equivalent to the ‘AND’ condition and the value of ‘ANY’ performs a search equivalent to the ‘OR’ condition.

Output Arguments

Values selected or stored in multi-select flex field

```
-----Inner Function----
    create or replace FUNCTION get_artifact_multiselect_any(artifact_xml IN XM
LTYPE, field_key IN INTEGER,selectedfields IN varchar2)
   RETURN varchar2
   IS
   flexs varchar2(300);
   arraylength integer default 1;
   loop1 integer default 1;
   loop2 integer default 1;
    i integer default 1;
    j integer default 1;
     v_array1 apex_application_global.vc_arr2;
     v_array2 apex_application_global.vc_arr2;
     v_array3 apex_application_global.vc_arr2;
     results varchar2(200) ;

   BEGIN
       SELECT x.val  INTO flexs FROM dual ,
        XMLTABLE ('/fields/field[@key=$keyalias]'
          PASSING artifact_xml,field_key as "keyalias"
          COLUMNS val VARCHAR2(4000) PATH '@val') x;
   v_array1 := apex_util.string_to_table(flexs,',');

   FOR  loop1 in 1..v_array1.count LOOP
     select value into v_array2(loop1) from  flex_field_value_dimension where fle
x_field_value_key=v_array1(loop1);
   END LOOP;

 v_array3 := apex_util.string_to_table(selectedfields,',');


 FOR   loop2 in 1..1 LOOP
                IF (v_array3(loop2)='ALL') THEN
                                 FOR i in 1..v_array2.count LOOP
                                       IF LENGTH(results)!=0 THEN
                                                     results:=results ||','||v
_array2(i);
                                                 ELSE
                                                      results:=v_array2(i);
                                                  END IF;

                             END LOOP;
                     EXIT;

            ELSE
```

```
                                     FOR  i in 1..v_array2.count LOOP
                                          FOR  j in 1..v_array3.count LOOP
                                               IF (v_array3(j)=v_array2(i)) TH
EN
                                                    IF LENGTH(results)!=0 THEN

                                                        results:=results ||','
||v_array3(j);
                                                        ELSE
                                                        results:=v_array3(j);
                                                    END IF;
                                                    EXIT;
                                                END IF;

                                          END LOOP;
                                     END LOOP;
                END IF;
  END LOOP;

 RETURN(results);
           END;
/

-----End Inner function-----

   create or replace FUNCTION get_artifact_multiselect_value(artifact_xml IN X
MLTYPE, field_key IN INTEGER,selectedfields IN varchar2,condition IN varchar2)
   RETURN varchar2
   IS
   flexs varchar2(300);
   loop1 integer default 1;
   loop2 integer default 1;
    i integer default 1;
    j integer default 1;
     v_array1 apex_application_global.vc_arr2;
     v_array2 apex_application_global.vc_arr2;
     v_array3 apex_application_global.vc_arr2;
     results varchar2(200) ;
     filteredField varchar2(200) ;
   valcount integer default 0;

   BEGIN
       SELECT x.val  INTO flexs FROM dual ,
        XMLTABLE ('/fields/field[@key=$keyalias]'
          PASSING artifact_xml,field_key as "keyalias"
          COLUMNS val VARCHAR2(4000) PATH '@val') x;
   v_array1 := apex_util.string_to_table(flexs,',');
   FOR  loop1 in 1..v_array1.count LOOP
     select value into v_array2(loop1) from  flex_field_value_dimension where fle
x_field_value_key=v_array1(loop1);
   END LOOP;

 v_array3 := apex_util.string_to_table(selectedfields,',');

 IF (UPPER(condition)=UPPER('ALL')) THEN
                      FOR   loop2 in 1..1 LOOP
                          IF (v_array3(loop2)='ALL') THEN
                                   FOR i in 1..v_array2.count LOOP
                                         IF LENGTH(results)!=0 THEN
                                           results:=results ||','||v_array2(i)
;
                                         ELSE
                                            results:=v_array2(i);
                                         END IF;

                                     END LOOP;
                                     filteredField:=results;
                                     EXIT;

                          ELSE


                                   FOR  i in 1..v_array3.count LOOP
                                      FOR  j in 1..v_array2.count LOOP
                                         IF (v_array3(i)=v_array2(j)) THEN
                                          IF LENGTH(results)!=0 THEN
                                            results:=results ||','||v_array3(j
);
                                            valcount:=valcount+1;
                                            ELSE
                                            results:=v_array3(j);
                                            valcount:=valcount+1;
                                          END IF;
                                          EXIT;
                                        END IF;

                                      END LOOP;
                                   END LOOP;
                                   IF (valcount=v_array3.count) THEN
                                   filteredField:=results;
                                   ELSE
                                   filteredField:='';
                                   END IF;
                          END IF;
                    END LOOP;
   ELSE
      select get_artifact_multiselect_any(artifact_xml,field_key,selectedfields)
 into results from dual;
        IF results IS NOT NULL THEN
          filteredField:=results;
        ELSE
          filteredField:='';
        END IF;
   END IF;
filteredField:=TRIM(TRAILING ',' FROM filteredField);

RETURN(filteredField);
           END;
```

Function for Multi-user Flex Fields

Input Arguments

- Flex field XML (derived automatically by joining the `artifact_transaction_fact` (or) `artifact_daily_snapshot_fact` and `artifact_flex_field` tables).
- Flex field key (derived automatically from `flex_field_dimension` table and other tables).
- Specific value(s) stored in the field or the keyword ‘ALL’ for retrieving all values stored in multi-select flex field.
- Logical conditional operator. Possible values are ‘ALL’ or ‘ANY’. The argument value of ‘ALL’ performs a search equivalent to the ‘AND’ condition and the value of ‘ANY’ performs a search equivalent to the ‘OR’ condition.

Output Arguments

Values selected or stored in multi-user flex field

```
------Inner Function-----
    create or replace FUNCTION get_artifact_user_any(artifact_xml IN XMLTYPE,
field_key IN integer,selectedfields IN varchar2)
   RETURN varchar2
   IS
   flexs varchar2(300);
   arraylength integer default 1;
   loop1 integer default 1;
   loop2 integer default 1;
    i integer default 1;
    j integer default 1;
     v_array1 apex_application_global.vc_arr2;
     v_array2 apex_application_global.vc_arr2;
     v_array3 apex_application_global.vc_arr2;
     results varchar2(200) ;

   BEGIN
       SELECT x.val  INTO flexs FROM dual ,
        XMLTABLE ('/fields/field[@key=$keyalias]'
          PASSING artifact_xml,field_key as "keyalias"
          COLUMNS val VARCHAR2(4000) PATH '@val') x;
   v_array1 := apex_util.string_to_table(flexs,',');

   FOR  loop1 in 1..v_array1.count LOOP
     select full_name into v_array2(loop1) from  user_dimension where user_key
=v_array1(loop1);
   END LOOP;

 v_array3 := apex_util.string_to_table(selectedfields,',');


 FOR   loop2 in 1..v_array3.count LOOP
                IF (v_array3(loop2)='ALL') THEN
                                 FOR i in 1..v_array2.count LOOP
                                       IF LENGTH(results)!=0 THEN
                                                     results:=results ||','||v
_array2(i);
                                                 ELSE
                                                      results:=v_array2(i);
                                                  END IF;

                             END LOOP;
                     EXIT;

            ELSE


                                     FOR  i in 1..v_array2.count LOOP
                                          FOR  j in 1..v_array3.count LOOP
                                               IF (v_array3(j)=v_array2(i)) TH
EN
                                                    IF LENGTH(results)!=0 THEN

                                                        results:=results ||','
||v_array3(j);

                                                        ELSE
                                                        results:=v_array3(j);
                                                    END IF;
                                                    EXIT;
                                                END IF;

                                          END LOOP;
                                     END LOOP;
                END IF;
  END LOOP;

 RETURN(results);
           END;

------End Inner Function----

create or replace FUNCTION get_artifact_user_value(artifact_xml IN XMLTYPE, fie
ld_key IN INTEGER,selectedfields IN varchar2,condition IN varchar2)
   RETURN varchar2
   IS
   flexs varchar2(300);
   loop1 integer default 1;
   loop2 integer default 1;
    i integer default 1;
    j integer default 1;
     v_array1 apex_application_global.vc_arr2;
     v_array2 apex_application_global.vc_arr2;
     v_array3 apex_application_global.vc_arr2;
     results varchar2(200) ;
     filteredField varchar2(200) ;
   valcount integer default 0;

   BEGIN
       SELECT x.val  INTO flexs FROM dual ,
        XMLTABLE ('/fields/field[@key=$keyalias]'
          PASSING artifact_xml,field_key as "keyalias"
          COLUMNS val VARCHAR2(4000) PATH '@val') x;
   v_array1 := apex_util.string_to_table(flexs,',');

   FOR  loop1 in 1..v_array1.count LOOP
     select full_name into v_array2(loop1) from  user_dimension where user_key
=v_array1(loop1);
   END LOOP;

 v_array3 := apex_util.string_to_table(selectedfields,',');

 IF (UPPER(condition)=UPPER('ALL')) THEN
                      FOR   loop2 in 1..1 LOOP
                          IF (v_array3(loop2)='ALL') THEN
                                   FOR i in 1..v_array2.count LOOP
                                         IF LENGTH(results)!=0 THEN
```

```
                                           results:=results ||','||v_array2(i)
;
                                         ELSE
                                            results:=v_array2(i);
                                         END IF;

                                     END LOOP;
                                     filteredField:=results;
                                     EXIT;

                          ELSE


                                   FOR  i in 1..v_array3.count LOOP
                                      FOR  j in 1..v_array2.count LOOP
                                         IF (v_array3(i)=v_array2(j)) THEN
                                          IF LENGTH(results)!=0 THEN
                                            results:=results ||','||v_array3(j
);

                                            valcount:=valcount+1;
                                            ELSE
                                            results:=v_array3(j);
                                            valcount:=valcount+1;
                                          END IF;
                                          EXIT;
                                        END IF;

                                      END LOOP;
                                   END LOOP;
                                   IF (valcount=v_array3.count) THEN
                                   filteredField:=results;
                                   ELSE
                                   filteredField:='';
                                   END IF;
                          END IF;
                    END LOOP;




   ELSE
      select get_artifact_user_any(artifact_xml,field_key,selectedfields) into r
esults from dual;
        IF results IS NOT NULL THEN
          filteredField:=results;
        ELSE
          filteredField:='';
        END IF;
   END IF;
```

```
filteredField:=TRIM(TRAILING ',' FROM filteredField);

RETURN(filteredField);
           END;
```

Sample Use Cases and Queries

> **NOTE:** Flex field Name, Value, Tracker title and so on that are used in the filtering conditions are case-sensitive. See Case 7 below for building a case-insensitive search using upper() or lower() functions.

Use Case 1: Filter Date and Text Flex Fields

Suppose you want to retrieve data based on the following assumptions:

- Date flex field named ‘CreatedDate’ that has a value of ‘2015-07-07 00:00:00’
- Text flex field named ‘TEXT’ with part of its value containing the pattern ‘cre’
- Tracker title is ‘TestDateTracker’

> **NOTE:** You can change the field name, tracker title and values based on the data in your system. The date passed as input should be of the format `YYYY-MM-DD HH24:MI:SS`.

Sample Query

```
SELECT t.date_of_trans  "Date",
       t.Project   ,
       t.Tracker   ,
       t.PlaningFolder  ,
       t.Priority,
       count(distinct t.artifact_key) "TotalArtifacts"  FROM (SELECT
b.date_of_trans,
                        e.title  AS Project,
     c.title AS Tracker,
                        d.title AS PlaningFolder,
                        'P'||a.priority AS Priority,
                        a.artifact_key
 FROM  artifact_transaction_fact a inner join date_dimension b on (a.trans_dat
e_key=b.date_key)
                      inner join  artifact_flex_fields ff  on (ff.artifact_flex_fi
elds_key = a.flex_fields_key)
                      inner join  tracker_dimension c on (a.tracker_key=c.trac
ker_key)
            inner join  pf_dimension d on  (a.pf_key=d.pf_key)
            inner join  project_dimension e on  (a.project_key=e.project_key)
```

```
                  inner join flex_field_dimension ffd on (a.tracker_key=ffd.trac
ker_key)
                  and ffd.name='Date'
                  and c.title='TestDateTracker'
            and get_artifact_date_value(ff.flex_fields,ffd.flex_field_key)='2015-0
8-12 00:00:00'

UNION


 SELECT            b.date_of_trans,
                           e.title  AS Project,
        c.title AS Tracker,
                           d.title AS PlaningFolder,
                           'P'||a.priority,
                            a.artifact_key
 FROM  artifact_transaction_fact a inner join date_dimension b on (a.trans_dat
e_key=b.date_key)
                       inner join  artifact_flex_fields ff  on (ff.artifact_flex_
fields_key = a.flex_fields_key)
                       inner join  tracker_dimension c on (a.tracker_key=c.tra
cker_key)
             inner join  pf_dimension d on  (a.pf_key=d.pf_key)
                   inner join  project_dimension e on  (a.project_key=e.projec
t_key)
                   inner join  flex_field_dimension ffd on (a.tracker_key=ffd.tr
acker_key)
                   and ffd.name='TEXT'
 and c.title='TrackerTest'
            and get_artifact_text_value(ff.flex_fields,ffd.flex_field_key) LIKE 'c
re%'
            and d.id='plan1004' ) t

GROUP BY  t.date_of_trans,t.Project,t.Tracker,t.PlaningFolder,t.Priority
ORDER BY  t.date_of_trans,t.Project,t.Tracker,t.PlaningFolder,t.Priority
```

Use Case 2: Filter Single-select Flex Fields

Suppose you want to retrieve data based on the following assumptions:

- Flex field name: ‘FSS1’
- Value selected or stored: ‘S1’

- Tracker title: ‘FTracker’

> **NOTE:** You can change the field name, tracker title and values based on the data in your system.

Sample Query

```
SELECT    b.date_of_trans  "Date",
                   e.title "Project",
                   c.title  "Tracker",
                   d.title  "Planingfolder",
                   'P'||a.priority  "Priority",
                  count( distinct a.artifact_key) "TotalArtifacts"
 FROM  artifact_transaction_fact a inner join date_dimension b on (a.trans_dat
e_key=b.date_key)
                             inner join  artifact_flex_fields ff  on (ff.artifac
t_flex_fields_key = a.flex_fields_key)
                             inner join  tracker_dimension c on (a.tracker_key
=c.tracker_key)
         inner join  pf_dimension d on  (a.pf_key=d.pf_key)
        inner join  project_dimension e on  (a.project_key=e.project_key)
                    inner join flex_field_dimension ffd on (a.tracker_key=ffd.tr
acker_key)
                    and ffd.name='FSS1'
                    and c.title='FTracker' and get_artifact_select_value(ff.fle
x_fields,ffd.flex_field_key)='S1'
--where a.effective_from <= sysdate and a.effective_till > to_date('2012-04-01
', 'YYYY-MM-DD')
GROUP BY  b.date_of_trans,e.title,c.title,d.title,a.priority
ORDER BY  b.date_of_trans,e.title,c.title,d.title,a.priority
```

Use Case 3: Filter Text Flex Fields

Suppose you want to retrieve data based on the following assumptions:

- Flex field name: ‘TEXT’
- Value contains: ‘cre

- Tracker name: ‘TrackerTest’

> **NOTE:** You can change the field name, tracker title and values based on the data in your system.

Sample Query

```
SELECT           b.date_of_trans  "Date",
                          e.title "Project",
        c.title  "Tracker",
                          d.title  "Planingfolder",
                          'P'||a.priority  "Priority",
                          count( distinct a.artifact_key) "TotalArtifacts"
 FROM  artifact_transaction_fact a inner join date_dimension b on (a.trans_dat
e_key=b.date_key)
```

```
                       inner join  artifact_flex_fields ff  on (ff.artifact_flex_
fields_key = a.flex_fields_key)
                       inner join  tracker_dimension c on (a.tracker_key=c.tra
cker_key)
             inner join  pf_dimension d on  (a.pf_key=d.pf_key)
                   inner join  project_dimension e on  (a.project_key=e.projec
t_key)
                   inner join  flex_field_dimension ffd on (a.tracker_key=ffd.tr
acker_key)
                   and ffd.name='TEXT'
 and c.title='TrackerTest'
            and get_artifact_text_value(ff.flex_fields,ffd.flex_field_key) LIKE 'c
re%'
            and d.id='plan1004'
--where a.effective_from <= sysdate and a.effective_till > to_date('2012-04-01
', 'YYYY-MM-DD')
GROUP BY  b.date_of_trans,e.title,c.title,d.title,a.priority
ORDER BY  b.date_of_trans,e.title,c.title,d.title,a.priority
```

Use Case 4: Filter Date Flex Fields

Suppose you want to retrieve data based on the following assumptions:

- Flex field name: ‘Date’
- Value: ‘2015-08-12 00:00:00’
- Tracker name: ‘TestDateTracker’

> **NOTE:** You can change the field name, tracker title and values based on the data in your system.

Sample Query

```
SELECT          b.date_of_trans  "Date",
                        e.title "Project",
     c.title  "Tracker",
                        d.title  "Planingfolder",
                        'P'||a.priority  "Priority",
                        count( distinct a.artifact_key) "TotalArtifacts"
 FROM  artifact_transaction_fact a inner join date_dimension b on (a.trans_dat
e_key=b.date_key)
                      inner join  artifact_flex_fields ff  on (ff.artifact_flex_fi
elds_key = a.flex_fields_key)
                      inner join  tracker_dimension c on (a.tracker_key=c.trac
ker_key)
            inner join  pf_dimension d on  (a.pf_key=d.pf_key)
            inner join  project_dimension e on  (a.project_key=e.project_key)
                  inner join flex_field_dimension ffd on (a.tracker_key=ffd.trac
ker_key)
                  and ffd.name='Date'
                  and c.title='TestDateTracker'
            and get_artifact_date_value(ff.flex_fields,ffd.flex_field_key)='2015-0
8-12 00:00:00'
--where a.effective_from <= sysdate and a.effective_till > to_date('2012-04-01
', 'YYYY-MM-DD')
GROUP BY  b.date_of_trans,e.title,c.title,d.title,a.priority
ORDER BY  b.date_of_trans,e.title,c.title,d.title,a.priority
```

Use Case 5: Multi-select Flex Fields

Suppose you want to retrieve data based on the following assumptions:

- Multi-select flex field name: ‘Multiselect’
- Value: ‘M12,M11’
- Tracker name: ‘TrackerTest’

- Conditional parameter: ‘ALL’

✓ You can change the field name, tracker title and values based on the data in your system.

✓ Flex field name, value and tracker title that are used in the SQL filter conditions are case sensitive.

✓ If you want to select all the values in User flex field, then pass ‘ALL’ as the conditional parameter.

✓ If you want to select any value, then pass ‘ANY’ as the conditional parameter.

Sample Query

```
SELECT          b.date_of_trans  "Date",
                              e.title "Project",
                          c.title  "Tracker",
                              d.title  "Planingfolder",
                         'P'||a.priority  "Priority",
               count(distinct a.artifact_key)
 FROM        artifact_transaction_fact a   inner join date_dimension b on (a.tr
ans_date_key=b.date_key)
               inner join  artifact_flex_fields ff  on (ff.artifact_flex_fields_ke
y = a.flex_fields_key)
               inner join  tracker_dimension c on (a.tracker_key=c.tracker_key
)
           inner join  pf_dimension d on  (a.pf_key=d.pf_key)
           inner join  project_dimension e on  (a.project_key=e.project_key)
               inner join flex_field_dimension ffd on (a.tracker_key=ffd.tracker
_key)
```

```
               and ffd.name='Multiselect'
               and c.title='TrackerTest'
               and
get_artifact_multiselect_value(ff.flex_fields,ffd.flex_field_key,'M11,M12','ALL')
IS NOT NULL
GROUP BY  b.date_of_trans,e.title,c.title,d.title,a.priority
ORDER BY  b.date_of_trans,e.title,c.title,d.title,a.priority
```

Use Case 6: Filter User Flex Fields

Suppose you want to retrieve data based on the following assumptions:

- User flex field name: ‘Select User’
- Value: ‘user1,user2’
- Tracker name: ‘TrackerTestUser’

- Conditional parameter: ‘ALL’

  ✓ You can change the field name, tracker title and values based on the data in your system.

  ✓ Flex field name, value and tracker title that are used in the SQL filter conditions are case sensitive.

  ✓ If you want to select all the values in User flex field, then pass ‘ALL’ as the conditional parameter.

  ✓ If you want to select any value, then pass ‘ANY’ as the conditional parameter.

Sample Query

```
  SELECT
          b.date_of_trans  "Date",
          e.title "Project",
      c.title  "Tracker",
          d.title  "Planingfolder",
         'P'||a.priority  "Priority",
         count( distinct a.artifact_key) "TotalArtifacts"
 FROM  artifact_transaction_fact a inner join date_dimension b on (a.trans_dat
e_key=b.date_key)
        inner join  artifact_flex_fields ff  on (ff.artifact_flex_fields_key = a.fl
ex_fields_key)
        inner join  tracker_dimension c on (a.tracker_key=c.tracker_key)
    inner join  pf_dimension d on  (a.pf_key=d.pf_key)
    inner join  project_dimension e on  (a.project_key=e.project_key)
        inner join flex_field_dimension ffd on (a.tracker_key=ffd.tracker_key)
        and ffd.name='Select User'
        and c.title='TrackerTestUser'
        and get_artifact_user_value(ff.flex_fields,ffd.flex_field_key,'user1,user2
','ALL') IS NOT NULL
--where a.effective_from <= sysdate and a.effective_till > to_date('2012-04-01
', 'YYYY-MM-DD')
GROUP BY  b.date_of_trans,e.title,c.title,d.title,a.priority
ORDER BY  b.date_of_trans,e.title,c.title,d.title,a.priority
```

Use Case 7: Filter Text flex fields: Case-insensitive Search

Suppose you want to retrieve data based on the following assumptions:

- Flex field name: ‘TEXT’
- Value contains: ‘cre’

- Tracker name: ‘TestDateTracker’

> **NOTE:** You can change the field name, tracker title and values based on the data in your system. The upper() function can be replaced with the lower() function appropriately for case-insensitive search.

Sample Query

```
 SELECT
           b.date_of_trans  "Date",
           e.title "Project",
      c.title  "Tracker",
           d.title  "Planingfolder",
          'P'||a.priority  "Priority",
           count( distinct a.artifact_key) "TotalArtifacts"
 FROM  artifact_transaction_fact a inner join date_dimension b on (a.trans_dat
e_key=b.date_key)
           inner join  artifact_flex_fields ff  on (ff.artifact_flex_fields_key =
 a.flex_fields_key)
           inner join  tracker_dimension c on (a.tracker_key=c.tracker_key)
       inner join  pf_dimension d on  (a.pf_key=d.pf_key)
       inner join  project_dimension e on  (a.project_key=e.project_key)
           inner join  flex_field_dimension ffd on (a.tracker_key=ffd.tracker_k
ey)
           and UPPER(ffd.name)=UPPER('TEXT')
           and upper(c.title)=upper('TestDateTracker')
       and upper(get_artifact_text_value(ff.flex_fields,ffd.flex_field_key)) like
 upper('cre%')
--where a.effective_from <= sysdate and a.effective_till > to_date('2012-04-01
', 'YYYY-MM-DD')
GROUP BY  b.date_of_trans,e.title,c.title,d.title,a.priority
ORDER BY  b.date_of_trans,e.title,c.title,d.title,a.priority
```

TrackerInitialJob - Parallel Processing

A parallel processing feature has been introduced in TeamForge 8.1 Patch 1 to improve the performance of the tracker initial load job.

A performance observation was conducted on the tracker initial load job when fetching data from TeamForge and loading them to datamart. For more information on the performance observation, see TrackerInitialJob Performance Observation.

Based on the result, a parallel processing feature has been introduced to reduce the data processing time. For the observation results on parallel processing, see here.

To enable parallel processing:

1. Change the JVM heap size by setting the Xmx value to -Xmx2048m in the `ETL_JAVA_OPTS` site options token (`/opt/collabnet/teamforge/etc/site-options.conf`).

    > **IMPORTANT:** To make this a permanent change, you must modify the above setting before installing the product. If you do not want to make this a permanent change due to hardware resource constraint or for any other reason, then change the value in the set-env.sh file located in `runtime/conf` directory as below.

    1. Open `set-env.sh` from `<teamforge-installer-base-dir>/runtime/conf` directory.
    2. In `ETL_JVM_OPTS` site options token, change the Xmx setting to `-Xmx2048m`.
2. Restart the CollabNet ETL service.

    ```
    teamforge restart
    ```

    > **WARNING:** If the heap size (Xmx) is not set to 2048 MB for the ETL service, the tracker initial load job is more likely to throw an exception due to insufficient memory.

TrackerInitialJob Performance Observation

A performance observation was conducted on the tracker initial load job when fetching data from TeamForge and loading them to datamart.

For this observation, the tracker initial load job was run on the following key configuration parameters of the environment (configuration/resource allocation/capacity of the test server):

- 64-bit CentOS 6.6

- 8 CPU machine type with 8 GB RAM and 8 GB swap space
- TeamForge application, ETL service and database server are all installed on the same box
- The ETL service is configured to use a maximum heap size of 1524 MB
- Postgres database
- Maximum connections of database configuration server - 400
- Database client connection pool configuration set to 30

It took, approximately, 15 hours and 30 minutes for the tracker initial load job to process the following number of records:

> **NOTE:** All the relevant processes (for example, building dimensions, populating details around facts and hierarchies (team, artifacts and planning folder) were run in a sequence and not in parallel.

- Projects: 570
- Trackers:1692
- Planning folders: 4731
- Artifacts: 184992
- Customers: 1692
- Statuses: 1692
- Releases: 4409
- Categories: 1692
- Teams: 45
- Users: 1556
- Flex fields: 3772
- Flex field values: 15427
- Audit change rows: 3516016
- Audit entry rows: 3110684
- Total number of fields: 27950
- Total number of field values: 62580
- Number of artifact transaction fact rows generated: 769212
- Number of artifact daily snapshot fact rows generated: 536672

Dimension/Fact name	ETL Start Date	ETL End Date	Record Count	Duration
flex_field_dimension	08-11-2015 05:16:03	08-11-2015 14:15:11	4511	9 hrs (approx)
flex_field_value_ dimension			17881	
pf_bridge			15621	

artifact_dependency_bridge			32515	
artifact_transaction_fact				
Batch 1	08-11-2015 14:15:11	08-11-2015 15:02:04	122350	47 minutes
Batch 2	08-11-2015 15:02:04	08-11-2015 16:40:42	130570	1 hr 38 minutes
Batch 3	08-11-2015 16:40:43	08-11-2015 17:42:59	107961	1 hr 02 minutes
Batch 4	08-11-2015 17:42:59	08-11-2015 18:21:24	104260	39 minutes
Batch 5	08-11-2015 18:21:24	08-11-2015 19:03:47	107449	42 minutes
Batch 6	08-11-2015 19:03:47	08-11-2015 20:07:46	104023	1 hr 3 minutes
Batch 7	08-11-2015 20:07:46	08-11-2015 20:41:37	92599	34 minutes
		Total number of `artifact_transaction_fact` rows	769212	
artifact_daily_snapshot_fact				
Batch 1	08-11-2015 20:41:38	08-11-2015 20:42:33	70275	11 minutes
Batch 2	08-11-2015 20:42:33	08-11-2015 20:44:02	87406	
Batch 3	08-11-2015 20:44:03	08-11-2015 20:45:32	78238	
Batch 4	08-11-2015 20:45:37	08-11-2015 20:47:09	72840	
Batch 5	08-11-2015 20:47:19	08-11-2015 20:48:36	72919	
Batch 6	08-11-2015 20:48:36	08-11-2015 20:50:33	79406	

Batch 7	08-11-2015 20:50:34	08-11-2015 20:52:24	75588	
		Total number of `artifact_daily_snapshot_fact` rows	536672	
			Overall time	**15 hrs 30 minutes (approx)**

TrackerInitialJob Performance Observation - Parallel Process

Using parallel processing, it was established that the time taken to process the data was considerably less when compared to the sequential flow.

For this observation, the tracker initial load job was run on the following key configuration parameters of the environment (configuration/resource allocation/capacity of the test server):

- 64-bit CentOS 6.6
- 8 CPU machine
- Overall 8 GB RAM and 8 GB swap space
- TeamForge application, ETL service and database server are all installed on the same box
- The ETL service is configured to use a maximum heap size of 2048 MB
- Postgres database
- Maximum connections of database configuration server - 150
- Database client connection pool configuration set to 40

It took, approximately, 8 hours 40 minutes for the tracker initial load job to process the following number of records:

- Projects: 570
- Trackers:1692
- Planning folders: 4731
- Artifacts: 184992
- Customers: 1692
- Statuses: 1692
- Releases: 4409
- Categories: 1692
- Teams: 45
- Users: 1556
- Flex fields: 3772
- Flex field values: 15427

- Audit change rows: 3516016
- Audit entry rows: 3110684
- Total number of fields: 27950
- Total number of field values: 62580
- Number of artifact transaction fact rows generated: 769212
- Number of artifact daily snapshot fact rows generated: 536672

Dimension/Fact Name	ETL Start Date	ETL End Date	Record Count	Duration	Remarks
flex_field_dimension	27-08-2015 05:59	27-08-2015 06:23	4511	24 minutes	This includes all the dimensions.
flex_field_value_ dimension			17881		
pf_bridge	27-08-2015 06:23	27-08-2015 09:00	15621		Running as three threads in parallel.
artifact_dependency_bridge	27-08-2015 06:23	27-08-2015 14:43	32515		
artifact_transaction_fact					
Batch 1	27-08-2015 06:23	27-08-2015 07:15	122350	52 minutes	
Batch 2	27-08-2015 07:15	27-08-2015 09:10	130570	1 hr 55 minutes	
Batch 3	27-08-2015 09:10	27-08-2015 10:16	107961	1 hr 7 minutes	
Batch 4	27-08-2015 10:16	27-08-2015 10:56	104260	40 minutes	
Batch 5	27-08-2015 10:56	27-08-2015 11:42	107449	46 minutes	
Batch 6	27-08-2015 11:42	27-08-2015 12:51	104023	1 hr 9 minutes	
Batch 7	27-08-2015 12:51	27-08-2015 13:28	92599	37 minutes	
Total number of `artifact_transaction_fact` rows			769212		
artifact_daily_snapshot					

Batch 1	27-08-2015 13:28	27-08-2015 13:29	70275	12 minutes	
Batch 2	27-08-2015 13:29	27-08-2015 13:31	87406		
Batch 3	27-08-2015 13:31	27-08-2015 13:32	78238		
Batch 4	27-08-2015 13:32	27-08-2015 13:34	72840		
Batch 5	27-08-2015 13:34	27-08-2015 13:36	72919		
Batch 6	27-08-2015 13:36	27-08-2015 13:38	79406		
Batch 7	27-08-2015 13:38	27-08-2015 13:40	75588		
Total number of `artifact_daily_snapshot_fact` rows			536672		
Overall time				**8 hrs 40 minutes (approx)**	

TeamForge enables user authentication both against its internal database and against other external authentication services such as LDAP, OAuth, and SAML. This section provides information on how to set up TeamForge for authenticating its users against these services.

TeamForge supports the following identity management frameworks:

- OAuth 2.0 Authorization Framework

  With the new TeamForge Identity Management built on OpenID Connect (OIDC) and OAuth 2.0 authorization frameworks, TeamForge can now act as an ID Provider (IdP). As an IdP, TeamForge can authorize a third-party client application to obtain limited access to its services either on behalf of a Resource Owner (user) or on behalf of the client application itself.

- SAML

  SAML is an XML-based open standard developed by OASIS Security Services Technical Committee. It defines a framework to perform web browser SSO using secure tokens for exchaning security information between web applications.

- LDAP

  LDAP (Lightweight Directory Access Protocol) is an application protocol that works on a layer on top of the TCP/IP stack and accesses your directory service providers such as Active Directory for providing user authentication. For more details on LDAP, see RFC2251 - Light-weight Directory Access Protocol (v3).

- SAML+LDAP

  With SAML+LDAP IdP, the TeamForge users can reap the benefits of both SAML and LDAP authentication mechanisms in a unified manner. With SAML+LDAP authentication, while SAML enables TeamForge users to access web applications, the LDAP authentication supports user authentication required for CLI applications. For example, if a user performs a source code commit in CVS/Git/SVN repository, the user can get authenticated via LDAP.

With the new TeamForge Identity Management built on OpenID Connect (OIDC) and OAuth 2.0 authorization frameworks, TeamForge can now act as an ID Provider (IdP). As an IdP, TeamForge can authorize a third-party client application to obtain limited access to its services either on behalf of a Resource Owner (user) or on behalf of the client application itself.

This topic discusses TeamForge OAuth architecture, scopes, supported authorization grant types, Federated Identity Management in TeamForge, and so on. This topic also discusses step by step how to add a client application that wants to access TeamForge's services.

TeamForge OAuth Nomenclature

IMPORTANT: It is assumed that you are familiar with the OAuth 2.0 and OpenID Connect frameworks, terms and concepts. It's recommended that you read The OAuth 2.0 Authorization Framework RFC 6749 and get familiar with **OAuth Roles**, **Protocol Flow**, **Access Tokens**, **Grant Types**, **Client Types** and so on before you proceed.

Here's a list of links and references for further reading.

Term	Description
OAUTH	The OAuth 2.0 Authorization Framework RFC 6749
OIDC	OpenID Connect Core 1.0
BEARER TOKEN	The OAuth 2.0 Authorization Framework: Bearer Token Usage RFC 6750
SAMLBEARER	SAML 2.0 Bearer Assertion Profiles for OAuth 2.0
JWE	JSON Web Encryption (JWE)
JWS	JSON Web Signature (JWS)
JWT	JSON Web Token (JWT)
JWA	JSON Web Algorithms (JWA)
JWK	JSON Web Key (JWK)

TeamForge OAuth Overview

In the traditional client-server authentication model, client applications must have access to the Resource Owner's credentials in order to request and access protected resources on the Resource Server. Sharing Resource Owner's credentials such as user names and passwords to third party applications poses several problems, security being one of the major ones.

Frameworks such as the OAuth 2.0 and OpenID Connect help in mitigating such security risks and hence TeamForge's authentication model has been rebuilt on top of OpenID Connect and OAuth 2.0 Authorization frameworks.

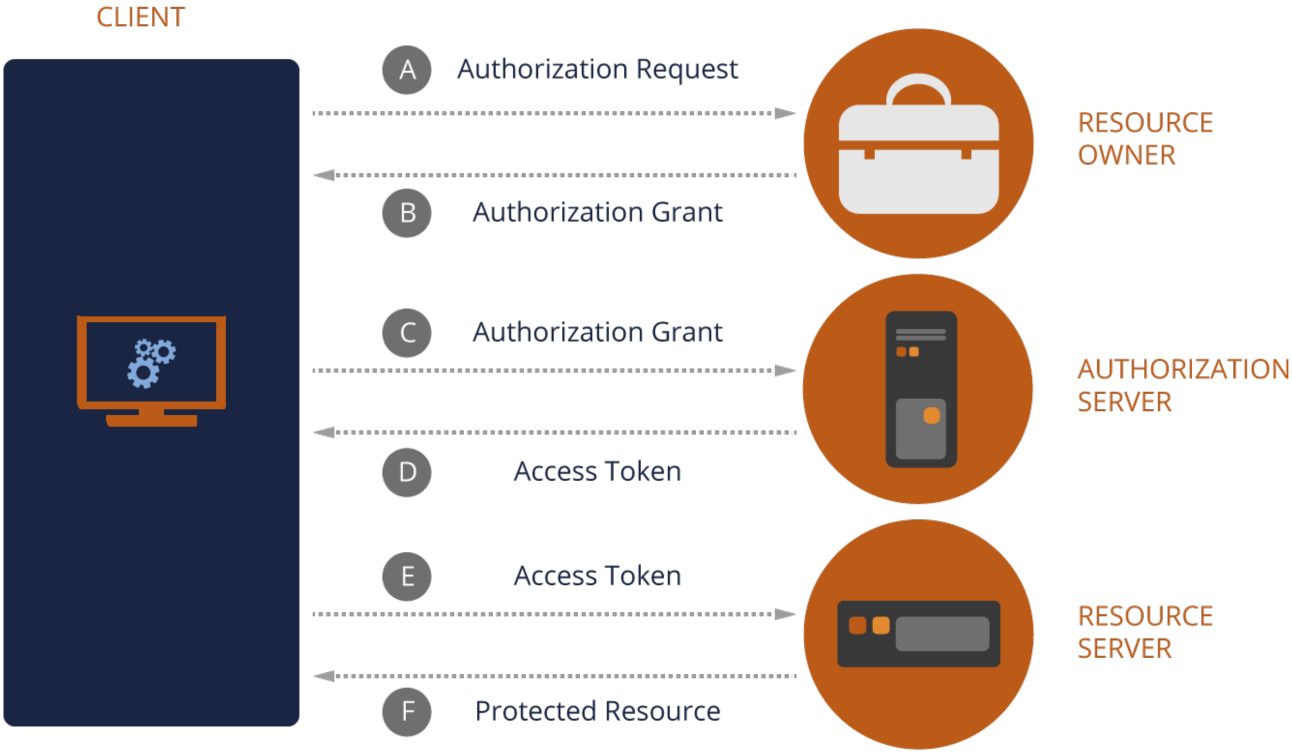
TeamForge OAuth Roles

TeamForge OAuth authorization process involves interaction between four entities (roles) such as the user, IdP, SP and client application. The following table lists the roles that an entity can assume in the TeamForge OAuth setup.

Entity/ Role	OAuth Term	OIDC Term	SAML Term	Description
User	Resource Owner	End-User	User	Human or machine that intends to access a resource of an SP. Users are, in general, humans. If a machine acts as a user, it is called `Agent`.
Service Provider (SP)	Resource Server	Relying Party	Service Provider, Relying Party	Entity that demands authentication to grant access to some resource. The SP trusts the IdP to perform the authentication on its behalf.
Client	Client	-	Browser	Application interacting with SP and IdP on the user's behalf.
Identity Provider (IdP)	Authorization Server	Provider	Identity Provider, Asserting Party	Entity that proves, identifies and authenticates a user.

OAuth 2.0 Abstract Protocol Flow

The following illustration shows the OAuth 2.0 protocol flow.



Abstract Protocol Flow

- (A) The client application requests authorization from the Resource Owner.
- (B) The Resource Owner responds to the client with a credential called `Authorization Grant`.
- (C) The client then redeems the `Authorization Grant` with the Authorization Server for a valid `Access Token` (D).
- (E) The client then reaches out to the Resource Server with the Access Token and gains access to the protected resource.

For more information about the OAuth 2.0 protocol flow, see OAuth 2.0 Protocol Flow.
TeamForge can act both as an IdP and Service Provider and at times as a client too. The TeamForge Web Application is one of the system defined clients that use TeamForge IdP's OAuth services. For more information about system defined clients, see System Defined Clients.

TeamForge API Authentication and Access Tokens

Both TeamForge REST and SOAP APIs use OAuth 2.0 access tokens for authentication. Clients can obtain access tokens from the token endpoint which is located at `/oauth/auth/token`.

Access Tokens

The tokens used by the TeamForge API are Bearer Tokens as specified in RFC 6750. This means that it is possible and allowed to share tokens with multiple clients or to have clients pass tokens to intermediate services, which then delegate tokens to TeamForge. TeamForge tokens use the JSON Web Token (JWT) standard. However, clients should consider access tokens to be opaque in order to guarantee compatibility with future TeamForge versions. It is the client's responsibility to protect the access token against theft. This means that access tokens should only be transmitted over SSL-secured connections and should not be persisted.

TeamForge OAuth Scopes

Scopes can be used to restrict which services a token can be used for. Limiting the number of scopes decreases the potential damage that could occur in case an access token is stolen, so it is advisable to restrict the number of scopes to a minimum.

TeamForge currently supports the following list of scopes.

Scope	Description
urn:ctf:services:ctf	CollabNet TeamForge application services. Use this scope to call the TeamForge REST API.
urn:ctf:services:svn	Subversion services. Use this to call the Subversion REST API.
urn:ctf:services:gerrit	Git/Gerrit services. Use this to call the Git REST API.
urn:ctf:services:soap60	SOAP60 services. Use this needed to call the TeamForge SOAP API or the EventQ Reporting API.

Client ID and Client Secret

In order to use TeamForge's OAuth services, a client application must be registered with TeamForge. Upon registration, the client gets its client credentials: `Client ID` and `Client Secret`. For more information about how to get the Client ID and Client Secret, see Add Clients.

Client ID	The Client ID is a unique identifier used by TeamForge to identify the client application and is used in building the Authorization URLs.
Client Secret	The Client Secret is a unique identifier used to authenticate the client application's identity.

TeamForge OAuth Grant Types

When a client requests access token from an Authorization Server, it uses the Authorization Grant, which is a credential that represents the Resource Owner's authorization to access protected resources. Of the four standard grant types defined in The OAuth 2.0 Authorization Framework RFC 6749, TeamForge supports the following three grant types:

- Authorization Code (`authorization_code`)
- Resource Owner Password Credentials (`password`)
- Client Credentials (`client_credentials`)

In addition, TeamForge supports the following custom grant type:

- Anonymous (`rn:ctf:grant_type:anonymous`)

TeamForge System Defined Grant Types

TeamForge also supports the following system defined grant types. However, these grant types are used by TeamForge's system defined clients to obtain access tokens and are therefore not supposed to be used for other custom clients.

- JSession (`urn:ctf:grant_type:jsession`)
- SCM Admin (`urn:ctf:grant_type:scmrequestkey:scmadmin`)
- SCM Viewer (`urn:ctf:grant_type:scmrequestkey:scmviewer`)
- Token (`urn:ctf:grant_type:token`)

Authorization Code Grant Type (authorization_code)

The Authorization Code grant type can be used if the client intends to use TeamForge as its IdP. This grant type, if used by clients, provides an access token for a valid `authorization code`. This is a redirection based grant type and hence you must add one or more `Redirect URIs` while adding clients that use this grant type.

NOTE: Inlcude a comma separated list of `Redirect URIs` in case you want to set up multiple URIs.

How to get an authorization code?

Clients (Relying Party) must request the `authorization code` from TeamForge IdP URL (`/oauth/oidc/authorize`) with the following parameters:

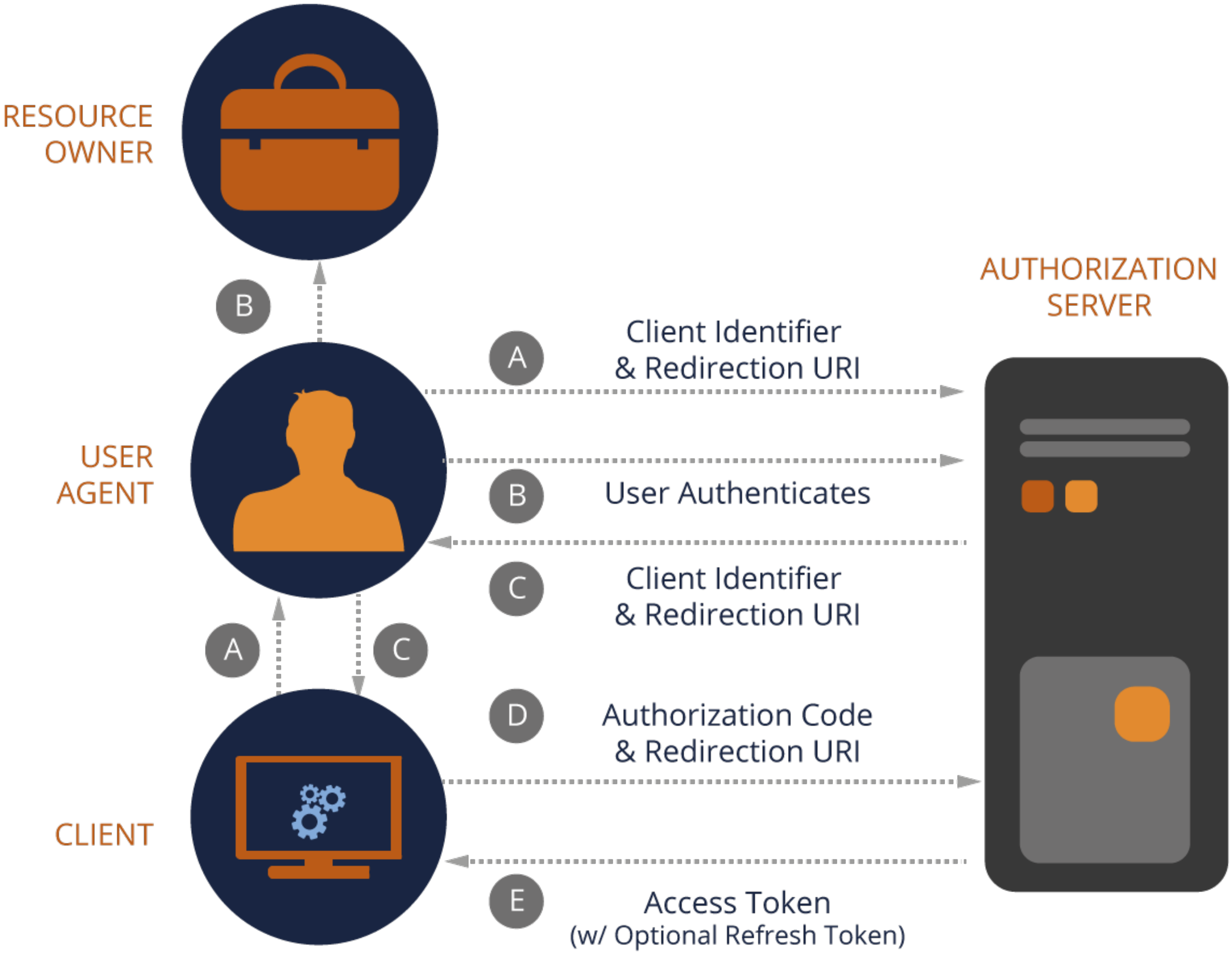
```
scope = open_id (mandatory for OIDC clients)
response_type = code (mandatory for OIDC clients)
client_id = <Client ID generated while registering the client with TeamForge>
redirect_uri = <URL encoded Redirect URI>
```

After successful authorization, TeamForge IdP provides the `authorization code` to the redirect URI configured while adding the client. Here’s a sample OIDC response after a successful authorization:

```
https://<your_redirect_uri>?code=<authorization_code>
```

Up on getting the `authorization code`, the client can redeem it with the Authorization Server to obtain an access token.

Protocol Flow


RESOURCE OWNER
AUTHORIZATION SERVER
B
A
Client Identifier & Redirection URI
USER AGENT
B
User Authenticates
C
Client Identifier & Redirection URI
A
C
D
Authorization Code & Redirection URI
CLIENT
E
Access Token (w/ Optional Refresh Token)

Authorization Code Flow

Usage Example to Obtain an Access Token

```
# Base URL of TeamForge site.
site_url="https://teamforge.example.com"

# Requested scope (all)
scope="urn:ctf:services:ctf urn:ctf:services:svn urn:ctf:services:gerrit urn:c
tf:services:soap60"
```

```
# Client's authorization code
code=<your authorization code>

curl -d "grant_type=authorization_code&client_id=<your Client ID>&client_secre
t=<your Client Secret>&scope=$scope&code=$code" $site_url/oauth/auth/token
```

A successful response will return the HTTP 200 status code and the response body will contain something like this:

```
            {
              "access_token": "eyJraWQiOiIxIiwiYWxnIjoiUlMyNTYifQ.eyJzdWIiOiJhZ
G1pbiIsImF1ZCI...",
              "token_type": "Bearer"
            }
```

In compliance with the Bearer Tokens specification (RFC 6750), TeamForge expects access tokens to be passed in the Authorization header:

```
GET /resource HTTP/1.1
Host: teamforge.example.com
Authorization: Bearer SAksa921hjsi...
```

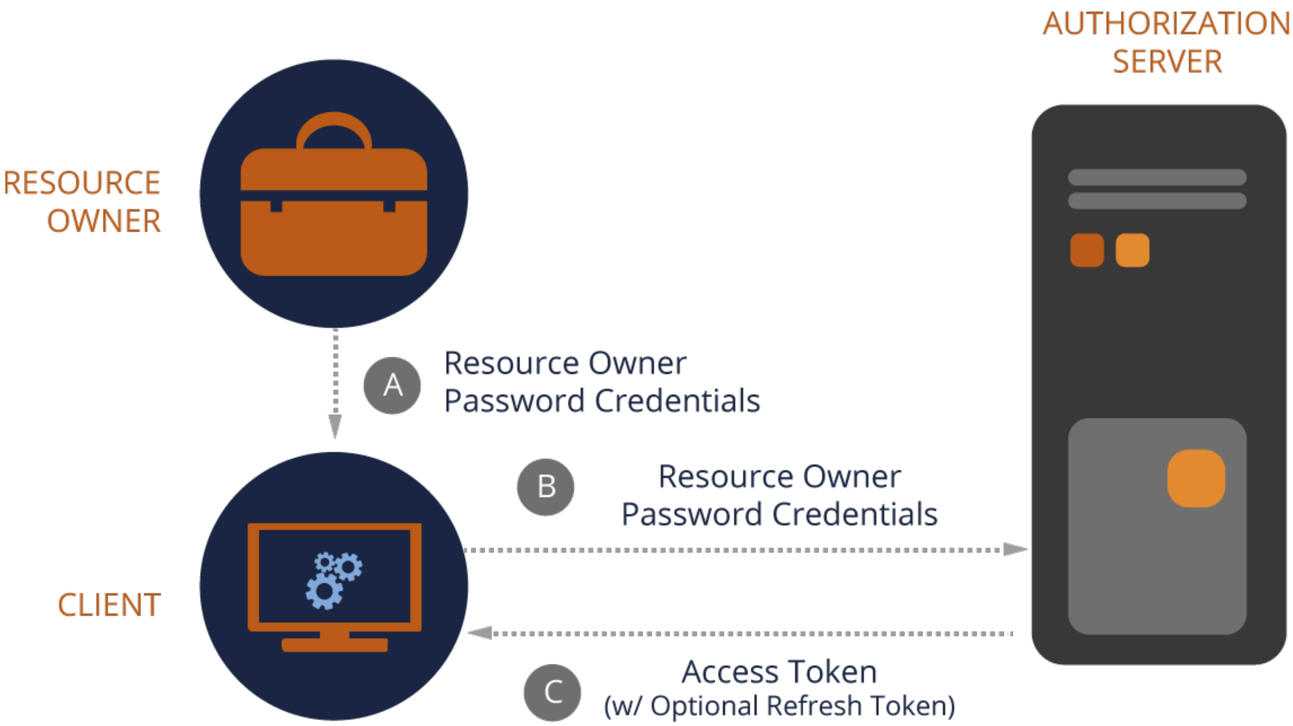
The only exception to this is the SOAP API which expects the token to be passed as part of the SOAP request payload in accordance with the API documentation.

For more information, see Authorization Code Grant Type.

Resource Owner Password Credentials Grant Type (`password`)

Clients can use the Resource Owner Password Credentials grant type to get an access token for a given username and password.

Protocol Flow


AUTHORIZATION SERVER
RESOURCE OWNER
A
Resource Owner Password Credentials
B
Resource Owner Password Credentials
CLIENT
C
Access Token
(w/ Optional Refresh Token)

Resource Owner Password Credentials Flow

Usage Example to Obtain an Access Token

```
# Base URL of TeamForge site.
site_url="https://teamforge.example.com"

# TeamForge authentication credentials.
username="foo"
password="bar"

# Requested scope (all)
scope="urn:ctf:services:ctf urn:ctf:services:svn urn:ctf:services:gerrit urn:c
tf:services:soap60"

curl -d "grant_type=password&client_id=<your Client ID>&scope=$scope&username=
$username&password=$password" $site_url/oauth/auth/token
```

You can see from this example that it is sending an HTTP POST request to the `/oauth/auth/token` endpoint, and the content of the request body contains the following data:

- `grant_type=password`: Indicates you are providing username and password.
- `client_id=<your Client ID>`: Client ID generated while adding the client.
- `scope= ...`: Space-separated list of requested scopes.
- `username=value&password=value`: The valid TeamForge user credentials.

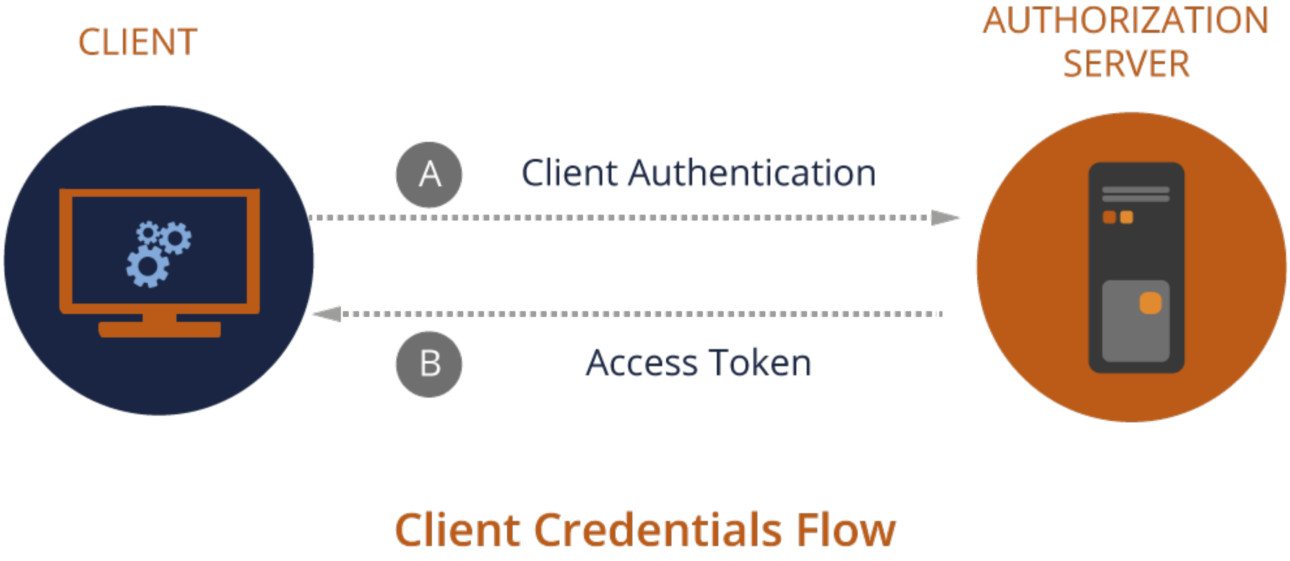
For more information, see Resource Owner Password Credentials Grant Type.

Client Credentials Grant Type (`client_credentials`)

Clients can use the Client Credentials grant type to get an access token for a given Client ID and Client Secret. Use this grant type only for trusted and confidential clients.

> **IMPORTANT:** Clients that use the Client Credentials grant type must be assigned with a valid TeamForge license in order for the clients to be able to access TeamForge resources as OAuth clients.

Protocol Flow



Usage Example to Obtain an Access Token

```
# Base URL of TeamForge site.
site_url="https://teamforge.example.com"

# Requested scope (all)
scope="urn:ctf:services:ctf urn:ctf:services:svn urn:ctf:services:gerrit urn:c
tf:services:soap60"

curl -d "grant_type=client_credentials&client_id=<your Client ID>&client_secre
t=<your Client Secret>&scope=$scope" $site_url/oauth/auth/token
```

For more information, see Client Credentials Grant Type.

Anonymous Grant Type (urn:ctf:grant_type:anonymous)

Use this grant type for clients that need to access TeamForge's public resources.

Usage Example to Obtain an Access Token

```
# Base URL of TeamForge site.
site_url="https://teamforge.example.com"

# Requested scope (all)
scope="urn:ctf:services:ctf urn:ctf:services:svn urn:ctf:services:gerrit urn:c
tf:services:soap60"

curl -d "grant_type=urn:ctf:grant_type:anonymous&client_id=<your Client ID>&cl
ient_secret=<your Client Secret>&scope=$scope" $site_url/oauth/auth/token
```

Unsupported Grant Types

The following grant types are not supported by TeamForge.

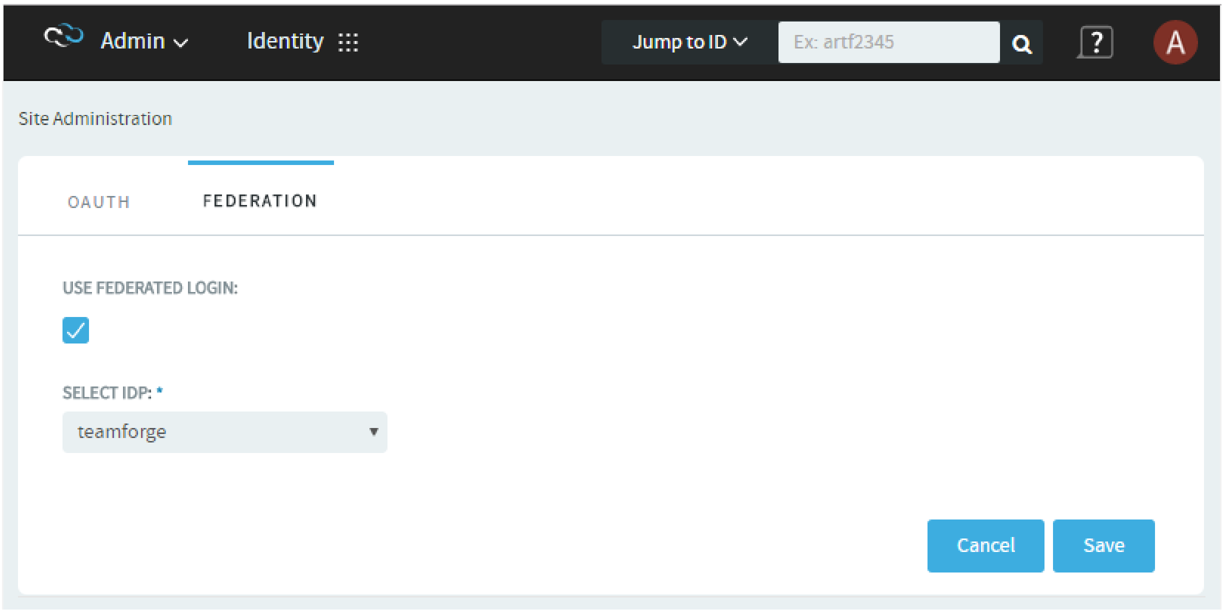
- refresh_token
- urn:ietf:params:oauth-grant-type:saml2-bearer

Federated Identity Management

Federated Identity Management refers to the process of storing user credentials with an IdP (such as TeamForge, SAML and so on) and having the Service Provider (SP) rely on the IdP to validate user credentials when he/she tries to access its resources.

TeamForge supports federated identity. By default, federated login is disabled in TeamForge. Site Administrators must enable federated login and select an IdP.

1. Log on to TeamForge as a Site Administrator.
2. Select **My Workspace > Admin**.
3. Select **My Page > Identity**.
4. Select the **Federation** tab.
5. Select the **Use Federated Login** check box and select an IDP from the drop-down list.



USE FEDERATED LOGIN:

☑

SELECT IDP: *

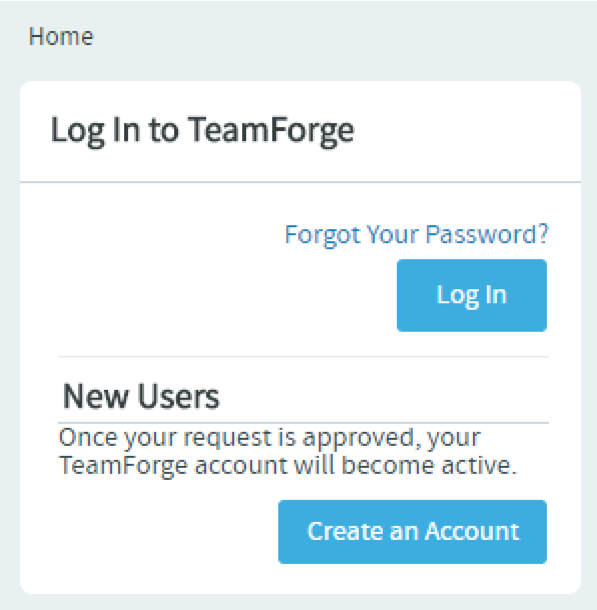
teamforge ▾

6. Click **Save**.

Logging into TeamForge in a Federated Login Setup

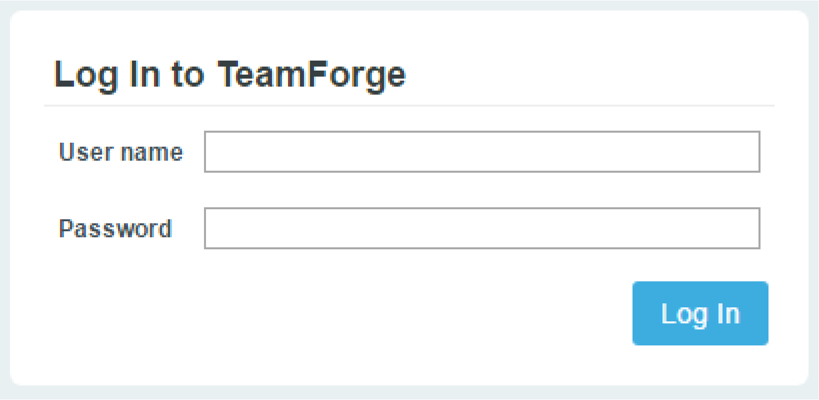
Once enabled, TeamForge Web Application, by virtue of being a system defined client application (see System Defined Clients), follows the OAuth authorization/authentication process to access TeamForge services on behalf of the Resource Owner (user).

1. Click **Login**.



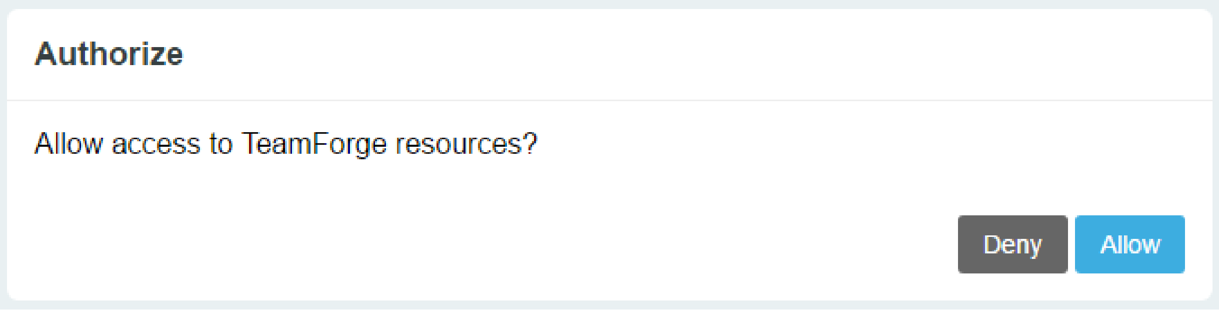
Once your request is approved, your TeamForge account will become active.

You are redirected to an authorization page.



2. Type the user name and password and click **Log In**.

   You, as a Resource Owner, are then prompted to either allow or deny the client application (TeamForge Web Application) to access the resources on the Resource Server (TeamForge Application Server).



3. Click **Allow** or **Deny** to allow or deny access respectively.

System Defined Clients

TeamForge OAuth consists of the following system defined clients:

- TeamForge Web Applicaion Client (`ctfweb`)
- Code Browser Client (`codebrowser`)
- SOAP60 Client (`soap60-client`)
- SCM Client (`scm-client`)
- API Client (`api-client`)
- App Client (`app-client`)

These clients have been pre-defined as illustrated in the following table.

System Defined Client	Grant Type	Services	Token Life Time (in seconds)
ctfweb	authorization_code urn:ctf:grant_type:jsession	soap60,ctf,gerrit,svn	3600
codebrowser	urn:ctf:grant_type:jsession	soap60,ctf,gerrit,svn	3600
soap60-client	password urn:ctf:grant_type:anonymous	soap60	3600
scm-client	urn:ctf:grant_type:scmrequestkey:scmviewer urn:ctf:grant_type:scmrequestkey:scmadmin	soap60,ctf	3600
api-client	password	soap60,ctf,gerrit,svn	3600
app-client	urn:ctf:grant_type:token	soap60,ctf,gerrit,svn	3600

You cannot edit these system defined clients. The following error message shows up if you try to edit these clients.

```
Error editing oauth client: System defined clients are not allowed to be
modified.
```

Add Clients

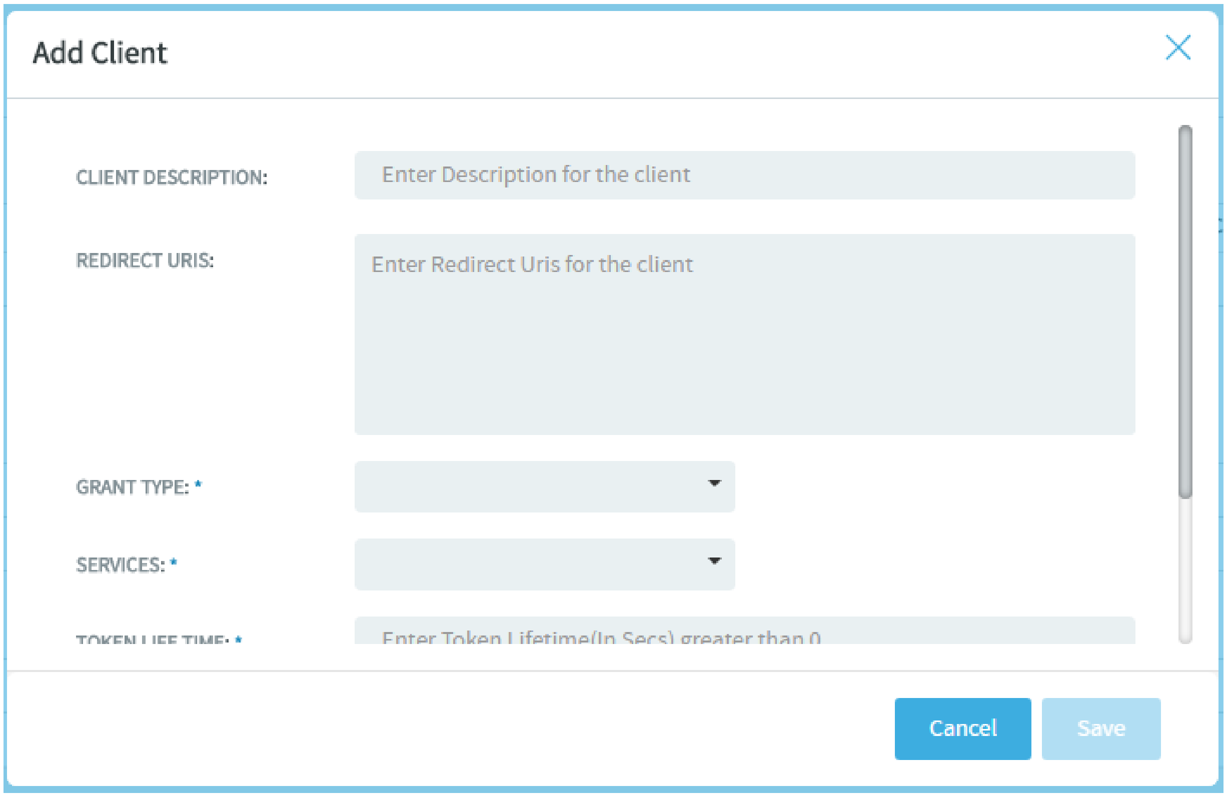
In order to use TeamForge's OAuth services, a client application must be registered with TeamForge. Upon registration, the client gets its client credentials: `Client ID` and `Client Secret`.

Client ID	The Client ID is a unique identifier used by TeamForge to identify the client application and is used in building the Authorization URLs.
Client Secret	The Client Secret is a unique identifier used to authenticate the client application's identity.

To add a TeamForge OAuth client:

1. Log on to TeamForge as a Site Administrator.
2. Select **Admin** from the **My Workspace** menu.
3. Select **Identity** from the **Projects** menu.
4. Click **Add Client** from the **OAuth** tab.

   The **Add Client** dialog box appears.

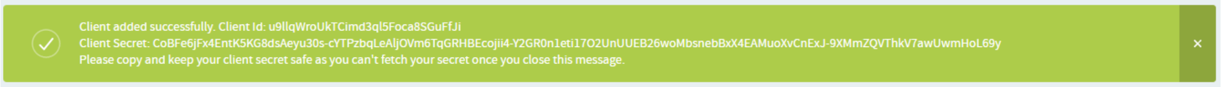

Add Client
CLIENT DESCRIPTION:
Enter Description for the client
REDIRECT URIS:
Enter Redirect Uris for the client
GRANT TYPE: *
SERVICES: *
TOKEN LIFE TIME: *
Enter Token Lifetime(In Secs) greater than 0
Cancel
Save

5. Enter client information.

The following parameters are required to register a client application with TeamForge:

- **Client Application Name**: Type a name for your client.
- **Redirect URIs**: Inlcude a comma separated list of `Redirect URIs` in case you want to set up multiple URIs. Redirect URIs are required only for clients that use the **Authorization Code** grant type.
- **Grant Type**: Select the client's grant type.
- **Services** (scopes): Select one or more TeamForge OAuth scopes.
- **Token Life Time** (in seconds): Type the duration (in seconds) for which the token is valid.
- **License Types**: Select a TeamForge license for the client. This is required only for clients that use the **Client Credentials** grant type.

6. Click **Save**. Once the client is added successfully, a message containing the client's credentials is shown.



Client added successfully. Client Id: u9llqWroUkTCimd3ql5Foca8SGuFfJi
Client Secret: CoBFe6jFx4EntK5KG8dsAeyu30s-cYTPzbqLeAljOVm6TqGRHBEcojii4-Y2GR0n1eti17O2UnUUEB26woMbsnebBxX4EAMuoXvCnExJ-9XMmZQVThkV7awUwmHoL69y
Please copy and keep your client secret safe as you can't fetch your secret once you close this message.

> **WARNING:** Once the client is added successfully, the `Client ID` and `Client Secret` are generated and shown in the message. You must copy and keep your `Client Secret` safe before closing the message as you cannot fetch your client secret later from anywhere else.

SAML is an XML-based open standard developed by OASIS Security Services Technical Committee. It defines a framework to perform web browser SSO using secure tokens for exchanging security information between web applications.

For more information about SAML, its concepts and components, see https://www.oasis-open.org/.

SAML Terms and Their Purpose

- **End User / Browser:** The end user is generally a human or a browser (agent) who accesses the Service Provider to get access to a service or a protected resource. The browser carries out all the redirections from the SP to the IdP and vice versa.

- **Service Provider (SP):** The entity that provides its protected resource when an end user tries to access this resource. To accomplish the SAML based SSO authentication, the Service Provider must have the Identity Provider's metadata.

> **NOTE:** It is not necessary that the authentication flow should start from a Service Provider. Even an IdP can initiate the authentication process.

- **Identity Provider (IdP):** Defines the entity that provides the user identities, including the ability to authenticate a user to get access to a protected resource / application from a Service Provider. To accomplish the SAML based SSO authentication, the IdP must have the Service Provider's metadata.
- **SAML Request:** This is the authentication request generated by the Service Provider to request an authentication from the Identity Provider for verifying the user's identity.
- **SAML Response:** The SAML Response contains the acutal assertion of the authenticated user and is generated by the Identity Provider. The SAML Response also consists of additional information such as user profile information, group or role information and so on based on what the Service Provider can support.
- **Service Provider-initiated Authentication Flow:** This describes the SAML authentication flow initiated by the Service Provider. The authentication process from the SP is triggered when the user tries to access a resouce or log on to the Service Provider application. A typical example is that a browser trying to access a protected resource from the Service Provider.
- **Identity Provider-initiated Authentication Flow:** This describes the SAML authentication flow initiated by the Identity Provider. Unlike the SP-initiated authentication flow in which the authentication is triggered by a redirection from the Service Provider, here the IdP initiates the SAML Response that is redirected to the SP to assert the user's identity.

How SAML-based SSO Works in TeamForge?

In addition to OAuth 2.0 (with Open ID Connect), TeamForge supports SAML (Security Assertion Markup Language) authentication and authorization protocol.

As in a typical SSO-enabled environment, Single Sign-on in TeamForge works in such a way that the Identity Provider “asserts” the identity of the user and the Service Provider consumes the “assertion” and passes the identity information to the application. This is done by exchanging digitally signed XML documents.

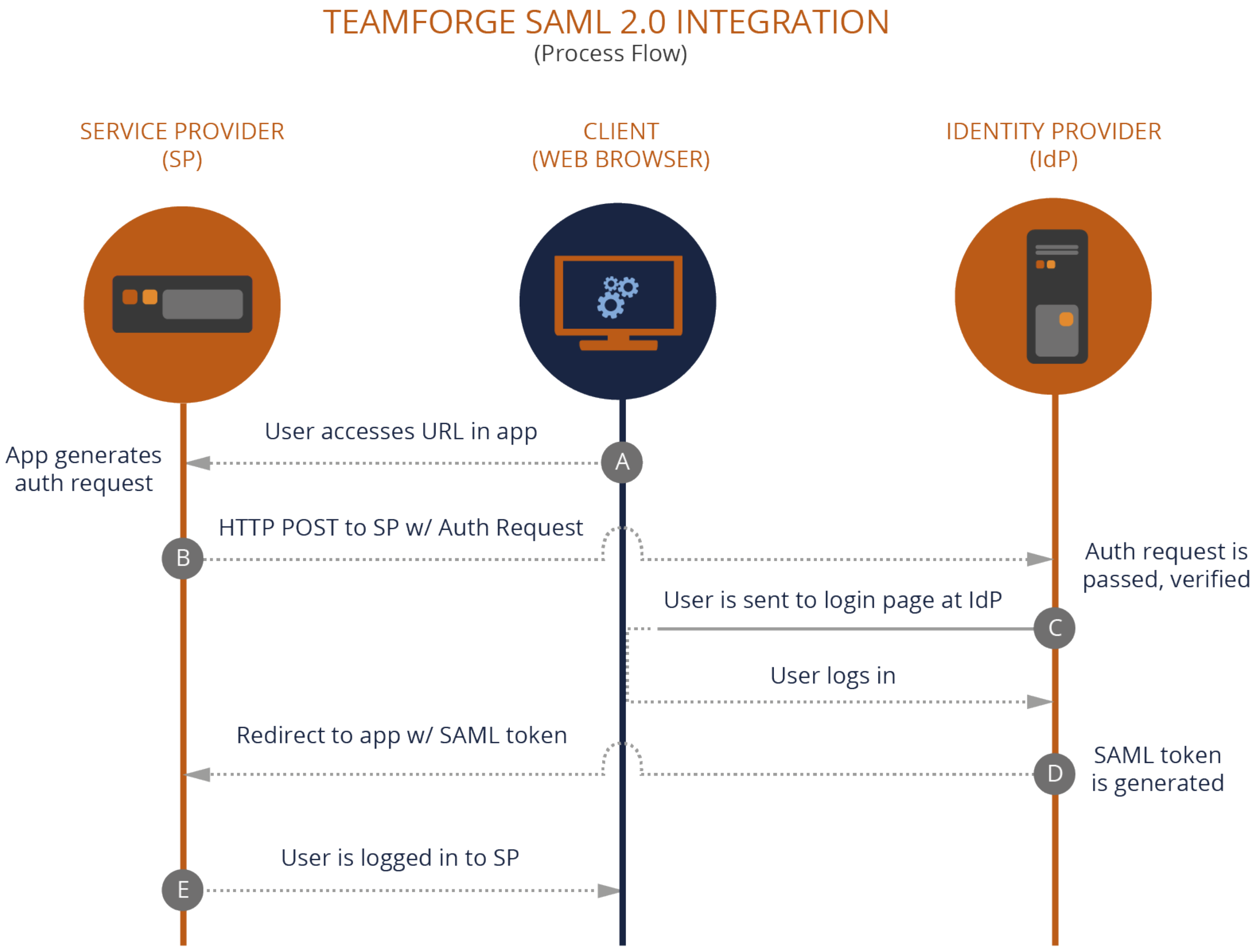
TeamForge, as a SAML compliant Service Provider, can be integrated with any SAML compliant Identity Provider. TeamForge Administrators should make sure that the Identity Provider is SAML 2.0 compliant and must keep the IdP metadata handy before configuring the IdP details in TeamForge.

SAML metadata is an XML file that contains configuration information to be shared between the Service Provider and the Identity Provider.

- The *Service Provider metadata XML* file contains the SP certificate, the entity ID, ACS parameters and so on. To get the TeamForge (Service Provider) metadata, check at: https://«hostname»/oauth/metadata.jsp

- The *Identity Provider metadata XML* file contains the IdP certificate, the entity ID, redirect URL, logout URL and so on.

TeamForge Administrator must keep the IdP metadata handy before integrating TeamForge with a SAML IdP. The following illustration shows the high-level authentication flow of SAML integration in TeamForge. This is typically a Service Provider-initiated SSO workflow.


TEAMFORGE SAML 2.0 INTEGRATION
(Process Flow)
SERVICE PROVIDER (SP)
CLIENT (WEB BROWSER)
IDENTITY PROVIDER (IdP)
User accesses URL in app
A
App generates auth request
HTTP POST to SP w/ Auth Request
B
Auth request is passed, verified
User is sent to login page at IdP
C
User logs in
Redirect to app w/ SAML token
D
SAML token is generated
User is logged in to SP
E

Let's see how it works:

1. The end user tries to access a resource URL within the application provided by the Resource Server (Service Provider) via the Client application / Web Browser (A).

2. The Resource Server application generates the authentication request for user and sends it to Authorization Server (Identity Provider). The Service Provider uses the HTTP POST binding component to send the request to the IdP (B).

3. If the authentication is successful, the user is redirected to the Authorization Server's (IdP) login page, through which the user logs in (C).

4. The IdP generates the security token based on SAML assertions for the user and sends the response with the SAML token to the Resource Server application (D).

5. The user now gets a session in the Resource Server and can access the resources in it.

Setting up TeamForge in a SAML-compliant Third-party IdP Environment

For TeamForge to support SAML based SSO from a SAML-compliant third-party IdP, it is required to set up TeamForge in the IdP environment. This means that it is necessary to configure the SAML IdP with the details of TeamForge, who in this case is the SAML Service Provider.

Configuring a SAML IdP is beyond the scope of TeamForge Administrators, as you can use any third-party IdP based on the business requirements.

Once a SAML IdP has been set up, the SAML IdP administrator can set up TeamForge as a Service Provider with the SAML IdP and keep both the IdP and SP metadata handy for creating the TeamForge-SAML IdP integration in TeamForge.

For setting up SAML IdP integration in TeamForge, enable Federated Login in TeamForge.

1. Log on to TeamForge as a Site Administrator.

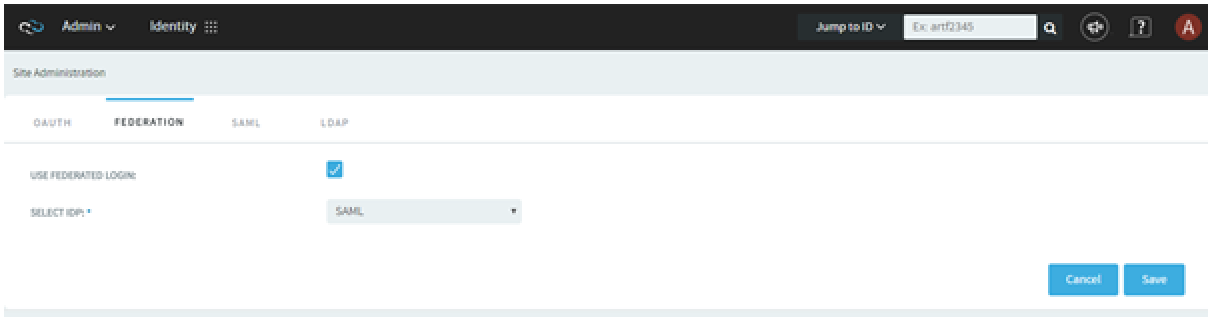
2. Select **My Workspace > Admin**.

3. Select **Projects > Identity**.

4. Select the *Federation* tab.

5. Select the **Use Federated Login** check box and select *SAML* as the IdP from the drop-down list.

6. Click **Save**.

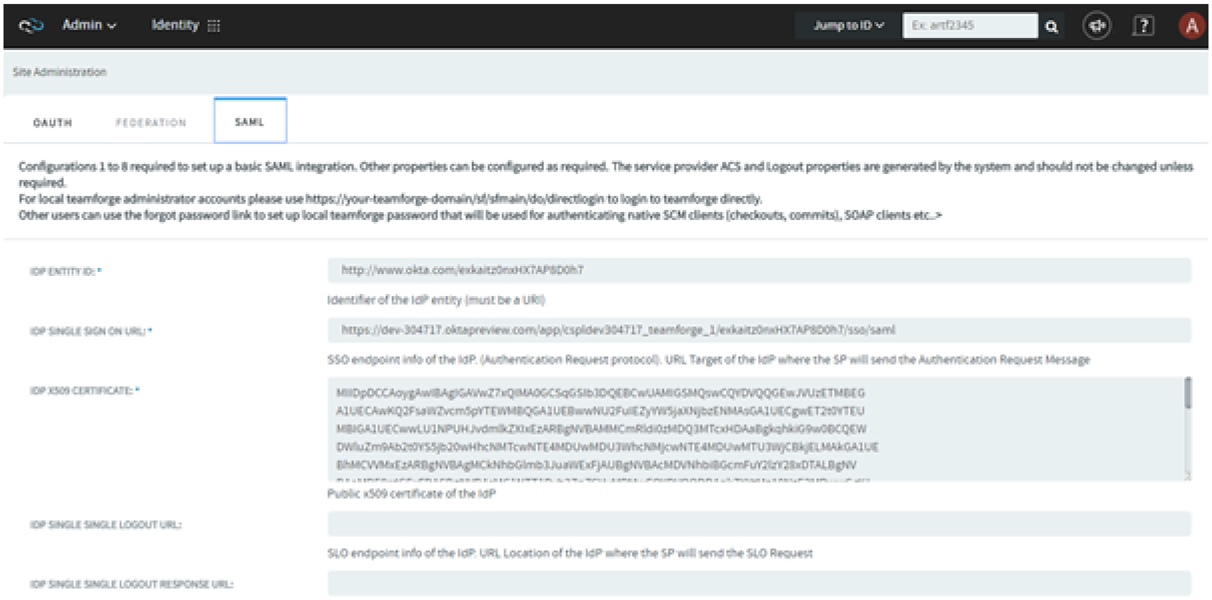


USE FEDERATED LOGIN: ☑

SELECT IDP: * SAML

7. Select the *SAML* tab. This page is used to capture the security configurations of TeamForge and the SAML IdP. The IdP details that you provide in this page is obtained from the metadata XML of the third-party IdP.

> **IMPORTANT:** The Service Provider ACS (Assertion Consumer Service) Logout related properties are generated by the system, hence they should not be changed unless required.



Configurations 1 to 8 required to set up a basic SAML integration. Other properties can be configured as required. The service provider ACS and Logout properties are generated by the system and should not be changed unless required.
For local teamforge administrator accounts please use https://your-teamforge-domain/sf/sfmain/do/directlogin to login to teamforge directly.
Other users can use the forgot password link to set up local teamforge password that will be used for authenticating native SCM clients (checkouts, commits), SOAP clients etc..>

IDP ENTITY ID: * http://www.okta.com/exkaitz0nxHXTAP8D0h7

Identifier of the IdP entity (must be a URI)

IDP SINGLE SIGN ON URL: * https://dev-304717.oktapreview.com/app/cspldev304717_teamforge_1/exkaitz0nxHXTAP8D0h7/sso/saml

SSO endpoint info of the IdP. (Authentication Request protocol). URL Target of the IdP where the SP will send the Authentication Request Message

IDP X509 CERTIFICATE: *

MIIDpDCCAoygAwIBAgIGAVwZ7xQIMA0GCSqGSIb3DQEBCwUAMIGSMQswCQYDVQQGEwJVUzETMBEG
A1UECAwKQ2FsaWZvcm5pYTEWMBQGA1UEBwwNU2FuIEZyYW5jaXNjbzENMAsGA1UECgwET2t0YTEU
MBIGA1UECwwLU1NPUHJvdmlkZXIxEzARBgNVBAMMCmRldi0zMDQ3MTcxHDAaBgkqhkiG9w0BCQEW
DWluZm9Ab2t0YS5jb20wHhcNMTcwNTE4MDUwMDU3WhcNMjcwNTE4MDUwMTU3WjCBkjELMAkGA1UE
BhMCVVMxEzARBgNVBAgMCkNhbGlmb3JuaWExFjAUBgNVBAcMDVNhbiBGcmFuY2lzY28xDTALBgNV

Public x509 certificate of the IdP

IDP SINGLE SINGLE LOGOUT URL:

SLO endpoint info of the IdP. URL Location of the IdP where the SP will send the SLO Request

IDP SINGLE SINGLE LOGOUT RESPONSE URL:

This table provides the parameters and their description used in the SAML configuration page.

> **IMPORTANT:** Configuration details are mandatory for fields 1 through 8 for a basic SAML integration.

Parameter Name	Description

IDP Entity ID	Defines the unique identifier of the Identity Provider. It must be an URI.
IDP Single Sign on URL	Defines the URL that defines the SSO endpoint of the IdP. It is the target URL of the IdP where the SP sends its authentication request message.
IDP X509 Certificate	Defines the digital certificate that verifies the public key of the IdP.
IDP Single Sign on Logout URL	Identity Provider’s Single Sign on Logout URL. If the IdP does not support logout, leave this blank.
IDP Single Sign on Logout Response URL	Defines the Single Sign-on Logout (SLO) endpoint of the IdP that specifies the URL location of the Idp where the SP will send the SLO response. If this is left blank, the same URL as logout service URL will be used. This property can be used, if the IdP uses a separate URL for sending a logout request and response.
Service Provider Entity ID	Defines the unique identifier of Service Provider. It must be an URI.
Assertion Consumer Service URL	Defines the URL of the Service Providers Assertion Consumer Service, where the assertion from the IdP will be sent.
Service Provider Logout URL	Defines the URL of the Service Provider where the Logout Response message will be returned.
Assertion Consumer Service Binding	Defines which SAML protocol binding to be used when returning the Response message. TeamForge supports “urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST“ binding only.
Service Provider Logout Binding	Defines which SAML protocol binding to be used when returning the logout response or sending the logout request message. TeamForge supports “urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect” binding only.
Name ID Format	Defines the constraints on the name identifier to be used to represent the requested subject. It is a mandatory attribute sent by the IdP in its SAML response to make the federation. TeamForge supports the following three Name ID formats: • urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified [default] • urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress • urn:oasis:names:tc:SAML:2.0:nameid-format:transient
Service Provider X509 Certificate	Defines the digital certificate that verifies the public key of the SP.
Service Provider Private Key	Defines the private key of the Service Provider. **Required Format:** PKCS#8 BEGIN PRIVATE KEY.

	If you have PKCS#1 BEGIN RSA PRIVATE KEY, convert it by using “openssl pkcs8 -topk8 -inform pem -nocrypt -in sp.rsa_key -outform pem -out sp.pem”.
IDP Single Sign on Service Binding	Defines the SAML protocol binding to be used when returning the response message. TeamForge supports "urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect” binding only.
IDP Single Sign on Logout Service Binding	Defines the SAML protocol binding to be used when returning the response message. TeamForge supports "urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect” binding only.
IDP Certificate Finger Print	You can use the fingerprint instead of using the entire X509 certificate.
IDP Certificate Finger Print Algorithm	If an IdP fingerprint is provided, then the fingerprint algorithm is required to let the toolkit know which algorithm is used. **Possible values:** sha1 (default value), sha256, sha384.
Use Strict Mode	Values are *True* and *False*. TeamForge rejects the unsigned or unencrypted messages, if the strict mode is set to *True*.
Debug	This is used to set the log level to debug. Values are *True* and *False*.
Logout Name ID Encrypted	This indicates that the Name ID of the logout response sent by the Service Provider will be encrypted. Values are *True* and *False*.
Sign Authentication Request	This indicates whether the AuthnRequest message sent by the Service Provider is signed. The Metadata of the SP provides this information. Values are *True* and *False*.
Sign Logout Request	This indicates whether the logout request messages sent by the Service Provider is signed. Values are *True* and *False*.
Sign Logout Response	This indicates whether the logout response sent by this Service Provider is signed. Values are *True* and *False*.
Sign Messages	This indicates whether the messages are to be signed or not. Values are *True* and *False*.
Sign Assertions	This indicates whether the response, logout request, and logout response elements received by the SP need to be signed or not. Values are *True* and *False*.
Encrypt Assertions	This indicates whether the assertions received by the Service Provider need to be encrypted or not. Values are *True* and *False*.
Need Name ID	This indicates whether the Name ID is required or not in the SAML response. Values are *True* and *False*.
Name ID Encrypted	This indicates whether the Name ID received by the Service Provider need to be encrypted or not. Values are *True* and *False*.
Sign Metadata	This indicates whether the SP Metadata need to be signed or not. Values are *True* (sign using SP private key) and *False* (or null to not to sign).

Authentication Context	Defines the authentication context of the Service Provider. If no value is provided, then no authentication context will be sent in the AuthnRequest. Set the value as “urn:oasis:names:tc:SAML:2.0:ac:classes: urn:oasis: names:tc:SAML:2.0:ac:classes:Password”
Authentication Context Comparison	This allows the authentication context comparison parameter to be set. Default value is exact.
Validate XML	This indicates whether the Service Provider will validate all received XMLs. **Note:**To validate the XML, the Use Strict Mode to set to ‘strict’ and \`wantXMLValidation\` to be set to *True*.
Signature Algorithm	This indicates the algorithm that the toolkit will use during signing process. Some of the options: • http://www.w3.org/2000/09/xmldsig#dsa-sha1 • http://www.w3.org/2001/04/xmldsig-more#rsa-sha256 • http://www.w3.org/2001/04/xmldsig-more#rsa-sha384 • http://www.w3.org/2001/04/xmldsig-more#rsa-sha512
Reject Unsolicited Response To	This indicates where to send the rejected unsolicited response.
Compress Request	This indicates whether the request need to be compressed or not. Values are *True* and *False*.
Compress Response	This indicates whether the response need to be compressed or not. Values are *True* and *False*.
Technical Contact Name	This indicates the contact name of the Technical person at Service Provider’s end.
Technical Contact Email	This indicates the email id of the Technical person at Service Provider’s end.
Organization Name	This indicates the organization name at Service Provider’s end.
Organization Display Name	This indicates the organization’s display name at Service Provider’s end.
Organization URL	This indicates the URL of the organization at Service Provider’s end.
Username Attribute	This indicates the username attribute of the IdP.
Email Attribute	This indicates the email attribute of the IdP.
User Display Name Attribute	This indicates the display name attribute of the user.
Map Email to Username	This indicates whether the username need to be mapped to user’s email id or not. Values are *True* and *False*.

