

CollabNet Connector Framework 1.3.0 - Quick Start Guide

What is the CollabNet Connector Framework?

The CollabNet Connector Framework (CCF) SDK is an openAdaptor based SDK that allows rapid integrations and migrations dealing with the artifact data shared between different tools in the ALM cycle in combination with the collaborative platforms from CollabNet.

For an introduction to CCF, the underlying technologies, example scenarios, and detailed documentation, please see the ["How to get started"](#) page in the CCF project.

Things to do before you start using the connector

Before you start using the connector, it is recommended that you go through the checklist on the CollabNet Connector Framework project for things to do on your source, target and connector systems.

- [Things to do before getting started](#)

Note: In Quality Center, the connector user and the "resync" user must have TDAdmin permissions. To synchronize data, the CCF uses "commands" to issue custom SQL statements through Quality Center's COM API. These commands can be executed only by a user with the TDAdmin role. Enhancing the role of a non-admin user by providing all available permissions is not enough.

Set up the connector

A quick way to get started with the connector is to try out a sample scenario provided with the installation. The following topics take you through the process of setting up and running the connector for bidirectional synchronization between Quality Center and CollabNet TeamForge tracker.

Note: The process for setting up the connector for some other integration scenario, for example, between Quality Center and CollabNet Enterprise Edition Project Tracker, is largely similar. Differences, if any, are usually pointed out with examples.

Note: In CollabNet TeamForge 6.0 and later releases, Project Tracker can be installed as an integrated application with TeamForge. If you've been using the connector with Quality Center and CollabNet Enterprise Edition Project Tracker, the process for setting it up for Project Tracker in TeamForge, including mapping projects and fields in the two systems, is the same.

1. [Install the connector framework](#)
Get started by downloading and installing the CollabNet Connector Framework.
2. [Register the connector with Quality Center 11](#)
This step is required if you are using Quality Center 11. To enable CCF to connect to Quality Center, you need to install the Quality Center drivers on the system where CCF is installed.
3. [Install the Eclipse user interface](#)
Get the CCF user interface plugin for Eclipse to configure the connector, manage hospital artifacts, view log files and edit log4j settings.
4. [Copy the sample scenarios](#)
When you have installed the connector, it is recommended that you make a copy of the samples provided and use that as your production scenario.
5. [Rename the Log4j configuration file](#)
If you are using CollabNet Connector Framework version 1.1.1 or later, you must rename the log4j.xml.rename_me file in your installation to log4j.xml before you start using the connector.
6. [Start the database](#)
Before you can begin configuring the connector, you need to start the CCF database.
7. [Configure the Quality Center - TeamForge tracker connector](#)
Use the Eclipse GUI to create your connector landscape, map source and target projects and fields, and edit connector properties and log settings.
8. [Configure the Quality Center - Project Tracker connector](#)
Use the Eclipse GUI to create your connector landscape, map source and target projects and fields, and edit connector properties and log settings.
9. [Start connector services](#)
To be able to start synchronizing artifacts, start the services for your connector scenario.
10. [Start synchronizing artifacts](#)
When you have finished configuring the connector and made sure that the database and connector scenario services are running, you can start synchronizing artifacts.

Monitor status

While operating the connector, you can check the status of a service and see artifact details, quarantined and queued artifacts, hospital entries and more.

Right-click your connector landscape and select **Show Status**.

Tip:

- Along with various artifact details, Status shows whether a particular service is running. You can refresh details, and restart a service if required.

Note: When Running is checked, it indicates that that service is up, not necessarily that artifacts are being shipped in that direction.

- In Project Mappings, you can check the number of artifacts in the queue to get a sense of how long it would take to synchronize them.

View hospital artifacts

In the Hospital view, you can see the details of a quarantined artifact and the related exception.

- To get a list of artifacts in the hospital, right-click a connector landscape in the CCF Explorer and select **Show Hospital**.
- To narrow down your list of artifacts, click the **Filter** icon in the Hospital view and choose from a variety of filters such as exception details, source and target system attributes, framework component and status.
- To add or remove Hospital table columns, click the **View Menu** icon and then Columns. In the Preferences window, select the columns you want displayed.
-
- To check an artifact's payload, right-click an artifact and select **Examine quarantined artifact**.
- To see the exception that caused the artifact to fail, the stack trace and other details, right-click an artifact, select **View Details** and click Exception Details.
- For information on the framework component that could not handle the artifact, and other source and target system attributes, right-click an artifact, select **View Details** and click Hospital Details.

Work with a failed artifact

When an artifact is quarantined, you could examine its payload, correct the problem, and replay it. Or you could fix the problem and re-run the connector. When the artifact ships successfully, you could mark older versions as fixed. If the problem persists, you could reopen the failed artifact.

