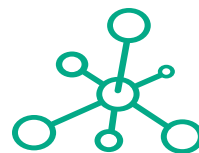# Project and Portfolio Management Center

Software Version: 9.40

Solution Integrations Guide

Document Release Date: September 2016
Software Release Date: September 2016

Hewlett Packard Enterprise

Legal Notices

Warranty

The only warranties for Hewlett Packard Enterprise products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Hewlett Packard Enterprise shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from Hewlett Packard Enterprise required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notice

© 2016 Hewlett Packard Enterprise Development LP

Trademark Notices

Adobe® is a trademark of Adobe Systems Incorporated.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

UNIX® is a registered trademark of The Open Group.

Documentation Updates

To check for recent updates or to verify that you are using the most recent edition of a document, go to: https://softwaresupport.hpe.com/.

This site requires that you register for an HP Passport and to sign in. To register for an HP Passport ID, click **Register** on the HPE Software Support site or click **Create an Account** on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HPE sales representative for details.

The following table indicates changes made to this document since the last released edition.

Support

Visit the HPE Software Support site at: https://softwaresupport.hpe.com/.

Most of the support areas require that you register as an HP Passport user and to sign in. Many also require a support contract. To register for an HP Passport ID, click **Register** on the HPE Support site or click **Create an Account** on the HP Passport login page.

To find more information about access levels, go to: https://softwaresupport.hpe.com/web/softwaresupport/access-levels.

HPE Software Solutions Now accesses the HPSW Solution and Integration Portal website. This site enables you to explore HPE Product Solutions to meet your business needs, includes a full list of Integrations between HPE Products, as well as a listing of ITIL Processes. The URL for this website is https://softwaresupport.hpe.com/km/KM01702731.

Contents

Part 1: Getting Started with PPM Solution Integrations

This part includes the following sections:

- Introduction to PPM Solution Integrations

- Installing and Setting Up ALM Content Bundle

- Using ALM Entities

- "Introduction to Agile Open SDK" on page 95

Chapter 1: Introduction to PPM Solution Integrations

PPM integrations support Change Management and Release Management within your organization. Some of these integrations use HPE Application Lifecycle Management (ALM) content bundle, which adds to PPM a set of entities that support standard Information Technology Infrastructure Library (ITIL) processes. These entities can be configured to meet your business needs.

Applications that PPM can be integrated with using ALM content bundle

- HPEService Manager, for management of Service Manager changes and associated PPM requests

- HPE Universal Configuration Management Database (Universal CMDB), for impact analysis of PPM requests

- HPEQuality Center / HPE Application Lifecycle Management

- HPERelease Control

Applications that PPM can be integrated with not using ALM content bundle

- HPEService Manager, for integration of PPM tasks with Service Manager RFCs

- HPEUniversal CMDB, for:

  - Service portfolio functionality

  - Automating the creation and update of requests in APM module of PPM by pushing CIs from UCMDB

- HPE Application Lifecycle Management, for viewing and optimizing management of project quality

- HPE Agile Manager for project managers to view and manage hybrid projects

These integrations help your IT organization standardize and enforce processes that manage application changes throughout the entire software development lifecycle, including development, testing, and deployment.

The *System Requirements and Compatibility Matrix* describes which versions of these products are supported for integration with PPM.

Introduction to ITIL

The Information Technology Infrastructure Library (ITIL) offers the world's most widely accepted approach to IT Service Management (ITSM), furthering the goal of aligning IT with business goals and priorities. ITIL provides frameworks for both the organization of ITSM as well as a cohesive set of industry best practices.

ITIL is a process framework, and Project and Portfolio Management Center (PPM) is unique in its ability to customize, automate, and digitize processes, simplifying repeatability, enforcement, and measurement.

ITIL defines the Service Support discipline. Building on this advanced-process model, ALM content bundle provides predefined request types (forms), workflows, and special commands to automate processes and information gathering, portlets and reports to track key performance indicators (KPIs), and ALM-specific security groups.

ALM supports the following ITIL processes:

- **Change Management.** "Overview of ITIL Change Management" below provides an overview of the ITIL Change Management process and how ALM supports the process. "Using ALM Entities" on page 33 describes the entities provided by ALM for ITIL Change Management.

- **Release Management.** "Overview of ITIL Release Management" on page 17 provides an overview of the ITIL Release Management process and how ALM supports the process. "Using ALM Entities" on page 33 describes the entities provided by ALM for ITIL Release Management.

ALM can be used as a starting point, and then extended to support the process requirements that meet the specific needs of your organization. Using these tools, ALM helps enforce repeatable ITIL processes to reduce their operating cost and risk.

"Installing and Setting Up ALM Content Bundle" on page 26 provides instructions for installing ALM and configuring PPM to ensure that the integrations work properly.

"Using ALM Entities" on page 33 provides information about the ALM entities, except for a few that are used only for integration of PPM with Quality Center.

Overview of ITIL Change Management

ITIL defines a *change* as the addition, modification, or removal of an approved, supported, or baselined hardware component, network, software, application, environment, system, desktop build, or

associated documentation. The primary goal of the ITIL Change Management process is to ensure that standardized methods and procedures are used for efficient and prompt handling of all changes, in order to minimize the impact of change-related incidents upon service quality. Although changes often arise as a result of unexpected problems, most changes result from planned requests for change (RFCs) from the business or IT organization.

ALM content bundle allows users to submit RFCs along a predefined Request for Change process toward resolution. ALM portlets can be added to a user's PPM Dashboard to monitor key performance indicators (KPIs) related to the submitted RFCs. Additionally, reports can be run to obtain summaries and scheduling details.

"Using ALM Entities" on page 33 discusses the PPM entities that ALM provides for use in the ITIL Change Management process, including the following:

- Change Management request type, named ALM - Request for Change (RFC)

- Associated Change Management workflow, named ALM - Request For Change

- Associated Change Management portlets

- Associated Change Management reports

These ALM entities simplify each of the supported integrations with PPM, as introduced in "Available PPM Integrations " on page 18.

Change Management Roles

The following roles play an active part in the ITIL Change Management process:

- System Owner

- Change Manager

- CAB group (Change Advisory Board) or CAB/EC group (Change Advisory Board – Emergency Committee)

  - Change Manager

  - Customer(s)

  - Applications Development Manager

  - QA Manager

  - Operations Manager (or maintainers where appropriate)

- Change Builder

- Independent Tester

- Applications Development Manager

In addition to being valuable divisions of responsibility, these roles are used to designate user security for the default ALM - Request For Change workflow (see "ALM - Request For Change Workflow" on page 43).

Overview of ITIL Release Management

ITIL defines a *release* as a collection of new and/or changed components that are tested and introduced into the live (production) environment together. The ITIL Release Management process helps to design and implement efficient procedures for the distribution and installation of changes. This process includes coordinating build and testing activities to help ensure that only the authorized and tested versions of changes are implemented in production. Ultimately, releasing a change requires the following actions:

- Developing the change

- Designing the change

- Testing the change for functionality, quality, and performance

- Pushing the change into production

ALM content bundle makes it easier for users to collect information for a release using a release form, which is submitted along a predefined release management process (see "ALM - Release Request Workflow" on page 81). After the initial release process steps have been completed, the release is created and the RFCs that are being processed along the ALM - Request For Change workflow (see "ALM - Request For Change Workflow" on page 43) can be added to the release in preparation for its deployment to the test and live environments. The release management process then moves through testing and deployment steps toward completion. ALM portlets can be added to a user's PPM Dashboard to monitor the status of releases. Additionally, reports can be run to obtain release summaries and scheduling details.

"Using ALM Entities" on page 33 discusses the PPM entities that ALM provides for use in the ITIL Release Management process, including the following:

- Release Management request type, named ALM - Release Management

- Associated Release Management workflow, named ALM - Release Request

- Associated Release Management portlets

- Associated Release Management reports

Release Management Roles

The following roles play an active part in the ITIL Release Management process:

- Release Manager

- Test Manager

- Applications Development Manager

- Operations Manager

- Change Manager

- CAB group (Change Advisory Board)

In addition to being valuable divisions of responsibility, these roles are used to designate user security for the default ALM - Release Request workflow (see "ALM - Release Request Workflow" on page 81).

Available PPM Integrations

As described in the following sections, PPM version 9.40 can be integrated with various HPE products to enhance the functionality of PPM and those products.

Product documentation for PPM and the products with which it integrates is available at the following Web site:

https://softwaresupport.hpe.com

Integration of PPM Tasks with HPE Agile Manager

> **Note:** This integration does not use the ALM entities and does not require installing the ALM content bundle.

You can integrate PPM tasks with the HPE Agile Manager to offer PPM project managers, portfolio managers, and other project stakeholders:

- Visibility into real-time status and progress of agile development projects from within PPM, without having to log on to HPE Agile Manager for details

- A consolidated view of their tasks and agile development initiatives

The integration allows project managers to map a task under a project to a specific release managed in HPE Agile Manager. After the mapping relationship is established, the project managers are able to view real-time agile development related charts retrieved from HPE Agile Manager. In addition, project managers can also view the overall release hierarchy information of a specific work package from within PPM.

For more information about the integration of PPM with HPE Agile Manager, see "Integrating PPM Tasks with HPE Agile Manager" on page 98.

Integration of PPM Time Sheets with HPE Agile Manager

> **Note:** This integration does not use the ALM entities and does not require installing the ALM content bundle.

The integration between PPM time sheets and HPE Agile Manager enables end users to import agile effort from HPE Agile Manager into their PPM time sheets, freeing them from reporting their time repeatedly and thus ensuring data consistency between different management tools.

For more information about the integration of PPM with HPE Agile Manager, see "Integrating PPM Center Time Sheets with Agile Manager" on page 116

Integration of PPM with Quality Center/HPE ALM, Using ALM

Integrating PPM with Quality Center/HPE ALM using ALM content bundle allows you to incorporate the quality assurance (QA) process into the change request development process, providing the following benefits:

- Integration and enforcement of QA testing into the change process. The process can be initiated when the development process begins, and QA planning activities can proceed in parallel with development. The QA teams can then spend more time testing changes after development has been completed.

- Management and tracking of test requirements, plans, and results in Quality Center version 10.00, ALM version 11.00 or later, with visibility in PPM,.

- Data sharing between PPM and Quality Center/ALM.

- Automatic activation of Quality Center/ALM processes by PPM. Creating a request in PPM can create a requirement or defect in Quality Center/ALM.

- Automatic creation of a request in PPM when a defect is created in ALM version 11.00 or later.

- Automatic ongoing synchronization of defects and requirements in Quality Center/ALM with requests in PPM, as well as *hierarchical* synchronization of requirements in Quality Center/ALM with requests in PPM.

This guide assumes that Quality Center version 10 or ALM version 11.00 (or later) has been installed and is available for integration.

For detailed information about configuring and using integration of PPM with Quality Center, see "Integrating PPM with HPE Quality Center, Using ALM " on page 126. Configuration procedures are different for integration with Quality Center version 10.00 and ALM version 11.x or later, and this section describes integration with each of those versions.

Integration of PPM Projects with HPE ALM Releases - View Project Quality

> **Note:** This integration does not use the ALM entities and does not require installing the ALM content bundle.

You can integrate PPM projects with the Releases module of HPE Application Lifecycle Management to offer PPM project managers visibility into quality KPIs and ALM scorecards from PPM Center projects, and to allow them to optimize project quality by a) introducing and enforcing consistent workflows for all major application delivery processes and b) initiating and managing both application and testing projects and ALM Releases.

For more information about the integration of PPM projects with HPE ALM releases, see "Integrating PPM Projects with HPE ALM Releases — View Project Quality" on page 229.

Integration of PPM Tasks with ALM Releases - View Project Quality

> **Note:** This integration does not use the ALM entities and does not require installing the ALM content bundle.

This integration enables project managers to manage multiple ALM releases with one PPM project by integrating PPM Center tasks with ALM releases.

The integration of PPM tasks with the Releases module of the HPE Application Lifecycle Management (ALM) allows project management officers, project managers,development managers, and QA managers to have visibility into quality KPIs and ALM scorecards of multiple ALM releases from PPM Center projects by integrating PPM tasks with ALM releases, and allows them to optimize the management of project quality by:

- Introducing and enforcing consistent workflows for all major application delivery processes.
- Initiating and managing both application and testing projects and ALM releases.

The integration allows project managers to link a task under a project to a specific release managed in ALM. Each PPM Center task is associated with a single release in ALM throughout the task life cycle. This is a one-way one-to-one mapping relationship. This association begins when a project manager maps one task to a specific ALM release.

After the mapping relationship is established, the project managers are able to view quality KPIs and ALM scorecard report for a release retrieved from ALM. In addition, project managers can also view the overall release hierarchy information of a specific work package from within PPM Center.

For more information about the integration of PPM tasks with ALM Releases, see "Integrating PPM Tasks with ALM Releases - View Project Quality" on page 249.

Integration of PPM with Release Control, Using ALM

Integrating PPM with Release Control using ALM content bundle allows you to assist IT managers and the Change Advisory Board in providing the following benefits:

- Assessing the business impact of changes that have been developed and tested, and deciding whether to approve the changes for deployment

- Providing information about the components in the organization's IT environment that will be impacted by the developed changes

- Proactively sending notifications of the business risk involved in each change

- Identifying potential conflicts among concurrently scheduled changes

- Improving visibility over the change deployment process

During the impact analysis phase of the Request for Change lifecycle, if PPM is integrated with both Release Control and Universal CMDB, the **Launch HPE Release Control** button appears on the request. When launched, Release Control provides additional impact analysis that helps users assess and approve changes.

For each change request, the ALM - Releases portlet provides a link to log in to Release Control, where various tabs contain information about the change requests.

This guide assumes that Release Control has been installed and is available for integration.

For detailed information about configuring and using integration of PPM with Release Control, including details about configuring parameters in the `server.conf` file, see "Integrating PPM with HPE Release Control, Using ALM" on page 269.

Integration of PPM Requests with Service Manager Changes, Using ALM

Integrating PPM with the Service Manager using ALM content bundle provides the following benefits:

- Changes that originate in Service Manager can be automatically imported into PPM as requests that PPM manages.

- Changes in Service Manager can be automatically updated, based on revisions to requests in PPM.

- PPM acts as a single, comprehensive repository of application change requests collected throughout IT, including change records (tickets) from Service Manager.

The ALM content bundle provides configurable adapter files that serve as the software interface between PPM and Service Manager. An adapter file includes filters and field mappings to convert changes from one data model to the other.

This guide assumes that one of Service Managers has been installed and is available for integration.

For detailed information about configuring and using integration of PPM with Service Manager, including details about configuring parameters in the `server.conf` file, see "Integrating PPM Requests with HPE Service Manager Changes, Using ALM" on page 274.

> **Note:** Any request type you use for Service Manager must include the fields in the **Service Desk System Info** section of the ALM - Request for Change (RFC) request type, as shown in "ALM - Request for Change (RFC) Request Type" on page 33, and those fields must be completed. For detailed request type field specifications, see the ALM - Request for Change (RFC) request type in the PPM Workbench.

Using Integration of PPM with Quality Center To Enhance Integration of PPM with Service Manager

Functionality of integration of PPM with Service Manager is enhanced if PPM and Quality Center are also integrated—in both Service Manager and PPM, you can see the Quality Center status for an RFC.

> **Note:** For general information about the benefits of integrating PPM and Quality Center whether or not PPM and Service Manager are integrated, see "Integration of PPM with Quality Center/HPE ALM, Using ALM" on page 19.

See "ALM - Request for Change (RFC) Request Type" on page 33 for descriptions of the ALM - Request for Change (RFC) request type fields that are related to integration of PPM with Quality Center.

Integration of PPM Project Tasks with Service Manager RFCs

> **Note:** This integration does not use the ALM entities and does not require installing the ALM content bundle.

You can integrate PPM project tasks with Service Manager requests for change (RFCs) to allow PPM project managers to specify which tasks in a project, if any, automatically create corresponding RFCs in Service Manager. As the RFCs are completed in Service Manager, the status of the associated PPM tasks are automatically set to Complete (or Cancelled).

> **Caution:** This integration is not available for all projects where MSP Integration is set to **Microsoft controls all shared work plan information**.

For more information about the integration of PPM tasks with Service Manager RFCs, see "Integrating PPM Tasks with HPE Service Manager RFCs" on page 332.

Integration of PPM with Universal CMDB for Impact Analysis of Requests, Using ALM

Integrating PPM with Universal CMDB using ALM content bundle provides the following benefits:

- The Change Advisory Board can use the integration to run an impact analysis in Universal CMDB and forecast the effects that each change request will have on the organization's IT environment.

- The resulting report provides the Change Advisory Board with an indication of how the organization's system will cope with each change if the change is later developed and deployed, and thus assists the Change Advisory Board in deciding whether to approve certain changes for development.

- Your system infrastructure, such as servers or applications, might get modified while the software change is being developed and evaluated for quality. As a result, the original impact analysis for the change might not be valid. After the change has been evaluated and approved for deployment, you can perform another impact analysis.

This guide assumes that Universal CMDB has been installed and is available for integration.

For detailed information about configuring and using integration of PPM with Universal CMDB for impact analysis of requests, including details about configuring parameters in the `server.conf` file, see "Integrating PPM with HPE Universal CMDB, Using ALM" on page 350.

Integration of PPM with Universal CMDB for Service Portfolio

> **Note:** This integration does not use the ALM entities and does not require installing the ALM content bundle.

You can integrate PPM with Universal CMDB for service portfolio functionality—the tracking of labor costs categorized by service. In a PPM request, users can select (and may be required to select) a service from a list of services related to IT initiatives. The services can be as specified in ITIL definitions. With this integration, the service list is retrieved from Universal CMDB when needed.

For more information about retrieving service lists in this way, see "Integrating PPM with HPEUniversal CMDB for Service Portfolio" on page 358.

> **Note:** Service lists can also be managed in PPM for use in requests and project tasks. For more information, see the *Demand Management User's Guide* regarding requests and the *Project Management User's Guide* regarding tasks.

Integration of APM with Universal CMDB

> **Note:** This integration does not use the ALM entities and does not require installing the ALM content bundle.

The integration between APM and Universal CMDB (UCMDB) enables you to share information from UCDMB with APM.

You can use the integration to automate the creation and update of requests in APM, freeing you from repetitive and manual input of information in APM. This also ensures that APM is kept up to date with real, accurate, discovered data in your environment.

For more information about the integration of APM with Universal CMDB, see "Integrating HPE APM with HPE Universal CMDB" on page 368.

Chapter 2: Installing and Setting Up ALM Content Bundle

Installing and setting up ALM bundle includes the following procedures:

- Installing the ALM bundle

- Configuring particular ALM-related entities in PPM

- Restarting the PPM Server

Configuration activities that are unique to the integrations with HPE Service Manager, HPE Quality Center/HPE ALM, HPE Release Control, and HPE Universal CMDB are described in their respective sections.

System Requirements

To use ALM entities, you must install ALM content bundle version 9.40 in the following case:

- PPM is at version 9.40 and ALM content bundle has not been previously installed.

> **Caution:** If you already deployed an ALM content bundle and used the ALM entities to integrate with other HPE Software products before you upgrade your PPM to version 9.40, do NOT install the ALM content bundle version 9.40 after upgrade, otherwise you may risk data loss.

To use the request types and workflows provided by ALM content bundle, verify that you have user licenses for PPM Demand Management. For information about permissions, configuration, and security, see the *Demand Management Configuration Guide* and the *Security Model Guide and Reference*.

Installing ALM Content Bundle

Install ALM content bundle as described in the following sections.

General Preparations for Installations

1. Obtain the ALM content bundle.

2. Log on to the PPM Server.

3. Confirm that the system requirements have been met. See "System Requirements" on the previous page.

4. Save the ALM content bundle installation file (`ppm-940-ALM.jar`) to the `<PPM_Home>` directory. `<PPM_Home>` represents the path where the PPM instance is installed. For example: `xyzserver/E/PPMServer.`

   Note: You do not need to unpack the installation file. The installation process automatically unpacks it.

Perform Backup and Restart the PPM Server in Restricted Mode

The steps in this section are recommended but not required.

Note: For more information about the steps in this procedure, see the *Installation and Administration Guide*.

Before installation, do the following:

1. Back up the database and file system for the PPM Server.

2. Stop the PPM Server and restart it in restricted mode, as follows:

   a. Stop the PPM Server.

   b. Run the following script:

      ```
      sh ./setServerMode.sh RESTRICTED
      ```

   c. Start the PPM Server.

Run the Installation Script

1. Navigate to the `<PPM_Home>/bin` directory.

2. Run the following script:

   ```
   sh ./kDeploy.sh -i ALM
   ```

3. As `kDeploy.sh` runs, respond to its prompts.

   When the installation completes successfully, the following message is displayed:

   ```
   Deployment ALM has been successfully installed.
   ```

Configuring ALM-Related Entities in PPM

After installing the ALM content bundle, perform the procedures described in the following sections.

Creating Contact User Data

To use the ALM - Request for Change (RFC) request type, you must create a global user data field of type **Contact User Data,** whether or not you will be establishing any of the integrations of PPM with other applications. When you select a contact in the RFC, the value in the contact's **USER_DATA1** field populates the **Contact Location** field in the RFC. Table 2-1 describes the important parameters for this field. Figure 2-1 shows the User Data Context window.