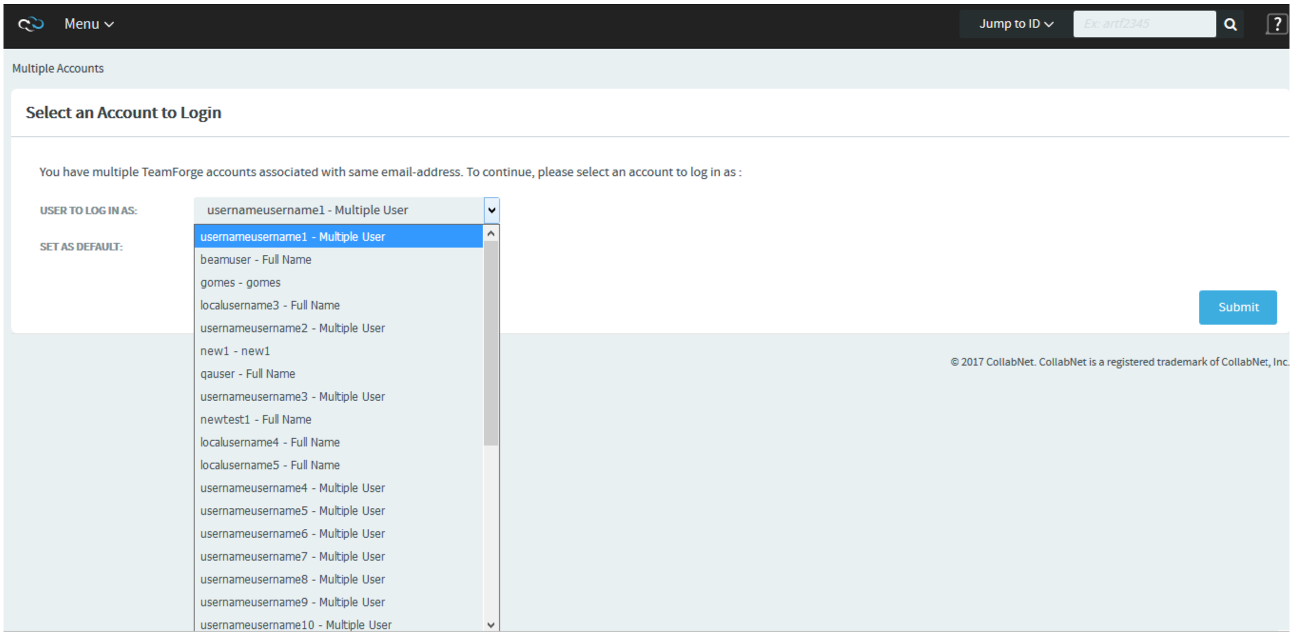
8. Click **Test Connection** to verify whether the integration works properly with the IdP configured in this page.

9. Click **Get SP Metadata** to obtain the Service Provider's metadata.

10. Click **Save** to save the configuration.

Intermediate Login Page for Multiple User Accounts

In a SAML enabled and SAML+LDAP enabled environment, if you have multiple user accounts for the same email address, you will be redirected to an intermediate login page before the third party IdP for authentication. In this intermediate login page, you can see the list of accounts associated with your email address. Select one from this list and set it as default account so that you would log on with this account every time you are authenticated via the third party IdP. After you have set a default account, you would not see the intermediate login page the next time you are authenticated via SAML.


Menu
Jump to ID
Ex: artf2345
?
Multiple Accounts
usernameusername1 - Multiple User
beamuser - Full Name
gomes - gomes
localusername3 - Full Name
usernameusername2 - Multiple User
new1 - new1
qauser - Full Name
usernameusername3 - Multiple User
newtest1 - Full Name
localusername4 - Full Name
localusername5 - Full Name
usernameusername4 - Multiple User
usernameusername5 - Multiple User
usernameusername6 - Multiple User
usernameusername7 - Multiple User
usernameusername8 - Multiple User
usernameusername9 - Multiple User
usernameusername10 - Multiple User
Submit

Select an Account to Login

You have multiple TeamForge accounts associated with same email-address. To continue, please select an account to log in as :

USER TO LOG IN AS: usernameusername1 - Multiple User

SET AS DEFAULT:

© 2017 CollabNet. CollabNet is a registered trademark of CollabNet, Inc.

If you want to change or reset the default user account at any point in time, you can do so from the **My Settings > Edit User Information** page.

Direct Login to TeamForge

In a SAML-enabled TeamForge environment, TeamForge administrators are provided with a direct login URL which can be used to log on to TeamForge without any intermediaries such as an IdP whenever things go wrong (due to some changes in SAML IdP endpoints) and if TeamForge is not accessible.

You can also use this direct login to fix any issues with SAML configuration. The direct login URL for local TeamForge administrators can be found in the TeamForge SAML configuration UI. TeamForge Administrators are advised to bookmark or keep this URL handy.

> **NOTE:** Other users can use the ‘Forgot Password’ link at the login page to set the local TeamForge password that should authenticate them to carry out any other user activities based on their RBAC permissions.

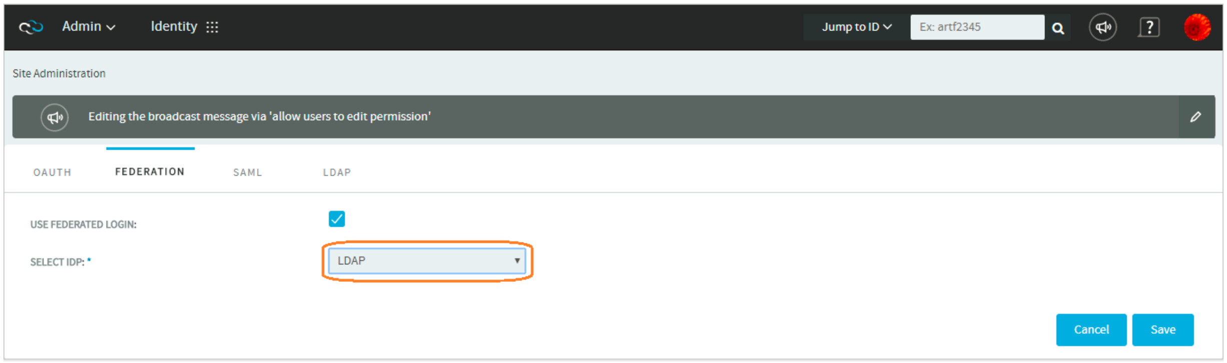
TeamForge supports integration with LDAP. Once integrated with LDAP servers, TeamForge can use LDAP credentials for user authentication.

LDAP (Lightweight Directory Access Protocol) is an application layer protocol that works on top of the TCP/IP stack and accesses your directory service providers such as Active Directory for providing user authentication. For more information, see RFC2251 - Light-weight Directory Access Protocol (v3).

Enable LDAP as an IdP

This section walks you through the steps to enable LDAP as an IdP in TeamForge.

1. Log on to TeamForge as a Site Administrator.
2. Select **My Workspace > Admin**.
3. Select **Projects > Identity**.
4. Select the **Federation** tab.
5. Select the **Use Federated Login** check box and select *LDAP* as the IdP from the drop-down list.
6. Click **Save**.



Site Administration

Editing the broadcast message via 'allow users to edit permission'

USE FEDERATED LOGIN: ☑

SELECT IDP: * LDAP

TeamForge-LDAP Authentication–Single LDAP Server Setup

In this section, you can see the configuration required for setting up TeamForge for authentication using a single LDAP server.

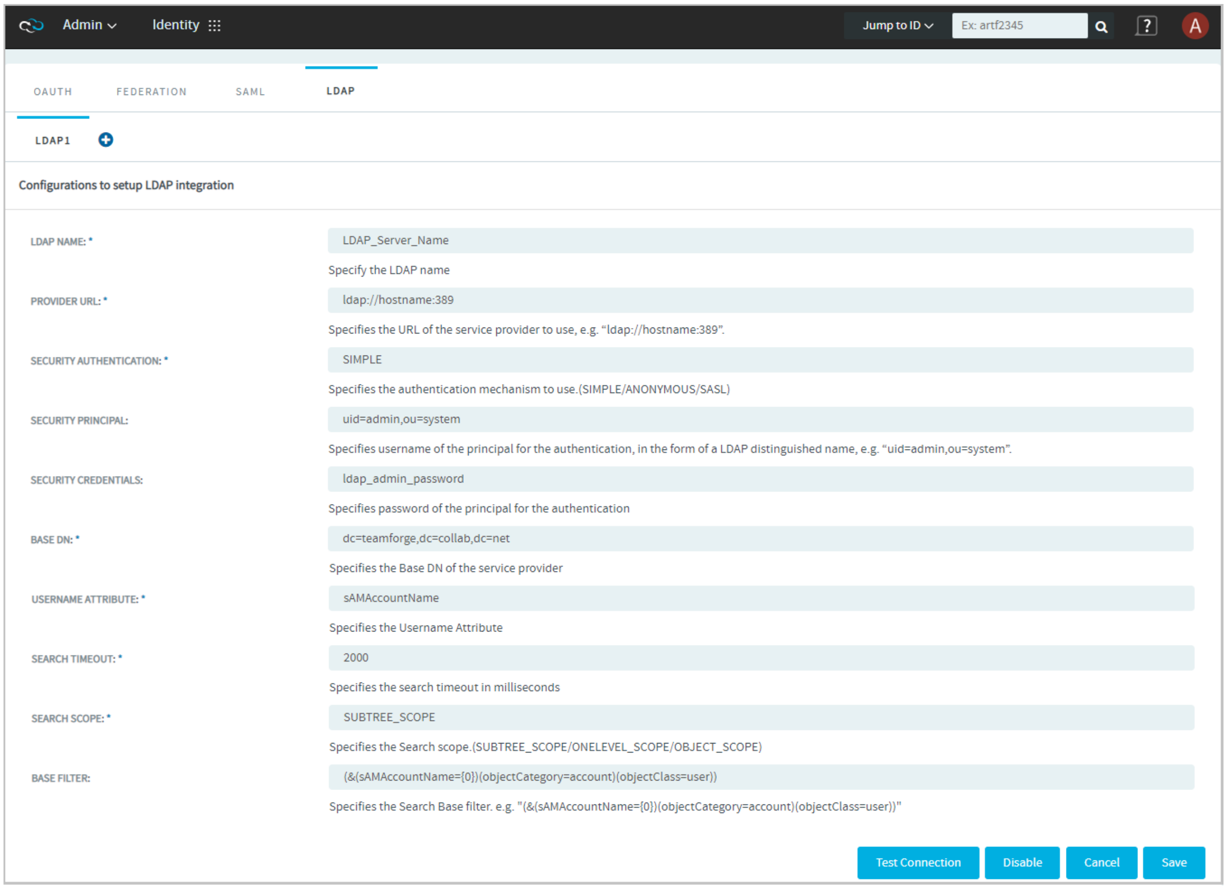
Before You Begin

- Once you have your LDAP server set up, you must configure the following `site-options.conf` tokens in TeamForge before integrating TeamForge with an LDAP server. Use your discretion and configure these tokens to suit your site's requirements.
    - USE_EXTERNAL_USER_AUTHENTICATION=true
    - APPROVE_NEW_USER_ACCOUNTS=false
    - REQUIRE_PASSWORD_SECURITY=false
    - LINUX_USERNAME_MODE_ENABLED=true
    - MINIMUM_PASSWORD_LENGTH=0
    - PASSWORD_REQUIRES_MIXED_CASE=false
    - PASSWORD_REQUIRES_NON_ALPHANUM=false
    - PASSWORD_REQUIRES_NUMBER=false
    - REQUIRE_USER_PASSWORD_CHANGE=false
- In addition to the above tokens, configure the ALLOW DATABASE AUTHENTICATION IF LDAP IS ENABLED parameter. To select this check box, select **My Workspace > Admin** and select **Projects > System Tools > Configure Application**. This parameter is listed in the **External Authentication** section. Select the ALLOW DATABASE AUTHENTICATION IF LDAP IS ENABLED check box to have LDAP credentials stored in TeamForge and have users authenticated via TeamForge every time a user logs in. This helps improve performance by optimizing the number of authentication calls between the TeamForge and LDAP servers.

  > **IMPORTANT:** Selecting this option is mandatory for sites with internally managed CVS servers.

- If you have enabled database authentication, LDAP user credentials are stored when users login for the first time and continue to login using the locally stored LDAP credentials. However, you can restrict such indefinite usage of the stored LDAP credentials and force user re-authentication at regular intervals by setting up this configuration parameter. For example, setting a value of 24 would force user re-authentication (by the LDAP server) every 24 hours. For more information, see FORCE RE-AUTHENTICATION WITH LDAP SERVER.

1. Log on to TeamForge as a Site Administrator.
2. Select **My Workspace > Admin**.
3. Select **Projects > Identity**.
4. Select the **LDAP** tab. This tab lets you configure the TeamForge-LDAP integration.



Configurations to setup LDAP integration

LDAP NAME: * LDAP_Server_Name
Specify the LDAP name

PROVIDER URL: * ldap://hostname:389
Specifies the URL of the service provider to use, e.g. "ldap://hostname:389".

SECURITY AUTHENTICATION: * SIMPLE
Specifies the authentication mechanism to use.(SIMPLE/ANONYMOUS/SASL)

SECURITY PRINCIPAL: uid=admin,ou=system
Specifies username of the principal for the authentication, in the form of a LDAP distinguished name, e.g. "uid=admin,ou=system".

SECURITY CREDENTIALS: ldap_admin_password
Specifies password of the principal for the authentication

BASE DN: * dc=teamforge,dc=collab,dc=net
Specifies the Base DN of the service provider

USERNAME ATTRIBUTE: * sAMAccountName
Specifies the Username Attribute

SEARCH TIMEOUT: * 2000
Specifies the search timeout in milliseconds

SEARCH SCOPE: * SUBTREE_SCOPE
Specifies the Search scope.(SUBTREE_SCOPE/ONELEVEL_SCOPE/OBJECT_SCOPE)

BASE FILTER: (&(sAMAccountName={0})(objectCategory=account)(objectClass=user))
Specifies the Search Base filter. e.g. "(&(sAMAccountName={0})(objectCategory=account)(objectClass=user))"

Fields	Description
LDAP Name	Descriptive name for each LDAP configuration set.
PROVIDER URL	The string that encapsulates the IP address and port of a directory server.
SECURITY AUTHENTICATION	The authentication method used to bind to the LDAP server. There are 3 types of security authentication in LDAP:

	• **Anonymous** - When a client sends a LDAP request without binding, then it is called an “anonymous client”. • **Simple** - In this type of authentication, the LDAP server sends the fully qualified DN (Distinguished Name) and the clear text password of the client. • **SASL** - SASL (Simple Authentication and Security Layer) authentication provides a challenge response protocol to exchange data between the client and server for the authentication and establishment of security layer to carry out further communication. **NOTE:** TeamForge supports only one of the authentication methods, which is **Simple**.
SECURITY PRINCIPAL	The distinguished name of the user to authenticate. Example: "uid=admin,ou=accounts"
SECURITY CREDENTIALS	The password or other security credentials of the user to authenticate.
NOTE: Select the `<<token_name>>` check box in the Configure Your Site’s Settings page to mandate the use of Security Principal and Security Credentials when a LDAP user tries to log on to TeamForge for the first time.	
BASE DN	The base distinguished name from where a server will search for users. This is a sequence of related distinguished names connected by commas and with the format "attribute=value". Example: dc=help,dc=collab,dc=net
USERNAME ATTRIBUTE	Attribute name to be used to match the username provided in the UI. Example: sAMAccountName (for Active Directory). **NOTE:** Please contact LDAP administrator for more information.

SEARCH TIMEOUT	The read timeout in milliseconds for an LDAP operation. This is used to control the LDAP request made by a client in a timely manner, so that the client need not wait for a long time for the server to respond. For example, if the search timeout value is 5000 milliseconds, the LDAP service provider can abort the read timeout if the server does not respond within this 5 seconds.
SEARCH SCOPE	The starting point of an LDAP search and the depth from the base DN to the levels until which the search should occur. There are three types of search scope in an LDAP search: • **OBJECT_SCOPE:** This limits the search scope only to the base object or base DN. • **ONELEVEL_SCOPE:** This enables search only up to the immediate children objects under the base DN in a search tree. • **SUBTREE_SCOPE:** This searches the entire subtree including the base DN. TeamForge recommends this as the default search scope in its LDAP configuration.
BASE FILTER	The group DN in which the users are member of. Sets the LDAP default search filter for the users to search and load all users from the database of active user accounts belonging to a specific OU (organizational unit) provided in the search filter. This is an optional field. **Example value**: *(&(sAMAccountName={0})(objectCategory=account)(objectClass=user))*

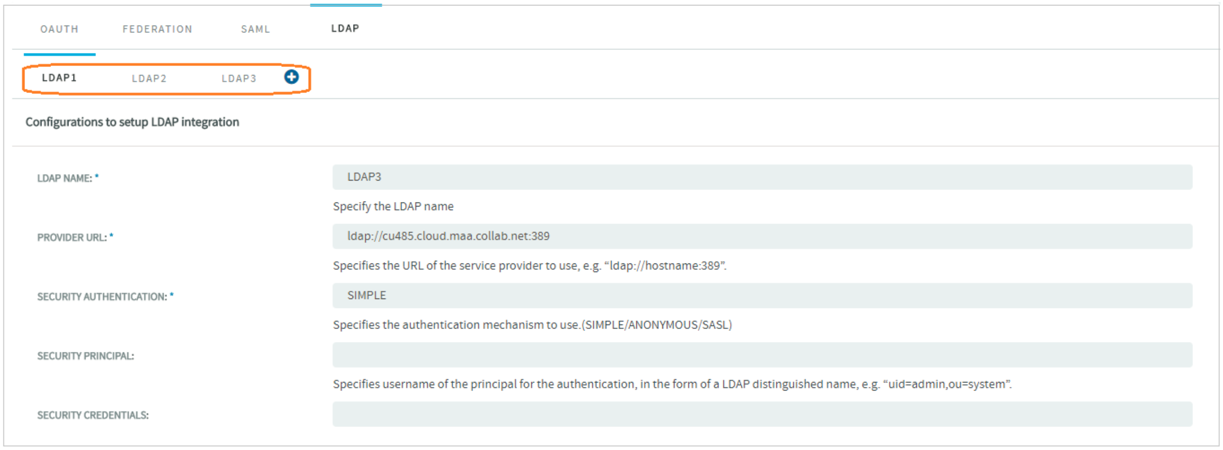
5. Click **Save.**

TeamForge-LDAP Authentication–Multiple LDAP Servers Setup

You can configure multiple LDAP servers for authentication with TeamForge 18.1 and later.

1. Log on to TeamForge as a Site Administrator.
2. Select **My Workspace > Admin**.
3. Select **Projects > System Tools > Configure Application**.

4. Set the number of LDAP servers in the LDAP CONFIGURATIONS MAXIMUM LIMIT parameter (in **External Authentication** section).
5. Select **Projects > Identity**.
6. Select the **LDAP** tab.
7. Add as many LDAP servers as required (click the ⊕ icon) and configure individual LDAP servers as discussed in Set up TeamForge for Single LDAP Server Authentication.



Configurations to setup LDAP integration

LDAP NAME: * LDAP3

Specify the LDAP name

PROVIDER URL: * ldap://cu485.cloud.maa.collab.net:389

Specifies the URL of the service provider to use, e.g. "ldap://hostname:389".

SECURITY AUTHENTICATION: * SIMPLE

Specifies the authentication mechanism to use.(SIMPLE/ANONYMOUS/SASL)

SECURITY PRINCIPAL:

Specifies username of the principal for the authentication, in the form of a LDAP distinguished name, e.g. "uid=admin,ou=system".

SECURITY CREDENTIALS:

8. Click **Save**.

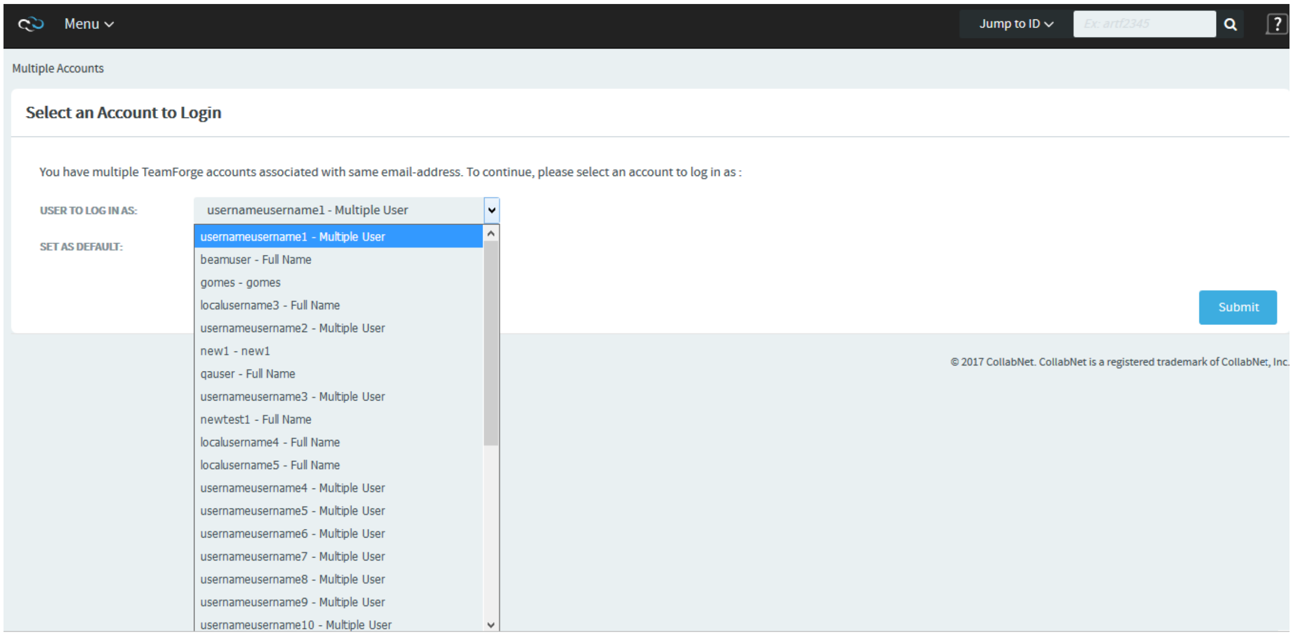
By setting up the SAML+LDAP IdP, TeamForge users can reap the benefits of both SAML and LDAP authentication mechanisms in a unified manner. With SAML+LDAP authentication, while SAML enables TeamForge users to access web applications, the LDAP authentication supports user authentication required for CLI applications. For example, if a user performs a source code commit in CVS/Git/SVN repository, the user can get authenticated via LDAP.

To set up SAML+LDAP authentication, you must set up the SAML and LDAP configurations as discussed later in this topic and then select **SAML+LDAP** as the IdP.

One E-mail Address–Multiple User Accounts: Intermediate Login Page

In a SAML enabled and SAML+LDAP enabled environment, if you have multiple user accounts for the same email address, you will be redirected to an intermediate login page before the third party IdP for authentication. In this intermediate login page, you can see the list of accounts associated with your email address. Select one of the user accounts from the list, which would be the default user account used for authentication by the third party IdP for your subsequent logins.

After you have set a default account, you are not taken to the intermediate login page the next time you log on. If you want to change or reset the default user account at any point in time, you can do so from the **My Settings > Edit User Information** page.



Multiple Accounts

Select an Account to Login

You have multiple TeamForge accounts associated with same email-address. To continue, please select an account to log in as :

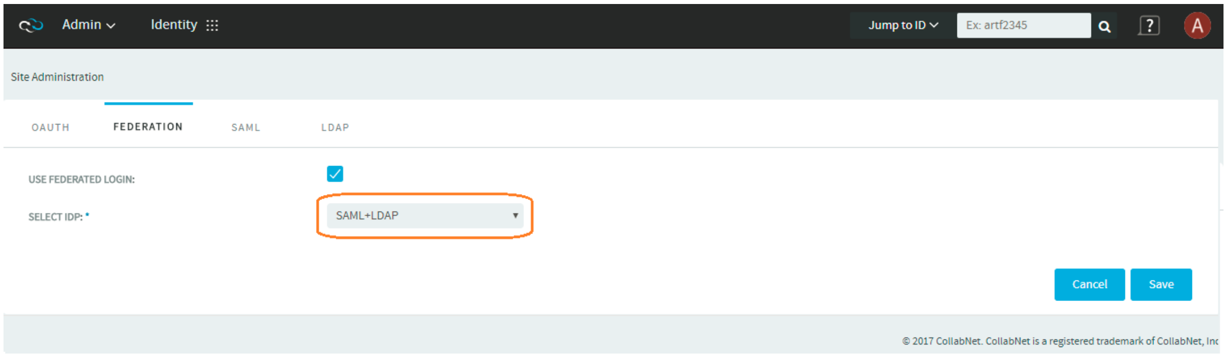
USER TO LOG IN AS: usernameusername1 - Multiple User

SET AS DEFAULT:

- usernameusername1 - Multiple User
- beamuser - Full Name
- gomes - gomes
- localusername3 - Full Name
- usernameusername2 - Multiple User
- new1 - new1
- qauser - Full Name
- usernameusername3 - Multiple User
- newtest1 - Full Name
- localusername4 - Full Name
- localusername5 - Full Name
- usernameusername4 - Multiple User
- usernameusername5 - Multiple User
- usernameusername6 - Multiple User
- usernameusername7 - Multiple User
- usernameusername8 - Multiple User
- usernameusername9 - Multiple User
- usernameusername10 - Multiple User

© 2017 CollabNet. CollabNet is a registered trademark of CollabNet, Inc.

1. Set up SAML and LDAP integrations. See:
   - Set up SAML
   - Set up LDAP
2. Log on to TeamForge as a Site Administrator.
3. Select **My Workspace > Admin**.
4. Select **Projects > Identity**.
5. Select the **Federation** tab.
6. Select the **Use Federated Login** check box and select **SAML+LDAP** as the IdP from the drop-down list.



Site Administration

USE FEDERATED LOGIN: ☑

SELECT IDP: * SAML+LDAP

© 2017 CollabNet. CollabNet is a registered trademark of CollabNet, Inc

7. Click **Save**.

Follow these steps to convert your TeamForge installation to authenticate against your corporate OpenLDAP server.

Set up LDAP Integration for TeamForge

Follow these steps to convert your TeamForge installation to authenticate against your corporate OpenLDAP server.

> **NOTE:** Refer to the Installation Requirements for TeamForge for the supported OpenLDAP versions.

1. Stop TeamForge.

   ```
   teamforge stop
   ```
2. Edit the `site-options.conf` file.
   1. Enable TeamForge to use LDAP authentication by editing the `site-options.conf` file, for example, edit `/opt/collabnet/teamforge/etc/site-options.conf` file. Under **External User Authentication**, uncomment the line that follows and change its value to `true`.

      ```
      USE_EXTERNAL_USER_AUTHENTICATION=true
      ```
   2. Configure the `site-options` tokens.

      > **NOTE:** The values specified for the following tokens are only for illustration purpose.

      ```
      EXTERNAL_AUTHENTICATION_TYPE=ldap
      LDAP_DN_PREFIX=cn=
      LDAP_DN_SUFFIX=,cn=Users,dc=testldap,dc=qa,dc=collab,dc=net
      LDAP_SERVER_URL=ldap://testldap.qa.collab.net:3268
      ```
3. Provision services.

   ```
   teamforge provision
   ```

   TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.

Turn off LDAP Authentication

During some maintenance operations, such as upgrades, you may need to turn off LDAP authentication.

1. Open the `site-options.conf` file, the master configuration file that controls your TeamForge site.

   ```
   vi /opt/collabnet/teamforge/etc/site-options.conf
   ```

> **NOTE:** `vi` is an example. Any *nix text editor will work.

2. In the `site-options.conf` file, comment out these variables:

   - USE_EXTERNAL_USER_AUTHENTICATION
   - LOGIN_CONFIG_XML_FILE
   - MINIMUM_PASSWORD_LENGTH

3. Recreate the runtime environment.

   ```
   ./install.sh -V -r -d /opt/collabnet/teamforge
   ```

4. Review the variables you have changed, then save the `site-options.conf` file.

Authenticate Users with LDAP Using Auth Manager

Use the Auth Manager to effectively manage and synchronize user profiles with LDAP.

There are a few limitations to the conventional method used to enable LDAP authentication. For example, when an external authentication source is configured using the conventional method, the template XML file needs to be copied and edited manually. The manual intervention may lead to ambiguities and it makes the process error-prone.

TeamForge makes the entire external authentication process easier with the Auth Manager add-on. It provides users the ability to create multiple profiles, manage profiles, and perform LDAP synchronization.

The Auth Manager allows you to:

- import login-config.xml file.
- maintain multiple profiles in LDAP servers.
- manage each profile individually.
- activate a profile without recreating runtime.
- store files in the Integration Data Service (IDS) that survives an upgrade.

Install Auth Manager

The Auth Manager add-on provides customers with a central authentication service the ability to integrate TeamForge with external authentication services such as LDAP, Active Directory, and Kerberos.

The Auth Manager TeamForge add-on is available as an RPM file that you have to download and install. Contact CollabNet Support for more information.

1. Log on to TeamForge as a root user.
2. Extract the RPM file. Extracting creates the add-on directory at `/opt/collabnet/teamforge/add-ons`
3. Navigate to the new add-ons directory.

   ```
   cd /opt/collabnet/teamforge/add-ons/ctf_authentication_manager
   ```

4. Install the Auth Manager:

   ```
   ./install
   ```

5. Choose to synchronize with LDAP for user data or make use of the user provided data, as required. For example, if you want to:
   - create a user profile quickly, use the data available in LDAP by enabling LDAP sync and running `hide.sh` script. It displays only the **Re-type password** field to the user.
   - create a user profile using the data provided by the user, disable LDAP and run the `show.sh` script. It displays all the fields that you expect the user to fill in and requires site administrator's approval. This is fairly a time-consuming process.
6. Set up your site's master configuration file.

   ```
   vim /opt/collabnet/teamforge/etc/site-options.conf
   ```

   1. Set *APPROVE_NEW_USER_ACCOUNTS* token as `false`.
      - **Hide fields** - To skip the approval process and create an user profile with the data available in LDAP, you have to enable LDAP Sync and run the `hide.sh` script after installation. It conceals all the fields on the **Create New User** page except the **Re-type password** field.

- **Show fields** - To get the data from the user and not through the LDAP sync, you have to disable LDAP Sync run the `show.sh` script after installation. It shows all fields including **full name**, **email**, **locale string**, and **license type** on the **Create New User** page.

2. Set REQUIRE_PASSWORD_SECURITY as `false`.
3. Set PASSWORD_REQUIRES_NUMBER as `false`.
4. Set PASSWORD_REQUIRES_NON_ALPHANUM as `false`.
5. Set USE_EXTERNAL_USER_AUTHENTICATION as `true`.
6. Set REQUIRE_USER_PASSWORD_CHANGE as `false`.
7. Set MINIMUM_PASSWORD_LENGTH as `0`.
8. Set PASSWORD_REQUIRES_MIXED_CASE as `false`.

7. Protect Auth Manager with SSL, if preferred. Click here for more details.
8. Provision services.

```
teamforge provision
```

TeamForge 18.1 installer expects the system locale to be `LANG=en_US.UTF-8`. TeamForge create runtime (`teamforge provision`) fails otherwise.

9. Start TeamForge.

```
teamforge start
```

To ensure that the installation has been completed successfully and the external authentication functionality works do the following:

✓ Login to the TeamForge through UI as an admin user and check if the add-on is appearing as Auth Manager in the project navigation bar. Also, for fresh installation, an active **Default TeamForgeDatabase** profile appears under **Manage Existing Profiles**, by default, with the green status indicator.

✓ Alternatively, in the CLI, scrutinize the log files, for example, `/opt/collabnet/teamforge/log/apps/server.log`.

Configure Auth Manager in TeamForge

You can configure Auth Manager as a linked application in TeamForge.

As a result of this configuration, the Auth Manager appears in TeamForge's project navigation bar as a linked application.

1. Log on to the TeamForge as a site administrator and go to the look project.
2. Click **PROJECT ADMIN** from the **Project Home** menu.
3. On the **Project Admin** Menu, click **Project Toolbar**. Then click **LINKED APPLICATIONS**. A list of all currently linked applications in the project is displayed.
4. Click **Create**.
5. On the **Create Linked Application** page, enter the name for the linked application as **AUTH MANAGER**.
6. Enter the URL: `http://<host>/authenticationManager`.
7. Enable Single Sign On (SSO) to allow TeamForge users to automatically log into the Auth Manager.
8. Click **Save**. The application is linked to the TeamForge and the it appears as **AUTH MANAGER** in the project navigation bar.

Manage Authentication Profiles

TeamForge utilizes its own database to validate the user name and password and also supports external authentication sources such as LDAP, Active Directory, Kerberos and Master Password.

Create a User Profile

You can create a profile that determines the authentication method and settings in TeamForge.

You need to have at least one user profile or more in active status before configuring the external authentication.

1. Log on to the TeamForge as a site administrator and go to the **look** project.
2. From the project navigation bar, click **AUTH MANAGER**.

3. From the **Main Menu** pane on the left, click **Create Profile**.

> **NOTE:** In the **Manage Existing Profiles** page, if you do not find the desired one in the list of existing authentication profiles, you can click **New Profile** and proceed.

4. From the drop-down menu, select the type of the new authentication profile.

    - **LDAP** - It uses the user name and password provided by the user to bind to the LDAP. If the bind is successful, the user is authenticated. This is a simple method of authentication. Click here for more information.

    - **LdapExtended** - It uses a service account to bind to the LDAP. Customizable filters are used to bind with the user and to validate authentication. Click here for more information.

        > **TIP:** Use this module if the users are spread over LDAP or when a group membership is required to access TeamForge.

        > **IMPORTANT:** You can use only the **LdapExtended** profile as a source for LDAP Sync.

    - **Active Directory** - It uses the user data configured through Microsoft's Active Directory. This is a simple method of authentication.

    - **Kerberos** - It uses MIT KRB5 authentication. Contact your network admin for the host configuration settings.

5. Set the Jboss flag that determines the behaviour of the control flag with multiple login-modules.

    - **Sufficient** - The login-module is not required to succeed. If it does succeed, control immediately returns to the application (authentication does not proceed down the login-module list). If it fails, authentication continues down the login-module list.

    - **Optional** - The login-module is not required to succeed. If it succeeds or fails, authentication still continues to proceed down the login-module list.

    - **Required** - The login-module is required to succeed. If it succeeds or fails, authentication still continues to proceed down the login-module list.

6. Enter the value for each module property listed for the chosen profile type.

7. Click **Create**. The confirmation message, *The authentication profiles have been imported. Activate the profiles to apply to TeamForge authentication*, appears.

> **NOTE:** The newly created profile is listed under Authentication Profiles in the Manage Existing Profile page. It is now inactive and the status indicator is yellow. You must activate the newly created user profile.

> **TIP:** Before you create any profiles using Auth Manager, you may see an inactive auto-imported **TeamForgeDatabase** profile appearing under **Authentication Profiles**. It is recommended to delete the **Auto-imported UsernamePasswordInDatabaseLoginModule** after creating and activating your first profile. Because the subsequent login and authentication request pass only through the active profile(s).

Configure Master Password

With the Site Administrative privileges, you can generate a master password and use it along with any user name registered with the TeamForge.

You can create an authentication profile for master password users in Auth Manager. Impersonation is not supported in TeamForge but the master password feature facilitates any user in TeamForge, irrespective of the roles, to login as another user if they have a valid master password.

The Site Admin needs to generate the password through scripts and configure the authentication user profile using Auth Manager. It is important to have this configured for a web-based SSO system.

1. On the commad-line interface, login to TeamForge as a root user to generate the master password.

2. Navigate to the new add-ons directory.

   ```
   cd /opt/collabnet/teamforge/add-ons/ctf_authentication_manager
   ```

3. Set the master password.

   ```
   ./mpasswd.sh
   ```

4. Enter the password and re-enter when prompted for confirmation.

> **NOTE:** Ensure that the password is masked while entering it. Also keep the master password confidential and share it with authenticated users on demand. It is a good practice to keep changing the master password frequently.

5. Log on to the TeamForge as a site administrator and go to the **look** project.

6. From the project navigation bar, click **AUTH MANAGER**.

7. From the **Main Menu** pane on the left, click **Create Profile**.

> **NOTE:** In the **Manage Existing Profiles** page, if you do not find the desired one in the list of existing authentication profiles, you can click **New Profile** and proceed.

8. On the **Create Authentication Profile** page, select **MasterPassword** from the drop-down list.

9. Enter an appropriate name for the new MasterPassword user profile.

10. Set the Jboss flag that determines the behaviour of the control flag with multiple login-modules.

    - **Sufficient** - The login-module is not required to succeed. If it does succeed, control immediately returns to the application (authentication does not proceed down the login-module list). If it fails, authentication continues down the login-module list.

    - **Optional** - The login-module is not required to succeed. If it succeeds or fails, authentication still continues to proceed down the login-module list.

    - **Required** - The login-module is required to succeed. If it succeeds or fails, authentication still continues to proceed down the login-module list.

11. Click **Create**. The confirmation message, *The authentication profiles have been imported. Activate the profiles to apply to TeamForge authentication*, appears.

> **NOTE:** The newly created profile is listed under **Authentication Profiles** in the **Manage Existing Profile** page. It is now inactive and the status indicator is yellow. You must activate the newly created user profile.

> **TIP:** Before you create any profiles using Auth Manager, you may see an inactive auto-imported **TeamForgeDatabase** profile appearing under **Authentication Profiles**. It is recommended to

> delete the **Auto-imported UsernamePasswordInDatabaseLoginModule** after creating and activating your first profile. Because the subsequent login and authentication request pass only through the active profile(s).

Upgrade a Profile

You can upgrade a legacy authentication profile by importing the corresponding configuration file for authentication.

You can upgrade an authentication profile that was used in TeamForge 6.2 or older versions. For each profile, you have to import and save the respective login-config.xml file. It imports all the user profiles automatically into the `standalone.xml` file which is later converted into `standalone-full.xml`. The `login-config.xml` and `standalone-full.xml` files are placed at the same location.

1. Log on to the TeamForge as a site administrator and go to the **look** project.
2. From the project navigation bar, click **AUTH MANAGER**.
3. From the **Main Menu** pane on the left, click **Upgrade Legacy Config**.
4. Select an option to upload your existing authentication configuration.
    1. **Detected Configuration Files** - The list of files detected in the default location.
    2. **Specify Server-based Configuration File** - Specify the location of the login-config.xml file in the TeamForge Server.
    3. **Upload Existing Configuration File** - Specify the legacy login-config.xml file saved in the local.
5. Click **Save**. The confirmation message, *The authentication profiles have been imported. Activate the profiles to apply to TeamForge authentication*, appears.

> **NOTE:** The newly created profile is listed under **Authentication Profiles** in the **Manage Existing Profile** page. It is now inactive and the status indicator is yellow. You must activate the newly created user profile.

Reorder Profiles

You can set the order in which a user profile needs to be considered for authentication. On the **Manage Existing Profiles** page, you can manually reorder the list of authentication profiles.

For example, consider LDAP server ‘A’ that has 100 users profiles and LDAP server ‘B’ with just 15 users profiles. To reduce the network traffic effectively, you may arrange authentication profiles in such a away that the authentication requests pass through ‘A’ first and then through ‘B’. When you enable reorder option, you can manually move the profiles back and forth within the list.

1. Log on to the TeamForge as a site administrator and go to the **look** project.
2. From the project navigation bar, click **AUTH MANAGER**.
3. From the **Main Menu** pane on the left, click **Manage Existing Profiles**.
4. Click **Enable Reorder**. You are now able to drag and drop profiles within the list.
5. Reorder the list as required and click **Save Order**.

Activate a Profile

To pass authentication requests through a profile, you must change the status of a profile to active. The status indicator is green for active profiles in .

By default, the status of a newly created profile is inactive. A profile creation process in is considered complete when you change the status from inactive to active. Activated profiles are automatically listed in the `standalone-full.xml` file.

You can activate a newly created profile or an existing profile that is currently inactive. The profiles that are listed in **Manage Existing Profiles** page with yellow status indicators are all inactive or deactivated. You can have multiple profiles in the active status and reorder them within the list, if required.

> **TIP:** Before you create any profiles using Auth Manager, you may see an inactive auto-imported **TeamForgeDatabase** profile appearing under **Authentication Profiles**. It is recommended to delete the **Auto-imported UsernamePasswordInDatabaseLoginModule** after creating and activating your first profile. Because the subsequent login and authentication request pass only through the active profile(s).

1. Log on to the TeamForge as a site administrator and go to the **look** project.
2. From the project navigation bar, click **AUTH MANAGER**.
3. From the **Main Menu** pane on the left, click **Manage Existing Profiles**.
4. To activate all the profiles listed in the **Manage Existing Profiles** page atonce, click **Activate All**.
5. To activate a particular profile in the **Manage Existing Profiles** page, click the profile name that needs to be activated. The profile details are displayed.

> **NOTE:** Ensure that the current status of the profile is inactive and marked yellow.

6. Click **Activate**.

> **NOTE:** This button is visible only if the current status of the profile is inactive; scroll down to see this.

7. On the confirmation window, click **OK** to proceed. The profile is activated now and the status turns green.

Deactivate a Profile

At any point in time, you can deactivate an active profile in . The status indicator is yellow for inactive profiles in .

You may decide to deactivate a profile when you encounter issues with LDAP. If you are half way through the configuration setup and you do not want the profile to be acive for user authentication, you can deactivate it. Or If you want to troubleshoot and identify if the issue encounted prevails in the particluar LDAP server, you can deactivate the authentication profile.

The profiles that are listed in **Manage Existing Profiles** page with yellow status indicators are all inactive or deactivated. By default, the status of a newly created profile is inactive. You can have multiple profiles in the inactive status and reorder them within the list, if required. Deactivated profiles are automatically removed from the `standalone-full.xml` file.

> **IMPORTANT:** The deactivated profile remains in the Integrated Data Space (IDS) and you can reterive the same profile by activating it. The deactivated profile is permanently removed from the application only when it is deleted.

1. Log on to the TeamForge as a site administrator and go to the **look** project.
2. From the project navigation bar, click **AUTH MANAGER**.
3. From the **Main Menu** pane on the left, click **Manage Existing Profiles**.
4. To deactivate all the profiles listed in the **Manage Existing Profiles** page, click **Deactivate All**.
5. To deactivate a particular profile in the **Manage Existing Profiles** page, click the profile name that needs to be deactivated. The profile details are displayed.

> **NOTE:** Ensure that the current status of the profile is active and marked green.

6. Click **Deactivate**.

> **NOTE:** This button is visible only if the current status of the profile is active; scroll down to see this.

7. On the confirmation window, click **OK** to proceed. The profile is deactivated now and the status turns yellow.

Delete a Profile

You can delete an inactive profile from the permanently.

You can delete only deactivated profiles. If you do not want to have a profile in your authentication profile list, you need to deactivate it first and then delete it totally from the system.

> **IMPORTANT:** A deactivated profile remains in the Integrated Data space (IDS) until you perform deletion. To remove it completely from the system, delete the deactivated profile.

1. Log on to the TeamForge as a site administrator and go to the **look** project.
2. From the project navigation bar, click **AUTH MANAGER**.
3. From the **Main Menu** pane on the left, click **Manage Existing Profiles**.
4. Click the profile that you want to delete. The profile details are displayed.
5. Ensure that the current status of the profile is inactive.

> **NOTE:** The profile status is marked yellow if inactive.

6. Click **Delete**.

> **NOTE:** This button is visible only if the current status of the profile is inactive.

7. On the confirmation window, click **OK** to proceed. The profile has been deleted.

Configure LDAP Sync

Perform LDAP Sync to update TeamForge with the user data available in the LDAP server. You can use an extended LDAP profile as a source for LDAP sync.

LDAP Sync, basically searches in the LDAP server for the user data configured in a login-module. Then it fetches the user data to the TeamForge and performs synchronization. You have options to selectively include login-modules for the LDAP Sync. For example, you have two LDAP accounts, out of which only one needs to be considered for LDAP sync. You need to turn off the LDAP account that you do not want to include in LDAP Sync.

The extended LDAP profile needs to have Bind DN, Bind Credentials, Base Filter, and Base DN values for the synchronization. The Base DN, which is not available in other simple LDAP authentications, makes the synchronization possible in **ExtendedLDAP** profile.

> **IMPORTANT:** Provide the site admin user name and obfuscated site admin password explicitly in `/opt/collabnet/teamforge/var/etc/soap-provider.properties` and do not provide ADMIN account credentials.

1. Log on to the TeamForge as a site administrator and go to the look project.
2. From the project navigation bar, click **AUTH MANAGER**.
3. From the **Main Menu** pane on the left, click **Manage Existing Profiles**.
4. Select your desired profile that needs to have the LDAP Sync enabled.
5. Click **Edit**.
6. Select *true* from the **Use As LDAP Sync Source** drop-down list to consider the selected profile for the synchronization.
7. From the **Main Menu** pane on the left, click **LDAP Sync**.
8. Click **Global Settings** and enable LDAP sync by selecting *true* from the drop-down list.
9. Click **Group Sync Settings** and do the following to synchronize groups:
    1. Select *true* form the **Enable Group Sync** drop-down list to enable the LDAP/AD group synchronization.
    2. Enter appropriate value in **Group Job Cron Interval** to set the time interval for running group synchronization. For example, click here.

3. Enter the attribute in **Group Search Filter Expression**, to specify LDAP/AD expression for testing.

4. Enter the search text with '*' at the end in **Group Search Filter Arguments** to synchronize with groups within the search results that have a particular prefix.

    > **TIP:** To include all the groups in synchronization, just enter '*'.

    Example: `CTF*`, `TF*` or `*`

10. Click **User and User Data Sync Settings** and do the following to synchronize user status and attributes:

    1. Select *true* form the **Enable User Data Sync** drop-down list to enable synchronization of user data.

    2. Enter appropriate value in the **User Data Sync Cron Interval** to set the time interval for group synchronizations. For examples, click here.

    3. Enter the search text with '*' at the beginning in **User Search Filter Arguments** to synchronize with groups within the search results that have a particular prefix.

        > **TIP:** To include all the groups in synchronization, just enter '*'.

        Example: `CTF*`, `TF*` or `*`

    4. Select *true* from the **Enable LDAP Status Sync** drop-down. It enables the LDAP Sync for the user status. If the user account is disabled in LDAP/AD, it flags and then disables the user account in TeamForge.

    5. Enter the number of days in **User Grace Period** beyond which the user account is disabled in TeamForge, if the user is not existing in LDAP.

    6. Select *false* from the **Enable User Disable Action** drop-down list. It specifies if the user who has not logged into TeamForge for the specified period, needs to be disabled in TeamForge.

    7. Enter the number of days in **User Disable Interval (Days)** beyond which the user account is disabled in TeamForge, if the user has not logged into TeamForge for the specified period.

8. Select *true* from the **Enable User Delete Action** drop-down list to enable the deletion of a disabled user account based on the delete interval.

9. Enter the number of days in the **User Delete Interval (Days)** drop-down list beyond which the user account is deleted from TeamForge. It is mandatory to have **Enable LDAP Status Sync** or **Enable User Disable Action** enabled.

   - When **Enable LDAP Status Sync** is *true*, users that do not exist in Active Directory are deleted after the ‘delete’ and ‘grace’ intervals from the flagged date. However, users that are existing in Active Directory cannot be disabled or deleted.

   - When **Enable User Disable Action** is *true*, users are deleted, irrespective of their existence in Active Directory, after the ‘disable’ and ‘delete’ intervals. It is calculated from the last login date.

   - When both **Enable LDAP Status Sync** and **Enable User Disable Action** are set to *true*, users that are not existing in Active Directory are deleted after the ‘grace’ and ‘delete’ intervals from the flagged date. And the users that are existing in the Active Directory are deleted after the ‘disable’ and ‘delete’ intervals from the last login date.

10. Select *true* from the **Enable User Membership Action** drop-down list to include user’s LDAP/AD group membership in the synchronization. This needs to be enabled when the **Enable Group Sync** is set to *true*.

11. Select *true* from the **Allow User Re-enabling** drop-down list to re-enable LDAP active users that have pending or disabled status in TeamForge.

12. Enter the user names in **Excluded Usernames** to skip the respective users accounts during synchronization.

13. Enter the email address in **Default Email Address** that needs to be associated with the TeamForge user account. It is used as the default email address when the email field is found null or empty.

14. Enter the number of user batches in **Split Users for LDAP Sync** to perform the synchronization.

    > **NOTE:** The user batch number entered splits the existing number of users into batches and then performs synchronization. Entering ‘0’ or ‘1’ considers all the existing users for the synchronization. Whereas entering ‘7’ splits the existing users into seven batches and completes synchronization on the seventh run.

11. Click to expand **Mail and Reporting Settings** and do the following:

    1. Select *true* in **Enable User Email Reports** drop-down list to enable the email notification and reporting.
    2. Enter the SMTP host to mail through. If you are using the TeamForge James mail server, enter `localhost` in **Mail Transport Host**.
    3. Enter the email address of the recipient in **Mail To** that is usually a TeamForge Discussion Forum or a Tracker.
    4. Enter the email address of the sender in **Mail From** that is usually a TeamForge Discussion Forum or Tracker. The sender should be a valid TeamForge user.
    5. Enter any optional email address in **Mail CC** that needs a carbon copy of the email.
    6. Enter the subject line for the email in **Mail Subject**.
    7. Enter the user name in **Mail Username** that authenticates connection to the SMTP servers, if required. Its optional to fill in this field.
    8. Enter the password in **Mail Password** that is required to connect to SMTP servers, if required. Its optional to fill in this field.

12. To save and apply all the changes you made to the profile, click **Save**.
13. To run the LDAP Sync once (on an ad hoc basis), click **Run Once**.
14. Click **Stop** and **Start** buttons to reinitiate the synchronization service.

Configure Selective Sync

The Selective sync, otherwise known as single user sync, is similar to the LDAP sync that is performed only for a particular user and not for the users in the entire directory server. The Site Admin can perform synchronization for a selective user on the three selective attributes: full name, email, and organization.

The single user sync is helpful when you encounter inconsistencies or discrepancies in the LDAP behaviour. Some of the common scenarios include network delay, LDAP disconnectivity, and failure of authenticating a valid user.

In cases where the LDAP is not completely reliable, you can consider performing selective sync. For example:

- While performing LDAP sync for a huge site that has thousands of users, there are chances that a handful of users are not synchronized and found missing in the report. Just for users that are missed out, you can synchronize the full name, email, and organization of individual users.
- When the default (dummy) email is used to create a profile, the user may not receive any email notifications. In this case, the Site Admin can update only the email address of the user and perform single user sync. It saves the time spent on LDAP sync that scans through the entire directory server for a single user's email update.

> **NOTE:** You cannot perform selective sync on users that are disabled and deleted. Single user sync is not applicable for the 'ADMIN' user.

1. Log on to the TeamForge as a site administrator and go to the **look** project.
2. From the project navigation bar, click **AUTH MANAGER**.
3. From the **Main Menu** pane on the left, click **LDAP Sync**.
4. Select *true* form the **Enable Group Sync** drop-down list under **Global Settings > Group Sync Settings**. It enables the LDAP/AD group synchronization.
5. Select *true* form the **Enable User Data Sync** drop-down list under **Global Settings > User** and **User Data Sync Settings**. It enables synchronization of the user data.
6. To save and apply all the changes you made to the profile, click **Save**.
7. Go to **My Workspace > Admin**.
8. On the site administration navigation bar, click **USERS**.
9. On the *USERS* tab, click the name of the user whose account you want to edit.
10. On the **User Details** page, click **EDIT**.
11. On the **Edit User Information** page, make your changes to the full name of the user, email address and organization.
12. Click **Sync User Info** to synchronize the modified user data with the authentication source.

> **IMPORTANT:** Selective Sync is limited to full name, email, and organization of the user. If you have issues with other attributes of a user profile, try LDAP sync through **Auth Manager**.

Uninstall Auth Manager

You can remove the Auth Manager completely from the TeamForge. Unlike other linked applications in TeamForge, you need not run the default installer to uninstall the athis add-on.

Before uninstaling, use the Auth Manager GUI to remove all the authentication profiles except the TeamForgeDatabase profile.

1. Login to TeamForge as a root user.
2. Navigate to the new add-ons directory.

   ```
   cd /opt/collabnet/teamforge/add-ons/ctf_authentication_manager
   ```

3. Uninstall the **AUTH MANAGER**.

   ```
   ./uninstall.sh
   ```

4. Provision services.

   ```
   teamforge provision
   ```

   TeamForge 18.1 installer expects the system locale to be LANG=en_US.UTF-8. TeamForge “provision” command fails otherwise.

5. Start TeamForge.

   ```
   teamforge start
   ```

Field Description for Auth Manager

The credential store and identity manager properties that are required to create an authentication profile in the Auth Manager are described here.

Field	Description
allowEmptyPasswords	A flag indicating if empty (length 0) passwords should be passed to the LDAP server. An empty password is treated as an anonymous login by some LDAP servers and this may not be a desirable feature. Set this to false to reject empty passwords or true to have the LDAP server validate the empty password. The default is `true`.
baseCtxDN	It defines the fixed DN of the context to search for user roles. Consider that this is not the Distinguished Name of where the actual roles are located but the DN of where the objects containing the user roles are located (that is, for active directory, this is the DN with the user account).
baseFilter	It defines the search filter used to locate the context of the user to authenticate. The input username/userDN as obtained from the login module callback substitutes the {0} expression.

	This substitution behavior comes from the standard DirContext?.search(Name, String, Object[], SearchControls? cons) method. An common example search filter is "(uid={0}).
bindCredential	It defines the bindDN password. The password can be encrypted if the jaasSecurityDomain is specified.
bindDN	It defines the DN used to bind to the LDAP server. This is a DN with read/search permissions to the defined baseCtxDN and rolesCtxDN.
java.naming.factory.initial	The classname of the InitialContextFactory implementation. This defaults to the Sun LDAP provider implementation com.sun.jndi.ldap.LdapCtxFactory.
java.naming.provider.url	This property specifies the host name and port of the DNS server used by the initial DNS context, as well the initial context's domain name.
java.naming.referral	It indicates the service providers how to handle referrals.
java.naming.security.authentication	Specifies the authentication mechanism and the security level to use. This defaults to simple java.security.krb5.kdc It defines the host name on which the Active Directory server runs.
java.security.krb5.realm	It defines the Microsoft domain in which the Active Directory server runs.
principalDNPrefix	A prefix to add to the username to form the user distinguished name.
principalDNSuffix	A suffix to add to the username when forming the user distinguished name. This is useful if you prompt a user for a username and you don't want the user to have to enter the fully distinguished name.
roleAttributeID	It defines the role attribute of the context that corresponds to the name of the role. If the roleAttributeIsDN property is set to true, this property is the DN of the context to query for the roleNameAttributeID attribute. If the roleAttributeIsDN property is set to false, this property is the attribute name of the role name.
roleAttributeIsDN	It defines if the role attribute contains the fully distinguished name of a role object or the role name. If false, the role name is taken from the value of the user's role attribute. If true, the role attribute represents the distinguished name of a role object. The role name is taken from the value of the roleNameAttributeId attribute of the corresponding object. In certain directory schemas (for example, Microsoft Active Directory), role (group)attributes in the user object are stored as DNs to role objects and not as simple names. In such case, set this property to true. The default value of this property is false.
roleFilter	It defines a search filter used to locate the roles associated with the authenticated user. The input username/userDN as obtained from the login module callback substitutes the {0} expression in the filter definition. The authenticated userDN substitutes the {1} in the filter definition. An example search filter that matches the input username is (member={0}). An alternative that matches the authenticated userDN is (member={1}). **If you omit the roleFilter attribute, the role search will use the UserDN as the DN to obtain the roleAttributeID value**.
roleNameAttributeID	It defines the role attribute of the context which corresponds to the name of the role. If the roleAttributeIsDN property is set to true, this property is used to find the name attribute of the role object. If the roleAttributeIsDN property is set to false, this property is ignored.
rolesCtxDN	The fixed DN of the context to search for user roles. Consider that this is not the Distinguished Name of where the actual roles are; rather, this is the DN of where the objects containing the user roles are (e.g. for active directory, this is the DN where the user account is).
searchScope	Sets the search scope to one of the following (the default value is SUBTREE_SCOPE): • OBJECT_SCOPE - searches the named roles context only. • ONELEVEL_SCOPE - searches directly in the named roles context.

	• SUBTREE_SCOPE - searches only the object if the role context is not a DirContext?. If the roles context is a DirContext?, the subtree rooted at the named object and the named object itself are searched.
searchTimeLimit	It defines the timeout for the user and role searches in milliseconds (defaults to 10000, that is 10 seconds).

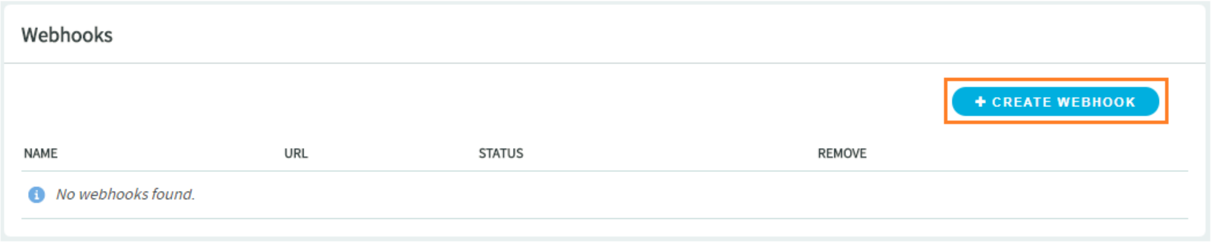
Webhooks can be configured for tracker artifacts and events are generated and posted to the webhook URL whenever an artifact is created, updated, moved, cloned and deleted.
Before You Begin

Keep the Webhook URL (to which the tracker artifact events would be sent) handy before you proceed with setting up Webhooks in TeamForge. You can set up Webhooks for Tracker artifacts.

Create a Webhook

1. Log on to TeamForge and select a project from the **My Workspace** menu.
2. Click **PROJECT ADMIN** from the **Project Home** menu.
3. Click **Webhooks** from the **Project Admin Menu**.

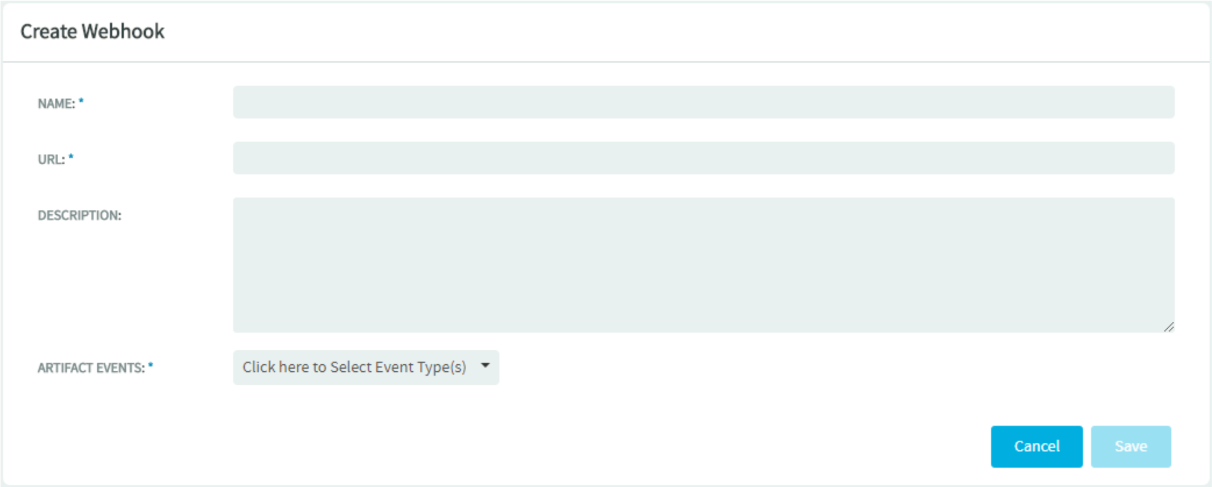
4. Click **Create Webhook** from the webhooks list page. The **Create Webhook** page appears.



Webhooks

NAME	URL	STATUS	REMOVE
No webhooks found.			

5. Enter a name for the webhook.



Create Webhook

NAME: *

URL: *

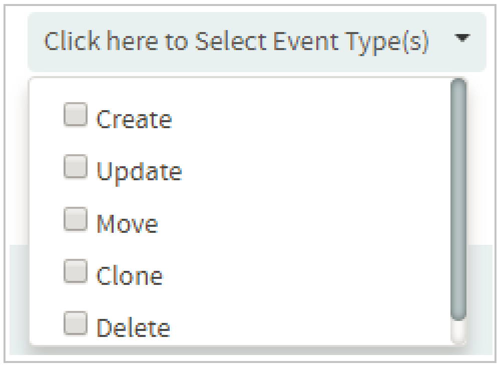
DESCRIPTION:

ARTIFACT EVENTS: * Click here to Select Event Type(s)

6. Type the webhook URL.

7. Type a description for the webhook.

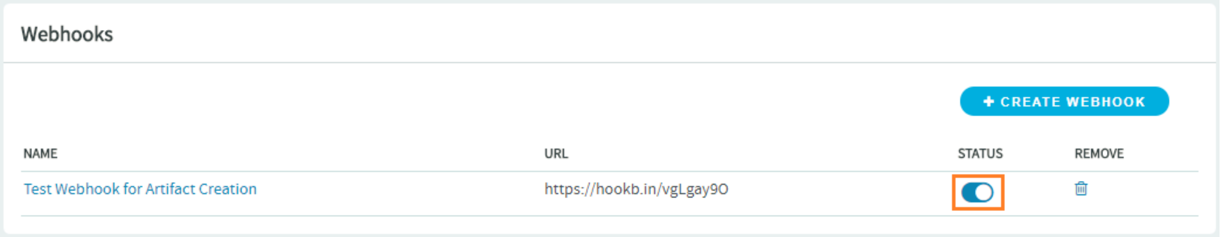
8. Select the desired events from the list of artifact events.



9. Click **Save**.

   You have successfully created a webhook for selected artifact events.

   By default, the status of the webhook is enabled. You can enable or disable the webhook using the **Status** toggle button.



NAME	URL	STATUS	REMOVE
Test Webhook for Artifact Creation	https://hookb.in/vgLgay9O	●	🗑

10. Repeat steps 4 through 9 to add more webhooks.

Update a Webhook

1. On the webhooks list page, click the webhook that you want to edit.

2. Make the desired changes on the **Edit Webhook** page.

3. Click **Save**.

Delete a Webhook

1. On the webhook list page, click the Delete icon of the webhook that you want to delete.

2. A confirmation message shows up. `Are you sure you want to remove the webhook from project?`

3. Click **OK** to delete.

Related Links

- Set up Webhooks for Repositories
- Set up Webhooks for Projects

Webhooks can be configured both at a project level or for select repositories. Once set up, SCM events such as commit and merge are published to the Webhooks for other applications to consume.

Keep the Webhook URL (of the application that consumes TeamForge SCM event information) handy before you proceed with setting up Webhooks in TeamForge. You can set up Webhooks for Git, Subversion and CVS repositories.

Create a Webhook

1. Log on to TeamForge and select a project from the **My Workspace** menu.

2. Click **SOURCE CODE** from the **Project Home** menu.

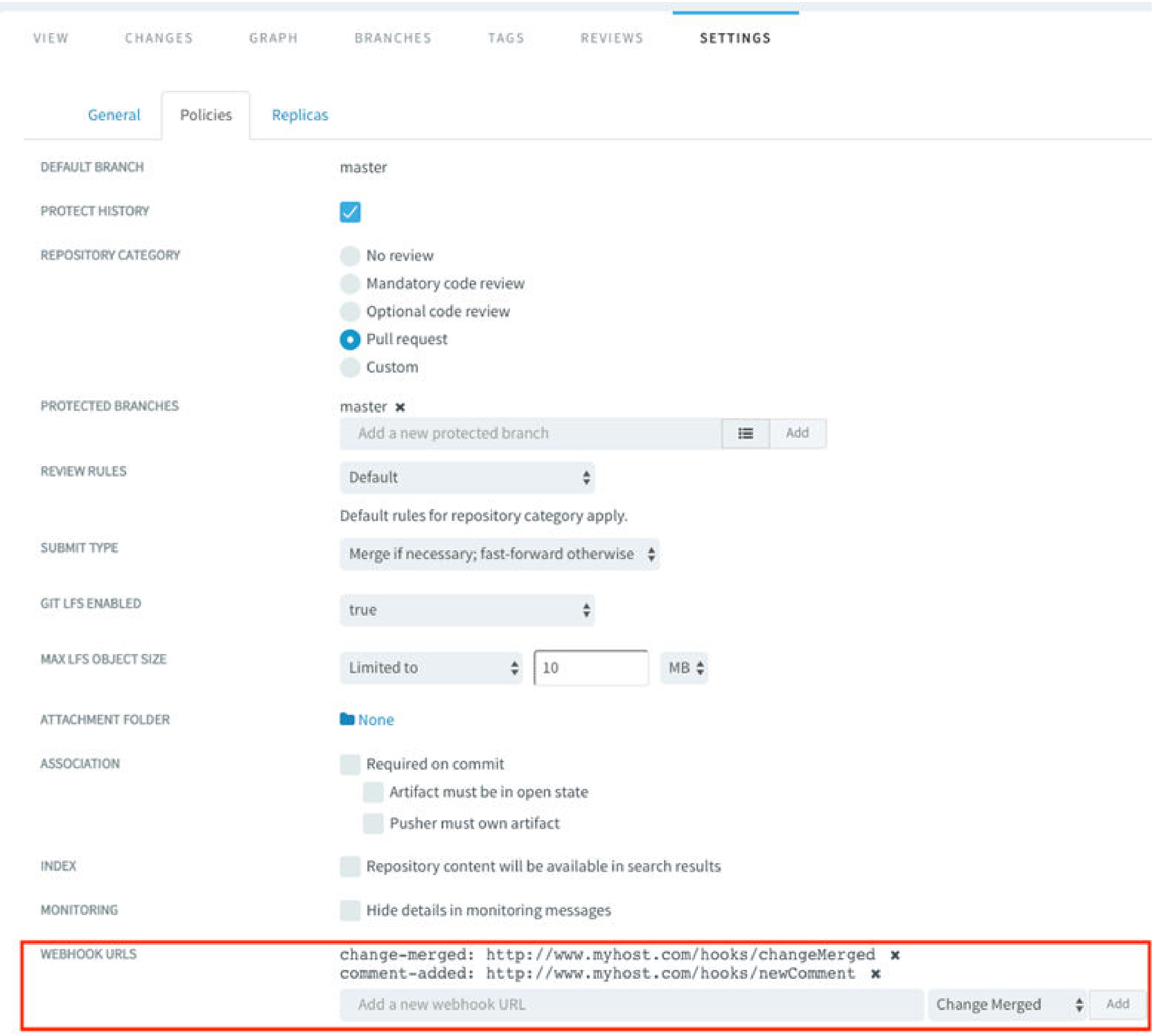
3. Select a repository and select the **SETTINGS** tab.

4. Select the **POLICIES** tab.

5. Type the Webhook URL in the **WEBHOOK URLS** field, select an event type from the drop-down list and click **Add**.

   You have successfully created a webhook for the repository.

6. Repeat steps 4 and 5 to add more webhooks, if required.



DEFAULT BRANCH: master

PROTECTED HISTORY: ☑

REPOSITORY CATEGORY:
○ No review
○ Mandatory code review
○ Optional code review
● Pull request
○ Custom

PROTECTED BRANCHES: master ✖
Add a new protected branch

REVIEW RULES: Default
Default rules for repository category apply.

SUBMIT TYPE: Merge if necessary; fast-forward otherwise

GIT LFS ENABLED: true

MAX LFS OBJECT SIZE: Limited to 10 MB

ATTACHMENT FOLDER: None

ASSOCIATION:
□ Required on commit
□ Artifact must be in open state
□ Pusher must own artifact

INDEX: □ Repository content will be available in search results

MONITORING: □ Hide details in monitoring messages

WEBHOOK URLS:

```
change-merged: http://www.myhost.com/hooks/changeMerged ✖
comment-added: http://www.myhost.com/hooks/newComment ✖
```

Add a new webhook URL | Change Merged

Related Links

- [Set up Webhooks for Tracker Artifacts]
- [Set up Webhooks for Projects]

Webhooks can be configured both at a project level or for select repositories. Once set up, SCM events such as commit and merge are published to the Webhooks for other applications to consume.

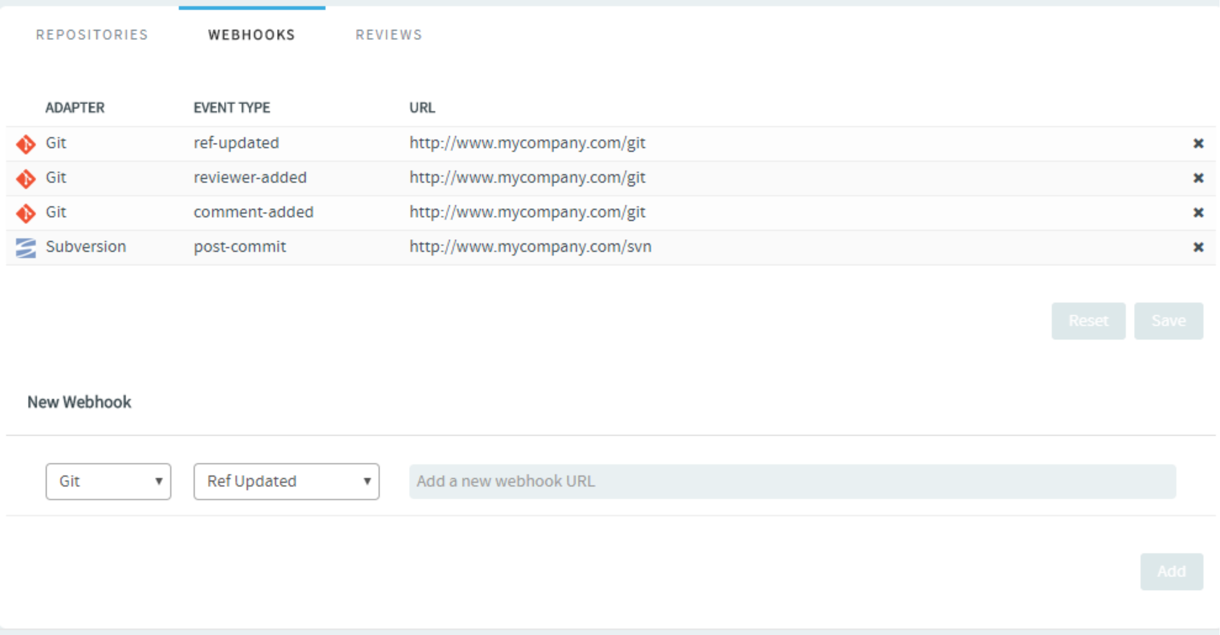
Keep the Webhook URL (of the application that consumes TeamForge SCM event information) handy before you proceed with setting up Webhooks in TeamForge. You can set up Webhooks for Git, Subversion and CVS repositories.

Create a Webhook

1. Log on to TeamForge and select a project from the **My Workspace** menu.

2. Click **SOURCE CODE** from the **Project Home** menu.

3. Select **WEBHOOKS** tab.



ADAPTER	EVENT TYPE	URL	
Git	ref-updated	http://www.mycompany.com/git	✖
Git	reviewer-added	http://www.mycompany.com/git	✖
Git	comment-added	http://www.mycompany.com/git	✖
Subversion	post-commit	http://www.mycompany.com/svn	✖

New Webhook

Git ▾ | Ref Updated ▾ | Add a new webhook URL

4. Select a repository type from the drop-down list (Git, Subversion or CVS) for which you want to create a Webhook.

5. Select an event type from the drop-down list.

6. Type the Webhook URL.

7. Click **Add**.

   You have successfully created a Webhook for all the repositories in the project of a particular SCM tool such as Git, Subversion and CVS.

8. Repeat steps 4 through 7 to add more webhooks, if required.

Related Links

- Set up Webhooks for Tracker Artifacts
- Set up Webhooks for Repositories

TeamForge provides you with the interrelated extension features to suit your organization's specific needs.

- Custom Event Handlers in TeamForge
- Add a Custom Event Handler to Your TeamForge Site
- Reference Information About Custom Event Handlers
- Authenticate Your Integrated Application with TeamForge
- Internationalize Your Integrated Application
- TeamForge SOAP API Reference
- Integrated Application References

You can create custom workflows in TeamForge using custom event handlers.

How TeamForge Custom Event Handlers Work?

The TeamForge custom event handling framework allows third-party event handlers to register for TeamForge-specific application events and notifies them whenever such an event occurs.

The event handling framework implements an extended flavor of the observer pattern. The TeamForge application events are triggered whenever a property of a TeamForge object (e.g. tracker item, discussion item, wiki page) has been changed or is going to be changed, if no event handler objects (i.e. blocks the event).

For example, you can block deletions of projects for all users, add a comment to a tracker item whenever an association has been modified, or design your own tracker workflow engine.

Writing custom event handlers requires at least some basic knowledge in the Java programming language or (if you use the examples shipped with this post) a script language that is installed on the TeamForge server, such as shell, Python, or Perl. For these instructions, we'll assume that you are familiar with basic programming techniques.

The event handler framework differentiates between two types of events:

- **Asynchronous** - If a handler registers for asynchronous events, it is informed that a change has just happened. The handler can decide to trigger further changes by calling TeamForge web services, but it cannot block the change because it has already happened.

Asynchronous event handlers are good for triggering system events, such as changing an artifact status or sending an email. See Using an Asynchronous Event Handler: Trigger Follow-up Events.

- **Synchronous** - If a handler has registered for synchronous events, it gets informed whenever a change has been anticipated by a user. It can examine the properties that should be changed and decide whether to accept the change or block it. A synchronous event handler cannot trigger further changes on the currently processed object, since other handlers in the event handler chain must also have the chance to block the anticipated change.

A synchronous event handler is the appropriate way to show an alert directly in the TeamForge UI, for example. See Using a synchronous event handler: Send event handler output to the TeamForge UI.

Technically, all event handlers have to be part of a Java archive (JAR) file with a TeamForge specific deployment descriptor that describes which events should be intercepted. This JAR file then has to be uploaded to the TeamForge application server. No restart is necessary, but the event handling cache has to be refreshed.

In practice, you can customize a TeamForge site's behavior without any knowledge of Java if you can write scripts in a language that can deal with environment variables, write to standard out/error (to control what will be displayed in the TeamForge UI as result of the handler's execution) and control the return code (to decide whether to block the event or not).

We will show you how to come up with your own custom event handlers based on two examples.

Event Handler Example - Comment on Associations

This event handler adds a comment to a tracker item whenever an association is added to or deleted from this tracker item.

This example illustrates how to intercept a specific event, trigger a follow-up action by calling TeamForge web services, and add a comment based on the formatting template which is specified as part of a property file.

The code for example one can be found here.

> **TIP:** You can treat a JAR file like a zip file: just extract it with your favorite zip program and have a look at the files that are part of it.

1. When you extract the JAR file you'll find a structure like this: `com/vasoftware/sf/plugin`. This directory contains the class file of your event handler. There may be additional directories containing Java class files. If you like to include Java libraries, you have to unpack their JAR files and add their class files (including directory structure) in the event handler JAR.

The `META-INF/event.xml` file contains the events and operations your handler class will intercept. It is also common to have additional files in the META-INF directory, such as property files to control the behavior of the event handler.

This example event handler is provided as-is (i.e. not supported as part of any TeamForge release). As with all event handlers, use it at your own risk. CollabNet cannot guarantee any SLAs on third-party code.

> **TIP:** If you want to change the behavior of custom event handlers at runtime (without redeploying the JAR file), you may want to look at the TeamForge integration data API. For the purpose of this example, we will stick with property files.

2. In the `event.xml` file, you will find some lines like this:

```
<event api="6.1" mode="asynchronous">
 <type>relationship</type>
 <operation>*</operation>
 <handler>com.vasoftware.sf.plugin.AsynchronousRelationshipEventListener</h
andler>
</event>
```

This tells the event handling framework that the class `AsynchronousRelationshipEventListener` is responsible for intercepting events of type “relationship” (aka associations) for every possible operation. The handler will be called after the event has happened (asynchronous mode) and the passed data structures will be compatible with the events format defined in TeamForge 6.0 (in other words, the event handler extends the EventHandler60 class). If you only want to intercept certain operations, you can specify those instead of the wildcard character (*). Supported operations are usually create, update, move and delete, but every event can define its own operations.

1. The `config.properties` file is used to control the formatting of the comment that gets added when an association has been modified. The initialize-method of the handler class (`AsynchronousRelationshipEventListener`) shows how property files can be parsed within a custom event handler.

2. By default, the user triggering the event is also the user executing the event handler. If you want to run your event handler with a different user account, specify it in the user element, like this:

```
<event api="6.1" mode="synchronous">
  <type>*</type>
  <operation>*</operation>
  <handler>com.vasoftware.sf.plugin.AsynchronousRelationshipEventListener</
```

```
handler>
  <user>foo</user>
</event>
```

> **NOTE:** Be careful with this option, because running code on behalf of a different user opens the door for all kind of exploits if you do not check the user's input properly. Also, this option will not allow you to access the original user's session id any more, but you can always create a new session with a super user account (credentials saved in a property file) if you have to.

Event Handler Example - Execute a Hook Script

When a TeamForge event arrives, this event handler looks to see whether there is a script in the TeamForge file system with the name/operation of the event, and then calls that script with all information from the event contained within environment variables.

This example illustrates how to intercept arbitrary TeamForge events, examine the event's properties, map them to system environment variables and call a script in the file system with a name corresponding to the intercepted event.

You can use this event handler to customize your TeamForge site's behavior without any knowledge of the Java programming language as long as you can write scripts in a language that can deal with environment variables, write to standard out/error (to influence what will be displayed in TeamForge's UI as result of the handler's execution) and influence the return code (to decide whether to block the event or not).

The code for this example can be found here.

> **TIP:** You can treat a JAR file like a zip file: just extract it with your favorite zip program and have a look at the files that are part of it.

1. When you extract the JAR file you'll find a structure like this: `com/collabnet/ctf/events`. This directory contains the class file of your event handler. There may be additional directories containing Java class files. If you like to include Java libraries, you have to unpack their JAR files and add their class files (including directory structure) in the event handler JAR.

The `META-INF/event.xml` file contains the events and operations your handler class will intercept. It is also common to have additional files in the META-INF directory, such as property files to control the behavior of the event handler.

This example event handler is provided as-is (i.e. not supported as part of any TeamForge release). As with all event handlers, use it at your own risk. CollabNet cannot guarantee any SLAs on third-party code.

TIP: If you want to change the behavior of custom event handlers at runtime (without redeploying the JAR file), you may want to look at the TeamForge integration data API. For the purpose of this example, we will stick with property files.

1. In the `event.xml` file, you will find some lines like this:

   ```
   <event api="6.1" mode="asynchronous">
       <type>*</type>
       <operation>*</operation>
       <handler>com.collabnet.ctf.events.AsynchronousHookScriptEventListener</
   handler>
   </event>

   <event api="6.1" mode="synchronous">
       <type>*</type>
       <operation>*</operation>
       <handler>com.collabnet.ctf.events.SynchronousHookScriptEventListener</h
   andler>
   </event>
   ```

   These lines tell TeamForge to register two event handlers, one asynchronous (`AsynchronousHookScriptEventListener`) and one synchronous (`SynchronousHookScriptEventListener`) for arbitrary events (wildcard *). It also tells TeamForge that the handler classes will extend the `EventHandler60` interface.

2. By default, the user triggering the event is also the user executing the event handler. If you want to run your event handler with a different user account, specify it in the user element, like this:

   ```
   <event api="6.1" mode="synchronous">
       <type>*</type>
       <operation>*</operation>
       <handler>com.collabnet.ctf.events.SynchronousHookScriptEventListener</h
   andler>
       <user>foo</user>
   </event>
   ```

   NOTE: Be careful with this option, because running code on behalf of a different user opens the door for all kind of exploits if you do not check the user’s input properly. Also, this option will not

> allow you to access the original user's session id any more, but you can always create a new session with a super user account (credentials saved in a property file) if you have to.

It is possible to register multiple handlers for different events, but you can also use one handler to intercept both synchronous and asynchronous events.

Event Handler Example: Hook Scripts

These sample hook scripts should give you an idea how custom event handlers can be written. Feel free to adjust them to your own needs.

> **NOTE:** These sample event handlers are not officially supported by CollabNet and must be used at your own risk.

- Hooks must be owned by `sf-admin` for security, and must have the executable bit set.
- To configure a site to prevent projects being deleted, we could create this file: `/opt/collabnet/teamforge/hooks/synchronous/project_delete`

```
#!/bin/sh
echo Sorry, projects cannot be deleted on this site 1>&2
exit 1
```

- To automatically create an initial directory structure in an SVN repository when the repository is created, you might create this file: `/opt/collabnet/teamforge/hooks/asynchronous/repository_create`

```
#!/bin/sh
/usr/bin/svn
mkdirhttp://localhost/svn/repos/${tf_original_RepositoryDirectory:9:999}/trun
k
http://localhost/svn/repos/${tf_original_RepositoryDirectory:9:999}/tags
http://localhost/svn/repos/${tf_original_RepositoryDirectory:9:999}/branches
-m "Inital Structure"
--username admin --password mypassword --non-interactive --no-auth-cache
exit 0
```

Extend EventHandler60 for Your TeamForge Event Handler Classes

All event handlers (synchronous and asynchronous) have to extend com.vasoftware.sf.events.EventHandler60.

This is how our example “Comment on associations” event handler does it:

```
package com.vasoftware.sf.plugin;
...
public class AsynchronousRelationshipEventListener extends EventHandler60 {
...
}
```

The folloiwng table lists the JAR file location for different versions of TeamForge that contains EventHandler60.

TeamForge Version	Jar File
6.1	/opt/collabnet/teamforge/dist/deployments/saturn.ear/sfevents.jar
6.1.1	/opt/collabnet/teamforge/dist/deployments/saturn.ear/lib/sfevents.jar
6.2	/opt/collabnet/teamforge/dist/deployments/saturn.ear/lib/sfevents.jar
6.2.0.1	/opt/collabnet/teamforge/dist/deployments/saturn.ear/lib/sfevents.jar
7.0	/opt/collabnet/teamforge/dist/deployments/saturn.ear/lib/sfevents.jar
7.1	/opt/collabnet/teamforge/dist/deployments/saturn.ear/lib/sfevents.jar
8.0	/opt/collabnet/teamforge/dist/deployments/saturn.ear/lib/events-api.jar
8.0.0.1	/opt/collabnet/teamforge/dist/deployments/saturn.ear/lib/events-api.jar
8.1	/opt/collabnet/teamforge/dist/deployments/saturn.ear/lib/events-api.jar
8.1.0.1	/opt/collabnet/teamforge/dist/deployments/saturn.ear/lib/events-api.jar
16.3	/opt/collabnet/teamforge/dist/deployments/lib/events-sdk.jar
16.7 and later	/opt/collabnet/teamforge/dist/deployments/lib/event.jar

- EventHandler60 has one abstract method you have to override:

  ```
   @Override
   public void processEvent() throws Exception {
   ...
  }
  ```

- This method does not take any parameters, so how do we access the relevant information?

There are some interesting methods of the base class (EventHandler60) you can call:

- **getSessionKey** - Returns a session id of the user who is going to (synchronous handler) / has triggered (asynchronous handler) the event we just intercepted. If you used the user element in event.xml, it will contain a session id for the user you specified there.
- **getEventContext** - Returns a data structure containing the event type, operation, project, comment and user name.
- **getOriginalData** - In case of a synchronous event handler, this will return a representation of the object the event is going to change. In case of an asynchronous event handler, this will return the representation of the object before it was changed by the event. The data structure used to represent the object is the same that would have been used in CollabNet's SOAP API.
- **getUpdatedData** - In case of a synchronous event handler, this will return a representation of the object how it will look after the event has happened (you can still block it). In case of an asynchronous event handler, this will return the representation of the object after it was changed by the event.

Let's assume a user wants to change the priority of a tracker item from 3 to 4. If you have registered a synchronous event handler, this one is triggered before the change can actually be performed. `getOriginalData` returns an `ArtifactSoapDO` object of the tracker item with the priority field set to 3. `getUpdatedData` contains an `ArtifactSoapDO` object of the tracker item with the priority field set to 4.

- If you block the event (by throwing an exception), the change does not happen and the user is presented with an error message. (See next section for how to influence this error message.)
- If you do not block the event (by just returning from the processEvent method), all registered asynchronous handlers are called. `getOriginalData` and `getUpdatedData` contain exactly the same objects as in the synchronous case. However, the semantic is different: They are no longer representing the current and anticipated state, but the previous and current state of the object in question.

- The following code snippet (taken from our "Hook script" event handler example) shows how to retrieve all information available to an event handler.

```
String type = getEventContext().getType();
String operation = getEventContext().getOperation();
String projectId = getEventContext().getProjectId();
String comment = getEventContext().getComment();
String userName = getEventContext().getUsername();
String originalDataClassName = getOriginalData().getClass().toString();
String updatedDataClassName = getUpdatedData().getClass().toString();
Object originalData = getOriginalData();
Object updatedData = getUpdatedData();
```

Using a Synchronous Event Handler: Send Event Handler Output to the TeamForge UI

When we have extracted all data available to the event handler, how do we interact with the user interface?

> **NOTE:** Only synchronous event handlers can directly communicate with the UI, because if the event has already happened (as it has, in the case of asynchronous handlers), the user who triggered the event may already have been logged out.

- You use three independent actions to interact with the TeamForge UI:

  - Add a success message to the UI that gets displayed as the result of the action just triggered by the user. This can be done by calling the `addSuccessMessage` method of the `EventHandler` base class (see SynchronousHookScriptEventListener.java of example two for details).

  - Add an error message to the UI that gets displayed as the result of the action just triggered by the user. This can be done by calling the `addErrorMessage` method of the `EventHandler` base class.

  - Block the event you intercepted. This can be done by throwing an exception in the `processEvent` method. The payload of your exception will be displayed in the UI.

  All three forms of UI feedback can be used in combination. For example, it is possible to display an error message even if you did not block the event, and it is possible to show many error and success messages together.

- What happens if the event in question was not triggered by a user logged into the TeamForge Web UI but by a client using the TeamForge web services?

In this case, error and success messages do not reach the SOAP client. However, the payload of the exception object thrown when the event was blocked is delivered as part of the SOAP fault element.

- While synchronous event handlers enable you to block events and/or to provide additional feedback to the currently logged in user, they should not be used to trigger follow-up actions (like changing TeamForge artifacts or interacting with external systems).

Remember that these handlers are running in the main TeamForge event loop and nothing else will happen until you return from the `processEvent` method, so return as fast as you can.

Using an Asynchronous Event Handler: Trigger Follow-up Events

Use an asynchronous event handler to communicate with TeamForge, external systems, processes or system resources.

> **IMPORTANT:** To avoid accidentally locking the main TeamForge event queue down (and essentially rendering the system unusable), use only asynchronous event handlers (not synchronous event handlers) to trigger events.

Interacting with TeamForge is done as you would do it if you had to write a Java program to interact with TeamForge using its web services API. The only difference is that you will connect to `localhost` (since your handler is running locally) and that you already have a valid session ID.

- You do not have to include the SOAP SDK classes in your event JAR file, because they are already in the TeamForge class path. This code snippet extracted from our association converter example ([Event Handler example: Comment on Associations][/ teamforgecustomeventhandlers.html#commentonassociations]) shows how to do it:

```
 ITrackerAppSoap trackerClient = (ITrackerAppSoap) ClientSoapStubFactory.getSo
apStub(
 ITrackerAppSoap.class, "http://localhost:8080");
 ...
 ArtifactSoapDO artifact = trackerClient.getArtifactData(getSessionKey(), orig
inId);
 trackerClient.setArtifactData(getSessionKey(), artifact, finalComment, null, n
ull, null);
```

> **TIP:** You may use external libraries in your event handler by placing their `.class` files into your event JAR file. The only tricky part is if TeamForge is using a different version of this library (which will take precedence). In this case, you would have to recompile your library with a different package namespace.

- Using the session key provided by the event handler is actually only going to work if the SOAP call you are using is not throwing an exception. The session ID passed into your handler is associated with an already running transaction that will be aborted if an exception is thrown as part of this session. Part of rolling back the transaction is rolling back the JVM's call stack which contains your event handling code, so you will not be able to catch the web services exception. If you like to to handle web service exceptions, you have to create your own session id by logging into TeamForge again by calling

ICollabNetSoap.login with some credentials stored as part of your handler. (You can store them in a property file in the META-INF directory.)

Best Practices for Working with Custom TeamForge Event Handlers

In general, watch out for deadlocks and favor asynchronous over synchronous event handlers.

Beware of Deadlocks

Having custom event handlers that modify other objects can be dangerous if there it is possible for that handler or another handler to chain in the opposite direction. An example of this is an event handler that updates a task when an associated artifact is updated and updates the artifact when the associated task is updated. It is possible for two users to modify each object at the same time causing the two event handlers to wait on each other. The task handler would have a lock on the task bean in the application server while the artifact handler would have a lock on the artifact bean. When the custom event handlers fired, they would wait for the locks to be released but since the two threads have the locks each other needs and are waiting on the opposite objects, a deadlock would occur.

Asynchronous is Safer

Custom event handlers will be the least worrisome when they are responsible for data validation or secondary object creation (or association creation). Object modification is possible but adds greater complexity due to the risks involved with locking multiple objects across many threads. If you are unsure, use asynchronous handlers to modify objects instead since the lock on the original object will be gone by the time the asynchronous handler is executed.