- Let's say an artifact failed to ship, you examined its payload and corrected the reason for failure. To now replay it, right-click the artifact in the Hospital view and select **Replay**.

Tip: You may want to filter the artifacts based on specific criteria and replay a bunch of them.

- Let's say you incorrectly mapped a source field to a target field, so the artifact failed to ship and got quarantined. You subsequently updated the mapping, and re-ran the connector so that synchronization succeeded and the artifact got shipped. However, the hospital entry for the failed artifact still exists. To denote that this is now fixed, right-click the artifact in the Hospital view and select **Mark as fixed**.

The Fixed status for the artifact is now set to "true". You will no longer receive warnings about the older version of this artifact.

- Consider a situation where you think you've corrected a problem, and prematurely marked a hospital entry as fixed. However, the problem still exists. In this case, right-click the hospital artifact and select **Reopen**.

The artifact's Fixed column is set to "false". You will continue to receive warnings for this artifact.

You can only reopen an artifact that you had marked as fixed.

Delete a hospital artifact

To keep the hospital database table from getting too big, periodically delete older entries, and artifacts that get reprocessed and successfully replayed.

In the Hospital view, right-click an artifact and select **Delete**.

Tip: You may want to filter a set of artifacts based on some criteria such as error message or status and delete them all at once.

Work with CCF from a remote system

You can use the JMX console to remotely observe or modify some connector features.

In the CCF Explorer, right-click a connector landscape, select Open JMX console in a browser and then a synchronization direction.

You can:

- Edit the source and target repository mappings
- Restart the connector
- Change the log level of CCF components
- Observe the configuration and metrics of CCF components
- Monitor CCF's memory consumption

For more information on the features you can remotely monitor or control, and details about the CCF JMX interface, see [this section](#) of the FAQs.

Best Practices, Frequently Asked Questions and more

The CollabNet Connector Framework project contains further information on best practices for using the connector, frequently asked questions, and a quick reference guide.

- [Best Practices](#)
- [Frequently Asked Questions](#)
- [Quick Reference](#)

- **Why synchronize?**
The CollabNet Connector Framework makes it possible to synchronize different development tools used by project members. With the CollabNet connector to HP Quality Center, you can synchronize requirements and defects between a CollabNet tracker and Quality Center. Team members can work with either tool, and seamlessly see the latest changes made in the other one.

- **How are artifacts and defects synchronized?**
Quality Center defects are synchronized with CollabNet tracker artifacts when you add or modify them in either system.

- **How are artifacts and requirements synchronized?**
Quality Center requirements are synchronized with CollabNet TeamForge tracker artifacts or planning folder artifacts in TeamForge 6.0, when you add or modify them in either system.

- **What are the connector requirements for HP Quality Center?**
The CollabNet Connector Framework works with HP Quality Center versions 9.0, 9.2 and 10.0 and 11.

- **What are the CCF perspective and views?**
The Eclipse plugin for the connector user-interface provides a CCF perspective and these views: CCF Explorer, Hospital, Identity Mapping and Identity Mapping Consistency Check.
- **What is the central database?**
The CollabNet Connector Framework maintains the synchronization status and the identity mappings of all artifacts synchronized by the connector in a central HSQL database. Additionally, the CCF database contains information about artifacts that failed to ship to the target system.
- **Can I install the CCF GUI without Internet access?**
Yes. If your company policy does not allow you to connect to the CCF Eclipse update site directly, you can create a zipped local update site.
- **What are artifact conflicts and how are they handled?**
When you run the connector, a conflict typically occurs when a target artifact is changed by a user after the last update by the connector.
- **What is the artifact hospital?**
The hospital is a database that stores artifacts that failed to ship to the target system during synchronization.
- **How do I back up the connector?**
To back up the connector, copy your production scenario file folder found at <CCF_HOME>\production. If you followed the recommended process, this was the folder that was copied and renamed from the "samples" folder provided with the installation.
- **How do I pause synchronization?**
Pausing the synchronization is performed from the same menu that is used after configuration to resume the synchronization.
- **Where can I find the logs?**
Information about errors is available in .log files located in the respective /logs folder for each scenario. In addition, when you run a connector scenario as a service, the Java Service Wrapper logs all console output from the JVM process to the wrapper.log file directly under that scenario.
- **How do I handle failed artifacts?**
Artifacts that fail are stored in the HOSPITAL database table. Depending on the exception that caused the failure, you could re-run or continue running the connector, or replay the artifact.
- **How is setting up the connector for Project Tracker in CollabNet Enterprise Edition different from setting it up for Project Tracker in TeamForge?**
If you've been using the connector with Quality Center and CollabNet Enterprise Edition Project Tracker, the process for setting it up for Project Tracker in TeamForge, including mapping projects and fields, is the same.