In the PPM Workbench, select **Configuration > User Data** and select **Contact User Data** to access the User Data Context window. For more information about creating user data, see the *Demand Management Configuration Guide* .

Table 2-1 Contact user data field parameters

Field Name	Value
Prompt	Location:
Token	LOCATION
User Data Column	USER_DATA1
Displayed	Y
Component Type	Text Field
Validation	(any text field of reasonable length)

Figure 2-1 Contact user data field

Configuring the CRT - Priority - Enabled Validation

To use the ALM - Request for Change (RFC) request type, you must add the values listed in Table 2-2 to the **CRT - Priority - Enabled** validation, whether or not you will be establishing any of the integrations of PPM with other applications. These priority values are used as default values and in rules. The rules determine priority from values specified in the **Impact** and **Severity** fields in the RFC. Figure 2-2 shows the Validation window.

In the PPM Workbench, select **Configuration > Validations** and select **CRT - Priority - Enabled** to access the Validation window. For more information about modifying validations, see the *Commands, Tokens, and Validations Guide and Reference*.

Table 2-2 Values to add to CRT - Priority - Enabled validation

Code	Meaning
MEDIUM	Medium
IMMEDIATE	Immediate
PLANNING	Planning

Figure 2-2 CRT - Priority - Enabled validation

Assigning Users to ALM Security Groups

ALM content bundle provides the following security groups:

- ALM - Application Developer

- ALM - Applications Development Manager

- ALM - CAB group (Change Advisory Board)

- ALM - Change Builder

- ALM - Change Manager

- ALM - Customer

- ALM - IT Executive Board

- ALM - Independent Tester

- ALM - Operations Manager

- ALM - QA Manager

- ALM - Release Manager

- ALM - SOX - System Owner

Users must belong to an appropriate security group to be able to see particular portlets. Add users to these security groups, whether or not you will be establishing any of the integrations of PPM with other applications. In the PPM Workbench, select **Sys Admin > Users,** open a user, and click the **Security Groups** tab. For more information about security groups, see the *Security Model Guide and Reference*.

Assigning Security Groups to ALM Workflows

ALM content bundle provides several workflows. The major ones are described in detail in this document. Assign security groups to each workflow step in the manner that best meets your business needs, whether or not you will be establishing any of the integrations of PPM with other applications. In the PPM Workbench, select **Configuration > Workflows,** open a workflow, and edit it. For more information about configuring security groups in workflows, see the *Demand Management Configuration Guide*.

Restarting the PPM Server in Normal Mode

> **Note:** For more information about the steps in this procedure, see the *Installation and Administration Guide*.

After you have completed all installation and configuration procedures, if you previously restarted the PPM Server in restricted mode, stop and restart the PPM Server in normal mode as follows:

1. Stop the PPM Server.

2. Run the following script:

   ```
   sh ./setServerMode.sh NORMAL
   ```

3. Start the PPM Server.

For More Information

ALM content bundle provides request types, workflows, portlets, and reports that can be configured to fit your business needs. For detailed information, see "Using ALM Entities" on page 33.

You can use the ALM entities whether or not you establish any of the supported integrations of PPM with other applications. You can configure those integrations at any time after you have installed and configured ALM as described in this section. For more information, see "Available PPM Integrations " on page 18 and the integration-related sections to which that section refers.

Chapter 3: Using ALM Entities

This section describes the request types, workflows, portlets, reports, and special commands (the "entities") provided in ALM content bundle to facilitate implementation of ITIL processes. Some of these entities are used by the integrations with other HPE products.

The ALM entities provided for change management are described first, then the entities for release management are described, and then the special commands.

Several ALM entities that are used only for integrations of PPM with HPEQuality Center or its new version Application Lifecycle Management are described in "Integrating PPM with HPE Quality Center, Using ALM " on page 126.

For More Information

The request types, workflows, portlets, reports, and special commands provided with ALM can be configured to fit your business needs. The following table lists the types of entities and the associated PPM guides to which you should refer for configuration information.

Table 3-1. PPM Center entities and associated configuration guides

Entity	Configuration Guide
Request type	*Demand Management Configuration Guide*
Workflow	*Demand Management Configuration Guide*
Portlet	*Creating Portlets and Modules*
Report	*Reports Guide and Reference*
Special command	*Commands, Tokens, and Validations Guide and Reference*

ALM - Request for Change (RFC) Request Type

In its implementation of the ITIL Change Management process, ALM uses the ALM - Request for Change (RFC) request type and sends an RFC request along the ALM - Request For Change workflow (see "ALM - Request For Change Workflow" on page 43).

ALM - Request for Change (RFC) Request Fields

Figure 3-1 and Figure 3-2 show the top and bottom of the Create New ALM - Request for Change (RFC) page that appears when you create a request and select the ALM - Request for Change (RFC) request type. Table 3-2 describes the fields in the ALM - Request for Change (RFC) request, including some fields that do not appear until the request is created or until other conditions are met.

Figure 3-1. Top of ALM - Request for Change (RFC) request

Create New ALM - Request for Change (RFC)

Submit   Cancel

Expand All | Collapse All

− RFC Summary

Created By:
Admin User

RFC Status: **Contact Name:**   **Contact Phone:**
Logged

RFC Priority: **Contact Email:**   **Contact Location:**

RFC Summary:

Expected Start Date: **Expected Finish Date:**

Assigned Developer: **Release ID:**

− QC/ALM Info

QC/ALM Instance   **QC/ALM Domain**

QC/ALM Project   **Assigned To**

Requirement No.   **Requirement Status**

Requirement Integration Message   **Synchronize to QC/ALM Requirement**
                                        ● Yes ○ No

Requirement Attachment URL   **QC/ALM Requirement Type:**
(No Link)

QC/ALM Requirements coverage

QC/ALM Open Defects

− RFC Details

*RFC Source: ***Urgency:**   ***Impact:**

*Reason For Change: **Category:**   **RFC Type:**

Service:

+ Change Item:

*Effect of no change:

RFC Description:

− SOX Information

*System: **SOX - In Scope System:**   **SOX Risk:**
                                                                        Low

− Implementation Details

Figure 3-2. Bottom of ALM - Request for Change (RFC) request

Table 3-2. ALM - Request for Change (RFC) request fields

Field Name (*Required)	Description
RFC Summary section	
RFC ID	(Read-only. Added after the RFC is created.) Number of the RFC, linked to the RFC.
Created By	(Read-only) User who created the RFC.
Created On	(Read-only. Added after the RFC is created.) Date the RFC was created.
RFC Status	(Read-only) Status of the RFC.
Contact Name	Name of the person proposing the change.
Contact Phone	(Read-only) Telephone number of the person proposing the change.
RFC Priority	(Read-only) Priority of the change request. Determined by a combination of the **Urgency** and **Impact** fields.
Contact Email	(Read-only) Email address of the RFC contact person.
Contact Location	(Read-only) Location of the person proposing the change.
RFC Summary	Summary description of the RFC request.
Expected Start Date	Expected start date for work on the RFC.
Expected Finish Date	Expected end date for work on the RFC.
Assigned Developer	Developer assigned to work on the RFC.
Release ID	Number of the release if the change was released.
RFC Details section	
*RFC Source	Source of the RFC request (for example, from a problem or incident).
*Urgency	Urgency of the change request (for example, from problem urgency).
*Impact	Business impact of doing or not doing the change (for example, from problem impact).
*Reason For Change	Reason for the change.
Category (Required only after the request is created)	Category of the change, based on the scope of the change.
RFC Type	Type of change being requested.

Table 3-2. ALM - Request for Change (RFC) request fields, continued

Field Name (*Required)	Description
Service	IT service that needs the change.
Change Item	(Looks like a separate section in the interface) Expand to display a table of change items. If the RFC has been created, click the **Edit** icon to switch to editable mode and add a change item. The table consolidates existing change items (CIs), each having an automatically assigned sequence number (**Seq**), a **CI Type** (**Software, Hardware,** or **Network**), **CI ID**, a **CI Name,** and a **CI Description.** **Note:** These CIs are not the same as the configuration items (CIs) that are retrieved when integration with HPE Universal CMDB is established.
*Effect of no change	Effect of not implementing the change.
RFC Description	Description of the change request.
Authorized By	(Added after the RFC is created and assigned to a developer) Person who authorized the change.
Authorization Date	(Added after the RFC is created and assigned to a developer) Date the developer was assigned.
SOX Information section	
*System	System that is impacted by the change.
SOX - In Scope System	(Read-only) SOX requirement: SOX oversight is required for any application that directly or indirectly affects financial reporting. This field is automatically populated with a value of **Yes** or **No** before the RFC is created, based on the option chosen for the **System** field.
SOX Risk	SOX requirement: Risk is determined as part of SOX oversight. **Note:** A change to a non-SOX system could be high-risk based on possible infrastructure/network impact.
System Owner	(Read-only. Added after the RFC is created.) Owner of the system. This field is automatically populated, based on the option chosen for the **System** field.
Implementation Details section	
Actual Start Date	Actual start date for creation of the change.
Actual Finish Date	Actual finish date for creation of the change.

Table 3-2. ALM - Request for Change (RFC) request fields, continued

Field Name (*Required)	Description
Actual Duration	Actual duration for creation of the change.
Actual Effort	Actual effort expended during creation of the change.
Assigned Change Builder	Details of the change builder/implementer.
Actual Cost	Actual cost of the change.
Functional Specifications	Allows you to add and view the functional specification document directly on the RFC.
Design Specifications	Allows you to add and view the design specification document directly on the RFC.
Impact & Resource Assessment section	
Impact Severity	Specify after evaluating the Impact Analysis Report. This field becomes required by the ALM - Impact & Resource Assessment Sub WF subworkflow after CIs are selected in the **Impacted Configuration Items** section.
Impact Analysis Report	Allows you to add and view an Impact Analysis Report generated by Universal CMDB directly on the request.
Impact Assessment Summary	Risk assessment of the impact of the change on related components in the configuration management database (CMDB).
Impact Assessment Report	Allows you to add and view a manually generated impact assessment report based on the Impact Analysis Report directly on the RFC.
Expected Duration	Expected duration for creation of the change.
Expected Effort	Expected effort for creation of the change.
Expected Cost	Expected cost of the change.
Backout Plan	Allows you to add and view the backout plan document directly on the RFC.
CAB Recommendations	CAB recommendations, where appropriate.
Users Impacted	Users expected to be impacted by the change.
Impacted Configuration Items section [a]	
Select Configuration Items button	Button to launch the Configuration Items (CIs) selector applet provided by Universal CMDB.
Impacted Configuration Items list	List of CIs added to the request, both manually and by using the CI selector applet from Universal CMDB.

Table 3-2. ALM - Request for Change (RFC) request fields, continued

Field Name (*Required)	Description
QA Details section	
Assigned Tester	Person assigned to test the change.
Test Plan	Allows you to add and view the test plan directly on the RFC.
Detailed Test Results (SOX)	Allows you to add and view the detailed test results directly on the RFC.
QC/ALM Info section [b]	
QC/ALM Instance	URL of the Quality Center/ALM instance with the project used for the integration.
QC/ALM Domain	Domain of the project in Quality Center or ALM.
QC/ALM Project	Quality Center or ALM project that is integrated with this request type.
Assigned To	The Quality Center/ALM Requirement assigned to user.
Requirement No.	(Read-only) Requirement number in Quality Center or ALM.
Requirement Status	(Read-only) Status of the requirement in Quality Center or ALM.
Requirement Integration Message	(Read-only) Quality Center/ALM status message indicating success or error in the most recent operation.
Synchronize to QC/ALM Requirement	Select Yes or No radio button to indicate synchronizing to Quality Center/ALM requirement or not.
Requirement Attachments URL	(Read-only) URL of the list of attachments to the Quality Center/ALM requirement.
QC/ALM Requirement Type	Quality Center/ALM requirement type
QC/ALM Dashboard Subject	Quality Center/ALM Dashboard subject name
QC/ALM Requirement Coverage	(Read-only) The Quality Center/ALM requirement coverage
QC/ALM Open Defects	(Read-only) Number of open defects in Quality Center/ALM.
Service Desk System Info section [c]	
System Name	(Read-only) Name of the service desk application—Service Manager.
Ticket Id	(Read-only) Ticket ID in Service Manager.
Ticket Creation Date	(Read-only) Ticket creation date in Service Manager.

Table 3-2. ALM - Request for Change (RFC) request fields, continued

Field Name (*Required)	Description
Ticket Info	(Read-only) Ticket info from Service Manager.
Ticket Priority	(Read-only) Ticket priority in Service Manager.
Ticket Last Update	(Read-only) Date the ticket was last updated in Service Manager.
Review Summary section	
Review Date	Review date for the change.
Review Summary	Summary of the review for the change.

  a. The Impacted Configuration Items section is visible only if the Universal CMDB Impact Analysis field group is enabled in the request type. Data is presented for the Impacted Configuration Items list in this section only if PPM is integrated with Universal CMDB.

  b. Fields in the Quality Center Info section remain visible by default but are not used if PPM is not integrated with Quality Center.

  c. Fields in the Service Desk System Info section remain visible by default but are not used if PPM is not integrated with Service Manager. However, when this request type (or any other request type) is used for Service Manager, these fields are required.

Note: The administrator can remove the **Impacted Configuration Items** section or the **QC/ALM Info** section from the request type by removing the Universal CMDB Impact Analysis field group or the QC/ALM Info field group, respectively, from the ALM - Request for Change (RFC) Header request header type.

See the *Demand Management Configuration Guide* for details about request header types and field groups.

How to submit an ALM - Request for Change (RFC) request

1. Log on to PPM.

2. From the menu bar, select **Create > Request.**

   The Create New Request page appears.

3. On the Create New Request page, in the **Request Type** field, select **ALM - Request for Change (RFC)** and click **Create.**

   The Create New ALM - Request for Change (RFC) page appears, displaying the appropriate RFC fields.

Required fields have a red asterisk. All other fields are optional, but are often helpful when others are reviewing an open request. For information concerning a specific field, click the **Help** icon next to the field (if available).

4. Complete the fields in all sections as appropriate.

   The **Notes** section contains fields where notes and information concerning the RFC can be entered and stored. Typically, when you create an RFC, you do not need to add a note to it. However, add a note if you want to convey additional information to the reviewers and processors of the RFC.

   In the **References** section of the RFC, you can add useful references such as a Web-accessible file or a document or file attached from a local machine. Additionally, other entities such as packages, releases, or other related requests may be automatically referenced based on the workflow steps that have been executed. For more information about adding references, see the *Demand Management User's Guide*.

5. On the Create New Request page, click **Submit.**

   The RFC is submitted. The Request Creation Confirmed page appears.

   > **Note:** PPM can be configured to allow you to save the request before you submit it. To have this feature enabled, see your application administrator.

   After submitting the request, on the Request Creation Confirmed page you can click the link for the particular request number in the **Request #** field to view the detail page of the newly generated RFC.

   When the RFC is submitted, it is assigned an initial status, such as New. The RFC is then routed along the ALM - Request For Change workflow (see "ALM - Request For Change Workflow" below).

ALM - Request For Change Workflow

The ALM - Request For Change workflow is the sequence of approvals, decisions, or actions by which the RFC is processed. The RFC starts at the beginning of the workflow. When the RFC reaches the end of the workflow, its lifecycle is complete.

Upon creation, an ALM - Request For Change (RFC) request is automatically set to use the ALM - Request For Change workflow.

Figure 3-3 shows the workflow. Table 3-3 lists the important steps in the workflow and the user roles associated with those steps.

Figure 3-3. ALM - Request For Change workflow
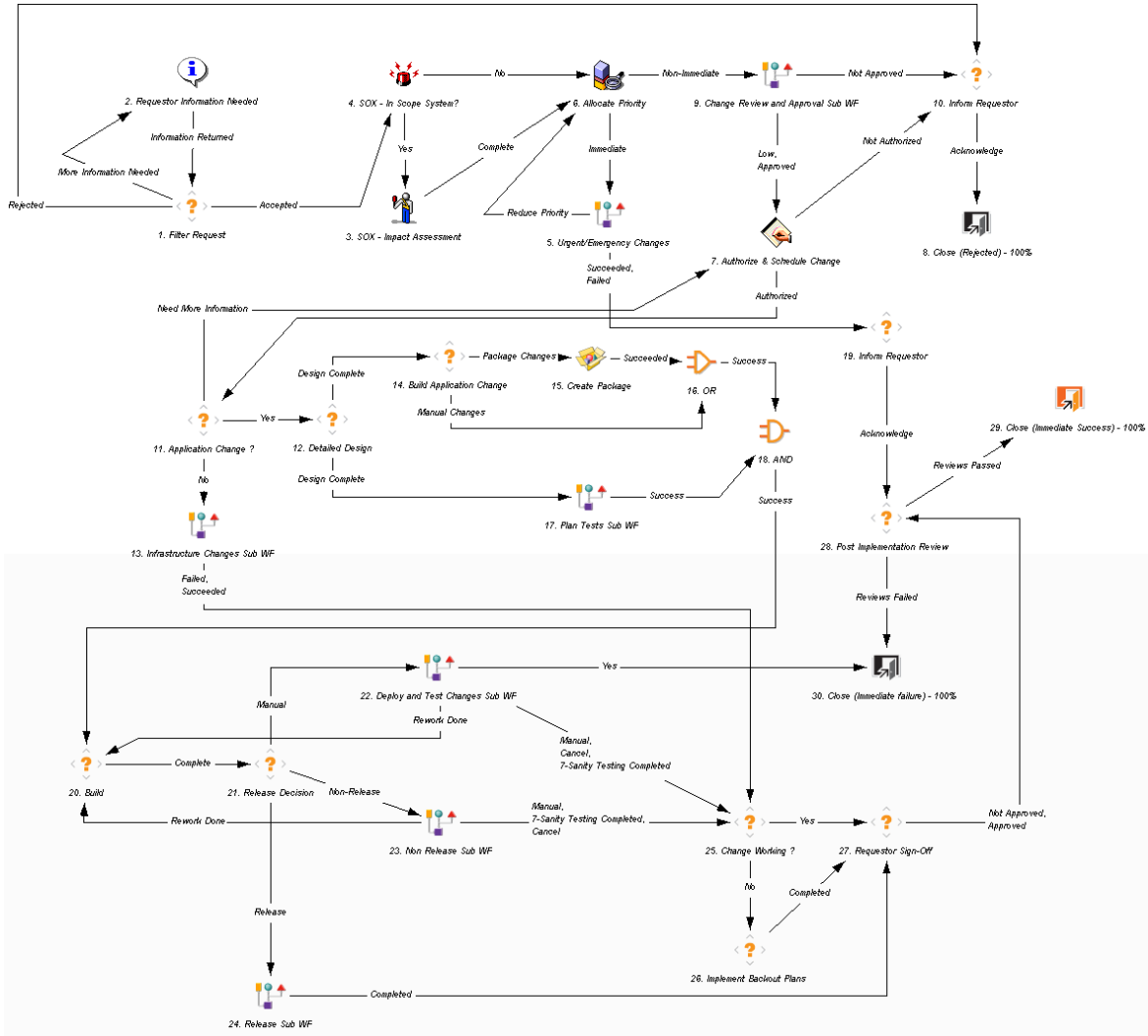
Table 3-3. ALM - Request For Change workflow steps

Step	User Security	Description
1. Filter Request	ALM - Change Manager	Perform initial review and classification of the change request, and determine if this RFC is acceptable based on company policy.
4. SOX - In Scope System?	Fully automated step	SOX requires identification of key systems related to accurate financial reporting, directly or indirectly. This field is automatically determined based on the system selected.
3. SOX - Impact Assessment	ALM - SOX - System Owner	SOX requires additional impact assessment for any change that could affect financial reporting. The impact of not doing the change must be considered as well.

Table 3-3. ALM - Request For Change workflow steps, continued

Step	User Security	Description
6. Allocate Priority	ALM - Change Manager	Validate RFC priority and determine if this is an Urgent Change request.
5. Urgent/Emergency Changes	(None)	Call a subworkflow designed to handle urgent or emergency changes, described in "ALM - RFC - Urgent Change Management Sub WF Subworkflow" on page 47.
9. Change Review and Approval Sub WF	ALM - Change Manager	Call a subworkflow to manage the review and approval process for the RFC, described in "ALM - Change Review and Approval Sub WF Subworkflow" on page 50. (This subworkflow can, in turn, call the ALM - Impact & Resource Assessment Sub WF subworkflow.)
7. Authorize & Schedule Change	ALM - Change Manager	Authorize the change request and schedule change for implementation.
11. Application Change?	ALM - Change Manager	Determine if this is an application change.
13. Infrastructure Changes Sub WF	(None)	Call a subworkflow to handle non-application changes, described in "ALM - Infrastructure Changes Sub WF Subworkflow" on page 53.
12. Detailed Design	ALM - Application Developer	Create functional and design specification documents.
14. Build Application Change	ALM - Application Developer	Build application code for the change.
15. Create Package	ALM - Application Developer	Create a package with the code changes. This step automatically creates a package and adds the package as a reference to the RFC request. "ALM - Request For Change Workflow" on page 43 illustrates a typical RFC package.
17. Plan Tests Sub WF	(None)	Call a subworkflow to manage the test planning process for the RFC, described in "ALM - Plan Tests Sub WF Subworkflow" on page 53.
20. Build	ALM - Change Builder	Build the change, in preparation for implementation. For an application change, add code components to the referenced package in the PPM Workbench (see the Create Package step).