Calling and waiting for synchronous hooks currently doesnt have a timeout. As long as your synchronous hook is running, the whole TeamForge site will be blocked for all users accessing the site. Some events trigger other events. For example creating a project actually calls the create project hook, wiki page hooks, and so on. Badly written or slow hooks can cripple a site.

Write to a File

Write your diagnostics messages in a file and not on stdout/stderr, since TeamForge does not read from stdout/stderr before the script completes. In the case of synchronous hooks, this could lead to a situation where the script blocks because the pipes buffer between the script process and the TF process is completely filled.

No Cascading

Due to the nature of custom event handling, custom events cannot cascade. This means that if a custom event handler catches an event and creates an object that it or another custom event handler would normally process, the event bypasses the custom event handlers. This is to prevent looping and infinite object creation. While there are ways for event handlers to avoid this, it would be a fairly difficult task since all of your event handlers would have to use a circular event detection algorithm. Rather than adding that complexity, we just eliminated the possibility.

Event Handler Life Cycle

For every single call to the processEvent method, a new object of your class will be instantiated. A best practice to avoid costly reinitialization every time (remember that the TeamForge event loop thread is blocked while you are doing this) is to delegate all synchronization work to a method you always call in your constructor which checks a static variable whether the initialization has already been done and if not, just returns without any further action (code snippet from example one):

```
private static boolean initialized = false;
public  AsynchronousRelationshipEventListener() {
initialize();
}

private synchronized void initialize() {
if (initialized ) {
return;
}
initialized = true;
// proceed with initialization
...
}
```

Logging in into TeamForge and initializing network connections file resources are costly operations that should be handled in such a method instead of doing it all over again.

Event Spooling

While it is true that asynchronous handlers may consume considerably more time than synchronous ones, there is only one thread for those handlers, so events may queue up if you do expensive operations. A best practice is to capture the event in your asynchronous event handler, write all necessary information to the local file system (comparable to a mail spooling directory) and return. At the same time, you can have a separate application reading from the spooling directory. This way, you never get into a situation where you miss TeamForge events, or things queue up just because you run into a blocking operation.

Incremental Changes

The event handler parser is really picky on the exact format of your JAR file. A best practice is to base your work on an already existing event handler and then adapt it to your own needs by doing incremental changes while checking whether it still works.

Watch out for Loops

Your follow-up actions may trigger your handler to be called again. You have to protect your handler from an infinite update loop if that happens. A best practice is to add a check to your event handler to see whether the user initiating the event is the same user you are using to perform follow-up actions.

Roll Back Sparingly

Throwing an exception in a synchronous event handler blocks the intercepted event and rolls back the transaction associated with the change. Rolling back transactions also means that the data the user entered is not saved. If this happens accidentally due to a wrongly programmed event handler, it can be frustrating to your users, so make sure that you only throw exceptions in your handler code when you really want to enforce the rollback.

Catch Errors Generically

It is quite easy to miss an exception you did not expect (like a null pointer exception, parsing exception, time out exception, any other malfunction in your own code). A best practice is to introduce a generic catch block in your handler and only rethrow the exception if it was an intended exception (see SynchronousHookScriptEventListener):

```
} catch (Exception e) {
 if (!intendedException) {
 log.error("Exception occured: " + e.getMessage(), e);
 } else {
...
 throw e;
 }
 }
```

When you add an event handler to your TeamForge site, you can automatically react to system events in ways that help your site's project members or administrators.

An event handler is a program that watches for events on a TeamForge site and communicates them to another system. You can add your own event handlers to the set that are built into TeamForge.

For example, some of your site’s members may be using TeamForge alongside a legacy issue tracking system. You may want to write an event handler that listens to the `Artifact Create` event on your

TeamForge site and sends the details about any newly created artifact to the legacy system through a webservice.

1. Create your custom event handler and package it as a .jar file.
2. Check with your system administrator that the ENABLE_UI_FOR_CUSTOM_EVENT_HANDLERS token in the site configuration file is set to `true`.
3. Go to **My Workspace > Admin**.
4. Click **SYSTEM TOOLS** from the **Projects** menu.
5. Click **Customizations**.
6. Click **Create** and use the **Browse** control to locate your `.jar` file.
7. Click **Add**.

> **NOTE:** If the system reports `Error Parsing Event Jar File`, debug your event handler until the error message no longer appears.

Your `.jar` file is uploaded to your TeamForge site and the event cache is cleared. All the events you specified in your event handler are now captured and sent to the external web service.

Here is some stuff you may need to know to work with event handlers.

Create a New “event.xml” File for Your Custom Event Handler

Here is an example of an `event.xml` file. Use it to create a new event.xml file for your custom event handler in the META-INF directory of your jar file.

In this example, the CreateTestCaseWorkflow class will be invoked only on artifact creation and will run after the artifact creation is successfully committed. The EventTestListener will run for every single event and will run synchronously thereby having the opportunity to cancel the event.

```
<?xml version="1.0"?>
<!DOCTYPE event-handler
  PUBLIC "-//VA Software, Inc.//DTD Data Object 1.1//EN"
  "http://schema.vasoftware.com/sf/dtd/sf-event-handler_1_0.dtd">

<event-handler>
    <event api="6.1" mode="asynchronous">
       <type>artifact</type>
       <operation>create</operation>
       <user>admin</user>
  <handler>com.collabnet.ctf.events.CreateTestCaseWorkflow</handler>
    </event>
  <event api="6.1" mode="synchronous">
       <type>*↙type>
       <operation>*↙operation>
  <handler>com.collabnet.ctf.events.EventTestListener↙handler>
    ↙event>
↙event-handler>
```

DTD for Custom Event Handler

An event handler works by subscribing to system events and then responding when event occurs. You subscribe to events by placing an XML file in the custom event handler JAR that describes what events are being monitored and who is doing the monitoring.

Here is the current DTD:

```
<!ELEMENT event-handler (description?, event+)>

<!-- Optional description for the Event Handler -->
<!ELEMENT description (#PCDATA)>

<!ELEMENT event (type, operation, user?, handler)>

<!-- This is the API level that the event handler expects the ObjectSoapDO obj
ects to be marshaled for -->
<!ATTLIST event api (6.1|6.2) #REQUIRED>

<!--
    Whether the event handler should run synchronously (allows it to cancel th
e event) or asynchronously after the event
    has completed and is committed to the database.
-->
<!ATTLIST event mode (synchronous|asynchronous) #REQUIRED>

<!-- The object type being handled (e.g., artifact, task, document). '*' is al
l object types. -->
<!ELEMENT type (#PCDATA)>

<!-- This is the operation that is being listened for (e.g., create. delete. m
ove, update). '*' is all operations. -->
<!ELEMENT operation (#PCDATA)>

<!--
    If the event handler needs to run as a different user (ie., a site admin a
ccount) the username should be
    passed in this attribute.  If the username is not valid the event handler
will throw an abort exception.  That
    means that if the handler is processing synchronously the event will be ca
```

```
nceled and the user will get an
    exception message.
-->
<!ELEMENT user (#PCDATA)>

<!--
   Name of a fully qualified class to handle the event.
-->
<!ELEMENT handler (#PCDATA)>
```

To help third party developers write integrated applications, CollabNet provides an SDK.

At the heart of the SDK is the IntegratedAppSupport helper class. You use IntegratedAppSupport to authenticate integrated application requests with TeamForge, and provide context information for subsequent processing of forms and links within the application.

You can download the SDK from here.

You must call the IntegratedAppSupport class for every request made by the integrated application. This class takes HttpServletRequest and HttpServletResponse for each request and determines whether the user is already authenticated. It also provides project- and user-related information that can be used throughout the request.

> **TIP:** It is a good idea to store this as a ThreadLocal so that it can be used from anywhere in the application.

Here's what you need to do to call IntegratedAppSupport from the application:

- To create an object, pass in these three parameters:
    - TeamForge Base URL – this is where TeamForge is installed
    - Integrated application URL – this is where the application is installed
    - Application Name – the name of the application as defined in the integrated application's descriptor

  Typically these parameters are stored as web.xml initialization parameters so that they are passed to IntegratedAppSupport's constructor.

- When you have a validated IntegratedAppSupport object, store it in a ThreadLocal object so that it can be used in other places. For example, when constructing the blog URLs, we get IntegratedAppSupport from ThreadLocal and retrieve the IntegratedAppId from there. Once the object is created, another method called processRequest is called on it. This takes an HttpServletRequest and HttpServletResponse object that the servlet or a filter gets.

- For each request that enters your system and needs to be validated, create an IntegratedAppRequest object.

  Subsequently, you also need to call the processRequest method to set the required parameters internally.

Where you construct the IntegratedAppSupport and call ProcessRequest subsequently depends on the architecture of your application. We've provided two cases – for servlet filters and httpservlets.

- If your application has servlet filters, use them as shown in the example of respective integrated application.
- If your application does not use servlet filters, but directly calls an HttpServlet as the point of entry, place the call to IntegratedAppRequest where the application enters the HttpServlet code.

ProcessRequest does the following tasks:

- If the URL is coming in for the first time from the TeamForge, it will have the singlesignon token. The method validates the token and stores a cookie which identifies the soap session for the user as well as the project for which this request is being made. This will also be the case for a Go URL coming in from TeamForge.
- If it is a subsequent request (from a form submission or a click from a previous page), then the cookie information is picked up and used in IntegratedAppSupport. For this, the integrated application expects that the linkid (the id linking the integrated application and the project, obtained through IntegratedAppSupport.getIntegratedAppId()) is provided either as a request attribute or as a request parameter or in the request path – it can be any one of these:

  ```
  http://my.integrated.application/reach/linkid/prplxxxx/me
  http://my.integrated.application/reach/me?linkId=prplxxxx
  ```

  a request parameter obtained through any other means and set as a Request Attribute.

It is advisable to set it so that all originating URLs (forms and other links to the application) have the `/linkid/prplxxxx` in their request URL. This helps subsequent URLs to be validated correctly. You can retrieve the linkid using IntegratedAppSupport.getIntegratedAppId.

You can configure your integrated application to display languages based on the TeamForge site user's browser locale.

These parameters can be internationalized:

- The description of the integrated application.

- The title and description of the integrated application when it appears as a component on a project page.
- The name and description of any configuration parameter.

1. In the XML application configuration file, select the values for which you want to provide localized content.
2. Inside the `bundles` tag, create a `bundle` tag identical to the default English bundle tag. (Keep the `en` bundle.)
3. Change the value of the `locale` attribute of the new `bundle` tag to the language you are going to provide. The value for each of the internationalized tags must start with `l10n`.

   For example:

```
<bundles>
      <bundle locale="en">
        <key name="l10n.application.description">My Test App</key>
        <key name="l10n.blogname.title">Blog Name</key>
        <key name="l10n.blogname.description">Please provide a name for th
e Blog. This appears on all blog pages</key>
            ........
      </bundle>
      <bundle locale="ja">    <!-- Japanese Language Translations -->
        <key name="l10n.application.description">My Test App</key>
        <key name="l10n.blogname.title">Blog Name</key>
        <key name="l10n.blogname.description">Please provide a name for th
e Blog. This appears on all blog pages</key>
            ........
      </bundle>
</bundles>
```

4. Save the application configuration file.

TeamForge provides a SOAP service for each tool in the application.

What can I do with the SOAP API?

You can use the TeamForge SOAP API to do almost anything a user can do in the TeamForge web user interface.

TeamForge exposes a subset of the APIs defined by the application server as web services, through the SOAP protocol.

A SOAP proxy server and a SOAP API layer, both running on Apache Axis, expose a set of web services representing each TeamForge application. The SOAP server serves the following functions:

- Provides web services by accepting SOAP requests from the clients.
- Performs SOAP client authentication.
- Implements TeamForge role-based access control (RBAC) and caching service.
- Accesses the application server via RMI stubs.

You can get the API here: http://www.collab.net/community/teamforge

The SOAP API can provide more types of functionality than the Web UI alone. For example, it can be time-consuming for a user with Tracker Admin permissions to copy a tracker workflow from one tracker to another in the Web UI. However, the SOAP method that provides a workflow copy function makes this task easy.

Access permissions

For access permissions (authorization, as distinct from authentication), the SOAP API follows the same rules imposed by the TeamForge role-based access control (RBAC) system. A user using a SOAP API-based client has no more access to data than they have in the Web UI.

The CollabNet TeamForge architecture allows you to quickly and easily develop integration points between CollabNet TeamForge and applications you develop.

User-centric services

All services and APIs are user-centric, meaning that all integrated applications must establish an individual connection to the SOAP server for each user. This differs from programming directly with an application server where one connection can be established for any number of users.

Activities that can be performed using the SOAP interface are by definition user-based, such as retrieving a list of a user's projects, tasks, or assigned tracker artifacts. These activities therefore require an individual connection for each user.

Requiring individual connections also ensures that role-based access control is checked for each action performed by each user. To ensure that security is enforced, RBAC checks are performed on each SOAP API call and cannot be disabled at the client level.

Consistent interaction

While each CollabNet TeamForge service has its own SOAP interface, interaction with each service is designed to be as consistent as possible. The calls for each service are similar, although the data format and specific call parameters may be different.

For example, the following calls are consistent across all services:

- list
- get
- set
- delete
- create

The call parameters, however, are different. For example:

- When working with the CollabNet service, you might call `getProjectList(string sessionID)`.
- When working with the TaskApp service, you might call `getTaskList(string sessionID, string taskFolderID)`.

Get started with the CollabNet TeamForge SOAP API

A CollabNet TeamForge plugin can provide any of the features a user can access through the Web interface.

1. Browse the CollabNet User Help to get a full picture of the functionality you can deliver with the SOAP API.

2. Get an idea of the sorts of applications that have been developed.

    - Look at the available applications.

    - Review the reference applications on the CollabNet community site. These samples illustrate some of the more common methods for using TeamForge via the SOAP API. They interact with most of the TeamForge data objects, including users, projects, trackers and artifacts, code commits and tasks.

    > **NOTE:** If you find a defect in a sample program or have a suggestion for improvement, please post a message in the developer discussion forum.

3. Set up the tools you will need:

    - A Java JDK, 1.8.0_131 or later, from http://www.oracle.com/technetwork/java/index.html.

    - Apache Ant, version 1.7.0 or later, from http://ant.apache.org.

4. Choose a place to host your plugin project online. We recommend http://www.collab.net/community, because you get access to all the collaboration support tools that CollabNet TeamForge provides.
5. Get the SDK from http://www.collab.net/community/teamforge. The SDK includes everything needed to develop and deploy your application:
    - Source files and compiled output.
    - Full JavaDoc reference material. Inside the package, look for doc/com/collabnet/ce/soap60/webservices.
    - Annotated code examples.
6. Get help from other CollabNet TeamForge users and CollabNet staff in the TeamForge API discussion forum.

Send Commit Data to TeamForge via SOAP

Use the Commit Tool provided with TeamForge to transmit your commit data to TeamForge.

The SCM Adapter allows you to develop integrations with almost any SCM tool, then exchange commit data with TeamForge. After you have created an SCM integration using the TeamForge API, you will use the Commit Tool provided with TeamForge to transmit your commit data to TeamForge.

For any SCM tool that supports triggers, you can use the tool's triggering mechanism to script the following actions. Otherwise, you can perform them manually for each commit.

> **NOTE:** The SCM Adapter option is always available, but will work only if you have developed an integration with an SCM tool using the TeamForge API.

1. Get the Commit Tool from `$SOURCEFORGE_HOME/integration/CommitTool.py`.
2. Add the SCM server to your TeamForge installation. For the end user instructions, see Integrate a Source Code Server.
3. Create a repository on the SCM server. For the end user instructions, see Create a Source Code Repository.
4. Use the Commit Tool to start a commit.

```
CommitTool.py create <systemId> <path> <username> <host> <port>
```

where:

- `systemId` is the external system identifier that was given to the integration server. You can find this on the **Repository Details** page.
- `path` is the path on the external system that was given to the repository. See the **Repository Details** page.
- `username` is the TeamForge user name that will be performing the commit.
- `host` is the TeamForge application server machine hostname.
- `port` is the TeamForge application server machine SOAP port.

5. After the commit has been created, use the Commit Tool to add files with versions and actions.

   ```
   CommitTool.py add <filename <version [<status [<fromfile <fromversion]]
   ```

   where:

   - `filename` is the name of the file that is being placed into the commit.
   - `version` is the version of the file that is being placed into the commit.
   - `status` is the status of the file in the repository. Valid status values are 'added', 'deleted', 'modified', 'moved', and 'copied'. Status is optional and will default to 'added' if no value is provided.
   - `fromfile` is the name of the original filename if this file was copied or moved. It is required on a copy or move, but is not allowed on an add, delete, or modify.
   - `fromversion` is the version of the original filename is this file was copied or moved. It is required on a copy or move, but is not allowed on an add, delete, or modify.

6. When all actions and files have been added to the commit, use the Commit Tool to transmit the commit to CollabNet TeamForge.

   ```
   CommitTool.py commit <commitmessage>
   ```

   where `<commitmessage>` is a description about the commit that will be displayed along with the files, versions, and operations.

> **NOTE:** To create an association, use the standard associate command []. No special syntax is required when using the Commit Tool.

For all of the above commands, when run correctly, there will be no output, and the return value of the program will be “0”, indicating success. Failure will show a message and return with a non “0” return value.

To see the status of the current commit, you can use the Commit Tool to print the output of the current commit:

```
CommitTool.py print
```

It will show output similar to this:

```
Repository: <username>@<systemid>:<path> Modified filename 1.2 Added documen
t 1.1 Copied newfile 1.1 (From: origfile 1.3) <status> <filename> <version> (
From: <fromfile> <fromversion>)
```

Update an application to TeamForge 18.1 API

To use any of the new API calls introduced in TeamForge 18.1, you must update your existing applications to use the 18.1 API.

See TeamForge 18.1 Javadoc for more information.

- For calls that use the TeamForge-provided Java SDK classes, update all references to the package containing the classes.

  The package is located at `com.collabnet.ce.soap60`.

- For calls that access the WSDL directly, point your application to the new WSDL location.

  The WSDL is located at `http://<teamforge-server>/ce-soap60/services/<service-name>?wsdl`.

- Update any calls that have been changed in the 18.1 API.

TeamForge Services Available via SOAP

TeamForge provides a SOAP service for each tool in the application.

NOTE: The WSDL for each TeamForge service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/<myService>?wsdl`. For an example, to review the WSDL for the Tracker service, see http://ctf.open.collab.net/ce-soap60/services/TrackerApp?wsdl. For information on the SOAP API, including the pluggable component package and classes, see the SOAP package summary.

The CollabNet Service

The CollabNet service is the primary service that handles logging into and out of CollabNet TeamForge , finding and creating users, and retrieving lists of projects, project members, and project data.

The CollabNet service must always be accessed first, to authenticate the user and return a sessionId. The sessionId is a necessary parameter that must be passed to the methods in the other services.

The CollabNet service also handles retrieving and managing lists of users and projects, which are often the first steps that an application must perform.

Some examples of activities that are managed by the CollabNet service are:

- Returning a list of all CollabNet TeamForge users or projects.
- Creating a new user.
- Providing a user with a list of all projects of which he or she is a member.
- Creating an association between two items.
- Managing user groups and their membership.

User groups, added in CollabNet TeamForge Enterprise Edition 4.4 Service Pack 1, simplify permission management for projects. User groups are site-wide sets of users, managed by site administrators. Project administrators can add user groups to roles, just like they can add individual project members.

After authenticating the user through the CollabNet service, you can then access all of the other services that manage the items and activities associated with the other CollabNet TeamForge applications, such as trackers, tasks, documents, or file releases. A new “Anonymous Login” capability introduced in CollabNet TeamForge Enterprise Edition 4.4 adds the ability to log in with the privileges and access rights of a non-authenticated user.

For a complete description of each method, including its parameter definitions and SOAP faults and for a list of new and changed methods, see the JavaDoc for this service.

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/CollabNet?wsdl`. For example, here’s the copy on the CollabNet web site: http://ctf.open.collab.net/ce-soap60/services/CollabNet?wsdl.

The Discussion Service

The DiscussionApp service handles the activities and items associated with forums.

Some examples of activities that are managed by the DiscussionApp service are:

- Creating a forum, forum topic, or forum post.
- Returning a list of all posts in a forum.
- Returning a list of forum posts created by a specified user.
- Returning a list of forum posts matching a search string.
- Deleting a forum, forum topic, or forum post.

After authenticating the user through the CollabNet service, you can then access the DiscussionApp service to work with forums, forum topics, and forum posts.

For a complete description of each method, including its parameter definitions and SOAP faults and for a list of new and changed methods, see the JavaDoc for this service.

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/DiscussionApp?wsdl`. For example, here's the copy on the CollabNet web site: http://ctf.open.collab.net/ce-soap60/services/DiscussionApp?wsdl.

The Document Service

The DocumentApp service handles the activities and items associated with documents and document folders.

Some examples of activities that are managed by the DocumentApp service are:

- Creating a document or document folder.
- Editing a document's details.
- Returning the name of the user who last edited or locked a document.
- Returning a list of documents matching a search string.
- Returning a list of all document folders in a project.
- Returning a list of all open document reviews in a project.
- Moving a document or document folder.

After authenticating the user through the CollabNet service, you can access the DocumentApp service to work with documents and document folders.

For a complete description of each method, including its parameter definitions and SOAP faults and for a list of new and changed methods, see the JavaDoc for this service.

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/DocumentApp?wsdl`. For example, here's the copy on the CollabNet web site: http://ctf.open.collab.net/ce-soap60/services/DocumentApp?wsdl.

The File Release Service

The FrsApp service handles the activities and items associated with file releases.

Some examples of activities that are managed by the FrsApp service are:

- Creating a package, release, or file.
- Editing a package, release, or file.
- Returning a list of all packages in a project.
- Returning a list of all releases in a package.
- Returning the status, maturity value, or other attribute of a release.

After authenticating the user through the CollabNet service, you can access the FrsApp service to work with file releases.

For a complete description of each method, including its parameter definitions and SOAP faults and for a list of new and changed methods, see the JavaDoc for this service.

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/FrsApp?wsdl`. For example, here's the copy on the CollabNet web site: http://ctf.open.collab.net/ce-soap60/services/FrsApp?wsdl.

The File Storage service

The FileStorageApp and SimpleFileStorageApp services handle uploading of files and simple data bytes.

Some examples of activities that are managed by the FileStorageApp and SimpleFileStorageApp services are:

- Uploading a file to CollabNet TeamForge.
- Downloading a file from CollabNet TeamForge.
- Checking the size of a file.

After authenticating the user through the CollabNet service, you can then access the FileStorageApp and SimpleFileStorageApp services to work with file uploads.

For a complete description of each method, including its parameter definitions and SOAP faults and for a list of new and changed methods, see the JavaDoc for this service.

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/FileStorageApp?wsdl`. For example, here’s the copy on the CollabNet web site: http://ctf.open.collab.net/ce-soap60/services/FileStorageApp?wsdl.

The Integration Data Service

The IntegrationDataApp service enables applications to associate their own data or metadata with CollabNet TeamForge objects.

An application can register its own namespace, to prevent collisions with other applications, and then store key-value pairs of data that are associated with any CollabNet TeamForge object.

- Developers who are creating integrations between CollabNet TeamForge and external applications can use this service to maintain associations between CollabNet TeamForge objects and counterparts in other systems.
- Developers creating customizations or enhancements can use this service to store additional data not directly supported by other CollabNet TeamForge APIs. For example, you can create extra, hidden fields on any object (such as an artifact, a user or a project) and use a Velocity customization to display that data where you need it.

For a complete description of each method, including its parameter definitions and SOAP faults and for a list of new and changed methods, see the JavaDoc for this service.

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/IntegrationDataApp?wsdl`. For example, here’s the copy on the CollabNet web site: https://ctf.open.collab.net/ce-soap60/services/IntegrationDataApp?wsdl.

The News Service

The NewsApp service handles the activities and items associated with news posts.

News posts for a project are visible on the project's home page, and the application's main page lists news posts for all projects visible to the current user.

Starting with CollabNet TeamForge , project news is presented on the project's home page via a “Project News” page component, which can optionally be deleted or added to other pages. Each instance of the Project News component is a view of the same set of project news posts.

Some examples of activities that are managed by the NewsApp service are:

- Returning a list of all news posts in a project.
- Returning a list of all news posts matching a search string.
- Returning a list of all news posts in all projects of which a specified user is a member.
- Creating a news post.

- Deleting a news post.

After authenticating the user through the CollabNet service, you can access the NewsApp service to work with news posts.

For a complete description of each method, including its parameter definitions and SOAP faults and for a list of new and changed methods, see the JavaDoc for this service.

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/NewsApp?wsdl`. For example, here’s the copy on the CollabNet web site: https://ctf.open.collab.net/ce-soap60/services/NewsApp?wsdl.

The Planning Folder Service

The Planning Folder service enables you to create and maintain planning folders from external applications.

Planning folders are visible in the tracker tool of each project, as an organizing level sitting on top of trackers. Planning folders are deisgned to support agile management practices.

Some examples of activities managed by the PlanningApp service are:

- Creating and deleting planning folders.
- Reordering planning folders.
- Setting the statuses of planning folders.

For a complete description of each method, including its parameter definitions and SOAP faults and for a list of new and changed methods, see the JavaDoc for this service.

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/PlanningApp?wsdl`. For example, here’s the copy on the CollabNet web site: https://ctf.open.collab.net/ce-soap60/services/PlanningApp?wsdl.

The Project Pages Service

The PageApp service handles the activities and items associated with project pages.

The project pages feature enables project owners to customize their project home by creating their own custom pages. Multiple pages can be added, and these pages can be assembled into a hierarchy, browseable via a “tree” navigation on the left side of the project's home. The content of each page can be customized by adding components of various types, such as Text, News, and Document Folder.

Some examples of activities that are managed by the PageApp service are:

- Creating pages and components.
- Managing the page hierarchy.

- Managing the order of components on a page.
- Updating text components with new HTML content.

For a complete description of each method, including its parameter definitions and SOAP faults and for a list of new and changed methods, see the JavaDoc for this service.

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/PageApp?wsdl`. For example, here's the copy on the CollabNet web site: https://ctf.open.collab.net/ce-soap60/services/PageApp?wsdl.

The Project Categorization Service

The CategorizationApp service handles the activities and items associated with creation and management of Project Categorization.

Some examples of activities that are managed by the CategorizationApp service are:

- Creating categories
- Adding categories to projects
- Retrieving category and project data.

For a complete description of each method, including its parameter definitions and SOAP faults and for a list of new and changed methods, see the JavaDoc for this service.

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/CategorizationApp?wsdl`. For example, here's the copy on the CollabNet web site: https://ctf.open.collab.net/ce-soap60/services/CategorizationApp?wsdl.

The Role-based Access Control Service

The RbacApp service handles the activities and items associated with role creation and management.

Some examples of activities that are managed by the RbacApp service are:

- Creating a role.
- Adding a user to a role.
- Adding permissions to a role.
- Adding a user group to a role.

Permissions are managed in terms of "clusters." Permission clusters correspond to the sets of permissions that are managed in the role administration section of the application's user interface, such as "view" or "create/edit/view." Clusters can be associated with top-level folders in the application (e.g. with trackers in the Tracker application, or packages in the File Releases application). The RbacApp service allows this same permission management to be performed via SOAP.

For a complete description of each method, including its parameter definitions and SOAP faults and for a list of new and changed methods, see the JavaDoc for this service.

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/RbacApp?wsdl`. For example, here's the copy on the CollabNet web site: https://ctf.open.collab.net/ce-soap60/services/RbacApp?wsdl.

The Software Configuration Management (SCM) Service

The ScmApp service handles the activities and items associated with integrated software configuration management (SCM) applications.

Some examples of activities that are managed by the ScmApp service are:

- Returning a list of files associated with a code commit.
- Returning a list of all repositories in a project.
- Returning a list of all code commits in a repository.
- Editing commit information.

After authenticating the user through the CollabNet service, you can access the ScmApp service to work with integrated SCM applications.

For a complete description of each method, including its parameter definitions and SOAP faults and for a list of new and changed methods, see the JavaDoc for this service.

The WSDL for this service is available on your TeamForge site at `https://<mysite.com↗/ce-soap60/services/ScmApp?wsdl`. For example, here's the copy on the CollabNet web site: https://ctf.open.collab.net/ce-soap60/services/ScmApp?wsdl.

SCM Adapter

The SCM Adapter allows you to develop integrations with almost any SCM tool, then exchange commit data with CollabNet TeamForge. An SCM Adapter option is added to the Type menu on the CollabNet TeamForge Create Integration page. Use this option to add your SCM integration server to CollabNet TeamForge.

> **NOTE:** The SCM Adapter option is always available, but will work only if you have developed an integration with an SCM tool using the CollabNet TeamForge API.

After you have created an SCM integration using the CollabNet TeamForge API, you will use the Commit Tool provided with CollabNet TeamForge to transmit your commit data to CollabNet TeamForge.

The Tag Service

The TagApp service provides SOAP services for tag objects.

Some examples of activities that are managed by the TagApp service are:

- Create a tag in a specific project.
- Deleting a tag from a project.
- Get tag data for a given tag ID.
- Gets the list of tags in a project.

For a complete description of each method, including its parameter definitions and SOAP faults and for a list of new and changed methods, see the JavaDoc for this service.

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/TagApp?wsdl`. For example, here's the copy on the CollabNet web site: https://www.open.collab.net/community/cif_sfee/samples/TagApp.xml.

The Tasks Service

The TaskApp service handles the activities and items associated with tasks and task folders.

Some examples of activities that are managed by the Task App service are:

- Returning a list of all tasks assigned to a specified user.
- Returning a list of all task folders in a project.
- Returning a list of all tasks matching a search string.
- Listing task dependencies.
- Creating a task or task folder.
- Editing a task.
- Moving tasks and task folders.

After authenticating the user through the CollabNet service, you can access the TaskApp service to work with tasks and task folders.

> **NOTE:** Task folders are referred to as task groups in the method descriptions.

For a complete description of each method, including its parameter definitions and SOAP faults and for a list of new and changed methods, see the JavaDoc for this service.

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/TaskApp?wsdl`. For example, here's the copy on the CollabNet web site: https://ctf.open.collab.net/ce-soap60/services/TaskApp?wsdl.

The Tracker Service

The TrackerApp service handles the activities and items associated with trackers and tracker artifacts.

Some examples of activities that are managed by the TrackerApp service are:

- Returning a list of all trackers in a project.
- Returning a list of all tracker artifacts matching a search string.
- Returning a list of all tracker artifacts assigned to a specified user.
- Creating a tracker artifact.
- Editing a tracker artifact.
- Moving a tracker artifact.

After authenticating the user through the CollabNet service, you can access the TrackerApp service to work with trackers and tracker artifacts.

For a complete description of each method, including its parameter definitions and SOAP faults and for a list of new and changed methods, see the JavaDoc for this service.

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/TrackerApp?wsdl`. For example, here's the copy on the CollabNet web site: https://ctf.open.collab.net/ce-soap60/services/TrackerApp?wsdl.

The Wiki Service

The WikiApp service handles activities associated with project wiki pages.

Some examples of activities that are managed by the WikiApp service are:

- Creating a Wiki page.
- Adding Wiki content.
- Retrieving Wiki content in HTML format.

For a complete description of each method, including its parameter definitions and SOAP faults and for a list of new and changed methods, see the JavaDoc for this service.

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/WikiApp?wsdl`. For example, here's the copy on the CollabNet web site: https://ctf.open.collab.net/ce-soap60/services/WikiApp?wsdl.

Context-specific Objects for Manipulating Velocity Pages

The rendering context is a set of objects made available to the Velocity template. You can access the members of these objects in a template using the standard Velocity syntax.

ArrayTool

This is necessary because Velocity does not provide accessors into arrays and some of the API calls will return SoapNamedValues objects that have two matching arrays of name and value.

length(Object[] array)

The array will be examined and will get the length of the array.

The function returns the length of the array (null array will return 0).

get(Object[] array, int index)

Get the object at the specified index of the array that contains the object of interest. The index is from which we pull the object from.

This function returns the object stored at that array of null if the array is empty or position is invalid.

FORM [FormTool]

Manages opening and closing of forms.

- **startForm(String action, String formId)**

  Get the opening tag for a form.

  - **action** - The path to the action that is handling the form (e.g. `/cemain/do/ login`).
  - **formId** - The id/name of the form (they will be the same).

  This function returns the open tag for the form and any standard hidden elements.

- **startForm(String action, String formId, String formName)**

  Get the opening tag for a form.

  - **action** - The path to the action that is handling the form (e.g. `/cemain/do/ login`).
  - **formId** - The id of the form.
  - **formName** - The name of the form.

  This function returns the open tag for the form and any standard hidden elements.

- **endForm()**

  Close out a form.

  This function returns the form closing tag.

LINK [LinkTool]

Creates links; see default templates for examples.

- **getPageUrl(String urlBase, String action)**

  Get the url for a page being linked to.

  - **urlBase** - The base url.
  - **action** - The action being linked to.

  This function returns the value to place in the href attribute of a link.

- **getUserUrl(String username, String fullName)**

  Generates URL for user details.

  - **username** - User name.
  - **fullName** - User’s full name.

  This function returns the URL to user details page.

MESSAGE [MessageTool]

Enables internationalization of templates, providing functions for localizing strings.

- **get(String bundle, String key[, String arg0[, String arg1[, String arg2]]])**

  Returns the message.

  - **bundle** - Resource bundle.

  - **key** - Message key.
  - **arg0** - Message argument.
  - **arg1** - Message argument.
  - **arg2** - Message argument.

  This function returns the message.

- **getFieldLabel(String bundle, String key, boolean required)**

  Returns the message with the appropriate label modifier (‘:’ and possibly an asterisk if field is required).

  - **bundle** - The bundle where the message lives.
  - **key** - The key where the message is stored.
  - **required** - If the field is required.

  This function returns the message with appropriate label modifiers.

STRING [StringTool]

Returns a formatted file size string, such as 45KB, 100MB, or 1GB.

- **formatFileSize(long fileSize)**

  Returns the file size string.

  fileSize - File size in bytes.

  This function returns the file size string.

- **wordWrap(String text, int width)**

  Wraps text string at word boundaries.

  - **s** - String to word wrap.
  - **width** - Number of characters at which to wrap.

  This function returns word wrapped string.

- **escapeXml(String s)**

  Escape out the xml from the text.

  s - The text that we need to escape xml from.

This function returns the text with xml escaped.

TEXTPARSER [TextParserTool]

Converts text into “linkified” html, with IDs and wiki words converted into appropriate links.

- **parseText(String text)**

  Returns linkified text (add href tags to object ids and urls).

  text - Text to process.

  This function returns the replacement text with link replacement.

PAGE_INFO [PageInformationTool]

Allows access to information specific to the page that is passed in.

Information	Description		
actionName	The name of the action that was requested (e.g. viewArtifact, listTrackers)		
currentUser	User object that contains information about the current user: • $PAGE_INFO.currentUser.username (Login ID) • $PAGE_INFO.currentUser.fullName (Full name) • $PAGE_INFO.currentUser.email (E-mail address) • $PAGE_INFO.currentUser.locale (Locale, such as en_US) • $PAGE_INFO.currentUser.lastLogin (Date and time of last login)		
path	The full path string of the request's target object (e.g. projects.test/trackers.foo/bar). If no path is present on the request, this returns an empty string.		
projectPath			The project path (e.g. projects.test) on the current folderPath or an empty string if none exists (no path context or just a project context).

itemName	The name of the item (e.g. artf1234, Home) on the current request or an empty string if there is no path context or just a project or folder context.		
objectId	The id of the requested object (e.g. artf1234, proj1234).		
objectType	The type string of the current object (i.e. An artifact would have Tracker.Artifact)		
projectId	The ID of the project on the request or the project that contains the current folder or item.		
requestUrl	The URL that was requested (e.g. /ce/tracker/do/viewArtifact/projects.test/tracker.foo/artf1234).		
isSuperUser	True, if the current user is a site admin.		
isLoggedIn	True, if the current user is logged into the system.		

GLOBAL [GlobalTool]

Allows access to various kinds of static information about the site.

Static Information	Description
datePattern	The standard date pattern used for date conversions.
imageRoot	The root URL path of CollabNet TeamForge images.
isApprovalRequiredForNewUsers	A flag that indicates whether new users must be approved before they can use the site.
isRequireAssociationOnDocumentCreate	A flag that indicates whether creation of a document requires an association with an object.
isSelfCreationEnabled	A flag that indicates whether visitors are allowed to create new accounts in the application by themselves.
isUsingExternalAuthentication	A flag that indicates whether logins are authenticated against an external server. (Currently this means LDAP.)

ApiTool

API60 [ApiTool] exposes the CollabNet TeamForge SOAP service interfaces for use by Velocity templates.

While the interfaces corresponding to the SOAP API are provided, the implementation is efficient; the SOAP network protocol is not actually used.

> **NOTE:** API50 is present for backward compatibility and may be removed in future. API44 and API43 are removed completely.

Interfaces

Interface	Description

sessionKey	Gets the soap session id that should be passed in as the first parameter to soap service methods.
discussionApp	Gives access to the methods in the Discussion application.
documentApp	Gives access to the methods in the Documents application.
frsApp	Gives access to the methods in the File Releases application.
integrationDataApp	Gives access to the methods for Data Integration.
newsApp	Gives access to the methods in the Project News application.
rbacApp	Gives access to the methods for role-based access control.
scmApp	Gives access to the methods for version control (SCM) integration.
SourceForge	Gives access to the methods in the main CollabNet TeamForge application.
taskApp	Gives access to the methods in the Tasks application.
trackerApp	Gives access to the methods in the Tracker application.
wikiApp	Gives access to the methods in the Wiki application.
categorizationApp	Gives access to the methods in project categorization.
emptyFilter	Gets an empty filter that can be passed into methods that require soap filters.
PageApp	Handles the activities and items associated with project pages.
PlanningApp	Gives access to methods in the Planning application.

Example

This is an example of how you might use some popular API calls in a Velocity template:

```
## Display Server version, current project title and description
#set( $sessionKey = $API60.sessionKey )
#set( $projectId = $PAGE_INFO.projectId )
#set( $projectData = $API60.sourceForge.getProjectData($sessionKey,$projectId)
 )

<p>Server Version: $API60.sourceForge.getVersion($sessionKey)</p>
<p>Project Title: ${projectData.title}</p>
<p>Project Desc: ${projectData.description}</p>
```

Lab Management API Methods

Almost all Lab Management functionality is available via an API method.

Lab Management API methods come in two forms: signed and unsigned.

- Signed methods require authentication, in accordance with the User Authentication documentation. Most of the API methods in Lab Management, and all of the most useful ones, are signed methods which require authentication to use. The `cubit_api_client.py` client makes using these signed methods easy by handling the authentication negotiation for you.

- Unsigned methods do not require authentication.

> **NOTE:** This API is under active development. We have started with what we feel are the most common actions which would need to be performed on an automated basis, and will be expanding this list over time as we receive feedback about what our users want and need. Please let us know if there are missing features or bugs!

CheckHostConsistency

Check the setup of all hosts in the domain to make sure their representation is internally complete within Lab Management.

Overview

This method is only available to users to Lab Management Domain Admins and can be considered a maintenance method that should rarely, if ever, need to be run. Three areas of setup consistency are checked for each host:

1. Verification and re-application of the host's internal database representation within Lab Management. This may be needed after an upgrade of the Lab Management application is installed.

2. Verification that the IP address assigned to the host matches the hostname. If you suspect DNS setup issues in your domain, this test should catch them (unless the errors are transient). Lab Management will not let you create hosts with invalid DNS, but if DNS for the host gets changed improperly after the host is created, your host will experience many problems in the Lab Management environment until it is fixed.

3. Verification of the IP mask/gateway configured for the host with the actual one it resolves to.

4. Re-generation of statistics system performance monitoring configuration files for the host. If monitoring is not properly working for a host, regenerating the configuration files will usually fix the problem.

Depending on the number of hosts in your domain, this method can take some time to complete. If you have questions on the output of this method, please see your local Lab Management Administrator or contact CollabNet directly for assistance.

URL

```
/cubit_api/1/check_host_consistency
```

Authentication

This method requires authentication using an API key.

Parameters

- **sig** (Required, once)
  - API authentication hash signature.
  - Type: String
- **userid** (Required, once)
  - The login name of the user initiating the request.
  - Type: String

Example Response

Successful completion of host consistency check:

```
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
    <output>All Hosts OK</output>
</cubit>

If the user is not a Lab Management Domain Admin:
<?xml version='1.0'?>
<cubit version='1'>
    <error>User 'grue' is not authorized to run this method.</error>
</cubit>

If there is an error re-generating the host's internal database
representation for at least one host:
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
        <output>Hosts which were not successfully updated:
        cu011.dev.cubitdemo.net</output>
</cubit>

If the system performance statistics configuration files are not
properly generated for at least one host:
<?xml version='1.0'?>
<cubit version='1'>
```

```
    <status>OK</status>
        <output>Hosts which are not properly setup to collect
        performance data: cu011.dev.cubitdemo.net</output>
</cubit>

If there is a DNS mismatch for at least one host:
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
        <output>Hosts which require re-ip'ing:
        cu011.dev.cubitdemo.net</output>
</cubit>

If there is a IP mask mis-match:
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
        <output>Hosts whose IP mask/gateway does not match with the
        IP mask/gateway it resolves to:
        cu011.dev.cubitdemo.net</output>
</cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

CloudCreateHosts

Create one or more hosts from a cloud given the host-type and size.

The cloud will match the host-type and size to one (or more) source(s) from the cloud in order to fill the entire order. Host types and sizes cannot be mixed in a single API call. All the hosts must be the same type/size. The method will fail if the cloud's sources cannot accomodate that many hosts of the given type/size.

Any user who is a member of a project that has been given permission to allocate from this cloud can create instances.

URL

```
/cubit_api/1/allocate_hosts_from_cloud
```

Authentication

This method requires authentication using an API key.

Parameters

- **alloc_hours** (zero or once)
    - Amount of time for which to allocate the host. The allocation limit is subject to the limit set by the project or cloud administrator.
    - Type: Float
- **alloc_minutes** (zero or once)
    - Similar to alloc_hours parameter, but in minutes. It is mutually exclusive with the `alloc_hours` option, neither of these options should be specified, or only one of these options should be specified. If `alloc_hours` or `alloc_minutes` is set to 0, or if `alloc_hours` and `alloc_minutes` is unset, the allocation time defaults to the longest possible time available in the project.
    - Type: Integer
- **cloud** (Required, once)
    - Name of Lab Management cloud to allocate instances from.
    - Type: String
- **count** (zero or once)
    - The number of identical instances to bring up. The instances will all be running the same profile and version. Must be an integer greater than 0. If unset, defaults to 1.
    - Type: Integer
- **descr** (zero or once)
    - Set the specified string as a description for the allocated hosts. This is useful if you wish to uniquely identify a group of hosts which have all been allocated to the same user at the same time for the same purpose.
    - Type: String
- **dont_delete** (zero or once)
    - By default, EC2 instances will automatically be deleted when they are deallocated. If `dont_delete` is set to `True`, the instances created will not be deleted when they are deallocated, and will instead revert back to the project pool.
    - Type: String

- **host_type** (Required, once)
  - The name of the host_type of machine to create. Must be one of the host-types supported by the cloud.
  - Type: String
- **profile** (Required, once)
  - Name of the profile to assign to the host. The profile must, of course, be a profile that is eligible to be built by the host-type/size you selected.
  - Type: String
- **project** (Required, once)
  - Name of Lab Management project to place hosts into. You must have permission to add hosts to this project: that is, you must either be a Project Admin or Delegated Host Management must be turned on in your project.
  - Type: String
- **revision** (zero or once)
  - Revision number of the profile you wish to assign to the host. Mutually exclusive with the `version` option, but at least one of the `version` or `revision` options must be specified.
  - Type: Integer
- **sig** (Required, once)
  - API authentication hash signature.
  - Type: String
- **size** (Required, once)
  - The name of the machine size to create. Must be one of the sizes supported by the given host_type.
  - Type: String
- **userid** (Required, once)
  - The login name of the user initiating the request.
  - Type: String
- **version** (zero or once)

- Version number or tag name of the profile you wish to assign to the host. The special version tag `HEAD` always denotes the latest version of the profile at the moment of execution. Note that profile tags can move between versions: this is a useful feature, but you should be aware of it. Mutually exclusive with the `revision` option, but at least one of the `version` or `revision` options must be specified.
- Type: String

Example Response

Successful allocation of 10 instances:

```
    <?xml version='1.0'?>
    <cubit version='1'>
        <status>OK</status>
        <output>ec2-pending-1204832590-85</output>
        <output>ec2-pending-1204832595-09</output>
        <output>ec2-pending-1204832599-97</output>
        <output>ec2-pending-1204832611-49</output>
        <output>ec2-pending-1204832615-60</output>
        <output>ec2-pending-1204832620-29</output>
        <output>ec2-pending-1204832624-25</output>
        <output>ec2-pending-1204832628-46</output>
        <output>ec2-pending-1204832640-90</output>
        <output>ec2-pending-1204832645-60</output>
    </cubit>

    <?xml version='1.0'?>
    <cubit version='1'>
        <status>OK</status>
    </cubit>

    If the user lacks permission to create hosts in the project:
    <?xml version='1.0'?>
    <cubit version='1'>
        <error>You are not authorized to create hosts in this project.</error>
    </cubit>

    If the project does not have the permissions to allocate from the
    requested cloud:
    <?xml version='1.0'?>
    <cubit version='1'>
        <error>This project is not authorized to create hosts in this cloud.</e
rror>
    </cubit>

    If the cloud does not have enough space on the sources to allocate that ma
ny hosts:
    <?xml version='1.0'?>
    <cubit version='1'>
        <error>This cloud does not have enough resources to create [n] hosts.↙
error>
    ↙cubit>
```

Response Codes

- 400 - Login failed / Insufficient permissions

GetAuthToken

GetAuthToken(userid) -> token

URL

`/cubit_api/1/get_auth_token`

Authentication

This method does not require authentication.

Parameters

This method does not take any parameters.

Response Codes

This method does not have any documented response codes.

HostAllocate

Allocate a host to a user.

Hosts must be in the Free state to be allocated, although users with Project Admin access can re-allocate hosts which are in the `Allocated` state to other users in their project. Domain Admin users can re-allocate machines between projects and users, although if a machine is reallocated to a user in another project, that user must also be a member of the destination project. The user must be authorized to allocate the host. If the host is in any of the following states, this method will fail.

- Immutable

- Powercycle
- Rebuild or
- Rebuilding

URL

`/cubit_api/1/alloc_host`

Authentication

This method requires authentication using an API key.

Parameters

- **alloc_hours** (zero or once)
  - Amount of time for which to allocate the host. The allocation limit is subject to the limit set by the project or cloud administrator.
  - Type: Float
- **alloc_minutes** (zero or once)
  - Similar to alloc_hours parameter, but in minutes. It is mutually exclusive with the `alloc_hours` option, neither of these options should be specified, or only one of these options should be specified. If `alloc_hours` or `alloc_minutes` is set to `0`, or if `alloc_hours` and `alloc_minutes` is unset, the allocation time defaults to the longest possible time available in the project.
  - Type: Integer
- **alloc_proj** (zero or once)
  - The project to allocate the machine to. The alloc_user must be a valid Lab Management user in the project, or else this method will fail. Only Domain Admins can change the project that a host is assigned to. If not specified, the project is not changed. At least one of `alloc_user` or `alloc_proj` must be specified.
  - Type: String
- **alloc_user** (zero or once)

    - Login name of user to allocate host to. Only Project Admins or better can specify a user other than themselves. Other users can only allocate machines to themselves. If not specified, defaults to userid. At least one of `alloc_user` or `alloc_proj` must be specified.
    - Type: String

- **force** (zero or once)

    - If the machine is currently in the Allocated state, the force option must be given to reassign the host to another user. The force option is only available to Project Admins and above. If the host is a virtual host, and any virtual guests of this host are in the Allocated state, this option must also be used, or the entire allocation will fail. The only valid value for this parameter is `True`.
    - Type: String

- **guests** (zero or once)

    - If the machine is a virtual host and has active virtual guests, setting this parameter to `True` will move the virtual guests along with the virtual host. Because moving a virtual host will move several hosts at once, we have a separate parameter to confirm this action. The only valid value for this parameter is `True`. If the host has no virtual guests, this option has no effect.
    - Type: String

- **host** (Required, once)

    - Fully qualified hostname to allocate.
    - Type: String

- **sig** (Required, once)

    - API authentication hash signature.
    - Type: String

- **userid** (Required, once)

    - The login name of the user initiating the request.
    - Type: String

Example Response

Successful host allocation:

```
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
```

```
</cubit>

If the user is unauthorized, or the host does not exist:
<?xml version='1.0'?>
<cubit version='1'>
    <error>You are not authorized to allocate this host.</error>
</cubit>

If the host is allocated to a deleted project:
<?xml version='1.0'?>
<cubit version='1'>
    <error>Cannot allocate host 'cu001.dev.cubitdemo.net' to a
    deleted project 'look'</error>
</cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

HostAssignProfileAndRebuild

Assign a profile to the host and rebuild it with that profile/version.

The user must be authorized to rebuild the host. If the host is in any of the states which disallow rebuilding, such as `Immutable`, `Rebuild`, or `Rebuilding`, this method will fail. The host must also not be of a type which prohibits rebuilding (e.g., an EC2 host). If you do not want to change the profile a host is running, and just want to rebuild a host, you can save a few keystrokes and use the Rebuild method instead.

URL

`/cubit_api/1/assign_rebuild`

Authentication

This method requires authentication using an API key.

Parameters

- **host** (Required, once)
  - Fully qualified hostname to assign profile to and rebuild.
  - Type: String

- **profile** (Required, once)
  - Name of the profile to assign to the host.
  - Type: String
- **revision** (zero or once)
  - Revision number of the profile you wish to assign to the host. Mutually exclusive with the version option, but at least one of the version or revision options must be specified.
  - Type: String
- **sig** (Required, once)
  - API authentication hash signature.
  - Type: String
- **userid** (Required, once)
  - The login name of the user initiating the request.
  - Type: String
- **version** (zero or once)
  - Version number or tag name of the profile you wish to assign to the host. The special version tag `HEAD` always denotes the latest version of the profile at the moment of execution. Note that profile tags can move between versions: this is a useful feature, but you should be aware of it. Mutually exclusive with the `revision` option, but at least one of the `version` or `revision` options must be specified.
  - Type: String

Example Response

Successful initiation of rebuild:

```
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
</cubit>

If the user is unauthorized, or the host does not exist:
<?xml version='1.0'?>
<cubit version='1'>
    <error>You are not authorized to allocate a profile to this host and re
```

```
build it.</error>
    </cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

HostDelete

Delete one or more hosts.

The host must not be in the `Immutable` state for this method to work. Hosts can be deleted by users under the following conditions:

- Lab Management Domain Admins can delete any host, regardless if it is physical, virtual, or from a remote cloud.
- Users who are not Lab Management Domain Admins can delete `virtual guests` if the parent host is allocated to them and Delegated Host Management is turned on in their project.
- Users who are not Lab Management Domain Admins can also delete `remote cloud hosts` allocated to them if Delegated Host Management is turned on in their project.

URL

`/cubit_api/1/delete_host`

Authentication

This method requires authentication using an API key.

Parameters

- **force** (zero or once)
    - If the machine being deleted does not belong to the user initiating the request, the force option is required. The only valid value for this parameter is `True`.
    - Type: String
- **guests** (zero or once)

- ◦ If the machine is a virtual host and has active virtual guests, setting this parameter to `True` will delete the virtual guests along with the virtual host. Because deleting a virtual host will delete several hosts at once, we have a separate parameter to confirm this action. The only valid value for this parameter is `True`. If the host has no virtual guests, this option has no effect. Use of this option can be really convenient, or it can really ruin someone’s day!
- ◦ Type: String

- **hosts** (Required, once)
  - List of comma-separated, fully qualified hostnames to delete. Hosts must not be in the `Immutable` state.
  - Type: String
- **sig** (Required, once)
  - API authentication hash signature.
  - Type: String
- **userid** (Required, once)
  - The login name of the user initiating the request.
  - Type: String

Example Response

Successful host delete of the host `cu011.cubit.example.com`:

```
    <?xml version='1.0'?>
    <cubit version='1'>
        <status>OK</status>
        <output>cu011.cubit.example.com</output>
    </cubit>

    If the user is unauthorized:
    <?xml version='1.0'?>
    <cubit version='1'>
        <error>0 of 1 hosts deleted successfully. Failed hosts below.</error>
        <output>cu013.cubit.example.com: You are not authorized to delete this
 host.</output>
    </cubit>

    If a deletion of hosts partially succeeds:
    <?xml version='1.0'?>
    <cubit version='1'>
```

```
        <error>6 of 8 hosts deleted successfully. Failed hosts below.</error>
        <output>cu013.cubit.example.com: You are not authorized to delete this
host</output>
        <output>cu016.cubit.example.com: Host deletion failed</output>
    </cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

HostFree

Free a host currently assigned to a user.

Hosts must be in the `Allocated` state to be freed. A user can only free hosts already assigned to them, although Project Admins and Domain Admins can free hosts in their project which are currently allocated. If it is a virtual guest and “Delete on Deallocation” flag is ON, the virtual guest would be deleted after it is set Free.

URL

`/cubit_api/1/free_host`

Authentication

This method requires authentication using an API key.

Parameters

- **force** (zero or once)
    - If the machine being freed does not belong to the user initiating the request, the force option is required. Only Project Admins and Domain Admins can free hosts which do not belong to them. The only valid value for this parameter is `True`.
    - Type: String
- **host** (Required, once)
    - Fully qualified hostname to free. Hosts must be in the Allocated state to be freed.
    - Type: String

- **sig** (Required, once)
  - API authentication hash signature.
  - Type: String
- **userid** (Required, once)
  - The login name of the user initiating the request.
  - Type: String

Example Response

Successful freeing of host:

```
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
</cubit>

If the user is unauthorized, or the host does not exist:
<?xml version='1.0'?>
<cubit version='1'>
    <error>You are not authorized to free this host.</error>
</cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

HostGetAssignedProfile

Get the assigned profile for the host, and returns the profile name along with the revision and version number.

Output is available in both text and XML formats. You can also get this information from the QueryAlloc method, but the output of this method is easier to parse if you have scripts checking a profile. The user requesting information about a host must have at least login privileges to that host.

URL

`/cubit_api/1/assigned_profile`

Authentication

This method requires authentication using an API key.

Parameters

- **host** (Required, once)
  - Fully qualified hostname to query.
  - Type: String
- **output** (Required, once)
  - Output type, either ‘txt’ (for text output) or ‘xml’ (for XML output).
  - Type: String
- **sig** (Required, once)
  - API authentication hash signature.
  - Type: String
- **userid** (Required, once)
  - The login name of the user initiating the request.
  - Type: String

Example Response

Sample XML output:

```
<?xml version='1.0'?>
<cubit version='1'>
  <profile revision="1838" version="5">solaris10_x86</profile>
</cubit>

Sample text output:
solaris10_x86  1838  5

If user is unauthorized:
<?xml version='1.0'?>
<cubit version='1'>
    <error>You do not have permissions to view this host</error>
</cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

HostGetMyAssignedProfile

Get the profile and version assigned to the host invoking this method.

URL

```
/cubit_api/1/my_profile
```

Authentication

This method does not require authentication.

Parameters

- **output** (Required, once)
    - Output type, either ‘txt’ (for text output) or ‘xml’ (for XML output).
    - Type: String

Example Response

Sample XML output:

```
<?xml version='1.0'?>
<cubit version='1'>
  <profile revision="26" version="2">rhel3_base</profile>
</cubit>

Sample text output:
rhel3_base  26  2
```

Response Codes

- 200 - ok

HostPerfColl

Get performance collection data from remote hosts managed by Lab Management.

This method takes one parameter, `data`. The hostname is auto-derived using reverse DNS. There is no other way to verify the sending host.

The currently supported statgroups are:

- **net** - Returns a dictionary of network interface transfer metrics. The key is the name of the interface. Value is a tuple containing: (`bytes_in`, `bytes_out`, `errors_in`, `errors_out`, `pkts_in`, `pkts_out`).
- **loadavg** - Returns a tuple containing the (1, 5, 15) minute load average on the system.
- **storage** - Returns a dictionary of disk storage metrics. The key is the name of the mounted partition and the value is a tuple of: (`block_size`, `blocks_avail`, `blocks_used`, `block_errors`).
- **systatd** - Returns a string containing the systatd output file for the system.

URL

`/cubit_api/1/host_perfcoll`

Authentication

This method does not require authentication.

Parameters

This method does not take any parameters.

Response Codes

This method does not have any documented response codes.

HostPowercycle

Powercycle a host.

This method immediately executes a powercycle of the host. No graceful shutdown is run on the host at the operating system level. It is recommended that this method only be run when the normal operating shutdown/reboot sequence has failed or is unavailable. You can only powercycle machines in the `Allocated`, `Free`, and `Immutable` states.

URL

`/cubit_api/1/powercycle`

Authentication

This method requires authentication using an API key.

Parameters

- **force** (zero or once)
  - If the machine being powercycled is a virtual host, the force option is required. The only valid value for this parameter is `True`.
  - Type: String
- **host** (Required, once)
  - Fully qualified hostname to powercycle.
  - Type: String
- **sig** (Required, once)
  - API authentication hash signature.
  - Type: String
- **userid** (Required, once)
  - The login name of the user initiating the request.
  - Type: String

Example Response

Successful completion of powercycle:

```
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
</cubit>

If the user is unauthorized, or the host does not exist:
<?xml version='1.0'?>
<cubit version='1'>
```

```
    <error>You are not authorized to powercycle this host.↙error>
↙cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

HostRebuildCancel

Cancel a rebuild for a host.

The host must be in `Rebuild` state for this method to work. In order to cancel a rebuild for a host, the user must have permission to rebuild the host, that is, they must either:

- Have the host allocated to them
- Be a Lab Management Project Admin in the project, or
- Be a Lab Management Domain Admin

URL

`/cubit_api/1/rebuild_cancel`

Authentication

This method requires authentication using an API key.

Parameters

- **host** (Required, once)
    - Fully qualified hostname to cancel rebuild of. Hosts must be in the `Rebuild` state.
    - Type: String
- **sig** (Required, once)
    - API authentication hash signature.
    - Type: String
- **userid** (Required, once)

    - The login name of the user initiating the request.
    - Type: String

Example Response

Successful host rebuild cancel:

```
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
</cubit>

If the user is unauthorized:
<?xml version='1.0'?>
<cubit version='1'>
    <error>You are not authorized to cancel a rebuild for this host.</error
>
</cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

HostRebuild

Rebuild a host with the currently selected profile and profile version.

The user must be authorized to rebuild the host. The host must also not be in any of the states which prohibit rebuilding, such as `Immutable`, `Powercycle`, `Rebuild`, or `Rebuilding`, or this method will fail.

The host must also indicate that it is rebuildable (e.g., not an EC2 host), or this method will fail.

URL

`/cubit_api/1/rebuild`

Authentication

This method requires authentication using an API key.

Parameters

- **host** (Required, once)
    - Fully qualified hostname to rebuild.
    - Type: String
- **sig** (Required, once)
    - API authentication hash signature.
    - Type: String
- **userid** (Required, once)
    - The login name of the user initiating the request.
    - Type: String

Example Response

Successful initiation of rebuild:

```
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
</cubit>

If the user is unauthorized, or the host does not exist:
<?xml version='1.0'?>
<cubit version='1'>
    <error>You are not authorized to rebuild this host.</error>
</cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

PblChangeDesc

Project Build Library (PBL) file and directory description changing interface. This function is used to change the description for files and directories in the Project Build Library. This function may be useful if you are building your own PBL client.

URL

`/cubit_api/1/pbl_changedesc`

Authentication

This method requires authentication using an API key.

Parameters

- **comment** (zero or once)
  - An optional comment to leave about the operation being performed. The comment will not appear in the PBL, but it will be in the audit log entry for this event.
  - Type: String
- **desc** (Required, once)
  - The new text description of the file. This description will completely replace any description currently set for the file.
  - Type: String
- **path** (Required, once)
  - The path to the file being operated on. For example, if the complete file URL is `/pbl/zork/pub/foo/bar/test.txt`, the path would be `/foo/bar/test.txt`.
  - Type: String
- **proj** (Required, once)
  - The name of the project which contains the file we are operating on.
  - Type: String
- **sig** (Required, once)
  - API authentication hash signature.
  - Type: String
- **type** (Required, once)
  - The type of file that we are operating on. Valid values are ‘pub’ and ‘priv’.
  - Type: String

- **userid** (Required, once)
    - The login name of the user initiating the request.
    - Type: String

Example Response

```
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
</cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

PblDelete

Project Build Library (PBL) file and directory delete interface. This function is used to permanently remove files and directories in the Project Build Library. This function may be useful if you are building your own PBL client.

URL

`/cubit_api/1/pbl_delete`

Authentication

This method requires authentication using an API key.

Parameters

- **comment** (zero or once)
    - An optional comment to leave about the operation being performed. This will not appear in the PBL, but it will be in the audit log entry for this event.
    - Type: String
- **dryrun** (Required, once)

  - If this parameter is set to True, the specified path will not actually be deleted. The only valid value for this parameter is `True`.
  - Type: String

- **force** (zero or once)
  - If the force option is present and set to True, and the specified path argument is a directory, a recursive delete of the directory will be performed. The only valid value for this parameter is `True`.
  - Type: String

- **path** (Required, once)
  - The path to the file being operated on. For example, if the complete file URL is `/pbl/zork/pub/foo/bar/test.txt`, the path would be `/foo/bar/test.txt`.
  - Type: String

- **proj** (Required, once)
  - The name of the project which contains the file we are operating on.
  - Type: String

- **sig** (Required, once)
  - API authentication hash signature.
  - Type: String

- **type** (Required, once)
  - The type of file that we are operating on. Valid values are ‘pub’ and ‘priv’.
  - Type: String

- **userid** (Required, once)
  - The login name of the user initiating the request.
  - Type: String

Example Response

```
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
</cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

PblMove

Project Build Library (PBL) file and directory move interface. This function may be useful if you are building your own PBL client.

URL

`/cubit_api/1/pbl_move`

Authentication

This method requires authentication using an API key.

Parameters

- **comment** (zero or once)
  - An optional comment to leave about the operation being performed. This will not appear in the PBL, but it will be in the audit log entry for this event.
  - Type: String
- **destpath** (Required, once)
  - The path to move the destination file to. For example, if the complete file URL is `/pbl/zork/pub/foo/bar/test.txt`, the srcpath would be `/foo/bar/test.txt`. Two important things to note about this parameter:
    - If you specify a path which does not exist, that path will be automatically created for you as part of the move.
    - If the `destpath` parameter ends with a slash ("/"), the destination will be assumed to be a directory. If it does not end with a slash, the destination will be assumed to be a file.
  - Type: String
- **destprj** (zero or once)

  - The destination project for the file. You must have permissions to upload files into both the `srcprj` and the `destprj`, or the move will fail. If not specified, defaults to `srcprj`.
  - Type: String

- **desttype** (zero or once)
  - The destination type of the file. Valid values are 'pub' and 'priv'. If not specified, defaults to `srctype`.
  - Type: String

- **force** (zero or once)
  - If the force option is present, the file will be moved even if a file with that name currently exists, and the previous file will be deleted and replaced with this file. The only valid value for this parameter is `True`.
  - Type: String

- **sig** (Required, once)
  - API authentication hash signature.
  - Type: String

- **srcpath** (Required, once)
  - The path to the source file being operated on. For example, if the complete file URL is `/pbl/zork/pub/foo/bar/test`.txt, the srcpath would be `/foo/bar/test.txt`.
  - Type: String

- **srcprj** (Required, once)
  - The project in which the source file is located.
  - Type: String

- **srctype** (Required, once)
  - The source type of the file. Valid values are 'pub' and 'priv'.
  - Type: String

- **userid** (Required, once)
  - The login name of the user initiating the request.
  - Type: String

Example Response

```
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
</cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

PblUpload

Project Build Library (PBL) file and directory upload interface. This function is used to create files and directories in the Project Build Library. Unless you are building your own PBL client, this function will not be very useful to you.

URL

`/cubit_api/1/pbl_upload`

Authentication

This method requires authentication using an API key.

Parameters

- **comment** (zero or once)
  - An optional comment to leave about the operation being performed. This will not appear in the PBL, but it will be in the audit log entry for this event.
  - Type: String
- **desc** (zero or once)
  - The text description of the file.
  - Type: String
- **file** (Required, once)

  - URL-encoded name and contents of the file, encoded as per RFC 1867.
  - Type: String

- **force** (zero or once)

  - If the force option is present, the file will be uploaded even if a file with that name currently exists. A file cannot be uploaded over a directory of the same name, even if the force parameter is present. The only valid value for this parameter is `True`.
  - Type: String

- **md5sum** (Required, once)

  - The md5 checksum of the file being uploaded.
  - Type: String

- **path** (Required, once)

  - The path to the file being operated on. For example, if the complete file URL is `/pbl/zork/pub/foo/bar/test.txt`, the path would be `/foo/bar`.

    > **NOTE:** The path parameter is used slightly differently in this method than in other methods, since the filename is not appended to the end of the parameter.

  - Type: String

- **proj** (Required, once)

  - The name of the project which contains the file we are operating on.
  - Type: String

- **sig** (Required, once)

  - API authentication hash signature.
  - Type: String

- **type** (Required, once)

  - The type of file that we are operating on. Valid values are ‘pub’ and ‘priv’.
  - Type: String

- **userid** (Required, once)

    - The login name of the user initiating the request.
    - Type: String

Example Response

```
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
</cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

ProfileAdd

Add a profile to the system.

The following conditions must hold for a profile to be added by name:

- The `profile_name` must be specified.
- The `profile_file` must not specified.
- The profile must already exist in SVN.
- The user calling this method must be a Domain Admin.

The following conditions must hold for a profile to be uploaded:

- The `profile_file` must be specified.
- The `profile_name` must not be specified.
- The user calling this method must have permission to add profiles to the specified project.

URL

`/cubit_api/1/add_profile`

Authentication

This method requires authentication using an API key.

Parameters

- **can_users_modify** (Required, once)
  - Set this to `True` if user's with Root Access or better should be able to make modifications to the project. If set to `False`, only the owning user and users with Project Admin access (or better) will be able to modify the profile.
  - Type: String
- **is_prebuilt** (Required, once)
  - Set this to `True` if this profile is a prebuilt image, `False` if this profile is installed via a network install method.
  - Type: String
- **is_public** (Required, once)
  - If the profile is public (i.e can be built by any project). Valid value can be either `True` or `False`.
  - Type: String
- **owner** (zero or once)
  - Lab Management user that this profile will be associated with. The user must be a member of the owning project in order to be able to modify the profile. If unset, this profile will not belong to any particular user.
  - Type: String
- **profile_file** (zero or once)
  - The filename of a profile to be uploaded. The specified file must:
    - Exist on your local system.
    - Contain a valid profile name. Profile names can only contain letters, numbers, underscores ("_"), and dashes ("-"). Profile names are case sensitive.
    - Not currently exist as a profile name on the system.
    - Be a valid XML file, and contain all the needed attributes for a Lab Management profile.
  - Type: String
- **profile_name** (zero or once)
  - The name of a profile that already exists in Lab Management's Subversion Repository.

    - Type: String

- **project** (zero or once)
    - Project that will own this profile. If unset, this profile will not belong to any particular project.
    - Type: String

- **sig** (Required, once)
    - API authentication hash signature.
    - Type: String

- **summary** (Required, once)
    - Summary of the purpose of this profile.
    - Type: String

- **userid** (Required, once)
    - The login name of the user initiating the request.
    - Type: String

Example Response

Successful addition of the profile rhel3_base:

```
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
    <output>rhel3_base</output>
</cubit>

If the user is unauthorized:
<?xml version='1.0'?>
<cubit version='1'>
    <error>Failed to add profile rhel3_base.</error>
    <output>User 'userid' does not have Domain Admin rights.</output>
</cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

ProfileDelete

Delete a profile from the system.

Profiles can be deleted under the following conditions:

- No host has ever been built or is currently building with this profile.
- The user calling this method has permission to modify this profile.

URL

`/cubit_api/1/delete_profile`

Authentication

This method requires authentication using an API key.

Parameters

- **profiles** (Required, once)
  - List of comma-separated, profile names to delete. Profiles must never have been used to build any hosts. The user invoking this API must have modify rights on the profiles.
  - Type: String
- **sig** (Required, once)
  - API authentication hash signature.
  - Type: String
- **userid** (Required, once)
  - The login name of the user initiating the request.
  - Type: String

Example Response

Successful delete of the profile rhel3_base:

```
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
```

```
        <output>rhel3_base↙output>
    ↙cubit>

    If the user is unauthorized:
    <?xml version='1.0'?>
    <cubit version='1'>
        <error>0 of 1 profiles deleted successfully. Failed profiles below.↙erro
r>
        <output>rhel3_base: You do not have modify rights on this profile.↙outp
ut>
    ↙cubit>

    If a deletion of profiles partially succeeds:
    <?xml version='1.0'?>
    <cubit version='1'>
        <error>7 of 8 profiles deleted successfully. Failed profiles below.↙erro
r>
        <output>rhel3_base: This profile has been used to build hosts.↙output>
    ↙cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

ProfileGetMyPkglist

List the packages associated with the profile assigned to the calling host.

URL

```
/cubit_api/1/getmypkglist
```

Authentication

This method does not require authentication.

Parameters

- **output** (Required, once)
    - Output type, either ‘txt’ (for text output) or ‘xml’ (for XML output).
    - Type: String

Example Response

Sample XML output:

```
<?xml version='1.0'?>
<cubit version='1'>
  <package version="1.4.3-6">libtool-libs</package>
  <package version="2.1.15-10">cyrus-sasl-md5</package>
  <package version="20020927-11.30.4">iputils</package>
  <package version="3.03-28">perl-HTML-Tagset</package>
  <package version="1.4.1-2">python-optik</package>
  ...
  <package version="1.7-23">time</package>
  <package version="3.2.3-54">cpp</package>
  <package version="8.3.5-92.4">tcl</package>
  <package version="0.2.3-7.1">audiofile</package>
</cubit>

Sample text output:
libtool-libs  1.4.3-6
cyrus-sasl-md5  2.1.15-10
iputils  20020927-11.30.4
perl-HTML-Tagset  3.03-28
python-optik  1.4.1-2
...
time  1.7-23
cpp  3.2.3-54
tcl  8.3.5-92.4
audiofile  0.2.3-7.1
```

Response Codes

- 200 - ok

ProfileGetPkglist

List packages associated with a particular version or revision of a profile.

If neither version nor revision is set, the latest (HEAD) revision of the profile is selected. You must have permissions to view the profile in order to retrieve a package listing. The profile must be public, or you must be a valid user in the project that owns the profile.

URL

```
/cubit_api/1/getpkglist
```

Authentication

This method requires authentication using an API key.

Parameters

- **output** (zero or once)
    - Ouput mode. ‘txt’ or ‘xml’
    - Type: String
- **profile** (Required, once)
    - Profile to get package list for.
    - Type: String
- **revision** (zero or once)
    - Revision of profile to get package list for.
    - Type: Integer
- **sig** (Required, once)
    - API authentication hash signature.
    - Type: String
- **userid** (Required, once)
    - The login name of the user initiating the request.
    - Type: String
- **version** (zero or once)
    - Version of profile to get package list for.
    - Type: Integer

Example Response

```
(output='xml'):

<?xml version='1.0'?>
<cubit version='1'>
  <package version="1.2.3">pkgname1</package>
```

```
      <package version="4.5.6">pkgname2</package>
      <package version="7.8.9">pkgname3</package>
    </cubit>

    (output='txt'):
    pkg1  1.2.3
    pkg2  4.5.6
    pkg3  7.8.9
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

QueryAlloc

Determine, at either a `domain`, `project`, `user`, or `host` level, how hosts are allocated and who they are allocated to.

Users are only allowed to get information about hosts which they have access to see. For example, if you are not a member of a project, you cannot query that project's host allocations. If you query the domain, you will only see allocation information for projects to which you belong. This method is used by Lab Management internally, and it is made available for users as well, since they may find this data useful.

URL

`/cubit_api/1/query_alloc`

Authentication

This method requires authentication using an API key.

Parameters

- **full** (zero or once)
  - If `full=true` then the complete information about the host is printed.
  - Type: String
- **maxhosts** (zero or once)

  - The maximum number of hosts to return for the query. If `type=domain`, this will be the maximum number of hosts to return per project (i.e. if this is set to 5 and there are two projects in the domain, up to 10 hosts may be returned). If `type=host`, this parameter will not be used. If maxhosts is set, the returned hosts will be sorted by the host's name.
  - Type: Integer

- **name** (zero or once)

  - The name of the host, project, or user to be queried. If `type=domain`, this argument is not required. If `type={host,project}` then this argument is required. If `type=user`, this argument is optional, defaulting to the name of the user who you authenticate as. Only Lab Management Domain Admins are allowed to query user allocations for users other than themselves.
  - Type: String

- **sig** (Required, once)

  - API authentication hash signature.
  - Type: String

- **skiphosts** (zero or once)

  - The number of initial hosts that will be skipped over when returning the results. This only has an effect when maxhosts is also used. If type=domain, the hosts skipped are per per project. If `type=hosts`, this paramenter will not be used.
  - Type: Integer

- **type** (Required, once)

  - The type of service. Valid values are `host`, `project`, `user` and `domain`.
  - Type: String

- **userid** (Required, once)

  - The login name of the user initiating the request.
  - Type: String

Example Response

Here is a sample output where we query type=host and

```
name=cu011.dev.cubitdemo.net:

<?xml version='1.0'?>
<cubit version='1'>
  <project name="zork">
    <host version="2">
      <name>cu011.dev.cubitdemo.net↙name>
      <alloc_user>yz↙alloc_user>
      <alloc_project>zork↙alloc_project>
      <labels>
        <label name="cubit_user">yz↙label>
        <label name="cubit_project">zork↙label>
      ↙labels>
    ↙host>
  ↙project>
↙cubit>

Here is a sample output where we query type=host,
name=cu011.dev.cubitdemo.net and full=True:

<?xml version='1.0'?>
<cubit version='1'>
      <project name=zork>
        <host version="2">
<name>cu011.dev.cubitdemo.net↙name>
<alloc_user>yz↙alloc_user>
<alloc_project>zork↙alloc_project>
<hardware>
        <vendor>Collabnet↙vendor>
        <modelname>HP DL380↙modelname>
        <ncpu>2↙ncpu>
        <cpuarch>i386↙cpuarch>
        <cpuname>Xeon↙cpuname>
        <cpumhz>1400↙cpumhz>
        <memsize unit="MB">2048↙memsize>
        <disksize unit="GB">144↙disksize>
↙hardware>
<labels>
  <label name='cfprofile' rev='322' version='4'>rhel3_base↙label>
    <label name='install_mode'>kickstart-net↙label>
  <label name='cubit_state'>Allocated↙label>
  <label name='cubit_alloc_time'>1170799967↙label>
  <label name='cubit_state_mtime'>1170799967↙label>
  <label name='cubit_build_time'>1169682748↙label>
  <label name='cubit_project'>zork↙label>
  <label name='cubit_user'>yz↙label>
  <label name='vmware-server'↗
  <label name='cubit_lom_addr'>cu3-l.sjc.collab.net↙label>
↙labels>
```

```
<groups>
  <aigroup>DEFAULT↙aigroup>
↙groups>
<network>
  <routing>
    <default>
      <ipv4_addr>192.168.202.1↙ipv4_addr>
     ↙default>
  ↙routing>
     <interface>
         <name>eth0↙name>
         <sol_name>e1000g0↙sol_name>
         <mac_addr>00:16:35:c3:9b:ed↙mac_addr>
         <bootproto>static↙bootproto>
         <addr>
             <ipv4_addr>192.168.202.21↙ipv4_addr>
             <ipv4_mask>255.255.255.0↙ipv4_mask>
             <dnsname>cu011.dev.cubitdemo.net↙dnsname>
         ↙addr>
     ↙interface>
↙network>
↙host>

      ↙project>
↙cubit>

Here is a sample output where we query type=domain:

<?xml version='1.0'?>
<cubit version='1'>
  <domain>
  <project name="zork">
      <host>
        <name>cu011.dev.cubitdemo.net↙name>
        <alloc_user>yz↙alloc_user>
        <alloc_project>zork↙alloc_project>
        <labels>
          <label name="cubit_user">yz↙label>
          <label name="cubit_project">zork↙label>
        ↙labels>
      ↙host>
      <host>
        <name>cu012.dev.cubitdemo.net↙name>
        <alloc_user>yz↙alloc_user>
        <alloc_project>zork↙alloc_project>
        <labels>
          <label name="cubit_user">yz↙label>
          <label name="cubit_project">zork↙label>
        ↙labels>
      ↙host>
  ↙project>
  ↙domain>
↙cubit>

Here is a sample output where we query type=domain and full=True:

<?xml version='1.0'?>
<cubit version='1'>
  <domain>
  <project name="zork">
      <host version="2">
<name>cu011.dev.cubitdemo.net↙name>
<alloc_user>yz↙alloc_user>
<alloc_project>zork↙alloc_project>
<hardware>
        <vendor>Collabnet↙vendor>
        <modelname>HP DL380↙modelname>
        <ncpu>2↙ncpu>
        <cpuarch>i386↙cpuarch>
        <cpuname>Xeon↙cpuname>
        <cpumhz>1400↙cpumhz>
        <memsize unit="MB">2048↙memsize>
        <disksize unit="GB">144↙disksize>
↙hardware>
<labels>
  <label name='cfprofile' rev='322' version='4'>rhel3_base↙label>
    <label name='install_mode'>kickstart-net↙label>
  <label name='cubit_state'>Allocated↙label>
  <label name='cubit_alloc_time'>1170799967↙label>
  <label name='cubit_state_mtime'>1170799967↙label>
  <label name='cubit_build_time'>1169682748↙label>
  <label name='cubit_project'>zork↙label>
  <label name='cubit_user'>yz↙label>
  <label name='vmware-server'↗
  <label name='cubit_lom_addr'>cu3-l.sjc.collab.net↙label>
↙labels>
<groups>
  <aigroup>DEFAULT↙aigroup>
↙groups>
<network>
  <routing>
    <default>
      <ipv4_addr>192.168.202.1↙ipv4_addr>
     ↙default>
  ↙routing>
     <interface>
         <name>eth0↙name>
         <sol_name>e1000g0↙sol_name>
         <mac_addr>00:16:35:c3:9b:ed↙mac_addr>
         <bootproto>static↙bootproto>
         <addr>
             <ipv4_addr>192.168.202.21↙ipv4_addr>
             <ipv4_mask>255.255.255.0↙ipv4_mask>
             <dnsname>cu011.dev.cubitdemo.net↙dnsname>
         ↙addr>
     ↙interface>
↙network>
↙host>

      <host version="2">
<name>cu012.dev.cubitdemo.net↙name>
<alloc_user>yz↙alloc_user>
<alloc_project>zork↙alloc_project>
<hardware>
        <vendor>Collabnet↙vendor>
        <modelname>HP DL380↙modelname>
        <ncpu>1↙ncpu>
        <cpuarch>i386↙cpuarch>
        <cpuname>Xeon↙cpuname>
        <cpumhz>1400↙cpumhz>
        <memsize unit="MB">2048↙memsize>
        <disksize unit="GB">144↙disksize>
↙hardware>
<labels>
  <label name='cfprofile' rev='322' version='4'>rhel3_all↙label>
    <label name='install_mode'>kickstart-net↙label>
  <label name='cubit_state'>Allocated↙label>
  <label name='cubit_alloc_time'>1170800012↙label>
  <label name='cubit_state_mtime'>1170800012↙label>
  <label name='cubit_build_time'>1169687903↙label>
  <label name='cubit_project'>zork↙label>
  <label name='cubit_user'>yz↙label>
  <label name='vmware-server'↗
  <label name='cubit_lom_addr'>cu4-l.sjc.collab.net↙label>
↙labels>
<groups>
  <aigroup>DEFAULT↙aigroup>
↙groups>
<network>
  <routing>
    <default>
      <ipv4_addr>192.168.202.1↙ipv4_addr>
     ↙default>
  ↙routing>
     <interface>
         <name>eth0↙name>
         <sol_name>e1000g0↙sol_name>
         <mac_addr>00:16:35:5a:fe:39↙mac_addr>
         <bootproto>static↙bootproto>
         <addr>
             <ipv4_addr>192.168.202.22↙ipv4_addr>
             <ipv4_mask>255.255.255.0↙ipv4_mask>
             <dnsname>cu012.dev.cubitdemo.net↙dnsname>
         ↙addr>
     ↙interface>
↙network>
↙host>
↙project>
↙domain>
↙cubit>
```

Here is a sample output where we query type=project and name=zork:

```
<?xml version='1.0'?>
<cubit version='1'>
  <project name="zork">
      <host version="2">
        <name>cu011.dev.cubitdemo.net↙name>
        <alloc_user>yz↙alloc_user>
        <alloc_project>zork↙alloc_project>
        <labels>
          <label name="cubit_user">yz↙label>
          <label name="cubit_project">zork↙label>
        ↙labels>
      ↙host>
      <host version="2">
        <name>cu012.dev.cubitdemo.net↙name>
        <alloc_user>yz↙alloc_user>
        <alloc_project>zork↙alloc_project>
        <labels>
          <label name="cubit_user">yz↙label>
          <label name="cubit_project">zork↙label>
        ↙labels>
      ↙host>
      <host version="2">
        <name>cu013.dev.cubitdemo.net↙name>
        <alloc_user>grue↙alloc_user>
        <alloc_project>zork↙alloc_project>
        <labels>
          <label name="cubit_user">grue↙label>
          <label name="cubit_project">zork↙label>
        ↙labels>
      ↙host>
      <host version="2">
        <name>cu014.dev.cubitdemo.net↙name>
        <alloc_user>root↙alloc_user>
        <alloc_project>zork↙alloc_project>
        <labels>
          <label name="cubit_user">root↙label>
          <label name="cubit_project">zork↙label>
        ↙labels>
      ↙host>
      <host version="2">
        <name>cu019.dev.cubitdemo.net↙name>
        <alloc_user>root↙alloc_user>
        <alloc_project>zork↙alloc_project>
        <labels>
          <label name="cubit_user">root↙label>
          <label name="cubit_project">zork↙label>
        ↙labels>
      ↙host>
      <host version="2">
        <name>cu022.dev.cubitdemo.net↙name>
        <alloc_user>grue↙alloc_user>
        <alloc_project>zork↙alloc_project>
        <labels>
          <label name="cubit_user">grue↙label>
          <label name="cubit_project">zork↙label>
        ↙labels>
      ↙host>
      <host version="2">
        <name>cu091.dev.cubitdemo.net↙name>
        <alloc_user>root↙alloc_user>
        <alloc_project>zork↙alloc_project>
        <labels>
          <label name="cubit_user">root↙label>
          <label name="cubit_project">zork↙label>
        ↙labels>
      ↙host>
  ↙project>
↙cubit>

Here is a sample output where we query type=project, name=zork and
full=True:

<?xml version='1.0'?>
<cubit version='1'>
  <project name="zork">
      <host version="2">
<name>cu011.dev.cubitdemo.net↙name>
<alloc_user>yz↙alloc_user>
<alloc_project>zork↙alloc_project>
<hardware>
        <vendor>Collabnet↙vendor>
        <modelname>HP DL380↙modelname>
        <ncpu>2↙ncpu>
        <cpuarch>i386↙cpuarch>
        <cpuname>Xeon↙cpuname>
        <cpumhz>1400↙cpumhz>
        <memsize unit="MB">2048↙memsize>
        <disksize unit="GB">144↙disksize>
↙hardware>
<labels>
  <label name='cfprofile' rev='322' version='4'>rhel3_base↙label>
    <label name='install_mode'>kickstart-net↙label>
  <label name='cubit_state'>Allocated↙label>
  <label name='cubit_alloc_time'>1170799967↙label>
  <label name='cubit_state_mtime'>1170799967↙label>
  <label name='cubit_build_time'>1169682748↙label>
  <label name='cubit_project'>zork↙label>
  <label name='cubit_user'>yz↙label>
  <label name='vmware-server'↗
  <label name='cubit_lom_addr'>cu3-l.sjc.collab.net↙label>
↙labels>
<groups>
  <aigroup>DEFAULT↙aigroup>
↙groups>
<network>
  <routing>
    <default>
      <ipv4_addr>192.168.202.1↙ipv4_addr>
     ↙default>
  ↙routing>
     <interface>
         <name>eth0↙name>
         <sol_name>e1000g0↙sol_name>
         <mac_addr>00:16:35:c3:9b:ed↙mac_addr>
         <bootproto>static↙bootproto>
         <addr>
             <ipv4_addr>192.168.202.21↙ipv4_addr>
             <ipv4_mask>255.255.255.0↙ipv4_mask>
             <dnsname>cu011.dev.cubitdemo.net↙dnsname>
         ↙addr>
     ↙interface>
↙network>
↙host>

      <host version="2">
<name>cu012.dev.cubitdemo.net↙name>
<alloc_user>yz↙alloc_user>
<alloc_project>zork↙alloc_project>
<hardware>
        <vendor>Collabnet↙vendor>
        <modelname>HP DL380↙modelname>
        <ncpu>1</ncpu>
        <cpuarch>i386</cpuarch>
        <cpuname>Xeon</cpuname>
        <cpumhz>1400</cpumhz>
        <memsize unit="MB">2048</memsize>
        <disksize unit="GB">144</disksize>
</hardware>
<labels>
  <label name='cfprofile' rev='322' version='4'>rhel3_all</label>
    <label name='install_mode'>kickstart-net</label>
  <label name='cubit_state'>Allocated</label>
  <label name='cubit_alloc_time'>1170800012</label>
  <label name='cubit_state_mtime'>1170800012</label>
  <label name='cubit_build_time'>1169687903</label>
  <label name='cubit_project'>zork</label>
  <label name='cubit_user'>yz</label>
  <label name='vmware-server'/>
  <label name='cubit_lom_addr'>cu4-l.sjc.collab.net</label>
</labels>
<groups>
  <aigroup>DEFAULT</aigroup>
</groups>
<network>
  <routing>
    <default>
      <ipv4_addr>192.168.202.1</ipv4_addr>
     </default>
  </routing>
     <interface>
         <name>eth0</name>
         <sol_name>e1000g0</sol_name>
         <mac_addr>00:16:35:5a:fe:39</mac_addr>
         <bootproto>static</bootproto>
         <addr>
             <ipv4_addr>192.168.202.22</ipv4_addr>
             <ipv4_mask>255.255.255.0</ipv4_mask>
             <dnsname>cu012.dev.cubitdemo.net</dnsname>
         </addr>
     </interface>
</network>
</host>

      <host version="2">
<name>cu013.dev.cubitdemo.net</name>
<alloc_user>grue</alloc_user>
<alloc_project>zork</alloc_project>
<hardware>
        <vendor>Collabnet</vendor>
        <modelname>HP DL380</modelname>
        <ncpu>1</ncpu>
        <cpuarch>i386</cpuarch>
        <cpuname>Xeon</cpuname>
        <cpumhz>1400</cpumhz>
        <memsize unit="MB">1024</memsize>
        <disksize unit="GB">25</disksize>
</hardware>
<uuid>51 51 bb a0 00 15 15 51-81 9d fb 47 77 00 51 61</uuid>
<labels>
  <label name='cfprofile' rev='322' version='6'>rhel4_all</label>
    <label name='install_mode'>kickstart-net</label>
  <label name='cubit_state'>Allocated</label>
  <label name='cubit_alloc_time'>1170960674</label>
  <label name='cubit_state_mtime'>1170963906</label>
  <label name='cubit_build_time'>1170963906</label>
  <label name='cubit_project'>zork</label>
  <label name='cubit_user'>grue</label>
  <label name='vmware-guest'>cu011.dev.cubitdemo.net</label>
  <label name='cubit_lom_addr'>cu011.dev.cubitdemo.net</label>
</labels>
<groups>
  <aigroup>DEFAULT</aigroup>
</groups>
<network>
  <routing>
    <default>
      <ipv4_addr>192.168.202.1</ipv4_addr>
     </default>
  </routing>
     <interface>
         <name>eth0</name>
         <sol_name>e1000g0</sol_name>
         <mac_addr>00:50:56:00:00:01</mac_addr>
         <bootproto>static</bootproto>
         <addr>
             <ipv4_addr>192.168.202.23</ipv4_addr>
             <ipv4_mask>255.255.255.0</ipv4_mask>
             <dnsname>cu013.dev.cubitdemo.net</dnsname>
         </addr>
     </interface>
</network>
</host>
  </project>
</cubit>
```

Response Codes

- 200 - ok

- 400 - Login failed / Insufficient permissions

QueryMembership

Determine the projects and the hosts that the user has permission to access.

If name parameter is not passed, the permission details of the initiating user will be displayed. Only Lab Management Domain Admins are permitted to query users other than themselves.

URL

`/cubit_api/1/query_membership`

Authentication

This method requires authentication using an API key.

Parameters

- **name** (zero or once)
  - The name of the user to get the permission details. This argument is optional, defaulting to the name of the user who you authenticate as. Only Lab Management Domain Admins are permitted to query users other than themselves.
  - Type: String
- **sig** (Required, once)
  - API authentication hash signature.
  - Type: String
- **userid** (Required, once)
  - The login name of the user initiating the request.
  - Type: String

Example Response

Here is a sample output where we query name=grue:

```
<?xml version='1.0'?>
<cubit version='1'>
```

```
  <user name="grue">
    <project name="testing">
      <summary>Cubit Testing</summary>
      <roles>CUBIT - Domain Admin</roles>
    </project>
    <project name="zim">
      <summary>invador</summary>
      <roles>CUBIT - Domain Admin</roles>
    </project>
    <project name="zork">
      <summary>For dorks</summary>
      <roles>CUBIT - Domain Admin</roles>
    </project>
  </user>
</cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

Status

Return the status of the Lab Management web service. This provides some indication of whether or not the Lab Management web service is working properly. The only status returned is OK.

URL

/cubit_api/1/status

Authentication

This method does not require authentication.

Parameters

This method does not take any parameters.

Example Response

```
<?xml version='1.0'?>
<cubit version='1'>
```

```
        <status>OK</status>
    </cubit>
```

Response Codes

- 200 - ok

StatusSigned

Return the status of the Lab Management web service, and the name of the authenticated user when authentication is used via a signed method.

This provides proof that the API, and authentication to the API are working properly. The only status returned is `OK`. If the authentication is not successful, you will get a `Permission Denied` error. If the API key is expired, you will get a `Permission Denied: API key expired` error.

URL

`/cubit_api/1/status_signed`

Authentication

This method requires authentication using an API key.

Parameters

- **sig** (Required, once)
  - API authentication hash signature.
  - Type: String
- **userid** (Required, once)
  - The login name of the user initiating the request.
  - Type: String

Example Response

```
    <?xml version='1.0'?>
    <cubit version='1'>
        <status>OK</status>
        <userid>grue</userid>
    </cubit>
```

In addition to returning the OK message, the username of the authenticated user is provided.

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

UserCanLogin

Given a userid and a hostname, return whether or not the user is authorized to login provided that the appropriate credentials are presented.

The HTTP response code will always be 200 for a successful query whether or not access is permitted. The document body will be text only, and contain one of the following:

- **0** - no access
- **1** - user level access
- **2** - root level access
- **3** - host owner access
- **4** - project admin access

URL

```
/cubit_api/1/can_login
```

Authentication

This method does not require authentication.

Parameters

- **host** (Required, once)
    - The fully qualified hostname of the host to test for access.
    - Type: String
- **userid** (Required, once)
    - The Lab Management userid of the user to test for access.

    - Type: String

Example Response

```
3
```

Response Codes

- 200 - ok

Use the TeamForge Lab Management API

The Lab Management Web Services API is a set of callable methods that enable your build and test systems to integrate with Lab Management's capabilities to manage infrastructure for software development.

The API requires a client which speaks at least HTTP, if you are going to be making requests only from the secure local network that Lab Management nodes are located on. HTTPS support is required to make API requests from anywhere else. It is recommended that you use HTTPS for your API requests whenever possible.

The Lab Management API is implemented as a client distributed with Lab Management, `cubit_api_client.py`. This client runs on any platform that Python runs on. Pre-compiled executables, which do not require Python, are available for Windows. The client is open-source, and you are free to modify it and redistribute it in accordance with its license terms. You can use it to build the basis of your own client in Python or any other language

> **NOTE:** Use of Lab Management's Web Services API is governed under the same terms as your use of the rest of TeamForge Lab Management. While we do not place any restrictions on the number of API calls any user can make in any time period, we request that users make use of only the API calls that they really need, and we reserve the right to limit the access of users who overuse the service.

1. Generate a new API key or view your current API key from your Lab Management **Start** page.

    > **IMPORTANT:** Your API key allows you to execute any API calls within Lab Management as you, as if you were logged into the Web interface. Keep your key safe using the same types of precautions you would use for your password.

2. Set yourself up to use *signed* API methods. Lab Management API methods are *signed* to avoid embedding your API key in the URL body of requests and keep it secure from snooping.

  1. Make a request to the server for a *token*. This will be used along with your API key to encrypt the arguments.

  2. Encode the arguments into a hash using a known secret (your API key, plus your token). We call this hash the *signature*.

  3. Make a second request to the server, passing the *signature* and the list of *key/value argument pairs*. If your key/value pairs contain unicode data, the hashed list of arguments must be UTF-8 encoded, otherwise, UTF-8 encoding of the argument hash is optional.

3. To use a signed method, pass the following command-line options to the `cubit_api_client.py` program:

-s	This indicates that the method is a signed method, and an authentication should be performed.
--api-user|-u username	This argument specifies the username to authenticate as.
--api-key|-k api_key	This argument specifies your API key. You can generate and view your API key from your Lab Management Start Page. You can change your API key at anytime independently of your Lab Management password.

In addition to these required arguments, most web services require other parameters. These are specified as space-separated key=value pairs. For example, consider the following command, which will allocate the host “cu012.cubit.domain” to the user “alice” in the project “webtesting”, while authenticating as the user “bob” with bob’s API key:

```
cubit_api_client.py --api-url=http://cubit.domain/cubit_api/1 --api-user=b
ob --api-key=713cdf90-2549-1350-80c3-2d0bcf9a1142 -s alloc_host  host=cu01
2.cubit.domain alloc_user=alice alloc_proj=webtesting
```

4. To use an unsigned method, fetch the following URL: `https://<cubit.domain>/cubit_api/1/status"`↗(where `<cubit.domain>` is the domain name of your Lab Management site).

> **TIP:** The `status` method is the most basic of all the API methods, and will always return an OK string as a response.

To demonstrate the unsigned status method with `cubit_api_client.py`:

```
cubit_api_client.py -l http://cubit.domain/cubit_api/1 status
                    cubit_api_client.py: OK
```

Again, this time with XML output:

```
cubit_api_client.py -l http://cubit.domaincubit_api/1 --xml status
                    <?xml version='1.0'?>
                    <cubit version='1'>
                    <status>OK↙status>
                    ↙cubit>
```

Is the CollabNet TeamForge SOAP API backward-compatible?

The CollabNet TeamForge Enterprise Edition 5.x and CollabNet TeamForge Enterprise 6.x APIs (including Service Pack updates) are fully compatible with CollabNet TeamForge.

Applications developed using these earlier APIs will continue to function after you upgrade to CollabNet TeamForge. A release of CollabNet TeamForge may be accompanied by updates to the API. These changes are always backward compatible with earlier versions of the API. However, the calls from the different API versions are not interchangeable.

- For updates and patch releases, existing methods are incremented, not overwritten. For example, if an update is made to the `createDocument` method, it is reflected in a new method `createDocument1`. You do not have to update your existing applications to reflect a new API with any Service Pack or Hot Fix release.
- For a major CollabNet TeamForge release, such as this 18.0 release, updates are merged into a new API version. At this point, if you want to use the new API calls, you must update your existing applications.

How should my API client store user passwords?

Any reputable method of storing passwords will work, as long as your site is protected by SSL.

All client tools rely on the TeamForge SOAP API for authentication, and therefore use whatever authentication method TeamForge is using.

> **IMPORTANT:** Because SOAP is simply XML transmitted over HTTP, all values are sent in clear text. For that reason it is very important that your TeamForge site be SSL-enabled and protected by server-side SSL certificates. This will ensure that any usernames or passwords sent from a client tool will be encrypted.

Many standalone client tools are able to cache a copy of the user’s credentials to make it easier for them to access the site. The CollabNet Eclipse Desktop stores passwords in the encrypted Java keystore, and the CollabNet Windows clients use the Windows keystore.

CollabNet’s Subversion clients and other Subversion clients, such as Tortoise and Subclipse, are also able to store user credentials. While CollabNet has no control over how third-party tools store such credentials, it our experience that the mainstream tools all use an appropriate keystore for secure storage of user credentials. CollabNet recommends that customers independently verify the storage methods of those tools and set a policy appropriate with their own security guidelines.

Subversion users on Linux systems have the option to use the Gnome keyring to securely store user credentials. CollabNet recommends that customers set their own policy for how their users should use the Linux Subversion client.

How does an application interact with TeamForge SOAP services?

TeamForge exposes a subset of the APIs defined by the application server as web services, through the SOAP protocol.

SOAP APIs

A SOAP proxy server and a SOAP API layer, both running on Apache Axis, expose a set of web services representing each TeamForge application.

The SOAP server provides the following functions:

- Provides web services by accepting SOAP requests from the clients.
- Performs SOAP client authentication.
- Implements TeamForge role-based access control (RBAC) and caching services.
- Accesses the application server via RMI stubs.

While each TeamForge service has its own SOAP interface, interaction with all the services is designed to be as consistent as possible. The calls for each service are similar, although the data format and specific call parameters may be different.

For example, the following calls are consistent across all services:

- list
- get
- set
- delete
- create

The call parameters, however, are different. For example:

- When working with the CollabNet service, you might call `getProjectList(string sessionID)`.

- When working with the TaskApp service, you might call `getTaskList(string sessionID, string taskFolderID)`.

User-centric services

All services and APIs are user-centric, meaning that all integrated applications must establish an individual connection to the SOAP server for each user. This differs from programming directly with an application server where one connection can be established for any number of users.

Activities that can be performed using the SOAP interface are by definition user-based, such as retrieving a list of a user's projects, tasks, or assigned tracker artifacts. These activities therefore require an individual connection for each user.

Requiring individual connections also ensures that role-based access control is checked for each action performed by each user. To ensure that security is enforced, RBAC checks are performed on each SOAP API call and cannot be disabled at the client level.

How do I enable/disable path-based permissions via SOAP?

The Path Based Permissions (PBP) are handled via the *roleList, *Cluster methods of rbacAppSoap. To enable PBP, use the “scm_fgp” (fine grained permissions) argument to addCluster.

See the below psudeocode.

```
                from com.collabnet.ce.soap60.webservices import *
            from com.collabnet.ce.soap60.webservices.ClientSoapStubFactory imp
ort getSoapStub
            from com.collabnet.ce.soap60.types import *

            hostname = "http://server/"
            username = "admin"
            password = "admin"
            project = "proj1007"
            roleName = "tracker"

            sfSoap = getSoapStub(cemain.ICollabNetSoap, hostname)
            sfSession = sfSoap.login(username,password)
            rbacAppSoap = getSoapStub(rbac.IRbacAppSoap, hostname)

            roles = rbacAppSoap.getRoleList(sfSession,project).getDataRows().t
olist()
            roleId = None
            for row in roles:
            if row.getDescription() == roleName :
            roleId = row.getId()
            print "found tracker role, %s" % roleId

            if roleId == None:
            raise Exception("Cant find role ")

            clusters = rbacAppSoap.listClusters(sfSession, roleId).getDataRows
().tolist()
            for row in clusters:
            print row.getFolderId(), row.getOperationClusterName()
            if row.getOperationClusterName() == "scm_commit":
            print "found target!"
            rbacAppSoap.removeCluster(sfSession, roleId, row.getOperationClust
erName(), row.getFolderId())
            rbacAppSoap.addCluster(sfSession, roleId, "scm_fgp", row.getFolder
Id())

            sfSoap.logoff(username,sfSession)
```

Here is some stuff you may need to know to work with integrated applications.

IntegratedAppSupport

CollabNet provides servlet helper libraries that take care of authenticating integrated application requests with TeamForge. They also provide context information for each subsequent form processing within the application or links clicked within the application.

The heart of this is the `IntegratedAppSupport` class, which must be called during every request of an integrated application. This class takes the HttpServletRequest and HttpServletResponse for each request and determines whether the user is already authenticated. It also provides project- and user-related information that can be used throughout the request.

> **TIP:** It is a good idea to store this as a ThreadLocal so that it can be used from anywhere in the application.

IntegratedAppSupport, on successful validation, provides several objects that can be used through the integrated application. .

Parameters

The method is specified within parentheses at the end of each parameter.

SoapSessionId

This identifies a user. It can be used for making any TeamForge webservice calls over SOAP. TeamForge expects that every call be associated with a valid SoapSessionId. This method lets you fetch the SoapSessionId for the current session of the user. (getSoapSessionId())

WebSessionId

This is the jsessionid for the current user. This is the servlet-container-specific id that might be used for some integrated applications. (getWebSessionId())

CtfBaseUrl

Retrieves the base URL for Collabnet TeamForge (getCtfBaseUrl()).

ProjectPath

Retrieves the TeamForgeProject path for this project. This would typically be “projects.” (getProjectPath())

CollabNetSoap handle

Retrieves the handle for CollabNet SOAP (Refer to ctf_xxx_sdk.zip for SOAP calls that can be made using TeamForge) (getCollabNetSoap()). Documentation for the SOAP methods exposed are available in ctf_xxx_sdk.zip.

PluggableAppSoap handle

Retrieves the handle for pluggable application SOAP calls (getPluggableAppSoap()).Documentation for the SOAP methods exposed are available in ctf_xxx_sdk.zip.

RbacAppSoap

Retrieves the handle for RBAC SOAP Calls (getRbacAppSoap()).Documentation for the SOAP methods exposed are available in ctf_xxx_sdk.zip.

IntegratedAppId

Retrieves the linkid for this request. (getIntegratedAppId())

IntegratedAppPrefix

Retrieves the Prefix to use for go-urls, associations and linkifications (getIntegratedAppPrefix()).

IntegratedAppName

Retrieves the name of this integrated app (getIntegratedAppName()).

ProjectId

Retrieves the TeamForge project id (getProjectId()).

userSoapDO

Retrieves the SOAP data object information for the current user (getUserSoapDO()).

SOAP Calls for Integrated Applications

When you integrate an application with TeamForge, you will need to make SOAP calls from the integrated application into TeamForge to respond to events on either side.

To integrate an application with TeamForge using the Integrated Application Framework, you have to implement the following methods for TeamForge to communicate with the integrated application.

The interface should be exposed via SOAP, and the endpoint for this interface should be defined by the `<endpoint>` tag in the XML Application configuration file.

> **TIP:** There is no direct API for managing users/groups from TeamForge to an integrated application. However, if your integrated application has the notion of users and user groups, you can listen to “user events” or “group events” through an event handler in TeamForge and call this interface using that data. You can also make SOAP calls from the integrated application directly into TeamForge to get user information.

SOAP Calls for Adding, Editing or Deleting an Integrated Application

You can use these SOAP calls to enable project administrators to add an integrated application to their projects, edit an integrated application, or delete it from their projects.

createProjectConfig

This method is called when an integrated application is added to a project from **Project Admin > Tools > Add Tool**.

editProjectConfig

This method is called when editing an integrated application in a project from **Project Admin > Tools**.

deleteProjectConfig

This method is called when deleting an integrated application in a project from **Project Admin > Tools**.

SOAP Calls for SCM in Integrated Applications

You can use these SOAP calls to enable users to change data in an integrated application based on actions in a TeamForge source code repository.

scmPreCommit

This serves as the pre-Commit hook for an integrated application. TeamForge will call this method if require-scm-integration flag is set to true in the integrated app xml descriptor. This would mean that a commit made into TeamForge will only succeed if this method returns a “true” (String).

> **NOTE:** All TeamForge ids and ids from other integrated applications are removed from this list even though it could be part of the original commit message. Also these ids are without the prefix. The integrated application can use these ids to find if they are valid for the particular commit and respond with a `true` or a `false` followed by an optional error message that can be displayed to the user who is making the commit.

scmPostCommit

This method is called after a commit is made and post processing needs to happen for that particular commit in the integrated application. A typical use case will be to store the commit information as part of some object within the integrated application. Please note that any errors thrown as part of this will not stop the commit from happening as the commit has been already made.

SOAP Calls for Search in Integrated Applications

The **getSearchResults** method is called by TeamForge as part of doing a regular search. Use this method to include data from your integrated application in TeamForge search results.

SOAP Calls for Project Templates in Integrated Applications

You can use these SOAP calls to enable an integrated application to be included in project templates created by TeamForge.

createTemplate

This method is called when a TeamForge template is created on a project that contains this integrated application.

getTemplateMetadata

This method is called when displaying the template content in “Project Tools/Included in Template” section at the time of template creation.

getTemplateContent

This method is called when viewing the content of an existing template.

validateParametersForTemplatizedProject

This method is called to validate the configuration parameters provided when creating a project from a template.

createTemplatizedProjectConfig

This method is called when a project is created from a template and this integrated application is part of the template. This is the equivalent of the “createProjectConfig” call except that it can be called when a project gets created from a template.

TeamForge EventQ provides a message queue (MQ) based interface for adding custom adapters and the Extensible Data Source (XDS) system for adding new product domains and tools.

Overview

TeamForge EventQ APIs can be classified into three groups: Extensibility APIs, Reporting APIs and HTTP APIs.

- Extensibility APIs: To add new integrations to EventQ such as Commits API, Builds API, Review Requests API and so on.
- Reporting API: To retrieve data from the EventQ database.
- HTTP APIs: To administrate EventQ programmatically.

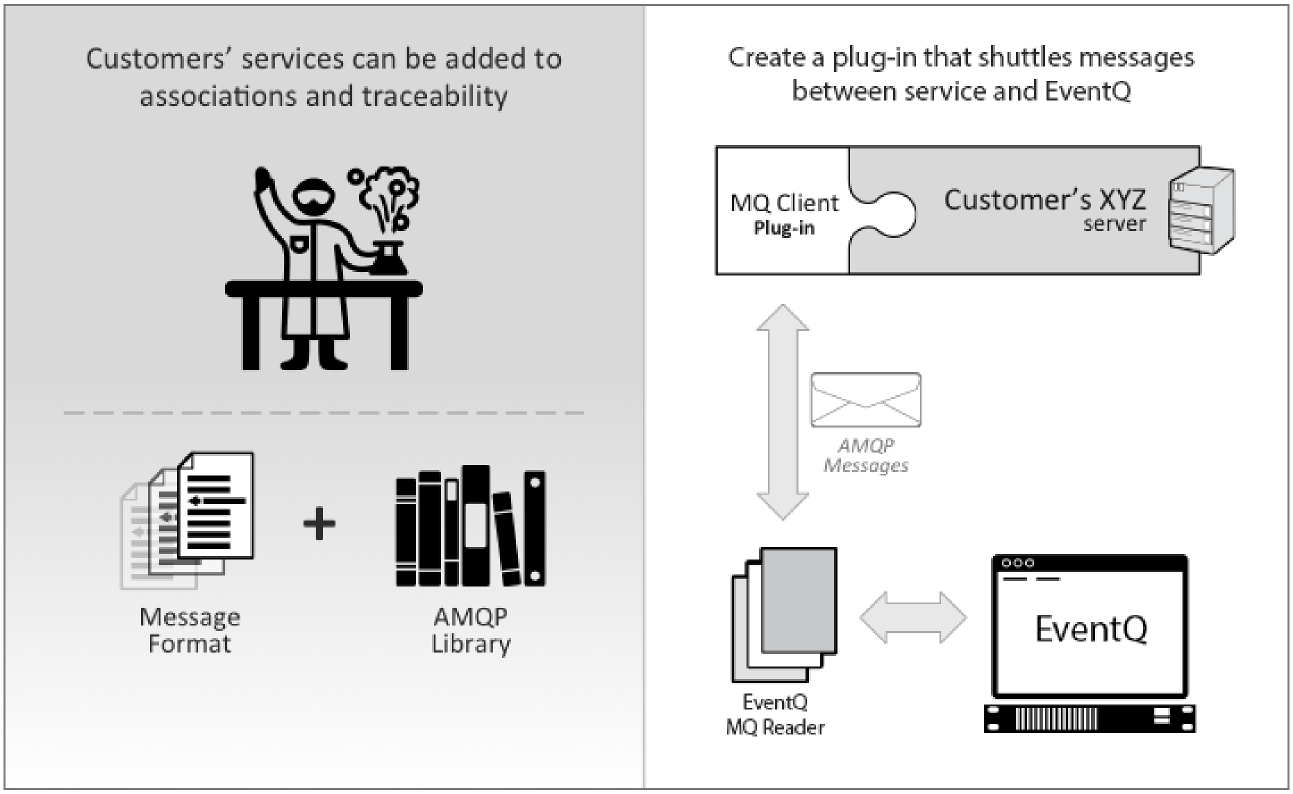
EventQ can be extended via the Message Queue (MQ) API. The MQ API is provided for purposes of creating custom adapters for stock product classes/domains; build, commit, code review, and work item sources can post activity information to the corresponding message queues using the formats described in the API Documentation (see references below).

New product classes may be established using the Extensible Data Source API, or XDS API. For instance, TeamForge EventQ does not have stock product class for “deployment”, so the XDS API may be used to establish a new product domain for “deployment” which can then be used like other sources.

MQ API — Writing Your Own Adapters

Each TeamForge EventQ instance has an associated “message queue” service, which is used to collect and queue data from integrated tools. Extensibility is achieved by writing “adapters” that communicate specially formatted messages to the TeamForge EventQ message queue service.

The message communication protocol is called “AMQP” and the RabbitMQ web site links to open source AMQP protocol libraries that can be used to create adapters for numerous languages and platforms. The API documentation referenced below houses detailed message format specifications, example messages, and code samples for interacting with AMQP libraries.


Customers' services can be added to associations and traceability
Message Format
+
AMQP Library
Create a plug-in that shuttles messages between service and EventQ
MQ Client
Plug-in
Customer's XYZ
server
AMQP Messages
EventQ MQ Reader
EventQ

TeamForge EventQ is extensible via MQ for these types of tools: SCM and version control, continuous integration/build, code review, and work item systems.

XDS API — Authoring New Product/Tool Domains

EventQ can be configured to accept data from a wide range of sources through the XDS API. In addition to the stock activity APIs defined for commits, builds, reviews, and work items, the XDS feature enables you to create your own activity APIs by defining and registering an “XDS Schema” that describes the expected activity message format for a particular tool or product domain in question. Once an XDS Schema is registered with EventQ, adapters can send EventQ activity messages in compliance with the registered schema and the resulting XDS activities may show up in activity streams and as associations. In this way, XDS Schemas give users a way to publish new activity APIs for arbitrary, user-defined data sources.

XDS Schemas can be added to EventQ either through the MQ or HTTP API. See Extensible Data Sources (XDS) Walk-through and refer to the API documentation.

Once a Schema is established, “XDS Sources” can be added through the “Custom” step type. See Extensible Data Source (XDS) Overview.

Authentication

For the MQ interface, authentication is provisioned by the RabbitMQ server. When a new source is created, a unique username/password combination is created and exposed on the EventQ source edit screen. These credentials must be supplied by the posting client. These user credentials are independent of the TeamForge user store and belong only to RabbitMQ.

Extensibility APIs

Commits API

Submission Parameters

Submission Queue

- **queue_name**: eventq.commits
- **auto_delete**: false
- **durable**: true

Top Level

Parameter	Required/Optional	Description
api_version	Required	A string that matches the API version this document is for.
source_association_key	Required	A key generated by TeamForge EventQ that links incoming commit messages with the appropriate source server.
	Example: "source_association_key": "31e8dd90-164f-0130-c4d0-406c8f04c05d"	
commit_data	Required	The commit.

Objects

commit_data - The following fileds should be in this commit post.

Field	Required/Optional	Description
repo_id	Optional	A unique identifier for this repository. For example, the repo UUID.
revision_id	Required	A unique identifier for the commit action (eg., revision number or commit id).
branch_name	Optional	Identifier for the applicable repository branch.
type	Required	A string representing the activity type

	Accepted values post-commit - A commit has been completed.	
message	Optional	The message associated with this activity.
changes	Optional	An array of changed items. Required information about affected paths (an array of hashes).
	Accepted values • **path** (Required) - A string with the relative path to the file that created, modified, etc. • **action** (Required) - A string representing the action. ◦ added - Added ◦ deleted - Deleted ◦ modified - Modified ◦ props_modified - File properties (not contents) modified ◦ type_changed - Type changed (e.g., from 'file' to 'symlink', 'symlink' to 'directory', etc.) ◦ copied - Path is copied from another source ◦ renamed - Path is renamed from another source • **from** (Optional) - If the activity is a copy or rename, the path from which it was copied or renamed (if it can be determined or inferred with some accuracy; this is not guaranteed to be absolutely correct)	
event_time	Required	A string containing a timestamp in UTC timezone and RFC 3339 format.
created_by	Required	The user name of the person who took the action.

Examples

The following code examples are Copyright 2015 CollabNet, Inc., licensed under the Apache License, Version 2.0 (the “License”); you may not use this code except in compliance with the License. You may obtain a copy of the License at [http://www.apache.org/licenses/LICENSE-2.0]

Sample Submission JSON

```
{
      "api_version": "1",
      "source_association_key" : "150e8951-19e1-4066-8b74-601026dd6cde",
      "commit_data" : {
        "repo_id": "390e8951-19e1-4066-8b74-601026dd6cde",
        "revision_id": "2315",
        "branch_name": "master",
        "type": "post_commit",
        "message": "here's my commit [orc-277]",
        "changes": [
          { "path": "/source/file/1", "action": "modified" },
          { "path": "/source/file/2", "action": "added" },
          { "path": "/source/file/3", "action": "deleted" },
          { "path": "/source/file/4", "action": "modified" },
          { "path": "/source/file/4", "action": "props_modified" }
        ],
        "event_time": "2012-10-02T17:15:32.320Z",
        "created_by": "username"
      }
    }
```

Ruby Submission Example

You will need to change the following example before you can run it. Set the Rabbit MQ hostname and the association key, and paste a JSON block following the commit format specified on this page.

```
require 'amqp'

      QUEUE = 'eventq.commits'
      COMMIT = '{
        "api_version": "1",
        "source_association_key" : "150e8951-19e1-4066-8b74-601026dd6cde",
        "commit_data" : {
          "repo_id": "390e8951-19e1-4066-8b74-601026dd6cde",
          "revision_id": "2315",
          "branch_name": "master",
          "type": "post_commit",
          "message": "here\'s my commit [orc-277]",
          "changes": [
            { "path": "/source/file/1", "action": "modified" },
            { "path": "/source/file/2", "action": "added" },
            { "path": "/source/file/3", "action": "deleted" },
            { "path": "/source/file/4", "action": "modified" },
            { "path": "/source/file/4", "action": "props_modified" }
          ],
          "event_time": "2012-10-02T17:15:32.320Z",
          "created_by": "username"
        }
      }'

      # Event loop
      EventMachine.run do
        connection = AMQP.connect('amqp://guest:guest@example-mq')
```

```
    # Set up our RabbitMQ information
    channel = AMQP::Channel.new(connection)
    queue = channel.queue(QUEUE, :auto_delete => false, durable: true)
    exchange = channel.direct('')

    # Publish the activity and exit the loop
    exchange.publish COMMIT, :routing_key => queue.name do
      connection.disconnect {EventMachine.stop}
    end
  end
```

Builds API

Submission Parameters

Submission Queue

- **queue_name**: eventq.builds
- **auto_delete**: false
- **durable**: true

Top Level

Parameter	Required/Optional	Description
api_version	Required	A string that matches the API version this document is for.
source_association_key	Required	A key generated by TeamForge EventQ that links incoming builds with the appropriate source server.
	Example: "source_association_key": "6t5qM5AuLLWLkmqFJeKp"	
build_data	Required	The raw build data, as provided by the build system.

Objects

build_data - The following fields should be in this build object.

Field	Required/Optional	Description
remote_id	Required	A string to uniquely identify the build. It will appear in the UI.
duration	Required	A string containing time (in seconds) it took for a build agent to build.
event_time	Required	A string containing a timestamp in UTC timezone and RFC 3339 format.
build_url	Required	A string with the URL to the build summary. Please note that the fully qualified URL with the URL scheme is required.
created_by	Optional	A string containing the username of the person who triggered the build.

status	Required	An object that declares the status of the build object.
	Accepted values • **type** (Required) - A string representing the type of the status. Please note that any type other than the listed below will be used as UNKNOWN inside TeamForge EventQ. ◦ SUCCESS ◦ FAILURE ◦ UNSTABLE ◦ ABORTED • **name** (Required) - A user-friendly display name for the status.	
test_results	Optional	An object containing the test results summary.
	Accepted values • **passed_count** - A string containing the number of passed tests. • **failed_count** - A string containing the number of failed tests. • **ignored_count** - A string containing the number of ignored tests. • **url** - A string containing the URL to the test results. Please note that the fully qualified URL with the URL scheme is required.	
revisions	Optional	An array of objects containing the SCM revisions related to this build.
	Accepted values • **revision** (Required) - A string containing the commit revision id. • **repository_url** (Required) - A string containing the repository url.	
event_time	Required	A string containing a timestamp in UTC timezone and RFC 3339 format.
created_by	Required	The user name of the person who took the action.

Examples

The following code examples are Copyright 2015 CollabNet, Inc., licensed under the Apache License, Version 2.0 (the “License”); you may not use this code except in compliance with the License. You may obtain a copy of the License at [http://www.apache.org/licenses/LICENSE-2.0]

Sample Submission JSON

```
{
    "api_version": "1",
    "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
    "build_data": {
    "remote_id": "700",
        "event_time": "2012-10-02T17:15:32.320Z",
        "duration": "200",
        "build_url": "http://example.com/builds/600",
        "created_by": "admin",
        "status": {
            "type": "SUCCESS",
            "name": "SUCCESSFUL"
        },
        "test_results": {
            "passed_count": "10",
            "failed_count": "0",
            "ignored_count": "0",
            "url": "http://example.com/builds/600/tests"
        },
        "revisions":
            [{ "revision": "fb849a2440dda438f4c6ab25f8c3266ed82d8797",
             "repository_url": "ssh://gitserver/project"
            }]
        }
  }
```

Ruby Submission Example

You will need to change the following example before you can run it. Set the Rabbit MQ hostname and the association key, and paste a JSON block following the build format specified on this page.

```
 require 'amqp'

      QUEUE = 'eventq.builds'
      BUILD = '{
        "api_version": "1",
        "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
        "build_data": {
            "remote_id": "700",
            "event_time": "2012-10-02T17:15:32.320Z",
            "duration": "200",
            "build_url": "http://example.com/builds/600",
            "created_by": "admin",
            "status": {
                "type": "SUCCESS",
                "name": "SUCCESSFUL"
            },
            "test_results": {
                "passed_count": "10",
                "failed_count": "0",
                "ignored_count": "0",
                "url": "http://example.com/builds/600/tests"
             },
            "revisions":
             [{ "revision": "fb849a2440dda438f4c6ab25f8c3266ed82d8797",
             "repository_url": "ssh://gitserver/project"
              }]
        }
      }'
      # Event loop
      EventMachine.run do
        connection = AMQP.connect('amqp://guest:guest@example-mq')

        # Set up our RabbitMQ information
        channel = AMQP::Channel.new(connection)
        queue = channel.queue(QUEUE, :auto_delete => false, durable: true)
        exchange = channel.direct('')

        # Publish the build and exit the loop
        exchange.publish BUILD, :routing_key => queue.name do
          connection.disconnect {EventMachine.stop}
        end
      end
```

Review Requests API

Submission Parameters

Submission Queue

- **queue_name**: eventq.reviews
- **auto_delete**: false
- **durable**: true

Top Level

Parameter	Required/Optional	Description
api_version	Required	A string that matches the API version this document is for.
source_association_key	Required	A key generated by TeamForge EventQ that links incoming review requests with the appropriate source server.
	Example: "source_association_key": "31e8dd90-164f-0130-c4d0-406c8f04c05d"	
request_data	Required	The raw review request data, as provided by the review server.

Objects

request_data - The following fields should be in this review request object.

Field	Required/Optional	Description
remote_id	Required	A string to uniquely identify the review request.
link	Optional	A URL that points back to this individual review.
summary	Optional	A string to summarize the review request.
description	Optional	A longer string that can be used to further describe the review request. It will appear in the UI.
created_by	Required	"A string containing the username of the person who submitted the request (e.g., initiated this request event, NOT necessarily the creator of the review; in some instances, this may consistently be an administrative user)."
status	Required	An object that declares the status of the review request.
	Accepted values • **type** (Required) - A string representing the type of the status. ◦ open - The review request is currently open for review. ◦ submitted - The review request changes have been submitted. ◦ rejected - The code in the request was rejected. ◦ discarded - The review request has been discarded. ◦ other - A status type that may not fit into the above list. • **name** (Required) - A user-friendly display name for the status.	
event_time	Required	A string containing a timestamp in UTC timezone and RFC 3339 format.
updated_by	Optional	A string containing the username of the person performing the update.

files	Optional	An array of objects with optional SCM revision information. If included, only the file name is required.
	Accepted values • **file** (Required) - A string with the full path to the file that was being reviewed. • **revision** (Optional) - A string representing the revision number of the commit that includes this file. • **repository_url** (Optional) - A string with the URL to the SCM server. For associations to commits, this must match one defined as a source in the same pipeline.	
reviewers	Optional	An array of user hashes. Optional reviewer information. If included, only the user name is required. For some systems (e.g., Crucible), reviewers are only reported when they have marked the review 'completed' or 'reviewed'; other systems (e.g., Review Board) report reviewers when they add comments or make other changes to a review.
	Accepted values • **username** (Required) - A string of the user's name. • **link** (Optional) - A string representing the URL to the users profile.	

Examples

Sample Submission JSON

```
{
    "api_version": "1",
    "source_association_key" : "31e8dd90-164f-0130-c4d0-406c8f04c05d",
    "request_data" : {
      "remote_id": "42",
      "link": "http://reviewserver/review/42/",
      "summary": "The summary of the request",
      "description": "An extended description of what the request covers",
      "status": {
        "type": "open",
        "name": "Review"
      },
      "event_time": "2012-10-02T17:15:32.320Z",
      "created_by": "username",
      "updated_by": "username",
      "files": [
        {
          "revision": "ef9f8fe661992f48c3539c05fc5c69c61918fa17",
          "file": "commit1_file1.txt",
          "repository_url": "ssh://gitserver/project"
        },
        {
          "revision": "ef9f8fe661992f48c3539c05fc5c69c61918fa17",
          "file": "commit1_file2.txt",
          "repository_url": "ssh://gitserver/project"
        },
        {
          "revision": "fb849a2440dda438f4c6ab25f8c3266ed82d8797",
          "file": "commit2_file1.txt",
          "repository_url": "ssh://othergitserver/project"
        }
      ],
      "reviewers": [
        { "username": "adent", "link": "http://reviewserver/user/adent" },
        { "username": "fprefect", "link": "http://reviewserver/user/fprefect"
}
      ]
    }
}
```

Ruby Submission Example

You will need to change the following example before you can run it. Set the Rabbit MQ hostname and the association key, and paste a JSON block following the review format specified on this page.

```
require 'amqp'

    QUEUE = 'eventq.reviews'
    REVIEW = '{
      "api_version": "1",
      "source_association_key" : "31e8dd90-164f-0130-c4d0-406c8f04c05d",
      "request_data" : {
        "remote_id": "42",
        "link": "http://reviewserver/review/42/",
        "summary": "The summary of the request",
        "description": "An extended description of what the request covers",
        "status": {
          "type": "open",
          "name": "Review"
        },
        "event_time": "2012-10-02T17:15:32.320Z",
        "created_by": "username",
        "files": [
          {
            "revision": "ef9f8fe661992f48c3539c05fc5c69c61918fa17",
            "file": "commit1_file1.txt",
            "repository_url": "ssh://gitserver/project"
          },
          {
            "revision": "ef9f8fe661992f48c3539c05fc5c69c61918fa17",
            "file": "commit1_file2.txt",
            "repository_url": "ssh://gitserver/project"
          },
          {
            "revision": "fb849a2440dda438f4c6ab25f8c3266ed82d8797",
            "file": "commit2_file1.txt",
            "repository_url": "ssh://othergitserver/project"
          }
        ],
        "reviewers": [
          { "username": "adent", "link": "http://reviewserver/user/adent" },
          { "username": "fprefect", "link": "http://reviewserver/user/fprefect
" }
        ]
      }
    }'

    # Event loop
    EventMachine.run do
      connection = AMQP.connect('amqp://guest:guest@example-mq')

      # Set up our RabbitMQ information
      channel = AMQP::Channel.new(connection)
      queue = channel.queue(QUEUE, :auto_delete => false, durable: true)
      exchange = channel.direct('')

      # Publish the activity and exit the loop
      exchange.publish REVIEW, :routing_key => queue.name do
        connection.disconnect {EventMachine.stop}
      end
    end
```

Work Item API

Integrating TeamForge EventQ with the Work Item systems requires for the adapters to be able to send the configuration details of the underlying systems (trackers configuration etc. (see “TODO put a link on configuring work item sources”)) as well as the actual data (work items) generated by these systems. These are two distinct types of messages that use different formats.

There are two types of Work Item APIs:

- Work Item Configuration API

- Work Item Message API

Work Item Configuration API

Submission Parameters

Submission Queue

- **queue_name**: eventq.work_items
- **auto_delete**: false
- **durable**: true

Top Level

Parameter	Required/Optional	Description
api_version	Required	A string that matches the API version this document is for.
source_association_key	Required	A key generated by TeamForge EventQ that links incoming work items with the appropriate source server.
	Example: "source_association_key": "6t5qM5AuLLWLkmqFJeKp"	
workitem_settings	Required	Tracker configuration details.

Objects

workitem_settings - The following fields should be in this workitem_settings object.

Field	Required/Optional	Description
event_time	Required	A string containing a timestamp in UTC timezone and RFC 3339 format.
url	Required	A string with the browse URL for the system. EventQ will append an individual work item's id to that url to link back to the work item.
tracker	Required	Tracker configuration information. A JSON object describing the structure of the tracker.
	Accepted values • **name** (Required) - A name for the tracker. • **id** (Required) - A string containing a unique (on the reporting system) identifier for the tracker. • **key** (Optional) - A string containing the tracker key (usually used to prefix the work-item ids). • **regex** (Required) - A string containing a regular expression that EventQ will use to associate work-items with commits using the commit message. • **icon** (Optional) - A string containing a URL for this tracker's icon.	

	• **statuses** (Required) - An array of status objects associated with the tracker. ◦ **id** (Required) - A string containing a unique(on the underlying system) identifier for the status. ◦ **name** (Required) - A string containing the name of the status.

Examples

The following code examples are Copyright 2015 CollabNet, Inc., licensed under the Apache License, Version 2.0 (the “License”); you may not use this code except in compliance with the License. You may obtain a copy of the License at [http://www.apache.org/licenses/LICENSE-2.0]

Sample Submission JSON

```
{
      "api_version": "1",
      "source_association_key" : "150e8951-19e1-4066-8b74-601026dd6cde",
      "workitem_settings" : {
      "url": "http://www.example.com/jira",
      "event_time": "2012-10-02T17:15:32.320Z",
      "tracker": {
                   "name": "EventQ/Stories",
                   "id": "01234567abcdef01234567",
                   "key":"ORC",
                   "regex": "ORC-\d+",
                   "icon": "http://www.example.com/jira/browse/bug.jpig",
                   "statuses": [
                       {"name": "Open", "id": "abcdef"},
                       {"name": "In progress", "id": "123abc"},
                       {"name": "Closed", "id": "456123"}
                   ]
                 }
      }
}
```

Ruby Submission Example

You will need to change the following example before you can run it. Set the Rabbit MQ hostname and the association key, and paste a JSON block following the build format specified on this page.

```
require 'amqp'

      QUEUE = 'eventq.work_items'
      WORK_ITEM_CONFIGURATION = '
      {
        "api_version": "1",
        "source_association_key" : "150e8951-19e1-4066-8b74-601026dd6cde",
        "workitem_settings" : {
            "url": "http://www.example.com/jira",
            "event_time": "2012-10-02T17:15:32.320Z",
            "tracker":  {
                            "name": "EventQ/Stories",
                            "id": "01234567abcdef01234567",
                            "key":"ORC",
                            "regex": "ORC-\d+",
                            "icon": "http://www.example.com/jira/browse/bug.jp
ig",
                            "statuses": [
                                {"name": "Open", "id": "abcdef"},
                                {"name": "In progress", "id": "123abc"},
                                {"name": "Closed", "id": "456123"}
                            ]
                         }
      }
    }'
      # Event loop
      EventMachine.run do
        connection = AMQP.connect('amqp://guest:guest@example-mq')

        # Set up our RabbitMQ information
        channel = AMQP::Channel.new(connection)
        queue = channel.queue(QUEUE, :auto_delete => false, durable: true)
        exchange = channel.direct('')

        # Publish the work item configuration and exit the loop
        exchange.publish WORK_ITEM_CONFIGURATION, :routing_key => queue.name d
o
          connection.disconnect {EventMachine.stop}
        end
      end
```

Work Item Message API

Submission Parameters

Submission Queue

- **queue_name**: eventq.work_items
- **auto_delete**: false
- **durable**: true

Top Level

Parameter	Required/Optional	Description
api_version	Required	A string that matches the API version this document is for.
source_association_key	Required	A key generated by TeamForge EventQ that links incoming work items with the appropriate source server.
	Example: "source_association_key": "6t5qM5AuLLWLkmqFJeKp"	
workitem_settings	Required	The work item data.

Objects

work_item - The following fields should be in this work item object.

Field	Required/ Optional	Description
created_by	Optional	A string containing the username of the person who submitted the work item.
updated_by	Optional	A string containing the username of the person performing the update.
id	Required	A string to uniquely identify the work item.
event_time	Required	A string containing a timestamp in UTC timezone and RFC 3339 format.
summary	Required	A string to summarize the work item.
description	Optional	A longer string that can be used to further describe the work item.
status_id	Required	A string to uniquely identify the status. This value must match the identifier within the work item configuration message.
tracker_id	Required	A string to uniquely identify the tracker. This value must match the identifier within the work item configuration message.
closed	Optional	A boolean for the open or closed state of the work item. True if the work item is closed.
deleted	Optional	A boolean for the deleted state of the work item. True if the work item has been deleted.
priority	Optional	A string containing the priority of the work item.
assigned_to	Optional	A string containing the username of the person who is assigned to the work item.
creation_time	Optional	A string containing a timestamp in UTC timezone and RFC 3339 format. The time the item was first created in the source system.
tags	Optional	An array of strings containing the tags, or labels, for this work item.

Examples

Sample Submission JSON

```
{
      "api_version": "1",
      "source_association_key" : "150e8951-19e1-4066-8b74-601026dd6cde",
      "work_item": {
          "created_by": "admin",
          "updated_by": "username",
          "summary": "title",
          "id": "KDA-59",
          "description": "description",
          "status_id": "10001",
          "event_time": "2014-03-12T20:47:41.295Z",
          "closed": false,
          "deleted": false,
          "tracker_id": "10200-2",
          "priority": "Major",
          "assigned_to": "admin",
          "creation_time": "2014-03-29T00:26:20.091Z",
          "tags": [
            "Tag1",
            "Tag2",
            "Tag3"
          ]
      }
}
```

Ruby Submission Example

You will need to change the following example before you can run it. Set the Rabbit MQ hostname and the association key, and paste a JSON block following the build format specified on this page.

```
require 'amqp'

      QUEUE = 'eventq.work_items'
      WORK_ITEM = '
      {
        "api_version": "1",
        "source_association_key" : "150e8951-19e1-4066-8b74-601026dd6cde",
        "work_item": {
            "created_by": "admin",
            "updated_by": "admin",
            "summary": "title",
            "id": "KDA-59",
            "description": "description",
            "status_id": "10001",
            "event_time": "2014-03-12T20:47:41.295Z",
            "closed": false,
            "deleted": false,
            "tracker_id": "10200-2",
            "priority": "Major",
            "assigned_to": "admin",
            "creation_time": "2014-03-29T00:26:20.091Z",
            "tags": [
                "Tag1",
                "Tag2",
                "Tag3"
            ]
        }
      }'
      # Event loop
      EventMachine.run do
        connection = AMQP.connect('amqp://guest:guest@example-mq')

        # Set up our RabbitMQ information
        channel = AMQP::Channel.new(connection)
        queue = channel.queue(QUEUE, :auto_delete => false, durable: true)
        exchange = channel.direct('')

        # Publish the work item and exit the loop
        exchange.publish WORK_ITEM, :routing_key => queue.name do
          connection.disconnect {EventMachine.stop}
        end
      end
```

Extensible Data Sources (XDS) Walk-through

This walk-through is intended to provide a detailed step-by-step implementation example of the Extensible Data Sources (or XDS) feature. Please read the parent overview section of this document first.

Setting the stage: EventQ does not ship with a stock activity API for deployment tools, so our fictional example lifecycle includes a process to deploy software into various environments. EventQ users would like to see these deployment activities bubble up to activity streams and any deployment associations in the context of other activities. So let’s define an XDS Schema modeling the “deployment” domain and then send activities to the newly registered XDS Activity API for deploy.

Define the Schema Attributes - The first step is craft an XDS Schema to adequately describe our deployment domain space and the events we expect to send to EventQ. The submission of the data type schema is itself governed by an API, which is documented at XDS Schema API.

1. API version The version number of the XDS Schema API (not your new schema’s version number).
2. Source association key This value is obtained by creating a new “XDS Source” in EventQ’s web client. Supplying a source association key is optional during schema registration. If provided, the source will be associated with the schema when the schema is received. Otherwise, the XDS source will be associated to a registered schema when the first activity message is received. There is no “preferred” option; this flexibility exists purely to suit various types of integration models.
3. Schema name Should be a human-friendly identifier of the data source.

4. Schema ID Takes a UUID as the value and the value should be unique per schema. If UUID generation is not possible, characters are limited to alphanumerics and the dash symbol.
5. Schema version Allows for the same schema to be updated without sending a separate schema ID. This is an integer field, best practice is to start at 1 and increment.

6. Event time The time at which your newly crafted schema is submitted to EventQ. Requires UTC RFC3339 format.

   The schema ID is user-defined; we suggest a long "UUID" style string. The schema name is also user-defined. In our deploy example, we can add a schema ID and name as follows:

```
// Example Deploy XDS Schema
{
  "api_version": "1",
  "source_association_key" : "8578c900-f8df-0131-84ff-3c07547a48b0",
  "custom_schema": {
  "name": "Faux Schema",
  "schema_id": "fc358dd0-11dc-0132-b9fe-3c07547a48b0",
  "schema_version": 2,
  "event_time": "2015-02-02T17:15:32.320Z",
  }
}
```

> **NOTE:** The source association key value was obtained from our instance of EventQ. It is an attribute of the parent XDS Collection with which you intend to register the new schema.

Defining the fields expected by your schema - Since you are defining an API at this stage, the next step is to define the list of fields expected by your API. Fields are the attributes you wish to supply with each event message sent by an adapter. We'll address optional vs. required fields in the next section, but for now we want a comprehensive list of all fields represented in your new activity message format. For instance, in our deploy example you may want to supply deploy environment, time, duration of deploy activities, etc.

The XDS Activity API categorizes fields into three classes: mandatory fields, stock fields, and custom fields. "Mandatory fields" are required in all registered schemas and thus must be present in all activity messages. "Stock fields" are a set of non-mandatory fields that have special relevance to the EventQ application, such as appearing in specific places in the user interface. "Custom fields" are truly arbitrary and may be used to capture desired data not encompassed by mandatory or stock fields.

1. **Mandatory fields** - all of these fields are required in every XDS schema.

   1. schema_id Activity messages will need to send this to identify themselves as governed by a specific schema.

    2. schema_version Should match the same schema_version as sent by the schema. This makes sure that the message is displayed with the correct version of the schema.
    3. event_time The UTC RFC3339 time of the event. 4.remote_id Represents the source tool’s identifier for the activity. For instance, if we were modeling Subversion commits, the remote_id would be the revision number. Messages with the same remote_id are treated as part of the same activity and are aggregated as “sub-events”.

2. **Stock fields** - Optional fields with special relevance in the EventQ application.

    1. associated_remote_id If your activity has an association this field is used to specify one or more IDs for purposes of association.
    2. created_by The user who instantiated the activity.
    3. summary The primary heading or title of the activity.
    4. description A detailed description of the activity.
    5. status_name The current status of the activity, corresponds to the “status_type” field.
    6. status_type The current status class, either “open”, “success”, or “fail” each of which maps to status icons displayed in the user interface.

    7. link_to Creates a link back to the activity on the source system. The link_to field can take an array with a link label along with the URL to customize the display text in the user interface. The default label is “See More”. Following our example, one could display “Deployment Details” in the user interface in place of the default “See More”.

        Let’s return to our deployment example and see how fields are defined in the Data Type Schema:

```
// Example Deploy XDS Schema
{
 "api_version": "1",
 "source_association_key" : "8578c900-f8df-0131-84ff-3c07547a48b0",
 "custom_schema": {
 "name": "Faux Schema",
 "schema_id": "fc358dd0-11dc-0132-b9fe-3c07547a48b0",
 "schema_version": 2,
 "event_time": "2015-02-02T17:15:32.320Z",
 "fields": ["schema_id", "schema_version", "event_time", "remote_id",
 "associated_remote_id", "summary", "status_name", "status_type",
 "link_to", "rule", "deploy_url", "deploy_message"],
  }
}
```

Note that we've included the four mandatory fields, a couple of "stock fields", and some custom fields modeling our "deploy" specific domain data.

Requiring specific fields - You may impose required fields. Remember that some fields are always required, so this section is not optional for that reason. But you can extend the set of required fields beyond those required by the API.

1. required Pass in an array of required fields. Any activity message without all of these fields will be discarded. Let's return to our deploy example and see how fields are defined in the Data Type Schema:

```
// Example Deploy XDS Schema
{
    "api_version": "1",
    "source_association_key" : "8578c900-f8df-0131-84ff-3c07547a48b0",
    "custom_schema": {
    "name": "Faux Schema",
    "schema_id": "fc358dd0-11dc-0132-b9fe-3c07547a48b0",
    "schema_version": 2,
    "event_time": "2015-02-02T17:15:32.320Z",
    "fields": ["schema_id", "schema_version", "event_time", "remote_id",
    "associated_remote_id", "summary", "status_name", "status_type",
    "link_to", "rule", "deploy_url", "deploy_message"],
   "required": ["schema_id", "schema_version", "event_time", "remote_id",
"rule"],
  }
}
```

The four fields required by the API are represented, along with one additional field we added above named "rule". Any activity message that fails to supply all five of these fields will be discarded.

Select "important" fields that result in activity stream posts - Not every event that triggers an activity message is necessarily worthy of a post in the activity stream. List only those fields that you deem worth of an activity stream update in the "important" field.

1. important Array of field names. Only changes to these fields will trigger a new entry in the activity stream. In this case, we only want changes to the "summary" or "status_name" field to trigger activity stream posts:

```
// Example Deploy XDS Schema
{
   "api_version": "1",
   "source_association_key" : "8578c900-f8df-0131-84ff-3c07547a48b0",
   "custom_schema": {
```

```
   "name": "Faux Schema",
   "schema_id": "fc358dd0-11dc-0132-b9fe-3c07547a48b0",
   "schema_version": 2,
   "event_time": "2015-02-02T17:15:32.320Z",
   "fields": ["schema_id", "schema_version", "event_time", "remote_id",
   "associated_remote_id", "summary", "status_name", "status_type",
   "link_to", "rule", "deploy_url", "deploy_message"],
   "required": ["schema_id", "schema_version", "event_time", "remote_id",
"rule"],
  "important": ["summary", "status_name"],
  }
}
```

Parse certain fields for TeamForge object references - Say you have a field that often contains a reference to a tracker artifact and you want EventQ to automatically convert those into links. An association will also be created if EventQ is aware of the referenced object.

In addition, any URL reference will be converted to a HTML link in the EventQ user interface. Say, www.example.com is referenced in a “parse” field, then it will be converted to a HTML link, e.g. www.example.com

1. parse Array of field names. Converts any TeamForge object references to links, converts URL references to links.

```
// Example Deploy XDS Schema
{
   "api_version": "1",
   "source_association_key" : "8578c900-f8df-0131-84ff-3c07547a48b0",
   "custom_schema": {
   "name": "Faux Schema",
   "schema_id": "fc358dd0-11dc-0132-b9fe-3c07547a48b0",
   "schema_version": 2,
   "event_time": "2015-02-02T17:15:32.320Z",
   "fields": ["schema_id", "schema_version", "event_time", "remote_id",
   "associated_remote_id", "summary", "status_name", "status_type",
   "link_to", "rule", "deploy_url", "deploy_message"],
   "required": ["schema_id", "schema_version", "event_time", "remote_id",
"rule"],
   "important": ["summary", "status_name"],
   "parse": ["deploy_message"],
    }
  }
```

By adding “deploy_mesage” to the “parsed” field, any CTF object will be linked and possibly associated. Any web addresses referenced will also be converted to a full HTML link.

Register your schema with EventQ - Now that your new schema is formulated, register your schema with your EventQ instance using either:

```
 * **eventq.custom message queue**, or
 * **XDS Schema HTTP API** (see `/api/1/docs/http_api_xds_schema`)
```

There is no preferred method for schema registration; pick the method that works best for your integration scenario. Registered schemas may be used site-wide. In our deploy example, perhaps we’ve written an extension to our deploy product and have registered its schema. The same schema may be reused by multiple sources (e.g., multiple instances of the deploy product) across the site.

Note also that successive submissions of the same schema ID and version will be ignored and it is a safe practice to send the same schema multiple times if it is programmatically convenient to do so.

To modify a schema, increment the schema_version value and re-register the schema with EventQ using the methods described above.

1. Success You can use the **XDS Schema HTTP API** (see `/api/1/docs/http_api_xds_schema`) to inspect registered schemas and their versions.

2. Troubleshooting Check the server log “custom_queue.log” to see if the schema was rejected and why it was rejected.

Send Activity messages - Now that you’ve defined an API for your tool/domain you can start supplying activity messages to EventQ that coincide with your new schema. For a technical reference on crafting activity messages, see the XDS Schema API documentation.

1. Submit XDS Activity messages to eventq.custom queue Note that the activity stream will only reflect new entries when an “important” field is modified or set in an activity message.

```
// Example Deploy XDS Activity Message
{
  "api_version": "1",
  "source_association_key" : "8578c900-f8df-0131-84ff-3c07547a48b0",
  "custom_data": {
 //required fields
  "name": "Faux Schema",
  "schema_id": "fc358dd0-11dc-0132-b9fe-3c07547a48b0",
  "schema_version": 2,
  "event_time": "2015-02-02T17:15:32.320Z",
  "remote_id": "deploy-1234",
  "rule": "1 comes before 2",
 // stock fields
  "summary": "Deploy at 4:52pm on a Friday",
  "deploy_message": "Deploying to fix [artf12345] and [artf12346] as critic
al bugs.",
  "associated_remote_id": "b4c8e50c908ade5c6f053e1ef6422f88",
  "link_to": ["Deploy Details", "http://www.example.com/deploy/deploy-1234
"],
  "status_type": "open",
 // other custom fields
  "environment": "Dev Stage 1"
  }
}
```

Optional Features - The XDS API offers some optional features to help pretty things up and make life a bit easier.

1. Interpolate TeamForge URLs

   It may be desirable to create URL references to TeamForge pages or functions (like SSO authentication). There are two interpolation functions in XDS:

   Reference TeamForge base URLs: %CTF_BASE_URL%

   Reference TeamForge “Go” URLs: %CTF_GO_URL%

   In our example, we may wish to reference a TeamForge Go URL like so:

   ```
   // Example Deploy XDS Activity Message
   ...
   "summary": "Deploy at 4:52pm on a Friday. See %CTF_GO_URL%/page1001",
   ```

2. Control field visibility Say you wanted to load specific data into EventQ’s database for reporting purposes, but you don’t want to expose all of those fields in the user interface. The solution is to add those fields to the hidden array in the XDS Schema message. The hidden array takes only custom fields (optional, non-stock). Custom fields are visible by default.

   Continuing our example, the newly introduced “environment” and “authorized_by” custom fields will be stored in the EventQ database, but not exposed in the user interface:

```
// Example Deploy XDS Schema
{
   "api_version": "1",
   "source_association_key" : "8578c900-f8df-0131-84ff-3c07547a48b0",
   "custom_schema": {
   "name": "Faux Schema",
   "schema_id": "fc358dd0-11dc-0132-b9fe-3c07547a48b0",
   "schema_version": 2,
   "event_time": "2015-02-02T17:15:32.320Z",
   "fields": ["schema_id", "schema_version", "event_time", "remote_id", "as
sociated_remote_id",
   "summary", "status_name", "status_type", "rule", "deploy_url", "deploy_
message",
   "environment", "authorized_by"],
   "required": ["schema_id", "schema_version", "event_time", "remote_id",
"rule"],
   "important": ["summary", "status_name"],
   "parse": ["deploy_message"],
   "link_to": ["deploy_url"],
   "hidden": ["environment","authorized_by"]
  }
}
```

XDS Schema API

Submission Parameters

Submission Queue

- **queue_name**: eventq.custom
- **auto_delete**: false
- **durable**: true

Top-Level

Parameter	Required/Optional	Description
api_version	Required	A string that matches the API version this document is for.
source_association_key	Optional	A key generated by TeamForge EventQ that links incoming work items with the appropriate source server. (Optional)
	Example: "source_association_key": "kLM56uWtKAqpeLLm5qJF"	

custom_schema	Required	a schema for the XDS events.

Objects

custom_schema - The fields specified below define the schema that activities will conform to.

Field	Required/Optional	Description
name	Required	A visible name that will allow pipeline-creators to recognize this schema when populating a pipeline.
schema_id	Required	A unique id defining this schema. Every activity sent by a single adapter should reference the same schema id. If this changes, the data source is considered a new data source, and data will not populate into pipelines that listened for the old one. If the schema changes, you can use the version to show that it's intended to be parsed differently, but continue to deliver messages to existing pipelines.
	Example: "schema_id": "fc358dd0-11dc-0132-b9fe-3c07547a48b0"	
schema_version	Required	A string to uniquely identify the work item.
	Example: "schema_version": 1	
event_time	Required	A string containing a timestamp in UTC timezone and RFC 3339 format.
	Example: "event_time": "2012-10-02T17:15:32.320Z"	
fields	Required	A string to summarize the work item.
	Example: "fields": ["schema_id", "schema_version", "event_time", "remote_id", "assigned_to", "hours_played", "severity", "canary"]	
required	Required	A longer string that can be used to further describe the work item.
	Example: "required": ["schema_id", "schema_version", "event_time", "remote_id"]	
important	Optional	A string to uniquely identify the status. This value must match the identifier within the work item configuration message.
	Example: "important": ["assigned_to", "severity"]	
parsed	Optional	A string to uniquely identify the tracker. This value must match the identifier within the work item configuration message.
	Example: "parsed": ["description"]	
hidden	Optional	A boolean for the open or closed state of the work item. True if the work item is closed.
	Example: "hidden": ["comment"]	

Examples

The following code examples are Copyright 2015 CollabNet, Inc., licensed under the Apache License, Version 2.0 (the “License”); you may not use this code except in compliance with the License. You may obtain a copy of the License at [http://www.apache.org/licenses/LICENSE-2.0]

Sample Submission JSON

```
{
    "api_version": "1",
    "source_association_key" : "8578c900-f8df-0131-84ff-3c07547a48b0",
    "custom_schema": {
      "name": "Faux Schema",
      "schema_id": "fc358dd0-11dc-0132-b9fe-3c07547a48b0",
      "schema_version": 2,
      "event_time": "2012-10-02T17:15:32.320Z",
      "fields": ["schema_id", "schema_version", "event_time", "remote_id", "ass
ociated_remote_id", "title", "rule", "deploy_url", "deploy_message", "comment"
],
      "required": ["schema_id", "schema_version", "event_time", "remote_id", "
rule"],
      "important": ["rule"],
      "parsed": ["deploy_message"],
      "hidden": ["comment"]
    }
}
```

XDS Activity API

Submission Parameters

Submission Queue

- **queue_name**: eventq.custom
- **auto_delete**: false
- **durable**: true

Top Level

Parameter	Required/Optional	Description
api_version	Required	A string that matches the API version this document is for.
source_association_key	Required	A key generated by TeamForge EventQ that links incoming work items with the appropriate source server.
	Example: "source_association_key": "kLM56uWtKAqpeLLm5qJF"	
custom_data	Required	The object containing custom activity data conforming to the referenced schema/version.

Objects

custom_data - The fields specified below conform to the schema referenced, and provide activity information to EventQ:

Field	Required/Optional	Description

schema_id	Required	The identifier of the XDS schema this message conforms to. The schema needs to be sent before an activity referencing that schema is sent.
schema_version	Required	The version of the referenced schema this message conforms to.
event_time	Required	The identifier of this activity; this is used to collapse related activities together.
remote_id	Required	A string to summarize the work item.
Stock Fields	Optional	
	The following field names are treated specially by EventQ, allowing a developer to show specific data in our UI where it can fit with our display patterns. All the fields are optional. • **associated_remote_id** - Specify the singular ID or array of IDs that your event should associate with, these IDs will be searched for in the source chosen in the pipeline custom source and shown in the Association section of the Details page. • **created_by** - The username or email address of the person who took the action. This is shown in the Details page. • **updated_by** - The username or email address of the person who took an action. This is shown in the Details page. • **summary** - The main title of the event. Shown in bold at the top of the Details page and also in the Activity Stream. • **description** - Detailed description of the event. Shown in full in the Details page, a truncated version is shown in the Activity Stream. • **status_name** - The name of the current status. This is a text field that is displayed in the Details page and Activity Stream. The value should be representational of the current state of the activity and match with the status_type. status_type - The type of status icon to show, containing values of "open", "success", "failed", each of which maps to a status icon displayed in the user interface. • **link_to** - The link back to the event on the source system. It take an array or single string. Array takes a URL along with the link label to customize the display text in the user interface. If only a URL string is passed without the label, the default label is "See More".	
Custom Fields	Optional	These custom fields can only contain String data and will be displayed in a key value list in the Details page.

Examples

The following code examples are Copyright 2015 CollabNet, Inc., licensed under the Apache License, Version 2.0 (the “License”); you may not use this code except in compliance with the License. You may obtain a copy of the License at [http://www.apache.org/licenses/LICENSE-2.0]

Sample Submission JSON 1: A deploy example with string value for link_to field

```
{
    "api_version": "1",
    "source_association_key" : "8578c900-f8df-0131-84ff-3c07547a48b0",
    "custom_data": {
        //required fields
        "schema_id": "fc358dd0-11dc-0132-b9fe-3c07547a48b0",
        "schema_version": 2,
        "event_time": "2012-10-02T17:15:32.320Z",
        "remote_id": "deploy-1234",
        // stock fields
        "created_by": "Build-bot",
        "updated_by": "Deploy-bot",
        "summary": "Deploy at 4:52pm on a Friday",
        "description": "Deploying to fix [FOO-245] and [BAR-900] as critical bu
gs.",
        "associated_remote_id": "b4c8e50c908ade5c6f053e1ef6422f88",
        "link_to": "http://www.example.com/deploy/b4c8e50c908ade5c6f053e1ef642
2f88",
        "status_name": "Deploying",
        "status_type": "open",
        // custom fields
        "environment": "production",
        "branch": "master"
    }
}
```

Sample Submission JSON 2: A build example with array value for the link_to field

```
{
    "api_version": "1",
    "source_association_key" : "1ebbbef0-2025-0132-8f68-3c15c2d51622",
    "custom_data": {
        //required fields
        "schema_id": "5193ca40-2025-0132-8f68-3c15c2d51622",
        "schema_version": 1,
        "event_time": "2012-10-02T17:15:32.320Z",
        "remote_id": "build-1234",
        // stock fields
        "created_by": "Build-bot",
        "updated_by": "Deploy-bot",
        "summary": "Build successful!",
        "description": "Build containing commits 109 through 112",
        "associated_remote_id": ["109","110","111","112"],
        "link_to": ["http://www.example.com/build/build-1234", "Build-1234"],
        "status_name": "Success",
        "status_type": "success",
        // custom fields
        "tests_passed": "1400",
        "tests_failed": "0",
        "tests_pending": "23"
    }
}
```

Reporting API

Overview

TeamForge EventQ Reporting API provides a means to query activity data in TeamForge EventQ. This API is for data retrieval only.

- **Submission URL**: `https://<eventq_hostname>/orc/api/<version>/reporting`.
- **Method**: POST (only).
- **Authentication**: Requires a valid TeamForge session id.
- **Authorization**: Requires the TeamForge EventQ “EventQ READ”, “REPORTING API” permission, or site admin permissions.

Convention

API requests are constructed from two building-blocks: commands with parameter(s) that can be chained one after another, followed by a single return type with its parameter(s). As such, API requests take the form:

```
command parameter(s)
command parameter(s)
command parameter(s) ...
return type parameter(s)
```

- Each chained command is treated as an “AND” query filter.
- Use separate queries to approximate “OR” requests.
- Requests are limited to a single return type which must be supplied after the last command.
- A query itself is a String constructed by chaining commands each separated from the next (or return) by a new line ‘\n’.
    - This ‘\n’ is very important since the parser looks for this when tokenizing the query.

Authentication and Authorization

In order to use the reporting api, your reporting client must have access to the endpoint of your TeamForge instance.

When sending query requests to the reporting api, a custom header named ‘X-EventQ-Session’ must be added to each http post request, the value of which is the ‘session id’ returned from a successful login call to the TeamForge endpoint. The TeamForge endpoint is found here: `http(s)://TeamForgeHost/ce-soap60/services/CollabNet`.

To gain access to the Reporting API the user must have one either of the below permissions set in TeamForge:

- TeamForge SITE ADMINISTRATOR
  - This will provided access to all active projects in EventQ.
- REPORTING API EventQ Permissions on at least one project
  - This limits the projects to query against to only those which the user has REPORTING API permissions.
  - If the user is both SITE ADMIN and REPORTING API, SITE ADMIN will be used. There is no way to limit a SITE ADMIN's access.

See the Examples below for a simple Java client.

Commands and Parameters

Command	Description
restrict_to (string)	Filter query results for certain classes of activity objects.
	Example: restrict_to reviews
	Accepted values: builds, commits, reviews, work_items, custom.
associated_to (array)	Takes one or more object IDs and returns a list of activity summaries with associations to the object IDs specified.
	Example: associated_to ["artf1002", "artf1002", "artf1003"]
associations (none)	Used in command chains to modify query results to return associations. No parameters.
	Example: associations
in_project (array)	Restrict query results to activities in a given project. Takes array of project IDs as a parameter.
	Example: in_project ["proj1123", "proj1134"]
sort_by (array)	Sort the result set by the fields of the objects which are supplied as a parameter, ascending or descending order.
	Example: sort_by ["status", "asc"]
	Accepted values: any object field; see MQ API docs for object descriptions or XDS schema for custom objects. For instance, builds may be sorted by remote_id, status, test_results, duration, created_by, or time_created.
	Accepted values: asc, desc
has_any (string)	Filter query results to those activities that have matching class associations.
	Example: has_any "builds"
	Accepted values: builds, commits, review, work_items, custom.

has_none (string)	Filter query results to those activities that do not have matching class associations.
	Example: has_none "reviews"
	Accepted values: builds, commits, review, work_items, custom.
limit (integer)	Limit query results to the specified integer for pagination of results and performance. May be used in conjunction with offset.
	Example: limit 100
	Accepted values: positive integers.
offset (integer)	Offset query results to specified integer for pagination of results and performance. May be used in conjunction with limit.
	Example: offset 10
	Accepted values: positive integers.
before (date/time)	Filter results set to those activity summaries with events that occurred prior to the supplied date and time. May be used in conjunction with after.
	• **Example:** before 2014-10-02T17:15:45.320Z • **Example:** before 2.weeks.ago
	Accepted values: RFC 3339 format date times in UTC and Ruby-style date references (minutes, hours, days, weeks, months, years).
after (date/time)	Filter results set to those activity summaries with events that occurred after the supplied date and time. May be used in conjunction with before.
	• **Example:** after 2014-11-02T12:45:45.320Z • **Example:** before 1.day.ago
	Accepted values: RFC 3339 format date times in UTC and Ruby-style date references (minutes, hours, days, weeks, months, years).
tagged_as (array)	Filter results set to those activities from sources labeled with the supplied tag. Takes multiple tags to create an OR condition. Two or more tagged_as commands may be chained which constructs an AND condition, returning only those activities with tag A and tag B.
	Example: tagged_as ["deploy","binary repository"] tagged_as ["production"]

Return Types

Conclude your query with a single “return” method to transform the data and return the results. All results are returned wrapped in a top-level json hash containing the key items, which contains the results of the query. The return may also contain the optional keys source_display_data, containing a hash of display metadata for the associated sources keyed by source id, and work_item_urls, containing urls to external work items keyed by id (if applicable).

Return Example

```
{
  "items": [
    {
      "_type": "WorkItemSummary",
      "assigned_to": "dev-bot",
      "associated_activity_summary_ids": [

      ],
      "associated_ctf_ids": [

      ],
      "closed": true,
      "comments": [

      ],
      "created_at": "2015-04-22T20:54:26.063Z",
      "created_by": "itembot!",
      "creation_time": "2014-03-29T00:26:20.091Z",
      "deleted": false,
      "description": "Autogenerated workitem description",
      "id": "55380a82109a01555d00000c",
      "opaque_id": "6kkd",
      "priority": "mecha-critical",
      "remote_id": "AUTO-6",
      "remote_status_id": "3",
      "settled": true,
      "source_id": "5537ed65109a012cec000004",
      "summary": "Release the integration",
      "tags": [
        "up",
        "strange"
      ],
      "time_created": "2015-04-22T20:54:26.000Z",
      "updated_at": "2015-04-23T16:49:19.657Z"
    }
  ],
  "source_display_data": {
    "5537ed65109a012cec000004": {
      "_type": "WorkItemSource",
      "_links": {
        "type_icon": {
          "href": "http://ctf.example.com/orc/assets/icons/work-item.png",
          "type": "image/png"
        },
        "source_icon": {
          "href": "http://external.example.com/images/icons/issuetypes/bug.png
",
          "type": "image/png"
        }
      },
      "name": "External Bug Tracker",
      "background_color": "#f2511b"
    }
  },
  "work_item_urls": {
    "55380a82109a01555d00000c": "http://tracker.example.com/a_tracker_source/A
UTO-6"
  }
}
```

Return Keys

Return Key	Description
items (array or hash)	Query-dependent results.
source_display_data (hash)	Display metadata for sources references in results.
work_item_urls (hash)	URLs for referenced work items. This key is present in the return structure only if 'items' contains WorkItemSummary objects.

Query Return Data Types

Data item	Description
fields (array or hash)	Returns json formatted results set limited to the fields supplied as parameters.
	Example: fields ["time_created", "remote_id"]
	Accepted values: any set of object fields; see MQ API docs for object descriptions or XDS schema for custom objects. For instance, builds may take: remote_id, status, test_results, duration, created_by, or time_created.
field_contains (nested array)	Returns json formatted results set limited to exact value supplied as parameters.
	Example: field_contains ["remote_id", ["artf1001", "art1002"]]
	Accepted values: any object field with matching value set; see MQ API docs for object descriptions or XDS schema for custom objects. For instance, builds may take: remote_id, status, test_results, duration, created_by, or time_created.

group_by_field(string)	Returns json formatted results set grouped by the field supplied as parameter.
	Example: group_by_field "status"
	Accepted values: created_at, remote_id, status
count (none)	Returns a json structure with a count of all matching activities.
	Example: count
timeseries (date/time)	Useful for charting applications, the output will be a hash, where the keys are the starting points of the time series, and the value is the array of objects that fall into that series. It takes a Ruby-style date reference parameter to specify the results granularity. Requires before and after in the same query.
	Example: timeseries 5.minutes
	Accepted values: Ruby-style date references (minutes, hours, days, weeks, months, years).

Examples

Use Cases and Query Examples

- Use case: “Find the number of all failed builds across all projects”

```
#Query:
    restrict_to builds
    field_contains ["status_type", ["FAILURE"] ]
    count
```

- Use case: “Given a set of artfs as inputs, give me all the associated activities”

```
#Query:
    associated_to: ["artf1234", "artf1235", "artf1236"]
```

- Use case: “Across two specific projects, give me the builds that pass vs fail”

```
#Query:
    in_project ["proj1001", "proj1002"]
    restrict_to builds
    group_by_field "status_type"
```

- Use case: “Given a set of artfs as inputs (say, in a planning folder) give me build stats (total #, pass, fail, etc)”

```
#Query:
    associated_to ["artf1001", "artf1002", "artf1003"]
    associations
    restrict_to builds
    group_by_field "status_type"
```

- Use case: “For a given user, give me a list of commits over X timeline, associated build stats”

```
#Query:
    restrict_to commits
    after 2.weeks.ago
    before 1.week.ago
    field_contains [ "created_by", ["jdoe"] ]
    associations
    restrict_to builds
    group_by_field "status_type"
```

- Use case: “Give me code review stats on a whole project (total #, total reviewed commits, total un-reviewed)”

```
#Query 1 to find un-reviewed commits
    in_project ["proj1706"]
    restrict_to commits
    has_none "reviews"
    count

    #Query 2 to find reviewed commits:
    in_project ["proj1706"]
    restrict_to commits
    has_any "reviews"
    count
```

- Use case: “Given a set of artfs, give me associated build results starting 1 week ago ending 5 minutes ago, grouped into 4 hour intervals.”

```
#Query:
    associated_to ["artf1001", "artf1002", "artf1003"]
    associations
    restrict_to builds
    after 1.week.ago
    before 5.minutes.ago
    timeseries 4.hours
```

- Use case: “Give me all binary artifact activities grouped by status”

```
#To query XDS sources like binary artifacts, restrict_to "custom" source type,
 then
    #filter to narrow the results using tagged_as
    #Query:
    restrict_to custom
    tagged_as ["binary artifact"]
    group_by_field "status_type"
```

Code samples

Create a Java client using Apache Axis & Apache HttpComponents

Get an X-EventQ-Session value from TeamForge.

```
/**
    * @param ctfEndpoint
    * @param ctfUserName
    * @param ctfUserPassword
    * @return sessionId (X-EventQ-Session)
    */
    private static String getSessionIdFromCTF(String ctfEndpoint, String ctfUs
erName, String ctfUserPassword) {
    try {
      Service service = new Service();
      Call call = (Call) service.createCall();
      call.setTargetEndpointAddress(new java.net.URL(ctfEndpoint));
      call.setOperationName(new QName("http://schema.open.collab.net/sfee50/so
ap60/service", "login"));
      String sessionId = (String) call.invoke(new Object[]{ctfUserName, ctfUse
rPassword});
      return sessionId;
    } catch (Exception e) {
      System.err.println(e.toString());
      return e.toString();
      }
    }
```

Post a request to the Reporting endpoint containing the custom header and a query.

```
/**
    * @param url
    * @param token
    * @param query
    * @return queryResults
    * @throws Exception
    */
    String sendPostHttpClient(String url, String token, String query) throws E
xception {
    String queryResults;
    CloseableHttpClient client = HttpClients.custom().build();

    HttpUriRequest request = RequestBuilder.post().setUri(url)
    .setHeader(HttpHeaders.CONTENT_TYPE, "application/x-www-form-urlencoded")
    .setHeader("X-EventQ-Session", token)
    .setEntity(new StringEntity(query))
    .build();

    CloseableHttpResponse response = client.execute(request);
    queryResults = EntityUtils.toString(response.getEntity(), "UTF-8");
    return queryResults;
    }
```

Build your client using the above methods.

```
public class ReportingClient {

    /* EventQ End Point represents the EventQ instance which you will be query
ing against*/
    static String eventqEndPoint = "http(s)://EventQHost/orc/api/1/reporting/"
;

    /* teamForgeEndPoint represents the soap endpoint for the TeamForge instan
ce which your EventQ
    * server is bound to*/
    static String teamForgeEndPoint = "http(s)://TeamforgeHost/ce-soap60/servi
ces/CollabNet";

    /* orcReportingQuery is your query in String format. Each LINE represents
a different command
    * so in this example every distinct command must be separated by a newline
 character '\n'*/
    static String orcReportingQuery = "restrict_to reviews\ngroup_by_field \"st
atus_type\"";

    /* TeamForge username and password for the account which the queries will
be run under. This account
    * must have REPORTING_API permissions on at least one project in EventQ or
 must be a SITE ADMIN
    * in Teamforge. If the user is NOT a TeamForge SITE ADMIN, only those proj
ects for which the user has
    * REPORTING_API permissions will be considered in the query.*/
    static String teamForgeUserName = "userName";
    static String teamForgeUserPassword = "userPassword";


    public static void main(String... args) {
      EventQReportingClient sampleClient = new EventQReportingClient();
      String xEventQSession = sampleClient.getSessionIdFromCTF(teamForgeEndPoi
nt,
      teamForgeUserName, teamForgeUserPassword);
      try {
        String serverResponse = sampleClient.sendPostHttpClient(eventqEndPoint
,
        xEventQSession, orcReportingQuery);
        System.out.println(serverResponse);
      } catch (Exception e) {
        System.out.println(e.toString());
      }

      }
    }
```

Efficient Querying with the EventQ Reporting API

Here's a few guidelines for building efficient EventQ Reporting API queries.

Cutting vs Building

Unlike a SQL query, which connects separate tables together to form a result, the EventQ Reporting API starts with the full set of all activities, and is cut down to the size of the result by successively applying query restrictions. This difference makes query optimization quite different between SQL and NoSQL.

Ordering Guidelines

Some query commands will result in larger reductions to the than others. This of course also depends on the range applied to the query. In example: Restricting to a week of activities using "after 1.week.ago" will make a very small set of activities to work on, compared to "after 1.year.ago" which will include a large number of activities. In general however, the commands here are listed by most efficient to least efficient in scoping the query result:

- **Good**:

```
:before ,  type :  :time
:after ,  type :  :time
:restrict_to ,  type :  :choice ,  choices :  %w(builds commits reviews work_i
tems custom)
:in_source ,  type :  :json
:in_repository ,  type :  :string
:in_project ,  type :  :json
```

- **Mid**:

```
:associated_to ,  type :  :json
:tagged_as ,  type :  :json
:offset ,  type :  :integer
```

- **Poor**:

```
:associations
:field_contains ,  type :  :json
```

```
:sort_by ,  type :  :json
:has_any ,  type :  :string
:has_none ,  type :  :string
```

- **Returning command (must be run last)**:

```
:limit ,  type :  :integer
:timeseries ,  type :  :duration
:group_by_field ,  type :  :string
:fields ,  type :  :json
:count
```

Use Cases and Examples

```
:in_project ["proj1211"]
after 14.days.ago
restrict_to commits
field_contains ["settled",[true]]
has_any "reviews"
```

Exceptions in ordering and other exceptional behaviours to note

- **Limit:** The limit command is best used at the end of the set. Using it before the end can result in more than the limit of items being returned at the end of the query.

- **Tagged_as:** Is specifically for custom activities.

HTTP APIs

HTTP API for XDS Schema Creation and Viewing

Overview

- TeamForge EventQ HTTP API for XDS Schema provides a means to create an XDS schema or view a schema and its versions. The API supports both data retrieval and submission.
- Authentication: requires a valid TeamForge session id, passed with the ‘X-EventQ-Session’ header field. One way to acquire this session id is to programmatically log into TeamForge using TeamForge SOAP API.
- Authorization: requires the TeamForge EventQ “EventQ CREATE” permission, or site admin permissions for creating schema. No permission required for viewing schema.

Convention

API requests for viewing and creating schemas are constructed as detailed below.

- List all schemas
  - **Request Type**: GET
  - **Request URL**: `https://<eventq_hostname>/api/1/schemas`
  - **Return Format**: Array of JSON objects. Each of the JSON object is a schema.
- List all schemas with a same schema_id
  - **Request Type**: GET
  - **Request URL**: `https://<eventq_hostname>/api/1/schemas/SCHEMA_ID`

    > **NOTE:** SCHEMA_ID needs to be encoded (escaped).

  - **Requires Parameter**: SCHEMA_ID
  - **Return Format**: Array of one or more JSON object(s). Each of the JSON object is a schema with the given schema_id and all the versions.
- Get one schema
  - **Request Type**: GET
  - **Request URL**: `https://<eventq_hostname>/api/1/schemas/SCHEMA_ID/VERSION`

    > **NOTE:** SCHEMA_ID needs to be encoded (escaped).

  - **Requires Parameter**: SCHEMA_ID, VERSION
  - **Return Format**: Array with one JSON object. The JSON object is a schema with the given schema_id and given version.
- Create a schema
  - **Request Type**: POST
  - **Request URL**: `https://<eventq_hostname>/api/1/schemas`
  - **Return Format**: An encoded (escaped) URL. This URL can be used to view the schema once it has been processed and saved.

TeamForge EventQ HTTP API for Tool Creation and Viewing

Overview

- TeamForge EventQ HTTP API for Tool provides a means to create an Tool or view a tool. The API supports both data retrieval and submission.
- Authentication: requires a valid TeamForge session id, passed with the 'X-EventQ-Session' header field. One way to acquire this session id is to programmatically log into TeamForge using TeamForge SOAP API.
- Authorization: requires the TeamForge EventQ "EventQ CREATE" permission, project admin, or site admin permissions.

> **NOTE:** PROJECT_URL_NAME is auto generated by TeamForge with project name initially. This should be found in last path of URL at project home page.

Convention

API requests for viewing and creating tools are constructed as detailed below.

- List all tools
  - **Request Type**:
  - **Request URL**: `https://<eventq_hostname>/api/stable/projects/PROJECT_URL_NAME/tools`
  - **Return Format**:

```
{
   "tools": [
   {
    "id": "unique_tool_id_1",
    "name": "Sample tool name One",
    "service_name": "Sample service name One",
    "project_id": "project id ",
    "active": true,
    "source_ids": [

      ]
   },
   {
    "id": "unique_tool_id_2",
    "name": "Sample tool name two",
    "service_name": "Sample service name two",
    "project_id": "project id ",
    "active": true,
    "source_ids": [
    "SourceID1",
    "SourceID2"
     ]
   }
  ]
}
```

- Return Codes
  - **Success**: 200
  - **Error**: 404, 503
- Get one tool
  - **Request Type**: GET
  - **Request URL**: `https://<eventq_hostname>/api/stable/projects/PROJECT_URL_NAME/tools/TOOL_ID`
  - **Return Format**:

```
{
  "tool": {
  "id": "TOOL_ID",
  "name": "Sample tool name two",
  "service_name": "Sample service name two",
  "project_id": "project id ",
  "active": true,
  "source_ids": [
    "SourceID1",
    "SourceID2"
   ]
 }
}
```

- Return Codes
    - **Success**: 200
    - **Error**: 404, 503
- Create a tool
    - **Request Type**: POST
    - **Request URL**: `https://<eventq_hostname>/api/stable/projects/PROJECT_URL_NAME/tools`
    - **Requires Parameter**: PROJECT_URL_NAME
    - **Request body**:

```
{
  "tool": {
  "name": "Sample tool name two",
  "source_ids": [
  "SourceID1",
  "SourceID2"
    ]
  }
}
```

- Return Format:

```
{
  "tool": {
    "id": "unique_tool_id_x",
    "name": "Sample tool name two",
    "project_id": "project id ",
    "active": true,
    "source_ids": [
      "SourceID1",
      "SourceID2"
    ]
  },
  "errors": {
    "base": [
      "error msg"
    ],
    "attribute_key": [
      "error msg"
    ]
  }
}
```

- Return Codes
    - **Success**: 200
    - **Error**: 404, 503, 422
- Update a tool
    - **Request Type**: PUT
    - **Request URL**: `https://<eventq_hostname>/api/stable/projects/PROJECT_URL_NAME/tools/TOOL_ID`
    - **Requires Parameter**: PROJECT_URL_NAME, TOOL_ID
    - **Request body**:

      ```
      {
        "name": "Tool New Name",
        "source_ids": [
        "sourceID3"
       ]
      }
      ```
- Return Format:

  ```
  {
  "errors": {
    "base": [
      "error msg"
    ],
    "attribute_key": [
      "error msg"
    ]
  }
  }
  ```
- Return Codes
    - **Success**: 200
    - **Error**: 404, 503, 422
- Activate a tool

  - **Request Type**: PUT
  - **Request URL**: `https://<eventq_hostname>/api/stable/projects/PROJECT_URL_NAME/tools/TOOL_ID/activate`
  - **Requires Parameter**: PROJECT_URL_NAME, TOOL_ID
  - **Return Format**:

    ```
    {
       "errors": {
       "base": [
          "error msg"
       ],
       "attribute_key": [
        "error msg"
      ]
     }
    }
    ```

- Return Codes
  - **Success**: 200
  - **Error**: 404, 503, 422
- Deactivate a tool
  - **Request Type**: PUT
  - **Request URL**: `https://<eventq_hostname>/api/stable/projects/PROJECT_URL_NAME/tools/TOOL_ID/deactivate`
  - **Requires Parameter**: PROJECT_URL_NAME, TOOL_ID
  - **Return Format**:

    ```
    {
      "errors": {
       "base": [
         "error msg"
       ],
      "attribute_key": [
         "error msg"
      ]
    ```

```
  }
}
```

- Return Codes
    - **Success**: 200
    - **Error**: 404, 503, 422

HTTP API for Source CRUD

Overview

- TeamForge EventQ HTTP API for Sources provides a means to create or view a source. The API supports both data retrieval and submission.
- Authentication: requires a valid TeamForge session id, passed with the ‘X-EventQ-Session’ header field. One way to acquire this session id is to programmatically log into TeamForge using TeamForge SOAP API.
- Authorization: requires the EventQ “EventQ CREATE” permission or TeamForge’s project-admin / site-admin permissions.

> **NOTE:** PROJECT_URL_NAME is auto generated by TeamForge with project name initially. This should be the last value in the URL at project home page.

Convention

API requests for viewing and creating sources are constructed as detailed below.

- List all sources by Project
    - **Request Type**: GET
    - **Request URL**: https:///api/stable/projects/PROJECT_URL_NAME/sources
    - **Return Format**:

```
{
   "sources": [
     {
       "id": "5578cd1c0fbb6ad771000001",
       "_type": "Build",
       "tool_id": "552c3736c69d3b3dfb000004",
       "active": true,
       "display_name": "Sample_Build_Source",
       "queue_username": "4d4eaf60-f1f9-0132-4687-185aeb8b25b8",
       "queue_password": "d70bc954b64e368f1f650bfeaba2e2ff6e5525774ae4
2f9a959f4ccf2314e9cb",
       "queue_server": "amqp://rabbitmqserver.example.com:5672",
       "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
       "associated_commit_server_id": "551d8560c69d3b474b000102"
     }
   ]
}
```

- **Return Codes**
    - **Success**: 200
    - **Error**: 404, 503

- List all sources by Tool
    - **Request Type**: GET
    - **Request URL**: `https://<ctf_hostname>/api/stable/projects/PROJECT_URL_NAME/tools/TOOL_ID/sources`
    - **Return Format**:

```
{
  "sources": [
   {
     "id": "5578cd1c0fbb6ad771000001",
     "_type": "Build",
     "tool_id": "552c3736c69d3b3dfb000004",
    "active": true,
    "display_name": "Sample_Build_Source",
    "queue_username": "4d4eaf60-f1f9-0132-4687-185aeb8b25b8",
    "queue_password": "d70bc954b64e368f1f650bfeaba2e2ff6e5525774ae42f9
a959f4ccf2314e9cb",
    "queue_server": "amqp://rabbitmqserver.example.com:5672",
    "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
    "associated_commit_server_id": "551d8560c69d3b474b000102"
   }
  ]
 }
```

- Create a source
  - **Request Type**: POST
  - **Request URL**: `https://<ctf_hostname>/api/stable/projects/PROJECT_URL_NAME/sources`
  - **Request body**:

    Required: _type, display_name

    ```
    {
       "source": {
       "_type": "Build",
       "tool_id": "552c3736c69d3b3dfb000004",
       "display_name": "Sample_Build_Source",
       "associated_commit_server_id": "551d8560c69d3b474b000102"
      }
    }
    ```

  - **Return Format**:

    ```
    {
      "source": {
      "id": "5578cd1c0fbb6ad771000001",
      "_type": "Build",
      "tool_id": "552c3736c69d3b3dfb000004",
      "active": true,
      "display_name": "Sample_Build_Source",
      "queue_username": "4d4eaf60-f1f9-0132-4687-185aeb8b25b8",
      "queue_password": "d70bc954b64e368f1f650bfeaba2e2ff6e5525774ae42f9a9
    59f4ccf2314e9cb",
      "queue_server": "amqp://rabbitmqserver.example.com:5672",
      "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
      "associated_commit_server_id": "551d8560c69d3b474b000102"
       },
       "errors": {
       "base": [
         "error msg"
    ```

```
   ],
   "attribute_key": [
   "error msg"
   ]
  }
}
```

- **Return Codes**
  - **Success**: 200
  - **Error**: 404, 503, 422

- Get one source
  - **Request Type**: GET
  - **Request URL**: `https://<ctf_hostname>/api/stable/projects/PROJECT_URL_NAME/sources/SOURCE_ID`
  - **Return Format**:

```
{
   "source": {
   "id": "5578cd1c0fbb6ad771000001",
   "_type": "Build",
   "tool_id": "552c3736c69d3b3dfb000004",
   "active": true,
   "display_name": "Sample_Build_Source",
   "queue_username": "4d4eaf60-f1f9-0132-4687-185aeb8b25b8",
   "queue_password": "d70bc954b64e368f1f650bfeaba2e2ff6e5525774ae42f9a
959f4ccf2314e9cb",
   "queue_server": "amqp://rabbitmqserver.example.com:5672",
   "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
  "associated_commit_server_id": "551d8560c69d3b474b000102"
  }
}
```

  - **Return Codes**
    - **Success**: 200
    - **Error** : 404, 503

- Update one source
    - **Request Type**: PUT
    - **Request URL**: `https://<ctf_hostname>/api/stable/projects/PROJECT_URL_NAME/sources/SOURCE_ID`
    - **Request body**:

      ```
      {
        "tool_id": "552c3736c69d3b3dfb000004",
        "display_name": "Sample_Build_Source",
        "associated_commit_server_id": "551d8560c69d3b474b000102"
      }
      ```

    - **Return Format**:

      ```
      {
        "errors": {
         base": [
          "error msg"
         ],
        "attribute_key": [
          "error msg"
         ]
        }
      }
      ```

    - **Return Codes**
        - **Success**: 200
        - **Error**: 404, 503, 422
- Activate one source
    - **Request Type**: PUT
    - **Request URL**: `https://<ctf_hostname>/api/stable/projects/PROJECT_URL_NAME/sources/SOURCE_ID/activate`
    - **Return Format**:

      ```
      {
        "errors": {
      ```

```
    "base": [
      "error msg"
    ],
    "attribute_key": [
    "error msg"
    ]
   }
 }
```

  - **Return Codes**

    - **Success**: 200
    - **Error**: 404, 503, 422

- Deactivate one source

  - **Request Type**: PUT
  - **Request URL**: `https://<ctf_hostname>/api/stable/projects/PROJECT_URL_NAME/sources/SOURCE_ID/deactivate`

  - **Return Format**:

```
{
   "errors": {
   "base": [
     "error msg"
   ],
   "attribute_key": [
     "error msg"
   ]
  }
}
```

  - **Return Codes**

    - **Success**: 200
    - **Error**: 404, 503, 422

Return Types

The source objects have a set of common fields, while some of the fields depend on the _type of the source. The examples below show the type of source and the expected structure of the returned objects.

- **Source Type:** Build

```
{
        "source": {
          "id": "5578cd1c0fbb6ad771000001",
          "_type": "Build",
          "tool_id": "552c3736c69d3b3dfb000004",
          "active": true,
          "display_name": "Sample_Build_Source",
          "queue_username": "4d4eaf60-f1f9-0132-4687-185aeb8b25b8",
          "queue_password": "d70bc954b64e368f1f650bfeaba2e2ff6e5525774ae42f9a9
59f4ccf2314e9cb",
          "queue_server": "amqp://rabbitmqserver.example.com:5672",
          "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
          "associated_commit_server_id": "551d8560c69d3b474b000102"
         }
      }
```

- **Source Type:** Commit

```
{
        "source": {
          "id": "5578cd1c0fbb6ad771000001",
          "_type": "Commit",
          "tool_id": "552c3736c69d3b3dfb000004",
          "active": true,
          "display_name": "Sample_ExternalRepository",
          "queue_username": "4d4eaf60-f1f9-0132-4687-185aeb8b25b8",
          "queue_password": "d70bc954b64e368f1f650bfeaba2e2ff6e5525774ae42f9a9
59f4ccf2314e9cb",
          "queue_server": "amqp://rabbitmqserver.example.com:5672",
          "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
          "repository_uri": "ssh://sample_gitserver/sample_project"
         }
      }
```

- **Source Type:** CommitCTF

```
{
        "source": {
          "id": "5578cd1c0fbb6ad771000001",
          "_type": "CommitCTF",
          "tool_id": "552c3736c69d3b3dfb000004",
          "active": true,
          "ctf_id": "cmmt1234",
          "display_name": "Sample_ExternalRepository",
          "queue_username": "4d4eaf60-f1f9-0132-4687-185aeb8b25b8",
          "queue_password": "d70bc954b64e368f1f650bfeaba2e2ff6e5525774ae42f9a9
59f4ccf2314e9cb",
          "queue_server": "amqp://rabbitmqserver.example.com:5672",
          "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
          "repository_uri": "ssh://sample_gitserver/sample_project"
         }
      }
```

- **Source Type:** XDS

```
{
        "source": {
          "id": "5578cd1c0fbb6ad771000001",
          "_type": "XDS",
          "tool_id": "552c3736c69d3b3dfb000004",
          "active": true,
          "display_name": "Sample Custom Activity Name One",
          "queue_username": "4d4eaf60-f1f9-0132-4687-185aeb8b25b8",
          "queue_password": "d70bc954b64e368f1f650bfeaba2e2ff6e5525774ae42f9a9
59f4ccf2314e9cb",
          "queue_server": "amqp://rabbitmqserver.example.com:5672",
          "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
          "associated_activity_source_id": "55429d045667ba223e000010",
          "activity_source_tag_names": ["Exclusive", "Limited", "Sample"],
          "background_color_value": "#61c0e5",
          "icon_uri": "/sample_icons/custom_source/Sample_Icon_White_12.png"
          "custom_schema_store_id": object.custom_schema_store_id
        }
      }
```

- **Source Type:** Review

```
{
        "source": {
          "id": "5578cd1c0fbb6ad771000001",
          "_type": "Review",
          "tool_id": "552c3736c69d3b3dfb000004",
          "active": true,
          "display_name": "Sample Review server Name One",
          "queue_username": "4d4eaf60-f1f9-0132-4687-185aeb8b25b8",
          "queue_password": "d70bc954b64e368f1f650bfeaba2e2ff6e5525774ae42f9a9
59f4ccf2314e9cb",
          "queue_server": "amqp://rabbitmqserver.example.com:5672",
          "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
          "commit_association_prefix": "Review"
          }
        }
```

- **Source Type**: WorkItemSource

```
{
        "source": {
          "id": "5578cd1c0fbb6ad771000001",
          "_type": "Review",
          "tool_id": "552c3736c69d3b3dfb000004",
          "active": true,
          "display_name": "Sample Work Item Name One",
          "queue_username": "4d4eaf60-f1f9-0132-4687-185aeb8b25b8",
          "queue_password": "d70bc954b64e368f1f650bfeaba2e2ff6e5525774ae42f9a9
59f4ccf2314e9cb",
          "queue_server": "amqp://rabbitmqserver.example.com:5672",
          "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
          "associated_commit_server_id": "551d8560c69d3b474b000102"
         }
      }
```

Validations

The following constitute the validations that are performed for the API request-response parameters.

- **_type**
    - type: string
    - present for all sources
    - must be one of: Commit, CommitCTF, WorkItem, CustomActivity, Review, Build, XDS
- **active**
    - type: boolean
    - present in all sources
    - must be true or false
    - default value: true
- **associated_activity_source_id**
    - type: string
    - attribute in XDS, need not be present
    - references an existing activity source if present
- **commit_association_prefix**

  - type: string
  - attribute in Review, must be present
  - must be unique
  - key length must be less than 101

- **activity_source_tag_names**

  - type: array with string elements
  - attribute in XDS, need not be present

- **background_color_value**

  - type: string
  - attribute of XDS, must be present
  - must be a valid RGB value or a HTML5 color name

- **associated_commit_server_id**

  - type: string
  - attribute in Build and WorkItem, need not be present
  - must be associated with a valid commit server if present

- **custom_schema_source_id**

  - type: string
  - attribute in XDS, need not be present
  - must be associated to an existing custom_schema_source if present

- **display_type**

  - type: string
  - must be present in all sources
  - maximum length less than 101

- **icon_uri**

  - type: string
  - attribute of XDS, need not be present
  - has the default value: “/assets/icons/custom_source/PNG_White/CN_Icon_White_12.png”
  - must be a valid uri if present

- **repository_uri**

    - type: string
    - must be present in Commit and CommitCTF
    - must be a valid uri if present
    - must be a valid protocol (http, https, ssh, git) if present

- **ctf_id**

    - type: string
    - must be present in CommitCTF
    - must be a valid ctf id if present

- **source_association_key**

    - type: string in all sources
    - must be present

- **tool_id**

    - type: string
    - need not be present or present in any source
    - must be associated with a valid tool if present

Change log of site-options.conf tokens.

TeamForge 18.1

New Tokens

- GERRIT_USER_EMAIL
- BROWSER_NO_CACHE

Unsupported *_JAVA_OPTS Token Options

TeamForge 18.1 (and later) supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens before upgrading to TeamForge 18.1.

- JBOSS_JAVA_OPTS
- PHOENIX_JAVA_OPTS
- INTEGRATION_JAVA_OPTS
- ETL_JAVA_OPTS

TeamForge 17.11

Obsolete Tokens

- ENABLE_CACHING_WITH_MEMCACHED
- MEMCACHED_SERVER_HOST
- MEMCACHED_SERVER_PORT
- MEMCACHED_SERVER_TTL
- SSL_CA_CERT

subversion-caching service

A new TeamForge service, `subversion-caching`, has been added in TeamForge 17.11. Add this service to the `SERVICES` token of the TeamForge `site-options.conf` file to have Memcached installed. For more information, see Install Memcached. With this change, the following tokens are no longer supported:

- ENABLE_CACHING_WITH_MEMCACHED
- MEMCACHED_SERVER_HOST

- MEMCACHED_SERVER_PORT
- MEMCACHED_SERVER_TTL

Separate Ports for Database and Datamart on the Same Server

The ability to run separate PostgreSQL instances for TeamForge database and datamart on the same server has been deprecated.

- During TeamForge installation, the `REPORTS_DATABASE_PORT` token should no longer be used to assign a separate port for datamart on the server that also runs the TeamForge database. The following warning shows up if you use the `REPORTS_DATABASE_PORT` token with a custom port number (other than the default value, which is 5432).