Table 3-3. ALM - Request For Change workflow steps, continued

Step	User Security	Description
21. Release Decision	ALM-Change Builder	Select the option to implement this application change as part of a release or independently.
22. Deploy and Test Changes Sub WF	(None)	Call a subworkflow for deploying and testing changes, described in "ALM - Deploy and Test Changes Sub WF Subworkflow" on page 56.
23. Non Release Sub WF	(None)	Call a subworkflow for change deployment not involving a release, described in "ALM - Non Release Sub WF Subworkflow" on page 58.
24. Release Sub WF	(None)	Call a subworkflow for change deployment involving a release, described in "ALM - Release Sub WF Subworkflow" on page 60.
25. Change Working?	ALM - Change Manager	Review whether the change was successfully implemented with no adverse impact.
26. Implement Backout Plans	ALM - Operations Manager	If the change is not working, implement backout plans to back out the change from the LIVE environment.
27. Requestor Sign-Off	ALM - Change Manager	Get sign-off from the requestor of the change to acknowledge that the change was implemented.
28. Post Implementation Review	ALM - Change Manager	Review the change after implementation to determine whether the change process was followed correctly.
29. Close (Immediate Success) - 100%	(None)	Update status to **Closed.**

Step 15 in the ALM - Request for Change workflow creates a package to deploy and test changes, and the step adds the package as a reference to the RFC request. This package requires specifying a workflow, and the default is the ALM - Change Migration workflow, as shown in the example in Figure 3-4. Step 22 in the ALM - Request for Change workflow calls the ALM - Deploy and Test Changes Sub WF subworkflow to deploy and test the package.

Figure 3-4. Package created for ALM - Request For Change workflow

ALM - RFC - Urgent Change Management Sub WF Subworkflow

ALM provides an "Urgent Change" process. If a change is categorized as **Urgent,** the RFC is routed along the Urgent Change process. The ALM - RFC - Urgent Change Management Sub WF subworkflow is designed to efficiently handle impact assessment, prioritization, and creation of the change. Figure 3-5 shows the subworkflow.

Figure 3-5. ALM - RFC - Urgent Change Management Sub WF subworkflow

1. Call CAB/EC Meeting

Escalate

2. Quick Impact & Resource Assessment

Completed

3. Determine Priority — Reduce Priority ——▶ 4. Return

Immediate/Emergency

5. Urgent/Emergency Build Application Change

Complete

6. Urgent Testing

Complete

7. Deploy Changes to LIVE

Succeeded

8. Evaluate Quality

Succeeded

9. Review Change — Succeeded ——▶ 10. Return
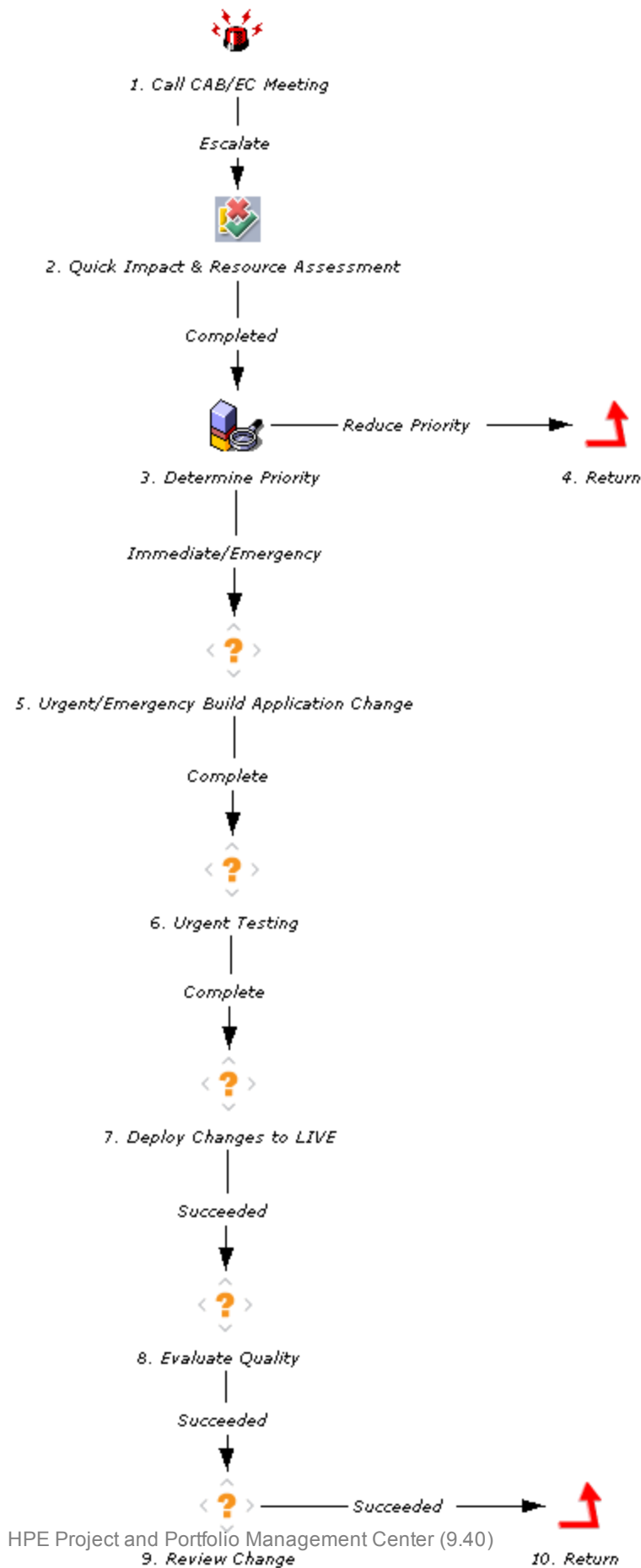
ALM - Change Review and Approval Sub WF Subworkflow

The ALM - Change Review and Approval Sub WF subworkflow provides a modular review and approval process for the RFC. Figure 3-6 shows the subworkflow. Table 3-4 lists the important steps in the subworkflow and the user roles associated with those steps.

Figure 3-6. ALM - Change Review and Approval Sub WF subworkflow
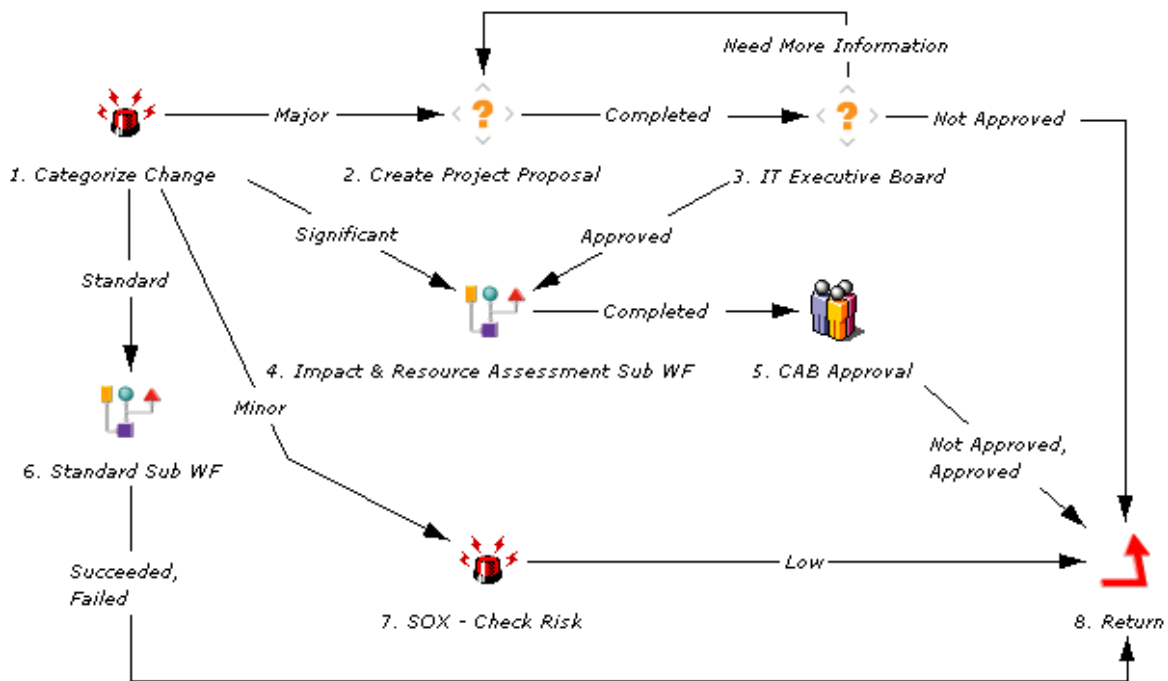
Table 3-4. ALM - Change Review and Approval Sub WF subworkflow steps

Step	User Security	Description
1. Categorize Change	ALM - Change Manager	Categorize the change to determine the next step in the workflow.
2. Create Project Proposal	ALM - CAB group (Change Advisory Board)	If the change is classified as "Major," create a project proposal that includes impact.
3. IT Executive Board	ALM - IT Executive Board	If the change is classified as "Major," an IT Executive Board is responsible for approving the change.
4. Impact & Resource Assessment	(None)	If the change is classified as "Significant," call a subworkflow to determine the impact on dependent infrastructure components and estimate the time and

Table 3-4. ALM - Change Review and Approval Sub WF subworkflow steps, continued

Step	User Security	Description
Sub WF		cost of resources, as described in "ALM - Impact & Resource Assessment Sub WF Subworkflow" below.
5. CAB approval	ALM - CAB group (Change Advisory Board)	Iterative review by CAB members, resulting in an authorization go/no go decision (includes change priority, schedule, impact, and cost).
6. Standard Sub WF	(None)	Call a subworkflow to track standard changes.

ALM - Impact & Resource Assessment Sub WF Subworkflow

The ALM - Impact & Resource Assessment Sub WF subworkflow provides a modular process for assessing change impact and planning resource usage. Figure 3-7 shows the ALM - Impact & Resource Assessment Sub WF subworkflow. Table 3-5 lists the important steps in the subworkflow and the user roles associated with those steps.

Figure 3-7. ALM - Impact & Resource Assessment Sub WF subworkflow

Table 3-5. ALM - Impact & Resource Assessment Sub WF subworkflow steps

Step	User Security	Description
1. Resource Assessment	ALM - Change Manager	User estimates time and cost of resources.
2. Select CIs	ALM - Change Manager	User manually selects an initial set of configuration items (CIs) for the change.
3. Impact Assessment	ALM - Change Manager	User approves having Universal CMDB generate the Impact Analysis report, which adds CIs to the CI list based on predefined rules in Universal CMDB. User then evaluates the Impact Analysis report and creates an impact assessment report with recommendations.

Table 3-5. ALM - Impact & Resource Assessment Sub WF subworkflow steps, continued

Step	User Security	Description
4. Review Impact Analysis	ALM - Change Manager	User reviews the request, the list of selected CIs, and the Impact Analysis and impact assessment reports. The user can approve the change based on the impact reports, or return to the Select CIs step to select a different initial set of CIs.

ALM - Infrastructure Changes Sub WF Subworkflow

If an authorized request for change is not an application change, the ALM - Request for Change workflow calls the ALM - Infrastructure Changes Sub WF subworkflow to manage the approval and deployment of changes to TEST and LIVE environments. Figure 3-8 shows this subworkflow.

Figure 3-8. ALM - Infrastructure Changes Sub WF subworkflow
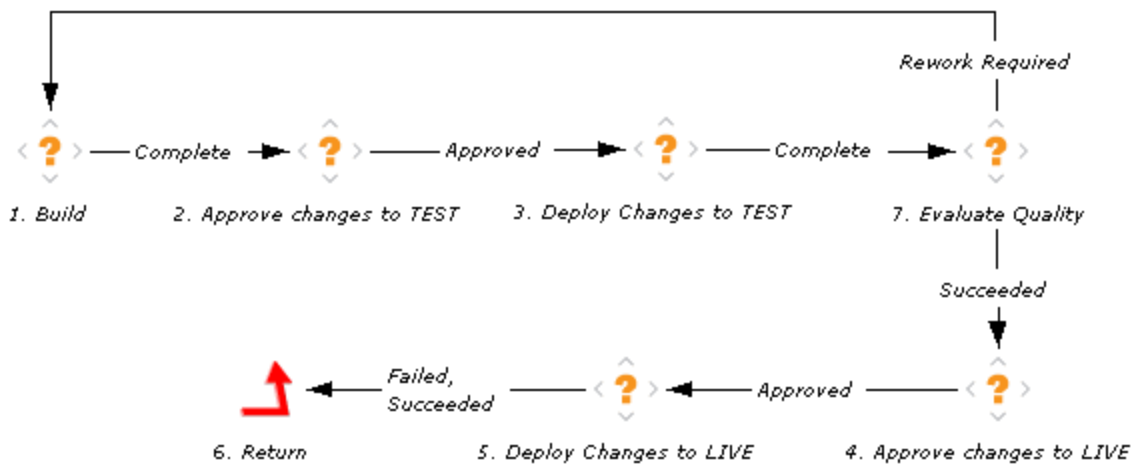
ALM - Plan Tests Sub WF Subworkflow

The ALM - Plan Tests Sub WF subworkflow provides a modular process for planning tests that can communicate automatically with Quality Center. If no integration exists, a manual process is also provided. Figure 3-9 shows the subworkflow. "ALM - Plan Tests Sub WF Subworkflow" above lists the important steps in the subworkflow and the user roles associated with those steps.

Figure 3-9. ALM - Plan Tests Sub WF subworkflow
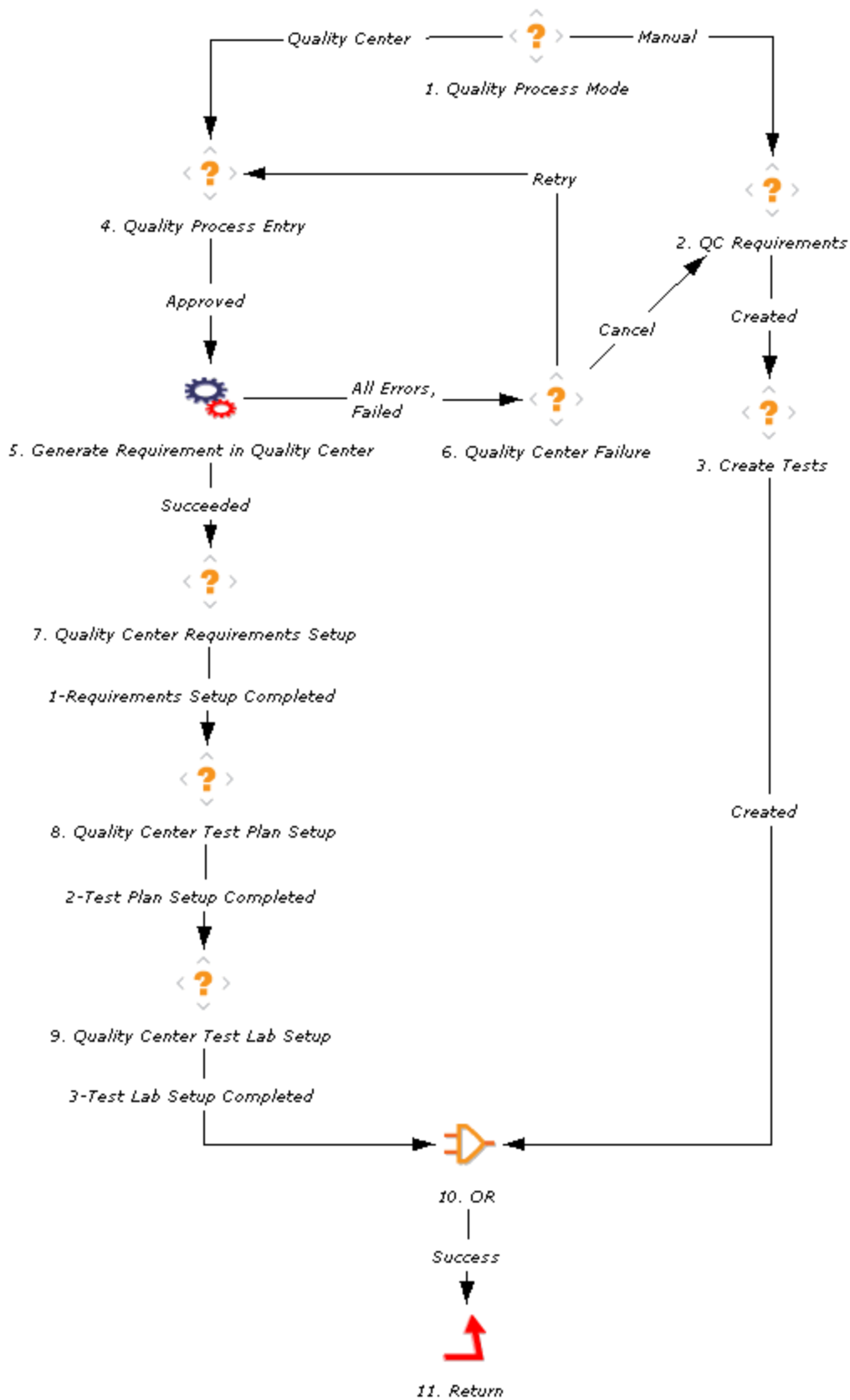
Table 3-6. ALM - Plan Tests Sub WF subworkflow ste

Step Name	User Security	Description
1. Quality Process Mode	ALM - QA Manager	Determine the method of test planning, either automatic through Quality Center, or manual.
4. Quality Process Entry	ALM - QA Manager	The quality process entry needs to be approved for integration of PPM with Quality Center.
5. Generate Requirement in Quality Center	ALM - QA Manager	Automated step that generates a requirement in Quality Center.
7. Quality Center Requirements Setup	ALM - QA Manager	QA to complete requirement setup in Quality Center.
8. Quality Center Test Plan Setup	ALM - QA Manager	QA to complete test plan setup in Quality Center.
9. Quality Center Test Lab Setup	ALM - Independent Tester	QA to complete test lab setup in Quality Center.
2. QC Requirements	ALM - QA Manager	Create test requirements in Quality Center.
3. Create Tests	ALM - QA Manager	Create test plans in Quality Center.

ALM - Deploy and Test Changes Sub WF Subworkflow

The ALM - Deploy and Test Changes Sub WF subworkflow provides a modular process for deploying and testing changes. The process can communicate automatically with Quality Center. If no integration exists, a manual process is also provided. Figure 3-10 shows the subworkflow. Table 3-7 lists the important steps in the subworkflow and the user roles associated with those steps.