  ```
  Using two separate Postgres clusters for database and datamart on the same machine is deprecated. Consider deploying the two clusters on two machines or using a single cluster for both databases.
  ```

- If you have the TeamForge database and datamart running on separate PostgreSQL instances on the same server:
  - **New Hardware Upgrade**: If you are upgrading on a new hardware, it is highly recommended to create a dump of both the database and datamart and load them into the same PostgreSQL instance. For more information, see Create a single cluster for both Database and Datamart.
  - **Same Hardware Upgrade**: If you are upgrading on the same hardware, you may still choose to use the `REPORTS_DATABASE_PORT` and have the database and datamart running on two separate PostgreSQL instances. However, support for `REPORTS_DATABASE_PORT` token may end in one of the future TeamForge releases, when you may have to dump and load both the database and datamart on the same PostgreSQL instance anyway.

TeamForge 17.8

Obsolete tokens

- JAMES_ACCEPTED_RELAYS
- REPORTS_LIFECYCLE_METRICS

TeamForge 17.4

Obsolete Tokens

- GERRIT_FORCE_HISTORY_PROTECTION
- ORC_HOSTNAME[1]
- ORC_SSL_CA_CERT_FILE
- ORC_PORT
- ORC_PROXIED_PATH
- ORC_PROTOCOL
- ORCHESTRATE_ENABLED
- POSTGRES_INTERFACE_IP
- POSTGRES_INTERFACE

New Tokens

- LISTEN_IP

Default Values for site-options.conf Tokens

Default values have been assigned to the following `site-options.conf` tokens and are therefore removed from the default TeamForge 17.4 `site-otpions-default.conf` file.

> **WARNING:** If you are upgrading from TeamForge 17.1 (or earlier) to TeamForge 17.4 (or later) and if you have been using your own values for the following tokens, you must make sure you use the same values in your site-options.conf file post upgrade to TeamForge 17.4.

```
# CTF Core
SCM_ADMIN_PASSWORD=$auto$
IAF_DBPASS=$auto$
SCM_DEFAULT_SHARED_SECRET=$auto$
SOAP_ANONYMOUS_SHARED_SECRET=$auto$

# Database
DATABASE_USERNAME=teamforge
DATABASE_NAME=teamforge
DATABASE_PASSWORD=$auto$
DATABASE_READ_ONLY_USER=teamforge_reader
DATABASE_READ_ONLY_PASSWORD=$auto$
```

```
# Datamart
REPORTS_DATABASE_USERNAME=teamforge_datamart
REPORTS_DATABASE_NAME=teamforge_datamart
REPORTS_DATABASE_PASSWORD=$auto$
REPORTS_DATABASE_READ_ONLY_USER=teamforge_datamart_reader
REPORTS_DATABASE_READ_ONLY_PASSWORD=$auto$

# ETL
ETL_SOAP_SHARED_SECRET=$auto$

# Gerrit
GERRIT_DATABASE_PASSWORD=$auto$

# RabbitMQ
RABBITMQ_APP_ADMIN_USER=guest
RABBITMQ_APP_ADMIN_PASSWORD=$auto$
RABBITMQ_APP_CTF_USER=ctf
RABBITMQ_APP_CTF_PASSWORD=$auto$
RABBITMQ_APP_SERVICES_USER=eventq
RABBITMQ_APP_SERVICES_PASSWORD=$auto$
RABBITMQ_PERMISSION_PUBLISHER=permission_publisher
RABBITMQ_PERMISSION_PUBLISHER_PASSWORD=$auto$

# Mongo
MONGODB_APP_DATABASE_NAME=eventq
MONGODB_ADMIN_DATABASE_NAME=admin
MONGODB_APP_ADMIN_USER=admin
MONGODB_APP_ADMIN_PASSWORD=$auto$
MONGODB_APP_BACKUP_USER=backup
MONGODB_APP_BACKUP_PASSWORD=$auto$
MONGODB_APP_SERVICES_USER=eventq
MONGODB_APP_SERVICES_PASSWORD=$auto$

# HAProxy
HAPROXY_STATS_PASSWORD=$auto$

# Binary
IAF_DBNAME=iafdb
IAF_DBUSER=iafdbusr

# James Admin (for management interface)
JAMES_ADMIN_USER=admin
JAMES_ADMIN_PASSWORD=$auto$
```

TeamForge 17.1

Obsolete Tokens

- BDCS_ADMIN_PASSWORD
- BDCS_ADMIN_USERNAME
- BDCS_HOST
- BDCS_SSL
- BDCS_TOMCAT_PORT
- BDCS_SDK_SEARCH_LIMIT_MAX
- BDCS_SSL_CERT_FILE
- BDCS_SSL_KEY_FILE
- BDCS_SSL_CA_CERT_FILE
- BDCS_SSL_CHAIN_FILE
- BDCS_SCAN_SOURCE_DIR_ROOT
- BDCS_INSTALL_PATH
- BDCS_PGSQL_HOME_DIR_ROOT
- BDCS_PGSQL_PORT
- BDCS_TOMCAT_MX_IN_MB
- BDCS_TOMCAT_SHUTDOWN_PORT

New Tokens

- ELASTICSEARCH_JAVA_OPTS

TeamForge 16.10

Obsolete Tokens

- SSH_TUNNEL_ENABLED
- SELINUX_SETUP
- SELINUX_ENABLED

New Tokens

- DISABLE_REMOTE_PUBLISHING

TeamForge 16.7

Obsolete Tokens

- TeamForge 16.7 installer automatically sets `JAVA_HOME` during installation or upgrade. Therefore, the `JAVA_HOME` site options token, if added to your `site-options.conf` file, must be removed while upgrading to TeamForge 16.7 and later.
- The `DEDICATED_INSTALL` token is no longer supported. CollabNet recommends removing this token from the `site-options.conf` file. However, this token is ignored (has no effect whatsoever) if you continue to have it in your `site-options.conf` file post upgrade to TeamForge 16.7 and later.
- Debug settings can be done via the `JAVA_OPTS` tokens (such as JBOSS_JAVA_OPTS and ETL_JAVA_OPTS). Hence, the following tokens are no longer supported:
  - JBOSS_DEBUG
  - PHOENIX_DEBUG
  - ETL_DEBUG
  - INTEGRATION_DEBUG
  - ETL_DEBUG_PORT
  - JBOSS_DEBUG_PORT
  - INTEGRATION_DEBUG_PORT
  - PHOENIX_DEBUG_PORT
- The SITE_DIR and DATA_DIR tokens are no longer supported. Starting from TeamForge 16.7 (and later) SITE_DIR and DATA_DIR are unconditionally set to `/opt/collabnet/teamforge` and `/opt/collabnet/teamforge/var` respectively during runtime creation.
- The MODPAGESPEED_ENABLED token is no longer supported.
- SSL certificates are validated by default. Hence, VALIDATE_SSL_CERTS token is no longer supported. If in use, remove this token from the `site-options.conf` file post upgrade to TeamForge 16.7 and later.
- While the SSL_CA_CERT_FILE token is still supported, it has been removed from the `site-options-default.conf` file as it is only needed for add-ons. References to this token have been removed from TeamForge documentation as well.

New Tokens

- ALLOW_CASE_INSENSITIVE_LOGIN
- HTTP_MAX_PARAMETERS
- ENABLE_SITE_NEWS
- LOGROTATE_ARCHIVE_COUNT

- MAX_PASSWORD_LENGTH

TeamForge 8.2

Obsolete Tokens

- PERFORCE_CLIENT_DIR
- PERFORCE_GROUP
- PERFORCE_LICENSE_FILE
- PERFORCE_LOG_DIR
- PERFORCE_PORT
- PERFORCE_REPOSITORY_BASE
- PERFORCE_SERVICE_CMD
- PERFORCE_USER
- HELP_AVAILABILITY
- REMOTE_HELP_URL
- EXTERNAL_TOMCAT_INSTALL_DIR
- INTEGRATION_BUILTIN_TOMCAT
- ETL_BUILTIN_TOMCAT
- CEE_COMPATIBLE

New Tokens

- ENABLE_CACHING_WITH_MEMCACHED
- MEMCACHED_SERVER_HOST
- MEMCACHED_SERVER_PORT
- MEMCACHED_SERVER_TTL

TeamForge 8.0

New Tokens

- NOTIFY_SITE_ADMINS_FOR_SITE_ACTIVITIES
- BINARY_SETUP_TYPE

1. EventQ installation is being taken care of by the TeamForge installer. Remove the ORC_* tokens from the `site-options.conf` file while upgrading to TeamForge 17.4 or later. ↩

Here's a list of TeamForge `site-options.conf` tokens and configuration information.

host:SERVICES

The `host:SERVICES` token is used to define the TeamForge services running on a host.

The syntax for defining the services running on a TeamForge host is:

```
<hostname>:SERVICES = list of services separated by space
```

Where `<hostname>` can be `localhost` or the server name as returned by the `hostname` command on the console. The latter is recommended as this allows reuse of the same `site-options.conf` file across all servers in a distributed setup. Here's a few examples.

Example 1: Default Single-server Setup

```
localhost:SERVICES=ctfcore ctfcore-database ctfcore-datamart etl mail search c
odesearch subversion eventq redis mongodb rabbitmq
```

Example 2: Single-server Setup with Git Integration

```
localhost:SERVICES=ctfcore ctfcore-database ctfcore-datamart etl mail search c
odesearch subversion eventq redis mongodb rabbitmq gerrit gerrit-database
```

Example 3: Single-server Setup with Review Board Integration

```
localhost:SERVICES=ctfcore ctfcore-database ctfcore-datamart etl mail search c
odesearch subversion eventq redis mongodb rabbitmq reviewboard reviewboard-dat
abase
```

Example 4: Three-server Setup with Git and Binary Integration

```
server01:SERVICES=ctfcore etl mail search codesearch eventq redis mongodb rabb
itmq binary binary-database
server02:SERVICES=subversion gerrit
server03:SERVICES=ctfcore-database ctfcore-datamart gerrit-database
```

Example 5: Distributed Setup with Multiple Git Integration Servers

```
server01:SERVICES=ctfcore ctfcore-database ctfcore-datamart etl mail search co
desearch eventq redis mongodb rabbitmq binary binary-database
server02:SERVICES=subversion
server-03:SERVICES=gerrit gerrit-database
server-04:SERVICES=gerrit gerrit-database
```

host:PUBLIC_FQDN

The `host:PUBLIC_FQDN` token is used to define the domain name of your TeamForge site. Assign a public FQDN (optional, but strongly recommended). Make sure there is a DNS A or CNAME record for this FQDN. Here's a few examples.

```
server01:PUBLIC_FQDN = teamforge.example.com
server02:PUBLIC_FQDN = scm.example.com
```

> **NOTE:** You cannot have a separate PUBLIC_FQDN for EventQ.

Service-specific FQDNs

Installing TeamForge with service-specific FQDNs (instead of machine-specific host/domain names) is highly recommended so that you will be able to change the system landscape at a later point in time without having any impact on the URLs (in other words, end users do not have to notice or change anything). For example, you can create FQDNs specifically for services such as Subversion, Git, mail, Codesearch and so on.

- All such service-specific FQDNs must belong to a single sub domain and it is recommended to create a new sub domain for TeamForge.
- A wildcard SSL cert is required if you are using service-specific FQDNs. SNI SSL cert cannot be used.
- When SSL is enabled and no custom SSL-certificates are provided, a self-signed wildcard cert is generated for the sub domain.
- When SSL is enabled and a custom SSL-certificate is provided, the CN of the certificate is verified to be a wildcard CN.

The following configuration shows the list of services and FQDNs for specific services such as ctfcore, subversion, gerrit and mail in a single server TeamForge installation:

```
localhost:SERVICES = ctfcore ctfcore-database ctfcore-datamart etl mail search
 codesearch subversion
localhost:PUBLIC_FQDN = app.forge.collab.net
localhost:ctfcore:PUBLIC_FQDN = ctf.forge.collab.net
```

```
localhost:subversion:PUBLIC_FQDN = svn.forge.collab.net
localhost:gerrit:PUBLIC_FQDN = git.forge.collab.net
localhost:mail:PUBLIC_FQDN = mail.forge.collab.net
```

In a single server setup, all these domain names point to a single server. However, when services are later distributed across multiple servers, all it takes to avoid an end user impact is to adjust these domain names to point to different servers.

TeamForge 18.1 has no support for having service-specific FQDN for Review Board.

ACTIVITY_LINKS_CUSTOMIZATION

When this token is set to `true`, the **TeamForge Activity Chart** and the **Most Active Projects List** do not appear on the main page before the user logs in.

Values: true or false

Default: false

ADMIN_EMAIL

The `ADMIN_EMAIL` token specifies a valid email address for the site administrator. The mail account specified must be hosted on a separate server outside of the TeamForge Application Server. The `SYSTEM_EMAIL`, `ADMIN_EMAIL`, and `JAMES_POSTMASTER_EMAIL` tokens can specify the same address.

> **IMPORTANT:** In TeamForge 6.x (and later), the sender name and address for system-generated emails is taken from the value of the `SYSTEM_EMAIL` token. Therefore, changing the admin user’s full name or email address does not affect the sender details of system-generated emails. This is different from TeamForge 5.x, in which the sender name and address for system-generated emails is derived from the admin user’s full name and email address.

Values: Email address specification

Default: `root@{\__APPLICATION_HOST\__}`

ALLOW_CASE_INSENSITIVE_LOGIN

In general, TeamForge usernames are validated case-sensitively. Set this token to true so that username validations are done case-insensitively.

Values: true or false

Default: false

ALLOW_NO_PASSWORD_ON_USER_CREATION

The `ALLOW_NO_PASSWORD_ON_USER_CREATION`token, when set to true, allows the admin to create users with null password through SOAP when external authentication is enabled.

Values: true or false

Default: false

ALLOW_USERNAME_IN_PASSWORD

The `ALLOW_USERNAME_IN_PASSWORD` token, when set to true, allows users to set a password that includes the string that they use for their user name on the site.

Values: true or false

Default: true

ALLOW_PASSWORD_DICTIONARY_WORD

You must set the `REQUIRE_PASSWORD_SECURITY` token to `true` in the `site-options.conf` file, for ALLOW_PASSWORD_DICTIONARY_WORD security setting to take effect.

APPLICATION_LOG_DIR

The `APPLICATION_LOG_DIR` token specifies the directory to which the application writes its log files.

Values: Path specification

Default: `/opt/collabnet/teamforge/log/apps`

APPROVE_NEW_USER_ACCOUNTS

The `APPROVE_NEW_USER_ACCOUNTS` token specifies whether a site administrator must approve the requests to join the site.

Values: true or false

Default: true

ARTIFACT_DESC_EDITOR

The `ARTIFACT_DESC_EDITOR` token allows you to choose the type of text that can be used for artifact description using the editor tool.

Values: Plain Text

Default: Plain Text

ARTIFACT_LIST_LIMIT

The `ARTIFACT_LIST_LIMIT` token specifies the maximum number of artifacts displayed in and exported from the **Planned Tracker Artifacts** tab available in **File Releases**.

Values: Integer

Default: 5000

AUTO_DATA

TeamForge 7.1 and later support automatic password creation. Once you turn on automatic password creation, the `AUTO_DATA` token is auto-generated by the installer during runtime recreation.

> **WARNING:** The `AUTO_DATA` token is not a user-editable site options token. Do not modify this token manually.

The following password-related `site-options.conf` tokens can have the passwords automatically created if set to *$auto$*. When set to *$auto$*, the passwords for the tokens are randomly generated, encrypted and stored in the `AUTO_DATA` token that is added automatically to the `site-options.conf` file during runtime recreation.

```
DATABASE_PASSWORD=$auto$
DATABASE_READ_ONLY_PASSWORD=$auto$
REPORTS_DATABASE_PASSWORD=$auto$
REPORTS_DATABASE_READ_ONLY_PASSWORD=$auto$
ETL_SOAP_SHARED_SECRET=$auto$
JAMES_ADMIN_PASSWORD=$auto$
BDCS_ADMIN_PASSWORD=$auto$
MIRROR_DATABASE_PASSWORD=$auto$
SCM_ADMIN_PASSWORD=$auto$
```

> **WARNING:** As this `AUTO_DATA` token is auto-generated and managed by the TeamForge installer, do not modify this token manually during TeamForge upgrades. When you upgrade TeamForge, you must

> copy this token intact (along with its value) from the old `site-options.conf` file to the upgraded site's `site-options.conf` file, before recreating the runtime. This applies to all the servers in a distributed set up (copy the AUTO_DATA token and its value to all the servers).

This feature is enabled by default. You can, however, override any of the above password-related tokens with the password of your choice.

BCC_MAIL_BATCH_SIZE

Set the `ENABLE_BCC_MONITORING` token to **true** and then depending on your site's requirements, you can set the value of this `BCC_MAIL_BATCH_SIZE` token to configure the number of monitoring emails delivered in a single delivery.

When a work item such as an artifact or discussion is updated, instead of sending separate monitoring emails to all monitoring users, you can now choose to have just one monitoring email sent with all monitoring users added to the BCC. This reduces the load on the email server and results in faster email delivery. You must enable this feature by setting `ENABLE_BCC_MONITORING` to **true** and then, depending on your site's requirements, you can optimize the value of this `BCC_MAIL_BATCH_SIZE` token to increase or decrease the number of emails delivered in a single delivery.

Values: Integer

Default: 100

This token was added in TeamForge 7.2.

BINARY_SETUP_TYPE

Set this token appropriately to have the binary application installed the way you want (as part of the TeamForge installation or upgrade).

TeamForge 8.0 and later support integration with Nexus OSS, an open source repository manager for binary artifacts. By default, the TeamForge installer installs a binary application (referred to as the `binary app` hereinafter), which is essentially a launching pad for all binary integrated applications such as Nexus OSS. Once this binary app is installed as part of TeamForge installation, you can integrate your Nexus servers and repositories with TeamForge.

Though the binary app is installed by default with TeamForge, you can change the value of this site option token, `BINARY_SETUP_TYPE`, to skip binary app installation altogether. You can also have this token configured to have the binary app installed and rolled out for all projects (both existing and new projects to be created) or only for new projects to be created or for select projects on a need basis.

Values:

- all: Binary app is installed and available for all projects.

- new: Binary app is installed and available for new projects only.
- manual: Binary app is installed at a site level and project administrators can add it to select projects on a need basis.
- none: TeamForge installer skips the binary app installation altogether.

Default: new

This token was added in TeamForge 8.0.

BROWSER_NO_CACHE

`BROWSER_NO_CACHE` is a new token added to `runtime-options.conf` file to enable or disable browser caching in TeamForge for better application performance. By default, the token is set to `false` in `runtime-options.conf` file. To disable browser caching, set the value to *true* in `site-options.conf` file.

Values: *true* or *false*

Default: false

DATABASE_NAME

The `DATABASE_NAME` token specifies the name of the site’s database.

Values: Alphanumeric string

Default: teamforge

DATABASE_PASSWORD

The `DATABASE_PASSWORD` token is the password for the Unix user that is authorized to read from and write to the site’s database.

Values: Alphanumeric string

Default: $auto$

DATABASE_SSL

To prevent your data from being exposed in a readable format on the network, use the Secure Socket Layer (SSL) to encrypt the network traffic between the Application and the Database servers. The `DATABASE_SSL` token turns SSL `on` or `off` on sites that use a dedicated operational database server.

Values: `on` or `off`

Default: `off`

DATABASE_TYPE

The `DATABASE_TYPE` token specifies the type of database in which the TeamForge site's data is stored.

Values: postgresql or oracle

Default: postgresql

DATABASE_USERNAME

The `DATABASE_USERNAME` token specifies the Unix user that is authorized to read from and write to the site's database.

Values: Alphanumeric string

Default: teamforge

Comments: For some advanced operations, you may need to log into the database as the database user. However, under normal conditions only the TeamForge site process itself needs to access the database.

DEFAULT_LOCALE

The `DEFAULT_LOCALE` token specifies the language in which automated email messages from the site are generated.

Values:

Default: en

DEFAULT_PROJECT_ACCESS

The `DEFAULT_PROJECT_ACCESS` token specifies the type of access that is assigned to a project when it is created. A project can be private, public, or gated.

Values: private, gated, public

Default: private

DISABLE_CREATE_INTEGRATION_SERVERS

The `DISABLE_CREATE_INTEGRATION_SERVERS` token specifies whether the creation of new SCM integrations is allowed.

Values: true or false

Default: false

Comments: When this token is set to its default value of `false`, you can add SCM integration servers to your TeamForge site. Also, the **Discover Subversion Edge Servers** option, which enables you to find and connect to Subversion Edge servers on your LAN, is available.

DISABLE_REMOTE_PUBLISHING

Publishing repository, like the branding repository, is one of the default repositories that's created automatically when a TeamForge project is created and is intended to contain publicly-consumable files. However, site administrators can toggle access to Publishing Repositories and restrict access based on defined RBAC.

For more information on Publishing Repository, see [What is a Publishing repository? How does it work?][faqs.html#publishingrepository]. This token is set to false by default, meaning the Publishing Repository is publicly accessible.

If set to `true`:

- The Publishing Repository is stripped of its public access and behaves like any other Subversion repository with RBAC applied to it.
- Project Administrators can no longer view or modify the **PROJECT HOME OPTIONS** (see Create a custom project home page).

Values: true or false

Default: false

# DISABLE_USER_SELF_CREATION {#DISABLE_USER_SELF_CREATION}

The `DISABLE_USER_SELF_CREATION` token restricts users from creating their own accounts on the TeamForge home page.

Values: true or false

Default: true

DISCUSSION_ADD_HEADERS

The DISCUSSION_ADD_HEADERS token allows you to add custom headers to the emails posted in the forum.

Values: You can choose to add or remove headers by specifying the particular information you want to be added or dropped from the header. For example, if you add <#d#> in the **Add header** field, the URL of that discussion will be added to the header of all the available messages in that discussion.

Default: None

Example:

```
DISCUSSION_ADD_HEADERS=headername1:value1, name2: value2 , post-id:<#n#>, foru
m-url:<#d#>, message-url:<#m#>, domain:<#h#>, list-name:<#l#>, list-address:<#
l#>@<#h#>
```

Comments: Add one or more header names. The match of any of these headers in an outgoing message (via email) causes its addition with appropriate notification to the posting user.

DISCUSSION_DROP_MIME_TYPES

The DISCUSSION_DROP_MIME_TYPES token allows you to delete the mime types submitted by email that contain arbitrary strings.

Values: image/jpeg,image/jpg,text/xml

Default: Regular expression

Example:

```
DISCUSSION_DROP_MIME_TYPES=image/jpeg,image/jpg,text/xml
```

Comments: Add one or more mime types to the Drop mime types filter. The presence of any of these mime types in an incoming message (via email) causes its deletion with appropriate notification to the posting user. If a mime type is specified in both the Reject and Drop mime filters, then the Reject mime type filter must take higher precedence than the Drop mime type filter.

DISCUSSION_EMAIL_MONITORING

The DISCUSSION_EMAIL_MONITORING token determines which users can monitor a forum on the site.

Values:

Value	Description
0	Allow only forum administrators
1	Users with role permissions
4	All logged in users
5	Allow all site users and guests

Default: 1

Example:

```
DISCUSSION_EMAIL_MONITORING=4
```

Comments: This setting applies to the site as a whole. Project owners can choose to be more restrictive in their own project by selecting a lower value on the project administration page.

DISCUSSION_EMAIL_POSTING

The `DISCUSSION_EMAIL_POSTING` token determines which users on your site can post to forums by e-mail.

Values:

Value	Description
0	Allow only forum administrators
1	Users with role permissions
4	All logged in users
5	Allow known email addresses only
6	Allow all site users and guests

Default: 1

Example:

```
DISCUSSION_EMAIL_POSTING=4
```

Comments: This setting applies to the site as a whole. Project owners can choose to be more restrictive in their own project by selecting a lower value on the project administration page.

DISCUSSION_FORUM_EDITOR

The `DISCUSSION_FORUM_EDITOR` token allows you to choose the type of text that can be used in discussion forum description using the editor tool.

Values: Plain Text

Default: Plain Text

DISCUSSION_MAX_ATTACHMENT_SIZE

The `DISCUSSION_MAX_ATTACHMENT_SIZE` token sets an upper limit to the size of files that users can attach to an email message sent to any discussion forum on the site.

Values: Integer (Megabytes)

Default: blank

Comments: A value of zero or less specifies that there is no limit, which is the same as the default behavior without the token.

DISCUSSION_POST_EDITOR

The `DISCUSSION_POST_EDITOR` token allows you to choose the type of text that can be used for posting in discussion forums using the editor tool.

Values: Plain Text

Default: Plain Text

DISCUSSION_REJECT_CONTENT

The `DISCUSSION_REJECT_CONTENT` token allows you to block the discussion messages submitted by email that contain arbitrary strings.

Values: Regular expression

Default: None

Example:

```
DISCUSSION_REJECT_CONTENT=(?s).*word.*,(?s).*spam.*
```

Comments: Add one or more entries. Each regular expression must match an entire entry. The match of any of these entries in the body or subject of an incoming message (via email) causes its rejection, with appropriate notification to the posting user.

> **NOTE:** The content entry is case sensitive.

DISCUSSION_REJECT_HEADERS

The DISCUSSION_REJECT_HEADERS token allows you to block different headers submitted by email that contain arbitrary strings.

Values: Regular expression

Default: None

Example:

```
DISCUSSION_REJECT_HEADERS=(?s).*headername1:value2.*,(?s).*name2:value2.*
```

Comments: Add one or more header names. Each regular expression must match an entire header name. The match of any of these headers in an incoming message (via email)causes its rejection, with appropriate notification to the posting user.

DISCUSSION_REJECT_MIME_TYPES

The DISCUSSION_REJECT_MIME_TYPES token allows you to delete the mime types submitted by email that contain arbitrary strings.

Values: Application/PDF,text/xml

Default: Regular expression

Example:

```
DISCUSSION_REJECT_MIME_TYPES=application/pdf,text/xml
```

Comments: Add one or more mime types to the Reject MIME types filter. The presence of any of these mime types in an incoming message (via email) will cause its deletion with appropriate notification to the posting user.

DISPLAY_TIMEZONE

The DISPLAY_TIMEZONE token, if set with a preferred time zone, takes precedence over the physical TeamForge server location’s time zone and will be the default time zone that’s displayed throughout the application. In other words, use this token to set a preferred time zone to display across the TeamForge application in case the TeamForge physical server and users are not on the same time zone.

Values: The ID for a time zone can be either a full name such as America/Los_Angeles, or a custom ID in the form GMT[+|-]hh[[:]mm] such as GMT-08:00. It can also be in the form of a three letter abbreviation such as PST.

Default: If set, this token overrides the default time zone of the TeamForge server. TeamForge uses the default time zone of the JVM otherwise.

DOCUMENT_MAX_FILE_UPLOAD_SIZE

By default, you can upload documents of any size in TeamForge. The `DOCUMENT_MAX_FILE_UPLOAD_SIZE` token sets an upper limit to the size of the documents that can be uploaded. Use this token only if you want to restrict the size of the documents.

Values: Integer (specify the number of Megabytes without the suffix `MB`)

Default: blank

Comments: A value of zero specifies that there is no limit, which is the same as the default behavior without the token. In other words, disabling this token, setting a value of 0, or leaving it blank sets the maximum file upload size to unlimited.

DOCUMENT_TEXT_EDITOR

The `DOCUMENT_TEXT_EDITOR` token allows you to choose the type of text that can be used for the document description using the editor tool.

Values: Plain Text

Default: Plain Text

ELASTICSEARCH_JAVA_OPTS

The `ELASTICSEARCH_JAVA_OPTS` token specifies the memory settings for the Java virtual machine that supports Elasticsearch, used by TeamForge Code Search.

Values: Java specifications

Default: `-Xms2g -Xmx2g`

> **NOTE:** By default, Elasticsearch JVM heap size is set to 2GB in TeamForge. You can increase this, if required.

Comments: TeamForge 18.1 (and later) supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.

- JBOSS_JAVA_OPTS

- PHOENIX_JAVA_OPTS
- INTEGRATION_JAVA_OPTS
- ETL_JAVA_OPTS
- ELASTICSEARCH_JAVA_OPTS

TeamForge provision fails on sites that use these options post upgrade to TeamForge 18.1.

ENABLE_BCC_MONITORING

When a work item such as an artifact or discussion is updated, instead of sending separate monitoring emails to all monitoring users, you can now choose to have just one monitoring email sent with all monitoring users added to the BCC. This reduces the load on the email server and results in faster email delivery. Set `ENABLE_BCC_MONITORING` to `true` to enable this feature.

Values: true or false

Default: false

Comments: Depending on your site's requirements, you can also optimize the value of `BCC_MAIL_BATCH_SIZE` to increase or decrease the number of emails delivered in a single delivery.

ENABLE_SITE_NEWS

Site news is disabled by default. Set this token to `true` in `site-options.conf` file, customize the home page of your site and recreate runtime if you want to publish site news on your site's home page.

To publish site news, set this token to `true` in `site-options.conf` file, customize the home page of your site (see "siteNews" html block) and recreate runtime if you want to publish site news on your site's home page.

Values: true or false

Default: false

Comments: This token was added in TeamForge 16.7. Until TeamForge 16.3, regardless of whether you have site news enabled or not, site news were processed in the background. With this ENABLE_SITE_NEWS token, there is no site news processing in the background (by default `ENABLE_SITE_NEWS=false`) thereby improving the site's home page performance a bit.

ENABLE_UI_FOR_CUSTOM_EVENT_HANDLERS

To support branding and customization changes, set the `ENABLE_UI_FOR_CUSTOM_EVENT_HANDLERS` token to `true`.

Values: true or false

Default: true

ENFORCE_MINIMUM_USERNAME_LENGTH

The `ENFORCE_MINIMUM_USERNAME_LENGTH` variable determines the minimum length that can be set for usernames.

Values: 0-31

Default: 0

ETL_JAVA_OPTS

The `ETL_JAVA_OPTS` token specifies the memory settings for the Java virtual machine that supports the ETL (Extract Transform and Load) job.

Values: Java specifications

Default:

```
-Xms160m -Xmx512m -server -XX:+HeapDumpOnOutOfMemoryError -XX:HeapDumpPath=/tm
p -verbose:gc -XX:+ PrintGCTimeStamps -XX:+PrintGCDetails -Dsun.rmi.dgc.client
.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000 - Djava.security.egd
=file:/dev/urandom
```

> **NOTE:** If you've enabled ETL_JAVA_OPTS token in `site-options.conf` file and have added any parameter, you must provide the required JVM heap size as the default heap size is not taken into account.

Comments: TeamForge 18.1 (and later) supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.

- JBOSS_JAVA_OPTS
- PHOENIX_JAVA_OPTS
- INTEGRATION_JAVA_OPTS
- ETL_JAVA_OPTS
- ELASTICSEARCH_JAVA_OPTS

ETL_JOB_THREAD_COUNT

The `ETL_JOB_THREAD_COUNT` token specifies the number of Extract, Transform and Load (ETL) jobs that can be run simultaneously.

Values: 1-100

Default: 2

Comments: If you only have a few jobs to be triggered few times a day, then one thread is sufficient. If you have tens of thousands of jobs, that needs to be triggered every minute, then you should consider increasing the thread count to 50 or 100 (this depends on the nature of the work that your jobs perform, and your resources).

ETL_JOB_TRIGGER_TIME

The `ETL_JOB_TRIGGER_TIME` token specifies the time and date for recurrent Extract, Transform and Load (ETL)jobs.

Values: Cron expression.

Default: 0 30 2 * * ?

Comments: This token takes a cron expression for a value, and not an absolute time value. The default value evaluates to 2.30 a.m. local time. For help with cron expressions, see Cron Trigger Tutorial.

ETL_SOAP_SHARED_SECRET

The `ETL_SOAP_SHARED_SECRET` token enables users to access site-wide reporting data via a SOAP client.

Values: String (possibly encrypted).

Default: mightyetlsoapsecret

FILTER_DROPDOWN_MAX_SELECTION

By default, the drop-down lists with multi-select feature let you select up to 10 filter values. However, you can set any value that suits your requirement for this `FILTER_DROPDOWN_MAX_SELECTION` token to increase or decrease the count.

Values: Any positive integer.

Default: 10

This token was added in TeamForge 7.1.

FORBIDDEN_PASSWORD

The `FORBIDDEN_PASSWORD` token restricts specified words from being used as passwords.

Values: Comma-separated strings

Default: None

GERRIT_DATABASE_HOST

This is the Gerrit Postgres database host. The Gerrit configuration property, database.hostname, is derived from the value of `GERRIT_DATABASE_HOST` token in the `runtime-options.conf`. It can be overridden in `site-options.conf` and requires runtime creation with execution of the post installation script.

Default: 127.0.0.1

GERRIT_DATABASE_NAME

This refers the Gerrit database schema. The Gerrit configuration property, database.database, is derived from the value of `GERRIT_DATABASE_NAME` in `runtime-options.conf`. It can be overridden in `site-options.conf` and requires runtime creation with execution of post installation script.

Default: reviewdb

GERRIT_DATABASE_USER

This is the PostgresDB role name that has access to the Gerrit database `GERRIT_DATABASE_NAME`. The Gerrit configuration property, `database.username`, is derived from the value of `GERRIT_DATABASE_USER` in the `runtime-options.conf`. It can be overridden in `site-options.conf` and requires runtime creation with execution of post installation script.

Default: gerrit

GERRIT_GIT_PUSH_THRESHOLD

The `GERRIT_GIT_PUSH_THRESHOLD` token determines the maximum number of commits in a single Git push. If the limit exceeds, only a single commit object is created in the TeamForge.

Values: Any positive integer.

Default: 30

GERRIT_GIT_REFRESH_PERIOD

The `GERRIT_GIT_REFRESH_PERIOD` token sets the interval in seconds after which Git Integration synchronizes all the repositories and all RBAC permission with TeamForge.

Values: Number of seconds

Default: 3600 seconds

GERRIT_REPLICATION_MODE

Use this site-options token to set the Git integration server as either `master` or `slave` server. In case you do not want replication (standalone mode) or you have only one primary source for repositories, set this token to master. On the other hand, if you have a master Git integration server and you want to replicate (mirror) its repositories on a secondary slave Git integration server, set this token to slave on the slave Git integration server.

Values: master or slave

Default: master

Comments: By default, in TeamForge 8.1 (and later), this token is set to master in the `runtime-options.conf` during runtime creation. As you cannot change the replication mode of a Git server after initial runtime creation, you have to set this to slave at the very beginning of your installation process in case you want to configure the server as a mirror of a `master` Git server. It is not possible to have a Git master and slave configured on the same node, but you can have multiple masters and slaves in your TeamForge environment. Each slave belongs to exactly one master. Once a replica server is set up, it is not possible to reassign it to a different master Git integration server at a later point in time.

GERRIT_REPLICATION_MASTER_EXTERNAL_SYSTEM_I

If `GERRIT_REPLICATION_MODE` is set to `slave`, this token specifies external system ID of the `master` Git integration server.

Values: Alphanumeric string (exsy<number>) of a master Git integration server

Comments: This token is mandatory if `GERRIT_REPLICATION_MODE` is set to `slave`, without which the runtime creation shall fail. On the contrary, the runtime recreation will also fail if `GERRIT_REPLICATION_MODE` is set to `master` and this token is present in the `site-options.conf` file.

GERRIT_SMTP_SERVER

This is the hostname of the SMTP mail server for Gerrit. The Gerrit configuration property, `sendmail.smtpServer`, is derived from the value of `GERRIT_SMTP_SERVER` in the `runtime-options.conf`. It can be overridden in `site-options.conf` and requires runtime creation with execution of post installation script.

Default: localhost

GERRIT_SYNCH_PORT

The Port over which TeamForge communicates to Gerrit. The Gerrit configuration property, `teamforge.apiPort`, is derived from the value of `GERRIT_SYNCH_PORT` in the `runtime-options.conf`. It can be overridden in `site-options.conf` and requires runtime creation with execution of post installation script.

Default: 9081

GERRIT_USER_EMAIL

This token sets the user email account for sending emails from Gerrit. This refers to all Gerrit servers specified in `site-options.conf` file or through cluster/server specific parameters. For example, the “clusterId/serverId” in `[clusterId/serverId]:gerrit:user.email` refers to the cluster or server that is used.

Values:

Default:

HIGHCHARTS_EXPORT_REQUEST_MAX_WAIT

This token is used to set the number of milliseconds that the Highcharts web application has to wait for response from the phantomjs server before it times out.

TeamForge chart export feature requires a pool of phantomjs servers to be running in the application server that is managed by a Highcharts web application. The phantomjs server pool runs as a blocking queue.

This token is used to specify the number of milliseconds that the Highcharts web application has to wait before it times out. In other words, the phantomjs server is expected to respond to the Highcharts web application within the time limit (in milliseconds) set in `HIGHCHARTS_EXPORT_REQUEST_MAX_WAIT`.

Default: 500 milliseconds

This token was added in TeamForge 7.2.

HIGHCHARTS_EXPORT_REQUEST_POOL_SIZE

TeamForge chart export feature requires a pool of phantomjs servers to be running in the application server that is managed by a Highcharts web application. This token is used to specify the number of phantomjs servers to be running in the pool.

Default: The default pool size is 10.

This token was added in TeamForge 7.2.

HTTPD_LOG_DIR

The `HTTPD_LOG_DIR` token specifies the path where information about the activity of the TeamForge site's Apache service is written.

Values: Path specification

Default: `{__LOG_DIR__}/httpd`

HTTP_MAX_PARAMETERS

The `HTTP_MAX_PARAMETERS` token determines the maximum number of parameters (fields) that could be submitted in one http request. This is relevant on pages such as the User-Role Matrix page. The `HTTP_MAX_PARAMETERS` can range from 0-25000.

Values: 0-25000

Default: 5000

INCLUDE_ORGANIZATION_USER_FIELD

The `INCLUDE_ORGANIZATION_USER_FIELD` token controls whether the organization entry is displayed while creating a user account.

Values: true or false

Default: true

INDEXING_TIMEOUT

The `INDEXING_TIMEOUT` token allows you to configure the time limit for indexing a file.

Values: Integer (number of minutes)

Default: 5

INITIAL_PASSWORD_CHANGE_ACTIVATION_CODE_TIM

An administrator can optionally supply a password when creating a user. If the password is not specified while creating the user, the user is sent an email with a ticket to set the password. This `INITIAL_PASSWORD_CHANGE_ACTIVATION_CODE_TIMEOUT` token sets the duration (in hours) for which the password ticket is valid.

Values: Integer (hours)

Default: 72

INTEGRATION_JAVA_OPTS

This token specifies the memory settings for the Java virtual machine that supports the site's integrated source control services.

Values: Java specifications

Default:

```
-Xms160m -Xmx160m -server -XX:+HeapDumpOnOutOfMemoryError -XX:HeapDumpPath=/tm
p -verbose:gc -XX: +PrintGCTimeStamps -XX:+PrintGCDetails -Dsun.rmi.dgc.client
.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000 - Djava.security.egd
=file:/dev/urandom
```

> **NOTE:** If you've enabled INTEGRATION_JAVA_OPTS token in `site-options.conf` file and have added any parameter, you must provide the required JVM heap size as the default heap size is not taken into account.

Comments: TeamForge 18.1 (and later) supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.

- JBOSS_JAVA_OPTS
- PHOENIX_JAVA_OPTS
- INTEGRATION_JAVA_OPTS
- ETL_JAVA_OPTS
- ELASTICSEARCH_JAVA_OPTS

TeamForge provision fails on sites that use these options post upgrade to TeamForge 18.1.

INTEGRATION_LOG_DIR

The `INTEGRATION_LOG_DIR` token specifies the path where information about the activity of the TeamForge site's source code integrations is written.

Values: Path specification

Default: `{__LOG_DIR__}/integration`

JAMES_GATEWAY_HOST

The `JAMES_GATEWAY_HOST` token specifies a mail server with Internet access, separate from the TeamForge server.

Values: Email address specification

Default: None

Comments:

- Specifying a gateway host assures delivery of site email to users if your TeamForge server cannot connect to a DNS server or cannot get outside connections over port 25.
- The mail account specified must be hosted on a separate server from the TeamForge site server.
- The SYSTEM_EMAIL, ADMIN_EMAIL, and JAMES_POSTMASTER_EMAIL tokens can specify the same address.

> **NOTE:** Specify the gateway host by its fully qualified domain name, not a host name.

JAMES_GATEWAY_*

You can set up TeamForge to relay emails through an SMTP gateway (such as Amazon AES) that uses authentication. By default, James sends emails directly. However, you may prefer relaying emails through an enterprise relay server. Configuring the `JAMES_GATEWAY_*` tokens let you do that.

- `JAMES_GATEWAY_HOST` and `JAMES_GATEWAY_PORT` tokens specify the relay server's FQDN and port to use respectively. The `JAMES_GATEWAY_HOST` token specifies a mail server with Internet access, separate from the TeamForge Application Server. Specify the gateway host by its fully qualified domain name (FQDN), not a host name.

- JAMES_GATEWAY_USERNAME and JAMES_GATEWAY_PASSWORD tokens specify the relay server credentials. These tokens are optional that should only be used if the relay server requires SMTP authentication.

For more information, see Relay Emails Through SMTP Gateway with Authentication.

JAMES_LOG_DIR

The JAMES_LOG_DIR token specifies the path where information about the activity of the TeamForge site's email component is written.

Values: Path specification

Default: {__LOG_DIR__}/james

JAMES_POSTMASTER_EMAIL

The JAMES_POSTMASTER_EMAIL token specifies a valid email address for the person or machine that handles email for the domain, such as postmaster@supervillain.org.

Values: Email address specification

Default: root@{__APPLICATION_HOST__}

Comments:

- The mail account specified must be hosted on a separate server outsideof the TeamForge Application Server.
- The SYSTEM_EMAIL, ADMIN_EMAIL, and JAMES_POSTMASTER_EMAIL tokens can specify the same address.

JBOSS_ALARM_TIMEOUT

The JBOSS_ALARM_TIMEOUT token specifies the time duration within which the JBoss service is expected to respond to requests sent by jboss_watchdog.

Values: Integer

Default: 20

JBOSS_JAVA_OPTS

The `JBOSS_JAVA_OPTS` token specifies the memory settings for the JBoss Java virtual machine.

Values: Java specifications

Default: `-Xms1536m -Xmx1536m`

> **NOTE:** If you've enabled JBOSS_JAVA_OPTS token in `site-options.conf` file and have added any parameter, you must provide the required JVM heap size as the default heap size is not taken into account.

Comments: All JVM parameters but `-Xms1536m` and `-Xmx1536m` have been hard-coded in the TeamForge core application.

You cannot manually configure any of the following default JVM parameters in the `site-options.conf` file.

```
-XX:+UseParallelGC
-XX:MaxMetaspaceSize=512m
-XX:ReservedCodeCacheSize=128M
-server
-XX:+HeapDumpOnOutOfMemoryError
-XX:HeapDumpPath=/tmp -verbose:gc
-XX:+PrintCodeCache
-Djsse.enableSNIExtension=false
-Dsun.rmi.dgc.client.gcInterval=600000
-Dsun.rmi.dgc.server.gcInterval=600000
-Djava.security.egd=file:/dev/urandom
-Djava.awt.headless=true.
```

> **WARNING:** When you change the default value of a JVM parameter such as `-XX:HeapDumpPath`, the JBoss runtime parameters include both the user defined and default values for the JVM parameter. However, JBoss runs with the default value and ignores any user defined value.

TeamForge 18.1 (and later) supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.

- JBOSS_JAVA_OPTS
- PHOENIX_JAVA_OPTS
- INTEGRATION_JAVA_OPTS
- ETL_JAVA_OPTS
- ELASTICSEARCH_JAVA_OPTS

TeamForge provision fails on sites that use these options post upgrade to TeamForge 18.1.

LINUX_USERNAME_MODE_ENABLED

The `LINUX_USERNAME_MODE_ENABLED` token, if set to `true`, overrides the default TeamForge user naming convention that bars user names with anything but an alphabet as the first character. This token is commented out (disabled) by default and is available in the `site-options.conf` file.

> **IMPORTANT:** It is recommended to use this token on sites with CVS integration. Do not set the `LINUX_USERNAME_MODE_ENABLED` to `true` on sites without CVS integration. Instead, you must set the RELAXED_USERNAME_MODE_ENABLED token to `true` in case your site has no CVS integration.

Values: true or false

Default: false

> **WARNING:** While TeamForge can allow user names with anything but an alphabet as the first character, the same may not be true with all or some of the integrated applications you may have on your site. As a word of caution, consider the job at hand and understand the consequences before overriding TeamForge’s default user naming convention.

LISTEN_BACKLOG

The `LISTEN_BACKLOG` token is used to specify the maximum length of the queue for the pending connections in the Apache server.

Values: Integer

Default: The default value is obtained from the system Kernel configuration.

```
/sbin/sysctl -n -e net.ipv4.tcp_max_syn_backlog
```

LOGIN_ATTEMPT_LOCK

This option controls locking out the user account after “n” invalid login attempts.

- Set this to zero or a negative number to lock the user account when the user provides an incorrect password for the first time.
- Set this to a positive number, say “2”, to allow the user two wrong password attempts. The user account would be locked at the “x+1” (here, third) attempt.

When a user’s account is locked, either an administrator must unlock it or the user can use the “Forgot Your Password?” link to reset the password.

You must set the `REQUIRE_PASSWORD_SECURITY` token to `true` in the `site-options.conf` file, for `LOGIN_ATTEMPT_LOCK` setting to take effect.

LISTEN_IP

In a distributed setup, you can use this `<host>:<service>:LISTEN_IP` token to control which IPs the services bind to so that you can make sure that services are not overexposed than necessary.

By default, services bind to the IP address corresponding to the `<host>:PUBLIC_FQDN` token. However, you can override this using the `<host>:<service>:LISTEN_IP` token.

A few use cases:

- In a distributed setup, you may want to bind a particular IP address of the TeamForge database server (PostgreSQL server) to the `ctfcore-database` service:
  `server-01:ctfcore-database:LISTEN_IP = 1.2.3.4`
- To bind the mail service to a particular IP:
  `localhost:mail:LISTEN_IP = 1.2.3.4`
- To make TeamForge listen to a specific IP of a particular server, say the SCM server:
  `myscmserver:LISTEN_IP = 1.2.3.4`
- To bind all your services to one IP address (typically in a single server setup):
  `localhost:LISTEN_IP = 1.2.3.4`

NOTE: You cannot use more than one IP address with the `LISTEN_IP` token.

LOG_DIR

The `LOG_DIR` token specifies the path where TeamForge log files are written.

Values: Path specification

Default: `{__SITE_DIR__}/log`

LOG_QUERY_TIME_THRESHOLD

The `LOG_QUERY_TIME_THRESHOLD` token enables you to log database requests at INFO level if they run longer than a given period.

By default, database requests are logged at DEBUG level. Configuring a value for `LOG_QUERY_TIME_THRESHOLD` causes requests that run for a period greater than that value to be logged at the INFO level, which makes them show up in `vamessages.log`.

Set the value to zero to log all database queries at INFO.

Values: Integer (milliseconds)

Default: 1000

LOGIN_ATTEMPT_LOCK

Use the `LOGIN_ATTEMPT_LOCK` token to set the permissible number of unsuccessful login attempts after which the user account is automatically locked.

Values: 1-3

Important: You can now selectively disable this feature even if you have set the `REQUIRE_PASSWORD_SECURITY` site options token to `true`. Set any negative value (such as ‘-1’ or ‘-10000’) in case you want to disable this feature altogether.

Note that with TeamForge 8.1 and earlier versions, setting `LOGIN_ATTEMPT_LOCK=-1` means the user account would be locked at the very first unsuccessful login attempt. This behavior has been removed in TeamForge 8.2 (and later).

Default: 3

LOGIN_CONFIG_XML_FILE

The `LOGIN_CONFIG_XML_FILE` token specifies the path to the LDAP configuration file.

Values: Path specification

Default: `{__DATA_DIR__}/etc/login-config.xml`

LOGROTATE_ARCHIVE_COUNT

Use the `LOGROTATE_ARCHIVE_COUNT` token to set the number of most recent logs to be preserved at any give point in time.

Values: Any positive integer.

Default: The default value is “7”. Meaning, logs for the last 7 days are preserved at any given point in time. Logs older than 7 days are removed from the log archive folder.

NOTE: This token replaces the `LOGROTATE_MAXAGE` token.

MAX_WWW_CLIENT

The `MAX_WWW_CLIENT` token specifies the maximum number of Tomcat request processing threads to be created by the HTTP connector.

Values: Integer

Default: 220

MIGRATION_LOG_DIR

The `MIGRATION_LOG_DIR` token specifies the path where information about the conversion of site data is written during an upgrade.

Values: Path specification

Default: `{__LOG_DIR__}/runtime`

MAX_PASSWORD_LENGTH

The `MAX_PASSWORD_LENGTH` token sets the longest password that the system allows when a user account is created.

Values: Integer (number of characters)

Default: 256

MAX_POSTS_PER_MINUTE

This token is used to control the number of posts per minute. By default, its value is `10`.

Values: Integer

Default: 10

Comments: Increase the value of this token if you don’t want your users to get blacklisted accidentally.

MINIMUM_PASSWORD_LENGTH

The `MINIMUM_PASSWORD_LENGTH` token sets the shortest password that the system allows when a user account is created.

Values: Integer (number of characters)

Default: 6

MINIMUM_USERNAME_LENGTH

The `MINIMUM_USERNAME_LENGTH` token sets the shortest username that the system allows when a user account is created.

Values: Integer (number of characters)

Default: 3

MIRROR DATABASE HOST

The `MIRROR_DATABASE_HOST` token is a TeamForge database token that specifies the host of the database. This token allows to extract the reporting data from the mirror TeamForge database through the Extract, Transform and Load (ETL) process.

Values: Alphanumeric string

Default: The `MIRROR_` token takes the value of `DATABASE_` token.

Example: Enter `MIRROR_DATABASE_HOST=cu349.cloud.sp.collab.net` (server name)

Add this token to the `site-options.conf` only if you setup a mirror database.

MIRROR_DATABASE_NAME

The `MIRROR_DATABASE_NAME` token is a TeamForge database token that specifies the name of the TeamForge database. This token allows to extract the reporting data from the mirror TeamForge database through the Extract, Transform and Load (ETL) process.

Values: Alphanumeric string

Default: The `MIRROR_` token takes the value of `DATABASE_` token.

Example: Enter `MIRROR_DATABASE_NAME=ctfdb`

Add this token to the `site-options.conf` only if you setup a mirror database.

MIRROR_DATABASE_PASSWORD

The `MIRROR_DATABASE_PASSWORD` token is a TeamForge database token that specifies the password of the database. This token allows to extract the reporting data from the mirror TeamForge database through the Extract, Transform and Load (ETL) process.

Values: Alphanumeric string

Default: The `MIRROR_` token takes the value of `DATABASE_` token.

Example: Enter `MIRROR_DATABASE_PASSWORD=ctfpwd`

Add this token to the `site-options.conf` only if you setup a mirror database.

MIRROR_DATABASE_PORT

The `MIRROR_DATABASE_PORT` token is a TeamForge database token that specifies the port number of the database. This token allows to extract the reporting data from the mirror TeamForge database through the Extract, Transform and Load (ETL) process.

Values: Port specification

Default: The `MIRROR_` token takes the value of the `DATABASE_` token.

Example: Enter `MIRROR_DATABASE_PORT=5432`.

Add this token to the `site-options.conf` only if you setup a mirror database.

MIRROR_DATABASE_USERNAME

The `MIRROR_DATABASE_USERNAME` token is a TeamForge database token that specifies the database user’s name. This token allows to extract the reporting data from the mirror TeamForge database through the Extract, Transform and Load (ETL) process.

Values: Alphanumeric string

Default: The `MIRROR_` token takes the value of the `DATABASE_` token.

Example: Enter `MIRROR_DATABASE_USERNAME=ctfuser`.

Add this token to the `site-options.conf` only if you setup a mirror database.

NOTIFY_SITE_ADMINS_FOR_SITE_ACTIVITIES

The `NOTIFY_SITE_ADMINS_FOR_SITE_ACTIVITIES` token ensures that the activities at the site level are intimated to the site administrators through email notifications.

The site administrator can receive notifications on the following operations:

- User creation
- Project creation
- Blacklisted users
- SCM operations

Values: true or false

Default: true

This token was added in TeamForge 8.0.

OBFUSCATION_ENABLED

The `OBFUSCATION_ENABLED` token is used to run the TeamForge application in the obfuscation mode for security purpose. Password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

Values: true or false

Default: true

Comments: When the TeamForge application is running in the obfuscation mode, the database login credentials, shared secrets etc., are encrypted and stored in the TeamForge configuration files for security reasons.

OBFUSCATION_KEY

The `OBFUSCATION_KEY` token is used by the TeamForge obfuscation component as an input to the obfuscation algorithm for encryption and decryption purposes.

Values: AlphaNumeric (length greater than or equal to 8 bytes)

Default: `XSJt43wN`

ONLY_SITE_ADMIN_CAN_EDIT_SINGLE_SIGN_ON

This site-options token, if set to `true`, ensures that only site administrators can turn on single sign on (SSO) for linked applications (including Build & Test). Set this token to `false` to have both site and project administrators turn SSO `on` and `off`.

Values: Either `true` or `false`.

Default: true

This token was added in TeamForge 7.2.

ORGANIZATION_EDITABLE

The `ORGANIZATION_EDITABLE` token allows or prevents editing the organization value of a user account.

Values: true or false

Default: true

PASSWORD_CONTROL_EFFECTIVE_DATE

The `PASSWORD_CONTROL_EFFECTIVE_DATE` token is used to set the date from which the password security feature takes effect.

Values: Date (mm/dd/yyyy)

Comments: The REQUIRE_PASSWORD_SECURITY is the master token that enables the password security feature.

> **IMPORTANT:** Setting the `PASSWORD_CONTROL_EFFECTIVE_DATE` token with a date is mandatory if `REQUIRE_PASSWORD_SECURITY` is set to `true`.

Example 1:

Consider a site with 130 users on which the password control kit (PCK) was not active. Of the 130 users, assume that:

- 100 users did not change password in last 100 days.
- 20 users did not change password in last 85 days.
- 10 users did not change password in last 75 days.

Assume that the following tokens are set on 01/01/2014 (current date):

```
REQUIRE_PASSWORD_SECURITY=true
PASSWORD_WARNING_PERIOD=20
PASSWORD_EXPIRY_PERIOD=90
PASSWORD_DISABLE_PERIOD=30
PASSWORD_DELETE_PERIOD=60
```

PCK runs on 01/01/2014 and if you have `PASSWORD_CONTROL_EFFECTIVE_DATE=01/10/2014` (set to a future date):

- 100 users with no password change for the past 100 days would get a warning message that their passwords will expire in 10 days.
- 20 users with no password change for the past 85 days would get a warning message that their passwords will expire in 10 days.
- 10 users with no password change for the past 75 days would get a warning message that their passwords will expire in 15 days.

Example 2: Consider the following scenario in which:

- Current date = 01/01/2014
- PASSWORD_CONTROL_EFFECTIVE_DATE=01/01/2013

In this scenario, the password control effective date is set to a date in the past. As a result, password control takes immediate effect and the PCK starts disabling, deleting or expiring user accounts right away.

PASSWORD_DELETE_PERIOD

The `PASSWORD_DELETE_PERIOD` token specifies the time frame within which a disabled user account is automatically deleted.

Values: Integer (number of days)

Default: 60

> **NOTE:** The PASSWORD_DELETE_PERIOD can be disabled by setting the value to zero.

PASSWORD_DISABLE_PERIOD

The `PASSWORD_DISABLE_PERIOD` token specifies the time frame within which a user (soft-expired) is turned into a disabled user.

Values: Integer (number of days)

Default: 30

Comments: A value of zero will disable this feature.

PASSWORD_EXPIRY_PERIOD

The `PASSWORD_EXPIRY_PERIOD` token specifies the number of days after which the users' password expires.

Values: Integer (number of days)

Default: 90

> **NOTE:** You cannot disable the password expiry feature by setting this token to `0`.

PASSWORD_REQUIRES_MIXED_CASE

The `PASSWORD_REQUIRES_MIXED_CASE` token specifies that the user password must contain mixed case letters.

Values: true or false

Default: true

PASSWORD_REQUIRES_NON_ALPHANUM

The `PASSWORD_REQUIRES_NON_ALPHANUM` token specifies that the user password must contain a non-alphanumeric character.

Values: true or false

Default: true

PASSWORD_REQUIRES_NUMBER

The `PASSWORD_REQUIRES_NUMBER` token specifies that the user password must atleast contain one number.

Values: true or false

Default: true

PASSWORD_WARNING_PERIOD

Set this token to alert users via emails about impending password expiration on a daily basis. Email alert starts “N” days before password expiration due date, where `PASSWORD_WARNING_PERIOD=N`, and ends only when the password is changed by the user.

Values: Positive integer (number of days).

Default: 14

PASSWORD_HISTORY_AGE

The maximum allowed value of `PASSWORD_HISTORY_AGE` token is `10`. This option disallows the previous “n” passwords, while setting a password. However, if this option is set to zero, a negative number or it is left empty, the user can use any previous password. The password being set must satisfy the existing password policy each time.

You must set the `REQUIRE_PASSWORD_SECURITY` token to `true` in the `site-options.conf` file, for `PASSWORD_HISTORY_AGE` security setting to take effect.

PHOENIX_JAVA_OPTS

The `PHOENIX_JAVA_OPTS` token specifies the memory settings for the Java virtual machine that supports the site’s ability to send and receive email and to index data for search.

Values: Java specifications

Default:

```
-Xms256m -Xmx256m -server -XX:+HeapDumpOnOutOfMemoryError -XX:HeapDumpPath=/tm
p -verbose:gc -XX:+ PrintGCTimeStamps -XX:+PrintGCDetails -Dsun.rmi.dgc.client
.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000 -Dsf .luceneOptimize
Every=100000 -Djava.security.egd=file:/dev/urandom
```

> **NOTE:** If you’ve enabled PHOENIX_JAVA_OPTS token in `site-options.conf` file and have added any parameter, you must provide the required JVM heap size as the default heap size is not taken into account.

Comments: TeamForge 18.1 (and later) supports Java 9. As a result of changes to the logging framework in Java 9, the `PrintGCDetails` and `PrintGCTimeStamps` logging options are no longer supported. Remove these options from the following tokens while upgrading to TeamForge 18.1 or later.

- JBOSS_JAVA_OPTS

- PHOENIX_JAVA_OPTS
- INTEGRATION_JAVA_OPTS
- ETL_JAVA_OPTS
- ELASTICSEARCH_JAVA_OPTS

TeamForge provision fails on sites that use these options post upgrade to TeamForge 18.1.

PGSQL_COMMIT_DELAY

The `PGSQL_COMMIT_DELAY` token specifies the time delay between writing a commit record to the write ahead log (WAL) buffer and flushing the buffer out to disk.

Values: Integer (in microseconds)

Default: 250

Comments: Together with the `PGSQL_COMMIT_SIBLINGS` token, this token allows a group of otherwise unrelated transactions to be flushed to disk at the same time, with possible significant performance gain.

PGSQL_COMMIT_SIBLINGS

The `PGSQL_COMMIT_SIBLINGS` token sets the minimum number of concurrent open transactions to require before performing the delay specified by the `PGSQL_COMMIT_DELAY` option.

Values: Integer

Default: 10

Comments: Together with the `PGSQL_COMMIT_DELAY` token, this token allows a group of otherwise unrelated transactions to be flushed to disk at the same time, with possible significant performance gain.

PGSQL_EFFECTIVE_CACHE_SIZE

The `PGSQL_EFFECTIVE_CACHE_SIZE` token specifies the size of the OS data cache that is available to PostgreSQL. PostgreSQL can use that data to select the optimal way to execute requests.

Comments: The right value for this token depends in part on the available RAM on the server where your site is running. Set this value at the highest amount of RAM that you expect to be always available to PostgreSQL.

See What are the right PostgreSQL settings for my site? for values recommended by CollabNet.

PGSQL_LOG_DIR

The `PGSQL_LOG_DIR` token specifies the path where information about the activity of the TeamForge site's PostgreSQL database is written.

Values: Path specification

Default: `{__LOG_DIR__}/pgsql`

PGSQL_MAINTENANCE_WORK_MEM

The `PGSQL_MAINTENANCE_WORK_MEM` token specifies the maximum amount of memory to be used in maintenance operations such as VACUUM.

Comments: See What are the right PostgreSQL settings for my site? for values recommended by CollabNet.

PGSQL_MAX_CONNECTIONS

The `PGSQL_MAX_CONNECTIONS` token determines the number of concurrent connections available to the database server.

Values: Integer

Default: 135

PGSQL_MAX_FSM_PAGES

The `PGSQL_MAX_FSM_PAGES` token tells the vacuum process how many pages to look for in the shared free-space map.

Values: Integer

Default: 500000

Comments: Each FSM page uses 6 bytes of RAM for administrative overhead, so increasing FSM substantially on systems low on RAM may be counter-productive.

PGSQL_MAX_FSM_RELATIONS

The `PGSQL_MAX_FSM_RELATIONS` token specifies how many relations (tables) will be tracked in the free space map.

Default: 500

PGSQL_MAX_STACK_DEPTH

The `PGSQL_MAX_STACK_DEPTH` token specifies the maximum safe depth of the server's execution stack.

Values: Integer

Default: 5120

PGSQL_SHARED_BUFFERS

The `PGSQL_SHARED_BUFFERS` token defines a block of memory that PostgreSQL will use to hold requests that are awaiting attention from the kernel buffer and CPU.

Comments: The right value for this token depends in part on the available RAM on the server where your site is running.

See What are the right PostgreSQL settings for my site? for values recommended by CollabNet.

PGSQL_STATEMENT_TIMEOUT

The `PGSQL_STATEMENT_TIMEOUT` token is set to prevent the Postgres queries from running for a long period of time.

Values: Integer (Milliseconds)

Default: 600000 (Milliseconds)

Comments: An error message is displayed for every timeout in the `postgres.log` file and the log message with the exid id is logged in the `vamessages.log` and `server.log` files.

PGSQL_VACUUM_COST_DELAY

The `PGSQL_VACUUM_COST_DELAY` token controls the length of time that an I/O process will sleep when the limit set by `vacuum_cost_limit` has been exceeded.

Values: Integer (milliseconds)

Default: 50

PGSQL_WAL_BUFFERS

The PGSQL_WAL_BUFFERS token specifies the number of buffers available for the Write Ahead Log.

Comments: If your database has many write transactions, setting this value bit higher than default may result better usage of disk space.

See What are the right PostgreSQL settings for my site? for values recommended by CollabNet.

PGSQL_WORK_MEM

The PGSQL_WORK_MEM token specifies the amount of memory to be used by internal sort operations and hash tables before switching to temporary disk files. .

Comments: The right value for this token depends in part on the available RAM on the server where your site is running.

See What are the right PostgreSQL settings for my site? for values recommended by CollabNet.

PLANNING_BOARD_SWIM_LANE_LIMIT

By default, not more than 250 cards are shown in a planning board swimlane. However, as a site administrator, you can increase or decrease the number of cards shown in the planning board swimlanes by configuring the site options token, PLANNING_BOARD_SWIM_LANE_LIMIT.

Values: A positive number.

Default: 250

Comments: When you select a planning folder in one of the swimlanes and if X is greater than N, (where X = number of artifacts in the selected planning folder and N = PLANNING_BOARD_SWIM_LANE_LIMIT), the message, Swimlanes in the Board View is currently configured to show N artifacts only, appears at the bottom of the swimlane.

PLANNING_FOLDER_DESC_EDITOR

The PLANNING_FOLDER_DESC_EDITOR token allows you to choose the type of text that can be used in the planning folder description using the editor tool.

Values: Plain Text

Default: Plain Text

RELAXED_USERNAME_MODE_ENABLED

The `RELAXED_USERNAME_MODE_ENABLED` token, if set to true, overrides the default TeamForge user naming convention that bars user names with anything but an alphabet as the first character. It is recommended to use this token on sites without CVS integration.

> **WARNING:** Do not set the `RELAXED_USERNAME_MODE_ENABLED` to `true` on sites with CVS integration. Instead, you must set the LINUX_USERNAME_MODE_ENABLED token to `true` in case your site has CVS integration.

Values: true or false

Default: false

Comments: This token is commented out (disabled) by default and is available in the `site-options.conf` file.

> **WARNING:** While TeamForge can allow user names with anything but an alphabet as the first character, the same may not be true with all or some of the integrated applications you may have on your site. As a word of caution, consider the job at hand and understand the consequences before overriding TeamForge's default user naming convention.

REPORTS_DATABASE_NAME

The `REPORTS_DATABASE_NAME` token specifies the name of the site's reporting database, also known as the datamart.

Values: Alphanumeric string

Default: teamforge_datamart

Comments: It is OK for this token to have the same value as `DATABASE_NAME`, because they are running in separate pgsql processes.

REPORTS_DATABASE_PASSWORD

The `REPORTS_DATABASE_PASSWORD` token is the password for the Linux user that is authorized to read from and write to the site's reporting database.

Values: Alphanumeric string

Default: $auto$

Comments: It is OK for this token to have the same value as DATABASE_PASSWWORD, because they are running in separate PostgreSQL processes.

REPORTS_DATABASE_PORT

The REPORTS_DATABASE_PORT token defines a separate port for the reporting database (aka datamart). Using a separate port can improve site performance when database utilization is high.

Values: Port specification

Default: 5632

Comments: As of TeamForge 6.1, only port 5632 is supported.

Separate Ports for Database and Datamart on the Same Server

The ability to run separate PostgreSQL instances for TeamForge database and datamart on the same server is being deprecated in TeamForge 17.11. During TeamForge installation, the REPORTS_DATABASE_PORT token should no longer be used to assign a separate port for datamart on the server that also runs the TeamForge database. The following warning shows up if you use the REPORTS_DATABASE_PORT token with a custom port number (other than the default value, which is 5432).

```
Using two separate Postgres clusters for database and datamart on the same mac
hine is deprecated. Consider deploying the two clusters on two machines or usi
ng a single cluster for both databases.
```

If you have the TeamForge database and datamart running on separate PostgreSQL instances on the same server:

- **New hardware upgrade**: If you are upgrading on a new hardware, it is highly recommended to create a dump of both the database and datamart and load them into the same PostgreSQL instance. For more information, see Upgrade on new hardware: Create a single cluster for both Database and Datamart.
- **Same hardware upgrade**: If you are upgrading on the same hardware, you may still choose to use the REPORTS_DATABASE_PORT and have the database and datamart running on two separate PostgreSQL instances. However, support for REPORTS_DATABASE_PORT token may end in one of the future TeamForge releases, when you may have to dump and load both the database and datamart on the same PostgreSQL instance anyway.

REPORTS_DATABASE_USERNAME

The REPORTS_DATABASE_USERNAME token specifies the Linux user that is authorized to read from and write to the site’s reporting database.

Values: Alphanumeric string

Default: teamforge_datamart

Comments: For some advanced operations, you may need to log into the database as the database user. However, under normal conditions only the TeamForge site process itself needs to access the database.

It is OK for this token to have the same value as `DATABASE_USERNAME`, because they are running in separate PostgreSQL processes.

REPORTS_ENABLE_REPORT_GENERATION

The `REPORTS_ENABLE_REPORT_GENERATION` token is used to enable or disable the **Reports** tab in the UI.

Values: true or false

Default: true or false

Comments Datamart is enabled by adding the 'datamart' service to the HOST_<hostname>token. The service is disabled if datamart is not added. The default value of the REPORTS_ENABLE_REPORT_GENERATION token is based on this service.

REQUIRE_PASSWORD_SECURITY

The `REQUIRE_PASSWORD_SECURITY` token, if set to true, enforces password security policy for the site.

Values: true or false

Default: true

Comments: This token can be useful when an organization's security policy prohibits users from entering passwords without any restrictions. You can also set the `PASSWORD_CONTROL_EFFECTIVE_DATE` token with a date from which the password policy would be enforced. For more information, see PASSWORD_CONTROL_EFFECTIVE_DATE.

REQUIRE_RANDOM_ADMIN_PASSWORD

The `REQUIRE_RANDOM_ADMIN_PASSWORD` token restricts users from setting a random admin password.

Values: true or false.

Default: True (SaaS), false (BTF)

Comments: This token, when set to `true`, checks for a valid mail id in the `ADMIN_EMAIL` token.

REQUIRE_USER_PASSWORD_CHANGE

The `REQUIRE_USER_PASSWORD_CHANGE` token determines if the user password needs to be changed during the first login instance.

Values: true or false.

Default: true

Comments: Setting this token to `true` makes the new system force users to change password during first login and false otherwise.

RUNTIME_LOG_DIR

The `RUNTIME_LOG_DIR` token specifies the path where information about the activity of the TeamForge site's runtime environment is written.

Values: Path specification

Default: `{__LOG_DIR__}/runtime`

SAFE_DOWNLOAD_MODE

Use this token to enforce downloading and saving of documents, attachments and files locally using the "Save" dialog box instead of inline views.

Values: true, false, none, all, html

You can set this token to "none" "all" or "html" that will force download of nothing, everything, or just html documents respectively.

Default: true

SCM_DEFAULT_SHARED_SECRET

The `SCM_DEFAULT_SHARED_SECRET` token allows SCM Integrations to securely communicate with the TeamForge Application Server.

Values:

- Alpha-numeric
- Special characters like '~!@#$%^&*'
- 16-24 byte length

Default: The default value is automatically generated during runtime.

SCM_SOAP_TIMEOUT

The SCM_SOAP_TIMEOUT token is used to specify the connection timeout of the SCM soap requests between the APP and SCM servers.

Values: Integer (Milliseconds)

Default: 300000

SCM_USER_ENCRYPTED_PASSWORD

The SCM_USER_ENCRYPTED_PASSWORD token is used to store the encrypted scmviewer password.

Values:

- Alpha-numeric
- Special characters like '~!@#$%^&*'

Default: The default value will be in the encrypted format. See password_util.sh for more information.

SEARCH_LOG_DIR

The SEARCH_LOG_DIR token specifies the path where information about the activity of the TeamForge site's Lucene search component is written.

Values: Path specification

Default: {__LOG_DIR__}/james

SEARCH_MAX_FILE_SIZE

The SEARCH_MAX_FILE_SIZE token sets an upper limit to the size of files that are indexed for search.

Values: Integer (bytes)

Default: 10M

Comments: A value of zero or less specifies that there is no limit, which is the same as the default behavior without the token.

SEARCH_SUPPRESS_ARCHIVE_SUB_DOCS

The `SEARCH_SUPPRESS_ARCHIVE_SUB_DOCS` token prevents archive files from being indexed for search.

Archive files include `zip`, `gzip`, `tar`, and similar file types. They also include document files that are stored in archive format, such as `docx` files from Microsoft Word 2007.

Values: true, false

Default: true

SESSION_COOKIES_ONLY

the `SESSION_COOKIES_ONLY` token restricts the persistence of all cookies to the user's current session.

If `SESSION_COOKIES_ONLY=true`, then all cookies created during the user session expire automatically when the user closes their browser. If it is `false`, the cookie expires according to the system logic for that particular cookie.

Values: true or false

Default: false

Comments: This token can be useful when an organization's security policy prohibits cookies that persist across user sessions.

SESSION_TIMEOUT

Use this token to set the user session timeout duration for newly created Application Server/Integration Server sessions.

Values: A positive value in the range of 1-1440.

Default: The default value of the `SESSION_TIMEOUT` token is 30 minutes (for security reasons). You may change this to any value in the range of 1-1440 (minutes). However, you must create runtime for the changes to take effect.

SOAP_ANONYMOUS_SHARED_SECRET

The `SOAP_ANONYMOUS_SHARED_SECRET` token allows users to have an anonymous login to the TeamForge site through SOAP.

Values: String (possibly encrypted)

Default: None

Comments: The token must be configured to a non-empty value if users need to have an anonymous login to the site through SOAP. A value must be provided if site-wide reporting is enabled.

SOAP_ARTIFACT_LIST_LIMIT

The `SOAP_ARTIFACT_LIST_LIMIT` token is used to limit the number of artifacts returned via SOAP calls.

Values: Integer

Default: -1

This means that the artifact list retrieved via SOAP is unlimited.

Comments: In TeamForge releases earlier than 6.1.1, SOAP calls returned everything that was asked for, and that is the default behavior in TeamForge 6.1.1 as well. However, sites with performance and stability issues (OutOfMemory errors) in returning a large number of artifacts can now limit the number using this token. Changing this value requires a recreate-runtime and thus a site restart.

> **IMPORTANT:** Increasing the number of artifacts beyond the optimal 20,000 - 25,000 range might cause a heap dump.

SSL

The `SSL` token activates Secure Socket Layer encryption for the TeamForge site.

Values: on or off

Default: on

SSL_CERT_FILE

The `SSL_CERT_FILE` specifies the path where the TeamForge site’s Secure Socket Layer certificate is stored.

Values: Path specification

Default: None

SSL_CHAIN_FILE

The `SSL_CHAIN_FILE` token specifies the path where the TeamForge site's SSL certficate chain file is stored.

Values: Path specification

Default: None

SSL_CIPHER_SUITE

The `SSL_CIPHER_SUITE` token disables some of the less secure methods.

Values: SSLCipherSuite method

Default:

```
ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM
-SHA384:ECDHE-ECDSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:DHE-DSS-AES128
-GCM-SHA256:kEDH+AESGCM:ECDHE-RSA-AES128-SHA256:ECDHE-ECDSA-AES128-SHA256:ECDH
E-RSA-AES128-SHA:ECDHE-ECDSA-AES128-SHA:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AE
S256-SHA384:ECDHE-RSA-AES256-SHA:ECDHE-ECDSA-AES256-SHA:DHE-RSA-AES128-SHA256:
DHE-RSA-AES128-SHA:DHE-DSS-AES128-SHA256:DHE-RSA-AES256-SHA256:DHE-DSS-AES256-
SHA:DHE-RSA-AES256-SHA:AES128-GCM-SHA256:AES256-GCM-SHA384:AES128-SHA256:AES25
6-SHA256:AES128-SHA:AES256-SHA:AES:CAMELLIA:DES-CBC3-SHA:!aNULL:!eNULL:!EXPORT
:!DES:!RC4:!MD5:!PSK:!aECDH:!EDH-DSS-DES-CBC3-SHA:!EDH-RSA-DES-CBC3-SHA:!KRB5-
DES-CBC3-SHA
```

SSL_PROTOCOL

The `SSL_PROTOCOL` token disables some of the less secure methods.

Values: SSLProtocol method

Default: all -SSLv3 -SSLv2

SSL_KEY_FILE

The `SSL_KEY_FILE` specifies the path where the TeamForge site's RSA private key is stored when Secure Socket Layer encryption is in effect.

Values: Path specification

Default: None

SUBVERSION_BRANDING_URI

The SUBVERSION_BRANDING_URI token specifies the path component of the data repository URL.

Values: BDB or FSFS

Default: BDB

SVN_AUTHNZ_TIMEOUT

The SVN_AUTHNZ_TIMEOUT token allows you to set the timeout value (in seconds) for the mod_authnz_ctf module.

Values: Timeout value in number of seconds.

Default: 60

SYSTEM_EMAIL

The SYSTEM_EMAIL token specifies a valid email address for the system administrator responsible for this site.

- System administrators can use this email address to set up outage alerts and other notifications.
- The mail account specified must be hosted on a separate server from the TeamForge site server.
- The SYSTEM_EMAIL, ADMIN_EMAIL, and JAMES_POSTMASTER_EMAIL tokens can specify the same address.

Values: Email address specification

Default: root@{__APPLICATION_HOST__}

> **IMPORTANT:** In TeamForge 6.x, the sender name and address for system-generated emails is taken from the value of the SYSTEM_EMAIL token. Therefore, changing the admin user's full name or email address does not affect the sender details of system-generated emails. This is different from TeamForge 5.x, in which the sender name and address for system-generated emails is derived from the admin user's full name and email address.

USE_BROWSER_CACHE_PASSWORD

The USE_BROWSER_CACHE_PASSWORD token restricts the storage of password in the browser when you login to the site.

Values: true/false

Default: true

USE_EXTERNAL_USER_AUTHENTICATION

The `USE_EXTERNAL_USER_AUTHENTICATION` token specifies whether users can be authenticated through a separate system, such as OpenLDAP.

Values: true or false

Default: false

USE_STATIC_SENDER_EMAIL

The `USE_STATIC_SENDER_EMAIL` token, if set to `true`, assigns the `Return-Path` parameter in the TeamForge notification email header with a `noreply@<user's email domain>` email ID. This prevents Out of Office replies from being posted to artifacts and discussion forums.

On the other hand, the `USE_STATIC_SENDER_EMAIL` token, if set to `false`, assigns the `Return-Path` parameter in the TeamForge notification email header with the email ID of the user (for example, tom@forge.collab.net) whose action triggers the notification email. In this case, Out of Office replies are posted to artifacts and discussion forums. If the user is a Site Administrator, the `Return-Path` parameter in the TeamForge notification email header is assigned with the email ID `root@<TeamForge domain name>`.

Values: true or false

Default: false

USER_ACCOUNT_RESTRICTED

The `USER_ACCOUNT_RESTRICTED` token determines whether newly created users are “restricted” or “unrestricted” users by default.

- Restricted users can access only public projects and projects of which they are members.
- Unrestricted users can access all projects except private projects of which they are not members.

Values: true or false

Default: true

USER_MONITORING_REMOVE_ENABLED

Set the USER_MONITORING_REMOVE_ENABLED token to true, if you want to enable the feature that lets you remove one or more users from monitoring selected TeamForge objects.

Values: true or false

Default: false

USER_NEED_PERMISSION_TO_VIEW_FULL_USER_DETA

The USER_NEED_PERMISSION_TO_VIEW_FULL_USER_DETAILS token restricts users from viewing other users' organization information.

Values: true or false

Default: false

USERS_WITH_NO_EXPIRY_PASSWORD

The USERS_WITH_NO_EXPIRY_PASSWORD token specifies the users for whom there is no expiry of password. The token is enabled by default.

Values: Specify the usernames (for the user accounts) for which there is no expiry of password.

Default: USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin

Using Multi-line Blocks for Site Options

The multi-line block configuration is generally used by old SFEE sites. To define a site-options.conf token with a multi-line block value, you need to follow a certain syntax.

- Declare the token name with the value START_MULTILINE_BLOCK. Syntax: <TOKEN_NAME>=START_MULTILINE_BLOCK
- Specify the multi-line values beneath the token.
- Complete the multi-line block with END_MULTILINE_BLOCK after all the multi-line values are specified. Syntax: END_MULTILINE_BLOCK

Example:

```
SOURCEFORGE_CONFIGURATION_PROPERTIES_APPEND=START_MULTILINE_BLOCK
email.suppress.project_member_added=true
email.suppress.scm_user_password_synchronized=true
END_MULTILINE_BLOCK
```

Use the Configure Application tool to define your site level TeamForge settings.

The **Configure Application** tool, added in TeamForge 17.4, makes TeamForge site level settings configurable via the user interface. This comes in handy for site administrators, who otherwise would have to work with the `site-options.conf`file and then recreate the TeamForge runtime for site level configuration changes.

To modify your site settings, select **My Workspace > Admin** and select **Projects > System Tools > Configure Application**, modify the available site settings as required and click **Save**.

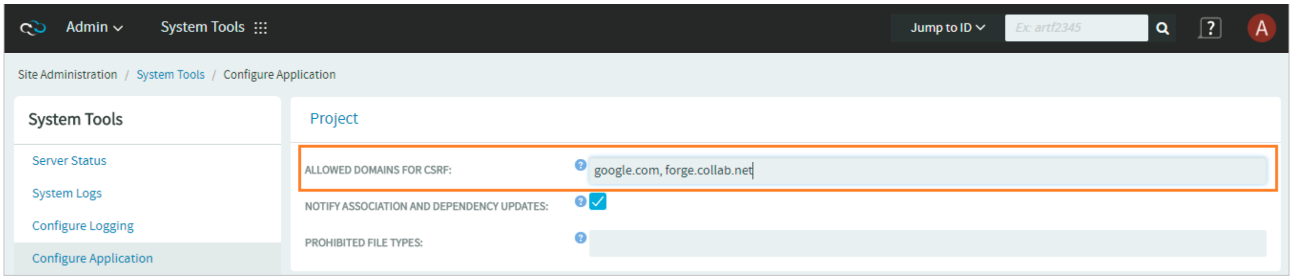
Project

Project setting(s) that apply globally to all the projects in your TeamForge site:

ALLOWED DOMAINS FOR CSRF

TeamForge administrators can now prevent CSRF by setting listing the domains that are allowed for CSRF.

The **ALLOWED DOMAINS FOR CSRF** parameter specifies the list of allowed domains to prevent users from performing any CSRF related activities. This list of domains is validated against the origin and the referer headers for any incoming requests into TeamForge. This is helpful while configuring TeamForge for cross-origin requests. The default value is “*” which allows all types of domains or URLs. Example URLs/domains: forge.collab.net, forge.collab.net/sf/sfmain/do/home.



ALLOWED DOMAINS FOR CSRF: google.com, forge.collab.net

NOTIFY ASSOCIATION AND DEPENDENCY UPDATES: ☑

PROHIBITED FILE TYPES:

NOTIFY ASSOCIATION AND DEPENDENCY UPDATES

Notification emails can be sent to all monitoring users when an association or dependency is added to an object. You can select the **NOTIFY ASSOCIATION AND DEPENDENCY UPDATES** check box to let TeamForge send notification emails to all monitoring users when an association or dependency is added. Clear this check box otherwise.

PROHIBITED FILE TYPES

You can restrict users from uploading specific file types. Add a list of comma-separated file extensions in this field to prevent those file types from being uploaded. For example, adding “exe,jar” prevents `.exe` and `.jar` files from being uploaded to TeamForge.

> **NOTE:** There are certain file upload restrictions already in place in TeamForge. For example, you can upload only an image file for a user’s profile picture in the **My Settings** page. Such restrictions, by default, override the site level settings configured in the **Configure Application** page.

Tracker

Tracker setting(s) that apply globally to all the trackers in your TeamForge site:

MASS IMPORT ARTIFACTS LIMIT

You can restrict the number of artifacts that can be mass-imported. Type the maximum number of artifacts that you want to allow via mass import in the **MASS IMPORT ARTIFACTS LIMIT** text box.

External Authentication

These settings are tied to any external authentication.

ALLOW DATABASE AUTHENTICATION IF LDAP IS ENABLED

Select this check box to have LDAP credentials stored in TeamForge and have users authenticated via TeamForge every time a user logs in. This helps improve performance by optimizing the number of authentication calls between TeamForge and LDAP servers.

> **IMPORTANT:** Enabling this parameter is mandatory for sites with internally managed CVS servers.

FORCE RE-AUTHENTICATION WITH LDAP SERVER

If you have enabled database authentication, LDAP user credentials are stored when users login for the first time and continue to login using the locally stored LDAP credentials. However, you can restrict such indefinite usage of the stored LDAP credentials and force user re-authentication at regular intervals by setting up this configuration parameter. For example, setting a value of 24 would force user re-authentication (by the LDAP server) every 24 hours.

LDAP CONFIGURATIONS MAXIMUM LIMIT

You can use this parameter to set the maximum number of LDAP configurations to be allowed on sites with multiple LDAP servers/directories. For example, if you set the value as 10, you can add only up to 10 LDAP configurations.

ALLOW LOCAL USER

On sites with LDAP/SAML/SAML+LDAP integrations, site administrators can designate select users that do not have a SAML or LDAP account as local users. Local users can log on to TeamForge using just the TeamForge credentials while bypassing the SAML/LDAP/SAML+LDAP authentication realm. A local user can also change and reset his password.

When you select this **ALLOW LOCAL USER** site setting, the **Create User** and **Edit User Information** pages let site administrators to select the **Local User** check box while creating or editing user accounts.

In addition, site administrators can manage user accounts and passwords in a SAML and/or LDAP enabled site, if this site parameter is enabled.

The pebble-dep.xml file, also known as the Pebble deployment configuration file, contains the data that Pebble needs to interact with the TeamForge site.

A sample pebble-dep.xml file for the REST service type

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE integrated-application
  PUBLIC "-//CollabNet, Inc.//DTD Integrated Application Descriptor 1.0//EN"
  "http://schema.open.collab.net/sfee50/dtd/sf-pluggable-deploy-descriptor_1_0
.dtd">
<integrated-application>
    <name>Pebble Blog</name>
    <baseurl>https://cu064.cloud.sp.collab.net:13001/pebble/index.jsp</baseurl>
    <gourl>https://cu064.cloud.sp.collab.net:13001/pebble/gourl/%p/%o</gourl>
    <endpoint>https://cu064.cloud.sp.collab.net:13001/pebble/services/rest/ctf
api</endpoint>
  <servicetype>REST</servicetype>
</integrated-application>
```

A sample pebble-dep.xml file for the SOAP service type

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE integrated-application
  PUBLIC "-//CollabNet, Inc.//DTD Integrated Application Descriptor 1.0//EN"
  "http://schema.open.collab.net/sfee50/dtd/sf-pluggable-deploy-descriptor_1_0
.dtd">
<integrated-application>
    <name>Pebble Blog</name>
    <baseurl>https://cu177.cloud.maa.collab.net:13001/pebble/index.jsp</baseurl
>
    <gourl>https://cu177.cloud.maa.collab.net:13001/pebble/gourl/%p/%o</gourl>
<endpoint>https://cu177.cloud.maa.collab.net:13001/pebble/services/PebbleInteg
rationService</endpoint>
    <servicetype>SOAP</servicetype>
</integrated-application>
```

The pebble-app.xml file, also known as the Pebble application configuration file, contains the text that the Pebble application displays in the TeamForge user interface.

This is an example of a default (unedited) `pebble-app.xml` file. To create your own integrated application config file, copy this one into a new file and replace the values with the values appropriate for the application you are integrating.

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE integrated-application
  PUBLIC "-//CollabNet, Inc.//DTD Integrated Application Descriptor 1.0//EN"
  "http://schema.open.collab.net/sfee50/dtd/sf-pluggable-application-descripto
r_1_0.dtd">
<integrated-application>
    <name>Pebble Blog</name>
    <description>l10n.application.description</description>
```

```
<permissions>
    <permission dapMappedTo="View">Blog Reader↙permission>
    <permission>Blog Contributer↙permission>
    <permission>Blog Publisher↙permission>
    <permission>Blog Owner↙permission>
↙permissions>
<prefix>PB↙prefix>
<id-pattern>↙id-pattern>
<require-per-project-prefix>true↙require-per-project-prefix>
<require-scm-integration>true↙require-scm-integration>
<is-search-supported>false↙is-search-supported>
<!-- Page components for Integrated apps is not implemented for Alpha -->
<page-component>
  <require-page-component>true↙require-page-component>
  <page-component-details>
    <inputtype>text↙inputtype>
    <resultformat>html↙resultformat>
    <description>l10n.pce.description↙description>
    <title>l10n.pce.title↙title>
  ↙page-component-details>
↙page-component>
<config-parameters>
    <!-- Pebble Configuration Parameters -->
    <param>
        <title>l10n.blogname.title↙title>
        <name>blogName↙name>
        <description>l10n.blogname.description↙description>
        <defaultvalue>My Blog↙defaultvalue>
        <displaytype valuetype="String" maxlength="25">TEXT↙displaytype>
        <editable>false↙editable>
    ↙param>
    <param>
        <title>l10n.blogdescription.title↙title>
        <name>blogDescription↙name>
        <description>l10n.blogdescription.description↙description>
        <defaultvalue>My Awesome Blog↙defaultvalue>
        <displaytype valuetype="String" maxlength="40">TEXT↙displaytype>
        <editable>true↙editable>
    ↙param>
    <param>
        <title>l10n.richtexteditor.title↙title>
        <name>richTextEditorEnabled↙name>
        <description>l10n.richtexteditor.description↙description>
        <defaultvalue>checked↙defaultvalue>
        <displaytype valuetype="String">CHECKBOX↙displaytype>
        <editable>true↙editable>
    ↙param>
    <param>
        <title>l10n.noofrecentblogentries.title↙title>
```

```
            <name>recentBlogEntries</name>
            <description>l10n.noofrecentblogentries.description</description>
            <defaultvalue>3</defaultvalue>
            <displaytype valuetype="String">SELECT</displaytype>
            <option name="3">l10n.three.value</option>
            <option name="5">l10n.five.value</option>
            <option name="7">l10n.seven.value</option>
            <option name="9">l10n.nine.value</option>
            <editable>true</editable>
        </param>
    </config-parameters>
    <bundles>
      <bundle locale="en">
        <key name="l10n.application.description">Pebble Blog App</key>
        <key name="l10n.pce.description">Display Blog Title for Given Date.</ke
y>
        <key name="l10n.pce.title">Enter Blog Date (in yyyy-mm-dd)</key>
        <key name="l10n.blogname.title">Blog Name</key>
        <key name="l10n.blogname.description">Please provide a name for the Bl
og. This appears on all blog pages</key>
        <key name="l10n.blogdescription.title">Blog Description</key>
        <key name="l10n.blogdescription.description">Please provide a descript
ion for the Blog. This appears below blog name on all pages</key>
        <key name="l10n.richtexteditor.title">Rich Text Editor</key>
        <key name="l10n.richtexteditor.description">Enable Rich Text Editor fo
r comments and Blog entries?</key>
        <key name="l10n.noofrecentblogentries.title">Recent Blog Entries</key>
        <key name="l10n.noofrecentblogentries.description">How many recent blo
g entries do you want to see in the home page?</key>
        <key name="l10n.three.value">3</key>
        <key name="l10n.five.value">5</key>
        <key name="l10n.seven.value">7</key>
        <key name="l10n.nine.value">9</key>
      </bundle>
    </bundles>
</integrated-application>
```

This is the sample application-policy block that you can copy into your login-config.xml file to support LDAP authentication.