Figure 3-10. ALM - Deploy and Test Changes Sub WF subworkflow
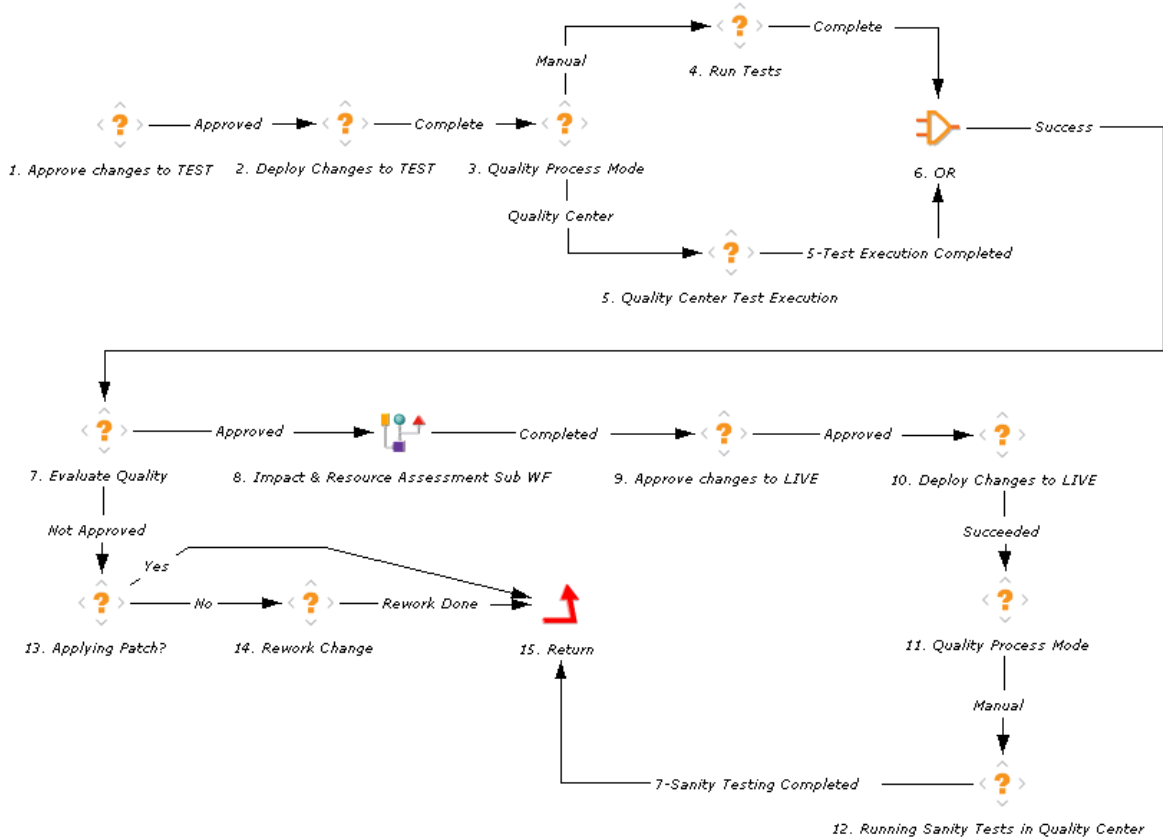
Table 3-7. ALM - Deploy and Test Changes Sub WF subworkflow steps

Step Name	User Security	Description
1. Approve changes to TEST	ALM - Change Manager	Approve deployment of changes to the TEST environment.
2. Deploy Changes to TEST	ALM - Change Manager	Deploy changes to the TEST environment.
3. Quality Process Mode	ALM - QA Manager	Determine the method of testing, either automatic through Quality Center or manual.
5. Quality Center Test Execution	ALM - Independent Tester	Quality Center tests the changes in a TEST environment.
7. Evaluate Quality	ALM - QA Manager	After test execution (manual or using Quality Center), evaluate quality.
8. Impact & Resource	(None)	Call a subworkflow to determine the impact of the changes that will be deployed, as described in "ALM - Impact & Resource

Table 3-7. ALM - Deploy and Test Changes Sub WF subworkflow steps, continued

Step Name	User Security	Description
Assessment Sub WF		Assessment Sub WF Subworkflow" on page 51.
9. Approve changes to LIVE	ALM - Change Manager	Approve deployment of changes to the LIVE environment.
10. Deploy Changes to LIVE	ALM - Change Manager	Deploy changes to the LIVE environment.
11. Quality Process Mode	ALM - QA Manager	Initiate sanity tests in Quality Center.
12. Running Sanity Tests in Quality Center	ALM - QA Manager	Run sanity tests in Quality Center.

ALM - Non Release Sub WF Subworkflow

The ALM - Non Release Sub WF subworkflow provides a modular process for change deployment that can communicate automatically with Quality Center. If no integration exists, a manual process is also provided. Figure 3-11 shows the subworkflow. "ALM - Non Release Sub WF Subworkflow" above lists the important steps in the subworkflow and the user roles associated with those steps.

Figure 3-11. ALM - Non Release Sub WF subworkflow
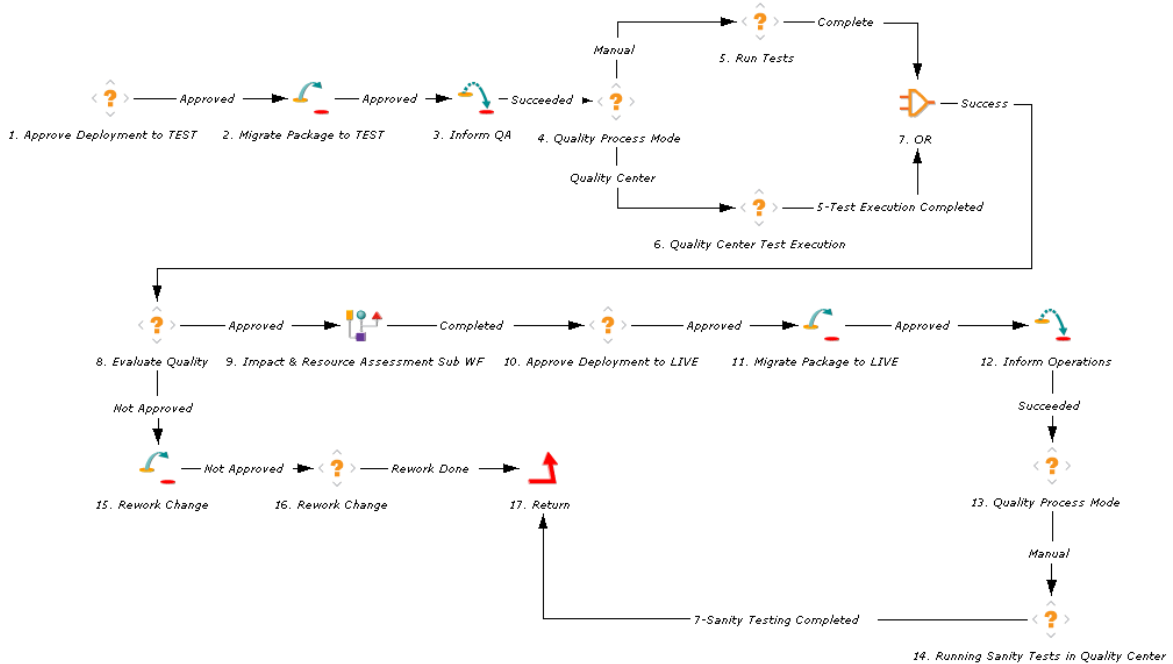
Table 3-8. ALM - Non Release Sub WF subworkflow steps

Step Name	User Security	Description
1. Approve Deployment to TEST	ALM - Change Manager	Coordinate the change implementation to the test environment.
4. Quality Process Mode	ALM - QA Manager	Determine the quality process mode, either automatic through Quality Center or manual.
5. Run Tests	ALM - Independent Tester	If manual mode is chosen in the Quality Process Mode step, the changes need to be manually tested based on test plans.
6. Quality Center Test Execution	ALM - Independent Tester	If Quality Center mode is chosen in the Quality Process Mode step, once QA signals through Quality Center that Test execution is complete, this step moves the workflow ahead.
8. Evaluate Quality	ALM - QA Manager	Approve/reject the quality of the change deployed to the test environment.
9. Impact & Resource Assessment Sub WF	(None)	Call a subworkflow to determine the impact on dependent infrastructure components and estimate the time and cost of resources, as described in "ALM - Impact & Resource Assessment Sub WF Subworkflow" on page 51.
15. Rework	ALM -	If the quality of the change deployed to the test environment is

Table 3-8. ALM - Non Release Sub WF subworkflow steps, continued

Step Name	User Security	Description
Change	Applications Development Manager	rejected, the change must be fixed.
10. Approve Deployment to LIVE	ALM - Change Manager	Coordinate the change implementation to the production environment.
14. Running Sanity Tests in Quality Center	ALM - Independent Tester	Run sanity tests in Quality Center.

ALM - Release Sub WF Subworkflow

The ALM - Release Sub WF subworkflow is called in order to add a change into an existing release. The subworkflow can be called multiple times, once for each change to be added to the release. Once a release is specified as Completed, no new changes can be added to that release. Figure 3-12 shows the subworkflow. (For information about creating a new release, see "ALM - Release Management Request Type" on page 73.)

Figure 3-12. ALM - Release Sub WF subworkflow
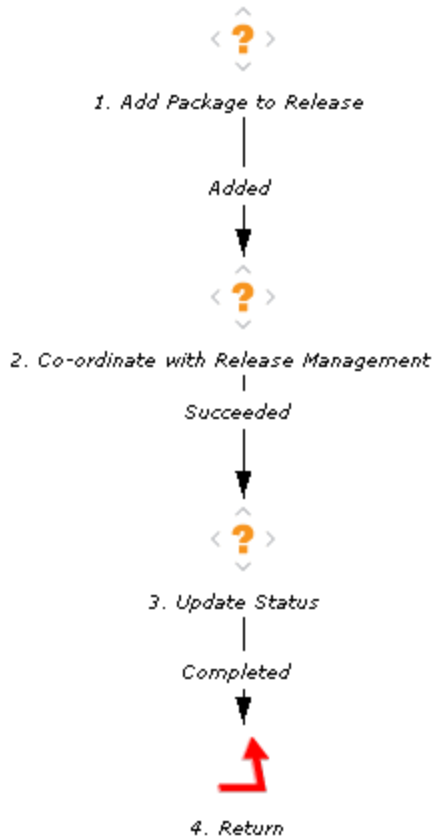
Changes with the Entities in PPM Version 9.20 and Later

This section describes changes with the entities in PPM version 9.20 or later.

> **Caution:** If you already deployed an older version ALM content bundle on an earlier version of PPM, do **NOT** deploy the new version ALM content bundle. Otherwise you may risk losing data of your existing integrations after you upgrade Quality Center from version 10.00 to ALM 11.x.

ALM - Request for Change Workflow

The following two request status are added to the workflow template:

- 3-Test Lab Setup Completed

- 5-Test Execution Completed

Caution: Do not use the ALM - Request for Change workflow template provided in the ALM content bundle directly. The workflow template provided aims to act as a template to minimize your workflow creation effort.

To create your own workflow based on the workflow template,

1. Customize the ALM - Request for Change workflow template by adding the following two request statuses to appropriate workflow steps as needed:

   ○ 3-Test Lab Setup Completed

   ○ 5-Test Execution Completed

   If the two request status do not exist, go to the Request Type Workbench and add two request status to applicable request types. For detailed instructions, see the *Demand Management Configuration Guide*.

2. Add necessary access grants and user groups to each of the modified steps.

Changes with Field Groups

The following entities were renamed since PPM version 9.20 to reflect its support for Quality Center's new version, the standard edition of Application Lifecycle Management (ALM):

- The **QC/ALM Defect Information** field group and its fields (see Table 6-4)

- The **QC/ALM Info** field group and its fields (see Table 6-5)

Table 6-4. The QC/ALM Defect Information field group and its fields

Field Name in 9.10 or earlier	Field Name in 9.20 or later
Quality Center Defect Information field group	QC/ALM Defect Information field group
Quality Center Instance	QC/ALM Instance
Quality Center Domain	QC/ALM Domain
Quality Center Project	QC/ALM Project
—	Synchronize to QC/ALM Defect a (added in 9.20)
Defect Number	QC/ALM Defect Number
Quality Center Defect Status	QC/ALM Defect Status
Quality Center Message	QC/ALM Integration Message

Table 6-4. The QC/ALM Defect Information field group and its fields, continued

Field Name in 9.10 or earlier	Field Name in 9.20 or later
—	QC/ALM Assigned To User [a] (added in 9.12)
Quality Center Attachments	QC/ALM Attachment URL
Detected in Quality Center by	Detected in QC/ALM by

a. For integration configurations upgraded from QC 10, to add the new fields to your existing request types, you can run the script provided with PPM version 9.20 or later to update the existing request types: `<PPM_Home>/bin/kUpgradeIntegrationRequests.sh`.
Note that the script does not update the field names.

Table 6-5. The QC/ALM Info field group and its fields

Field Name in 9.10 or earlier	Field Name in 9.20 or later
Quality Center Info field group	QC/ALM Info field group
Quality Center Instance	QC/ALM Instance
Quality Center Domain	QC/ALM Domain
Quality Center Project	QC/ALM Project
Quality Center Assigned To User	Assigned To
Quality Center Requirement No.	Requirement No.
Quality Center Status	Requirement Status
Quality Center Message	Requirement Integration Message
—	Synchronize to QC/ALM Requirement [a] (added in 9.20)
Quality Center Attachments	Requirement Attachment URL
—	QC/ALM Requirement Type a (added in 9.20)
—	QC/ALM Dashboard Subject (not in use)
—	QC/ALM Requirements Coverage (not in use)
—	QC/ALM Open Defects (not in use)

a. For integration configurations upgraded from QC 10, to add the new fields to your existing request types, you can run the script provided with PPM version 9.20 or later to update the existing request types: `<PPM_Home>/bin/kUpgradeIntegrationRequests.sh`.
Note that the script does not update the field names.

Although the two field groups were renamed respectively since PPM version 9.20, upgrading an existing integration configuration does not upgrade the names of the field groups and their field names. They still remain the same as before. The new field groups and their fields apply to new integration configurations you add in PPM version 9.20 or later.

ALM - Defect Template with Quality Center Integration Request Type and Workflow

The ALM - Defect Template with Quality Center Integration request type and the associated ALM - Defect Template with Quality Center Integration workflow are the only ALM entities that can be used only when a particular integration (with Quality Center, in this case) is established. These ALM entities are described in "Integrating PPM with HPE Quality Center, Using ALM " on page 126.

ALM - Change Migration Workflow

The ALM - Change Migration workflow is used to migrate changes from the DEV environment to the TEST environment and from the TEST environment to the LIVE environment. Figure 3-13 shows the workflow. The ALM - Change Migration workflow is the default workflow used in step 15 of the ALM - Request For Change workflow to create a package. See "ALM - Request For Change Workflow" on page 43.

Figure 3-13. ALM - Change Migration workflow
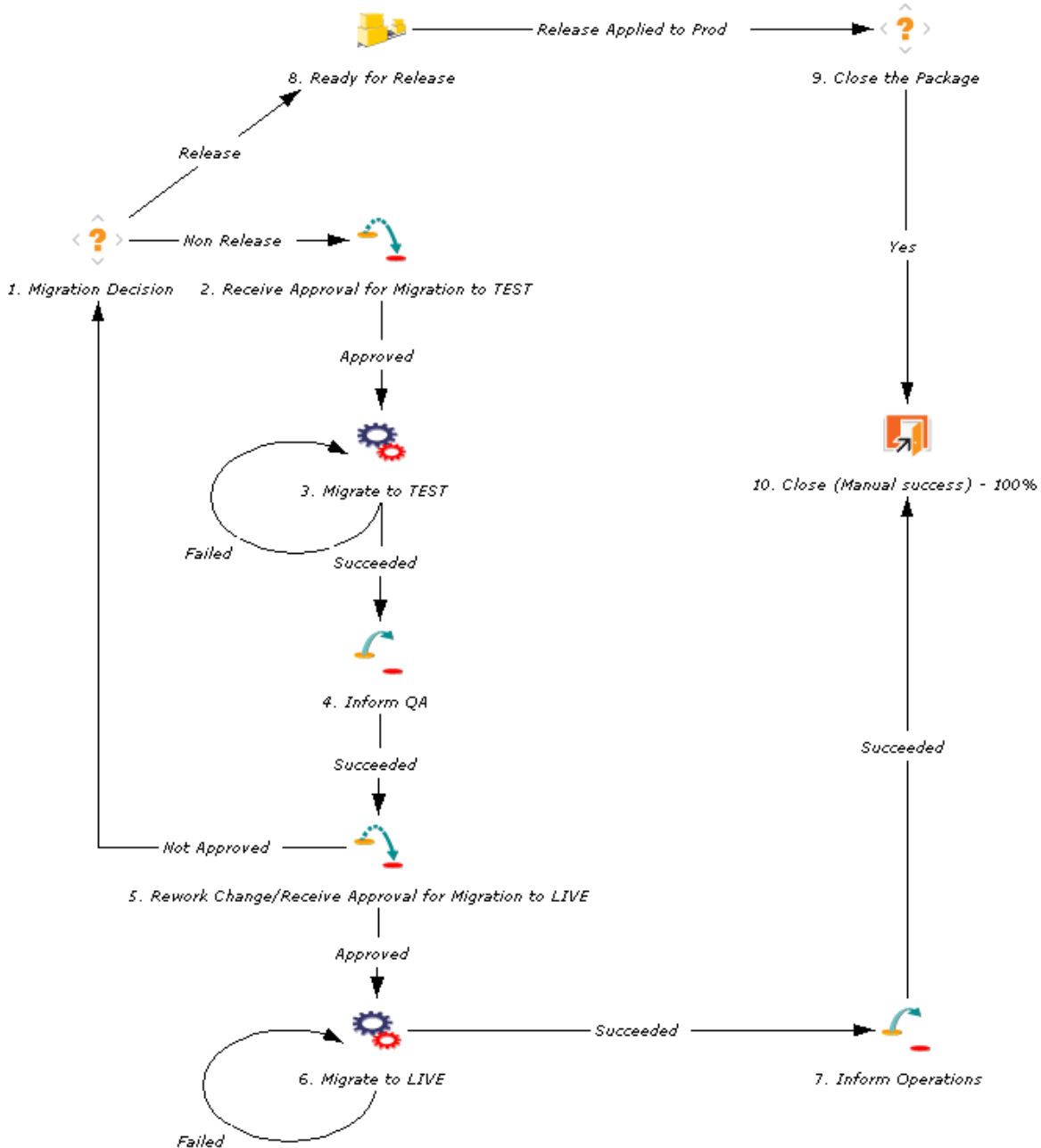
Change Management Portlets to Display KPIs

ALM provides several portlets that can be added to your PPM Dashboard to provide real-time views into several key performance indicators (KPIs).

- **ALM - My RFCs Portlet**

The ALM - My RFCs portlet is provided to users with the role of Change Manager. The portlet lists RFCs that have been created by or assigned to the logged-on user.
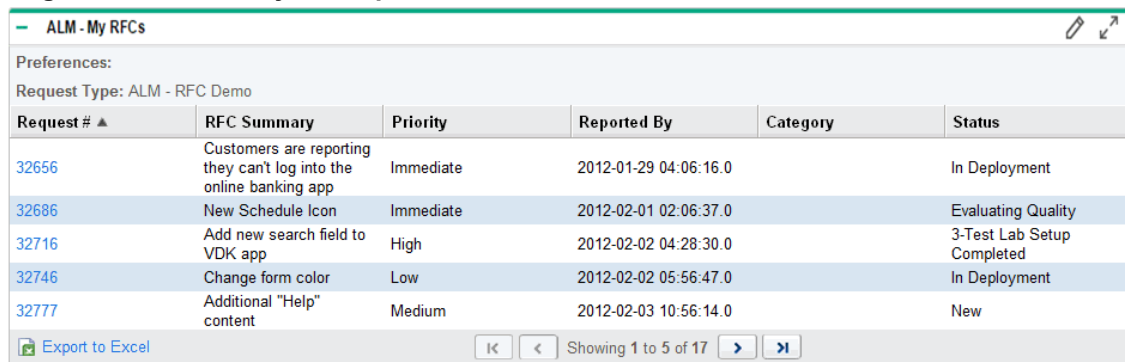
Table 3-9 describes the filter fields for the portlet.

Table 3-9. ALM - My RFCs portlet filter fields

Field Name	Description
Request Type	Request type to filter on
Category	Category of the RFC

Figure 3-14 shows an example ALM - My RFCs portlet.

Figure 3-14. ALM - My RFCs portlet

- **ALM - Open RFCs Portlet**

The ALM - Open RFCs portlet is provided to users with the role of Change Manager. The portlet lists RFCs that are currently being worked on.

Table 3-10 describes the filter fields for the portlet.

Table 3-10. ALM - Open RFCs portlet filter fields

Field Name	Description
Request Type	Request type to filter on
Assigned To	User to whom the RFC is assigned
Category	Category of the RFC

Figure 3-15 shows an example ALM - Open RFCs portlet.

Figure 3-15. ALM - Open RFCs portlet

Request # ▲	RFC Summary	Priority	Reported By	Category	Status
30084	Hiring freeze - need additional resources to finish on schedule	Low	2010-01-18 00:10:25.0		New
30085	Patch release incompatibility with desktop publishing SW	High	2010-01-18 00:10:25.0		New
30086	Possible power outage during cutover weekend	High	2010-01-18 00:10:26.0		New
30087	Glitches in building move during testing	Low	2010-01-18 00:10:26.0		New
30088	Hardware delivery is late	Normal	2010-01-18 00:10:27.0		New

Export to Excel — Showing 51 to 55 of 200

- **ALM - RFCs By Category Portlet**

  The ALM - RFCs By Category portlet is provided to users with the role of Change Manager. The portlet displays a pie chart showing the percentage of RFCs in each category.

  The only filter field for the portlet, **Request Type,** is the request type to filter on.

  Figure 3-16 shows the ALM - RFCs By Category portlet.

  Clicking the pie chart drills down to a list portlet.

Figure 3-16. ALM - RFCs By Category portlet
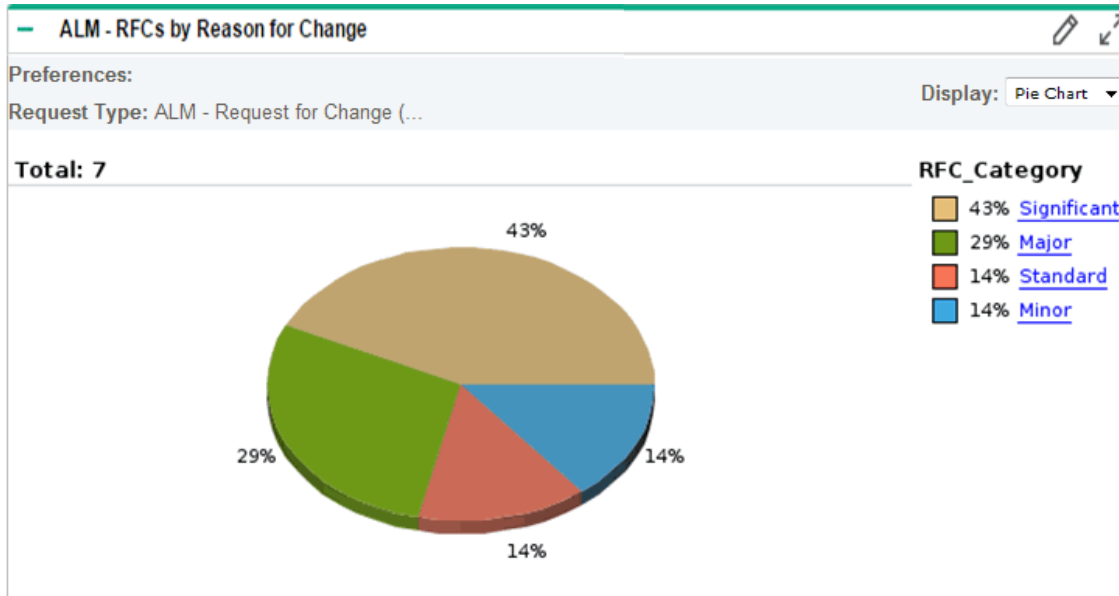
- **ALM - RFCs By Reason for Change Portlet**

The ALM - RFCs By Reason for Change portlet is provided to users with the role of Change Manager. This portlet displays a pie chart showing the percentage of RFCs by each type of change being considered (for example, new feature, enhancement, or defect fix).

Table 3-11 describes the filter fields for the portlet.

Table 3-11. ALM - RFCs By Reason for Change portlet filter fields

Field Name	Description
Request Type	Request type to filter on
Status	Status of the RFC

Figure 3-17 shows an example ALM - RFCs By Reason for Change portlet.

Clicking the pie chart drills down to a list portlet.

Figure 3-17. ALM - RFCs By Reason for Change portlet
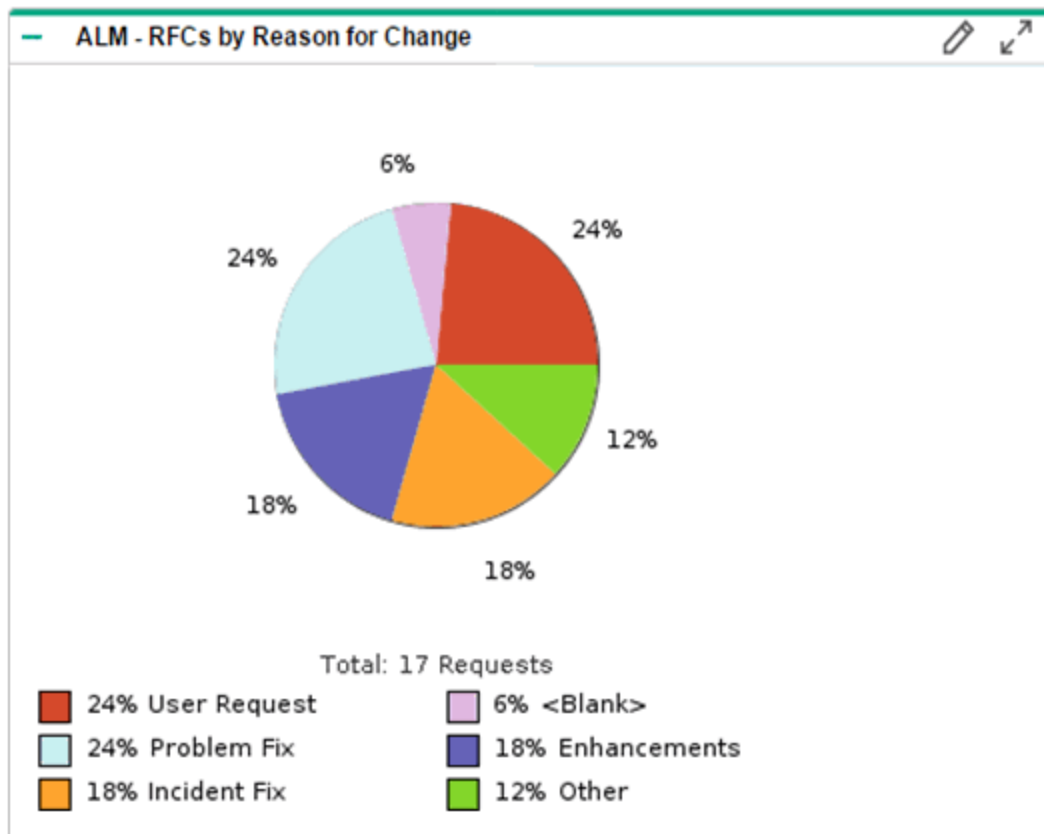
- **ALM - RFCs By Status Portlet**

The ALM - RFCs By Status portlet displays a pie chart showing the percentage of RFCs of each status.

The only filter field for the portlet, **Request Type,** is the request type to filter on.

Figure 3-18 shows an example ALM - RFCs By Status portlet.

Clicking the pie chart drills down to the ALM - RFCs By Status - List portlet.

Figure 3-18. ALM - RFCs By Status portlet
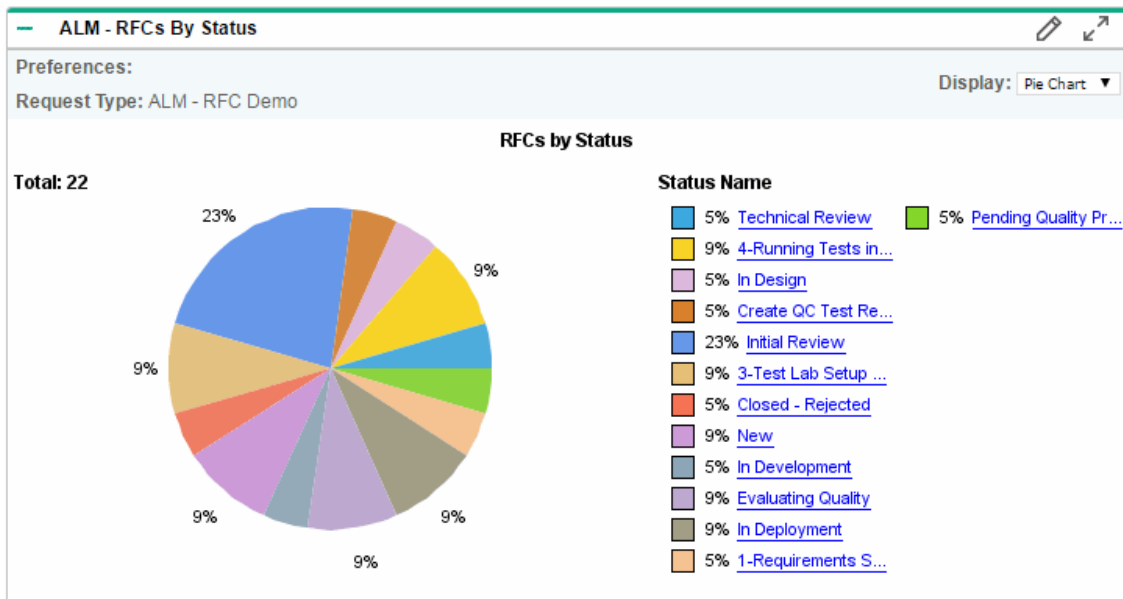
For information about adding portlets to your PPM Dashboard, see the *Getting Started* guide.

Change Management Reports

ALM provides several reports that can be run to provide summary data and scheduling information about RFCs in the system, as well as to process participant data for SOX auditing. To run an ALM report:

1. Log on to PPM.

2. From the menu bar, select **Open > Reports > Create Report.**

3. On the Submit New Report page, in the **Report Category** field, select **Demand Management.**

4. Click the link for the desired report, and complete all required and any optional filter fields.

5. Click **Submit.**

For more information about reports, see the *Reports Guide and Reference*.

ALM - Change Summary Report

The ALM - Change Summary Report provides a list of RFCs that have been implemented, grouped by change category.

Table 3-12 describes the filter fields for the report.

Table 3-12. ALM - Change Summary Report filter fields

Field Name	Description
Request Type	Request type to filter on
Change Status	Status of the change request
Change Priority	Priority of the change request
Time Period From	Earliest date the RFCs were created
Time Period To	Latest date the RFCs were created

Figure 3-19 shows sample output for the ALM - Change Summary Report.

Figure 3-19. ALM - Change Summary Report output

ALM - Forward Schedule of Changes for RFC Report

The ALM - Forward Schedule of Changes for RFC report is a key report used in the change management process. The output of this report is a list of all the RFCs that are scheduled to be implemented.

Table 3-13 describes the filter fields for the report.

Table 3-13. ALM - Forward Schedule of Changes for RFC report filter fields

Field Name	Description
Report Title	Title of the report. Type any alphanumeric string (up to 200 characters in length)

Table 3-13. ALM - Forward Schedule of Changes for RFC report filter fields, continued

Field Name	Description
Start FSC Period	Earliest start date of the scheduled RFCs
End FSC Period	Latest start date of the scheduled RFCs
Request Type	Request type to filter on

Figure 3-20 shows sample output for the ALM - Forward Schedule of Changes for RFC report.

Figure 3-20. ALM - Forward Schedule of Changes for RFC report output

ALM - Release Management Request Type

The ITIL Release Management process as modeled by ALM sends a release request of the ALM - Release Management request type along the ALM - Release Request workflow (see "ALM - Release Request Workflow" on page 81) to be examined and resolved.

ALM - Release Management Request Fields

Figure 3-21 shows the Create New ALM - Release Management page that appears when you create a

request and select the ALM - Release Management request type. Table 3-14 describes the fields in the ALM - Release Management request, including some fields that do not appear until the request is created or until other conditions are met.

Figure 3-21. ALM - Release Management request

Create New ALM - Release Management

Submit   Cancel

Expand All | Collapse All

− Summary

Request Status:

Logged

*Release Category: *Release Type:

*Release Summary:

Release ID:

− Work Item Fields

Scheduled Start Date: **Actual Start Date:**

Scheduled Finish Date: **Actual Finish Date:**

Scheduled Duration: **Actual Duration:**

Scheduled Effort: **Actual Effort:**

Workload?   **Workload Category:**
○ Yes ○ No

Role:

− QC/ALM Info

QC/ALM Instance   **QC/ALM Domain**

QC/ALM Project   **Assigned To**

Requirement No.   **Requirement Status**

Requirement Integration Message   **Synchronize to QC/ALM Requirement**
                                                               ○ Yes ○ No

Requirement Attachment URL   **QC/ALM Requirement Type:**
(No Link)

QC/ALM Requirements coverage

QC/ALM Open Defects

− Release Planning

Release Definition Summary: **Release Definition:**
                                             (No Document Attached)   Add

Release Policy Summary: **Release Policy:**
                                             (No Document Attached)   Add

Release Plans Summary: **Release Plans:**
                                             (No Document Attached)   Add

− QA Information

Release Test Plan Summary: **Release Test Plan:**
                                             (No Document Attached)   Add

Release Acceptance Criteria Summary: **Release Acceptance Criteria:**
                                             (No Document Attached)   Add

Test Results:
(No Document Attached)   Add

Known Defects Summary: **Known Defects:**

− Release Preparation

Communication Plan Summary: **Communication Plan:**
                                             (No Document Attached)   Add

Training Plan Summary: **Training Plan:**
                                             (No Document Attached)   Add

Release Backout Plans Summary: **Release Backout Plans:**
                                             (No Document Attached)   Add

License Agreements: **Support Agreements:**
(No Document Attached)   Add   (No Document Attached)   Add

Service Level Agreements: **Leasing Agreements:**
(No Document Attached)   Add   (No Document Attached)   Add

− Notes

    + Add notes

+ References

Table 3-14. ALM - Release Management request fields

Field Name (*Required)	Description
Summary section	
Request Status	(Read-only) Status of the release request. Preset to **Logged** before the request is created.
Request No.	(Read-only. Added after the request is created.) Number of the request.
*Release Category	Release category, based on the scope of the release (number of changes in a release).
*Release Type	Type of release.
*Release Summary	Summary description of the release.
*Release ID (Required only after the request is created)	ID for the release.
Work Item Fields section [a]	
QC/ALM Info section [b]	
QC/ALM Instance	URL of the Quality Center/ALM instance with the project used for the integration.
QC/ALM Domain	Domain of the project in Quality Center/ ALM.
QC/ALM Project	Quality Center/ALM project that is integrated with this request type.
Assigned To	Developer the QC/ALM Requirement is assigned to.
Requirement No.	(Read-only) Requirement number in Quality Center/ALM.
Requirement Status	(Read-only) Status of the requirement in Quality Center/ALM.
Requirement Integration Message	(Read-only) Quality Center/ALM status message indicating success or error in the most recent operation.
Synchronize to QC/ALM Requirement	Select Yes or No radio button to indicate synchronizing to Quality Center/ALM requirement or not.
Requirement Attachment URL	(Read-only) URL of the list of attachments to the Quality Center/ALM requirement.
QC/ALM Requirement Type	Quality Center/ALM requirement type.
QC/ALM Requirement Coverage	(Read-only) The Quality Center/ALM requirement coverage

Table 3-14. ALM - Release Management request fields, continued

Field Name (*Required)	Description
QC/ALM Open Defects	(Read-only) Number of open defects in Quality Center/ALM.
Release Planning section	
*Release Definition Summary	Summary of the definition of this release.
Release Definition	Allows you to add and view the release definition document directly on the release request.
*Release Policy Summary (Required only after the request is created)	Summary of the policy that governs this release.
Release Policy	Allows you to add and view the release policy document directly on the release request.
Release Plans Summary	Summary of rollout plans for this release.
Release Plans	Allows you to add and view the rollout plans for this release (for example: a timetable of events, a resource plan, and who will do what and when) directly on the release request.
QA Information section	
Release Test Plan Summary	Summary of the test plan for this release.
Release Test Plan	Allows you to add and view the release test plan (the plan that describes tests to be performed on this release in the TEST environment) directly on the release request.
Release Acceptance Criteria Summary	Summary of the release acceptance criteria for this release.
Release Acceptance Criteria	Allows you to add and view the release acceptance criteria document (which details criteria that qualify the acceptance of this release before deployment to the LIVE environment) directly on the release request.
Test Results	Allows you to add and view the test results directly on the release request.
Known Defects Summary	Summary of known defects that will be carried forward into the LIVE environment.
Known Defects	Used to specify RFCs relating to known defects that will be carried forward into the LIVE environment.
Release Preparation section	

Table 3-14. ALM - Release Management request fields, continued

Field Name (*Required)	Description
Communication Plan Summary	Summary of the communication plan for this release.
Communication Plan	Allows you to add and view the communication plan (the plan that describes the various notifications that need to be sent out prior to release deployment into the LIVE environment) directly on the release request.
Training Plan Summary	Summary of the training plan for this release.
Training Plan	Allows you to add and view the training plan (the plan that describes the training that needs to be provided prior to release deployment into the LIVE environment) directly on the release request.
Release Backout Plans Summary	Summary of the backout plans for this release.
Release Backout Plans	Allows you to add and view the backout plan (the release plan that describes procedures to back out the release to its original state) directly on the release request.
License Agreements	Allows you to add and view any license agreement documents for software licensed in this release directly on the release request.
Support Agreements	Allows you to add and view any support agreement documents for support policies of software licensed in this release directly on the release request.
Service Level Agreements	Allows you to add and view any SLAs for ordering new equipment or software directly on the release request.
Leasing Agreements	Allows you to add and view any leasing agreement documents for software leased in this release directly on the release request.

a. Fields in the Work Item Fields section are useful if you want to include Release Management requests as work items to be tracked using Resource Management. For more information, see the *Resource Management User's Guide*.

b. Fields in the QC/ALM Info section remain visible by default but are not used if PPM is not integrated with Quality Center.

Note: The administrator can remove the QC/ALM **Info** section from the request type by removing the QC/ALM Info field group from the ALM - Release Request Header request header type. See the *Demand Management Configuration Guide* for details about request header types and field groups.

How to Submit an ALM - Release Management Request

1. Log on to PPM.

2. From the menu bar, select **Create > Request.**

   The Create New Request page appears.

3. On the Create New Request page, in the **Request Type** field, select **ALM - Release Management** and click **Create.**

   The Create New ALM - Release Management page appears, displaying the appropriate release request fields.

   Required fields have a red asterisk. All other fields are optional, but are often helpful when others are reviewing an open request. For information concerning a specific field, click the **Help** icon next to the field (if available).

4. Complete the fields in all sections as appropriate.

   The **Notes** section contains fields where notes and information concerning the release can be entered and stored. Typically, when you create a release request, you do not need to add a note to it. However, add a note if you want to convey additional information to the reviewers and processors of the release request.

   In the **References** section, you can add useful references such as a Web-accessible file or a document or file attached from a local machine. For more information about adding references, see the *Demand Management User's Guide*.

5. On the Create New Request page, click **Submit.**

   The release request is submitted. The Request Creation Confirmed page appears.

   > **Note:** PPM can be configured to allow you to save the request before you submit it. To have this feature enabled, see your application administrator.

   After submitting the request, on the Request Creation Confirmed page you can click the link for the particular request number in the **Request #** field to view the detail page of the newly generated release request.

   When the release request is submitted, it is assigned an initial status, such as New. The request is then routed along the ALM - Release Request workflow (see "ALM - Release Request Workflow" on the next page).

ALM - Release Request Workflow

The ALM - Release Request workflow is the sequence of approvals, decisions, or actions that the release request follows. The release request starts at the beginning of the ALM - Release Request workflow. When the release request reaches the end of the workflow, its lifecycle is complete. After the release request has been through initial planning, the release is created, and the packages for the RFCs can be aggregated into the release. The deployment steps in the release request workflow coordinate with the actual release to automate the deployment of all packages in the release into the TEST and LIVE environments.

Upon creation, an ALM - Release Management request is automatically set to use the ALM - Release Request workflow.

Figure 3-22 shows the ALM - Release Request workflow. Table 3-15 lists the important steps in the workflow and the user roles associated with those steps.

Figure 3-22. ALM - Release Request workflow
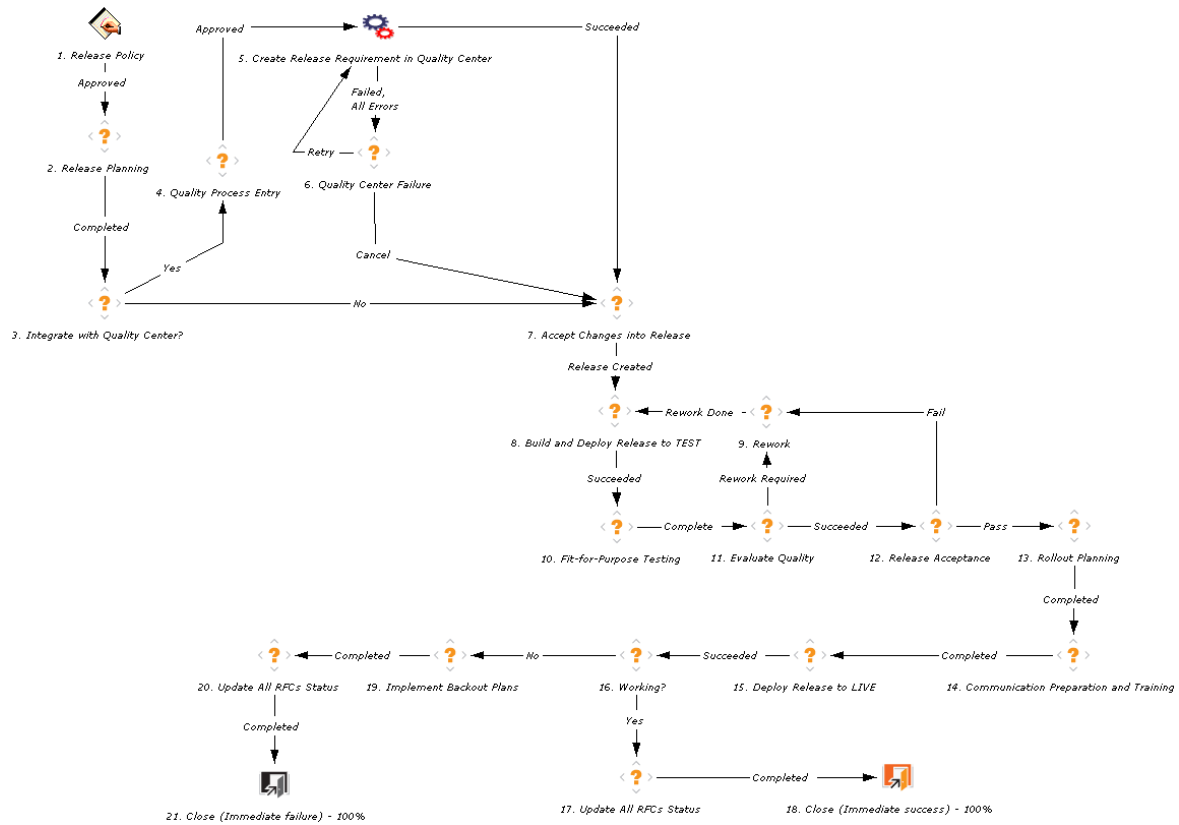
Table 3-15. ALM - Release Request workflow steps

Step Name	User Security	Description
1. Release Policy	ALM - Release Manager	Define the release policy for this release (including release number and rules for accepting changes into the release).
2. Release Planning	ALM - Release Manager	Review and approval of the release policy and other planning documents (such as release acceptance criteria).
3. Integrate with Quality Center?	ALM - QA Manager	Determine whether the user wants to use Quality Center integration, if enabled. See "Integration of PPM with Quality Center/HPE ALM, Using ALM" on page 19.
4. Quality Process Entry	ALM - QA Manager	Secure approval for release entry into Quality Center-integrated process.
5. Create Release Requirement in Quality Center	ALM - QA Manager	Create a test requirement in Quality Center for the release if integration has been enabled.