Notes

Replace the default application-policy block of the login-config.xml file with this code, then make the modifications specified in Set up LDAP integration for the TeamForge Site. Option values that must be modified are highlighted in bold.

- When the username is passed to the login module from TeamForge, it is translated into a DN for lookup on the LDAP server. The DN that is sent to the LDAP server is `<principalDNPrefix><username><principalDNSuffix>`.
- In this example `application-policy` block, the username is stored in the People organizational unit in the `dev.sf.net` domain. This is represented as `,ou=People,dc=dev,dc=sf,dc=net`
- This example contains a single login-module section. If you are authenticating against multiple LDAP servers, include one `login-module` section per LDAP server, with the required option values modified appropriately for each one. If the same username exists in more than one LDAP server, the instance on the first LDAP server will be used.

Sample Code

```
 <application-policy name="SourceForge">
   <authentication>
     <login-module code="org.jboss.security.auth.spi.LdapLoginModule" flag="suff
icient" >
       <module-option name="allowEmptyPasswords">false</module-option>
       <module-option name="principalDNPrefix">uid=</module-option>
       <module-option name="principalDNSuffix">,ou=People,dc=dev,dc=sf,dc=net</m
odule-option>
       <module-option name="java.naming.factory.initial">com.sun.jndi.ldap.Lda
pCtxFactory</module-option>
       <module-option name="java.naming.provider.url">ldap://util.dev.sf.net:3
89/</module-option>
       <module-option name="java.naming.security.authentication">simple</module
-option>
     </login-module>
   </authentication>
 </application-policy>
```

Sample Code for Active Directory Integration

Active Directory is not supported. However, these sample lines in the login-config.xml file may help you make it work for a simple AD setup, without complex directory structures requiring additional search parameters.

Set the values of `java.naming.provider.url`, `principalDNSuffix` and `rolesCtxDN` as appropriate to your site.

For more detailed instructions, see http://www.jboss.org/community/wiki/LdapLoginModule.

```
<login-module code="org.jboss.security.auth.spi.LdapLoginModule" flag="required
" >
```

```
 <module-option name="java.naming.provider.url">ldaps://<server_name>:636/</mod
ule-option>
 <module-option name="allowEmptyPasswords">false</module-option>
 <module-option name="principalDNSuffix">@foo.bar.com</module-option>
 <module-option name="rolesCtxDN">dc=Foo,dc=Bar,dc=Com</module-option>
 <module-option name="matchOnUserDN">true</module-option>
 <module-option name="uidAttributeID">sAMAccountName</module-option>
 <module-option name="roleAttributeID">memberOf</module-option>
 <module-option name="roleAttributeIsDN">true</module-option>
 <module-option name="roleNameAttributeID">name</module-option>
</login-module>
```

Run the add_eventq_to_projects.py script to add EventQ to existing projects.

Usage

Use the following command to run the this script.

```
[RUNTIME_DIR]/scripts/add_eventq_to_projects.py [[--tfuser=<username>] [--tfpasswd=<password>]]
```

Parameters

Parameter	Description
-u | --tfuser	The TeamForge site administrator user name.
-p | --tfpasswd	The password for TeamForge user.
-s | --silent	To prevent printing progress message on standard-out.
-h | --help	To view help information for the script.

The backup-rb-data.py script is used to back up the Review Board application data.

The Review Board application data includes Review Board database and files. If there are any files in the backup directory, the script overwrites these files.

Usage

```
python ./backup-rb-data.py --backupdir={dir}
```

Parameters

The following parameters are available for the `backup-rb-data.py` script.

Parameter	Description
-b | --both	Back up both the database and the filesystem. This is the default option.
-d | --database	Back up the database.
-f | --files	Back up the filesystem.
-h | --help	Provides a list of all available options for this script.

The db.py script can be used to dump and restore a PostgreSQL database.

This script can be used only for the PostgreSQL service. Don’t run this script on a remote database. Execute the script only when the database is up and running.

Usage

Run this script as follows:

```
./db.py --action=<action> --path=<destination directory>
```

Options

Mandatory options:

Option	Description
-a | --action	Action to be performed. You can pass either dump or `restore` as an option.
-f | --path	Path where the database backup file will be created. Must be a directory owned by the postgresql user (usually `/var/lib/pgsql/9.6/`). Can be a new directory.

Other options:

Option	Description
-t | --type	Specifies the type of database (`ctf` or `reporting`).
-h | --help	Print this usage message and exit.

The domain_change_db.py script handles all the steps required to change the domain name in the site database. It does not change anything in the file system.

Changing the domain through any other mechanism may cause problems.

Usage

Execute this script with a command like this:

```
[RUNTIME_DIR]/scripts/domain_change_db.py [--debug] [--dir] --old={domain_name
} --new={domain_name}
```

> **NOTE:** The new domain name must match the value defined for the [PUBLIC_FQDN][siteconfigtokens.html#hostpublic_fqdn] token in the `site-options.conf` file.

Parameters

Parameter	Description

--help	Show command help information
--debug	Include debugging information
--old	Old domain
--new	New domain
--dir	Run domain change in this directory only. You must specify the full path. Use this feature to do a subset of the data directory. This instructs the script to do a recurse in the specified directory looking for the old `domain_name` and replacing it with the new `domain_name`. **NOTE:** Without this option, only HTML, text, and VM files are modified.
--threadlimit	Defines the maximum number of simultaneous threads that can invoked by this program. The default value is 50.

The etl-client.py script allows you to access the Extract, Transform and Load (ETL) scheduler and check the status of the jobs configured. The script also supports triggering jobs manually.

Parameters

The following parameters are available for the `etl-client.py` script:

-s | --status
Prints the status of all the jobs configured in the ETL service.
-a | --status-all
Prints the status of incremental and historical jobs configured in the ETL service.
-v| --verbose
Chronicles the process of requested operation a bit more.
-r| --run=
Triggers a job manually for a given job.

Data Collection (ETL) Jobs Supported by the ETL Service

While some ETL jobs are scheduled to run automatically, some must be triggered manually. The following table lists the available ETL jobs in TeamForge.

Job Category	Job Name	Description
History Data Collection Historical data collection jobs must be triggered manually. As a best practice, these jobs are run as part of post migration activities. Refer to the "Related Links" for more information.	SCMCommitInitialJob	Collects the historical commit data from TeamForge.
	TrackerInitialJob	Collects the historical data of artifacts from TeamForge.

	LoadFlexFields **Note:** This job must be executed if and only if you are upgrading from TeamForge 8.0 and earlier versions.	Collects the historical data of custom-defined artifacts from TeamForge.
Incremental Data Collection Incremental data collection jobs collect data that are added or modified incrementally on an ETL run-to-run basis. These jobs are scheduled to run automatically on a regular basis.	SCMCommitActivityJob	Collects the SCM commit data incrementally on an ETL run-to-run basis.
	TrackerIncrementalJob	Collects the tracker artifacts data incrementally on an ETL run-to-run basis.
	UserLoginActivityJob	Collects the user logon activity data incrementally on an ETL run-to-run basis.
Imported Data Collection (Simbel) TeamForge supports bulk data import through Simbel. This job collects Simbel-imported data. This job must be triggered manually post data import into TeamForge.	SimbelImportJob	Collects all Simbel-imported data such as the user logon activity, SCM commit and tracker artifacts data.

The password_util.sh script is used to get the encrypted or decrypted password value for the user scmviewer.

Usage

- To encrypt:
  ```
  sudo /opt/collabnet/teamforge/runtime/scripts/password_util.sh -encrypt 'teamforge'
  [root@xx scripts]# ./password_util.sh -encrypt 'teamforge'
  Input String:teamforge
  Encrypted password:VBxJJvzbXb5tNx2SxR26egA==
  ```
- To decrypt:
  ```
  sudo /opt/collabnet/teamforge/runtime/scripts/password_util.sh -decrypt 'VBxJJvzbXb5tNx2SxR26egA=='
  [root@xx scripts]# ./password_util.sh -decrypt 'VBxJJvzbXb5tNx2SxR26egA=='
  Input String:VBxJJvzbXb5tNx2SxR26egA==
  Decrypted password:teamforge
  ```

The psql-wrapper script is used to connect to the TeamForge application.

Usage

```
sudo [RUNTIME_DIR]/scripts/psql-wrapper
```

Comments

- Run this script as a sudo user.
- Run this script with the postgres backend.
- You have full write access to the database for executing queries.

> **NOTE:** This script is not supported in the Oracle backend.

The psql-reporting-wrapper script is used to connect to the datamart.

Usage

```
sudo [RUNTIME_DIR]/scripts/psql-reporting-wrapper
```

Comments

- Run this script as a sudo user.
- Run this script with the postgres backend.
- You have full write access to the database for executing queries.

> **NOTE:** This script is not supported in the Oracle backend.

The restore-data.py script restores the compressed data from the named source directory and deletes any existing data. By default, the TeamForge and the reporting database are backed up to the destination directory. If reporting is disabled, only the TeamForge database is backed up.

Overview

`restore-data.py` finds and unpacks these data resources:

- Subversion repositories
- CVS repositories (if any)
- The data directory (/var)
- The SourceForge database

Usage

Run this script as follows:

```
./restore-data.py --source=<directory name>
```

where <directory-name> is the directory to which you backed up the data with the `backup-data.py` script.

Location

```
/opt/collabnet/teamforge/runtime/scripts/
```

Options

--source
    Directory where the compressed copy of the site's data is available.
--help | -h
    Print this usage message and exit.

The restore-rb-data.py script is used to restore the Review Board application data from the backup directory.

This script removes the existing Review Board application data present in the system and restores data from the backup directory.

Usage

```
python ./restore-rb-data.py --backupdir={dir}
```

Options

The following options are available for the `restore-rb-data.py` script:

Option	Description
-b | –both	Restore both the database and the filesystem. This is the default option.
-d | –database	Restore the database.
-f | –files	Restore the filesystem.
-h | –help	Provides a list of all available options for this script.

The SearchReindex.py script allows you to reindex the entire TeamForge data.

Overview

You can use this script to reindex the entire TeamForge data or you can choose to reindex the subset of data types. Usage

Run this script as follows:

```
./SearchReindex.py --<component name>
```

Example

- To perform a search reindex for the tracker, run this command:
  `./SearchReindex.py --trackers-only`
- To perform a search reindex for the wiki, run this command:
  `./SearchReindex.py --wiki-only`
- To perform a search reindex for documents run this command:
  `./SearchReindex.py --documents-only`

Options

--single-item itemId, | -i
: Schedules a re-index for just the given item. If the item id is for a project the scheduling results in the server re-indexing all of the project data.

---force-index | -f
: Force indexing (doesn't check if item is searchable already).

--artifacts only | -a
: Reindex all artifacts on the site that are currently not searchable or all artifacts if option f is selected.

--documents only | -d
: Reindex all documents on the site that are currently not searchable or all artifacts if option f is selected.

---posts only
: Reindex all posts on the site that are currently not searchable or all artifacts if option f is selected.

---trackers only
: Reindex all trackers on the site.

---document_folders-only
: Reindex all document folders on the site.

---topics-only
: Reindex all topics.

---forums-only
: Reindex all forums on the site.

---news-only
: Reindex all news.

---project_pages-only
: Reindex all project pages.

---packages
Reindex all packages.

---commits-only
Reindex all commits.

---frs_files-only
Reindex all frs files.

---releases-only
Reindex all releases.

---wikis-only
Reindex all wikis.

--project-id projectID | -p
Limit the re-indexing to data for single project when re-indexing only artifacts and or documents.

--verify | -x
Searches for each item that is scheduled for re-indexing. There is a one minute wait limit for each item to be re-indexed by the server.

--dryrun | -r
Executes all the steps for scheduling a re-index without actually sending any re-index requests to the server. This provides a list of items that need re-indexing.

output-file filePath, | -o
Prints the output for the given file.

--verbose | -v
Chronicles the process of scheduling the re-index a bit more.

Run the sync-orchestrate.py script to synchronize a TeamForge project with EventQ by triggering the world-state message for the given project.

Usage

Use the following command to run the this script.

```
[RUNTIME_DIR]/scripts/sync-orchestrate.py --projectId=<proj1234> [[--tfuser=<username>] [--tfpasswd=<password>]]
```

Parameters

Paramater	Description
--projectId	The TeamForge project ID.

-u | --tfuser	The TeamForge Site Administrator user name.
-p | --tfpasswd	The TeamForge Site Administrator password.
-h | --help	To view help information for the script.

Use the teamforge.py script to deploy and undeploy services, start and stop services, verify the status of services, verify the application environment, bootstrap or migrate data, back up and restore data and do much more.

Overview

Use the `teamforge` script wherever applicable as it subsumes the functions of the following legacy TeamForge scripts:

- `bootstrap-data.sh`
- `bootstrap-data.py`
- `bootstrap-reporting-data.sh`
- `bootstrap-reporting-data.py`
- `create-runtime.py`
- `collabnet`
- `migrate.py`
- `post-install.py`
- `snapshot.py`

✓ Starting from TeamForge 17.8, `/opt/collabnet/teamforge/bin/teamforge` has been linked to `/usr/bin`. You can simply run the `teamforge` command from any path.

Run the `teamforge` script as follows:

```
teamforge [command] [-s serviceName] [other parameters]
```

For example, the following command displays the status of all the services:

```
teamforge status
```

TeamForge components and services

Components

TeamForge comprises of a set of components such as ctfcore, subversion, james, and so on. Some components are always required to be installed, while others are optional. By spreading these components over multiple RPMs, we make sure that users do not have to install everything all the time. Though this is valuable, RPMs alone prove insufficient to manage the components and their inter-dependencies.

- Some components do not have a physical representation but are configuration-only.
- RPM dependencies are restricted to the local machine only; however, in a distributed installation, dependencies between components must be tracked across servers.

In addition to the physical componentization (using RPMs), there is also a need for a logical componentization of TeamForge.

Services

Services represent a logical component of TeamForge. Such a logical component may either be a feature, which the user explicitly opts to install (for example, CVS) or a technical component (for example, Apache and Logrotate), which other services depend on. Services come with additional metadata which makes it possible to track and manage dependencies to a more fine-grained level.

- Deployment dependencies specify which other services need to be deployed locally.
- Provided Endpoints specify which network endpoints the service offers.
- Required Endpoints specify which network endpoints the service depends on.

As such services are more fine-grained than RPMs and it is common to have a single RPM containing multiple services.

Service life cycle

A service can be in one of the following states:

- `Uninstalled`: A service is uninstalled if the RPM that contains it is not installed. Uninstalled services do not exist as far as TeamForge is concerned.
- `Undeployed`\`: The RPM containing the service is installed, but the service has not been deployed yet. Deployment is also referred to as “creating the runtime”, but is specific to one service.
- `Dependencies unavailable`: The service itself might be available, but at least one of its deployment dependencies is not in “Available” state.
- `Available`: Service is fully functional.

Services that manage data have the following additional states:

- `Not bootstrapped`: Data structures have not been initialized yet.
- `Not migrated`: Data structures are initialized, but data needs to be migrated.

Services that have a daemon have the following additional states:

- `Dependencies unavailable`: The service itself might be available, but at least one of its deployment dependencies is not in “Available” state.
- `Ports Blocked`: The service is impeded from starting up because at least one of the ports it needs is in use by a different process.
- `Stopped`: Service is an auto start-type service, yet is stopped.
- `Inactive`: Service is a demand start-type service and is stopped.
- `Starting`: Service is in the process of starting up.
- `Available`: Service is running and healthy according to its health check.
- `Unhealthy`: Service is running but unhealthy according to its health check.
- `Dead`: Service is supposed to be running, but the process disappeared.
- `Doomed`: Service is technically running, but it will never work properly because some part of it failed to initialize properly.
- `Stopping`: Service is in the process of stopping.

Parameters

teamforge.py script accepts the following command line parameters:

[-s | --service]

Use the *-s* parameter to selectively act on a specific service or component such as selectively start, stop, bootstrap, backup and restore a specific service. For example, the following command gets you the status of Jboss:

```
/opt/collabnet/teamforge/bin/teamforge status -s jboss
```

[-f | --site_options_file]

Use the -f parameter to pass the site-options.conf file's path as a command line parameter (default is /opt/collabnet/teamforge/etc/site-options.conf).

[--skip-verification]

Pass this parameter if you want to skip environment verification.

[-y | --yes]

Pass this parameter if you want to skip confirmation prompts.

[-p | --path]

The path to the backup or restore directory. Used with the `teamforge backup` and `teamforge restore` commands.</dd> </dl>

Commands

The `teamforge` script can perform the following actions:

status

Show status of all services. Use the `-s` parameter to know the status of a specific service.

bootstrap

Bootstrap data (re-create data structures). Use the `-s` parameter to selectively bootstrap a specific component. Suppose, you did not have SVN on your TeamForge site and if you add SVN while upgrading to TeamForge 16.10 or later. You can now selectively bootstrap SVN alone.

deploy

Deploy service(s).

migrate

Migrate data to latest schema.

provision

Provision/reprovision the server. The `provision` command performs tasks such as creating the runtime, starting and stopping services, bootstrapping (fresh install) or migrating (upgrade) data, deploying services, setting file permissions, setting SELinux permissions, initializing services and so on.

undeploy

Undeploy service(s).

start

Start service(s).

stop

Stop service(s).

kill

Terminates service(s) forcefully.

verify

Verify environment.

show-endpoints

Show endpoints

show-dependencies

Show deployment dependencies

reinitialize

Reinitializes all the TeamForge services

apply-permissions

Applies the TeamForge file system permissions when you restore the TeamForge data in a new server while upgraing to a latest TeamForge version.

snapshot

Dumps relevant diagnostic information to the console (stdout) for each deployed service.

info

Displays a summary of TeamForge configuration.

check-data-integrity

Verifies the integrity of the data defined in the service manifest.

update-data-integrity

Updates the calculated checksums for the data defined in the manifest.

await-dependencies

Waits for dependencies to become available.

apply-selinux

Applies selinux policies.

unload-selinux

Unloads selinux policies.

logs

Tails log files.

backup

Back up TeamForge data. The **-p** parameter is mandatory. For more information, see Back up and Restore TeamForge Data Using the teamforge.py Script.

restore

Restore TeamForge data. The **-p** parameter is mandatory. For more information, see Back up and Restore TeamForge Data Using the teamforge.py Script.

Back up and Restore TeamForge Data Using the teamforge.py Script

Use the `teamforge.py` script's `backup` and `restore` commands to back up and restore TeamForge data.

Back up and Restore TeamForge

If you are upgrading to a latest TeamForge version, it is still recommended to follow the usual backup and restore procedure. Use the `teamforge.py` script's `backup` and `restore` commands if you want to back up a particular service such as CVS and restore it on a new server (when you intend to move a specific service from one server to another, typically to move the service to a dedicated server of its own).

If you are backing up TeamForge as a whole, you must stop all the TeamForge services but PostgreSQL before running the `teamforge backup` and `teamforge restore` commands.

1. To back up TeamForge data:
   ```
   teamforge stop
   teamforge start -s postgres
   teamforge backup -p <path to the directory where the data is backed up
   >
   ```
2. Compress the backed up data and copy it to the target server where you want to restore the data.
3. To restore TeamForge data:
   ```
   teamforge stop
   teamforge start -s postgres
   teamforge restore -p <path to the directory where you have the data to
    be restored>
   ```
4. Provision services.
   ```
   teamforge provision
   ```

Back up and Restore a Specific Service

1. To back up a specific service (such as CVS):

```
teamforge backup -s <serviceName> -p <path to the directory where the data is backed up>
```

For example:

```
teamforge backup -s cvs -p /tmp/cvsbackup
```

2. Compress the backed up data and copy it to the target server where you want to restore the data.
3. To restore a specific service’s data:

   ```
   teamforge restore -s <serviceName> -p <path to the directory where you
    have the data to be restored>
   ```

   For example:

   ```
   teamforge restore -s cvs -p /tmp/cvsbackup
   ```
4. Provision services.

   ```
   teamforge provision
   ```

Did You Back up symlinked Directories?

Do this if and only if you had backed up and restored symlinked directories.

1. Move the `svnroot`, `cvsroot` and `sf-svnroot` from `/opt/collabnet/teamforge/var/scm` to the root directory.

   ```
   cd /opt/collabnet/teamforge/var/scm
   mv svnroot cvsroot sf-svnroot /
   ```
2. Create symlinks to the root directory.

   ```
   ln -s /sf-svnroot .
   ln -s /svnroot .
   ln -s /cvsroot .
   ```
3. Provision TeamForge.

   ```
   teamforge provision -y
   ```
4. Apply the TeamForge file system permissions.

   ```
   teamforge apply-permissions
   ```

Logging

The `teamforge` script writes entries to `/opt/collabnet/teamforge/log/runtime/runtime.log` file.

System administrators can use logs to debug problems and ensure that the application is performing to expectations.

Read Your Site's Logs

Inspecting the system logs for your TeamForge site may product useful information for solving problems.

1. Go to **My Workspace > Admin**.
2. On the site administration navigation bar, click **SYSTEM TOOLS**.
3. On the **System Tools** menu, click **System Logs**. All the logs the server has written are listed.
4. On the **System Logs** page, click the log file you are interested in.

Change the Location of Log Files

To change where log files are written to, set the value of the LOG_DIR token with the location where you want the log files to be written and provision TeamForge.

Configure Your Site's Log Level

Change your site's log level in the **Configure Logging** page.

The Configure Logging page lets you change the application server (JBoss) log level. Changing the application server log level affects only the `server.log` and `vamessages.log` files.

1. Go to **My Workspace > Admin**.
2. On the site administration navigation bar, click **SYSTEM TOOLS**.
3. On the **System Tools** menu, click **Configure Logging**.
4. On the **Configure Logging** page, select either **INFO** or **DEBUG** from the **LOG LEVEL** drop-down list.
   - The default log level is INFO.
   - JBoss restart is not required after changing the log level using the **Configure Logging** page.

   > **WARNING:** The `server.log` and `vamessages.log` files grow in size if you change the log level from `INFO` to `DEBUG`.

5. Click **Save**.

Raise the Logging Level of Long-running Database Requests

For easier troubleshooting, you can dictate that certain database requests that run for a specific duration get logged in a handy central log file.

For example, database requests that run longer than 10 seconds are likely candidates for troubleshooting. You can have such requests automatically logged in the `vamessages.log` file for your inspection. The exact length of time after which a request becomes problematic depends on your environment.

How it works:

- All database queries are logged at `DEBUG` level by default.
- By default, the vamessages.log file is configured to include all events logged at the `INFO` level or higher.
- Database queries that run over a configurable time limit are logged at `INFO` rather than `DEBUG`, which causes them to appear in `vamessages.log`.

For more information, see LOG_QUERY_TIME_THRESHOLD.

JBoss Logs

The JBoss application server writes several different logs under the `<SOURCEFORGE_INSTALL_DIR>/log` directory.

Log File	Description
boot.log	Lgs the JBoss startup and shut down notifications. This log is overwritten each time JBoss is (re)started.
localhost_access	The Records access to the application from a remote host, similar to the Apache `access_log`. This log is rotated each day, and the files have a date stamp appended to their name, such as `localhost_access2004-11-26.log`.
server.log	Logs all the activities of the application server, including any exceptions. This log is the best place to begin debugging TeamForge server error exception ids (exid).
session-info.log	Records when new sessions are created. This log is overwritten each time JBoss is (re)started.
vamessages.log	Records TeamForge-specific actions, including some SQL queries that are sent to the backend database. This log is rotated each time it reaches 100MB in size. When rotated the older files have a number appended to the end, such as `vamessages.log.1` and `vamessages.log.2`.

Oracle logging

The most important Oracle log is the `alert` log, which is found in `$ORACLE_HOME/admin/$SID/bdump/alert_$SID.log`.

An Oracle database performs logging on a wide array of functionality. The majority of the logs that are generated are stored under `$ORACLE_HOME/admin/$SID/`. Many logs are stored under this directory hierarchy, but alert is the most important. This log records all database activity, including serious problems.

The `alert` log is not rotated or overwritten, and can become quite large over time, especially on an active database.

Additional logs are created under the same directory hierarchy, for specific incidents. If a problem is recorded in the alert log, the other logs should be inspected for additional details.

For more information, as well as support in the maintenance of an Oracle database, contact Oracle Support or Oracle's Metalink site.

SCM (CVS and Subversion) Logs

Software configuration management (SCM) servers generate several logs from the TeamForge; however, in the interest of completeness they are all documented here.

Log File	Description
catalina.out	This log contains information on the startup and runtime activities of the Tomcat server. This log is not rotated, nor is it overwritten, and is appended continuously over the lifetime of the server.
localhost_log	This log contains a record of CVS or Subversion browsing URL construction. When a user attempts to browse a CVS or Subversion repository in his or her web browser, the URL construction process is documented in this log. This log is rotated for each date that there is activity.
localhost_admin_log	This log contains a record of the initial startup and deployment of the managed integration server. A new date stamped log is generated each time the integration server is started.
vaexternalintegration.log	This log contains information on the operations that are being executed by the managed integration server. This log is stored in `/log`.

Email Logs

Both the TeamForge email and search backends are managed from a parent daemon known as Phoenix. If the mail backend is not operating properly, the first troubleshooting step is to check the `phoenix.log` to see if it encountered difficulties starting up.

Overview

The Phoenix daemon logs its activities to the phoenix.log file, which is stored under `SOURCEFORGE_INSTALL_DIR/james/james-2.2.0/logs`. This log is overwritten each time Phoenix is (re)started. Phoenix is run as part of the TeamForge standalone server init script.

TeamForge email is handled by the JAMES server. JAMES logs all of its activities under `SOURCEFORGE_INSTALL_DIR/james/james-2.2.0/apps/james/logs`. A new log is created for each date when there is activity, and additional logs are created if james is restarted on a date when there is activity. The date is embedded in the log name (such as `james-2005-04-28-01-00.log`).

Active Logs

Sixteen different logs are created by james for different components of its functionality. This topic describes only the ones that are used actively by TeamForge.

Log File	Description
james-$date.log	The James log records the overall mail handling behavior of the James server.
mailet-$date.log	The mailet log records how each piece of email is handled. If there is a mail delivery problem, this log is the best place to begin investigation.
mailstore-$date.log	The mailstore log records the behavior of mail spools, and the storage of mail. This log should normally not contain errors unless James is unable to write or read mail to or from the file system.
smtpserver-$date.log	The smtpserver log records all inbound mail handling results. If email to discussion forums is not posting, or is getting rejected, this log would be the best place to begin investigation.
spoolmanager-$date.log	The spoolmanager log records the processing of mail spools. This log could be of value in troubleshooting mail delivery or handling problems.

Search Logs

Both the TeamForge search and email backends are managed from a parent daemon known as Phoenix. If the search backend is not operating properly, the first troubleshooting step is to check the `phoenix.log` file to see if it encountered difficulties starting up.

The Phoenix daemon logs its activities to the `phoenix.log` file, which is stored under `SOURCEFORGE_INSTALL_DIR/james/james-2.2.0/logs`. This log is overwritten each time Phoenix is (re)started.

Phoenix is run as part of the TeamForge standalone server init script.

Once started successfully, the search server waits for new content to be indexed or searches to be performed. The search server logs its activities under `SOURCEFORGE_INSTALL_DIR/james/james-2.2.0/apps/search/logs`. The logs that are created are all named default with the date stamp appended to them (such as `default-20041126.log`). A new log is created for each date that there is indexing activity.

If the search server is not running, or expected search results are not being provided, the default log is the best place to investigate further.

Project Build Library (PBL) Audit Log

You can use this page to view the complete list of actions performed in the Project Build Library.

Contents

Information about the following types of actions is displayed in this page:

- Change a File Description
- Create a Directory
- Delete a File or Directory
- Download a File
- Move a File or Directory
- Upload a File

> **NOTE:** The value displayed in the `Event` field is the value passed in the `--comment` parameter from the **Project Build Library** client.

Getting There

On the project home page, click **Build Library** in the left navigation bar and select the **Audit Log** tab.

Access

This page is accessible for all users who have at least the view permission for the project.

Profile Audit Log

Use this page to view the complete list of actions performed on a profile.

Getting There

On the Profile Library page, click the Audit Log tab.

Access

This page is accessible for all users who have at least the view permission for the project to which the profile is allowed.

Example

x When a user updates any of the profile fields on the **Profile Admin** page, the following details are displayed in this page:

- The old value for the field.
- The new value for the field.
- The name of user who updated the field.
- The time when the change occurred.

User Audit Log

You can use this page to view the list of actions performed by the user in the **TeamForge Lab Management** system.

For example, when a user logs into the web interface of the TeamForge Lab Management system, the event is displayed in this page.

Getting There

On the **Administration** tab, click **User Audit Logs** in the left navigation bar.

Access

This page is accessible to all users who have at least the Domain Administrator role.

Host Audit Log

You can use this page to view the complete list of actions performed on a host.

Getting There

On the TeamForge Lab Management Host home page, click the **Audit Log** tab.

Access

This page is accessible for all users who have at least the view permission for the project to which the host is assigned.

Example

When the IP address for the host is changed, the following details are displayed in this page:

- The old IP address.
- The new IP address.
- The name of user who changed the IP address.
- The time when the change occurred.

Project Audit Log

The Project audit log page shows the complete list of changes applied to a project.

Getting There

On the TeamForge Lab Management Project home page, click **Audit Logs** in the left navigation bar.

Access

This page is accessible for all users who have at least the view permission for the project.

Example

When a profile is added to the list of buildable profiles for this project, the following information appears on this page:

- The action that was taken.
- The user who performed the change.
- The time when this change occurred.

etl.log

This file contains information from extract-transform-load runs, including data transformation warnings and errors.

> **NOTE:** Transformation errors do not constitute a failed ETL run. For example, if a corrupt row of data in one of the source tables causes transformation errors, this is treated as a `skipped record` and gets logged.

Set the logging level appropriately to enable logging in vamessages.log.

1. Edit `$RUNTIME/jboss/bin/jboss-cli.sh` to enable logging in `vamessages.log`.

   > **NOTE:** You need not restart the site for JBoss to pick up these changes.

2. To change log levels, as a root user, perform the following:

   1. To enable debug logging, run the following command:

      ```
      /subsystem=logging/root-logger=ROOT:change-root-log-level(level=DEBUG)
      ```

   2. To disable debug logging, run the following command:

      ```
      /subsystem=logging/root-logger=ROOT:change-root-log-level(level=INFO)
      ```

3. To change log levels using your LDAP credentials, perform the following:

   1. To enable debug logging, run the following command:

      ```
      /subsystem=logging/logger=com.vasoftware:write-attribute(name="level",
       value="DEBUG")
      ```

   2. To disable debug logging, run the following command:

      ```
      /subsystem=logging/logger=com.vasoftware:write-attribute(name="level",
       value="INFO")
      ```

   3. To enable trace logging for LDAP, run the following command:

      ```
      /subsystem=logging/logger=org.jboss.security.auth.spi.LdapExtLoginModu
      le:add(level=TRACE,handlers=["VAFILE"])
      ```

   4. To disable trace logging for LDAP, run the following command:

      ```
      /subsystem=logging/logger=org.jboss.security.auth.spi.LdapExtLoginModu
      le:remove()
      ```

      > **TIP:** The LDAP debug output will be very limited unless you add <module-option name=`throwValidateErrors` value=`true`></>to the entry for the corresponding log-in module.

- Change in log levels will not need any site restart and these changes will not survive the JBOSS restart.
- Changes made using CLI will survive a restart but not a runtime recreation.

> **NOTE:** Changes are immediately saved to `runtime/jboss/standalone/configuration/standalone-full`.xml and the file, `standalone-full.xml` is recreated every time the TeamForge runtime is rebuilt.

- To retain the configurations after the site restart, edit `$RUNTIME/jboss/standalone/configuration/standalone-full.xml`.
- To make the configurations survive the recreate runtime, edit `$SITE_DIR/dist/conf-snippets/jboss/standalone-full.xml.d/20-standalone-full.xml.py`.

These are some of the frequently asked questions on TeamForge concepts and terms.

What is an Association?

TeamForge allows users to easily associate, or link together, any objects in the system to simplify knowledge sharing and provide traceability throughout the lifecycle.

For example, a discussion post regarding a customer problem could be linked to a document specifying the feature requirement. An issue might be created to track the defect, the source code commits that fix the issue, and the release that contains the fix. All of these records can be linked together with an association. When any item is associated with another, the link appears on the item's *ASSOCIATIONS* tab.

Associations enable development organizations to improve information sharing, capture institutional knowledge, and simplify regulatory compliance.

> **NOTE:** When an association is added to or removed from TeamForge objects such as tracker artifacts, tasks, documents, discussions, and file releases, a notification mail is sent to users monitoring these objects. An option is provided at site level and user level to make sure whether the notification mail has to be sent or not. For more information on this, see Configure your Site's Settings.

What is the Look Project?

The **look** project contains special files that can override your site's default appearance and content, such as the default icons, fonts, colors, and labels.

Unlike most projects, the look project has no members. It is only visible to users with site administration permission. Its only purpose is to control your site's look and feel, including such things as fonts, background colors, icons, and the wording of the onscreen labels that appear throughout your site.

Any project on your TeamForge site can have one or more Subversion repositories associated with it. The **look** project has just one Subversion repository. That repository is named **branding**.

When a user requests a page from your site, TeamForge checks the branding repository to see if any files there specify custom fonts, colors or text strings. If such specifications are found, TeamForge displays the page according to those specifications. If not, the page displays according to the default design.

Having your custom look-and-feel specifications in a Subversion repository enables you to roll back changes, track contributions, and use all the other features of a source code versioning system.

What is a Patch?

A patch is a package of code that fixes or adds to the functionality of a CollabNet product. Patches are also known as "component upgrades."

Things to Know About Patches

- Patches are cumulative. You don’t need to apply multiple patches sequentially to get to the desired patch level. You can move up (or down) one or more patch levels with a single operation.
- The Level option (-l) allows you to downgrade or upgrade to any patch level (within the maximum available in the cumulative patch).
- The Rollback option (-r) allows you to revert the site to the previous patch level it was at, before the current patch was applied.
- The Uninstall option (-u) allows you to downgrade the patch level on the site by one.
- When a patch installation fails you can use the Force option (-F) to proceed, without manually uninstalling previous patches.
- The system displays a summary of what happens during the patch installation.
- Before proceeding with the patch installation, you can use the “dry run” mode (-t option) to see the summary of actions that will be performed during the installation.

Best Practices

Before applying a patch, note the following principles.

- The upgrade scripts are usable only with an existing installation.
- No data migration will occur if any changes have been made to the database schema.
- You must use the `sudo` command or have an account that is equivalent to root in order to complete a patch installation successfully.

> **IMPORTANT:** Before installing a patch, verify that it has been fully tested and qualified.

What are Planning, Task, and Kanban Boards?

When you have set up your planning folders and teams, you have four views available to work with them: **List**, **Plan**, **Task**, and **Kanban**.

TeamForge user roles and permissions that are in place for planning folders apply to all the four views.

Planning Board

The Planning Board is an important tool for your TeamForge project's Agile planning activities. It enables you to plan and monitor the features that are required in each sprint (or iteration) and assign them from the product backlog to specific sprints. The planning board view complements the list view. While the latter offers you capabilities to accomplish various actions such as create, edit, and delete artifacts, planning folders and teams, the former offers product owners (or similar users) the ability to view, rank and move artifacts across the three planning folders (swimlanes) in a physical board-like user interface. In the Planning Board, planning folder are represented as swimlanes. In each swimlane, the tracker artifacts for the selected planning folders are represented as cards. You can also have a team's view of artifacts (backlogs and tasks), which is a swimlane representation of artifact cards for the selected team in the selected planning folder.

Task Board

The Task Board is an important tool in the Agile process. It helps the team to focus on the work at hand in the current sprint and feed progress data back into the system. Unlike the list view and planning board view, which can be used for agile project planning, the task board view is for tracking tasks in a sprint. TeamForge project administrators can configure the Task Board. Once that's done, project members can use the Task Board to break down the stories into tasks and then progress the tasks, and the store, towards completion.

The Task Board can have at least two and up to seven swimlanes depending on how your project administrator configures it. Every swimlane in the Task Board represents a task status. Once configured, team members can use the Task Board to view tasks in a selected planning folder or filter tasks for a specific team within the selected planning folder, add new tasks for a backlog item (epic, story, etc.) and move tasks from one swimlane to the other as tasks progress from one status to the other.

Kanban Board

The Kanban Board is an agile project management tool, which gives you a snapshot of the statuses of work items within a planning folder, how your project teams are placed in terms of work distribution and directs you to re-distribute the tasks to ensure optimal resource utilization.

Kanban Board uses the kanban method for project implementation. Kanban is built on two basic concepts: value stream mapping and WIP (work-in-progress) limits.

- **Value stream map** : A value stream, as defined in the APICS (American Production and Inventory Control Society) dictionary, comprises "the processes of creating, producing, and delivering a good or service to the market". In software development, a value stream map is an end-to-end mapping of the flow of activities (tasks/work items) from one state to another, starting from conceptualization to delivering the product to the customer.

- **WIP limits**: These are constraints (minimum and maximum) applied on each point or state (Planning, In Progress, etc.,) in the value stream to ensure optimal WIP. This defines the minimum and maximum artifact count that ought to be present in each state so that if these constraints are violated, Kanban Board flags the issue for you to fix the bottleneck.

For example, let us assume that you want to view the status of the work items for a planning folder called ‘Iteration 1’ and assess the distribution of work between teams, Team A and Team B within this planning folder. You want to see a maximum of 8 artifacts in the ‘In Progress’ state and you configure your Kanban Board accordingly. When you go to your Kanban Board and view the artifacts for the selected planning folder, you see a constraint violation because there are 14 artifacts in this status. In addition, when you see the work split-up between the teams, Team A has 14 artifacts whereas Team B has none. This is clearly an indication that not only is the overall count of artifacts more than the maximum constraint, but even between the teams, Team A is overloaded and Team B is underutilized; this calls for re-distribution of work items to avoid any delay in the development process.

You can configure the Kanban Board based on your project activities and you can configure as many boards as you require.

List, Plan, Task, and Kanban Views

You can click the LIST, PLAN, TASK and KANBAN buttons to toggle between these views.

When you make a planning folder or team selection in the respective list view, (either from the planning folder/team tree or through the Jump to ID box), the selection is retained when you move to the task board or kanban board view. Similarly, when you select a planning folder or team in the task board or kanban board view, the selection is retained as well when you move to the list view. These selections are retained across browser sessions as well.

> **NOTE:** An exception to this case is when you select the root planning folder or team (‘Project Teams’) in the list view, this selection is not retained when you move to the task or kanban board view; instead, the selection of a specific planning folder or team you had made prior to this one is retained.

In the Planning Board, multiple planning folder and team selections are retained when you navigate to other pages in TeamForge and return to your planning board.

What is a project dashboard page and what it consists of?

The **Project Dashboard** offers a centralized view into all development projects managed in TeamForge.

Overview

The **Project Dashboard** provides managers with an at-a-glance overview of the status of each of their projects. It provides summary information on the number and status of the tasks and tracker artifacts in each project, and calculates project overrun and underrun statistics.

The **Project Dashboard** also provides overview information such as project start and end dates and project ranking.

You can see the **Project Dashboard** if you have both the View Tracker and View Task permissions for one or more projects. Only those projects for which you have both the View Tracker and View Task permissions appear on your **Project Dashboard**.

Getting there

In the TeamForge navigation bar, click **My page > Dashboard**.

You can view the **Project Dashboard** if you have both the View Tracker and View Task permissions for one or more projects. Only those projects for which you have both the View Tracker and View Task permissions are displayed on your **Project Dashboard**.

Contents

For each project, the **Project Dashboard** displays the following information:

- **Project Activity ranking** - The activity of the project in relation to all other TeamForge projects.
- **Start Date** or **End Date** - The start and end date of the project, based on the start and end dates of all project tasks.
- **Task Status** - The status of the project, based on the “rolled-up” status of all project tasks and task folders. You can configure the “roll up” criteria for each project from the project’s **Task Manager Settings** page.
- **Status History** - The history of the project’s “rolled-up” status color. These figures are calculated in real time, but do not calculate time that the project’s status was **Not Started** or **Completed**.

  Click the status bar to go to the project’s **Task Summary** page.
- **Task and Tracker Effort** - The project’s current overrun or underrun, based on the difference between estimated and actual effort spent on project tasks and tracker artifacts.

  Only completed and closed tasks and tracker artifacts, with values in the estimated and actual effort fields, contribute to the overrun or underrun calculations.

- **Tracker Status** - The number of open tracker artifacts in the project, per priority value. The number of open tracker artifacts is indicated in parentheses.

Click the status bar to go to the project's **Tracker Summary** page.

What is a project template?

Project templates enable you to capture and re-use the structure and content of existing projects, including project pages, custom tracker fields, and work flow definitions, to speed new project creation and standardize lifecycle processes.

Project templates allow you to create and configure new projects quickly, enforce organizational standards, and facilitate process improvement.

To create a new project, you can use any project template created by any project administrator.

> **NOTE:** You can also create a new project without using a template.

The template name and description appear on the Templates tab of the **Projects** list, accessible from the navigation bar on your **My Workspace** page. If your site administrator has made it possible to preview the contents of project templates, click the name of a template to see what's in it.

A template can include the structure of the original project without any of the content, or it can include both the structure and the content of the original project.

- "Structure" means the folders and sub-folders in the original project.
- "Project content" means any work items you or any other project member have created as part of the project. For example, when you create a tracker artifact to manage a piece of work, that tracker artifact is part of your project's content. Any documents you upload to the project and any wiki pages you create are part of the project's content.

What is in a project template?

When you create a template from an existing project, each project tool contributes its own structure to the template, and its content if you want it.

For each tool, you can include or omit the actual content that was created with that tool.

For example, suppose you have a project for which you have created some tracker artifacts, and these artifacts have proved useful to members of the project.

- You can include those artifacts in your template, so that people who create a new project from that template will have access to the same artifacts that you developed for your original project.
- You can leave those artifacts out of the template and let future users create new artifacts to fit their own needs.

Imagine that you have documented company-wide process standards on wiki pages in an existing project.

- You can include those wiki pages in a project template, so that the manager of a project created from that template won't have to go find those pages and copy them into the new project.
- Or you can leave the wiki tool empty and let the new project's users create new wiki content for themselves.

> **NOTE:** You can choose not to include content from the original project, but you can't choose not to include a project tool. Every project template you create must include all the project tools, even if the project from which you created the template does not use all the tools. Every project you create from a template will include all the tools.

Tool	Always in Template	Can Be in Template
Tracker	You can have any number of separate trackers in a project. When you create a project template from that project, all the trackers that exist in the original project are copied into the project template. Any default column configurations, select field dependencies or text field validation rules are also included.	You can include the artifacts that were created in the original project, and any parent-child dependencies among them. If users of the original project have shared saved searches, these can be included too.
Planning folders		You can include all the planning folders in the original project, or none of them.
Documents	When you create a project template from an existing project, all the document folders that were created in the original project are copied into the project template.	If users have created documents and attachments in the original project, you can choose to include these documents in the project template.
Tasks	A project can have any number of task folders. All of those folders will be included in any project template that you create from that project.	You can include the tasks that were created in the original project, and any dependencies among them.
Discussions	All discussion forums that have been created in the original project will be included in any project template that you create from that project.	You can choose to include discussion topics, posts and attachments in your template, or leave them out.
File Releases	Any packages or releases that have been created in the original project will be included in the project template. If you have mapped a planning folder to a file release, that mapping is also included.	

Wiki		When you create a template from an existing project, all wiki pages that users have created in the original project can be included in the project template.
Integrated applications	All external applications you have integrated into the existing project become part of the project template.	Which elements are included depends on the application that is integrated. See the **Content includes** section of the *Projects* > *Templates* tab.

What is a story point?

A story point is a measure of effort that expresses the relative difficulty of implementing a user story. You can use story points, also referred to simply as “points”, to help estimate how much work can be done in a sprint.

Story points are useful for relative measurement. A story point has no specific value in hours, or lines of code, or anything else. When you are estimating work, use story points only to compare one piece of work with another.

When you record a value in the **POINTS** field of an artifact, that value is added to the total points (story points) in the planning folder that the artifact belongs to. You can then use that figure to support forecasting.

In the parent artifact, you can view the total of story points assigned to each of its children. The calculator icon indicates that the artifact’s points is a sum of its child artifacts’ points within the project. If the parent artifact has children in other projects across the TeamForge, you can include those points in the total as well. The icon ( ) indicates that the artifact’s points include its foreign child artifacts’ points.

NOTE: The Tracker Admin needs to set the tracker to include foreign children in points calculations.

What is a tracker workflow?

To help users handle their tracker items effectively, you can set up some work flow rules. Workflow rules require a user to do something to a tracker before they can reassign it or update its status.

Administrators can define these kinds of workflow rules:

- **Status transitions that a user can make based on an artifact’s current status** - For example, if an artifact is `Open`, you can specify that it can be changed to `Pending` or `Cannot Reproduce`, but not to `Closed`.
- **Status transitions that a user can make based on his or her role** - For example, if an artifact is `Pending`, you can specify that only users with the role **QA Engineer** can change it to `Closed`.

- **Field values that a user must provide when making a specific status change** - For example, if an artifact is `Closed`, you can specify that a user must enter a comment in the `Comments` field before changing the artifact's status to `Open`. You can also require an attachment.

You can create a workflow for each combination of tracker status values in a tracker. A tracker can be cloned within a project or across the projects along with the workflow. If a user role is not existing in the destination project, a new role is created with the same name. The permissions associated with the role are not copied from the source project.

What is a user group?

To manage multiple users at once, create a group to represent them.

You can create a group to facilitate managing many users who share one or more characteristics.

For example, giving all the users in the accounting division access to all financial projects might be laborious if you assigned the permissions one at a time. Instead, create a group and assign it access to the financial projects category.

A user group can have any number of roles. When a role is assigned to a group, every member of that group has that role.

If a project is a subproject of another project, it may inherit user groups and their associated roles from the parent project.

> **NOTE:** A user who has a role in a project by virtue of group membership is not necessarily a member of that project. Becoming a member of a project is a separate process.

Group permissions are cumulative. This means that each member of the user group has all the access permissions allowed by all of the assigned roles, plus any permissions that may have been assigned by other methods, such as application permissions or individually assigned roles.

Related Links

- Manage User Groups

Velocity and Average Velocity

Your team's velocity is the amount of work the team has shown it has completed in an iteration. You can use this information to help estimate how much work can be completed in future iterations.

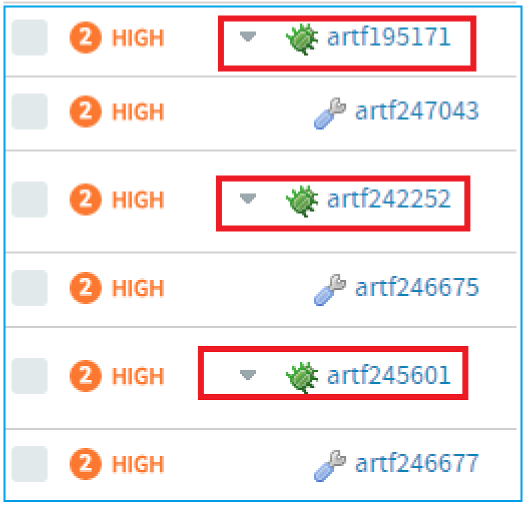
In the planning folder summary page, you can see your team's velocity for the specific Iteration planning folder.

NOTE: Velocity is displayed only on the Iteration planning folder summary page.

Velocity is calculated based on the sum of points of closed artifacts in an iteration.

Note that velocity calculation takes into account the following:

- Points of the parent artifacts, that is, only closed artifacts which are at the top level of the planning folder. For example, in the following illustration, only the points of the highlighted artifacts and not their child artifacts are taken into account for velocity calculation.



- Autosummed points of these closed parent artifacts do count as well.

Velocity only makes sense as a relative measure. There is no specific velocity that is good or bad or standard, because no two teams take on work of exactly the same scale or complexity. However, if your team's velocity is increasing from sprint to sprint, you can surmise that you are on the right track.

Average Velocity

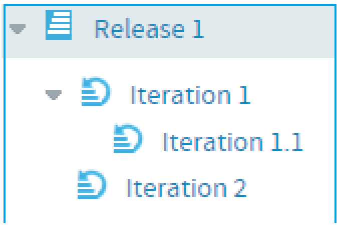
The average velocity is calculated for completed and ongoing iterations and is displayed only on the Release planning folder summary page. Average Velocity calculation is as follows:

```
Average velocity = Sum of iteration velocity of all iterations within the
specific Release/Total number of iterations
```

Note the following for velocity and average velocity calculation:

- Nested iteration planning folders (child iteration planning folders or subfolders) are not included for velocity or average velocity calculation for the parent planning folder. For example, in the following illustration:

  Average velocity for Release 1 = Sum of velocity of Iteration 1 and Iteration 2 / 2.

  In the above calculation, Iteration 1.1 (being the subfolder) is not included. Similarly, velocity calculation for Iteration 1 does not take into account its subfolder Iteration 1.1.

What is an activity table component?

An activity table is a special kind of text component that gives project members quick access to a focused set of artifacts, action items and work products.

An activity table includes organized links to up-to-the-minute information about work that is going on in the project. You can add, remove or change these links with the HTML editor that comes with every text component.

You can add static links to particular artifacts, or you can link to a search query. When you link to a query, you enable project users to get all the results of the query with a single click.

The activity table template is just a suggestion for how you might want to organize your information. You can change any of its properties to support the way your project members work.

What is a documents component?

A documents component allows team members to work with documents directly from a project page.

A documents component is like a window into your Documents tool. Instead of making users go to the documents tool and search for a document on their own, you can create a documents component on your project page that is linked to a folder in your documents tool. Each page can have its own list of relevant documents.

Users still go the documents tool, but now you guide them to what they need there.

What is a global project role?

A global project role is a ready-to-use role available in all projects. Only site administrators or restricted site administrators can create and manage a ready-to-use role.

As a project administrator, you can use global project roles provided by the site administrators instead of creating and managing roles tailored to your projects.

> **NOTE:** You can use the ready-to-use roles to set up your team faster and with little fuss. However, you may not be able to edit the ready-to-use roles.

Before you create a role in your project, it is a good idea to check all the available ready-to-use roles. You are likely to get ones that grant the desired set of permissions.

Global project roles serve a different purpose from that of a site-wide role. The site-wide roles enable site administrators to create restricted site administrators for providing assitance in site management. Besides that, site-wide roles can be used to grant tool/application access across the site to a user.

What is a project page?

A CollabNet project page is a place where users can see and add information about the project, such as messages from the project manager, open issues or documents you want people to read.

You can build your own project pages to design, manage and track your project's lifecycle. When you create a project page, users automatically see it in their navigation panes.

To post information or provide functionality on your project page, you add a page component of the appropriate type for the information or functionality you are working with.

For example, to let people know about how the project is coming along, add a Project Statistics component to the page. To let project members upload documents for other users to read, you add a documents component.

If the page is a part of a parent page, it appears as a node in the tree in the left navigation area.

You can add a page directly to the top of the project, or as a subpage.

> **TIP:** When you have your project pages defined and your process honed, you can export your project pages in the form of a project template, so other projects in your organization can reuse the resources you created. See Create a Project Template.

What is a project page component?

A project page is a collection of simple portlet-like components that enable you to add custom HTML content, reports, project tracker queries, and much more to the project home page so your team quickly access what is important.

Project page components add functionality to a project page. You can use components to communicate with users or project members, or to invite them to contribute information or resources to the project.

- Promote a standard lifecycle for all of your projects by using project pages to lay out information the way that works best for your business.
- Increase developers' productivity by putting the information most relevant to your project right on the homepage.
- Improve project coordination by combining related documents, graphs, artifacts, and content into single, digestible web pages.
- Maintain information security. The project manager controls who has access to each of the pages and components.
- Keep project members informed with live data and content that can be quickly updated as often as you like.

Project pages put the project manager in the driver's seat. You define how you want your project to look and what information you wish to share with your members and customers. Using a simple portlet-like layout, project pages let you add custom HTML content, reports, project tracker queries, and much more to the project home page so your team can quickly access what is important. All your pages are organized and displayed in a simple navigation menu on the left side of the home page.

In addition to customizing the project home page itself, you can also create and integrate additional web pages on the fly. Whether you need a page to display daily status reports or want to publish an HTML blog, a few clicks is all you need. No need to use any fancy tools or even know HTML.

For example, on a project page titled QA, you might have:

- A trend graph showing the open bugs in your project over a period of time.
- A Tracker query list showing you what hot bugs are currently in the system.
- A block of custom HTML content that the QA lead updates daily giving users his or her take on how the quality of the product is doing that day.

What is an Available upon Request role?

The **Available upon Request** role is a role which a project member in TeamForge can submit a request for. You must be a site administrator or project administrator to create an Available upon Request role.

While creating global or project roles, you can select the **Available upon Request** option to allow the project member to see the role in **Request Additional Roles** section on the **Project Home** page.

When a project member requests a role, the request is submitted to the project administrator for approval. The project member receives an email notification when the request is either approved or denied.

> **NOTE:** Based on the project administrator’s discretion, a few role requests may be granted immediately on request, while the other role requests may need approval.

What is a Tracker Search Results component?

A Tracker Search Results component provides quick access to Tracker information from a project page.

Some tracker searches are used so frequently that you want people to be able to see their results without exploring. For example:

- People working on your project may need to know at the beginning of every work day which artifacts have been updated overnight by a remote team.
- A cross-project leadership team may need a daily count of started and completed work items to support forecasting and planning.

Such heavily-used information might be a good candidate for a tracker search component on a prominent project page, such as the project home page. When you set up this component based on a popular tracker search that you devised and saved, project members and users can save time by viewing the information at a glance, without having to go find the information themselves.

> **NOTE:** If your site administrator has set up Project Tracker for your project, you can also use a project page query component to search your Project Tracker data.

What is a project statistics component?

A *project statistics component* gives users a graphical view of recent statistics for tasks, trackers, documents, and file releases from the project page.

For more information about the project statistics component, see Create a Project Page Component.

What is a text component?

A text component is a self-contained editor with which you can write anything you want on your project page.

You don’t have to know anything about HTML.

Text components include everything you would expect in a fully functional HTML editor. You can add tables, upload images to embed into the page, create hyperlinks, and assign font colors and styles. Advanced users can work directly with the HTML source.

What is the complete wiki syntax that TeamForge supports?

Since TeamForge makes use of the JSPWiki rendering engine internally, you may reference the JSPWiki syntax documentation at the link below for a complete list.

http://www.ecyrd.com/JSPWiki/wiki/TextFormattingRules

Please note however that JSPWiki has progressed beyond the release available in TeamForge currently, so there may be some discrepancy between what JSPWiki supports and what TeamForge supports. TeamForge also does not currently support the JSPWiki plugin framework, so any formatting plugins referenced on the URL will not be available for TeamForge.

What is Team?

Teams helps you create logical groups of cross-functional members comprising architects, developers, testers and so on.

A team’s view of backlogs complements the planning folder view. While planning folders represent backlogs (release, iteration, etc), a team’s list view represents the team’s view of the backlog (across releases, iterations, etc).

In an agile environment, project activities are broken into smaller units of work items and are assigned to various project teams. The Teams feature allows you to create these physical teams in TeamForge so that you get the specific team’s view of backlogs. This knowledge of the team’s view of work items helps you facilitate effective communication and execute project activities in a more structured and organized fashion. For example, you can easily view and filter artifacts which are taking longer than estimated within a team, analyze the scenario, and identify the impediments. Once identified, you can communicate them to the relevant team(s), re-assign it to other appropriate team(s) or team member(s), and resolve them quickly.

What is a team owner?

Any project member can be designated as a team owner.

The team owner does not have the permission to create or delete teams. They can view their team information and edit them; while editing their team details, they can add or remove members, or designate another member as the team owner.

What is a Publishing repository? How does it work?

Publishing repository, like the branding repository, is one of the default repositories that's created automatically when a TeamForge project is created and is intended to contain publicly-consumable files.

The Publishing repo has a www directory. Files in the www directory are checked out to a working directory and served by the Apache server with no user authentication checks. In other words, files stored in the www directory do not go through TeamForge's RBAC checks and are publicly accessible even if the user is not logged in (accessible via a direct link to the file). By design, the files stored in the www directory are meant to be public on both “public” and “private” projects no matter whatsoever. However, files stored in no other directories but www are publicly accessible.

Additional security Enhancements added to Publishing Repository Site administrators can now toggle access to Publishing Repositories and restrict access based on defined RBAC. See DISABLE_REMOTE_PUBLISHING for more information.

What is a CERT Advisory?

CollabNet Product Support monitors the CERT coordination center (http://www.cert.org/) for notification of vulnerabilities or exploits against applications that CollabNet TeamForge provides.

If CollabNet Technical Support identifies an advisory that may indicate potential challenges for users who have deployed CollabNet TeamForge, Support proactively releases a notification and a statement of action. CollabNet will provide product updates as it deems appropriate or necessary.

Advantages of Using the Apache TIKA Parser Library for Indexing

Starting TeamForge 7.0, the underlying parser library for indexing has been changed from Stellent to Apache TIKA.

The Apache TIKA parser library has the following advantages over the Stellent parser library:

Issue	Stellent	Apache TIKA
Stale process issue	Parsing of corrupt or unrecognized files by the Stellent parser libraries often result in stale processes that consume swap space and add to the load on the system, which may at times lead to site outage. To manage such processes, you may choose to create and deploy stale process monitors and the stale processes, when detected, must be removed manually to prevent site outage.	Parsing of unrecognized or corrupt files by Apache TIKA libraries is robust and needs no manual intervention as there are no stale process issues.

Search queue processing speed	It takes five minutes to timeout when the Stellent parser library encounters a corrupt or unrecognized file that it knows not how to parse. If there are more such corrupt or unrecognized files, more time is wasted by the indexer waiting for a response (or a timeout) from the Stellent parser, which in turn adversely impacts the search queue processing speed.	The Apache TIKA parser library is capable of determining whether a file it encounters can be parsed or not. As no time is wasted by the indexer waiting for a response (or a timeout) from the parser, the search queue processing speed is better with the Apache TIKA.
Multiple processes Vs Single JVM	For parsing files, the Stellent parser library spawns one subprocess per file. Meaning, the number of subprocesses is equal to the number of files to be parsed and it is possible that we may end up with the stale process issue as discussed earlier. As a result, if the Stellent processes consume more resources, other processes and applications are left with scarce resources.	The Apache TIKA, being a Java-based parser library, works within the JVM and makes the external resource pool available exclusively for other processes and applications. As the search JVM, where the Apache TIKA library lives, can also be separated starting TeamForge 7.0, it can be managed better.

These are some of the frequently asked questions on the installation, upgrade, and site admin related activities in TeamForge.

Why am I getting “Could not connect” status for my email and search server?

On the **System Tools** page, when you see “Could not connect status for search and email servers,” you must stop and start your phoenix.sh process.

You may also need to set the JAVA_HOME environment variable to the location of your JDK.

The stop/start Phoenix commands:

```
sh /opt/collabnet/teamforge/runtime/scripts/phoenix.sh stop
sh /opt/collabnet/teamforge/runtime/scripts/phoenix.sh start
```

Why are the dynamic images that TeamForge creates broken?

If you have a fresh install of TeamForge and you’re noticing that the dynamic images are not correct, you may be missing a library that is needed to create the images.

The easiest way to find this is to check and see if you have the xorg-x11-depreciated-libs rpm installed:

```
rpm -qva | grep xorg-x11-depreciated
```

Watch for the results. If you see that you have the xorg-x11-depreciated-libs rpm installed, and after a server restart you’re still not seeing the images, please open a support request. If you do not have the xorg-x11-depreciated-libs rpm installed, it can usually be installed by performing a simple `up2date xorg-x11-deprecated-libs` and restarting TeamForge.

Due to firewall restrictions I cannot send email from James. How can I resolve this?

If James is unable to send email directly due to firewall restrictions, or mail being rejected from the application servers IP address, you may have to configure it to use a gateway mail server to send outgoing messages through.

To do this, you will need to add the following to the `<mailet match="All" class="?RemoteDelivery">` directive in the james config file at `/opt/collabnet/teamforge/james/james-<version>/apps/james/SAR-INF/config.xml`:

```
<gateway>smtp.example.com</gateway>
<gatewayPort>25</gatewayPort>
```

You should find these commented out on line 362 of the config file. If your gateway mail server requires authentication to send email, you may also add the following directives:

```
<username>username</username>
<password>password</password>
```

Why am I not getting any error messages when executing the Subversion upgrade script?

Error messages may come when Subversion is installed with a dependent package from an unknown source.

The Subversion working copy script assumes that Subversion is installed with the dependent packages from a proper source repository(RHEL/CollabNet). If you install any dependent packages from any unknown source that is not authorized by RHEL/CollabNet, it will result in inconsistency and this cannot be handled by the Subversion working copy script.

Why do I get a JBoss error “failed to start in 240 seconds, giving up now” while installing TeamForge?

You get this error when the system’s RAM is less than the minimum recommended value of 4GB. However, it’s most likely that JBoss will start within a few minutes.

To make sure that JBoss starts up, check the service.log file using this command:

```
tail -f /opt/collabnet/teamforge/log/apps/service.log
```

If you see messages like the following, the TeamForge application will start in a few minutes.

```
Check Port Available PASSED: Port 4444 on localhost is available
Check Port Available PASSED: Port 4445 on localhost is available
Waiting for application server to start up.. this can take a few minutes.
```

JBoss crashed with out of memory error, how do I prevent this?

This can indicate that the JVM heap size is set too small.

You can adjust this by changing the -Xms and -Xmx settings of the *JBOSS_JAVA_OPTS* token in `site-options.conf` and rebuilding runtime.

This will appear if the JBoss application server has crashed and you find this error in the server.log:

```
INFO [STDOUT] java.lang.OutOfMemoryError: Java heap space
```

The default maximum heap size of 640MB can cause issues on a heavily used site. If the CTF application is the only thing running on the server, you can increase this to half of the total physical ram on the machine. This should still allow enough memory for the OS and other necessary processes. If you are also running the app, database and scm on the same machine a maximum heap size of 1/4 or the total ram maybe a better setting. Determining the right JVM settings for your install will require testing with your particular usage patterns and database.

You can view the current memory usage under the JVM Environment section of the JBoss webconsole at http://:8080/web-console/. You will need to log in using the CTF admin password.

JBoss status is in ‘starting’ for a long time. How to have JBoss started successfully?

If JBoss is not started successfully (status remains ‘starting’ for a long time), you may have to wait until it starts up successfully or you can kill JBoss process (using its PID) and restart it again.

The following error messages show up when you try to start or stop JBoss respectively while its status is still ‘starting’:

```
Cannot execute action as another process is holding the lock.
```

```
Cannot stop service 'jboss' while in status 'starting'
```

To kill the JBoss process:

```
kill 9 <JBoss PID>
```

Why am I not able to see the status of the Postgres in the collabnet startup script?

You may not be able to see the status of the Postgres if the host name of the HOST_ token is set to localhost in a SaaS multibox setup.

The Teamforge installer fails to add the IP address of the database box to the listen address in the postgresql.conf file if the host name of the HOST_ token is set to localhost in a SaaS multibox setup.

> **NOTE:** You must add the IP address of the database box to the listen address in the postgresql.conf file.

Why does the SOAP service show “could not connect” on the Server Status page when everything else appears to work?

This can be caused by an incorrect host name in `/etc/sourceforge.properties`. Rebuilding runtime will correct this, assuming the hostname is set correctly in the `site-options.conf` file.

This issue can occur when using the restore.py script to restore data from a TeamForge instance with a different hostname.

Why does startup fail or produce errors?

If TeamForge fails to start up, or is starting but is throwing errors on every page, then typically something went wrong during the JBoss bootstrap process.

Fortunately, JBoss logs this process to: `/opt/collabnet/teamforge/jboss/jboss-<version>/server/default/log/boot.log`.

TeamForge writes its startup and shutdown info, as well as any system errors to `/usr/local/soureforge/log/server.log`. If you encounter a system error while using TeamForge, it is logged here. Additionally, if you see an ‘exid’ string in the application, the Java stack trace for that exid will be logged in this file.

Why do I get a URL “not found” or “moved permanently” error after applying a patch/ upgrade?

If you are experiencing a URL “NOT FOUND” or “MOVED PERMANENTLY” error after applying a patch or upgrade, then set Apache ProxyPreserveHost token to ON in the `httpd.conf` file.

If you have applied a patch or upgrade and are now receiving the following error:

```
<The document has moved <a href="https://www.<site>/sf/global/jsp/buildtime.ht
ml"
    format="html" scope="external">here</a>.</p>
<hr> <address>Apache/2.2.3 (Red Hat) Server at www.<site>.com Port 80</address>
 </body></html>
Not Found
The requested URL /sf/sfmain/do/userPicker/projects.pftool//sfmain/do/listMoni
toringUsers/projects.pftool/discussion.
announcements was not found on this server
```

Or if you are trying to add users to a monitoring list, and are receiving the following error:

```
Not Found
The requested URL
    /sf/sfmain/do/userPicker/projects.pftool//sfmain/do/listMonitoringUsers/pr
ojects.pftool/discussion.announcements
    was not found on this server.
```

Set the ProxyPreserveHost token to ON in the `httpd.conf` file.

How do I require approval for new user accounts?

You can configure the system so that new users can create their own accounts, but the accounts are not activated until a site admin approves them.

To enable this mode of operation, add the following line to `/opt/collabnet/teamforge/sourceforge_home/etc/sourceforge_configuration.properties`:

```
sf.approveNewUserAccounts=true
```

Once this line has been added to the file, restart TeamForge for it to take effect.

Please note that site admins can still create accounts for new users and they will not be held for approval. Also note that the user will receive an email from TeamForge telling them to confirm their password by clicking on the given link, and the link will not work. The password is properly set on account approval.

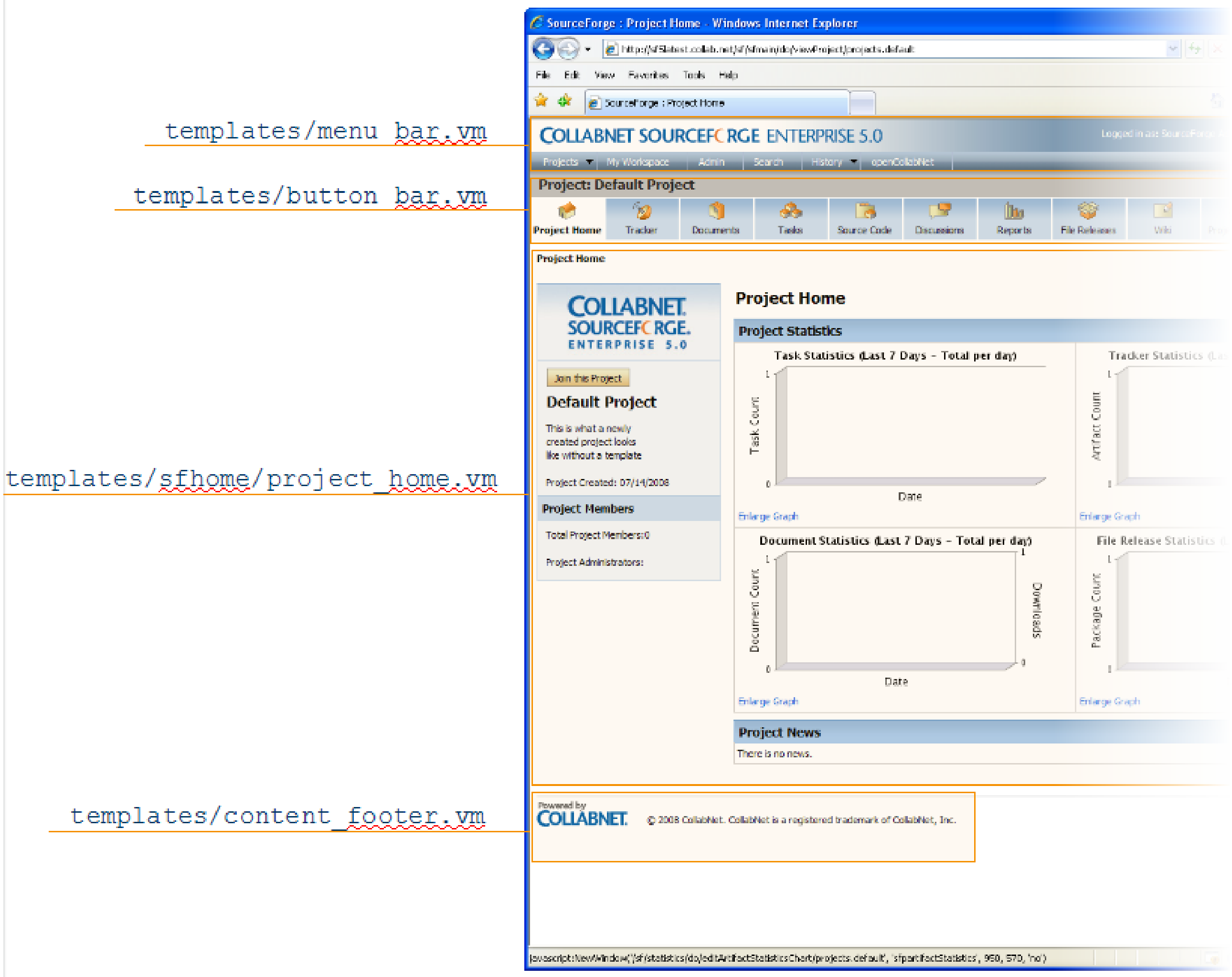
How does TeamForge use Velocity templates?

Velocity is the templating language that CollabNet TeamForge uses to render areas of the site with dynamic information.

You can override the instructions contained in any of these Velocity templates by placing a file of the same name in the equivalent path in the `branding` repository in the `look` project on your site.

Velocity templates are located in the `templates` directory in the branding repository.

Velocity File	Description
menu_bar.vm	Controls the rendering of the top bar across all pages in the system. Displays a small login form, the site logo and current user information as well as the search and projects drop down menus.
blank_menu_bar.vm	Contains only the top logo, without the menu that appears below it.
body_header.vm	Rendered immediately after the opening body tag. If a site requires everything to be contained in some other container, this template can be used.
body_footer.vm	Rendered immediately before the closing body tag. If a site requires everything to be contained in some other container, this template can be used.
button_bar.vm	Controls the rendering of the bar beneath the menu bar, which contains the 'Quick Jump" link as well as the buttons that appear on any project page (the one containing the applications). Site admin pages, user settings pages (e.g. my workspace, dashboard) and project pages use different sets of buttons that are passed into this template for rendering.
content_header.vm	Rendered after the button bar; wraps the actual contents of the page being viewed.
content_footer.vm	Rendered before the body footer; wraps the actual contents of the page being viewed. Contains the Copyright notice.
sfmain/home.vm	Velocity template that generates the site home page.
sfmain/project_home.vm	Velocity template that generates the default project home page.


templates/menu_bar.vm
templates/button_bar.vm
templates/sfhome/project_home.vm
templates/content_footer.vm
SourceForge : Project Home - Windows Internet Explorer
COLLABNET SOURCEFORGE ENTERPRISE 5.0
Project: Default Project
Project Home
Tracker
Documents
Tasks
Source Code
Discussions
Reports
File Releases
Wiki
Join this Project
Default Project
This is what a newly created project looks like without a template
Project Created: 07/14/2008
Project Members
Total Project Members:0
Project Administrators:
Project Statistics
Task Statistics (Last 7 Days - Total per day)
Document Statistics (Last 7 Days - Total per day)
Enlarge Graph
Project News
There is no news.
Powered by COLLABNET
© 2008 CollabNet. CollabNet is a registered trademark of CollabNet, Inc.

What happens when log files get too big?

Log files can grow very large over time. To maintain reasonable log file sizes, log files are rotated automatically on a schedule.

During this automatic log rotation, live logs are archived every day at `00:00`.

Archived logs are stored in compressed form in a directory alongside the live log. For example, if live logs are stored at `<LOG_DIR>/{apps, apache,...}`, then compressed log archives are stored at `<LOG_ARCHIVE_DIR>/{apps, apache,...}`.

The directory structure of the log directory is preserved in the log archive directory.

NOTE: Empty log files are not compressed.

How do I make the monitoring messages be sent from Forge Administrator?

You can change the default behavior for site options by changing the value from “false” to “true” in this statement:

```
# MONITORING_EMAIL_FROM_ADMINISTRATOR=false
```

If this site option token is set to `true`, then “From:” field is the Forge Administrator, else it is from the user who made the change that initialized the monitoring email.

How do I enable post-commit logging?

You do this by editing the `post-commit.py` file.

Edit the `/opt/collabnet/teamforge/runtime/sourceforge_home/integration/post-commit.py` file.

Search for `log.setLogging(False)` and modify the value from `False` to `True`.

What is the suggested log configuration for a production system?

To troubleshoot installation issues, the default log4j configuration is set to DEBUG. This can cause the log files to become quite large. Once your system is successfully installed and in use, you should drop the log levels down to INFO.

See Change the Logging Level on Your Site for how to do this.

If you still have a problem with very large log files, you may want to set up log rotation. Log rotation means to move the log files to a compressed archive to keep them under control.

How do I remove the build and test link from TeamForge pages?

The site administrator can remove these links by checking out and editing the branding repository from the look project.

To remove the build and test links from your TeamForge navigation panel, check out the branding repository from look project and reconfigure the links as shown in the following code sample:

NOTE: You must be logged on as administrator to perform this task.

```
[branding_stage]$ cd branding
[branding]$ mkdir -p i18n/com/vasoftware/sf/i18n/apps/sfmain
[branding]$ echo "configure_build_and_test.systemUrl.default=" > i18n/com/vasof
tware/sf/i18n/apps/sfmain/application.properties
[branding]$ svn add i18n
A i18n
A i18n/com
A i18n/com/vasoftware
A i18n/com/vasoftware/sf
A i18n/com/vasoftware/sf/i18n
A i18n/com/vasoftware/sf/i18n/apps
A i18n/com/vasoftware/sf/i18n/apps/sfmain
A i18n/com/vasoftware/sf/i18n/apps/sfmain/application.properties
[branding]$ svn commit
" Old: "enter paragraph information and use html tags for bullet points.
```

How do I resolve timeouts when calling web services?

This is due to the requested operation taking longer then your client SOAP stack is configured to wait before throwing a timeout. You will need to reference your client documentation to see how to update the timeout properties of the connection.

For AXIS in java, you can do this via the `sun.net.client.defaultReadTimeout` property.

```
System.setProperty("sun.net.client.defaultReadTimeout", "600000"); //10 minute
 timeout, in ms
```

These are some of the frequently asked questions on security.

What are the implications of deprecating TLS protocol versions 1.0 and 1.1?

In addition to security vulnerabilities, TLS protocol versions 1.0 and 1.1 do not support modern cryptographic algorithms. The software industry (including popular browsers such as Chrome, FireFox and so on) is set to deprecate the TLS protocol versions 1.0 and 1.1 by March 2020 and so is TeamForge. Customers are therefore advised to upgrade your sites to be able to negotiate with TLS 1.2 connections. Upgrade your clients to the latest version in case you face any SSL handshake issues while connecting to TeamForge.

With this move to deprecate TLS protocol versions 1.0 and 1.1, we must fix the `SSLProtocol` and `SSLCipherSuite` options in the `/etc/httpd/conf/httpd.conf` file and restart the Apache server.

Do this where you have Apache running. For example, TeamForge application server and Subversion servers have Apache.

```
SSLProtocol             all -SSLv2 -SSLv3 -TLSv1 -TLSv1.1
SSLCipherSuite          ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA
256:ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-CHAC
HA20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305:DHE-RSA-AES128-GCM-SHA256:DHE-RSA-AE
S256-GCM-SHA384
```

You can also have this fixed permanently by setting up the `SSL_PROTOCOL` and `SSL_CIPHER_SUITE` `site-options.conf` tokens for TeamForge 18.1 and later.

```
SSL_PROTOCOL= all -SSLv3 -TLSv1 -TLSv1.1
SSL_CIPHER_SUITE=ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECD
HE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-CHACHA20-PO
LY1305:ECDHE-RSA-CHACHA20-POLY1305:DHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GC
M-SHA384
```

```
teamforge provision
```

What are the security features available in TeamForge?

TeamForge is equipped with a number of security features, which include:

- SELinux Support
- SSL Support
- Site option tokens to enforce password control policies
- Automatic password creation and encryption
- Password obfuscation
- Prevention of cross-site scripting
- Role Based Access Control (RBAC) and Path Based Permissions (PBP)
- Size restriction for document uploads
- Document safe download mode
- Incorrect login attempts-account lock feature
- SCM_DEFAULT_SHARED_SECRET to allow SCM Integrations to securely communicate with TeamForge
- Support for OAuth/LDAP/SAML
- Restrict domains for CSRF, Prohibit harmful file uploads

After switching to ADS authentication, why did the Create button disappear from the user admin section?

When using external authentication such as LDAP, creating users from within the application is disabled. All users must be created via LDAP.

See login-config.xml for more information.

Can I block project data from public search engines?

Yes, edit the robots.txt file to specify the pages or directories that should not be indexed by search engines.

If your site has some content that you may not want to be publicly searched for, or if some search engine hits are causing the site to slow down, you can make this change.

Only one robots.txt file can be created for a site. This file can contain a list of url patterns that should not be indexed, against the web crawler name.

The default robots.txt is available in the SITE_DIR/var folder. The robots.txt file is accessible without logging into CollabNet TeamForge, via the `domain/robots.txt` URL. You can update and commit the robots.txt file into the branding root directory, and the system then places the file in the `SITE_DIR/var/overrides` folder.

> **TIP:** You must have the commit permissions to create a robots.txt file under the branding repository in the Look Project.

How can I enforce strong passwords?

You can configure the application to reject passwords that do not meet your security criteria.

To enforce password requirements, place the following lines in `/opt/collabnet/teamforge/sourceforge_home/etc/sourceforge_configation.properties` file.

```
system.password.min-length=5
password.requiresNumber=true
password.requiresNonAlphaNum=true
password.requiresMixedCase=true
```

Once these lines are in place, restart TeamForge for them to take effect. The above example would require a password of at least 5 characters that must include at least one (1) mixed case letter, at least one (1) number, and at least one non-alphabetic character, e.g. Us3r!

NOTE: These settings apply only to new passwords. Anyone in the system currently will be able to continue to use their existing, potentially weak, password. You should force all users to change their passwords after changing these.

How do I configure Subversion to authenticate against multiple LDAP domains?

For some configurations, a Subversion server may need to be authenticated against multiple LDAP domains. This is possible by modifying the Apache configuration.

This is now possible due to the mod_authn_alias module for Apache. The external link for the module contains multiple usage scenarios. You will need to confirm that your Apache has been compiled with the module enabled. (This is the case for CollabNet Subversion binary packages since 1.5.4). If it is compiled as a module, make sure it is enabled via the LoadModule directive in your Apache configuration.

Example for configuration usage for authentication against three LDAP servers :

```
<AuthnProviderAlias ldap ldap-US>
    AuthLDAPBindDN cn=ldapuser,o=company
    AuthLDAPBindPassword password
    AuthLDAPURL ldap://ldap-us.company.local/ou=Developers,o=company?sub?(obje
ctClass=*)
</AuthnProviderAlias>

<AuthnProviderAlias ldap ldap-EU>
    AuthLDAPBindDN cn=ldapuser,o=company
    AuthLDAPBindPassword password
    AuthLDAPURL ldap://ldap-EU.company.local/ou=Developers,o=company?sub?(obje
ctClass=*)
</AuthnProviderAlias>

<AuthnProviderAlias ldap ldap-IN>
    AuthLDAPBindDN cn=ldapuser,o=company
    AuthLDAPBindPassword password
    AuthLDAPURL ldap://ldap-in.company.local/ou=Developers,o=company?sub?(obje
ctClass=*)
</AuthnProviderAlias>

<Location /svn>
    DAV svn
    SVNParentPath /opt/subversion/repos
    AuthType Basic
    AuthName "Subversion Repository"
    AuthBasicProvider ldap-US ldap-EU ldap-IN
    AuthzLDAPAuthoritative off
    Require valid-user
</Location>
```

How do I authenticate multiple LDAP via Apache?

If you need to add multiple OU= values in the LDAP url you must have separate LDAP urls and utilize AuthnProviderAlias to check both LDAP searches.

Use the following AuthnProviderAlias to check LDAP searches.

```
LoadModule authn_alias_module
modules/mod_authn_alias.so

<AuthnProviderAlias ldap ldap-alias1>
AuthLDAPBindDN cn=youruser,o=ctx
AuthLDAPBindPassword yourpassword
AuthLDAPURL ldap://ldap.host/o=ctx
</AuthnProviderAlias>

<AuthnProviderAlias ldap ldap-other-alias>
AuthLDAPBindDN cn=yourotheruser,o=dev
AuthLDAPBindPassword yourotherpassword
AuthLDAPURL ldap://other.ldap.host/o=dev?cn
</AuthnProviderAlias>

Alias /secure /webpages/secure
<Directory /webpages/secure>
Order deny,allow
Allow from all

AuthBasicProvider ldap-other-alias ldap-alias1

AuthType Basic
AuthName LDAP_Protected_Place
AuthzLDAPAuthoritative off
Require valid-user
</Directory>
```

How does TeamForge authenticate CVS users?

CVS is treated as a special case when managed by a TeamForge site. It is not authenticated in the same way as SOAP API clients.

CVS relies on the Linux operating system to provide access and security. This includes permissions on individual repositories and access to the server itself. To add users, change passwords, create repositories, etc., the TeamForge integration simply changes the appropriate settings in the Linux operating system.

Users access CVS via an encrypted SSH session. To support this, TeamForge creates accounts on the Linux server that hosts the CVS repository. A typical CVS repository is created in the /cvsroot directory and is owned by root, with a group assigned by TeamForge. To gain access to a repository, TeamForge will add a user to the appropriate system group.

When TeamForge detects that a user’s password has changed, it changes the password for that user on the Linux server too.

Users also have the option to use SSH keys or Kerberos tickets.

When a user is added to the Linux server, the login shell is cvssh, which limits their activities to CVS commands.

> **NOTE:** Do not expose cvspserver (the TCP protocol over port 2401) either internally or to the Internet if there is any way you can avoid it.

Password Changes under Internal Authentication

To set the password for the user at the operating system level, TeamForge needs to have the clear-text version of the users password. The only time TeamForge has this is when the user uses the Change Password form in the Web UI. This is because the database-stored version, as an MD5 Password Hash, is a one-way encryption and can’t be decrypted.

On a successful password update, TeamForge makes a SOAP call to the integration server that manages CVS. For this reason, the integration server must be SSL-enabled.

Password Changes under External Authentication

When a password change happens in an external authentication system, TeamForge does not immediately know that the password has changed. TeamForge needs a way to detect that the password has changed.

To accomplish this it keeps a copy of the last password the user successfully logged in with as an MD5 Password Hash in the same database table and field that it normally uses for Internal Authentication.

Now that TeamForge has a reference point, it still needs a clear-text copy of the password to make the change at the Linux operating system level. The only time this is available is when the user logs into TeamForge via the Web UI or SOAP API. So upon a successful login TeamForge compares the password to the encrypted one. If it is different it tells the Linux operating system to change the users password and then saves it in the database (as an MD5 Password hash).

Until the user logs into TeamForge, the CVS server will still have and accept the old password. There is no CVS server-side way to trigger a password update, unless an alternative method is used, such as LDAP or Kerberos.

Alternative Authentication

Because users use SSH to access a TeamForge-managed CVS server, it is possible to configure SSH to accept other authentication features such as SSH keys and Kerberos tickets. It is even possible to disable the use of passwords and require the use of other alternative methods.

- TeamForge supports SSH Keys natively. The user uploads their public key into their profile under “My Settings” in the TeamForge Web UI. The key will automatically be copied to each CVS server that TeamForge manages.
- If TeamForge is using External Authentication and the method is Kerberos, then SSH can be configured to use the same Kerberos server. This allows users to use Kerberos tickets for CVS operations.

LDAP

Linux supports LDAP as an authentication source. If TeamForge is using External Authentication and the source is LDAP, then SSH can also use that same source. When you do this, passwords and user account status are observed in real time instead of as a mirror of TeamForge.

Can the users be forced to change their passwords at first login?

Yes, as a site administrator you can configure the CollabNet TeamForge site options to force the users to change their passwords at first login.

Setting the `REQUIRE_USER_PASSWORD_CHANGE` attribute as `true` in the `site-options.conf` file enforces password change on first login into CollabNet TeamForge.

> **NOTE:** You can not force password change on a user who had self-created the user account, or if a password-request had been raised for the user or if an administrator had reset the login password for that user.

Does TeamForge work with LDAP?

Yes, you can have your TeamForge installation authenticate against an LDAP server.

This is handy when users want to use a variety of different resources without having to maintain credentials for each one separately.