Table 3-15. ALM - Release Request workflow steps, continued

Step Name	User Security	Description
7. Accept Changes into Release	ALM - Release Manager	Add RFC-related changes to the release. This can be done from the RFC workflow directly (for changes that qualify for this release).
8. Build and Deploy Release to TEST	ALM - Release Manager	Deploy the entire release into QA or staging environment and prepare for integration testing. This step automatically migrates the release and related packages to the TEST environment.
10. Fit-for-Purpose Testing	ALM - Release Manager	Fit-for-purpose testing of this release.
11. Evaluate Quality	ALM - Release Manager	Testing of this release, including testing of backout plan.
12. Release Acceptance	ALM - Release Manager	Based on test results and known defects, determine if this release is acceptable for LIVE deployment.
13. Rollout Planning	ALM - Release Manager	Plan the rollout into LIVE environment. The key document is the release plan (who does what and when).
14. Communication Preparation and Training	ALM - Release Manager	Prepare for LIVE rollout, and determine logistics, training, and communication.
15. Deploy Release to LIVE	ALM - Release Manager	Distribute and install this release to the LIVE environment. This step automatically migrates the release and related packages to the LIVE environment. SOX Segregation of Duties prohibits developers and testers from being involved in deployment of code into production. It is important that users with either of these roles not be involved in this step.
16. Working?	ALM - Release Manager	Determine if the release is working based on sanity check and testing.
17. Update All RFCs Status	ALM - Release Manager	Update the status of RFCs related to this release.
18. Close	(None)	Update status to Closed.

Table 3-15. ALM - Release Request workflow steps, continued

Step Name	User Security	Description
(Immediate success) - 100%		
19. Implement Backout Plans	ALM - Release Manager	If the release is not working, implement the backout plan.
20. Update All RFCs Status	ALM - Release Manager	Update the status of RFCs related to this release.

ALM - Release Distribution Workflow and Subworkflow

The ALM - Release Distribution workflow, along with the ALM - Release Distribution Sub WF subworkflow that it calls to migrate the release to TEST and then to LIVE environments, are used to control distribution among environments of a release that consolidates multiple changes.

Figure 3-23 shows the ALM - Release Distribution workflow.

Figure 3-23. ALM - Release Distribution workflow
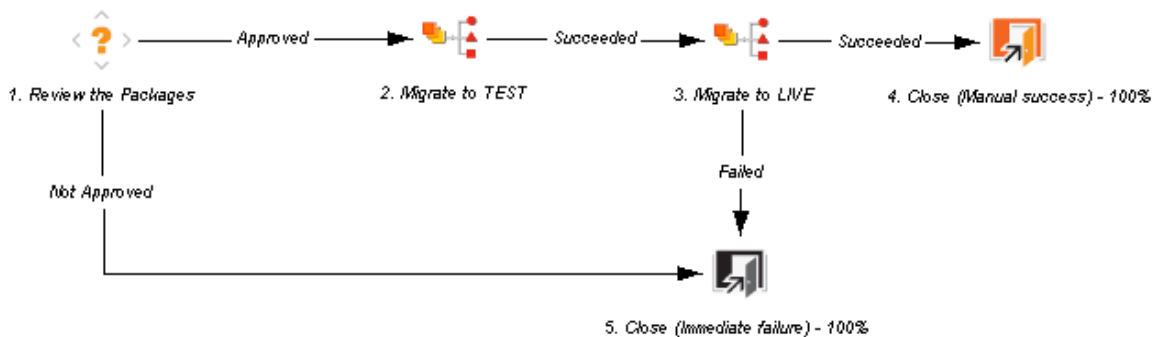
Release Management Portlets to Display KPIs

ALM provides several portlets that can be added to your PPM Dashboard to provide real-time views into several key performance indicators (KPIs).

- **ALM - Deployed Releases Portlet**

The ALM - Deployed Releases portlet is provided to users with the role of Release Manager. The portlet displays a list of recently deployed releases.

Table 3-16 describes the filter fields for the portlet.
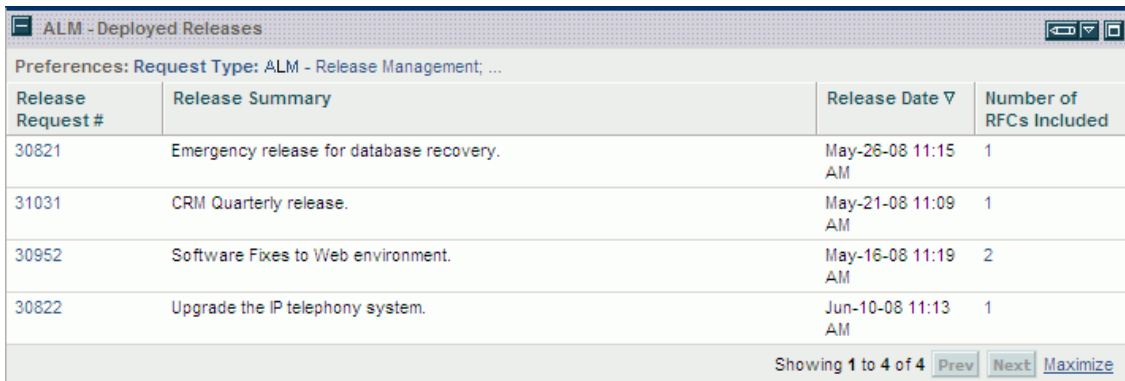
Table 3-16. ALM - Deployed Releases portlet filter fields

Field Name	Description
Child RFC Request Type	Request types, at least one of which must be in the releases that are to be listed
Status	Status of the release
Category	Category of the release (**Emergency, Major,** or **Minor**)
Type	Type of release (**Full, Delta,** or **Package Release**)
Request Type	Request types to filter on

Note: In order for the portlet to display data, RFCs need to be added as child or related references to the release request.

Figure 3-24 shows an example ALM - Deployed Releases portlet.

Figure 3-24. ALM - Deployed Releases portlet

Note: This portlet is only available when you integrate PPM Center with Quality Center 10.00.

- **ALM - My Releases Portlet**

The ALM - My Releases portlet is provided to users with the role of Release Manager and other roles involved in the release management process. The portlet lists release requests that have been created by or assigned to the logged-on user.
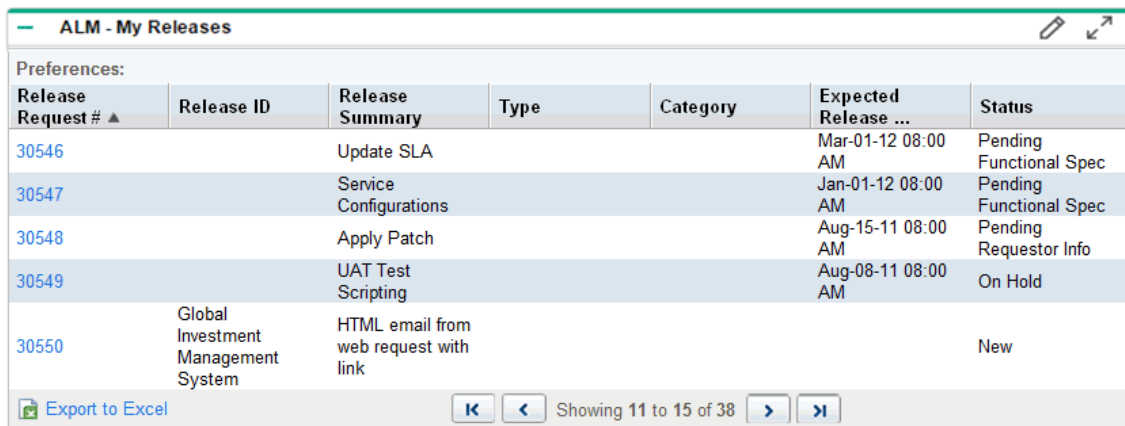
Table 3-17 describes the filter fields for the portlet.

Table 3-17. ALM - My Releases portlet filter fields

Field Name	Description
Request Type	Request type to filter on
Category	Category of the release (**Emergency, Major,** or **Minor**)

Figure 3-25 shows an example ALM - My Releases portlet.

Figure 3-25. ALM - My Releases portlet

- **ALM - Open Releases Portlet**

  The ALM - Open Releases portlet is provided to users with the role of Release Manager. The portlet lists releases that do not have the status of **Closed.**

  Table 3-18 describes the filter fields for the portlet.

  Table 3-18. ALM - Open Releases portlet filter fields

Field Name	Description
Request Type	Request type to filter on
Assigned to	User assigned to the release
Category	Category of the release (**Emergency, Major,** or **Minor**)

  Figure 3-26 shows an example ALM - Open Releases portlet.

Figure 3-26. ALM - Open Releases portlet

- **ALM - Releases Portlet**

The ALM - Releases portlet displays a list of release requests. If PPM is integrated with HPERelease Control, the portlet provides a Click to View link in the **View Impact** column for each request to log in to Release Control, where various tabs provide information about the request.

Table 3-19 describes the filter fields for the portlet.

Table 3-19. ALM - Releases portlet filter fields

Field Name	Description
Request Type	Request type to filter on
Status	Status of the releases

Figure 3-27 shows an example ALM - Releases portlet.

Figure 3-27. ALM - Releases portlet

For more information about tabs in Release Control, see "Using the Integration of PPM with Release Control " on page 272.

- **ALM - Releases By Category Portlet**

  The ALM - Releases By Category portlet is provided to users with the role of Release Manager. The portlet displays a pie chart showing the percentage of releases in each category.

  The only filter field for the portlet, **Request Type,** is the request type to filter on.

  shows an example ALM - Releases By Category portlet.

  Figure 3-28. ALM - Releases By Category portlet
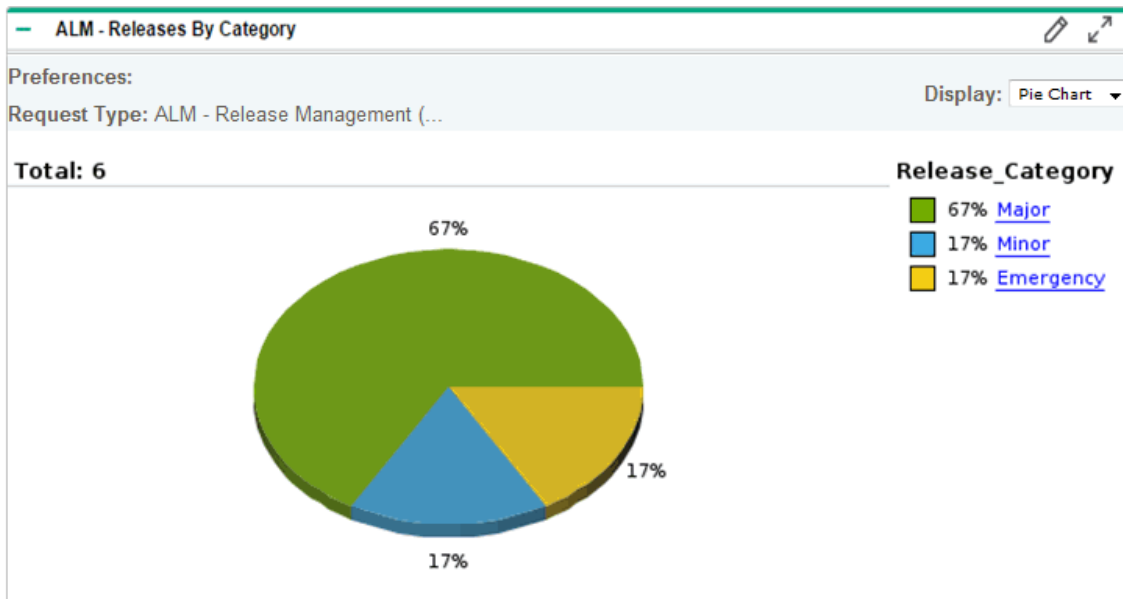
  Clicking the pie chart drills down to the ALM - Releases By List portlet.

- ALM - Releases By Type Portlet

  The ALM - Releases By Type portlet is provided to users with the role of Release Manager. The portlet displays a pie chart showing the percentage of releases of each type.

  The only filter field for the portlet, **Request Type,** is the request type to filter on.

  shows an example ALM - Releases By Type portlet.

  Figure 3-29. ALM - Releases By Type portlet
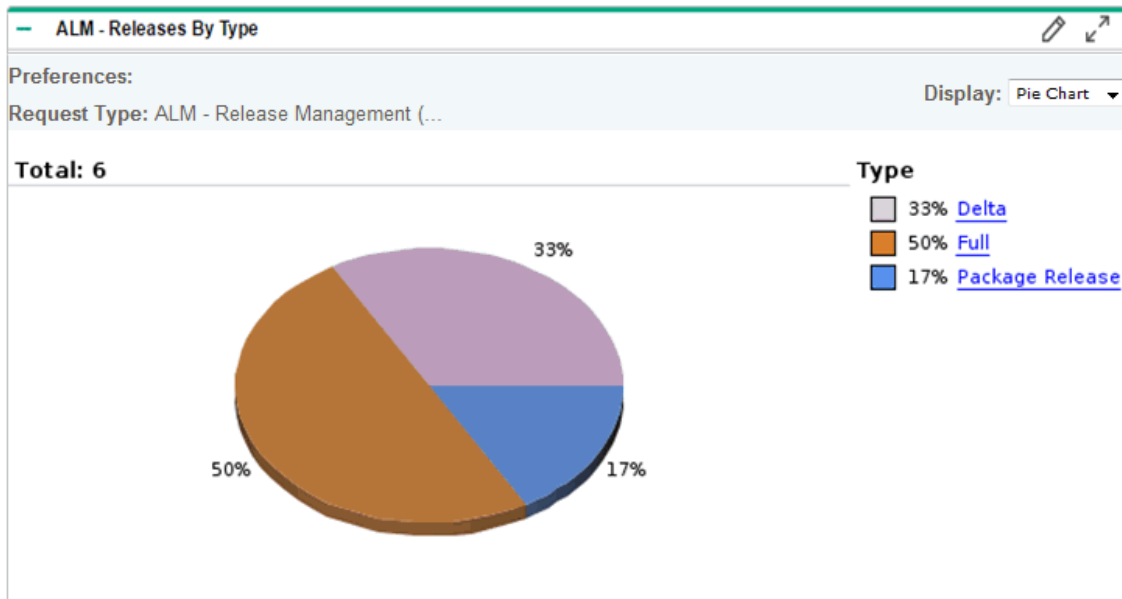
Clicking the pie chart drills down to the ALM - Releases By List portlet.

- **ALM - RFCs per Release Portlet**
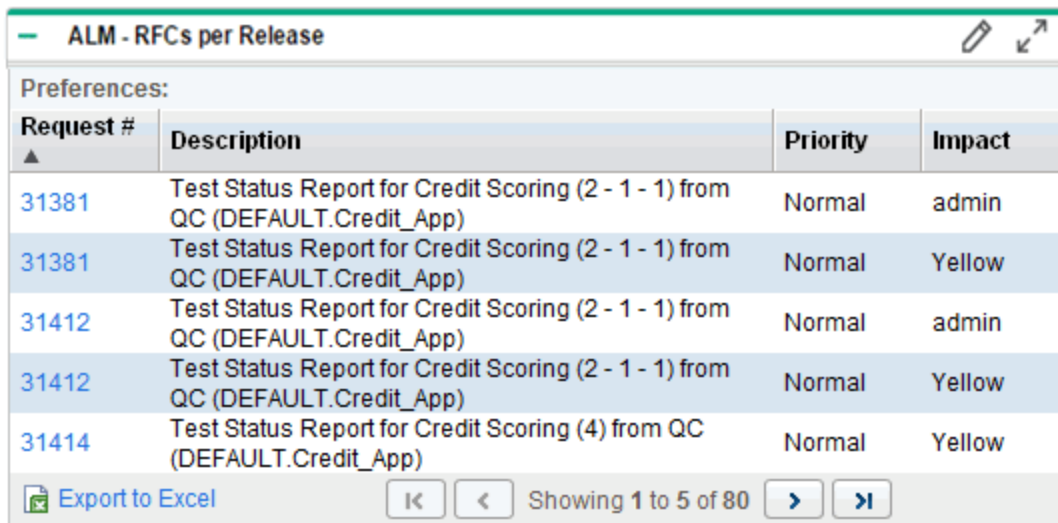
The ALM - RFCs per Release portlet is provided to users with the role of Release Manager. The portlet displays a list of RFCs that constitute a release.

Table 3-20 describes the filter fields for the portlet.

Table 3-20. ALM - RFCs per Release portlet filter fields

Field Name	Description
Release Request	Number of the release request
Request Type	Request types to filter on
Status	Status of the release
Priority	Priority of the release

Figure 3-30 shows an example ALM - RFCs per Release portlet.

Figure 3-30. ALM - RFCs per Release portlet

Request #▲	Description	Priority	Impact
31381	Test Status Report for Credit Scoring (2 - 1 - 1) from QC (DEFAULT.Credit_App)	Normal	admin
31381	Test Status Report for Credit Scoring (2 - 1 - 1) from QC (DEFAULT.Credit_App)	Normal	Yellow
31412	Test Status Report for Credit Scoring (2 - 1 - 1) from QC (DEFAULT.Credit_App)	Normal	admin
31412	Test Status Report for Credit Scoring (2 - 1 - 1) from QC (DEFAULT.Credit_App)	Normal	Yellow
31414	Test Status Report for Credit Scoring (4) from QC (DEFAULT.Credit_App)	Normal	Yellow

Export to Excel | Showing 1 to 5 of 80

For information about adding portlets to your PPM Dashboard, see the *Getting Started* guide.

Release Management Reports

ALM provides several reports that can be run to provide summary data and scheduling information about releases in the system.

To generate a report, from the PPM menu bar:

1. Select **Open > Reports > Create Report.**

2. On the Submit New Report page, in the **Report Category** field, select **Demand Management.**

3. Click the link for the desired report, and complete all required and any optional filter fields,

4. Click **Submit.**

For more information about reports, see the *Reports Guide and Reference*.

- **ALM - Forward Schedule of Releases Report**

  The ALM - Forward Schedule of Releases report is a key report used in the release management process. The output of this report is a list of all releases that are scheduled to be implemented into the LIVE environment.

  Table 3-21 describes the filter fields for the report.

Table 3-21. ALM - Forward Schedule of Releases report filter fields

Field Name	Description
Report Title	Title of the report. Type any alphanumeric string (up to 200 characters in length).
Start FSC Period	Searches for releases created after the specified FSC date.
End FSC Period	Searches for releases created before the specified FSC date.
Request Type	Request type to filter on.

Figure 3-31 shows sample output for the ALM - Forward Schedule of Releases report.

Figure 3-31. ALM - Forward Schedule of Releases report output

- **ALM - Release Content Report**

  The ALM - Release Content Report provides a list of RFCs that have been incorporated into a release.

  The table below describes the filter fields for the report.

Table 3-22. ALM - Release Content Report filter fields

Field Name (*Required)	Description
Report Title	Title of the report. Type any alphanumeric string (up to 200 characters in

Table 3-22. ALM - Release Content Report filter fields, continued

Field Name (*Required)	Description
	length).
Request Type	Request type to filter on.
*Release ID	Specify the release whose contents you want to list.

Figure 3-32 shows sample output for the ALM - Release Content Report.

Figure 3-32. ALM - Release Content Report output

- **ALM - Release Summary Report**

  The ALM - Release Summary Report provides a list of releases that have been implemented.

  Table 3-23 describes the filter fields for the report.

Table 3-23. ALM - Release Summary Report filter fields

Field Name	Description
Request Type	Request type to filter on.
Release Status	Searches for releases with the specified status.
Release Type	Searches for releases of a specific type.
Time Period From	Searches for releases created after the specified date.
Time Period To	Searches for releases created before the specified date.

Figure 3-33 shows sample output for the ALM - Release Summary Report report.

Figure 3-33. ALM - Release Summary Report output

Special Commands

Table 3-24 describes the special commands provided by ALM content bundle to support integration of PPM with Quality Center.

Table 3-24. Special commands for integration of PPM Center with Quality Center

Special Command	Description
ksc_create_defect_in_QC	Creates an execution step that will create a defect in Quality Center
ksc_create_requirement_in_QC	Creates an execution step that will create a requirement in Quality Center
ksc_get_QC_Assigned_To_List	Gets domain users from Quality Center
ksc_get_QC_Defect_Instance_ Domains_List	Gets a list of Quality Center instance domains for defect
ksc_get_QC_Defect_Instance_ Projects_List	Gets a list of Quality Center instance projects for defect
ksc_get_QC_Domain_List	Get a list of Quality Center domains
ksc_get_QC_Instance_Domains_ List	Get a list of Quality Center instance domains
ksc_get_QC_Instance_List	(Not in use) Gets a list of Quality Center instances
ksc_get_QC_Instance_Projects_List	Gets a list of Quality Center instance projects
ksc_get_QC_Project_List	Gets the available project for the Quality Center
ksc_run_QC_Automated_Testset	Runs Quality Center automated test set

For more information about how to use special commands, see the *Commands, Tokens, and Validations Guide and Reference*.

Chapter 4: Introduction to Agile Open SDK

Starting from version 9.30, PPM Center provides a solution called Agile Open SDK. With this solution, PPM Center can be integrated with any agile management system (both internal agile tools and external agile tools) by using different connectors. Therefore, this solution provides PPM project managers with a more comprehensive approach to manage hybrid projects in PPM Center.

PPM uses this solution in the following two integrations:

- "Integrating PPM Tasks with HPE Agile Manager" on page 98

- "Integrating PPM Center Time Sheets with Agile Manager" on page 116

Workflow of Agile Open SDK

The following steps illustrates how Agile Open SDK works in integrating PPM Center with agile management tools.

The descriptions about the workflow would be exemplified by the integration with Agile Manager since the integration is realized by Agile Open SDK in version 9.30.

1. Develop a connector.

   To start using Agile Open SDK, you need an agile tool related connector. This connector conveys to PPM Center such information as what data the agile project has and how the agile tool creates data through mapping with PPM. Agile Open SDK provides development rules for connector developers.

   For more information about developing a connector, download the development guide from the **Agile Open SDK, Version 2.0** section in
   https://lnast01pmp.saas.hpe.com/contentoffering/agile-manager-integration-plug-ppm.

2. Deploy the connector.

   After the connector is created, the administrator needs to deploy the connector on PPM Center.

   For example, the connector for the integration between PPM Center and Agile Manager is deployed on PPM Center by running the following command:

   `sh ./kDeploy.sh -i Connector-AGM`.

3. Configure instances.

After the connector is deployed on PPM Center, the connector (the icon, name, and version of the agile tool) is displayed in the **Hybrid Project** tab of the integration configuration landing page.

For example, the Agile Manager connector is displayed as follows:

qm Agile Manager Connector 2.0

As an administrator, you need to click the add icon next to the connector to add an agile instance. The instance information often contains instance name and base URL of the agile tool. More instance required information can be designed by the agile connector.

> **Note:** A connector can have more than one instance.

4. Configure PPM entities.

   ○ For Integration between PPM Center tasks and Agile Manager, see "Configuring PPM Center Project" on page 103

   ○ For integration between PPM Center time sheets and Agile Manager, see "Configuring the Integration between PPM Center Time Sheets and Agile Manager" on page 116.

5. Import data from HPE Agile Manager to PPM Center.

   ○ For Integration between PPM Center tasks and Agile Manager, see "Mapping a PPM Task to Agile Manager" on page 108

   ○ For integration between PPM Center time sheets and Agile Manager, see"Importing Agile Effort from Agile Manager to PPM Center Time Sheets " on page 118

Part 2: Integration with HPE Agile Manager

This part includes the following solution integration:

- "Integrating PPM Tasks with HPE Agile Manager" on page 98

- "Integrating PPM Center Time Sheets with Agile Manager" on page 116

Starting from version 9.30, the integration with Agile Manager is realized by the solution Agile Open SDK. For information about Agile Open SDK, see "Introduction to Agile Open SDK" on page 95.

Starting from version 9.40, the integration with Agile Manager uses Agile Manager Connector 2.0 which is installed upon installation of or upgrade to PPM 9.40.

Agile Manager Connector 1.0 is deprecated in PPM 9.40. However, the mappings between PPM tasks and Agile Manager established by using Agile Manager Connector 1 still remains after the upgrade to PPM 9.40. For information about using Agile Manager Connector 1, see the *Solution Integrations Guide* for 9.30.

Chapter 5: Integrating PPM Tasks with HPE Agile Manager

This section includes the following topics:

Introduction to Integrating PPM Tasks with HPE Agile Manager

PPM integrates with Agile Manager using the Agile Manager connector. The integration between PPM tasks and Agile Manager releases allows project managers, program managers, portfolio managers, and other project stakeholders to have:

- Visibility into real-time status and progress of agile development projects from within PPM, without having to log on to Agile Manager for details

- A consolidated view of their tasks and agile development initiatives

The integration allows project managers to map a task under a project to a specific release managed in Agile Manager. Each PPM task is associated with a single release in Agile Manager throughout the task life cycle. You can also associate multiple tasks with a single release in Agile Manager. This is a one-way one-to-one (recommended) or multiple-to-one mapping relationship. This association begins when a project manager maps one task to a specific release.

After the mapping relationship is established, the project managers are able to view real-time agile development related charts for a release retrieved from Agile Manager. In addition, project managers can also view the actual data for an Agile Manager release such as actual effort and related resources after synchronization from Agile Manager.

Consider the following about this integration solution:

- The integration can synchronize sprints and user stories of a mapped release from Agile Manager to PPM work plan as a tree of tasks.

- The integration works only when there are work plan(s) in a project. This integration is based on task level. If there is no work plan, there is no way to map any agile project.

- Currently this integration solution supports integrating with Agile Manager only.

- This integration can retrieve the Theme Status, Feature Status, Release Burn Up, and Sprint Burn Down charts from Agile Manager.

- Every time the External Work Plan Sync Service runs, it removes previous synchronized tasks and re-adds them. You may risk data loss if you modify anything regarding the synchronized tasks.

- For imported or synchronized tasks, project managers cannot modify them in work plans. They cannot be tracked using time sheet either.

The integration solution information is stored in the PPM_INT_SOLUTIONS_NLS table, and all Agile Manager server information related to the integration configurations are stored in the PPM_INT_CONFIGURATIONS table. Mapping information about PPM projects and Agile Manager releases are stored in the PPM_INT_AGILE_TASK_MAPPING table. To view error logs, see the PPM_INT_EVENTS table.

For information about the Agile Manager versions supported for integration, see the *System Requirements and Compatibility Matrix*.

For more information about Agile Manager, see Agile Manager documentation.

Workflow of Integrating Agile Manager with PPM Center Tasks

Step 1: Configuring Agile Manager instances

System administrators add an Agile Manager instance in the integration configurations landing page.

For more details, see "Configuring Agile Manager Instances" on the next page.

Step 2: Configuring PPM Center projects

With the Agile Manager instances available, project managers need to configure the PPM projects whose tasks are to be mapped with Agile Manager projects. After the configuration, the **Hybrid Project** tab appears on the Task Details page.

For more details, see "Configuring PPM Center Project" on page 103

Step 3: Mapping Agile Manager projects to PPM Center tasks

When all the configuration works are done, project managers should go to the **Hybrid Project** tab on the Task Details page of the task that is to be mapped with Agile Manager projects.

For more details, see "Mapping a PPM Task to Agile Manager" on page 108.

Step 4: Sync Service

After the mapping relationship is established, the External Work Plan Sync Service works for data sync between Agile Manager projects and PPM projects.

For more information about this service, see "Synchronization Rules" on page 106.

Step 5: Viewing information of the integration with Agile Manager

After the synchronization, project managers can view the following information in different locations of PPM Center:

- In the **Hybird Project** tab of the Task Details page: Sprint Burn Down chart, Release Burn Up chart, Theme Status chart, Feature Status chart. Project managers can decide the charts to be displayed or hidden in the mapping process.

- In the work plan: synchronized data for different Agile Manager projects, including the Agile Manager sprints and user stories with actual efforts and related resources.

  The **Quick** view of the work plan also provides a filter for tasks mapped to Agile Manager projects.

- On the Project Overview page: Agile Manager connector icon in the Overall Status section.

For more information, see "Viewing Mapped Agile Release Information" on page 110.

Configuring Agile Manager Instances

- "Adding an Agile Manager Instance" below

- "Modifying a Agile Manager Instance" on page 102

- "Configuring Global Proxy (Optional)" on page 102

Adding an Agile Manager Instance

Administrators need to add an Agile Manager instance and then project managers can map their projects to any agile releases managed in Agile Manager.

To add an Agile Manager instance,

1. Log on to PPM.

2. From the menu bar, select **Open > Administration > Integrations.**

   The integration configuration landing page opens.

3. Click **Hybrid Project** in the navigation pane.

4. In the Instances section, click **+** on the right of **Agile Manager Connector 2.0**

   The fields for registering a new instance are available in the Update Instance section.

5. Complete the fields described in the following table.

Field (*Required)	Description
*Instance Name	Specify a unique name for the Agile Manager instance. **Note:** The instance name shall not exceed 50 characters. Do not contain such special characters as `<>`.

*Base URL	URL of the Agile Manager server you want to integrate with. Format of the URL: `http(s)://<agile_server_address>/`
	
	For example, Agile Manager SaaS URL:
	
	`https://agmast01.saas.hp.com/`
	
	Note: If you have customized the Agile Manager server, make sure to include the port number in the URL as well:
	
	`http(s)://<agile_server_address>:<port>/`

Proxy Host	Host name of the proxy if you have configured proxy for PPM Server to access the Internet.
Proxy Port	Port of the proxy if you have configured proxy for PPM Server to access the Internet.
Use Global Proxy	Flags whether or not to use the proxy specified in the parameter `HTTP_PROXY_URL`. For details, see "Configuring Global Proxy (Optional)" on the next page.
*Client ID	The ID created for PPM via the Agile Manager API. Clients are assigned to specific workspaces and roles, similar to Agile Manager users. For information about generating client ID, see Agile Manager documentation.
*Client Secret	The password for a client. For information about generating client ID, see Agile Manager documentation.

6. Click **Save.**

    The instance is added to the list under Agile Manager Connector 2.0 in the Instances section.

    > **Note:**
    >
    > ○ The number on the right of a instances name indicates how many times this instance is used for the integration between PPM Center tasks and Agile Manager.
    >
    > ○ As for the two icons (**WP** and **TS**) in the Update Instance section of the integration landing page, the **WP** icon indicates that the instance is used to integrate Agile Manager with PPM Center tasks, and the **TS** icon indicates that the instance is used to integrate Agile Manager with PPM Center time sheets.

Modifying a Agile Manager Instance

1. Log on to PPM.

2. From the menu bar, select **Open > Administration > Integrations.**

    The integration configuration landing page opens.

3. Click **Hybrid Project** in the navigation pane.

4. From the list under Agile Manager Connector 2.0 in the Instances section, click the desired instance you want to modify.

    The instance details are displayed in the Update Instance section.

    > **Caution:** For an instance in use, which has a number on the right, HPE recommends that you do not modify the Base URL field in case of data loss. If you save the instance with wrong data, you can create a new one before using it to map a PPM Center task to an Agile Manager project.

5. Modify the values of the fields you want to edit.

6. Click **Save** to save the changes.

You cannot delete an instance in use until all the corresponding references are deleted.

Configuring Global Proxy (Optional)

To configure the global proxy,

1. Log on to PPM.

2. From the menu bar, select **Open > Administration > Open Administration Console**.

   The Administration Console opens.

3. In the left navigation pane, expand **Administration Task > Application configuration.**

4. Search for the HTTP_PROXY_URL parameter.

5. Click in the **Value** field, provide a valid HTTP proxy server URL, in the format of *<Proxy_Server_ IP>:<Port>* or *<Proxy_Server_Domain>:<Port>*.

6. Click **Save**.

Configuring PPM Center Project

If you want to map a PPM Center task to a Agile Manager project, the project to which the task belong should be set as a hybrid project.

To do so,

1. Log on to PPM.

2. From the menu bar, select **Search > Projects**.

   The Search Projects page opens.

3. Locate and open the desired project.

4. Click **Settings** in the upper-right corner of the Project Overview page.

5. Click the **Hybrid Project** tab in the left pane.

6. Select the option **Set the current project as a Hybrid Project**.

   Caution: If the project has been integrated with Time Management, you can set the project as a hybrid project only when time is tracked against the project at task level.

7. Click **Done** to save the setting and return back to the Project Overview page.

Important Notes about Configuring Project Settings

Note the following about setting a project as a hybrid project:

- If you set a project as a hybrid project, and you still want to use time management to track actuals against the project, time can only be tracked at task level.

- If you set a project as a hybrid project, and a task in the project is mapped to a waterfall project or an agile project, you cannot configure the Hybrid Project policy setting for the project type of that project.

  Only when the mapping relationships in the tasks of all the projects of a project type are deleted, can you configure the Hybrid Project policy setting for the project type.

Using the Agile Integration Solution to Manage Agile Projects

Project managers should not use the integration until PPM administrator has completed the configuration tasks described in "Configuring Agile Manager Instances" on page 100 and "Configuring PPM Center Project" on the previous page.

Before using this integration solution, you may want to review the following sections first:

- "Mapping Rules" below

- "Synchronization Rules" on page 106

After initiating a project in PPM as a project manager, you can do the following:

- Map a specific task under the project to an agile project on the Task Details page. See "Mapping a PPM Task to Agile Manager" on page 108.

- View mapped agile project information. See "Viewing Mapped Agile Release Information" on page 110.

- Delete a mapping relationship on the Task Details page. see "Deleting a Mapping Relationship" on page 114.

Mapping Rules

When mapping a task to an agile project, make sure that the task meets the following criteria:

- The task is a leaf task.

- The task is neither a milestone nor a summary task.

  > **Note:** If the task was a milestone (or a summary task) previously, and it was changed to a leaf task, then it is qualified for being mapped to an agile project.

- The task has no actuals rolled up from PPM.

  For example, if you set a task status to **In Process**, it would have actual start date, then it cannot be mapped to any Agile Manager release.

- Microsoft Project-related tasks

  When a project is already integrated with Microsoft Project, the work plan should be fully controlled by PPM (under PPM-controlled mode) if you want to map tasks to an agile project.

  When you already set a project as a hybrid project, the work plan can only be under PPM-controlled mode if you want to integrate the project with Microsoft Project. The other two modes are disabled.

- RFC tasks

  If a task is already mapped to an RFC in Service Manager, it is not available for mapping to an agile project.

If a task does not satisfy the above criteria, a warning message is displayed on the **Hybrid Project** tab, indicating that the task cannot be mapped to an agile release as well as the reasons why it cannot be mapped.

Important Notes about Mapping

When mapping a task to an agile release, also consider the following:

- For imported and mapped tasks, you cannot use time sheet to track time for them.

- Actual start and end dates in Agile Manager might not be consistent with the dates displayed in PPM.

  In Agile Manager, sprints under a release may have different start dates and end dates. The duration between the earliest start date and the latest end date might not cover the entire release time frame. However, PPM retrieves the earliest start date and the latest end date of all sprints under a release as the timeframe for that release. This results in difference between the actual start date and end date of a release in Agile Manager and the dates displayed in PPM.

- Timezone limitations

- If PPM's timezone is different from that of Agile Manager, the Sprint Burn Down chart and Release Burn Up chart retrieved from Agile Manager may display **Today + <*Date from PPM Center*>**.

- If a sprint in Agile Manager is scheduled to end at a time out of the working hours range of PPM, after synchronization, PPM display the end of working hours of the last working day as the sprint end time. For example, if a sprint is scheduled to end on Saturday in Agile Manager, PPM would display the end of working hours of Friday (such as 16:00) as the sprint end time.

Synchronization Rules

The mappings between PPM tasks and Agile Manager releases follow the synchronization rules below:

- If the proxy setting is out of date, the synchronization will fail and save the status.

- If you use MSP plug-in for MSP-PPM integration, and change project setting to MSP controlled or shared mode, the synchronization will skip this mapping and save the status.

- If an Agile Manager project is cancelled or closed, the synchronization will skip this mapping and save the status.

- If a PPM task is cancelled or closed, the synchronization will skip this mapping.

- If an Agile Manager release is deleted, the synchronization will skip this mapping, and save the status.

- The following table lists exceptional cases for the Agile Solution Sync Service.

Table 4-1. Exceptional cases for the Agile Solution Sync Service

Conditions (one of the conditions occur)	Agile Solution Sync Service Behavior
○ PPM project is deleted ○ PPM workplan is deleted ○ PPM task is deleted	Deletes the mapping from the mapping table
○ PPM Project is completed or canceled ○ The mapped PPM task is completed or canceled ○ PPM is integrated with MSP using MSP plug-in, and PPM project is set to Microsoft Project control mode or	Records specific state in the mapping table and skip synchronization on these mappings. Next time the service runs, it checks the state again.

Table 4-1. Exceptional cases for the Agile Solution Sync Service, continued

Conditions (one of the conditions occur)	Agile Solution Sync Service Behavior
shared mode ○ PPM task is mapped to a RFC	
An Agile Manager release is deleted	Records specific state in the mapping table
Current date is later than the end date of the Agile Manager release	Records specific state in the mapping table and still synchronizes these mappings, and sets the state of mapped PPM task to complete.
Network connection issue or user authentication issue	Stops synchronization for all the mappings on the specific server and records status in the mapping table

Important Notes about the External Work Plan Service

Consider the following:

- Frequency of the External Work Plan Sync Service

  The External Work Plan Sync Service synchronizes user stories in all sprints (including the current, the last, and the future sprints) from Agile Manager. The default synchronization frequency is 24 hours. Setting the interval to a value shorter than this may result in frequent import of tasks, which keeps the Cost Rollup Service busy.

  To get the complete information about user stories in an agile release, HPE recommends that you schedule the External Work Plan Sync Service to run at a longer interval, better a value shorter than the duration of the shortest sprint under a release.

  For example, if you have sprints of two weeks duration, the maximum value of the synchronization frequency should be 2*7*24= 336 hours.

- Update modified user stories

  The External Work Plan Sync Service works in the way of first deleting and then adding tasks to update the modified user stories or sprint information. HPE does not recommend you to modify the synchronized tasks in PPM. Otherwise you risk data loss after the synchronization service runs next time.

- Performance of the service

Performance of the External Work Plan Sync Service depends on your organization's project settings as well as the number of backlogs and sprints in Agile Manager Releases. HPE recommends that you do not run the service more than once a day.

Mapping a PPM Task to Agile Manager

To map a PPM Center task to Agile Manager,

1. Log on to PPM.

2. From the menu bar, select **Search > Projects**. The Search Projects page opens.

3. Locate and open the project you initiated.

4. Click **Edit Work Plan**.

5. Select the leaf task you want to map.

6. Double-click the leaf task or select **Edit** > **Task Details** to open the Task Details page of the leaf task.

7. Go to the **Hybrid Project** tab.

8. Click **Start Mapping**.

   The mapping process begins with Step 1: Select Instance.

9. From the **Select Instance** drop-down list, select a instance, with the Agile Manager icon in front of the instance name, that your system administrator registered for project managers.

10. Click **Next**.

    The mapping process moves to Step 2: User Configuration

11. Fill the fields as described in the following table.

Field (*Required)	Description
*Username	The user name you use to log on to the Agile Manager server.
*Password	The password you use to log on to the Agile Manager server.
*Domain	This filed loads after you fill the Username and Password fields. It is based on the account information you provided.

*Project	Select a project to map. It is based on the domain you selected.
*Workspace	Select a workspace in Agile Manager to map. It is based on the project you selected. **Note:** You can see all workspaces including the ones you do not have access to. Connectivity error occurs if you select a workspace you have no access to.
If you want to map the task with an existing release, fill the following three fields.	
*Release	Select a release to map. It is based on the project you selected.
Sprint	Select a sprint to map. It is based on the release you selected.
Level of Details to Synchronize	Select the level of data imported from Agile Manager to the PPM task. If you select a sprint in the Sprint field, the available options in this field are: ○ **User stories**: Each user story of the selected sprint will be imported to the task. ○ **Sprint total only**: Only the selected sprint will be imported to the task, with the effort in all user stories rolled up to the sprint level. If you leave the Sprint field blank, the available options in this field are: ○ **All (Sprints & User Stories)**: Every sprint including the user stories of the sprint will be imported to the task. ○ **Sprints only**: Only every sprint will be imported to the task, with the effort in all user stories rolled up to the sprint level. ○ **Release total only**: Only the selected release will be imported to the task, with the effort in all sprints rolled up to the release level.
If you want to map the task with a new release, fill the following fields.	
Create a New Release	Check this option if you want to create a new release.
*Name	Name of the new release.

Description	Description of the new release.
*Start Date	Start date of the new release.
*End Date	End date of the new release.
*Sprint Duration	Specify a number for the sprint duration. The number combined with the sprint duration number constitutes the sprint duration. For example, if you select 2 for the sprint duration and "week" for the sprint duration unit, each sprint in the new release lasts 2 weeks.
*Sprint Duration Unit	Select the duration unit of a sprint. Available choices are Day and Week.
Release Information	
Show Sprint Burn Down Chart Show Release Burn Up Chart Show Theme Status Chart Show Feature Status Chart	Select the charts you want to display in the Hybrid Project tab after the mapping is established.