Overview

CollabNet TeamForge is a JBoss2 based application and relies on the JBoss JAAS service for user authentication. This enables a TeamForge site to authenticate users internally or externally.

Internal User Authentication

Out of the box, TeamForge relies on its local database to manage user accounts. This includes username, password, full name, email address and a variety of other meta data values. Passwords are stored in the database using the standard MD5 Password hashing algorithm1. The database is only accessible by the application itself and a user with root access to the physical server. While running in this default configuration users are allowed to change their passwords in TeamForge, and any user with site administration privileges can create and approve new user accounts.

External User Authentication

The JAAS service comes with several standard providers that allow TeamForge to be integrated with services such as LDAP, Active Directory and Kerberos. The JAAS service allows more than one source to be configured in the event several sources are needed.

> **NOTE:** It is possible to use both types of authentication with a single TeamForge installation. See your CollabNet representative for details.

To ensure that you are not locked out of your site, the site administrator account is always validated by TeamForge, not by LDAP.

LDAP accounts must conform to the TeamForge rules for user names and passwords. For example:

- If a password is used in LDAP that is shorter than the minimum allowable password length in TeamForge, you cannot create the user in TeamForge.
- A user name that starts with a special character, such as an underscore, will not be accepted by TeamForge, even if it is valid in LDAP.

(For detailed TeamForge user name and password rules, see Create a New User Account).

How is life different for the user under external authentication?

- When you turn external integration on, every user account (except the site administrator account) must have a matching LDAP entry to log in. This may require changing some existing accounts to match their corresponding LDAP records. (Accounts created after LDAP is in place are validated with the LDAP server when they are created, so you don’t have to worry about this.)
- Every login attempt (Web UI and SOAP access) is passed to the external provider. This means that any changes to the user status in the external system take effect immediately. Users who have already logged in and have valid sessions are not affected.
- When TeamForge is using internal authentication, a site administrator can change a user’s password. This is disabled for external authentication.
- Under external authentication, passwords can’t be changed in the TeamForge web UI. Users have to use the interface provided by the third-party authentication source to change their password. Such password changes are available immediately to TeamForge for the next login attempt.
- Site administrators can no longer create user accounts. The end user must create their own account by logging into TeamForge just like a user who already has an account. At that point TeamForge detects that a new account needs to be created and presents the new user with a registration form, which requests the user’s password n the external authentication system. On submit, TeamForge verifies the user account with the external system, and only if the username/password is verified does TeamForge create the new account.
- Once a new user has created their account, TeamForge can optionally be configured to put every new account in a pending status so that a site administrator can approve the new account. By default, new users will have immediate access to the system.

LDAP for Source Control

LDAP is integrated into your TeamForge source control services.

- For Subversion, the integration server queries TeamForge as needed.
- CVS authentication is not managed directly by LDAP, but each TeamForge user’s SCM password is synchronized automatically with the user’s LDAP password upon logging into TeamForge.

What can go wrong?

When TeamForge is configured to authenticate against an LDAP server and the LDAP server is down, all TeamForge authentication is disabled until the LDAP server is restored.

If a user does not exist on the LDAP server, or is deleted from the server, that user cannot log into TeamForge.

Why do I get the “Invalid command ‘AuthLDAPAuthoritative’” error when I try to set LDAP for SVN users?

The invalid command `AuthLDAPAuthoritative` error may occur if you need to upgrade Apache from version 2.0 to 2.2.

CollabNet Subversion 1.5 is bundled with the latest version of Apache (currently 2.2.x). It includes the module mod_authnz_ldap and does not include mod_auth_ldap. Hence compatibility issues arise due to missing directives. Upgrade your Apache version to 2.2 if you get the following error when trying to install CollabNet SVN:

```
 bash-3.00# /etc/init.d/collabnet_subversion start
    Starting CollabNet Subversion:
    Syntax error on line 29 of
        /etc/opt/CollabNet_Subversion/conf/collabnet_subversion_httpd.conf:
    Invalid command 'AuthLDAPAuthoritative',
        perhaps misspelled or defined by a module not included in the server co
nfiguration
    FAILED
```

How does TeamForge handle multiple redundant LDAP servers?

When configuring LDAP authentication for a TeamForge instance, there may be a business need for using multiple LDAP servers.

Follow the guidelines below for configuring.

The additional LDAP servers can be added to the `java.naming.provider.url` option in `login-config.xml`:

```
login-config.xml:
    <module-option name="java.naming.provider.url">
    ldap://primary/ ldap://secondary/↙module-option>
```

Once the primary and secondary servers have been defined, they will be consulted in order of definition for every authentication request. First the primary, and if the primary fails, then the secondary. This prevents specifying multiple servers for round-robin handling of authentication, but it can still be used for redundancy needs.

What user activities are tracked?

In case of a data security compromise, a record of who is performing what activities will help resolve some of the security issues.

Typically web servers log every page (or URL) being accessed, including the IP address of the user, date and time of access, etc. These logs are very useful in tracking the source of any security violations that may occur.

CollabNet TeamForge auditing tools are a powerful way to track unwanted and/or unauthorized changes within the system.

J2EE Architecture and Security

CollabNet TeamForge is a J2EE application that employs three-tier architecture to provide a secure environment for mission-critical data.

In a multi-tier architecture, access to each tier is restricted to the tier above it, effectively securing the tiers behind the firewall. For example, while clients (users accessing the system through a web) access the web server, they neither have access to the application and backend servers nor are they aware of their existence.

Similarly, the web server itself does not have access to the backend servers (database, SCM, mail etc.)

Exceptions to this rule include:

- Direct client access provided to the SCM servers. SCM servers are accessed across the firewall typically through SSH protocol (for CVS), or HTTP or HTTPS (for Subversion). SCM server data is also accessible in a view only mode through the web interface.
- Clients must have access to the mail server for posting messages to mailing lists.
- Mail server must have access to deliver messages across the firewall.

Clients can also access the SOAP APIs through the web server. The web server in turn forwards SOAP requests to the application server for processing.

With self signed certificates in place, what is the recommended protocol (SSH or HTTPS) to use while cloning a Git repository from a TeamForge Git server running on RHEL/CentOS?

When using a self-signed certificate on a Teamforge Git server, you cannot clone a repository using the standard git client on RHEL/CentOS.

In RHEL/CentOS, the Linux certificates used by git and other tools are stored in the `/etc/pki/tls/certs/ca- bundle.trust.crt` file. This file is managed by the RPM package system. If a certificate is added to the ca-bundle.trust.crt file, Trusted Root Certification Authority updates are not installed automatically which in turn leaves the system vulnerable to potential attacks.

An alternative way for adding trusted certificates is available at http://stackoverflow.com/questions/9072376/configure-git-to-accept-a-particular-self-signed-server-certificate-for-a-partic. However, it appears that some tools do not read files outside of the main bundle.

Therefore, it is recommended to use SSH protocol while cloning a git repository from a TeamForge Git server running on RHEL/CentOS.

Do I have to use the password provided by administrator always?

No, you don't have to use the password provided by administrator beyond your first login into CollabNet TeamForge.

The site administrator may have provided you with a user name and password after creating your user account in CollabNet TeamForge. When you login using those credentials, you might be asked to change your password for security reasons. At this time, you can set a password of your choice.

> **TIP:** You will not be asked to change your password if you had created your own account, or if a password-request had been raised for you or if an administrator had reset your password.

These are some of the FAQs on the roles and permissions in TeamForge.

Can I block a user's access to source code?

From time to time, you may need to prevent a user from using a source code repository.

SCM access can be withdrawn from a user by doing one of the following:

- Change the user's access to specific paths in the repository. (Subversion only.)
- Remove the role providing the desired source code permissions from the user.
- Remove the user from the project.

How this works depends in part on whether the repository is managed by TeamForge.

- If the source control server that hosts the repository is managed by TeamForge, the user's access is removed immediately.
- In the case of a CVS server that is not managed by TeamForge, all TeamForge administrators are sent an email requesting that the change be made manually. The request appears in the **SCM Access Requests** section of the **Integrations** page for approval.

Both the project administrator and the user receive email notifications when the change is made.

Can I assign a role to all users of the site at once?

If you are a site administrator in CollabNet TeamForge, you can assign a role to all the system users (non-site administrators) at one go.

Now, CollabNet TeamForge provides a system created user group called "All Users", including all site users. All role assignments that can be made to a user group can be made even for the "All Users" user group.

> **NOTE:** The `All Users` group members can not be manually added, edited, viewed or deleted. This group is automatically updated each time a user is added or deleted from the site.

Can I request a role?

If you are a project member in CollabNet TeamForge, you can request a role in your project.

You can use the **Project Home > My Roles** page to request for a role. Your request is submitted to the project administrator for approval. You will receive an email notification when your request is either approved or denied.

> **NOTE:** Based on your project administrator's discretion, some role requests may be granted immediately, while the other role requests may need approval.

Why don’t I have access to the “Reported in Release” information in my artifact?

If you cannot see the “Reported in Release” information in your artifact, ask your project owner to add the role required to obtain access to this information.

Users with tracker artifact submit/view/edit permissions will not be able to see “Reported in Release” information in an artifact, as it is directly associated with the File Releases section.

You can request that the project owner add a role permission of View Only for All Packages in File releases for you to obtain access.

Who can access an application?

Application permissions help you minimize the need to create and assign many similar roles for individual users. Instead, you can permit or restrict access to individual applications within the project for whole classes of users.

Application permissions supplement role-based access control (RBAC.) For each application’s concepts, documents, file releases, trackers, and discussion forums, you can assign permissions globally based on user type.

For example, if you know that you want all project members to be able to view and submit to all project trackers, you can specify this application permissions. You need to configure these settings only once. All current and future project members will be able to view and submit to all trackers without having the tracker view/submit permission assigned to them individually via a role.

Before you do this, you should have identified your project as private, gated community, or public. Configuring permissions is a finer-grained level of control that operates within this hierarchy of project types.

Some applications may be invisible to some users based on the roles you assign. If you give a user a role that does not grant access to a particular application, that user cannot see the button for that application in the Web interface. (However, if that user also has some other role that does grant access to that application, the user can see the application button.

> **NOTE:** A user’s license type also influences what the user can see and do on your site. A user’s license type supersedes any role assignments. Ask your site administrator how many licenses of each kind are available for your users. For more information, see How do TeamForge Licenses Work?.

Exceptions

- If a user has an SCM license, that user can see only the tools that support the core source control functions of the site. The Tracker, Documents, and Tasks tools are not visible.

- If a user has any level of permission on a tracker or a task folder, that user can see the **Reports** button.
- If a user has tracker administration permission on any tracker, or task administration in any task folder, that user can see the **Project Admin** button.

User Classes

These are the classes of users to which you can assign application permissions:

User Class	Description
All users with Role Permissions	Only project members with appropriate RBAC permissions.
All project members	All project members.
All project members and unrestricted users	All unrestricted users, whether or not they are project members, plus all project members.
All logged-in users	All restricted and unrestricted users (all logged-in users,) whether or not they are project members.
All users	All users, whether or not they are logged in or have CollabNet TeamForge accounts.

Restrictions by type of site

On some types of sites, you can’t assign application permissions to certain classes of users. In such cases, you must use role-based access control (RBAC) permissions.

- On a private site, you cannot set application permissions for these classes of users:
  - All project members and unrestricted users
  - All logged-in users
  - All users
- On a “Gated Community” site, you cannot set application permissions for these classes of users:
  - All logged-in users
  - All users

Can I delete an item if I have ‘Administer’ permission?

No. You cannot delete an item if you have the administer permission.

The administer permission does not allow you to delete an item. You can create, edit, submit, and view items if you have the administer permission. For deletion, you must have the delete permission.

Can I disable creation of user accounts?

As of TeamForge 4.1 SP3, it is possible to disable the creation of new accounts by users so that only a ‘site admin’ can create new users.

To enable this mode of operations, simply add the following line to `/opt/collabnet/teamforge/sourceforge_home/etc/sourceforge_configuration.properties`:

```
sf.disableUserSelfCreation=true
```

Once this line is in place, restart TeamForge for it to take effect.

Why do I see the project home page though I lack the necessary permission?

This has been set so that when users are provided access to a project, they do not see an empty page.

Irrespective of whether the view permission on project pages is checked, users will be able to see the contents in the project home page.

Why do you need the authentication and authorization plugin for Hudson?

The Authentication and Authorization plugin allows you to set up your Hudson installation to authenticate against a CollabNet server and specify access control for CollabNet users.

> **NOTE:** Authentication and authorization are independent actions. You could set up your Hudson installation to authenticate against a SourceForge or TeamForge site, but not use that CollabNet site for authorizing users. Authorization is available only with CollabNet TeamForge 5.2, but authentication is possible with earlier CollabNet SourceForge Enterprise versions as well.

Authentication

Authentication determines user names and passwords. You establish user credentials when you enable your Hudson site to use the CollabNet security realm.

Authorization

Authorization determines what users can do on the Hudson site.

Your Hudson server can be shared between many CollabNet projects, and you can grant TeamForge users permissions at the site level. You specify site-level permissions when you configure the site to use a CollabNet server for authorization.

A job on your Hudson server may be involved with more than one TeamForge project. For example, a job that builds software can pull in source code from multiple project repositories. For the purpose of authorization, however, each job is associated with one CollabNet project, and you can give users project-level roles.

When your Hudson site is set up to use CollabNet authorization, TeamForge project administrators can assign these roles to project members:

- Hudson Build/Cancel
- Hudson Configure
- Hudson Delete
- Hudson Promote
- Hudson Read

Users get the highest permissions they are entitled to. For example, if a user is part of a group that has administration permissions for the Hudson site, that supersedes any Hudson-related role the user might be assigned within a specific CollabNet project.

Can I control user access to an integrated application?

TeamForge can integrate the permissions scheme of a separate application into the TeamForge role-based access control system.

To look at how this works, we’ll use the Pebble blogging tool as an example. Pebble is an application that you can quickly integrate with TeamForge.

Pebble brings with it a set of pre-determined roles that you can assign to project users. The roles are defined in the XML application configuration file.

Blog Reader

You can only read blogs and make comments, the comments are sent for moderation.

Blog contributor

You can add blog posts, but they will be sent for moderation.

Blog publisher

You can add blog posts, moderate comments and blog posts.

Blog owner

You can do all that a Blog publisher does as well as change the blog properties and security options.

Any site user with one or more of these roles can see the **Pebble Blog** button in their project toolbar. Clicking that button allows them to operate Pebble according to their access rights.

> **NOTE:** Note: Site Administrators don’t need any specific permissions; they have all permissions on all projects on the site.

How does inheritance work?

Site or project administrators can create hierarchical relationships between projects so that one project can inherit members, roles and permissions from a parent project.

When you define users, user groups and roles with specific permissions in one project, they can be inherited in one or more subprojects. This helps you avoid duplicating the effort of defining users, user groups and roles across projects.

You can still, if required, create roles specifically for this project and add direct members to any project. The direct members can be assigned inherited roles. The inherited members can be made direct members and/or assigned direct/inherited roles too.

While inheriting roles, only the permissions associated with all (top-level) folders are inherited.

In a subproject, you can select the inherited project members from the **Assigned to** lists or from the user picker, as the case may be. Inherited users may not be part of these lists if their role inheritance was prevented in the parent project.

> **NOTE:** At the time of role creation, you can choose to allow or disallow the role inheritance into private subprojects.

How does inheritance work for project groups?

Site or project administrators can create project groups and add member projects to the group, to manage several projects as a single unit.

Moreover, hierarchical relationships may exist between projects so that one project can inherit members, roles and permissions from its parent project.

Quite like a project can inherit roles and permissions from its parent project; projects can inherit only permissions via the project group.

Project Groups permissions when granted to users via site-wide roles, also allows the users to access project groups in all cases. Prevent inheritance option for roles does not apply in those cases.

The project in which a user gets permissions through role assignment in project groups appears on the **My Workspace > Projects > All Projects** page.

The roles created specifically for a project group are not made available as inherited roles in the group's member projects for assignment. Project group roles can only be assigned directly in the project group context. These roles are not requestable.

Can I create new user accounts as "unrestricted"?

You can make all new accounts be "unrestricted" by default instead of the more secure "restricted."

This is controlled by the *USER_ACCOUNT_RESTRICTED* variable in the `site-options.conf` file. Change that value to `false` and restart runtime.

> **NOTE:** This does not change any existing accounts. If someone was restricted before this flag was turned on, they remain restricted until the site admin edits their account.

Why can't Oracle connect to my TeamForge installation?

The simplest way to correct this is to overwrite the `.jar` included with TeamForge with the one from $ORACLE_HOME.

TeamForge uses the thick Oracle JDBC driver, which has two parts. One of these is provided by TeamForge, the other is in $ORACLE_HOME. If these two components are incompatible, TeamForge will be unable to make a connection to the database.

Follow these steps to overwrite the `.jar` included with TeamForge with the one from $ORACLE_HOME:

```
cp $ORACLE_HOME/jdbc/lib/ojdbc14.jar
        /opt/collabnet/teamforge/jboss/jboss-3.2.6/server/default/lib/
```

A restart of the application will be required to use the `new.jar`.

Why can’t I move these artifacts into this planning folder?

To avoid confusion, you must observe a few basic rules when you assign artifacts to planning folders.

When an artifact in a planning folder has a child artifact, the child artifact can only be assigned to the same planning folder as the parent artifact, or a sub-folder of that folder. For example:

- Before you assign an artifact to a planning folder, make sure the artifact and all its children are assigned to a single planning folder and its sub-folders.
- Don’t assign an artifact to a planning folder that is above the artifact’s parent artifact in the planning folder hierarchy.
- Before promoting a planning folder to a higher level in the hierarchy, make sure its member artifacts (and its members’ parent artifacts) are all assigned to the same planning folder or a sub-folder of it.

An artifact can only be assigned to a planning folder in its own project.

As a best practice, consider instead creating a matching artifact in this project and associate the two artifacts.

Before you assign an artifact to a planning folder, be sure you have permission to edit every artifact affected by the move.

Who can access a project?

You control access to your CollabNet TeamForge project by a combination of project settings, membership rules and user restrictions.

Who should I allow access into my project?

To decide how to control access to your project, think about what the project is for and who will be using it.

Consider these elements:

- The project’s access setting.

A project can be public, private, or gated community.

> **NOTE:** The site administrator can change the default project access permissions.

- The project's membership.
- Each site user's user type.
- Each site user's license type.
- Each TeamForge user's user type.
- Any parent projects from which members, user groups or roles are inherited.
- Any subprojects that inherit members, user groups or roles from this project.

User type

Users can be restricted or unrestricted.

- Restricted users can access only public projects and projects of which they are members.
- Unrestricted users can access all projects except private projects of which they are not members.

License type

Users can have an ALM license or an SCM license.

- An ALM license enables the user who holds it to use the full range of TeamForge features: both the core source-code management tools and the extended application life cycle management functionality.
- An SCM license enables the user who holds it to use the core TeamForge source-code management tools.

License type supersedes user type. For example, if you give a user an SCM license, and then declare that user an unrestricted user, the user can see only the core source code management tools in any project they can access.

Project access setting

A project can be private, gated, or public.

- **Private** - Private projects can only be accessed by project members. Private projects do not appear on the Home page, in the **All Projects** list, or in search or reporting results to users who are not project members.

  Create a private project when you want to strictly limit project access.

- **Gated** - Gated community projects can be accessed by project members and by unrestricted users. As with private projects, gated projects are not visible to users who are not allowed to access them.

  Create a gated community project when you want to exclude restricted users, but do not need to exclude other, unrestricted users. For example, your organization might wish to designate all contractors or outsourced staff as restricted users. They will not be able to see any gated community projects, but all of your full-time, regular staff will have access.

- **Public** - Public projects can be accessed by all users.

  Create a public project when you have no need to restrict access.

  360:The following table shows which user types can access projects with each project access setting.

> **NOTE:** Project access is not the same as project membership. Project access allows a user to see the project in the **All Projects** list, visit the project home page, and browse selected project data. A restricted user may be able to access a project without being a project member.

This table shows which user types can access projects with each project access setting.

Project access type	Project member (Unrestriced user)	Project member (Restricted user)	Non-project member (Unrestricted user)	Non-project member (Restricted user)
Private	Yes	Yes	No	No
Gated	Yes	Yes	Yes	No
Public	Yes	Yes	Yes	Yes

How do I give read-only anonymous access to cvs/svn repository?

Set the default access permissions of the project to public and allow Source Code to view permission for all or specific repositories to All users while restricting other permissions.

To give read-only anonymous access to cvs/svn repository within a project while still restricting write access, you could set the default access permissions of the project to public with Source Code view permission open to all users, while restricting other permissions to specific user classes.

The steps to set this up are as follows:

1. Click **Project Admin** from **Project Home** menu.
2. Click **Permissions** in the left navigation menu.
3. Click on *Default Access Permissions* tab.
4. Set **Project Access Permissions** as *Public*.
5. Under **Application Permissions**, choose **All users** from the drop-down for Source Code View permission.
6. For other application permissions, choose a user class based on your access requirement.

> **NOTE:** In Step 5 above, it is possible for you to choose whether you want to give View access to All users for all repositories or a specific repository.

Can I ensure that only site admins can create projects?

You can do this using Velocity.

Create `/opt/collabnet/teamforge/sourceforge_home/templates/body_footer.vm` and populate it with this:

```
#set($superUser = ${PAGE_INFO.isSuperUser()})
#if($superUser == false)
<script>
if(document.getElementById("createProject") != null){
var e = document.getElementById("createProject").parentNode.parentNode;
while (e.firstchild) {
e.removeChild(e.firstChild);
}
}
</script>
#end
```

Save the file, and TeamForge will start hiding the button for non-site admins immediately.

How do user roles work?

Project administrators can define specific access permissions for individual project members. They do this by using global project roles and/or creating roles and assigning the roles to project members.

Under role-based access control (RBAC), permissions are not assigned directly to an individual user. Instead, each user has the permissions that are attached to any role that is assigned to that user.

A project member can be assigned multiple roles.

In CollabNet TeamForge, site or project administrators assign roles to the site users or project members. Besides this, a project member can submit a role request to the project administrator. The project administrator can approve or reject such requests.

When you define users, user groups and roles with specific permissions in one project, they can be inherited in one or more subprojects. This helps you avoid duplicating the effort of defining users, user groups and roles across projects.

> **NOTE:** Permissions are cumulative. If a project member is assigned a role that provides a specific permission, and another role that does not, the user has that permission.

A role defines these things:

- The applications that project members with that role can and cannot access.
- The resources on which project members with that role can use the applications.
- The actions that project members can take in each application and on each resource.

> **NOTE:** If a user has an SCM license, that user can see only the tools that support the core source control functions of the site, even if the user has a role that would otherwise grant access to other resources.

When a user's roles do not include access to an application or resource, that application or resource is not visible to that user. For example, imagine that you are assigning roles to Jason, a software developer. Jason needs to check source code in and out in order to fix bugs, develop features and create software releases. However, Jason does not need access to the wiki. If you set up Jason's roles according to those requirements, Jason's experience is like this:

- On any page in the project site, Jason can see and click the **Trackers**, **Source Code**, and **File Releases** buttons along the top of his screen.
- Jason does not see the **Wiki** button.
- If someone sends Jason a link to a page in the Wiki application and Jason clicks the link, he gets an error message. (The message does not specify whether the page exists or not.)

- When he accesses the project directly from Eclipse or Visual Studio, Jason can expand the project node and browse the **Trackers**, **Source Code**, and **File Releases** nodes, but not the **Wiki** node.

  > **NOTE:** A user's license type also influences what the user can see and do on your site. A user's license type supersedes any role assignments. Ask your site administrator how many licenses of

> each kind are available for your users. For more information, see How do TeamForge licenses work?.

Applications

An application is a collection of related features designed to enable a user to collaborate on particular kinds of tasks. For example, the Documents application helps users create documents, share in document reviews, and publish documents, among other things.

In the Web interface, each application is represented by a button in the navigation bar at the top of any project page. A given user can see the buttons corresponding to applications they have access to by virtue of the roles assigned to them.

Applications are also known as “tools.”

Resources

- The tracker application might contain a bugs tracker and a feature request tracker. These are the tracker resources.
- A project can contain multiple SCM repositories. These are the SCM resources.

Permissions

View

Allows users to view and download items, but not to create or edit items, administer folders, or edit application settings.

Create or Submit

Allows users to create new items, but not to edit items, administer folders, or edit application settings. Users with the create or submit permission also have the view permission.

Edit

Allows users to edit items, but not to administer folders or edit application settings. Users with the edit permission also have the view permissions.

Administer

Allows users to create and edit items, administer folders, and edit application settings. Users with the administer permission also have the edit, create or submit, and view permissions. To delete items, the user needs to have the delete permission.

Delete

Allows users to delete items, but not to administer folders or edit application settings. Users with the delete permission also have view permissions. Without the delete permission, users with the administer permission are not allowed to delete items.

How does CollabNet TeamForge help protect data access?

Access to data must be strictly controlled to meet the security requirements of the enterprise. Strict data access control is achieved through a combination of firewalls, authentication, and authorization.

Firewalls and Network Configuration

A firewall provides the first level of protection by restricting access to the private network from the Internet. Sophisticated firewall configuration can provide strong security for all enterprise resources.

All CollabNet TeamForge application server nor the backend servers should ever be exposed to the Internet.

The CollabNet TeamForge application to function effectively, the following conditions must be met.

- Across the firewall, clients (users) must have access to:
    - The web server through a secure protocol such as HTTPS (port 443). The web server typically handles both the browser requests as well as the SOAP requests and forwards them to the CollabNet TeamForge application server.
    - Send mail to CollabNet TeamForge mail server via SMTP (port 25).
    - The SCM server through a secure protocol such as SSH (port 22).
- The web server must have access to the application server (typically port 8080).

> **NOTE:** This port is not exposed outside the firewall.

- The web server must have access to the SCM server for repository browsing functionality.
- The application server must have access to the backend (SCM, database, mail, etc.) servers.
- The SCM server must be able to access CollabNet TeamForge for commit notifications.
- The mail server must be able to deliver messages across the firewall.

Authentication and Authorization

To secure sensitive data, CollabNet TeamForge provides access control tools to restrict unauthenticated and non-member access.

User authentication is supported through verification of username and password during login. Project administrators can completely restrict access to authenticated members by marking projects as gated communities or private. A gated community is only accessible to unrestricted users, while a private project is only accessible to its members.

How does CollabNet TeamForge help protect my data?

Sensitive data must be protected from illegal access at various points in the system. Key areas where security is typically compromised include data transmission and data storage.

Data Transmission

Network traffic is not encrypted by default. The HTTP protocol (non-SSL) does not protect data during transmission. HTTPS provides Strong Encryption using the Secure Socket Layer and Transport Layer Security protocols (SSL/TLS).

> **NOTE:** The web server employed by a CollabNet TeamForge installation must be reconfigured to employ the HTTPS protocol.

Data Storage

Sensitive data, such as credit card numbers, financial information, etc., must be stored securely. Usually this is done by encryption. In the context of an application, like CollabNet TeamForge, passwords are stored as cryptographic hash, using SHA family of algorithms, to guarantee adequate data protection.

SHA-512 message digest algorithm defines a one-way hash function that is used to maintain data integrity through creation of a digest from data input. The one-way hash function is designed in such a way that it is hard to reverse the process, that is, to find a string that hashes to a given value. SHA-512 is currently a standard, Internet Engineering Task Force (IETF) Request for Comments (RFC) 6234. According to the standard, it is “computationally infeasible” that any two messages that have been input to the SHA-512 algorithm could have as the output the same message digest, or that a false message could be created through apprehension of the message digest.

How do site administrator roles work?

Site administrators are default site managers who can create additional site administrators and delegate few site administrative tasks to them.

They can also allow some CollabNet TeamForge users to use one or more CollabNet TeamForge tools across several projects, by creating site-wide roles with specific project permissions, minus site administrative permissions. They can also provide ready-to-use roles as global project roles, creating uniformity across the site.

In CollabNet TeamForge, site or project administrators assign roles to the site users or project members. Besides this, a project member can submit a role request to the project administrator. The project administrator can approve or reject such requests.

A role defines these things:

- The applications that project members with that role can and cannot access.
- The resources on which project members with that role can use the applications.
- The actions that project members can take in each application and on each resource.

When a user's roles do not include access to an application or resource, that application or resource is not visible to that user. For example, imagine that you are assigning roles to Jason, a software developer. Jason needs to check source code in and out in order to fix bugs, develop features and create software releases. However, Jason does not need access to project wiki. If you set up Jason's roles according to those requirements, Jason's experience is like this:

- On any page in the project site, Jason can see and click the **Trackers**, **Source Code**, and **File Releases** buttons along the top of his screen.
- Jason does not see the **Wiki** button.
- If someone sends Jason a link to a page in the Wiki application and Jason clicks the link, he gets an error message. (The message does not specify whether the page exists or not.)
- When he accesses the project directly from Eclipse or Visual Studio, Jason can expand the project node and browse the **Trackers**, **Source Code**, and **File Releases** nodes, but not the **Wiki** node.

Applications

An application is a collection of related features designed to enable a user to perform tasks and collaborate with other users. For example, the Documents application helps users create documents, share in document reviews, and publish documents, among other things.

In the Web interface, each application is represented by a button in the navigation bar at the top of any project page. A given user can see the buttons corresponding to applications they have access to by virtue of the roles assigned to them.

Applications are also known as "tools."

Resources

- The tracker application might contain a bugs tracker and a feature request tracker. These are the tracker resources.
- A project can contain multiple SCM repositories. These are the SCM resources.

Site Administration Responsibilities

The additional site administrators can be granted administrative rights for any of the site administrator responsibilities related to the following:

- Projects (includes project templates)
- Project Groups
- Users
- Groups
- Roles
- Categories
- System Tools
- Integrated Applications

Site Administration Permissions

View Only

Allows users to view and download items, but not to create or edit items, administer or edit application settings.

Create or Submit

Allows users to create or submit and edit items, but not to administer or edit application settings. Users with the create or submit permission also have the edit and view permissions.

Edit

Allows users to edit items, but not to administer items or edit application settings. Users with the edit permission also have the create / submit and view permissions.

Administer

Allows users to create, edit and administer items plus edit application settings, if required. Users with the administer permission also have the edit, create or submit, and view permissions. To delete items, the user needs to have the delete permission.

Delete

Allows users to delete items, but not to administer items or edit application settings. Users with the delete permission also have view permissions. Without the delete permission, users with the administer permission are not allowed to delete items.

Which role is assigned to me?

There are several ways in which you could have been assigned certain roles in CollabNet TeamForge. Your access to the projects depends largely on the permissions granted to you via your roles.

You could have been granted access to any CollabNet TeamForge project in any of the following ways:

- Through site-wide roles (assigned directly to the user)
- Through global project roles (assigned directly or inherited via another project, either individually or as a group member)
- Through project roles (assigned directly or inherited via another project, either individually or as a group member)
- Through permission inheritance (via project hierarchy)

There are two ways you can check which roles are assigned to you in any of your projects:

> **NOTE:** If you are a site administrator, you can view any user's access rights to the system.

To view the roles assigned directly to you:

Use the **My Workspace > My Page > Projects** page and click the *My Projects* tab.

> **NOTE:** You can click the role to view the folder-level permissions assigned to you in a project.

To view all the roles assigned to you:

Use the **My Workspace > My Page > My Settings** page and click the *Roles* tab.

You can select `Roles Created For a Project`, `Roles Inherited From a Parent Project` or `Site-wide roles` to view the corresponding roles assigned to you.

> **NOTE:** A global project role that has been assigned to you in a project where you are a direct member will be listed as a directly assigned role.

> **NOTE:** You can click the role to view the folder-level permissions assigned to you.

Related Links

- Control Project Access - Create a Role Based Access Control (RBAC)

Who can access project planning information?

A project manager should consider carefully who is able to change the project plan.

You can specify the people who can change the product plan based on their roles in the project. Only users with the role you select can view, create, modify, and delete artifacts in the planning folder hierarchy, and populate those artifacts with content.

Every project must strike its own unique balance between openness and control. These broad guidelines may help you decide how to set up the appropriate roles for your project.

- Err on the side of visibility. Most of the time you will want all project members to at least see the contents of all planning folders. This helps developers avoid duplicating their efforts, and it can enable team members to volunteer their help to each other when appropriate.

  However, occasionally you may want to hide all or part of the planning folder hierarchy from selected people. For example, when you bring on a contract developer for a short period, you may want to restrict that person's view to the specific artifacts they are working on directly.

- Limit the number of people who can modify or populate planning folders. In many cases, only the project manager really needs to do this.

  However, it may make sense to allow some people with a direct stake in the project to modify some folders. For example, if you are working with a product owner, it might be useful for that person to be assigning artifacts to future planning folders while your team is working on the current one.

Who can see a project page?

A user can see a given project page if the page is not hidden and the user's role includes permission to see the page.

> **NOTE:** When a page is hidden, it is only visible to users with the Project Admin role, and then only when the user has clicked **Configure: On**.

If a person can see your project, they can see the project home page and any of its subpages. To see any other page, they must be assigned permission explicitly.

Permissions determine two kinds of access:

- Anyone with access to the project can see the project homepage.
- Users whose role includes permission to see a given top-level project page can see all the subpages of that page.

Subpages inherit their permission setting from the top level page they belong to.

Who can see a project page component?

To see a given project page component, you must have permission to see both the project page where the component is and the tool that the component represents.

Your ability to see and edit the contents of a project page component is controlled by your permissions with regard to the tool the component represents.

For example:

- If you have permission to create documents in a selected folder in the Document Manager tool, you have that same permission when viewing that folder in a Documents component on a project page.
- If you do not have permission to view a folder in the Document Manager tool, you cannot see a document component that contains that folder on a project page.

> **NOTE:** Users with the Project Admin role can see project page components even if they do not have access to the related tool.

As a project admin, why don’t I have permissions to the wiki?

Most likely, your site was installed before TeamForge contained the wiki component.

When the wiki was added to TeamForge, it was decided that there was no way for us to know the security requirements at customer sites, so permissions for the new wiki component were not assigned to project admins by default. As a project admin, you can alter any existing role to grant this permission, or create a new role for this permission and then assign to the appropriate project members as needed.

These are some of the frequently asked questions on Associations in TeamForge.

My Associations do not appear. What should I do?

You may not be seeing your assocations for several reasons. If you are not able to see your associations, verify the following:

- Only repositories scoped to the configured TeamForge project will result in associations. That is, make sure the version control repository you're using is a part of the TeamForge project that has been mapped to the JIRA project in question.
- When a TeamForge project is first mapped to a JIRA project, the JIRA issues need to be synced into the TeamForge data store. Association will fail if you attempt to create an association between a commit and JIRA issue that has not yet been synced. Please refer to the Sync Issues functionality.
- Make sure the association syntax is correctly using square brackets. Association syntax is case sensitive.

Can I associate objects of different projects?

Yes, you can associate an object (for example from document to subversion commit) in one project to an object in another project if you have access or are a Site Admin.

You must have permission to view the object that you're associating with before you can associate with that object, unless you are a Site Admin.

These are some of the frequently asked questions on TeamForge database, datamart, ETL, PostgreSQL, Oracle and so on.

What are the right PostgreSQL settings for my site?

Your site's PostgreSQL settings depend on the conditions your site is operating under, especially the number and size of projects and the number of users.

The default values in the `site-options.conf` file are designed for a TeamForge site running on a system with 8GB of RAM. This table contains recommended values for systems with various amounts of RAM, based on testing carried out in CollabNet's performance lab. Use your discretion in selecting the right values for your environment.

> **IMPORTANT:** You must recreate the runtime environment after changing any value in the `site-options.conf` file.

Recommended values if PostgreSQL and TeamForge are on the same server

site-options.conf Tokens	8GB RAM	16GB RAM	32GB RAM	64GB RAM	128GB RAM
PGSQL_EFFECTIVE_CACHE_SIZE=	4GB	6GB	12GB	24GB	48GB
PGSQL_SHARED_BUFFERS=	1GB	2GB	4GB	8GB	8GB
PGSQL_WORK_MEM=	64MB	64MB	64MB	64MB	64MB
PGSQL_WAL_BUFFERS=	16MB	32MB	32MB	32MB	32MB
PGSQL_MAINTENANCE_WORK_MEM=	256MB	615MB	615MB	615MB	615MB

Recommended values if PostgreSQL is on a separate server

site-options.conf Tokens	8GB RAM	16GB RAM	32GB RAM	64GB RAM	128GB RAM
PGSQL_EFFECTIVE_CACHE_SIZE=	6GB	12GB	24GB	48GB	96GB
PGSQL_SHARED_BUFFERS=	2GB	4GB	8GB	8GB	8GB
PGSQL_WORK_MEM=	64MB	64MB	64MB	64MB	64MB
PGSQL_WAL_BUFFERS=	16MB	32MB	32MB	32MB	32MB
PGSQL_MAINTENANCE_WORK_MEM=	256MB	615MB	615MB	615MB	615MB

Why do ETL jobs fail post TeamForge upgrade?

ETL jobs can fail due to reasons such as incompatibility between the database and JDBC driver versions and ETL jobs not being able to connect to the Datamart. Try the following solutions.

Pentaho, used by TeamForge for data integration and transformation jobs, recommends using compatible JDBC drivers meant for specific database versions. See Pentaho's JDBC Drivers Reference for more information.

If ETL jobs fail post TeamForge upgrade due to incompatibility between the database and JDBC driver versions:

1. Refer to Pentaho's JDBC Drivers Reference page.

2. Click the JDBC driver reference URL corresponding to your database, Oracle or PostgreSQL.
3. Identify and download the compatible JDBC driver for your database.
4. Replace the JDBC driver found in the following directories with the one you downloaded. (The TeamForge ETL process refers to the JDBC driver available in these directories.)
   - `/opt/collabnet/teamforge/dist/tomcat/commonlib/`
   - `/opt/collabnet/teamforge/runtime/tomcat_etl/webapps/etl/WEB-INF/lib`

> **NOTE:** You can also refer to this page for more information about Pentaho-special database issues and resolutions.

If ETL jobs fail due to unavailable connections to the PostgreSQL Datamart:

Make sure that the following error message is found in `etl.log`

```
Invalid JNDI connection java:comp/env/jdbc/ReportsDS : FATAL: remaining connec
tion slots are reserved for non-replication superuser connections
```

If yes, restart the ETL service and restart the failed ETL jobs manually using `./etl-client.py` script in the `/opt/collabnet/teamforge/runtime/scripts/` directory. The ETL jobs should be able to connect to the PostgreSQL Datamart after the restart.

> **NOTE:** If the problem persists even after restarting, contact CollabNet Support.

What is ETL initial load job and when to run it?

You can run the initial load job any time after the site is upgraded to TeamForge 18.1. We recommend that you run it before you hand over the site to the users.

Why am I not able to see the charts for tracker metrics?

You may not be able to see the charts for the tracker metrics if the tracker initial load is not running correctly.

The incremental data collection is disabled until the initial load is run. You can check if the initial load is completed successfully by executing the query below from the Ad-hoc reporting page against the datamart.

```
select status from etl_job where job_name='tracker_initial_etl';
```

You must get the status value as 1 if the initial load is completed successfully. Otherwise, you must trigger the job manually by executing the command:

```
[RUNTIME_DIR]/scripts/etl-client.py -r TrackerInitialJob
```

Why am I getting a ‘Not running’ message when the Datamart service is stopped?

When TeamForge and Datamart are running in a single instance, the TeamForge database is stopped when you stop the Postgres services. The message, ‘Not running’ is displayed when you stop the Datamart service. You can ignore this message.

How do I enable the Postgresql log files archiving when the services are not started using the CollabNet startup script?

The Postgresql log files archiving can be enabled by running a simple command.

It is recommended you use the teamforge script to start and stop the Postgres services. However, if you use the Postgresql init script to start or stop the Postgres services, the postgresql log files are not archived by default.

To enable the Postgresql log files archiving, run the following command:

```
teamforge -s ctf-postgresql-9.6 start
```

Why am I getting an email specifying that the ETL job has failed?

You are getting this email because one of the Extract Transformation and Load (ETL) jobs has failed during the run.

You can see the `etl.log` for more details to find out the reason for the job failure.

The ETL job failure may happen because of the following reasons:

- Out of memory error.
- No response from the database.

If the ETL job failure is happening for the first time, you can restart the ETL (`[RUNTIME_DIR]/scripts/collabnet restart etl`) and check if the problem is occurring again. You can increase the JVM heap size by specifying the same in ETL_JAVA_OPTS if the problem keeps recurring. The default value is `-Xms160m -Xmx256m`. You can increase the heap size depending on the memory available in the box.

Check if both TeamForge and Datamart are up and responding to queries if there is no response from the database. Restart the ETL ([RUNTIME_DIR]/scripts/collabnet restart etl).

Contact CollabNet support if the problem persists.

What does the “psql: could not connect to the server: No such file or directory” error message mean?

This error indicates that the PostgreSQL database server is not running. You need to restart the server.

Use the following command to start the server:

```
service postgresql start
```

Error while modifying custom reports via branding repository. How to fix?

When you create a custom report, change its category via the branding repository and then attempt to view the original report that you created (before modifying the category), you will encounter an error.

Run the following query to update the report:

```
update report set category=<category_name> where id = <report_id> and surrogat
e_id = <surrogate_id>;
```

In the above query category_name is the ID from the report_category table, which you can get by running the following query:

```
select id from report_category;
```

Ensure that the report type in the report belongs to the new category using the report_type_meta_data table by running the following query:

```
select type, category from report_type_meta_data;
```

Why do I get performance issues when retrieving flex field information from Datamart?

Flex fields are stored in XML format in the flex fields table. Due to the complexity involved in parsing the XML data and retrieving the relevant flex field information, you can see some performance issues. To overcome this, the XML to non-XML conversion based solution or feature has been implemented to store the flex fields in non-XML format. This approach has redefined the flex field storage and retrieval mechanism and thus has improved the performance of reporting queries while retrieving flex field information from Datamart.

How to enable or disable this feature in Datamart?

To enable or disable this feature respectively, update the field “attribute_value” to either true or false for the record where entity_name is ‘flex_field_xml_to_nonxml_etl’ and attribute_name is ‘ALLOW_FLEXFIELD_XML_TO_NONXML’ in ETL Attributes table.

```
update etl_attributes set attribute_value='true' where entity_name = 'flex_field
_xml_to_nonxml_etl'
and attribute_name = 'ALLOW_FLEXFIELD_XML_TO_NONXML';
```

Does a new job have to be executed to get the benefits of this feature?

No, the existing tracker initial job and tracker incremental job will do the required data processing.

How are invalid XML records handled during XML to non-XML conversion?

Earlier, if any corrupted XML records were found being stored during the XML to Non-XML conversion, the administrators need to manually correct them and store the corrected data into the new flex field bridge tables. To save this manual effort, an automated healing approach was implemented. This auto healing process rectifies the invalid XML records from the audit tables, populates the new flex field bridge tables with the rectified records, and the status of these rectified records are marked as `processed` in their tables of origin.

What are the changes related to custom reports development on flex fields?

The custom report developers need to query the new flex field bridge tables to get the benefits from the XML to non-XML conversion feature implementation.

How to handle the ETL job failure due to OutofMemoryError: GC overhead limit exceeded?

For a permanent fix, see Why am I getting an email specifying that the ETL job has failed?

To fix this issue during runtime, perform these steps:

1. Stop the ETL service.

   ```
   kill -9 <etl process id>
   ```

2. Open the file `/opt/collabnet/teamforge/runtime/conf/set-env.sh`.

3. Increase the JVM heap size in *ETL_JAVA_OPTS* to `Xms512` and `Xmx2048m`.

   ```
   ETL_JVM_OPTS="-Xms512m -Xmx2048m -server -XX:+HeapDumpOnOutOfMemoryError -
   XX:HeapDumpPath=/tmp -verbose:gc -XX:+PrintGCTimeStamps -XX:+PrintGCDetail
   s -
   Dsun.rmi.dgc.client.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=6000
   00 -
   Djava.awt.headless=true -
   Dsourceforge.home=/opt/collabnet/teamforge/runtime/sourceforge_home -
   Dsourceforge.logdir=/opt/collabnet/teamforge/log/etl -
   Dapp.data=/opt/collabnet/teamforge/var -
   Dapp.runtime=/opt/collabnet/teamforge/runtime -
   Dapp.distribution=/opt/collabnet/teamforge/dist -
   Dapp.log=/opt/collabnet/teamforge/log"
   ```

4. Start the ETL service.

5. Run this command to make sure that you see the updated memory setting in the command output.

   ```
   ps -ef | grep etl | grep -v jboss
   ```

6. Run the ETL job again.

How to redo the entire XML to non-XML conversion if required for some reason?

Workaround 1: Bootstrap the reporting service and execute the tracker initial job. As XML to non-XML conversion is part of initial job as well, this would redo the conversion process.

Workaround 2: Truncate the relevant tables and reset the previously processed artifact flex field key to 0. This would reinitiate the XML to non-XML conversion while executing the tracker incremental job.

Relevant SQL process:

```
truncate table stage_flex_fields_bridge;
truncate table text_flex_fields_bridge;
truncate table user_flex_fields_bridge;
truncate table ms_flex_fields_bridge;
truncate table ss_flex_fields_bridge;
truncate table date_flex_fields_bridge;
truncate table bad_xml_records;

update etl_attributes set attribute_value = 0 where attribute_name = 'PREV_RUN
_ARTIFACT_FLEX_FIELDS_KEY' ;

commit;
```

What kind of objects can I create reports on?

Reports can report on data from trackers, tasks, planning folders, and repositories.

Task reports can report on data in a single project or across multiple projects. Tracker reports can report on data in a single tracker or across multiple trackers in a project or multiple projects. Task and Tracker reports are in tabular format and grouped under Table reports. You can also create reports on data in planning folders and on data in source code repositories.

How do I change the time to run the ETL jobs?

The *ETL_JOB_TRIGGER_TIME* can be modified to specify a different time.

By default, the ETL job runs at 2:30 AM (local time) everyday. It is recommended to run this once a day to avoid any performance degradation of the Teamforge site. See [ETL_JOB_TRIGGER_TIME][siteoptiontokens.html#etljobtriggertime] for more information.

How can I check the status of ETL?

The following command displays the status of the ETL process.

```
[RUNTIME_DIR]/scripts/collabnet status etl
```

You can get additional information about the various ETL jobs that are configured using the command:

```
[RUNTIME_DIR]/scripts/etl-client.py -a
```

What can I learn from a burndown chart?

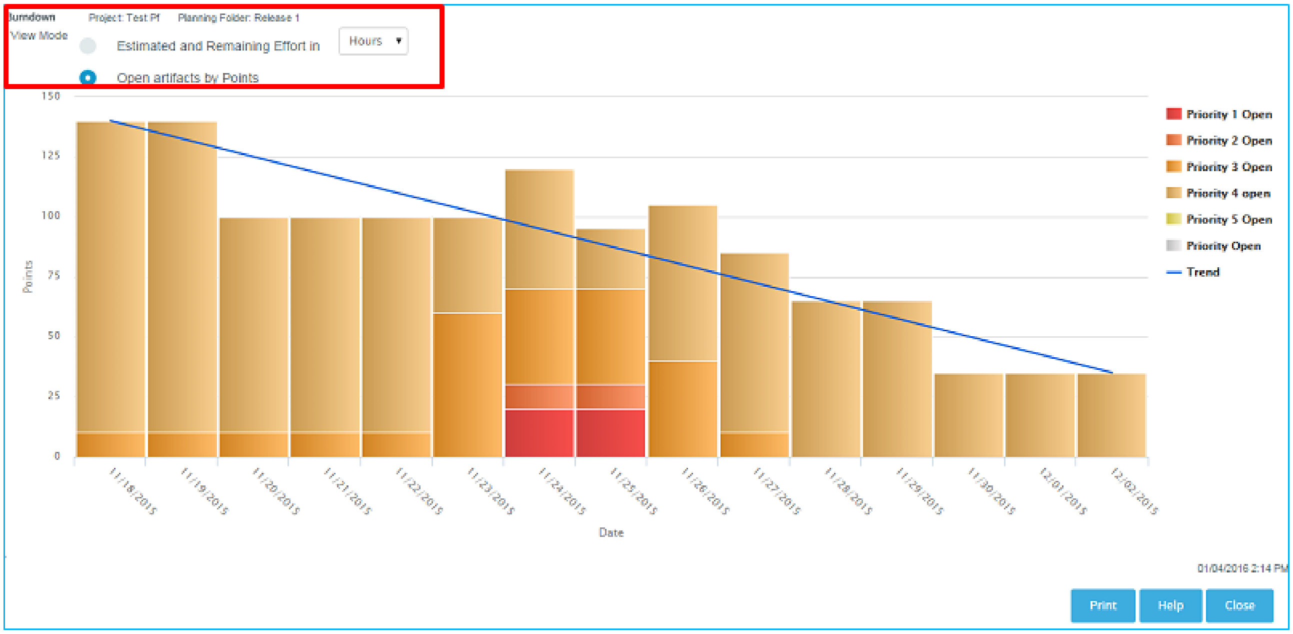
Burndown is the estimated amount of work that remains to be done in an iteration or a release, compared with the work originally estimated.

At any time, you can click on a planning folder (release, iteration or standard) and view the burndown chart which shows the work to be completed in terms of story points within the given time frame. This helps you see how the team is progressing towards “done”. In the burndown chart of a standard planning folder (‘Folder’) you can see how the sum of the estimated and remaining effort and points (story points) for all contained and descendant features is changing within a given time frame by choosing the options available in the View Mode.

In both ‘Release’ and ‘Folder’ burndown charts, you can use the trend line shown in your burndown chart to do some useful things:

- Project the time when all the work for this iteration will be completed.
- Predict the amount of work your team can expect to complete during any given iteration.

‘Folder’ burndown chart which has the View Mode

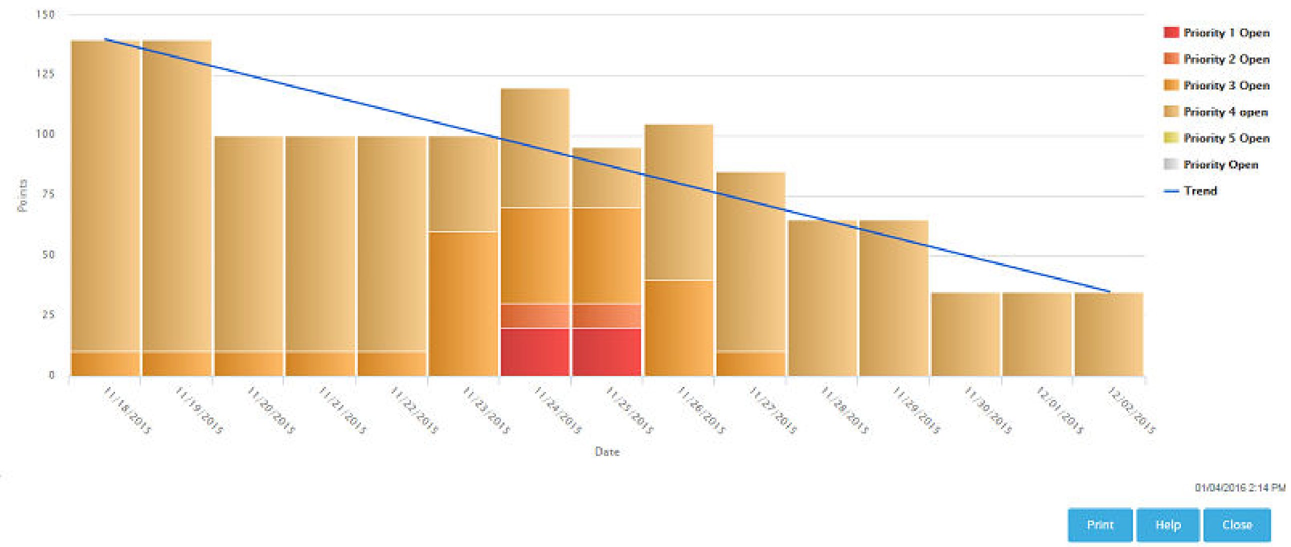


Burndown Project: Test Pf Planning Folder: Release 1

View Mode
○ Estimated and Remaining Effort in Hours
● Open artifacts by Points

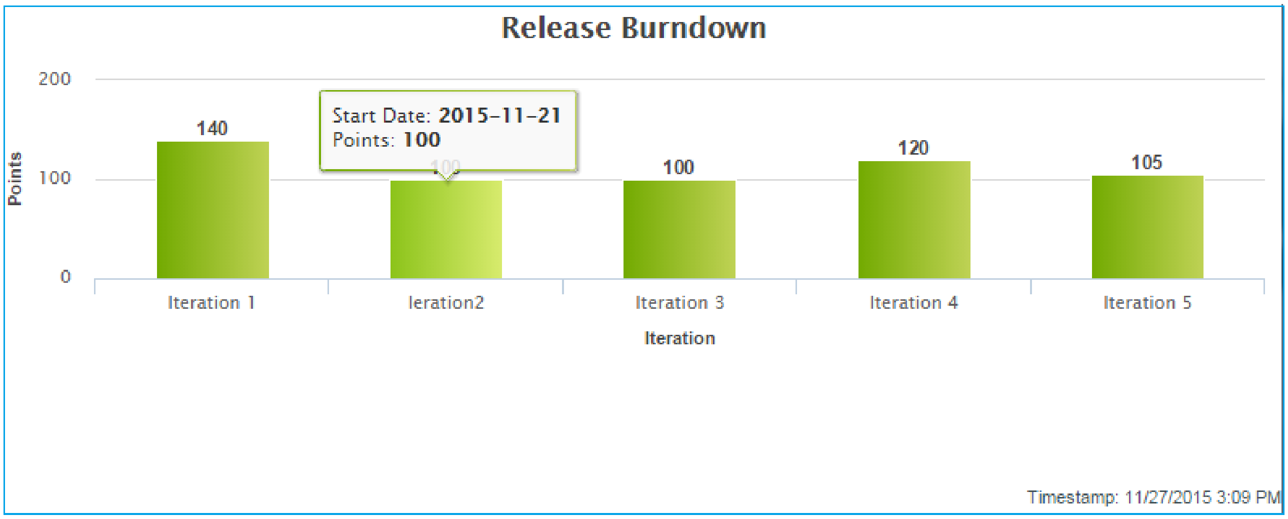
01/04/2016 2:14 PM

‘Iteration’ burndown chart



Release Burndown Chart

A release burndown chart shows an iteration-wise breakup of the work to be completed, that is, it shows how much work (in terms of points) is remaining at the start of each iteration.



The information you derive from your burndown chart will help revise the product scope, make accurate planning decisions, and refine implementation details.

In general, a burndown chart trends downward until it reaches zero. In practice, some events can reverse the downward trend of your burndown chart. For example, development work frequently uncovers a greater scope for a user story than was initially estimated. As a result, you'll revise your estimate of remaining work on that story, and your burndown figure may be pushed upwards.

The burndown chart uses whatever effort units you are using in the planning folder.

> **NOTE:** The story told in your burndown chart is only as reliable as the underlying data. The owners of individual task artifacts can help keep the burndown chart accurate by regularly updating their remaining effort and points (story points).

What can I learn from a capacity chart?

The capacity chart shows the project team's judgment of how much work can be done with the resources available and the time period represented in the planning folder.

- When you compare average capacity with individual users' assignments, you can see which team members are relatively over- or under-assigned. You can also see how much of the estimated and remaining effort is unassigned.
- Comparing average capacity with the relative priorities of artifacts is one way to gauge your team's ability to deliver the work that the product owner has defined.
- When you view average capacity against the count of open and closed artifacts, you have another way to assess your team's likelihood of meeting its sprint commitment.

The data behind the capacity chart is updated in real time, so that your team can respond quickly to changes in effort estimates or relative priorities.

Capacity is expressed as a numeric value, using the same scale as you are using for estimated, remaining, actual effort accounts.

You can see the capacity of a planning folder in the planning folder summary view.

> **TIP:** A planning folder's capacity must be equal to or greater than the total estimated effort (or total remaining effort) for the artifacts in the planning folder. To avoid confusion, you may want to wait for estimates to emerge before setting the planning folder's capacity.

What can I learn from an "Open by Priority" chart?

Dynamic planning is most effective when you know the relative priorities of work items you are tracking. The *Open by Priority* chart helps you check that your team is working on the optimal mix of artifacts.

Use the *Open by Priority* chart for a quick overview of the type of work your project is working with. A glance at this chart can suggest some broad generalizations, which you'll then want to test by examining your planning folder contents more closely.

- When the darker bars on the left side of the chart are high, you are probably working on the high-impact issues that come up earlier in a work cycle. Don't be surprised if the trend line in your *Burndown* chart is not yet sloping steeply downward to the right.
- When the lighter bars on the right side of the chart are high, you are likely to be working on cleanup issues and refinements. You may be approaching the end of the work cycle you defined in this planning folder. At this stage, the Closed slice of the *Open vs. Closed* chart is likely to be equal to or larger than the Open proportion.

> **NOTE:** The *Open by Priority* chart won't tell you how much work is involved in the artifacts that it measures. For that, check the *Burndown* chart.

What can I learn from an "Open vs. Closed" chart?

Sometimes you just want a raw count of the number of work items you are looking at.

Use the *Open vs. Closed* chart for a quick overview of your team's progress in purely numerical terms. A glance at this chart can suggest some broad generalizations, which you'll then want to test by examining your other charts and your planning folder contents more closely.

The *Open vs. Closed* chart is good for some basic summary information. You may want to share this chart with executive sponsors who need to know how things are going in quantitative terms, but don't need a lot of detail about the types of work being addressed.

How does TeamForge deliver activity reports?

The data in your reports comes from a special database that extracts live site data from the production database at intervals you specify.

You can specify the time at which the reporting data is refreshed from the production database. By default, the extraction takes place daily at 2:30 a.m. in the TeamForge application server's time zone. See Schedule Data extraction for Reporting.

The reporting database can be deployed on a separate machine to help channel load away from the application server. Historical data is available even if the application server no longer stores it.

Where does the reporting data come from?

An ETL application extracts data from the live production PostgreSQL or Oracle database where the TeamForge site stores most of its critical data. (Information about reporting configurations is also stored in the production database.) Some data is also gathered from the file system.

How is the production data converted into reporting data?

TeamForge extracts a snapshot of the production data, transforms it into a format that supports reporting requirements, and loads it into the datamart, which is optimized for fast retrieval. The Extract-Transform-Load (ETL) application is a Tomcat JVM running as a TeamForge service under the TeamForge integration server architecture.

Where is the reporting data kept?

After the ETL app collects and processes the live site data, it is stored in a separate database called the datamart. If the TeamForge site uses a PostgreSQL database, then the datamart is also a PostgreSQL database; likewise for Oracle. The datamart uses a Star Schema-based design for tables.

How are the reports shown in the TeamForge user interface?

The reports are rendered in the TeamForge UI using Adobe Flex.

> **NOTE:** When a site is upgraded, there will be a delay before reporting data is available to users, until the scheduled ETL run has occurred. Performing a manual ETL run immediately after an upgrade is not advisable, since it could consume a lot of system resources leading to performance problems.

Why do I get duplicate records in a tracker report?

Yes, this is a known limitation.

For a tracker report, when you select more than one value from a multi-select, user-defined field as filter and if an artifact is associated with all of the selected values, then that artifact’s record is duplicated for each of the selected values.

For example, assume that you have a ‘Select User’ multi-select, user-defined field with values ‘User 1’, ‘User 2’ and ‘User 3’ in a tracker report. All these three values are associated with ‘artifact 1001’. Select all three values as filter and generate the tracker report. You will see ‘artifact 1001’ record being duplicated, that is, you will see three individual ‘artifact 1001’ records created for each of the three users.

What is the difference between a stagger and normal header in query result heading settings?

The normal and the stagger header preference setting allows you to set a standard header for your query view or a staggered header.

For more information about queries in Issue Tracker, see: Query database of issues.

Can the query result be listed without the issue id?

No. Query results can never be listed without the issue id because issue id is the only field which identifies each issue uniquely.

These are some of the frequently asked questions on Discussions and Discussion Forums in TeamForge.

Is the discussion forum creator subscribed by default?

Yes. The user creating the discussion forum is subscribed by default.

How can I monitor a forum or mailing list?

There are two methods for monitoring a forum or mailing list in TeamForge: at the Discussion Forum level and at the My Workspace level.

At the Discussion Forum level:

- Check the box beside the forum in question.
- Select the **Monitor** button.
- Select **Monitor Selected**.

At the My Workspace level:

- Select **Monitoring**.
- Select the project name.
- Select the *Monitored Applications* tab.
- Select **Discussions**.

- Click **Save**.

> **NOTE:** You must monitor a forum to receive email from the mailing list associated with that forum.

How do I find the email address for a forum?

The email address for the forum is displayed on the Topic Summary page. If a mailing list is not enable, you will not see an email address on the Topic Summary page.

To find the email address for a forum:

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, click the title of the forum in which you want to create a forum topic.
   The **Topic Summary** page is displayed. The email address is listed in the **Mailing List** field.

How do I remove a user from Discussion?

Unfortunately, currently there is not a way for a TeamForge admin to remove a user from monitoring.

Instead, you can remove the user from monitoring the discussion by clicking the **Stop Monitoring Selected** option from the **Monitor** drop-down button on the **Forum Summary** page.

I can see the message I posted to a discussion in the web UI, but I didn't receive any of the responses through email.Why?

To receive forum posts through email, you must monitor the forum.

You can do this by selecting the forum you wish to monitor and choosing the **Monitor Selected** option from the **Monitor** drop-down button at the bottom of the **Forum Summary** page. For more information, see Monitor Many Items.

Why are some of the discussions threaded?

Posts that are sent the mailing list address of the discussion will create new topics, and hence will not be threaded.

Posts that are made in response to another post will be threaded beneath the original post.

What happens when I post to a moderated discussion forum?

A message to a moderated discussion forum is held until a moderator acts on it. (Except if you are a trusted user. These messages do not require moderation.)

- The moderators of the discussion receive an email notification that you have posted a message. The email notification contains the URL path to moderate the post.
- A moderator can either approve or reject your message.
- If the moderator accepts the message, the message status changes to “Accepted” and the message is posted to all the users monitoring the forum.
- If the moderator accepts the message and trusts the sender, the message status changes to “Approved and Trusted” and the message is posted to all the users monitoring the forum. All subsequest posts from you are automatically approved.
- If the moderator rejects the message, he/she can include his/her comments or reasons in the moderation rejection email, and the message is removed from the archive.
- In the *All Topics* tab, an hourglass icon indicates which topic contains posts that await approval.

Who can be a moderator?

A discussion forum moderator is selected by the forum administrator.

- You can be a moderator by being a project member with forum post permissions.
- A moderator can moderate a discussion forum using emails or the web UI or one of the CollabNet desktops.

  If the mailing list isn’t enabled for a moderated discussion forum, moderation can be done only through the Web UI.
- A moderator can trust the sender and approve a message or reject the message.

> **NOTE:** To enable or disable moderation or add or remove moderators, you must be either the forum administrator or project administrator.

Do project owners get automatically subscribed to the discussion forum started by another member?

A project owner does not get subscribed to the discussion forum started by another user.

To make sure the project owner is included, add the user as a member when creating the forum.

Can I subscribe to a TeamForge discussion forum’s mailing list through an email?

Sure, you can subscribe to a discussion forum’s mailing list to keep up with changes, through an email. When you subscribe or monitor a discussion forum, you are notified by an automatic email whenever there is any update to the discussion.

To subscribe to a mailing list - You can subscribe to the discussion forum’s mailing list by sending an email to `<mailing list name>-project name-subscribe@domain>`.

You must have the *Discussion: view* permission to be made a subscriber. Check with the forum administrator, if you do not have the permission.

As with other items that you may be monitoring, the discussion forum to which you subscribe through an email is also added to the **My Workspace > Monitoring > Monitored Objects** list. Your user name is added to the ‘Users Monitoring’ list of the discussion forum. Both these entries are removed when you unsubscribe from the mailing list.

To subscribe in digest type, send an email to `<mailing list name>-project name-digest-subscribe@<domain>`.

To unsubscribe from a mailing list - You can unsubscribe from the discussion forum’s mailing list by sending an email to `<mailing list name>-project name-unsubscribe@<domain>`.

Why would I want to make a discussion forum moderated?

Automated processes can only do so much to protect forum quality. Moderated posting can be thought of as a temperory transfer of control from automated processes to a human decision maker.

Software can instantly analyze an email to determine whether the sender is allowed to post messages directly to the list or if the message should be sent for moderation. However, the software doesn’t recognize the sender as a person who participates in the organization, nor does it have the sophistication needed to determine whether the message contents are on-topic for the mailing list.

Advantages of moderated posting:

- Moderators help keep content on-topic by rejecting off-topic messages and providing helpful suggestions to users whose posts are rejected.
- Moderators help maintain a positive environment for list users by rejecting messages containing harsh or abusive language.
- Moderators can allow deserving non-members to post to lists that are ordinarily closed to non-members.
- Moderators prevent spam.

Example

Ashish makes the development forum in his project a moderated one as he wants to make sure that all the messages posted to the discussion come to him for approval before they're included in the forum. When a message arrives, he reads the message, and if it's appropriate for the discussion, he accepts it; if not, he rejects it.

Over time, Ashish finds that the traffic in his discussion has increased and he is no longer able to moderate all the posts by himself. So he adds a couple of other senior developers in his project as moderators, who can share the responsibility of moderating the forum.

After a while, Ashish realises that he doesn't have to reject any messages posted to the forum as everyone seems to understand the purpose of the forum and users appropriate language in emails. so he removes the restriction and make the forum an unmoderated one. Now Ashish and the other moderators no longer receive emails for approval when a user posts a message to the discussion. Messages are directly included in the forum and delivered to the forum subscribers.

These are some of the frequently asked questions on Documents.

How does TeamForge support Documents?

A key element in a successful product development project is ensuring that stakeholders from all functional groups are involved in an alterative document review and approval process, particularly for critical documents, such as Product Requirements Documents.

It is also critical that any changes to key documents are recorded and the new information quickly communicated to everyone concerned.

The TeamForge Document Manager helps you manage your documents throughout their lifecycle and ensure the appropriate level of involvement of other project members. In addition to storing documents for reference purposes, the Document Manager provides a document review workflow process to allow you to actively engage other users in the review and approval of a document.

Simple Notification Process

If you would like to keep other TeamForge users informed when the status of a document, such as when a document is posted or updated, or there is a change in the conents or status of an existing document, but do not require a formal review and approval process, the following steps provide a best practice for this level of document management.

- Post the document to the Document Manager in the desired location. If you would like to prevent other users from editing, moving, or deleting the document, use the ‘Lock Document’ feature. (If you choose not to lock the document), any user with the document edit permission can edit and update the document.)
- Notify other users that the document is available and request that they begin monitoring it. All users monitoring an item receive email notification whenever the item is updated.

Document Workflow Review and Approval

In cases where more formal document review is desired, the TeamForge Document Manager provides an easy-to-use workflow process for managing a document’s review and approval.

After submitting a document, you can start a document review. You are prompted to provide the following information:

- The names of all required and optional reviewers (All reviewers must have permission to access the document.)
- The date by which the review must be completed.
- Email message text.

You will be notified each time a reviewer submits review comments.

When selected to review a document, users receive an email notification with the relevant details and a link to the document.

You can also choose to attach the document to the email notification. An item also appears in the **Documents Awaiting Review** section of each reviewer’s **My Page**, including the due date.

A **Submit a Response** section is provided within the Document Manager where reviewers can enter their review comments. Reviewers should be instructed to include “I approve this document” or similar text in the **Submit a Response” section to indicate that they have approved the document.

All reviewers and the document submitter will be able to read the reviews of all other reviewers once they have been submitted.

Why can’t I edit a document when it opens in my browser?

There are chances that your browser swallowed the running application.

If you click on a document in TeamForge (for example, a Word doc), and it opens in your browser instead of launching a separate MS Word window, chances are you will not be able to edit this document. This is because your browser has either called one of the MS document viewers that do not have native edit capabilities, or the browser has ‘swallowed’ the running MS Word process and the document has written to your %TEMP% directory as a read-only file.

You can either choose to download the file to your desktop and then click on it to edit, or configure your browser to not swallow the application (this is described in various MS KB articles).

How can a user who deleted a document get it back?

It is possible to recover the latest version of a document if someone has accidentally deleted it from TeamForge.

However, note that you can recover the raw document but not undelete it from the app. The user will have to add the document back into TeamForge, and the history of the document will not be recovered. If you need a true undelete so that the history is available, please contact Technical Support and ask for us to schedule our Professional Services group for you.

Let’s assume you have a project in TeamForge called Test, and the URL in your browser when you load this project is: http://your_sfee_server/sf/projects/Test. Please note the word after projects, this is your project name.

Now, let’s assume the file to be retrieved was in a folder called ‘foo’ in the document manager. Further, let’s assume the document was called ‘example’. On the database, run the following command:

```
select stored_file.raw_file_id, document.id, item date_last_modified, item.title,
    stored_file.file_name, sfuser.full_name from item, folder, document, documen
t_version,
    stored_file, sfuser where item.title like '%example%' and item.folder_id =
folder.id and
    folder.path = 'docman.root.b' and document.id = document_version.document_
id and
    document_version.stored_file_id = stored_file.id and item.is_deleted = 't' a
nd
    item.last_modified_by_id = sfuser.id ;
```

This will return something like:

```
raw_file_id | id | date_last_modified | title | file_name | full_name
```

Examine the rows of output to determine which of the returned results is the document you need, from the title and the name of the file that was uploaded. Once you've determined this, look for the file in `/opt/collabnet/teamforge/var/deleted_files`.

```
 yujr0C98B989011083DBFF4F534F | doc1001 | 2007-04-11 08:49:19-07 | example |
   foo.txt | Joe User bxma0C98B98901115B9CE99E82C2 | doc1002 | 2007-04-11 08:4
9:19-07 |
   example | test.txt | Joe User rpyf0C98B98901116DD1E40F9F8B | doc1003 | 2007
-04-11
   08:49:19-07 | example | findme.txt | Joe User ltrq0C98B9890111D76E74920A0E |
 doc1004
   | 2007-04-11 08:49:19-07 | example | project_logo.jpg | Joe User
```

Let's say the fourth document above was the one you wanted. You can find it at: `/opt/collabnet/teamforge/var/deleted_files/l/lt/ltr/ltrq/ltrq0C98B9890111D76E74920A0E`. Simply copy that file off the system, rename it to `project_logo.jpg` and return it to the user.

Why can't I edit a document when it opens in my browser?

There are chances that your browser swallowed the running application.

If you click on a document in TeamForge (for example, a Word doc), and it opens in your browser instead of launching a separate MS Word window, chances are you will not be able to edit this document. This is because your browser has either called one of the MS document viewers that do not have native edit capabilities, or the browser has 'swallowed' the running MS Word process and the document has written to your %TEMP% directory as a read-only file.

You can either choose to download the file to your desktop and then click on it to edit, or configure your browser to not swallow the application (this is described in various MS KB articles).

Why are some uploaded documents missing icons when displayed in TeamForge?

TeamForge has a small internal mapping of which icon is associated with which mimetype(s). This mapping is necessarily small, as a complete mapping would be exceedingly large, and almost always out of date.

You can examine the mimetype that TeamForge is receiving by examining the document's attributes using our SOAP API. You can also override this mimetype info via the API by uploading a new version of the document (which can be the exact same document contents). There is sometimes a generic icon for documents from certain applications. TeamForge accepts and stores the mimetype that the user's browser sends on the HTTP upload of the original document. As an example, let's examine the MS Word document.

Some browsers will send strings like `application/vnd.ms-word`; others will send `application/msword` and others will send something completely different.

Can I link to documents outside of TeamForge?

TeamForge has the concept of a ‘url doc’ to support this very usage.

When creating a document in TeamForge, simply choose this type from the Create screen and then enter the URL that references the document in the existing external system. This will create a ‘placeholder’ document in TeamForge that can be associated to, reviewed, and so on as if it were a normal document, while maintaining the document’s actual contents in the external application.

Can I lock a document in TeamForge?

Absolutely, yes. You can specify a document as locked at any point in time (document create, document edit, etc).

You can do this through the UI or through the SOAP API. Simply check relevant boxes available in the **LOCK DOCUMENT** section on the **Document Details** page.

By default, edit and download locks are disabled for a document. To lock and unlock your document, you must have the Document Edit permission in addition to the Document Create permission. To prevent others from editing or downloading a document, you can choose the edit lock or the download lock as required. Selecting the edit lock automatically enables the Document Administrator to edit the document and allows you to select the download lock option.

To edit or download a locked document, the user must have the Document Edit permission. The user who has locked the document can edit or download it at any point in time. A Document Administrator that has the option, **Allow Document Admin to edit** enabled in the **LOCK DOCUMENT** section or a Site Admin can also edit or download a locked document.

Does TeamForge automatically resolve conflicts in documents made by multiple concurrent editors?

If you store your documents in CVS, Subversion, or some other SCM tool, then that tool will handle conflict resolution per its normal means.

However, if you are using the Document Manager component of TeamForge, then there is no automated means within the product to prevent another user from uploading a newer version of a given document that does not contain the changes you just uploaded. To prevent this, we recommend you use the edit lock and download lock features of TeamForge to prevent others from editing and downloading a document you are currently editing.

What document types are supported in TeamForge?

TeamForge supports all known and unknown document types.

In fact, TeamForge makes absolutely no distinction on a file type when uploading a document. This means that you can even upload binary files (like a Zip archive) into the document manager. Note that while the TeamForge accepts any incoming data stream as a document, it does use the mimetype sent by the browser to determine if it has the proper icon to display for the document.

Why doesn’t an open review automatically close when a new version of the document is uploaded?

TeamForge does not automatically close a document review under any circumstances (new document version, review due date passes, and so on).

This was a conscious decision on our part in which it was decided that TeamForge cannot always be aware of the business rules or personnel availability at a customer. For example, TeamForge cannot know that the one person whose document review input is most needed is on vacation for the four days the document was under review. If TeamForge were to close the review, then it would disappear from the user’s **My Page**. Additionally, TeamForge cannot know that a new version of a document supersedes the prior versions (or that it does not supersede it). You may be uploading a ‘draft’ or ‘work-in-progress’ of the new version as document review feedback is received and if TeamForge were to close the review at this point, you might have received feedback from less than one percent of the reviewers.

Who can work with documents?

Any project member with the Document Edit permission can edit a document.

Can I set permissions so that users can move documents but not delete them?

You cannot configure document management permissions so that a user can move documents but not delete them.

It is not possible to separate move and delete permissions, because a move is actually a copy/delete action. The document is not really moved, it is copied to the new location and then deleted from the original location.

Why am I not able to access a folder in TeamForge documents?

You can access a folder in TeamForge documents only if you have permission for the folder or any of its parent folders.

For TeamForge documents, users having permissions to the parent folder can access all the subfolders. Users having permissions to a subfolder can only view the hierarchy of the parent folders; they cannot access the documents in those folders.

Are role-based permissions allowed for subfolders in the TeamForge Documents?

Yes. It is allowed in TeamForge Documents.

The Project Administrator can give permission to access sub folders in the TeamForge Documents based on the user roles, using the Roles option.

Can I set permissions for documents alone?

Yes, as a project administrator, you can set individual permissions only for documents.

The Document Admin permission allows a project member to create, edit and administer both documents and document folders. Though this does not include the delete permission, individual permissions to create, edit, and delete can be set separately for document folders and documents on the Documents Permissions page (**Project Admin > Permissions > Edit Role > Documents**). This is to restrict a project member from handling document folders or subfolders. For example, if you do not want a project member to create a document folder, but only create a document within a specific folder, you can set the ‘Create/View’ permission for documents alone. The available permissions are:

- Create/View
- Edit/View
- Delete/View
- View Only

Why can’t I reply when someone comments on my review?

By default, TeamForge doesn’t add the review initiator to a review, which would be needed to facilitate this.

To work around this, simply add yourself as an optional reviewer when creating the review. Alternatively, some customers have chosen to create a forum in TeamForge, and then include the forum's posting address in the review notes.

These are some of the frequently asked questions on projects.

How is a project template structured?

When you create a template that includes project content, each tool brings in its own kind of structure, depending on the type of content it manages.

Folders

Some tools, such as Documents, Tasks, and File Releases, can be thought of as folders that contain individual items, subfolders, or both.

For example, the Documents tool has a Root Folder, inside which you can create a tree of subfolders to organize your documents according to your project's needs. Every document lives inside one of these folder.

> **NOTE:** This works the same way regardless of the name of the root folder. For example,in the Tasks tool, the root folder is called the Tasks Summary, and in the Discussions tool, the root folder is called the Forum Summary.

In a project that has been in use for any significant time, users have probably created some number of documents, which they have shared by adding them to a folder in the Documents tool. When you create a project template from that project, you can choose to include those documents in the project template or not.

- If you choose to include the documents in the template, those documents will appear in any new project that is created using that template.
- If you do not include the documents in the template, new projects based on that template will include the Documents tool, including the root folder and its sub-folders, but none of the documents that existed in the original project.

Not Folders

The Reports and Wiki tools are not organized in folders. When you include the content from these tools in a template, new projects created from that template include a flat collection of all the Wiki pages or reports in the original project, with all their text and data.

Why do I get ‘Name is Invalid’ error when trying to create a project using createProject() method via the SOAP API?

You need to pass certain default parameters to createProject() method.

The default parameters that can be passed to createProject() method are:

```
(sessionId, name, title, description)
where, name = URL Name [in UI]
title = Project Name [in UI]
```

In TeamForge, URL Name [‘name’] accepts only lower case and the numeric characters. If the value of ‘name’ contains any other character than the aforesaid, user will end up seeing the ‘Name is Invalid’ error and the project creation would fail.

Why do I get a TeamForge system error in the project template creation page?

This may be because of a few stale permissions in the project in which you are trying to create the template.

You can resolve this by identifying and deleting the stale records using this SQL.

```
select role_id from role_operation ro left outer join ia_project_association i
a
on (ro.resource_value = ia.id) where ia.id is null and resource_value like '%p
rpl%' ;
select role_id from operation_cluster ro left outer join ia_project_associatio
n ia
on (ro.resource_value = ia.id) where ia.id is null and resource_value like '%p
rpl%' ;
```

How does TeamForge support dynamic planning?

TeamForge helps you maximize your team’s effectiveness by keeping you in close touch with the multiple moving targets facing your project.

In a development project, each piece of the picture constantly changes in response to changes in the other parts. As you continue to iterate through the process, a feedback loop like this takes shape. (Click a node for more detail.)

The new Dynamic Planning features in TeamForge 18.0 give you a more open and extensible platform that integrates and centralizes the software development tools necessary in modern application life cycles.

> **NOTE:** The Dynamic Planning features helps you effectively utilize Agile-like processes, but you can use any process model you like.

Tracker summary screen

TeamForge incorporates a new Tracker Summary screen display to accommodate planning folders, tree-views, and multiple tracker viewing. The Tracker Summary section is available at the top of the screen with summaries of open and closed artifacts as well as a summary of open artifacts by priority. The Planning Folders section is located at the bottom of the Tracker Summary screen. This section includes summaries of open and closed artifacts, a summary of open artifacts by priority, and a summary of Effort for each planning folder.

Tree view

TeamForge incorporates an expandable and collapsible tree view of the planning folder hierarchical structure to display parent/child relationships of artifacts.

The tree view allows for the viewing of artifacts in a hierarchical structure and displays parent/child relationships across multiple trackers.

How can I make the project pages invisible for any user who is authorized to see the project?

This can be customized by a Project Administrator.

The Project Administrator can set this using these steps:

1. Click the link **On** next to Configure, on the top right-hand side of the project home page.
2. Scroll to the page you want to hide and click the **Edit Properties** icon. This can be found on the right side of the title bar for the page you want to edit, next to **Edit** or **Create** buttons. The tooltip displays if you hover over the tool.
3. Choose **Hidden - Visible only to project administrators in configure mode**.
4. Click **Save**.

How do I put a notice to my users on the project home page?

To do this you will need to create a news item within the project. News items are posted and displayed on the project home page. News items are also displayed on the TeamForge home page.

To post a news item:

1. Navigate to the home page of the project in which you want to post the news item.
   - From within the project, click **Project Home**.
   - From anywhere in TeamForge, choose the project from the list pf projects available under **My Workspace > My Projects**.
2. In the Project News section, click **Add**. The **Create News Post** page is displayed.
3. Enter a title for the news item.
4. Enter the text for the news item in the message field (up to 4000 characters including spaces), then click **Add**.

The news item is submitted. It is displayed on the project home page and the TeamForge home page immediately.

How do I remove a news item?

You can delete any news item that you no longer want displayed on the project home page. Deleting a news item from a project also deletes it from the TeamForge home page.

To delete a news item:

1. Navigate to the project home page.
2. From within the project, click **Project Home** in the project navigation bar. From anywhere in TeamForge, choose the project from the **Projects** menu in the navigation bar.
3. In the **Project News** section, click **Delete** next to the news item that you want to delete. When prompted, confirm that you want to delete the news item, and it will be deleted.

These are some of the frequently asked questions on planning folders.

What is a planning folder?

A planning folder is a virtual contain that helps you organize and planning the work that goes into delivering a product.

You can create a hierarchy of planning folders to organize artifacts by product, release, iteration, etc. You can store artifacts from multiple trackers in a planning folder. This allows you to plan various stages of your project. (i.e. releases, iterations, etc.).

Selecting an individual planning folder provides a view of all artifacts from all trackers within the selected planning folder.

The **Planned For** field identifies which product, release, or iteration the artifact is planned for, based on the planning folder that is assigned to.

For example, in an agile development environment, a project manager breaks down the prospective product into its component parts and looks at what it would like to deliver each one. When all the parts planned for a given iteration are finished, the product is considered complete for that iteration.

Some parts of a product can be developed more or less in isolcation, but most depend on other parts. Tracking these relationships is one of the trickiest aspects of product development.

For example, you can only provide a graphical user interface for a shopping cart application if you also come up with a database for the customer’s payment data to be stored and accessed. That in turn requires a data storage and backup solution of some kind and so on.

Use a planning folder to tracke the dependencies among the part of your project as each moves toward completion. As you work through the question of what depends on what, you’ll move artifacts representing user stories into the appropriate release. As you proceed, you’ll find a pattern like this emerging:

- Product 1
  - Release 1
    - Iteration 1
    - Iteration 2
  - Release 2
- Product 2

> **NOTE:** `Move` is meant figuratively. When you move an artifact into a planning folder, it is still a member of the tracker where it livers, and you can still do all the things with it that you can do with an ordinary tracker artifact.

Your planning folder lets you see at a glance the pieces of the work that support other pieces, and the pieces that depend on other pieces. Think of this as “planning tree”. If you are responsible for a development project, you can use this tree view to understand and predict the time and effort required to deliver a given set of features.

What does the status of a planning folder mean?

The status of a planning folder communicates where it currently stands in the development process. You can set the status of a planning folder to values you define yourself.

The status of a planning folder can help project members make sense of what can be a complex development process. For example, consider these scenarios:

Drop everything!

Just when you have prepared a planning folder and are ready to kick off a sprint, the product owner gets word of a more pressing set of features and asks you to postpone this work and get right to the newly revealed priority. You are worried that your project's momentum lead project members to keep picking up tasks in this planning folder instead of switching to the new work.

Create a status called 'On hold', and specify it as an 'Inactive' status. When you give the planning folder your new 'On hold' status, project members whose view shows only active folders would not see this one in their planning folder list.

It's a wrap

Your sprint has just concluded. You don't want any further development work to happen in this planning folder, but you do want team members and others to be able to peruse it for insight into what went right and what went wrong.

Create a status called 'Restrospective', and specify it as an 'Active' status. When you assign this status to your planning folder, participants will be able to confirm that the sprint in question is finished and their comments are invited.

Status tips

- It is a good idea to choose a default status, such as 'Under construction', so that project members are not confused when new planning folders appear.
- You can organize your planning folder statuses in a way that makes sense for your team, such as alphabetically or chronologically.
- Delete statuses that are no longer being used, so you do not clutter up the section.
- When you delete a status that is in use by one or more planning folders, the status of those planning folders is changed to 'None'. Planning folders that were created in an earlier TeamForge version also get a 'None' status at first.

These are some of the frequently asked questions on Trackers.

What fields can I use in a tracker?

You can use fields provided by TeamForge, and you can create your own custom fields. Most trackers use a mix of TeamForge-provided fields and user-defined fields.

TeamForge-provided fields

Some fields are provided by TeamForge for all trackers.

- **System-defined Fields** - Some TeamForge-provided fields are always present in every tracker. These are identified as “system-defined” fields in the field list.
    - The **Artifact ID**, **Title**, and **Description** fields are system-defined fields. They are always present and always come first when you view an artifact.
    - Some system-defined fields can be reordered. For example, the **Assigned To** field is a system-defined field, so it is always present, but you can move it around in your tracker’s display.
- **Configurable Fields** - Some fields are available for any tracker, but can be modified in different ways. These are called “configurable” fields.
    - Any configurable field can be a required field. When a field is required, a user cannot submit an artifact without providing a value for the field.
    - Some configurable fields can be disabled. When a field is disabled, TeamForge does not store data for it or include it in calculations. For example, the **Reported in Release** field is meant for tracking bugs, so you may want to disable it for a tracker that’s for user stories. But the **Status** field is important for any tracker, so it can’t be disabled.
    - For some configurable fields, you can specify whether users see the field when they submit an artifact.
    - Some configurable fields need values for users to choose from. You have to provide those choices.
    - You can set the width and height of configurable text fields.

User-defined fields

When none of the TeamForge-provided field captures exactly the information you need to track, you should create a field tailored to your own needs. Such fields are identified in the field list as “user-defined” fields. See Create Custom Tracker Fields for details.

Can I limit the number of characters for fields in a tracker?

No, it is not possible in the current version of TeamForge.

Currently, while creating the tracker customer flex field of input type ‘Text Entry’, you cannot control the maximum number of characters to be allowed. There is ‘Field Width’ that takes values to control the size of the text field, but not to limit number of characters entered.

Not all my team members appear in the “assigned to” field. Why is that?

In order to appear in the Assigned To field, the user must belong to a role giving them edit/view permissions on the tracker.

How does TeamForge automatically sum up effort estimates?

As you refine your agile project plan, you break down user stories into smaller stories, and eventually into tasks. TeamForge can watch the changing effort estimates all the way down the hierarchy, and give you a running total for each parent artifact and all its children.

Example

Imagine that you originally created a user story to describe the need for a “Shopping Cart” feature in your e-commerce application. You gave this story a rough effort estimate of 20 units, and entered that figure in the **Effort** field.

Upon further discussion, your project team recommends breaking the story down into three tasks:

- Database (5 units)
- Back End (15 units)
- UI (8 units)

You create artifacts for those three tasks, identifying them as children of the user story artifact, and enter their respective effort estimates in the **Effort** field in each task artifact.

Back in the “Shopping Cart” story artifact, you select **Autosum effort** to indicate that effort number for this artifact should be rolled up from its children.

Now, rather than showing the manually entered number (20 units), your user story artifact shows the figure derived from adding the effort estimates for all three child artifacts (28 units).

Double-counting?

Effort numbers are never “double-counted” when auto-summing is on. For example, if the sum for planning folder A would include effort for child C and parent P, and P has autosumming turned on, C’s effort number is only counted once.

However, when auto-summing is off, the effort numbers from parent P and child C can both contribute to totals. So if a planning folder A includes parent P (autosum=off, effort=3) and child C (effort=5), then the pair contributes a total of 8 to A’s total.

An artifact’s effort number is calculated from the effort numbers of its immediate children artifacts only. This is true whether or not those children’s numbers are themselves automatically derived.

> **NOTE:** When a tracker is disabled, artifacts from that tracker do not contribute to the effort totals calculated for any planning folder they are in.

Permissions

If you are a long-time user of TeamForge tracker features, be aware that autosumming can lead to some situations you haven’t seen before. For example, picture these scenarios:

- Artifact A is assigned to Sanil. He therefore has permission to edit that artifact, but he does not have permission to edit the artifact’s parent, artifact B. Artifact B is assigned to Sergey, who has edit permission for both artifacts. Sergey turns on autosumming for artifact B. When Sanil updates his effort estimate for artifact A, the content of the **Effort** field in artifact B is updated as a result, even though only Sergey is explicitly authorized to change values in artifact B.
- Connie creates artifacts X and Y, and declares both of them children of artifact Z. Then she assigns artifact Z to Thiru. Thiru now has edit permission for artifact Z. But he cannot edit artifact X or artifact Y, because he does not have access to the separate tracker where they live. When Thiru turns on autosumming for artifact Z, that artifact’s **Effort** field includes the sum of artifacts X and Y, even though Thiru is not explicitly authorized to see any data from those artifacts. (In fact, he may not even know that artifacts X and Y exist.)

How do you measure “effort”?

Effort is a uniform span of time for measuring work on your product. Defining the unit of effort is an essential part of planning your work.

TeamForge shows you the effort that has been estimated, and actually expended, for each artifact on the tracker summary screen. Parent artifacts can automatically add up these effort figures from their child artifacts' effort figures. The calculator icon indicates that the artifact's effort is a sum of its child artifacts' effort within the project. When the effort from children artifacts in projects across the TeamForge (foreign children) is included in calculations, the icon ( ) appears.

> **NOTE:** When a tracker is disabled, artifacts from that tracker do not contribute to the effort totals calculated for any planning folder they are in.

The unit of effort should be something that makes sense in your environment. TeamForge does not require any particular unit. Your unit of effort might be hours per person, days, weeks, or something else. (If you are using a scrum-based project methodology, you may have opted to measure effort in relative terms, using points (story points), in which case you can leave the **Effort** fields blank.)

For example, some teams use the "ideal hour" as their standard unit of effort. To define an ideal hour, consider all the activities in a standard work day that must be done but don't directly contribute to development: installing and configuring tools, eating lunch, responding to email and instant messages, providing customer support, etc. For every hour of direct development work, how much time goes into these activities, on the average? If the answer is about a half hour, then your ideal hour is 1.5 clock hours.

Now consider a task that you judge to represent about four hours of direct development work. The value you'll enter in the **Effort** field is 6, because for each hour of direct development you'll need an extra half hour to make that development work possible.

In an environment with a lot of overhead – for example, a group that relies on a very complex tool set – your ideal hour might equal two clock hours, or perhaps much more. This is not in itself a problem: the point is not to suppress needed activities, but to plan realistically, in order to reduce the need for routine scheduling adjustments and to forecast more reliably.

(However, when units of effort seem radically out of line with reality, this may be a clue that something in the environment could be better optimized.)

For a quick view of the effort values you are working with, check the Planning Folder Summary page.

- The estimated, remaining and actual values are listed for the planning folder as a whole.
- The **Est** column shows the original estimated effort value for each artifact in the folder.
- The **Rem** column shows the value remaining for each artifact after the latest update to the artifact.
- The **Act** column shows the actual effort expended for each artifact, calculated after the work is done.

> **TIP:** To make your tracking easier, consider having TeamForge automatically generate a running total of estimated effort for a whole hierarchy of user stories or tasks.

These are some of the frequently asked questions on Tracker Artifacts.

Can I assign an artifact or a task to a group of users?

Unfortunately, this is not possible with the current implementation of groups in TeamForge.

As a workaround, you can create a new user account and set its email address to that of a mailing list that all group members are subscribed to. Then the submitting user can assign the artifact to this user account, and the mailing list will be notified.

How do I create an artifact through email?

To create a tracker artifact using email, send an email message to `<tracker id>@<TeamForge server>`.

You can find the tracker id on the **Artifact List View** page. Fields are mapped as follows:

- To: Tracker email address
- Subject: Artifact title
- Body: Artifact description
- Attachments: Attachments

Why do the open tracker counts differ from when I filter on ‘Open’?

The ‘Open’ category on the summary screen is based on the meta status ‘Open’. It includes multiple statuses: Open, Fixed, and any other statuses that are defined as equivalent to Open by the Tracker admin for your project

For example, suppose the Tracker Summary block shows 21 open artifacts. However, when filtering on the ‘Open’ status, only five artifacts are displayed.

If you filter by ‘All Open’, you will see all 21 issues with statuses defined as Open. If you filter specifically by ‘Open’, you will see only the artifacts that specifically have the status ‘Open’.

Is it possible to move an artifact from one tracker to another?

Yes, it is possible to move an artifact from one tracker to another tracker.

Simply use the **Cut** button to remove the artifact and then paste it to another artifact. You can also change the tracker type while editing the artifact using the **Edit Artifact** page. The destination tracker need not be in the same project, but if the tracker definitions differ, data could be lost.

What happens when a changed value makes a dependent field invalid?

When a user changes a field value on which the values of other fields depend, some dependednt values may need to be corrected.

This can happen if:

- A parent value has been changed in such a way that its child values are inconsistent.
- A text value has been changed in such a way that it no longer matches the regular expression that describes the acceptable values.

> **NOTE:** Linking fields in dependent relationships does not modify existing data. However, when users later modify fields that are linked, they do have to adhere to the relationships among the fields.

Changing Values

When a user changes the value of a field, and the value in another field becomes invalid as a result, the latter field (we call it the 'dependent' field) is reset to its default value the next time someone updates that artifact.

For example, picture a tracker called 'Interior decoration' in which, if the **Color Scheme** field is set to `Warm`, the **Carpet color** field can be set only to `Red`, `Orange` or `Yellow`. `Red` is the default value.

User 1 sets **Color Scheme** to `Warm` and **Carpet color** to `Yellow`.

User 2, after consulting the client, comes in and changes the color scheme to `Cool`. Now the allowed value of **Carpet color** are `Blue`, `Gray`, and `White`, with `Blue` as the default. But the actual value is still `Yellow`, which is now invalid. It's up to the user who is updating the artifact to choose a new color from the new set of valid colors. If the user doesn't make a choice, TeamForge automatically changes the value **Carpet color** to `Blue`, because that's the default value when **Color scheme** is set to `Cool`.

Changing Dependency Rules

When you restructure a hierarchy of dependent values in a way that leaves invalid child fields in multiple artifacts, each of those artifacts must be fixed the first time a user edits it.

For example, in the artifact `room1`, User 1 has set the **Color scheme** field to `Warm` and the **Carpet color** field to `Red`. Suddenly User 2, the project manager, realizes that red is a terrible color for carpeting. User 2 changes the tracker settings so that **Carpet color** can be set only to `Orange`, `Yellow`, or `Maroon`.

The next time User 1 comes back to the `room1` artifact, TeamForge warns that the field value is in an inconsistent state.If the user tries to edit the specific field that is inconsistent, changes to the artifact cannot be saved until the user selects one of the new allowed values for that field.

Changing Text Validation Rules

If you change the regular expression that governs what content is allowed in a text field, each affected field must be fixed the first time a user edits it.

For example, imagine that your ‘Interior decoration’ tracker includes a text field in which users can record the telephone number of a carpet installer. You set up a validation rule like `[(]\d\d\d[)]\s\d\d\d[-]\d\d\d\d` to ensure that only standard U.S.-format phone numbers can be entered.

The tracker has been in use for a week when you realize that email would be a better way to communicate with carpet installers. You change the validation rule to something like `\b[A-Z0-9._%+-]+@[A-Z0-9.-]+\.[A-Z]{2,4}\b`, to ensure that users will enter an email address.

The next time a user tries to update an artifact with a telephone number value in that field, TeamForge warns that the field value is in an inconsistent state. If the user tries to edit the specific field that is inconsistent, changes to the artifact cannot be saved until the user enters a value for that field that matches the new validation rule.

Deleting a Parent Field

If you delete a field that contains values that another field’s values depend on, the dependent field becomes a standard single-select field on its own. No correctionis needed.

Changing ‘requiredness’

- When a field has a parent field that is required, the child field’s default value is set to `None`. If that required parent field is deselected, the child field no longer has to be required, but `Required` remains the default.
- If you require a specific field value before an artifact can be placed in a given status, that field’s children are subject to the same requirements. See Create a Tracker Workflow for more about controlling what status an artifact can be in.

These are some of the frequently asked questions on Wiki.

Why are statistics charts broken on a wiki page?

Statistics charts such as artifact statistics, task statistics, FRS statistics and document statistics, when added to a wiki page, do not get displayed because the relevant markup formats `[sf:artifactStatistics]`, `[sf:taskStatistics]`, `[sf:frsStatistics]` and `[sf:documentStatistics]` have been deprecated from TeamForge 8.1 and later.

So, if you try to create a wiki page with these values, instead of displaying the charts, the following error messages are displayed:

- For task charts, “The Task Statistics are discarded.” is displayed.
- For all the other statistics charts, “The Statistics charts are deprecated viewing from Wiki. It can be accessible from Project Home Page.” is displayed.

How do I generate a wiki table of contents?

You can create a table of contents from any heading text that you have in your wiki page.

For versions 5.2 and earlier, generating a wiki table of contents requires the Wiki TOC plugin, available through CollabNet Professional Services.

To enable TOC for a Wiki page, place the following in your Wiki page at the spot where you want the Table of Contents to appear.

```
%%insert-toc
%%
```

The Table of Contents is generated automatically based on the heading markers in the wiki page. Example: `!!!Heading`.

How do I delete an attachment from a wiki page?

You can delete an attachment from the wiki page.

Use the following steps to delete an attachment:

1. Browse to the wiki page that contains the attachment and click **Edit**.
2. Update the wiki page, even though you have not made any changes to it.
3. Click **Show Details**.
4. Click the *Attachments* tab.

5. Select the attachment you want removed and click **Remove**.

How can I detect orphan wiki pages?

No. You cannot detect orphan wiki pages.

Unfortunately at this time, TeamForge does not have the ability to locate or display orphaned wiki pages within the UI. This functionality is slated to be included in a future release. If you have an immediate need, please contact Technical Support and a suitable SQL query will be devised.

How do I edit the wiki home page?

Navigate to the project home page and click the **Edit** button to edit the wiki home page.

Once you have enabled the wiki as the project home page option in Project Admin, you must return to the project home page by clicking the **Project Home** button. There will be an **Edit** button on the bottom right corner of the page under graphs. You may need to scroll your browser window to the right to see this button.

How do I make the version comment required for wiki updates?

You can use Velocity to customize the pages of TeamForge and make the version comment required.

Create `/opt/collabnet/teamforge/sourceforge_home/templates/body_footer.vm` with the following contents:

```
<script>
var updateButton = document.getElementById('edit_wiki_page.update');
if ( updateButton ) {
updateButton.href = 'javascript:submitWikiPageUpdate();';
}
function submitWikiPageUpdate () {
if ( document.editPage.versionComment.value ) {
submitForm(document.editPage, 'submit');
return;
}
alert("Please include a detailed Version Comment for this change.");
}
</script>
```

Ensure that the file is owned by the sf-admin user:

```
chown -R sf-admin.sf-admin /opt/collabnet/teamforge/sourceforge_home/templates
```

Ensure proper permissions:

```
chmod 0644 /opt/collabnet/teamforge/sourceforge_home/templates/body_footer.vm
```

TeamForge picks up the change immediately.

I have set my project to ‘use wiki homepage’. Why isn’t my wiki showing up?

TeamForge currently uses two distinctly different wikis.

- If there is the wiki you have already edited, which is available by clicking on the **Wiki** button at the top of any project page, and there is the ‘project home’ wiki, which is what you enabled in the **Project Admin** screen.

- If you visit the project home page after setting this option, and if you have the proper RBAC permissions, there should be an **Edit** button in the bottom-right corner of the home page under the graphs. Use this button to edit the project home page wiki.

Why would I attach things to a wiki?

There are specific scenarios that would require attachments to a wiki.

The most common example of when you would attach something to a wiki would be when you need an image in the wiki page and you are concerned that if the image is hosted remotely (a corporate web server, for example), it might be moved or removed. Additionally, you might wish to attach a file to a wiki page if the attachment is only truly important in the context of the wiki page and therefore is not important enough to be uploaded to the Document Manager of TeamForge.

This section provides solutions to common issues with the JIRA adapter.

I just installed the EventQ JIRA adapter and made a commit against a JIRA ticket but the association doesn’t appear.

EventQ is an event system that associates activities across systems. If the JIRA issue you made a commit against hasn’t been modified since installing the EventQ JIRA adapter, EventQ is unaware of the given issue and therefore won’t process an association to it. The solution is to make a nominal change to the JIRA issue. This will automatically register the issue with EventQ and associations will start happening after that.

Associations were working fine but now stopped working or appearing in JIRA.

First, test the connection from the JIRA project to the TeamForge project. In JIRA: **Project administration > TeamForge > Test Configuration**.

If you receive an error, make sure that the credentials used to map the JIRA project to the TeamForge project are still valid. TeamForge credentials can expire, user can be disabled or removed, which would invalidate the integration. The solution is to update the credentials in JIRA and re-test the configuration.

“Test Configuration” fails with authentication errors

Try the following troubleshooting steps:

- Check the username and password supplied. Note that this is a TeamForge username and password.
- Check that the Project Home URL is correct and is the intended Project Home URL.
- Check that the user supplied has proper permissions for the TeamForge project in question. The user whose credentials are supplied must have API permissions in both TeamForge and EventQ, or is a project admin.

I just re-mapped a JIRA project to a different TeamForge project and made a commit against a JIRA ticket but the association doesn’t appear.

EventQ is an event system that associates activities across systems. If the JIRA issue you made a commit against hasn’t been modified since re-mapping the JIRA project to another TeamForge project, EventQ is unaware of the given issue and therefore won’t process an association to it. The solution is to make a nominal change to the JIRA issue. This will automatically register the issue with EventQ and associations will start happening after that.

I get an error trying to re-map a JIRA project to a different TeamForge project

Check that the user supplied has proper permissions in both the original and re-mapped TeamForge projects. It is not sufficient for the user to have permission in just one of the two TeamForge projects in question as the re-mapping process touches endpoints in both projects.

My associations do not appear. What should I do?

Verify the following:

- Only repositories scoped to the configured TeamForge project will result in associations. That is, make sure the version control repository you're using is a part of the TeamForge project that has been mapped to the JIRA project in question.
- When a TeamForge project is first mapped to a JIRA project, the JIRA issues need to be synced into the TeamForge data store. Association will fail if you attempt to create an association between a commit and JIRA issue that has not yet been synced. Please refer to the Sync Issues functionality.
- Make sure the association syntax is correctly using square brackets. Association syntax is case sensitive.

This section provides solutions to common issues with the Jenkins adapter.

How do I get the queue server hostname and credentials? I need it to set up my adapter.

Adapters use a message queue server to transmit data to TeamForge EventQ. Adapters must therefore be configured with the queue server hostname, username, and password. When you create a new "Source" in TeamForge EventQ, a unique set of queue server credentials is created for use with your adapters. To create a Source and obtain queue server hostname and credentials, see the Manage Build Source documentation. Note that the same queue server hostname, username and password may be shared by multiple sources; however, the source association must be unique for each source.

My build is marked "Unstable"

Builds are marked "Unstable" if the Jenkins system configuration has not been saved and the Jenkins adapter cannot successfully communicate with TeamForge EventQ or the message queue. The console output for the build will include a message like this: Build information NOT sent: plug-in needs a Jenkins URL

Workaround: As a privileged Jenkins user, navigate to **Manage Jenkins > Configure System** and populate and save the required configuration.

Automated test results for my Jenkins builds are not showing in TeamForge EventQ

TeamForge EventQ relies on Jenkins to supply automated test results. Jenkins has a post-build action called “Publish JUnit test result report” which must be activated for test results to show up in TeamForge EventQ, regardless of whether JUnit is actually used. Workaround: To enable this configuration, add the `Publish JUnit test result report` post-build action to the desired Jenkins job configuration.

“java.lang.NoClassDefFoundError: javax.mail.internet.MimeMultipart” error is displayed while including ‘Collabnet Security Realm’ in Jenkins Security Configuration - Why?

If Jenkins runs along with any other applications such as TeamForge and JIRA on the same server, this error occurs. Hence, it is recommended to install Jenkins on a separate server. If you have installed Jenkins on the same server where other applications are installed as well, make sure that you host Jenkins on Tomcat to get rid of this issue. To know how to host Jenkins on Tomcat, check this wiki page: https://wiki.jenkins-ci.org/display/JENKINS/Tomcat.

Unable to embed Jenkins page into an iframe. How to fix it?

From the Jenkins version 1.532 and later, the Jenkins page is not displayed in an iframe.

Do you need a way to permit Jenkins to be visible in selected iframes? Here you go.

To configure X-Frame-Options:

If you try to load content into an iframe and the **X-Frame-Options** header is set to **SAMEORIGIN**, then the content loading is prevented. To override the default **X-Frame-Options** header settings, you have to perform the following:

- You need to have the Jenkins plugin installed to configure the **X-Frame-Options**. You can download the plugin from https://wiki.jenkins-ci.org/display/JENKINS/XFrame+Filter+Plugin.
- You must have Jenkins administrative privileges to do this. On the **Jenkins Configuration** page, in the **XFrame Filter Configuration** section, enter the value for **X-Frame-Options Options** as `ALLOW-FROM <ctf domain name>`.

For example, on the `http://jenkins.maa.relenghub.collab.net/jenkins/configure` page, enter `ALLOW-FROM http://forge.collab.net` for **X-Frame-Options Options**.

Now, the page can only be displayed in a frame on the specified origin.

This section provides solutions to common issues with TeamForge-Binary integrations.

The TeamForge-Binary integration is down after enabling SSL. What should I do?

TeamForge-Binary integration is found to be down soon after enabling SSL on sites which initially had SSL disabled. You must update the base URL, go URL and end point URLs stored in Postgres to access your Binary servers.

Run the following query with the psql-wrapper script (/opt/collabnet/teamforge/runtime/scripts/psql-wrapper) to update the base URL, go URL and end point URLs.

```
update integrated_application set base_url='https://<url>', go_url='https://<url>', end_point='https://<url>' where name='Binaries';
```

NOTE: In the above query, replace with valid base URL, go URL and end point URL for your site. The URLs must use https as illustrated above.

You may encounter these problems when using the Subversion adapter.

How do I get the queue server hostname and credentials? I need it to set up my adapter.

Adapters use a message queue server to transmit data to TeamForge EventQ. Adapters must therefore be configured with the queue server hostname, username, and password.

When you create a new “Source” in TeamForge EventQ, a unique set of queue server credentials is created for use with your adapters. To find the queue server hostname and credentials, log into TeamForge as a privileged EventQ user and navigate to: **EventQ (from the project navigation bar) > Manage Sources**. The Sources Overview page appears.

1. Select one of the adapters on the left hand side. The ‘Source List’ shows up on the right hand side.
2. From the list of sources, click the **EDIT** button of the desired source.
3. Locate the section named “Adapter Configuration Information”.
4. Copy the queue server’s hostname, username, and password and use them to configure your adapter.

Note that the same queue server hostname, username and password may be shared by multiple sources; however, the source association must be unique for each source.

Commits fail after installing the adapter hook script

After installing the hook script, I see an error complaining about svnlook:

```
Warning: post-commit hook failed (exit code 1) with output:
svnlook: Expected FS format between '1' and '4'; found format '6'
svnlook: Expected FS format between '1' and '4'; found format '6'
Traceback (most recent call last):
File "/home/user/SVN_Edge/csvn-4.0.0/data/repositories/svn_edge/hooks/post-com
mit", line 488, in <module>
raise Exception("post-commit failed: %s" % e)
Exception: post-commit failed: this constructor takes no arguments"
```

Workaround: This problem is caused by having different versions of svnlook installed: typically one in `/usr/bin` and another bundled with Subversion Edge. To remedy, edit the EventQ Subversion Adapter configuration file `post_commit_orc-publish_amqp.conf` and modify the “svnlook” value to point to the binary packaged with Subversion Edge (example: `/home/user/SVN_Edge/current/bin`).

You may encounter these problems when using the Crucible adapter.

How do I get the queue server hostname and credentials? I need it to set up my adapter.

Adapters use a message queue server to transmit data to TeamForge EventQ. Adapters must therefore be configured with the queue server hostname, username, and password.

1. When you create a new “Source” in TeamForge EventQ, a unique set of queue server credentials is created for use with your adapters. To find the queue server hostname and credentials, log into TeamForge as a privileged EventQ user and navigate to: **Manage Sources > Toggle Review > Edit the Source**.

2. Find the **Adapter Configuration Information** section and copy the queue server’s hostname, username, and password and use them to configure your adapter.

   Note that the same queue server hostname, username and password may be shared by multiple sources; however, the source association must be unique for each source.

Where are communication errors reported?

Communication problems with the TeamForge EventQ message queue server — such as inability to access the server URL or incorrect authentication credentials — will be reported in the `Fisheye-Crucible` error log, usually found in `[FISHEYE_INST]/var/log/atlassian-fisheye-YYYY-MM-DD.log`.

These are some of the frequently asked questions on Review Board.

Users are not getting email notifications for review requests and reviews. What should I do?

You must update the Review Board application's E-mail Server Settings with the user name and password used for the `JAMES_RELAY_USER` and `JAMES_RELAY_PASSWORD` tokens. Do this post install or upgrade of TeamForge and Review Board.

✓ Keep the values of the `JAMES_RELAY_USER` and `JAMES_RELAY_PASSWORD` tokens handy before you begin.

1. Log on to TeamForge as a Site Administrator.
2. Select **My Workspace > Admin**.
3. Select **Integrated Apps** from the **Projects** menu.
4. Select **Review Board** and click **Administer**.
5. Click **E-Mail** from the **System Settings** pane.
6. Under **E-MAIL SERVER SETTINGS**, type the `JAMES_RELAY_USER` and `JAMES_RELAY_PASSWORD` values in the **Username** and **Password** fields respectively.
7. Click **Save**.

What are the software requirements for installing Review Board as an integrated application in TeamForge 18.1?

- Review Board can run on RHEL/CentOS 7.4 and 6.9.
- In addition, Review Board needs PostgreSQL 9.6.5. See TeamForge Installation Requirements for more information.

Which version of Review Board does TeamForge 18.1 support?

TeamForge 18.1 supports Review Board 2.5.6.1.

Which repositories does Review Board support?

Review Board supports only Subversion repositories in TeamForge.

How do I manage users in Review Board?

You can manage Review Board users from TeamForge. Whenever you create or edit users in TeamForge, they are synchronized automatically in Review Board.

Can I specify ‘RB’ as a prefix in my project?

No. You cannot specify ‘RB’ as a prefix in your project. The prefix for Review Board must be unique for every project.

Is it possible to grant TeamForge specific-permissions as part of the system generated Review Board administrator?

No. It is not possible to grant TeamForge specific-permissions as part of the system generated Review Board administrator (integrated application specific role).

Can I use the Review Board ‘Search’ feature after integrating Review Board with TeamForge?

No. TeamForge does not support the ‘Search’ feature of Review Board.

What are the additional features available in Review Board after you integrate it with TeamForge?

Review Board uses some of the TeamForge features like object IDs, links, GO URLs, and SVN integration and associations. For more information, see How does an integrated application interact with other TeamForge tools?.

What are the other TeamForge features which Review Board does not support after you integrate Review Board with TeamForge?

Global search, page component, recent history and project template features of TeamForge are not supported in Review Board.

Where can I find the documentation for Review Board?

You can find the documentation for Review Board here.

How do I rectify incorrect pending review count in Review Board ?

You can reset the counters using the following command and restart the Apache server.

```
/opt/collabnet/reviewboard/bin/rb-site manage /opt/collabnet/reviewboard/ReviewBoard-2.5.6.1 fixreviewcounts
```

You may encounter these problems when using TeamForge EventQ.

Certain activities appear grayed-out and I cannot get an activity’s detail information

TeamForge EventQ sometimes learns about certain activities through an association reference before the details of the activity are officially reported by the source. Such activities are termed **Phantom Activities** and appear grayed-out temporarily. Once TeamForge EventQ receives information on this activity it creates a regular Details page and everything goes back to normal. Traceability graphs include these activities, but you cannot click on the commit nodes to get commit details.

Activities may be perpetually grayed-out when TeamForge EventQ is unable to retrieve activity information such as:

- Activity details of events that occurred before TeamForge EventQ was started — TeamForge EventQ begins to gather data from configured sources as soon as those connections are established, but cannot retrieve historical activity data.
- Activities that occurred during system down-time — In unusual circumstances such as power outage or network failure, TeamForge EventQ may never receive information about activities that occurred during such events.

Activities in the Activity stream appear out of chronological sequence

Solve this problem by synchronizing the clocks on servers that report data to TeamForge EventQ. The Activity stream shows activities in reverse chronological sequence. EventQ obtains the time stamp for activities from the originating server. Failure to synchronize the clocks of the various servers reporting data to TeamForge EventQ may result in activities that appear out of sequence. On Unix systems, use NTP or similar services to synchronize server clocks.

My custom event handler is not working! What is wrong?

If you get a message in server.log (or events.log) that your event handler could not be parsed, these are some points you might check.

- Did you change the package of your class containing handler code? - **This is not supported**.

- Did you move event.xml to another directory? - **This is not supported**.

- Did you compile your java code with a Java version that is not supported by the JVM running on the TeamForge server? Did you use a very long file name for your JAR file? - **Only 31 characters are supported**.

- Did you include libraries in your JAR file that have already been part of the default TeamForge class path? - **In this case, your libraries will not be picked up**.

- Did you try to reference internal TF classes from packages other than the SOAP namespace? - **In this case, the TeamForge security classloader will reject your JAR file**.

- Did you change the DTD location referenced in events.xml? - **This will not work**.
- Did you try to trigger a follow up action in a synchronous event handler? - **This will most likely result in a time out exception, since your code is already running in a transaction that locks the resources you are trying to obtain a lock for**.
- Did you trigger a TeamForge SOAP call with the session ID passed to your handler which threw an exception? - **See [Using an Asynchronous Event Handler: Trigger Follow-up Events][extend_teamforge.html#asynceventhandler] for an explanation why this does not work.

These are some of the frequently asked questions on SCM and CVS related activities in TeamForge.

What software configuration management tools are available in CollabNet TeamForge ?

In CollabNet TeamForge, users can browse the contents of a project's source code repositories and view detailed information about code commits, changed files, and associations with other CollabNet TeamForge items.

CollabNet TeamForge is not intended to replace your SCM tool. Code must be checked in using Subversion, CVS or GIT.

Who can Access Source Code?

Project administrators can give project members specific kinds of access to a whole Subversion repository or any path within that repository.

Overview

You can specify access permissions for users with a given role at the repository level, or at the path level within a repository.

- When you set a permission for a role on any directory in the repository, all directories and files under that directory get the same permission.
- When you set a permission on an individual file in the repository, there is no effect on the permissions assigned to paths above the level of that file. (A file is just a path that ends with a file name.)

How you use path-based permissions will depend on whether you view permissions primarily as a way to grant access or as a way to restrict access.

Full access, with exceptions

Give your company's own employees commit access to your whole source code base, while allowing developers from contracting firms to commit only to those parts of the code base that they are expected to work on.

No access, with exceptions

Assign all developers "No access" by default, then assign each type of developer access to certain directories and files according to their responsibilities.

> **NOTE:** You can control access to a path or to an individual file. This is different from normal Subversion checkout and commit operations, which are performed on directories but not individual files.

No Access

When you deny all access to a repository for a role, users with that role cannot see that the repository exists, except if:

- The role has "View and Commit" access to some directory within the repository. In this case, users with this role can see the directories that contain the directory they have access to.
- The user has another role that grants access to some part of the repository.

> **NOTE:** An individual user can have multiple roles. When two roles have permissions that conflict with each other, the role with the more expansive permissions takes precedence.

View Only

Users with a role that has "View Only" access to a path can browse the contents of the repository on the Web site or by connecting directly to the repository from a client, such as Tortoise, CollabNet Desktop for Eclipse, or CollabNet Desktop for Visual Studio.

View and Commit

Users with a role that has "View and Commit" permission to a path can browse and download code, and can also check code into the repository.

Wildcard-based Access Control and Path-based Permissions

You can create rules that use wilcards to control access to specific paths in a repository. See Wildcard-based Access Control and Path-based Permissions in TeamForge for more background information about wildcard and path based permissions.

Scenario 1

Controlling access to a specific path in a branch using authz rules can be very time consuming and inefficient. Assume you have three branches, `[/branches/foo/build]`, `[/branches/bar/build]`, and `[/branches/baz/build]`. If you create `authz` rules, you may have to write a separate rule for every branch in the repository, let alone the fact that you need to write such rules for a branch you may create in the future.

```
[/branches/foo/build]
@build = rw
@dev = r
[/branches/bar/build]
@build = rw
@dev = r
[/branches/baz/build]
@build = rw
@dev = r
```

Instead, you can write a rule using wildcards such as:

```
/branches/*/build
```

Later, you can create and assign roles to users such as “build engineers” and “developers” with “read-write” and “read-only” access permissions respectively.

Scenario 2

In this scenario, assume that there is a particular file or folder pattern that needs access control in a repository. For example, you may want to restrict all users but release managers from modifying `.iso` files. It is impossible to define such a rule using the `authz` file.

With TeamForge, you can write a rule that partially uses wildcards in file or folder names such as:

```
/trunk/build/*.iso
```

This rule applies to all files and folders that end with .iso under the path `/trunk/build`.

Scenario 3

In this scenario, assume that you want to control access to a particular folder no matter where the folder is in a branch. For example, you may want to control access to the “build” folder wherever it is in a repository. You can write a rule using wildcards such as:

```
/**/build
```

The “**” matches any number of folder levels in a repository. For example, this rule applies to the following paths:

```
/trunk/build
/branches/feature1/build
/trunk/external_module/build
/build
```

In addition to these scenarios, you can use wildcards to create rules that suit your requirement. A few examples of how you can create wildcard-based rules:

- `/**/*.iso` - Match any .iso file anywhere in a repository.
- `/branches/RB*` - Match any branch if the name starts with RB.
- `/branches/*/*.txt` - Match all .txt files one level under any branch.

Notes about path-based permissions

If two paths have different permissions, the permissions on the lower-level path take effect. For example, consider a role that has “No Access” set for the path `/branches/version3/users`, but has “View and Commit” access to `/branches/version3/users/vijay`. A user with this role can:

- Check code in and out of the vijay directory.
- Click down through all the directories in that path, including `users`. (Directories that are not included in the user’s permissions are not shown.)

Users with this role cannot:

- Check files in and out of the users directory.
- Monitor commits to users.
- Execute Subversion copy, move or delete operations that involve resources in the users directory (or any other paths where the user has View Only or No Access).

Commit Messages

When users monitor a repository, they receive commit announcements by email that include the resources that their role permits them to see.

Users with a “No Access” role on a repository cannot monitor that repository to receive commit messages by email.

By default, commit emails provide all the details about the commit that a logged-in user can view in the **Source Code** application. A repository owner can elect to have the notification emails omit most of the detail and provide only the commit ID and the committer’s user name.

Toolbar button

If none of the available permissions (**View Only**, **View and Commit**, or **Path-based Permissions**) is selected for any repository, and none of the options under **Source Code Permissions** is selected, users with this role do not see the **Source Code** toolbar button.

I deleted the email that verifies my CVS password. How can I retrieve my password?

You can sync up your TeamForge login and CVS passwords by changing your password in TeamForge.

To change your password

1. Log in and click on **My Workspace**.
2. Click on **My Settings**.
3. Click on **Change Password**.
4. Enter your old password and new password.
5. Click **Submit**.

> **NOTE:** The old and new passwords can be the same if you do not wish to actually change it.

Why don’t the branding repo changes get rendered into UI?

It may be due to the property ‘subversion_branding.repository_base’ pointing to /sf-svnroot instead of the /svnroot directory, which is used by the scm-integration of the csfe installation.

First, check the location of the branding repository in subversion_branding.repository_base=/sf-svnroot’ in `/opt/collabnet/teamforge/runtime/conf/sourceforge.properties`.

If it has to be /svnroot, then add an entry that states SUBVERSION_BRANDING_REPOSITORY_BASE=/svnroot

Then re-create a runtime and restart TeamForge.

Why do we have errors creating or altering repositories and adding or removing users?

The TeamForge SCM Integration server runs an instance of Tomcat and then launches TeamForge inside the Tomcat container.

If you are experiencing issues creating or altering repositories or adding and removing users from repository access, and the other TeamForge integration logs are not providing any clues, you may wish to review the Tomcat log at: `/opt/collabnet/teamforge/log/integration/catalina.out`.

Sometimes, OS-level errors will be flagged into this log and not others. In our experience, it is pretty rare to find something in this log that is not logged elsewhere.

Why is the password and login shell changed for users on my cvs/svn server?

TeamForge uses local user accounts on the SCM server to provide access to CVS repositories via ssh.

If any local user accounts on the SCM server match user names within TeamForge they will be changed.

If you are planning to use CVS in the SCM server, you should ensure that the local accounts do not conflict with the TeamForge user accounts. Adding a prefix to the local user accounts (local_username) would be one way to resolve this and prevent the usernames from conflicting. Alternatively, if you are not using CVS repositories, the CVS integration can be removed altogether.

Why do I get a proxy timeout when I try to view certain SCM pages?

If you are getting a proxy timeout error when you try to view a SCM page, you may need to configure the Apache 2.2 Proxy Timeout to 300 or less in the httpd.conf file.

If you get the following error while attempting to view a SCM page in SFEE:

```
The proxy server received an invalid response from an upstream server.
The proxy server could not handle the request
    GET /integration/viewcvs/viewcvs.cgi/ibe-rules/tags/phases/ibe-rules_09.02
```

```
.0-Ph-200902_test_20090105/
Reason: Error reading from remote server
```

Configure your Apache 2.2 Proxy Timeout to 300 or less in the `httpd.conf` file.

What’s the difference between a ‘managed’ and ‘unmanaged’ CVS server?

You can use TeamForge to control a CVS server, or you can run the CVS server stand-alone, or ‘unmanaged’.

> **NOTE:** Only CVS servers can run unmanaged. Subversion servers are managed by TeamForge.

Managed Server

A *managed server* gets its instructions from TeamForge to which it is integrated with. Use TeamForge for SCM tasks such as:

- Creating repositories
- Adding and removing users
- Enabling or disabling user access to the repositories

Both new SCM servers and those with existing repositories can be managed by TeamForge.

Unmanaged Server

On an *unmanaged CVS* server, you use the CVS’s own utilities to handle repository management, user management, and access permissions.

Add an unmanaged server if you know that all repository and user creation and management must be done manually. For example, set up an unmanaged CVS server when:

- A corporate IT policy prohibits applications such as TeamForge from making any changes to the users or repositories on the corporation’s CVS servers.
- A CVS server has existing repositories that cannot be changed to adhere to TeamForge standards.

Subversion Replication in TeamForge

Here’s some information on how replication could be useful in your TeamForge site and what to consider when you plan to set up a replica.

Why replicate?

Typically, you would deploy a replica for one these reasons:

- **Projects in remote locations with lower bandwidth or higher latency want the performance of a “local” server**.

  Your company has a number of developers clustered together at a remote location. When you install a replica inside the LAN of these developers, they can greatly improve their Subversion performance and keep a lot of their traffic off the WAN. In this scenario, you would probably want a replica in each such location. Keep in mind that a replica can only be a proxy for one master – so if your company has more than one Subversion master server, you may need more than one replica at each location.

- **You want to reduce the load on the master server**.

  For example, continuous integration tools can place a lot of load on the server and moving that load to a separate server can increase the response time for other users. In this scenario you probably only need to add one replica; you’d add it as close to the master as possible so that synchronization is quick. Of course the previous point can be a factor here. If the continuous integration server is at remote location, then you would want to put a replica near the continuous integration server.

Rules for using a replica

To create a replica in TeamForge, you start with Subversion Edge. A Subversion Edge wizard lets you convert the server into a replica of a Subversion server in TeamForge.

When you configure the replica server, you provide the TeamForge username and password to use for the replica. These are the credentials the replica uses when it replicates Subversion content. This user must be added to projects and given permissions to the repositories being replicated. Those permissions also control what parts of the repository will be replicated. So if you have folders that should not ever live on remote servers, you can set up path-based permissions and that content will never be replicated to the server.

If you forget to set up permissions, the replication will fail. However, there’s no real harm done, and once you fix the permissions, you can do it again.

The replica user can be a normal user account – it does not have to be an Admin account. If the replica is set up and maintained by an Operations team, they might want to just use an Admin account so that project teams do not have to worry about adding the user to the project or setting up permissions.

Permissions for end users accessing those repositories will follow the normal TeamForge rules.

How do I move an existing CVS repository into TeamForge?

You can import and manage an existing CVS repository with TeamForge.

Do the following to move an existing CVS repository:

1. Stop CVS access to the old repo.

   ```
   chmod -x /usr/bin/cvs
   ```

2. Tar the old repo.

   ```
   cd /cvsroot/old_repo
   tar zcvf /tmp/old_repo.tar.gz
   cd..
   mv old_repo /tmp
   ```

3. Restore CVS access.

   ```
   chmod +x /usr/bin/cvs
   ```

4. Transfer the repo to the TeamForge CVS server.

5. Create the new repo from within TeamForge.

   1. Browse to your project

   2. Click the **Source Code** button

   3. Create your new repo

6. Untar the old repo.

   ```
   cd /cvsroot/new_repo
   tar zxvf /tmp/old_repo.tar.gz
   ```

7. To synchronize permissions, perform the following steps.

   - Login as a TeamForge site admin.

   - Click the **Admin** link.

- Click the **Integrations** button.
- Check the CVS integration you want.
- Click the **Synchronize Permissions** button.

8. Verify the new repo.
9. Remove the old repo.

```
/bin/rm -r /tmp/old_repo
```

How do I move an existing SVN repository into TeamForge?

If you have an existing SVN repo, you can manage it with TeamForge.

To move an existing the SVN repo, perform the following steps:

1. Stop SVN access to the old repo.
2. Dump the old repo.

   ```
   svnadmin dump /svnroot/old_repo > /tmp/old_repo.dmp o mv /svnroot/old_repo
    /tmp
   ```

3. Restore SVN access.
4. Transfer the repo to the TeamForge SVN server.
5. Create the new repo from within TeamForge.
6. Browse to your project and click the **Source Code** button, then create your new repo.
7. Load the old repo.

   ```
   cat /tmp/old_repo.dmp|svnadmin load /svnroot new_repo
   ```

8. To synchronize permissions, perform the following steps:
   1. Login as an TeamForge site admin.
   2. Click **Admin**.

    3. Click **Integrations**.

    4. Select the SVN integration you want.

    5. Click the **Synchronize Permissions** button.

9. Verify the new repo.

10. Remove the old repo.

    ```
    /bin/rm -r /tmp/old_rep
    ```

How do I enable Neon debugging in my Subversion client?

This is easliy done in a Linux environment by editing the existing servers text file. On Windows it is not easily done and not recommended.

On a Linux system, edit the `~/.subversion/servers` file by adding the line “neon-debug-mask = 130” (without quotes) to the [global] section of the file, making sure that you also un-comment the [global] line as well. Once Neon debugging is enabled, you should see much more output from each svn command.

Although Neon debugging is possible on Windows, it involves steps that are too complex for most end users to undertake, including compiling Windows binaries for the specific platform and manually handling any errors that arise during that process.

These are some of the frequently asked questions on TeamForge-Git integration, Git history protection and so on.

Where can I find more information about Gerrit?

For more information on Gerrit, see the Gerrit Community Documentation page.

How can I log into Gerrit?

If your administrator has set up Gerrit as a linked application to Teamforge, you will automatically be logged into Gerrit (SSO) when you click its link. If not, access the URL `http(s)://<yourtfinstance>/scm integration server>/gerrit/` and provide your TeamForge credentials.

What are the Git protocols that work with the Git repositories managed by TeamForge?

The Git integration currently allows you to access a Git repository using SSH. That said, you must have generated an SSH key pair and uploaded the SSH public key to Teamforge in **My Settings > Authorized keys**.

Alternatively, you can use http(s) to clone and push to Git repositories. In this case, you can authenticate using your TeamForge user name and password.

How can I use http(s) for accessing Git repositories?

The clone URL for http(s) access follows this convention:

```
git clone https://$USERNAME@<yourtfinstance/scm integration server>/gerrit/p/<TFreponame>
```

When you run the above command on your Git client, you will be asked to provide your credentials. Use the same credentials you use to log into TeamForge's web interface.

How do I generate an SSH key pair?

You can generate an SSH key pair on a Unix machine by running the following shell command:

```
$ ssh-keygen -t rsa
```

You will be prompted to provide the location to store the key pair. The default is the home directory of the logged-in user.

After installing a Git client, I am able to clone a Git repository into my local work directory. However, I am not able to “push” anything to the remote repository in spite of having view and commit permissions. What should I do ?

Right after you clone, but before you commit any changes locally, you will need to configure Git if you haven't already.

```
$ git config --global user.name "<TeamForge username>"
$ git config --global user.email "<email used in TeamForge for the user>"
```

You should now be able to push your changes.

Is a commit association created in TeamForge after I push my commit to a remote Git repository?

Yes, when you push a local commit to the remote repository, an association will get created if the commit message contains a reference to a TeamForge item such as a tracker artifact, wiki or document in square brackets, for example [artf1234].

> **NOTE:** A commit association will not be created if you push your commit to Gerrit's `review branch` (push for review). It will be created once the change is merged into the real branch.

What happens if the TeamForge site is down or there are some network problems–will the Git integration still work?

The Git integration still works, but with the following limitations:

- If the TeamForge site is down, users will not be able to see commit associations created in TeamForge, but still be able to push commits to a Git repository.
- If the Git integration is hosted in LOCAL mode, network-related problems would definitely prevent changes being pushed to a Git repository.
- If the Git integration is hosted in REMOTE mode, the synchronization of roles and permissions will be cached during the period when TeamForge is down; Git will function with the roles and permissions synched already.

What is a “Jumbo Push”?

In contrast to Subversion, Git has the concept of local commits that stay in the local environment of a user, and at some point, get pushed to a remote repository all at once. This push checks in changes from all commits into the remote repository. For each of those commits, a commit object appears in the TeamForge (Source Code component). So, one push can have an unlimited number of commits and thus commit objects in TeamForge. You can, however, define the threshold for a single push based on how many commits should generate a commit object. A push' containing commits beyond that threshold is called a “Jumbo Push"'.

You can configure the Jumbo Push threshold by updating the site option token, GERRIT_GIT_PUSH_THRESHOLD in the site-options.conf file. You have to run the post-installation script after rebuilding the runtime environment. When the Git and Teamforge are hosted on the same server, the runtime involves TeamForge downtime.

What objects and relationships are mapped between TeamForge and the Git integration?

See the README (APPENDIX, Relationship and Object mapping section) or Mappings Between TeamForge and Gerrit.

When are the objects and relationships synchronized between TeamForge and the Git Integration?

TeamForge project roles, project role SCM permissions, global groups, SCM repositories, and global group/ project role membership are synched in two ways:

- **Synchronously**: after a regular interval (configurable using the post-installation script).
- **Asynchronously**: whenever there is a change related to roles or permissions within Teamforge, it triggers the sync between TeamForge and the Git integration.

TeamForge repositories are only synched if there is at least one project role with SCM permissions present in the corresponding TeamForge project.

TeamForge users are provisioned in Gerrit whenever you:

- Change their authorized keys in TeamForge.
- Log into Gerrit by clicking the linked application link or using TeamForge user name and password

- Access GitWeb (web interface for a Git repository) by clicking a Git repository link in the TeamForge Source Code page

> **NOTE:** Changes in Gerrit are not synched back to TeamForge.

Where can I find system logs for the Git integration?

You can find the logs under /opt/collabnet/gerrit/logs/.

Can I bypass Gerrit and access a Git repository directly?

No, Gerrit is used to enforce TeamForge access permissions.

I deleted a TeamForge Git SCM repository but the corresponding Gerrit project does not get deleted. What's wrong?

1. Delete the Git repository in TeamForge.
2. Go to the Git repository project in Gerrit.
3. Go to **Projects > General** and delete the Gerrit project.

You can delete the repository even if there are open changes (repository is permanently deleted) with an option to preserve the repository, if required. Select **Preserve Repository** if required.

How can I import an existing Git repository into Gerrit?

You can import an existing Git repository into Gerrit as a project admin from a local machine or as a System Admin from the server.

Option 1:

If you are project admin, create a new repository and configure your account so that it has at least Delete/View SCM permissions for the one in TeamForge. Clone your existing repository and force push its content to the empty TeamForge repository:

```
git clone --mirror [url of repo to be imported]
cd reponame
git gc
git remote add dest [url_to empty TeamForge Git repository]
git push -f --tags dest refs/heads/*:refs/heads/*
```

Option 2: If you are site admin, create a new repository from the TeamForge UI and perform the following steps from the server as a Gerrit system user to import a repository from the source into TeamForge:

```
# su gerrit
$ cd /tmp
$ git clone --mirror [url of repo to be imported]
$ cd reponame.git
$ git gc
$ git remote add dest file:///gitroot/reponame.git/
$ git push -f --tags dest refs/heads/*:refs/heads/*
```

> **NOTE:** When importing a repository previously hosted on a different Gerrit server, do not push the review branches as Gerrit numerical change numbers are not globally unique and duplication will result in wrong email notifications and problems submitting open reviews.

In the TeamForge web interface, I see the repository root parameter for Git set to “/tmp”. Can I change that?

For backward compatibility reasons, this parameter has to be set to “/tmp”. It does not affect where Gerrit actually stores its Git repositories–this is at `/gitroot`.

Do we have default hook scripts available for Git in TeamForge?

Associating artifacts based on commit messages and blocking commits without a commit message is a core TeamForge mechanism that is supported by Git as well. To add hook scripts, see Gerrit Code Review–Hooks.

Do we have email alerts for Git in TeamForge? If yes, where do we configure it?

Email alerts based on TeamForge commits is a core TeamForge feature, independent of the SCM involved. In addition, Gerrit sends out review emails using the SMTP server specified during installation (it defaults to the TeamForge SMTP server). The mail template is explained in Gerrit Code Review–Mail Templates. The blog post explains how to send information on Git pushes to Teamforge forums (which act as mailing lists too).

Do we have Role Based Access Control and Path Based Permissions for Git in TeamForge?

We support all SCM permission cluster options for TeamForge project roles, default access permissions, project admin permissions along with the site-wide roles and site admin permissions. However, path-based permissions are not relevant in Git since a Git commit always contains all files. If we did not ship certain files, this would result in a checksum error. Gerrit supports branch-based permissions though.

For more information on branch-based permissions, see CollabNet's blog post.

What is history protection?

History rewrites are non-fast-forward updates of remote refs and associated objects. History rewrites happen when a branch in a remote repository gets deleted, previously pushed commits get amended/tree filtered and forcefully re-pushed, or a remote branch/tag is pointed to an entirely different commit history.

Is it possible to turn on history protection for all Git repositories hosted on a Git integration server? If yes, how?

History protection is enabled by default in TeamForge 17.4 and later. However, site administrators can disable history protection at the site level if need be, after which project administrators can choose to have history protection enabled or disabled for individual repositories.

> **IMPORTANT:** The GERRIT_FORCE_HISTORY_PROTECTION site-options token is no longer supported in TeamForge 17.4 and later. Remove this token from the `site-options.conf` file while upgrading to TeamForge 17.4 or later.

I've enabled Protect History for my TeamForge project's Git repository. Will this be effective immediately?

To enable history protection immediately, a TeamForge user with the Source Code Admin permission must do this right after selecting the **Protect History** check box: temporarily remove any user with a project role with any SCM permission, and then add that user back. This will trigger an immediate sync after which history will be protected for the Git repository. Otherwise, history protection will be enabled after a periodical sync.

Can I turn off history protection for any particular Git repository when it is enabled server-wide?

No, when history protection is enabled server-wide for the Git integration server, it cannot be turned off for a particular Git repository.

Where can I see the backup branches generated by history protection?

Backup branches are generated based on the type of History Rewrite. For a remote branch that is deleted, this is under `refs/delete`. For a non-fast-forward push, this is under `refs/rewrite` with the branch name containing the timestamp, original branch and the user who rewrote history, for example, `refs/delete/20121112042512-test--david`.

Who can resurrect or permanently delete backup branches?

A user who is a member of the Gerrit Administrator group can resurrect or permanently delete backup branches. By default, the TeamForge site administrator whose credentials are used for running the post-installation script is part of the Gerrit Administrator group.

Who can see backup branches?

By default, a TeamForge user with SCM View (or more) permission can see all backup branches by executing `git fetch && git ls-remote origin`. In Gerrit, the user must be part of a group which has at least read access for `refs/delete` and `refs/rewrite` for the given Gerrit project (TeamForge Git repository).

Are the backup branches under `refs/rewrite` and `refs/delete` protected from Git garbage collection which removes unreferenced objects?

Yes, objects in backup branches under `refs/rewrite` and `refs/deleted` are referenced and cannot be cleaned up by Git’s garbage collection.

Do backup branches take up a lot of disk space on the Git server?

The backup branches on the Git server are mainly Git objects that are compressed deltas of original file versions. Git regularly compresses these objects to save disk space.

What is the difference between History Protect and Git reflog?

In Git, reflog records all activity on a branch, while History Protect only reports deleted branches/tags and history rewrites (non-fast-forward pushes) For more information, see Git reflog vs TeamForge-Git integration History ProtectGit reflog vs Perforce History Protect.

Which ports does the Git Integration use? My organization has a strict firewall policy, and I need to know which ports to make available for the Git integration.

The Git integration uses 3 ports: 9080,9081, and 29418. See the README file for more information. For the integration, Git integration uses 3 ports(9080,9081,29418 follow details in README) Only port 29418 should be exposed by the firewall.

I ran Gerrit manually (without the service script; now my secure `config` file is gone and Gerrit does not start up. What happened and how can I fix this?

If Gerrit is run with a different Unix user than `gerrit`, newly created and modified files may not belong to the `gerrit` user any longer. As a consequence, when you try to restart Gerrit using its services script (which switches to the gerrit user), Gerrit might not start up due to wrong file permissions. If Gerrit detects that the permissions of its secure config file have been tampered with, it even removes this file. You should, therefore, only run Gerrit using the service script provide, and reconfigure it by running the post-install script again. You can fix incorrect permissions by running the following (as sudo or root):

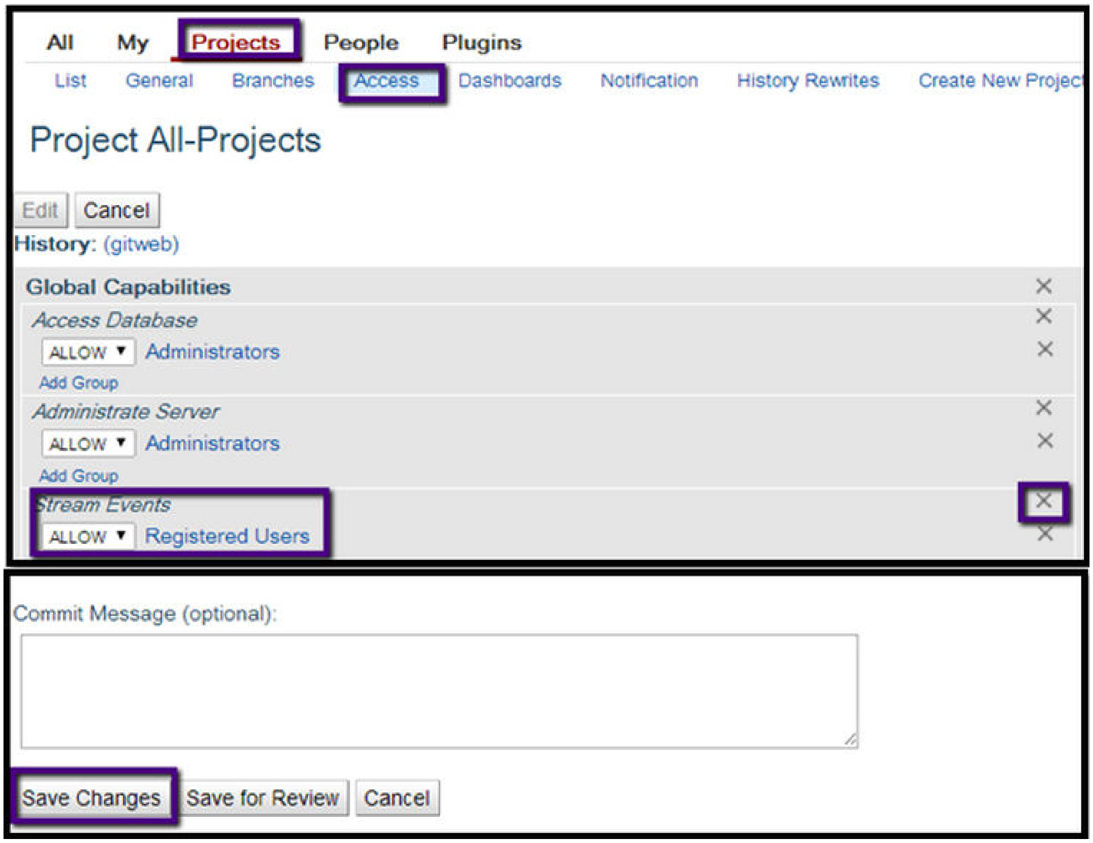
```
chown -R gerrit.gerrit /opt/collabnet/gerrit
```

I deleted the dedicated Teamforge Gerrit user account in TeamForge, and SCM permission synch is no longer working. How can I recover from this situation?

The easy, and recommended, approach is to ask CollabNet's Professional Services to undelete the TeamForge user in question. Otherwise, you would have to create a new dedicated site admin user in TeamForge, shut down Gerrit, re-run the post-install script and provide the credentials of that user. Then, you would have to start Gerrit again, and log in with as that new user via the web interface. You will see that the user does not have any special admin permissions. If you still have a working user in Gerrit's administrator group, you could add the dedicated Gerrit user to that group using Gerrit's web interface. If not, you would have to manually add the new user to the Gerrit administrator group by shutting down Gerrit, removing all files from its caching directory, inserting the `user id` of the new user into Gerrit's Postgres `reviewdb DB` group/user membership table, and starting Gerrit again. Since this probably requires you to consult CollabNet's Professional Services as well, we strongly recommend the previous option (undeleting the previously removed user).

Why does the Registered Users group has StreamEvents capability in Git Integration v8.1.x by default?

- As Gerrit 2.7 Stream Events capability is required for user whose account has been used to monitor Gerrit Events on repositories hosted on Gerrit.
- As Jenkins Gerrit Trigger plug-in uses such a capability to monitor Gerrit events.
- If you have been using **Jenkins CI with Gerrit Trigger** plug-in to automatically verify code review request and already upgraded to TeamForge-Git Integration v8.1.x, the **Registered Users** group has been given this capability by default. Therefore, you do not need add this capability manually and your Jenkins CI with Gerrit Trigger continues to function as usual.
- In case you're using **Jenkins CI with Gerrit Trigger** plug-in, you may remove the **Stream Events** capability as a user who is part of the **Gerrit Administrators** group.
    - To remove Stream Events:
        - Log on to Gerrit web UI.
        - Select the **Projects** tab.
        - Select **All Projects > Access**.
        - Click **Edit**.
        - Delete the **Stream Events** drop-down list.
        - Click **Save Changes**.



Post install fails for Git. What should I do?

Sometimes, the post installation fails during the GIT setup. In such a scenario, you need to manually kill the process using the following command and then do a post install again:

```
kill -9 <ProcessID>
```

These are some of the frequently asked questions on search operations in TeamForge.

What resources can be searched on a TeamForge site?

The resources that show up in search results depend on the context in which you are working.

Here is a summary:

Tool	Searchable resources
Projects	Project id, title/description, created by, project status

Project categories	Title, description
Discussions	• Web UI: author, subject, content, attachment. • Email: we index the original email after we process it. • Topics: author, title and description. • Forums: title, description, author.
Documents	• Document folders: title, description • Documents; version comment, title, status, description, the attachment itself (all versions), authors
SCM	Commit message, title, author
Tracker	• Tracker title, description, author • Artifacts: title, group, category, customer, status, description, authors, tracker, all text flex fields, single-select fields, multi-select fields • Artifact attachments: the attachment itself and comments
News	Body, title, author
File Releases	• Packages; title, description, author • Releases: title, description, author, maturity, status • Files: description, filename
Tasks	Title, description, authors, planned
Users	Username, full name, email, status, details
Pages	

	• HTML components: title, content • Subpages: page title, component title
Wiki	Content of wiki page, using wiki syntax

Why do I get a server status error when I perform a search?

Occasionally, an exceedingly large or complex document causes the search indexing service to abort.

This is typically when all searches in TeamForge return an exid to the user.

Check the server status page and see if the search server is listed as anything other than OK. If it is not OK, then you should restart the search service by logging into the TeamForge application server as root and issuing the following command:

```
/opt/collabnet/teamforge/james/james-/bin/phoenix.sh restart
```

Check the server status page again in TeamForge and ensure that it shows a status of OK. If it shows OK, then searches should now work, and the site will slowly catch up on any indexing requests that were logged while the service was down. If you continue to get exids returned for all searches even with an OK status, then you probably have corrupt search index files.

These are some of the frequently asked questions on Lab Management activities in TeamForge.

Can I get more systems for my project?

If your project needs more systems allocated to it, you must contact your Domain Administrator to increase your limit and allocate new systems to your project.

Your Domain Administrator will usually be a known support contact in your organization.

If additional systems are available in other projects, your Domain Administrator will be able to re-allocate those resources to you. If there are no free systems available, and none that can be reassigned, the Domain Administrator will be able to request additional systems from CollabNet, and then assign them to your project.

What happens when I move a machine between projects?

When you move a machine from one project to another, the machine is not automatically rebuilt. It still runs the same profile and all the same software it was running in the original project.

There can be reasons for wanting to move a machine between projects without rebuilding it. For example, due to a reorganization, the same users and hosts may be providing the same service, but under a different department or project.

Moving a machine does not reboot the machine or stop any running processes

If the old project was running processes on the machine that you do not want, you will have to stop them manually.

Moving a machine changes the access rights on the machine, but does not end existing login sessions

Once the machine has changed projects, after a few minutes, the authentication will have changed so that users from the old project can no longer log in and users from the new project can now log in.

Existing user login sessions, however, are not terminated. Those users will not have root (UNIX) or Administrator (Windows) access any more.

Moving a machine does not reset the contents of the “localadm” group.

The *localadm* group is used to maintain a list of users on the system who are local administrators. This group is not emptied when a machine moves projects. It could contain users which you do not want as administrators on the host in the new project.

What is an audit log?

Every action performed by the user in the TeamForge Lab Management system is recorded in the Audit Log.

For example, when a host is rebuilt using a profile, these are some of the details captured in the Audit log:

- The old profile.

- The new profile.
- How long it took to complete the rebuild process.

What is the correct procedure for modifying a hosted Lab Manager profile?

All profile modifications must be done through the Lab Management UI, under **Administration > Manage Profiles**. Lab Manager profiles should not be directly modified and changes should not be committed to subversion.

To modify your profile, follow these steps:

1. In the browser, login to https://mgr.cubit.domain.com/.
2. Click on **Administration**.
3. In the left pane, click on **Manage Profiles**.
4. Click on the profile (your_profile_name).
5. Click on the *Packages* tab and choose your options.

What is port forwarding?

Use port forwarding to let TeamForge Lab Management hosts connect to machines on other networks.

While TeamForge Lab Management is a secure and isolated environment, occasionally there are valid reasons to let other traffic in and out of TeamForge Lab Management.

Port forwarding consists of routing traffic from one network port on one host to another (same or different) network port on another host.

Take great care when exposing network services running on TeamForge Lab Management hosts to the outside world. Acquaint yourself with your organization’s security policies, or develop them if you don’t already have them, and make sure all services that are exposed comply with these policies. Even better, work with CollabNet to establish access controls around the access to TeamForge Lab Management, so only authorized hosts from within your enterprise – and trusted partners – can access TeamForge Lab Management.

> **IMPORTANT:** CollabNet strongly recommends that any service that can yield a shell account on a TeamForge Lab Management node such as SSH not ever be port forwarded to outside the TeamForge Lab Management environment using this interface.
>
> For CollabNet Hosted customers: CollabNet performs regular scans of its network, and if we see a dangerous or vulnerable network service configured, we may take any steps necessary to protect the

> overall security of the TeamForge Lab Management customer environment, including disabling the offending network service and the associated port forwarding.

What is involved in administrating profiles?

If you are a project admin in TeamForge Lab Management with profiles assigned to your project, or if you are a TeamForge Lab Management domain admin, it is important that you have an understanding of how to administer profiles.

> **IMPORTANT:** It is essential that you first have a solid understanding of how to retrieve and interpret profile details. Making changes to profiles without understanding the effects of what you are doing can cause disruption to your own project, and if the profile you are administering is a public profile, possibly many other projects that use this profile as well.

Profile definitions are versioned, which means the history of changes to profiles is preserved. This allows users to build with various versions of a profile, go back to earlier ones if a new one doesn't work for them, and upgrade to a newer version when available. As an administrator, you have the power to set descriptions on profiles and individual versions, block the use of certain versions, and so on.

Within the Profile Library, you can reach the Profile Admin page by selecting a profile in the Profile Library, and clicking on the *Admin* tab.

Properties can be version-independent or version-dependent.

- Version-independent properties are common across all profile versions. They do not vary, regardless of the profile's version. These are basically the settings that are configured when the profile is first being added to Lab Management with profiles assigned to your project, or if you are a TeamForge Lab Management. They are displayed at the top of the Profile Admin screen.
- Version-dependent properties are associated with a particular version of a profile.

  > **NOTE:** These properties, when set, will apply to the version in question, as well as to any subsequent version. Setting the property on a later version will overwrite the property on earlier versions.

  These properties are displayed below the version-independent properties on the Profile Admin screen. Most of the properties that you can edit are version-dependent properties.

Only one version's properties are displayed at any given time for editing. To view multiple versions at the same time, use the Profile Details page.

Version-independent properties

> **NOTE:** Some version-independent properties, set at profile creation time, cannot be changed.

There are three version-independent properties that you can change:

Summary

This is a brief summary of the profile. You can enter a more detailed description for each version in the properties section.

Project

The project that the profile belongs to. If you select “no project” using the “-“ option, the profile will belong to the domain. The **Project** selection box is only available to domain admins.

Is Public?

This setting governs whether the profile is usable by all profiles, or just to members of this project.

Version-dependent properties

There are many version-dependent properties that you can change for each profile. Some of these properties include:

- Those that specify hardware requirements needed to build this profile. This will make sure a profile is assigned to a host that can actually successfully be rebuilt with the profile.
- An option to specify whether the profile version can be used or not. Useful for marking a profile version “bad” so it cannot be inadvertently selected by users.
- The logo to associate with the profile version. This logo will be displayed in many locations throughout the system to easily identify profiles. Use the link to the Profile Logo Gallery to add and maintain logos.

  > **NOTE:** When working with these properties, be sure you are operating on the correct version!

Description

A description for this version of the profile. Try to put some informative text here, so potential users of this profile will have some guidance as to what this version contains, or how it is different from other versions.

Tag

A symbolic name for a profile version. Tags can be used to make versions easier to remember, and can be moved around between profile versions, similar to a “tag” in Subversion or CVS. Valid characters for Tags are: letters A-Z, numbers 0-9, and underscores ; although a profile cannot be all numeric, and must contain at least one non-numeric character. The Tag HEAD is reserved, and always refers to the latest version – buildable or not – of the profile. Tags are commonly used by project admins and other project leaders to instruct their users about the proper versions of profiles to use.

Can new systems be built with this version of the profile? (also known as “Buildable”).

For any number of reasons, you may wish to restrict profiles so that one or more versions of that profile are not buildable. For example, you may wish to force your users to always use the latest version of your profile: this would be easily accomplished by making all the profile versions not buildable except the most recent. You can change the buildability of a version at any time.

Icon

You can choose from any of the available icons for your profile, although a profile icon is strictly optional. Icons are not private to your project, and are shared among the whole domain, so do not upload anything too secret (or naughty!).

Buildable CPU Types

The types of CPU that can be used to build the profile, for example, “Xeon” or “UltraSparc IV”. Setting this property is strictly optional, even if the profile has CPU type restrictions.

Buildable CPU Archs

The CPU architectures that can be used to build the profile, for example, “x86_64” or “sun4v”. Setting this property is strictly optional, even if the profile has CPU architecture restrictions.

Buildable Number of CPUs

The number of CPU’s required to run the profile. Setting this property is strictly optional, even if the profile has restrictions around the number of CPU’s it can use.

Buildable Hardware Models

Specific hardware models which are required to run the profile, for example, “PDP-11”. Setting this property is strictly optional, even if the profile has hardware model restrictions.

Size (in GB)

The minimum hard disk size, in gigabytes (GB), required to install and run the profile.

Who controls which profiles can be used in a project?

As a project administrator, you can control which operating system profiles the users in your project can build hosts with.

TeamForge Lab Management’s Profile Library gives ownership of profiles to individual projects, or to the entire domain. Profiles can either be public (available to all projects on the site) or private (available only to the project which owns the profile).

However, that fact that a profile is public, or belongs to the project, does not necessarily mean that it can be used to rebuild hosts. The profile must be specifically allowed for use in the project by a project administrator before it can be used to rebuild hosts.

Sometimes it is desirable not to restrict an entire profile, but only one or more versions of that profile. Individual versions of a profile can be prohibited by the project administrator of the project which owns the profile using the Profile Admin screen.

If the profile does not belong to your project, you cannot restrict the use of that profile at the individual version level: that decision is made at the discretion of the owners of the profile.

If you are the project administrator of the project which owns a profile, you can change the profile’s public/private status at any time using the Profile Admin screen.

When you prohibit a profile from your project, this has no effect on hosts which are already built inside your project. They will continue to function normally running the profile they were already running. The owner of the system, however, will be unable to rebuild the system with its current profile, or any other profile which is not allowed in the project.

As a project administrator, you may wish to force your users to rebuild their systems once you have prohibited the use of a profile. On the Profile Summary page you can find a listing of all systems that are running each profile. This helps you track down systems running a profile that you have prohibited, and with your project administrator privileges, you can rebuild those systems with a profile of your choice.

How are my project systems being utilized?

Regardless of the number of hosts in a project, in practice it is common to find shortages of free hosts. At the same time, there are almost always hosts which are under-utilized or even completely idle, which could be re-allocated or consolidated.

Finding and reallocating these hosts allows more efficient use of your project infrastructure.

Project-Level Analytics

The *Analytics* tab in your project shows the following metrics. These metrics are the base available across all operating systems that TeamForge Lab Management supports.

- Load Average
- Processes
- CPU
- Logged-in Users

> **NOTE:** More metrics, including host-specific and operating-system-specific metrics, are available for individual hosts when you click on the host name in the results table.

To see a metric, click the name of the metric. You can see these time ranges:

Daily

Approximately the past 24 hours

Weekly

Approximately the past 7 days

Monthly

Approximately the last 30 days

Yearly

Approximately the last 365 days

If a given host says “no value” in any of its columns, this means that TeamForge Lab Management has been unable to collect this data over the requested time interval. If the machine is up, and not collecting data, contact a TeamForge Lab Management administrator to investigate why data collection is not working properly. If the machine is currently down, or was down during the requested time range, you will not be able to get any performance data for those times. There is no way to retroactively “catch up” data if collection is not working properly.

The data used to build each graph and chart presented to you can be exported in CSV (Comma-Separated Value) format, suitable for opening in any spreadsheet application. This allows you to build your own visualizations of the data to complement the ones TeamForge Lab Management creates.

Beating The System: Dealing With the Possibility of Users Generating “Fake” Load To Make Machines Seem Busier

Since we publish the metrics for determining how busy TeamForge Lab Management thinks machines are, it is possible for irresponsible users to generate automated jobs which simulate a busy machine, even if the machine is really not being used for anything.

We encourage all users and administrators of the project to make sure people in your projects understand that this type of behavior is not acceptable and may lead to loss of privileges or other actions against them. Presumably, if someone has a good reason for wanting to keep a machine, it is good for their project, and your organization if they do so.

As administrators, we strongly recommend you take the time to understand and listen to your users’ concerns about how machines are allocated. You may just need more machines in your project, or you may be overzealous about de-allocating machines once they drop below a certain usage threshold. De-allocating machines that seem to be not busy may seem efficient, but if those machines took their users a long time to set up, that might be counterproductive. Perhaps you can reach a middle ground and use virtual machines, or a smaller virtual or physical machine, for that user.

Finally, we always recommend talking to your users and trying to understand how they are really using their machines. The statistics that TeamForge Lab Management gathers are a starting point for managing your software development and testing infrastructure, but not the final word.

What is host URL mapping?

Host URL Mapping in TeamForge Lab Management allows you to access web services running inside the TeamForge Lab Management environment from anywhere, using a simple and consistent URL, with optional SSL encryption services added on.

Host URL Mapping provides three major benefits:

- Maintains a consistent URL even if the service is moved to a different TeamForge Lab Management node or different base URL.
- Provides external access to resources that would otherwise be only accessible inside the TeamForge Lab Management environment. Because Host URL Mapping uses standard HTTP/HTTPS ports, you will not be blocked by firewalls that sometimes prevent port forwarding from working properly.
- Allows you to transparently add SSL encryption to web services running inside your TeamForge Lab Management environment.

If your application mixes absolute and relative URL’s, host URL mapping dynamically rewrites the absolute portions of your URLs to help ensure that your applications display properly when they are mapped. While not perfect, this feature has been tested with a number of web applications and found to be effective.

CAUTION: Take the utmost care when exposing web services running on TeamForge Lab Management hosts to the outside world. Acquaint yourself with your organization's security policies, or develop them if you don't already have one, and make sure that all services that are exposed comply with these policies. Even better, work with CollabNet to establish access controls around the access to TeamForge Lab Management, so only authorized hosts from within your enterprise and trusted partners can access TeamForge Lab Management.

CollabNet strongly recommends that any web service that you expose be password-protected, or otherwise require authentication to access.

CollabNet Hosted customers: CollabNet performs regular scans of its network, and if we see a dangerous or vulnerable web service configured, we may take any steps necessary to protect the overall security of the TeamForge Lab Management customer environment, including disabling the offending service and the associated URL mapping.

NOTE: While you can convert non-SSL URL's into SSL using Host URL Mapping, you cannot map SSL URL's. You can use Port Forwarding to expose SSL URL's outside of the TeamForge Lab Management environment, however.

Why doesn't URL mapping work for me?

My page doesn't look right. Images and graphics are wrong, or functionality doesn't work. The same page looks and works fine when viewed through Port Forwarding.

Host URL Mapper attempts to "clean up" HTML pages which pass through it in order to clean up absolute links. But it is not hard to construct an application that will slip through TeamForge Lab Management's filters and still not properly render all of its referenced objects inside of its pages. We make all reasonable attempts to clean up HTML, but not all applications can be properly rendered using Host URL Mapping. This is especially true for applications which make heavy use of Javascript, ActiveX, Java applets, and other types of rich client-side web programming.

To test this, use either a direct connection to the host (if available) or a port forwarded connection to the host to see if this behavior is present on the original version of the page. If the original page does not have this behavior, the first step is to verify the **Dynamic Rewriting** level is set to *More Aggressive*. If that does not work, please file a support request with CollabNet to evaluate the page.

How does host URL mapping compare with port forwarding?

URL mapping is good if you don't want your connection blocked, while port forwarding is good if you need non-HTTP services.

- Port forwarding is a facility for making any TCP or UDP network service available outside the TeamForge Lab Management environment. Host URL mapping only allows you to expose HTTP-based services.
- A major limitation of port forwarding is that many organizations' firewall security policies prohibit outgoing connections to arbitrary high ports. Because URL mappings all use standard HTTP/HTTPS ports 80 and 443, they will never be blocked. If you can access TeamForge Lab Management itelf, you will be able to access any application configured with host URL mapping.
- Host URL mapping can automatically add SSL encryption to your web services that are not running SSL encryption. With port forwarding, if you want SSL encryption, you must set it up on each host.
- Using host URL mapping, you can expose only a part of a server's URL space. With port forwarding, the entire URL space is visible.
- Port forwarding does not need to rewrite links inside the HTML, so more web applications will work under port forwarding.

How are virtual guests different from physical machines?

Virtual guests work like physical hosts, with some important differences.

Creating a new virtual guest

The process of creating a new virtual guest is very similar to the process of creating a new physical host, with the following differences:

- You must select a virtual host – a physical machine – to run the virtual guest on. You cannot run a virtual guest inside of another virtual guest.
- The virtual guest's project cannot be set independently of the virtual host, and the virtual guest will always be in the same project as its virtual host.
- You are constrained, with hard limits, by the RAM and hard disk available on the virtual host. TeamForge Lab Management requires that each host have a minimum of 512MB of free RAM and 10GB of disk free.
- Even if you are within the hard limits for RAM and hard disk space usage, you are still sharing other resources – notably disk I/O bandwidth, network bandwidth, and CPU cycles – with the host machine

and any other guests already on the virtual host. Before creating a new virtual guest on a virtual host, we recommend carefully examining the virtual host's system performance to make sure it can handle the additional load. Of course, if you do find out later on that performance of your virtual guest is not as good as you would like, you can always migrate the virtual guest to another virtual host.

- You are not as constrained on your selection of MAC address with virtual guests ; you can choose any available address in the allowed range. Valid values for MAC addresses for virtual guests in TeamForge Lab Management are 00:50:56:01:00:01 to 00:50:56:3F:FF:FF. Using anything outside of this range will either result in the host not being reachable on the network, or the host coming in conflict with another MAC address on the network.
- You cannot specify the architecture or chip type for the virtual guest, since those properties are inherited from the parent.
- You do not need to specify a Lights Out Management IP address for the virtual guest, since the IP address used to manage the guest is always the IP address of the virtual host.

Disk size

While disk space is allocated to virtual guests at the time of virtual guest creation, it is not actually occupied on the host until it is needed by the guest. At the same time, TeamForge Lab Management does not keep an up-to-date count of exactly how much disk space is in use on the virtual host.

In practice, this means:

- You will likely have more disk space on your host than your virtual guests would indicate. But it is also possible to have less space than your virtual guests would indicate. For example, let's say you have a 100GB disk on your host, with two virtual guests, each with a 40GB disk allocated. But if you're only using 5GB of that 40GB in each guest, the remaining 70GB in unallocated. But on the flip side, let's say you allocated those two virtual guests at 10GB each, but they were using a total of 90GB on your local disk to store files. TeamForge Lab Management would let you make this allocation, but your virtual machines would crash when you tried to put more than a combined 10GB on them both.
- This translates into freedom in your management of disk space on the virtual hosts: you are free to temporarily "borrow" disk space from virtual guests that is not being currently used and use it for your virtual host.
- Along with this freedom comes the responsibility of being vigilant about maintaining sufficient space on your virtual hosts to always have enough disk space for any guests on the system. Monitor your usage carefully using Analytics page for the host.

Modifying hardware parameters of existing virtual guests

Some virtual guest hardware parameters can be modified after the virtual guest has been created.

Changing disk size

While changing the size of the disk is supported in TeamForge Lab Management, the change will not be reflected until you re-image your guest. And, in certain cases, it is possible for a change in the disk size of your guest to cause the guest to become unreachable, especially if you reduce the disk size. Therefore, we recommend that you change disk size on virtual guest only if you are going to immediately rebuild the guest.

Changing RAM size

Changing RAM size is a very low-risk operation. In order for the change to take effect, the virtual guest must be rebooted after the change is made in TeamForge Lab Management. Reducing the RAM available to your virtual guest can make the system run much slower.

> **NOTE:** In between changing RAM size of the virtual guest and rebooting the virtual guest, you must wait approximately 5 minutes to insure your change is propagated.

Changing the number of CPUs

Virtual guests can support either one or two processors. Two CPUs for virtual guests are supported only if the virtual host has at least two CPUs. Like changing a virtual guest's RAM allocation, changing the number of CPUs in a virtual guest is a low-risk operation, and the virtual guest must be rebooted after the change has been made in TeamForge Lab Management.

> **NOTE:** In between changing number of CPU's for the virtual guest and rebooting the virtual guest, you must wait approximately 5 minutes to insure your change is propagated.

What is involved in migrating a virtual guest?

When you migrate a virtual guest to a different host, there are a few things to keep in mind.

- The user performing the migration must be a TeamForge Lab Management Domain Admin.
- The current virtual host and guest, and the desired new virtual host, must all be in the same project. In other words, you cannot change both the project allocation and the parentage of a host in a single step.
- The target virtual host must have enough RAM, disk, and CPU to accommodate the virtual guest you wish to migrate to it. The rules governing the RAM, disk and CPU needed are the same as if you were creating a new virtual guest on the host.

TeamForge Lab Management calculates these values for you and presents you with a drop-down list of hosts that fulfill all these criteria. But if you are wondering why a host is not a valid potential new virtual host, see the host's current hardware parameters and allocations on the Host/Admin page for the virtual host.

- If your TeamForge Lab Management installation is composed of more than one subnet, both the virtual guest being moved and the target virtual host must be on the same subnet. If you are not sure if your installation has more than one subnet, check with your local TeamForge Lab Management support contact.
- Only one virtual guest at a time can be migrated to a given virtual host, although there is no limit on the number of virtual guests that can be moved from a given virtual host. The reason for this restriction is that moving a virtual guest to a host is a very I/O-intensive operation that will consume all available disk I/O on the destination host. Allowing more than one simultaneous move will not be any faster, and increases the risk of over system instability for the virtual host.
- If more than one virtual guest migration to a given virtual host is started at the same time, they will complete serially. The exact order of the migrations may not correspond exactly to the order in which they were initiated and cannot be accurately predicted.
- Occasionally, after the migration has completed successfully, the virtual guest will require an extra reboot to start up properly.

What's a good way to read log information?

To read log entries, download the log as a CSV file.

You can also filter the logs using the **Filter** option at the top of all audit log screens.

On any log page, you can click **Download all log entries in csv format** to download the corresponding CSV file.

> **NOTE:** The CSV file is formatted in Microsoft Excel format. If you import it into OpenOffice, set **Text Delimiter** to double quotes(").

These are some of the frequently asked questions on customizing the TeamForge site.

What elements of a site can I customize?

You can customize the site home page and the default home page of every project on the site. You can also customize the menu bars, headers and footers of any page.

For more information about branding, see: Customize anything on your site.

How does TeamForge use stylesheets?

The look and feel of much of TeamForge is controlled by cascading style sheets (CSS).

All default CSS styles can be customized to alter the look of the application. You can customize fonts (color, size, font face, etc.), links, backgrounds, headings, tables, tabs, and anything else that CSS can control.

The default TeamForge CSS file is `css/styles_new.css`.

New CSS files can be added to the css/ directory and reference them via `templates/body_header.vm`.

> **TIP:** If you override an existing CSS file, it will be used instead of the default CSS file. So you must be sure to include all the default styles in your customized file. A best practice is to add any new or overridden styles to the bottom of the CSS file so that they can be easily identified.

Can I substitute my own images for the default TeamForge images?

Every image in TeamForge can be edited or replaced by a new image file.

Images are stored in an `images` directory in the branding repository.

> **NOTE:** The default TeamForge image files are included in both the Quick Start and Advanced branding zip file.

Examples

- To replace the masthead graphic, replace the image file `images/masthead/logo.gif`.
- To replace the folder graphic, replace the image file `images/my/all_projects.gif`.

Can I use my own custom JavaScript on my site?

The JavaScript scripts used in TeamForge can be customized.

Existing JavaScript scripts are located in the js/ directory in the branding repository.

New scripts can be checked into the `js/` directory and then referenced from Velocity templates.

Can I customize the web interface?

You can customize the way the web interface looks and functions through the use of Velocity templates.

> **NOTE:** Even though Velocity as a technology is supported by CollabNet, Inc., the customizations themselves are not supported. Any future upgrade of TeamForge may, in fact, break your customization. Furthermore, making these types of changes to your installation is considered a customization and will impede our ability to support you. Should you experience issues and open a ticket with Technical Support, you may be asked to remove these customization to debug your issue.

These are some of the frequently asked questions on email communications in TeamForge.

What are the X-headers used to sort and filter emails sent by TeamForge?

X-headers are special instructions that are added to many email messages. Messages that come from your TeamForge site have X-headers describing the purpose of the email, the event that triggered it, and lots of other information.

You can use these X-headers to sort and filter emails sent by TeamForge.

X-header	Example	Description
X-TeamForge-Application	Tracker	Application, if applicable. Values include Documents, Wiki, Source Code, TeamForge, Tasks, Tracker, Discussion, File Releases.
X-TeamForge-Artifact-Monitoring	true	For artifacts, true if the user is monitoring the artifact directly. If value is false, you are monitoring the folder, not the artifact.
X-TeamForge-AssignedTo	admin	User assigned to artifact or task.
X-TeamForge-CommitPath	/path/to/my/branch	For Subversion commit emails: reports the greatest common path of all files in the commit. For example, if you modify the paths /a/b/file1 and /a/c/file2, the value of the X-TeamForge-CommitPath header is /a. If you modify /a/b/file1 and /file2, the value is / (the root path of the repository).
X-TeamForge-FolderId	tracker1005	If the object is a folder, the object’s ID, or the ID of the object’s container.
X-TeamForge-Forum	New user forum	For posts, this is the forum title.
X-TeamForge-PlanningFolder	/path/to/my/folder	The path to a planning folder.
X-TeamForge-ProjectId	proj1006	Project ID, if applicable.
X-TeamForge-ServerName	collab.net	Hostname portion of TeamForge URL.
X-TeamForge-Status	Open	Status of artifacts, tasks, documents, and document reviews. Values include:

		• For artifacts: Open, Closed, Pending, or user-defined. • For tasks: Not Started, OK, Complete, Warning, or Alert. • For documents: Draft, Review, or Final. • For document reviews: Open or Closed.
X-TeamForge-Type	Artifact	The type of the object, if applicable. Values include Tracker, Artifact, Document Folder, Document, Document Version, Wiki Page, Wiki Page Version, Task Folder, Task, FRS Package, FRS Release, Discussion Forum, Discussion Topic, SCM Repository, SCM Commit.

Why is my email taking a long time to arrive?

TeamForge uses the James MTA to send and parse all email coming to and from the system. In this case, the best course of action is to look in the james mailet logfile.

Your logfile will help you to determine what is going on with the emails that are being sent from your system. Your logfiles will look very similar to this:

```
07/02/07 07:54:43
INFO James.Mailet: ?RemoteDelivery:
Attempting delivery of Mail1170135534355-39088-to-domain.invalid to host domai
n.invalid at 192.168.0.1 to
addresses [invalid.user@domain.invalid] 07/02/07 07:55:43
INFO James.Mailet: ?RemoteDelivery: Could not connect to SMTP host: 192.168.0.
1, port: 25; nested exception
is: java.net.?ConnectException: connection to 192.168.0.1 timed out 07/02/07 0
7:58:43
INFO James.Mailet: ?RemoteDelivery: Storing message
Mail1170135534355-39088-to-domain.invalid into outgoing after 7 retries 07/02/
07
07:58:43 INFO James.Mailet: ?RemoteDelivery: Attempting delivery of
Mail1170831482124-2756-to-company.com to host mx.company.com. at 127.0.0.1 to
addresses
[good.user@company.com]
```

As you can see, the James MTA stores outgoing emails to resend at a later time. These files can be located in this directory: `/opt/collabnet/teamforge/james/james-<ver>/apps/james/var/mail/outgoing/`

When you do a directory listing of the files, you will see a listing of files very similar to this:

```
4D61696C313137303833323131343031322.Repository.?FileObjectStore
4D61696C313137303833323131343031322.Repository.?FileStreamStore
```

The FileObjectStore is a binary file, but, the FileStreamStore can be viewed with an editor or your favorite paging program in order to determine the contents. Sometimes, the directory can grow to a large number, where you will not be able to use a standard bash expander to delete all of the files. In that case, use the following shell script to remove all of the objects from the outgoing directory:

```
for i in * ; do /bin/rm $i; done
```

Why would some users not get email?

Check your dnsserver-.log.

James records all errors related to resolving DNS for outbound mail to: `/opt/collabnet/teamforge/james/james-<version>/apps/james/logs/dnsserver-.log`. If you find that some of your TeamForge users are receiving email, but significant groups of others are not, you should consult this log to determine if James is experiencing difficulties in resolving their domain or MX records.

Why do search and email server show “Could not connect”?

Typically this means the tomcat container for James and the search service are not running.

You can restart this with the commands shown below. You may need to set the JAVA_HOME environment variable to the location of your JDK.

```
sh /opt/collabnet/teamforge/dist/james/james-2.2.0/bin/phoenix.sh stop
sh /opt/collabnet/teamforge/dist/james/james-2.2.0/bin/phoenix.sh start
```

Why can’t TeamForge send my outbound mail?

If you are unable to send email directly due to firewall restrictions, or if mail is being rejected by the application server’s IP address, configure TeamForge to send outgoing messages through a gateway mail server.

Configure TeamForge to send outgoing message through a gateway mail server by adding the following to the `<mailet match="All" class="RemoteDelivery">` directive in the configuration file at `/opt/collabnet/teamforge/runtime/james/apps/james/SAR-INF/config.xml`:

```
<gateway>smtp.example.com</gateway>
<gatewayPort>25</gatewayPort>
```

If your gateway mail server requires authentication to send email, you may also add the following directives:

```
<username>username</username>
<password>password</password>
```

Can I customize my site’s email notifications?

The default text for CollabNet TeamForge email notifications can be customized.

Email templates are located in the `templates/mail` directory in the branding repository.

Some email templates that are commonly customized are:

- user_welcome.vm
- account_request_rejection.vm
- project_approve_pending.vm
- user_forgot_password.vm

Email template formatting follows the standard Velocity conventions:

```
user_welcome.vm
##subject
Welcome to CollabNet TeamForge  5.0!
##subject
##body
The Welcome message goes here...
##body
Email subject line
Email message text
```

Can I specify an alternate email address?

Yes, of course you can specify one or more alternate email address. TeamForge supports user profiles with one primary email address and up to three secondary email addresses.

The email address specified while creation of a user account is considered the primary email address. The alternate email addresses are optional and can be specified while updating the user profile.

> **NOTE:** To add your secondary email addresses, use the **Admin > Users** page to update your profile.

How can I check if port 25 is open?

If you know the mail server is up and running, check whether you can talk over port 25 to your mail server. This can be done using a one-line command:

```
telnet <appserver name> 25 Substitute the <appserver name> with your own serve
r.
```

Once you type this into your DOS window and hit return, you should see some sort of response from your mail server, as shown below:

Trying 208.75.196.84... Connected to cu190.cubit.sp.collab.net (208.75.196.84). Escape character is '^]'. 220 cu190.cubit.sp.collab.net SMTP Server (JAMES SMTP Server 2.2.0) ready Mon, 27 Jul 2009 06:38:20 -0700 (PDT)

When a discussion forum is set up, do all members receive a notification mail?

Yes. A mail informing users about the creation of discussion forum is sent to all the members of the project who have the Discussions (check box) selected as a monitored application.

Users can enable **Discussions** as a monitored application. To do that:

1. Go to **My Workspace > My Page > Monitoring** and select a project from the **Edit Monitoring Subscriptions and Preferences** pane.
2. Select the *Monitored Applications* tab.
3. Select the **Discussions** check box and click **Save**.

I am unable to edit a specific artifact via email, but I can do so via the web UI. Why is this?

There may be workflow rules applied to the tracker that require specific fields to be set. As you can only define the artifact title and description via email, the artifact creation fails. If you wish to create artifacts in this tracker via email, the tracker admin will need to disable these workflow rules.

In addition, you may not be able to edit artifacts belonging to a tracker via email, if the tracker has two mandatory flex fields with parent-child relationship, but no default value relationship between the default values of the parent and child flex fields. In such cases, it is recommended to always establish the default value relationship between flex fields that have a parent-child relationship.

How do I set up a local alias via James?

In situations where you need to obtain a SSL certificate for your domain, and your SSL certificate provider only permits you to use addresses related to your TeamForge domain, it may be necessary to generate an email alias from within TeamForge. Since there is currently no way to do this through the UI, you’ll have to do it from the James administrative interface.

First, you'll need to connect to the James administrative interface on your system. If you've followed our best practices guide in our knowledgebase, you'll know that you should have port 4555 firewalled to everyone but localhost. SSH to your TeamForge server, and then issue the following command:

```
telnet localhost 4555
```

This will bring up the Remote Administration Tool:

```
[root@app1 root]# telnet localhost 4555
Trying 127.0.0.1... Connected to localhost (127.0.0.1).
Escape character is '^]'. JAMES Remote Administration Tool 2.2.0
Please enter your login and password
Login id: admin Password: (text is echoed locally)
Welcome admin. HELP for a list of commands
```

First, we'll need to add a new user:

```
adduser <username> <password>
```

Then, we'll need to set the forwarding address of that user.

```
setforwarding <username> <email address where you want email to go>
```

Finally, we'll exit the James administrative interface.

```
quit
```

Your changes should be in place.

Does TeamForge support using /etc/aliases for local mail delivery?

No, TeamForge uses the James SMTP server, which does not use the `/etc/aliases` file.

To enable local mail aliases, you will need to configure user mapping in the `XMLVirtualUserTable` in the `/opt/collabnet/teamforge/runtime/james/apps/james/SAR-INF/config.xml` file.

> **NOTE:** Please note that while the James SMTP server is used as part of TeamForge, customizations such as these cannot be supported by CollabNet.

How can I stay informed about events on TeamForge site?

To keep up with changes, you can be notified by an automatic email when an item you are interested in is updated.

Monitoring items lets you stay up to date automatically on all changes without having to log into CollabNet TeamForge and check the status of each item.

You can monitor individual items, folders, and entire applications in each project for which you are a member. You can configure the frequency of email notifications to suit your personal preferences, and suspend monitoring messages entirely if you are out of the office or for any reason do not want to receive messages.

Items that can be monitored include:

- Entire applications, such as all tasks or all documents.
- Folders, such as task folders or document folders.
- All items in a tracker, forum, or package.
- Individual items, such as a task, a document, a tracker artifact, or a release.

You can tell by looking at each item whether you are currently monitoring it and how to begin or end monitoring it.

- For each item, list of items, or folder, you see a **Monitor** button or menu option.
- If you are currently monitoring an item, the **Monitor** option is replaced by a **Stop Monitoring** option, and the monitoring icon is displayed.

Whenever you create an item or edit an item, you automatically begin monitoring that item.

You do not receive monitoring notifications for your own changes to a monitored item.

If you have the appropriate permissions, you can see who is monitoring an item.

You can add other users to the monitored item. After a user is added to a monitored item, the user can continue monitoring the item, configure monitoring preferences for the item, or stop monitoring the item.

I receive a number of emails just because I am monitoring a folder. How can I restrict this?

Monitoring a folder might spam the mailboxes with emails that are generated for every small change.

Email notification preferences under the *Monitoring* sub-tab under My Workspace can be modified to Send Daily Digest Email, which provides a summary of the email alerts.

1. Go to **Projects > My Workspace**.
2. Select **Monitoring** from the **My Page** menu.
3. In **Edit Monitoring Subscriptions and Preferences** page, click **All Projects** and click on **Email notification preference**.
4. For each project, the monitoring folders under various components (Tracker/File Releases/Tasks/Discussions/Wiki) can be customized to Daily Digest Email.

Can TeamForge accept email for more than one domain?

You can configure James to accept email for more than one domain by adding the additional domains to the `<servernames>` section in the `config.xml` file.

Add the domains to the `<servernames>` section of this file: `/opt/collabnet/teamforge/james/james-2.2.0/apps/james/SAR-INF/config.xml`.

Around line 53, you should see the following:

```
<servernames autodetect=""true"" autodetectIP=""true"">
   <servername>localhost↙servername>
↙servernames>
```

You can add other host names for James to accept mail for by adding more `<servername>` blocks. The comments in the config.xml file explain this further. Please keep in mind that these changes may be overwritten by future TeamForge upgrades.

How do I configure TeamForge to send mail on a specific network adapter in a multi-NIC configuration?

When a host has multiple NICs, James will try to do the right thing when sending mail. In some network setups, this is not correct, and manual configuration is needed.

James requires multiple changes to fully configure how it interacts with the network. Open the `config.xml` file, located in `$SF_HOME/apps/james/james-2.2.0/apps/james/SAR-INF/` for version 5.1.

Locate the `<mailet match="All" class="RemoteDelivery">` section. Add a subnode `<bind>$addr↙bind>` where `$addr` is the ip address that James should be sending mail from.

Near that area, there is a `<servernames...>↙servernames>` section. Confirm/change the two autodetect options (`autodetect`, `autodetectIP`) to `false`. Next, add the fully qualified host name, and the ip address that will be used, to their own `<servername>` entry.

After the changes are complete, save the config.xml and restart the application.

How do I send email from a specific sender address instead of the member address?

To send mail from a specific sender address set the *MONITORING_EMAIL_FROM_ADMINISTRATOR* token to *true* in the `site-options.conf` file.

TeamForge uses the member’s email address as the sender address when it delivers the mail. To reconfigure this setting in the `site-options.conf` file, set the *MONITORING_EMAIL_FROM_ADMINISTRATOR* token to *true*, as shown in the following code sample:

```
MONITORING_EMAIL_FROM_ADMINISTRATOR=true
```