12. Click **Next**.

    The mapping process moves to Step 3: Confirmation, displaying the information you provided in the User Configuration step.

13. Click **Submit**.

    The mapping relationship is established. The charts you selected in step 11 are displayed. The Agile Manager connector icon (🔷) appears in the front of the task name, indicating the tasks has been mapped to Agile Manager releases.

Viewing Mapped Agile Release Information

After you map an agile release to a task, you can view different information about the mapped agile release in the Hybrid Project tab and the work plan of the project the task belongs to.

- "Viewing Agile Release Charts in Hybrid Project Tab" on the next page
- "Viewing Agile Release Actual Data in Work Plan" on page 112

Viewing Agile Release Charts in Hybrid Project Tab

After the mapping relationship is established, the following information are displayed int **Hybrid Project** tab right away.

- Agile Manager connector and instance information

- User configuration information

- Agile release related charts:

    - The Release Burn Up chart

    - The Sprint Burn Down chart

    - The Theme Status chart

    - The Feature Status chart

    The charts available depend on your selection in the User Configuration step of the mapping process.

Putting the cursor on a theme in the Theme Status chart, the detailed theme information is displayed. Similarly, if you put the cursor on a date in the Sprint Burn Down Chart, the capacity and remaining efforts of that date are displayed.

Note:

- Putting the cursor over **More** displays the hidden information.

- The last sync time and sync status are only shown after synchronization.

- If you use customized proxy or global proxy for PPM Server to access Agile Manager, the charts are not displayed in the **Hybrid Project** tab on the Task Details page of a mapped task.

Viewing Agile Release Actual Data in Work Plan

After the External Work Plan Sync Service is completed, you can view agile release actual data in the work plan of PPM project, including actual effort, status, actual start/finish date, and resource of an

agile release user story.

In the **Quick** view of the work plan, the tasks mapped with agile projects have the connector icon in front of the task names. You can quickly filter agile project-mapped tasks by clicking . For details, see *Project Management User's Guide*.

The actual data of the agile tasks are synchronized from Agile Manager and are read-only in the work plan. You are not able to edit any information of these agile tasks.

> **Note:** You can view the actual effort of an agile release only from the **Actuals** view of the work plan.

The following table illustrates the one-to-one correspondence between agile release data and PPM tasks data.

Data in Agile Manager	Data in PPM Center
Name	
User story/sprint/release name	Task name
Status	
New	Ready
In Progress	In Progress
In Testing	In Progress
Done	Completed
Scheduled Start	
User story start date	Task scheduled start date

Scheduled Finish	
User story finish date	Task scheduled finish date
Schedule Effort	
Estimation effort of user story tasks	Task scheduled effort
Actual Effort	
Invested hours of user story tasks	Task actual effort
Actual Start	
User story actual start date	Task actual start date If the status of the agile user story is New, the task actual start date is empty.
Actual Finish	
User story actual finish date	Task actual finish date, if the status of the agile user story is Done Otherwise, the task actual finish date is empty.
Percent Complete	
Invested Hours/(Invested Hours + Remaining Hours)	Task percent complete
Resource	
Owners of user story tasks	Task resources

Note: Every time the External Work Plan Sync Service runs, it removes previous synchronized tasks and re-adds them. If an unmapped task is indented or outdented to become the leaf task of a mapped task, the unmapped task is deleted after synchronization. HPE recommends that you do not indent or outdent unmapped tasks to make them become the leaf tasks of the mapped tasks.

Deleting a Mapping Relationship

To delete a mapping relationship,

1. Go to the **Hybrid Project** tab of the Task Details page for the desired task.

2. Next to the Agile Manager instance information, Click **Unlink**.

   A warning message pops up.

3. Click **OK**.

The mapping is deleted immediately.

Chapter 6: Integrating PPM Center Time Sheets with Agile Manager

This section contains the following:

Introduction to Integrating PPM Time Sheets with Agile Manager

The integration between PPM time sheets and Agile Manager enables end users to import agile effort from Agile Manager into their PPM time sheets, freeing them from reporting their time repeatedly and thus ensuring data consistency between different management tools. This integration also ensures information sharing between product development teams and project managers, which facilitates project execution with right decisions.

This integration is a one-way (from Agile Manager to PPM) and one-multiple mapping relationship (one time sheet to multiple releases).

Configuring the Integration between PPM Center Time Sheets and Agile Manager

This section describes the steps that PPM administrators must perform to set up the integration between PPM time sheets and Agile Manager.

Prerequisites

Make sure you satisfy the following prerequisites before you can proceed to set up the integration:

- PPM is on version 9.40.

- You have obtained a valid Agile Manager account.

Setting Up the Integration

To set up the integration,

1. Log on to PPM as an administrator.

2. (Optional) Launch the Administration Console and check that the `ENABLE_TM_WORK_ITEM_EXTERNAL_DATA` server configuration parameter is set to `true`.

Parameter Name	Description, Usage	Default and Valid Values

ENABLE_TM_ WORK_ITEM_ EXTERNAL_DATA	Setting the parameter to `true` makes the **External Data** option available in the following PPM Workbench windows:
	
	○ On the Work Items and Activities tabs of the Time Sheet Policy window
	
	Selecting the **External Data** option on the **Work Items** tab ensures that,
	
	   • The **Add External Data** option is available in the **Add Items** action list of the Time Sheet page
	
	   • The **External Data** option is available on the Create Work Allocations page as a work item type.
	
	For information about the usage of activities, see the *Time Management Configuration Guide* and the *Time Management User's Guide*.
	
	○ In the Work Item Type drop-down list (under the Dependencies section) of the Override
		Default value: `true`
		
		Valid values: `true`, `false`

Parameter Name	Description, Usage	Default and Valid Values
	Rule window	

The parameter is set to `true` by default. If you changed the parameter value to `false`, change it back to `true`.

3. In the PPM Workbench, enable the **External Data** option on the Work Items tab of the desired time sheet policy.

   a. From the menu bar, select **Open > Administration > Open Workbench** (or **Open Workbench on Desktop**).

   The PPM Workbench opens.

   b. From the shortcut bar, select **Time Mgmt > Time Sheet Policies**.

   The Time Sheet Policy Workbench opens.

   c. Click **List**.

   d. From the **Results** tab, select and open the desired time sheet policy.

   e. Click the **Work Items** tab.

   f. Select the checkbox for the **External Data** option.

   > **Note:** This setting applies to new time sheets only. Existing time sheets still load the old time sheet policy, therefore users cannot add external data to their old time sheets.

   g. Click **OK**.

4. Add an Agile Manager instance for the target Agile Manager server system on the integration landing page.

   For detailed instructions, see "Adding an Agile Manager Instance" on page 100.

Importing Agile Effort from Agile Manager to PPM Center Time Sheets

To import agile effort from Agile Manager into your PPM time sheet,

1. Log on to PPM.

2. From the menu bar, select **Create > Time Sheet**.

   The Create Time Sheet page appears.

3. Complete the following fields or option.

   ○ Time Period

   ○ Resource

   ○ Description

   ○ Include Items from My Items List

   For detailed descriptions about these fields, see the *Time Management User's Guide*.

4. Click **Create**.

   The time sheet is created and the Edit Time Sheet page appears.

5. Click **Add Items**, and then select **Add External Data** from the available actions.

   The Import External Data window appears.

   > **Note:** You can only add external data for time sheets where time is entered by Day in Hours policy.

6. From the **Select Instance** drop-down list, select a instance, with the Agile Manager icon in front of the instance name, that your administrator configured.

7. Click **Next**.

   You come to step 2: User Configuration.

8. In the **Username** and **Password** fields, enter the username and password you use to log on to the Agile Manager.

9. Click **Next**.

   The system retrieves and loads invested effort for tasks from available releases in Agile Manager.

The external item name is displayed in the following format:

<Release_Name>(<Domain_Name>><Project_Name>)

Review the list of available releases. If necessary, you can click the ✕ icon to remove the unwanted external item from the list.

10. Click **Submit**.

The Agile Manager efforts are imported to the time sheet. The efforts are allocated in the time sheet as they are in the Agile Manager.

Each release is added as an external item in the time sheet, and the daily effort of a release is equal to the daily total of invested effort of all tasks in the release.

The imported entries are displayed under the **External Data** section.

The time sheet line **Item** name is displayed in the following format:

<Agile_Instance_Name>:<Release_Name>(<Domain_Name>><Project_Name>)

11. Review and edit the time sheet lines if necessary.

12. Click **Save** or **Save&Submit**.

REST APIs Called in the Integration

The following Agile Manager REST APIs are called in this integration to retrieve and import data from Agile Manager into PPM time sheets.

Description	URL
Get graph entity	`https://`*<AgileManager_Server_URL>*`/qcbin/rest/domains/`*<domain name>*`/projects/`*<project_name>*`/analysis-items`
Get graph result	`https://`*<AgileManager_Server_URL>*`/qcbin/rest/domains/`*<domain name>*`/projects/`*<project_name>*`/graphs/`*<graph_name>*`/result`
Sign in	`https://`*<AgileManager_Server_URL>*`/qcbin/authentication-point/alm-authenticate`
Sign out	`https://`*<AgileManager_Server_URL>*`/qcbin/authentication-point/logout`

Get domain list	`https://<`*AgileManager_Server_URL*`>/qcbin/rest/domains`
Get projects of one domain	`https://<`*AgileManager_Server_URL*`>/qcbin/rest/domains/<`*domain_name*`>/projects`
Get work spaces of one project	`https://<`*AgileManager_Server_URL*`>/qcbin/rest/domains/<`*domain_name*`>/projects/<`*project_name*`>/product-groups`
Get releases of one project	`https://<`*AgileManager_Server_URL*`>/qcbin/rest/domains/<`*domain_name*`>/projects/<`*project_name*`>/releases`
Get tasks of one project	`https://<`*AgileManager_Server_URL*`>/qcbin/rest/domains/<`*domain_name*`>/projects/<`*project_name*`>/project-tasks`
Get backlogs of one project	`https://<`*AgileManager_Server_URL*`>/qcbin/rest/domains/<`*domain_name*`>/projects/<`*project_name*`>/release-backlog-otems`
Get sprints of one project	`https://<`*AgileManager_Server_URL*`>/qcbin/rest/domains/<`*domain_name*`>/projects/<`*project_name*`>/release-cycles`
Get task audits	`https://<`*AgileManager_Server_URL*`>/qcbin/rest/domains/<`*domain_name*`>/projects/<`*project_name*`>/project-tasks/<`*task_name*`>/audits`
Create a release	`https://<`*AgileManager_Server_URL*`>/agm/api/workspaces/<`workspace_id`>/releases`
Get time sheet data	`https://<`*AgileManager_Server_URL*`>/agm/api/reports/timesheet`
Get access token	`https://<`*AgileManager_Server_URL*`>/agm/oauth/token`

Troubleshooting and Limitations

This section includes the following:

- "Limitations " on the next page

- "Troubleshooting " on the next page

Limitations

- When you run the following Time Management reports, if you set the **Work Item Type** filter to **External Data**, and the **Work Item Set** filter to **Agile Manager**, no values are available for **Work Item**:

  ○ Actual Time Summary

  ○ Time Sheet Summary

  ○ Work Allocation Details

- When you set filters for the following Time Management portlets, if you set **Work Item Type** to **External Data**, and **Work Item Set** to **Agile Manager**, no values are available for **Work Item**:

  ○ Time Sheet Totals by Work Item

  ○ Time Sheet Totals by Resource Group

  ○ Work Allocation Details

  ○ Work Item Set Actuals vs. Allocations

- **External Data** is available for the following Time Management portlets only:

  ○ Time Sheet Totals for Direct Reports

  ○ Approve Time Sheets

  ○ Work Item Set Actuals vs. Allocations

  ○ Time Sheet Totals by Resource Group

  ○ Time Sheet Totals by Resource

  ○ My Time Sheets

  External Data is not available for the following portlets:

  ○ Time Sheet Totals by Work Item

  ○ Work Allocation Details

  ○ My Tasks

  ○ My Work Items

Troubleshooting

Problem: Importing the agile effort data for multiple times during the time period range does not automatically update the effort data in your PPM time sheet that you imported previously. Instead, it adds new time lines into your time sheets. You need to manually remove the old time lines of data.

Solution: Import your agile effort only once for each time period, approaching the end of the current period, so that the latest and more accurate effort data can be retrieved and imported.

Part 3: Integration with HPE Quality Center/HPE Application Lifecycle Management

This part includes the following solution integrations:

- Integrating PPM with Quality Center, Using ALM

- Integrating PPM Projects with HPE ALM Releases — View Project Quality

- Integrating PPM Tasks with HP ALM Releases

Chapter 7: Integrating PPM with HPE Quality Center, Using ALM

> **Note:** In this section, HPE Quality Center (QC) refers to HPE Quality Center Enterprise Edition version 10.00 (hereinafter referred to as "Quality Center" or "QC";
> HPE Application Lifecycle Management (ALM) refers to HPE Quality Center's later versions, the standard ALM edition version 11.00 and later, including versions 11.00, 11.20, 11.50, and 12.00 (hereinafter referred to as "HPE ALM" or "ALM").
>
> QC/ALM refers to HPE Quality Center/HPE Application Lifecycle Management .

Introduction to Integrating PPM with HPE Quality Center/HPE Application Lifecycle Management (QC/ALM)

Integrating PPM with Quality Center and/or ALM enables you to create requirements and defects in Quality Center and/or ALM while using PPM. This allows users of both applications to participate in the processing of defects, change requests, and release requests.

In PPM, a request type is a template, and when you create a request, you must select a request type. With the PPM–QC/ALM integration, creating the request in PPM automatically creates a defect or requirement in the integrated QC/ALM project. For example, depending on the integration configuration, creating a PPM request of type PPM_Defect could create a defect in QC/ALM project A or project B, and creating a PPM request of type PPM_Requirement could create a requirement in QC/ALM project A or project B.

In addition, for integration with ALM, when an ALM defect is created, an associated PPM request can be created automatically.

When an IT manager or business liaison enters a change request or creates a release comprising a group of change requests in PPM, PPM–QC/ALM integration causes a requirement to be created in Quality Center. This informs QA personnel that they should begin the QA process.

PPM–QC/ALM integration allows ongoing synchronization between fields such as status fields that have been mapped between a request type in PPM and an associated defect or requirement in a QC/ALM project.

Upon completion of the QA process, the IT manager or business liaison is notified on the PPM Dashboard and can complete the deployment process.

Using QC version 10.00, multiple PPM request types can be integrated with the same QC entity (either a defect or a requirement) in a project. Using ALM version 11.x, each integration must include a one-to-one, unique mapping between a PPM request type and an ALM entity (defect or requirement).

In any case, different PPM request types can be integrated with different entities of the same project or different projects in QC/ALM. If the request types are integrated with entities of different projects, the projects can be on the same or different QC/ALM servers, and each QC/ALM server can be at version 10.00 or version 11.x.

> **Note:** PPM version 9.40 can be integrated with QC 10.00 and ALM, but the integration configuration procedures are significantly different.

- For integration with QC version 10.00, PPM provides the PPM-Quality Center Integration Tool, which you can install on any computer. You use a software wizard to help create the required mapping files, which you then deploy to PPM and to Quality Center.

- For integration with ALM, you use a menu option in PPM to create the mapping files in a central PPM location rather than creating them elsewhere and deploying them to PPM or ALM.

> **Note:** Make sure you follow the appropriate integration procedures in this section, based on the installed version of QC or ALM. Do *not* attempt to use the PPM-Quality Center Integration Tool to configure integration with ALM.

For more information about the benefits of this integration, see "Integration of PPM with Quality Center/HPE ALM, Using ALM" on page 19 and "Benefits and Functionality of the Integration" on the next page.

For information about the QC/ALM versions supported for integration and, for integration with QC 10.00, the required version of .NET Framework for the machine on which the integration tool is run, see the *System Requirements and Compatibility Matrix* .

> **Note:** No software needs to be installed on the QC/ALM server to integrate PPM and QC/ALM. However, see the *System Requirements and Compatibility Matrix*.

For more information about QC/ALM, see its product documentation at the Web site described in "Available PPM Integrations " on page 18.

Benefits and Functionality of the Integration

Integrating PPM and QC/ALM provides the following benefits to users of each application:

- "Data Sharing " below

- "Process Integration" on the next page

- "Added Functionality of Integration with ALM" on the next page

- "Other Added Functionality to the Integration" on page 133

Data Sharing

The integration allows data sharing between PPM and QC/ALM. Business managers and IT personnel using PPM gain visibility into how a project is affected by both the quality control process and the parameters that are collected in QC/ALM. They can use QC/ALM capabilities when creating requests for changes.

- **Synchronization of mapped field values between PPM and QC/ALM.** When fields are mapped between a PPM request type and a QC/ALM defect or requirement, changing the value of a mapped field in one application can automatically change the value of the associated field in the other application. For example, when you change the status of a request related to defects to **Fixed** in PPM, the status of the associated QC/ALM project defect changes to **Fixed.**

  Users can specify that one of the following occurs for any pair of mapped fields:

  - Changing the value of the field in the QC/ALM defect or requirement automatically updates the value of the mapped field in the PPM request accordingly, that is, QC/ALM is dominant for (controls) the pair of mapped fields.

  - Changing the value of the field in PPM request automatically updates the value of the mapped field in the
    QC/ALM defect or requirement accordingly, that is, PPM is dominant for (controls) the pair of mapped fields.

  - Changing the value of the field in either application automatically updates the value of the mapped field in the other application accordingly, that is, the field mapping is bidirectional.

  Synchronization of defects enables the following:

○ Developers can use PPM to manage the defect-fixing process, while QA personnel continue to use QC/ALM.

○ Project managers and IT managers can view all the defects in the system, whether the defects originated in PPM or in QC/ALM. This helps the managers to decide on content for the next release or new requirements and enhancements.

○ QA personnel can use QC/ALM to manage defects created through PPM.

• Request hierarchy synchronization. The hierarchical structure of requirements in QC/ALM can be synchronized with the structure of the associated requests in PPM. That is, you can force the QC/ALM requirement hierarchy to match the PPM request hierarchy automatically.

• Synchronization with PPM notes. The integration allows you to synchronize a notes-related field in QC/ALM with notes in the associated PPM request. When you update the content of the notes in a PPM request, the associated notes field in QC/ALM is updated.

Process Integration

• **Inclusion of QC/ALM data in the workflow.** The PPM workflow is a well-defined process that allows IT managers to plan, track, and deploy software enhancements. Integration allows you to build your own workflows and steps in PPM while using fields and data from QC/ALM. The QC/ALM defect resolution capabilities become part of the workflow. This makes the quality process an integral and formal part of the IT processes.

• **QA-dependent workflow progression (for integration with QC 10.00 only).** Progress from one step in the PPM workflow to the next can be made dependent on progress by the QA team. In PPM, the IT manager can view how a project is affected by the quality defects that are collected in Quality Center and can decide whether a defect has been resolved or an enhancement can be deployed.

• **Direct activation of processes, and creation of QC/ALM requirements and defects from PPM.** Processes can be activated by PPM—creating a PPM request of a type that is integrated with QC/ALM creates a defect or requirement in QC/ALM when the relevant step in the PPM workflow is activated.

Added Functionality of Integration with ALM

Compared to integration with QC 10.00, integration with ALM provides a centralized, consolidated landing page for integrations in the PPM standard interface.

Note: For information about the View Project Quality integration solution, see "Integrating PPM Tasks with ALM Releases - View Project Quality" on page 249.

For information about the Agile Integration solution, see "Integrating PPM Tasks with HPE Agile Manager" on page 98.

You can perform all integration configuration tasks here, including:

- Add a new integration configuration easily.

  A drop-down list of eligible ALM request types and an **Add Integration Configuration** link are available at the bottom the **Integration Configurations** summary section. This allows you to add a new integration configuration easily.

- Map one ALM request type to multiple entities from different ALM domains or projects.

  For example, you can map one ALM request type to a defect in both project1 and project2 in ALM. When creating a request in PPM, you need to select the exact ALM project that you want the request to map to.

  Note: If a PPM request type is already mapped with ALM entity type defect (or requirement), then any new integration configurations for the request type shall also be mapped with the same ALM entity type of defect (or requirement).

- Configure an integration configuration easily, including enabling or disabling, deleting, or copying the integration configuration.

- Upgrade an existing integration configuration with QC 10 easily.

  Integration configurations with QC 10 are read-only ( ), with an Upgrade icon ( ) available at the right end of the entries. You can upgrade an integration configuration to integration with ALM version

11.00 or later. For more information, see "Upgrading from Integration with Quality Center 10.00 to Integration with ALM" on page 204.

○ Rationalized integration configuration status and action icons

- **Enabled** . The integration configuration is enabled and operable. Same as **Deployed** before PPM version 9.20.

- **Disabled** . The integration configuration disabled and not operable. However, you can enable it by clicking the icon at a later time as necessary. Same as **Undeployed** before PPM version 9.20.

- **Read-Only** . The integration configuration with QC is operable but not configurable.

- **Delete** . If an integration configuration is not used any more, you can click **Delete** to remove the integration configuration from PPM permanently. Same as **Disable** before PPM version 9.20.

- **View Log** . User can click this icon for an integration configuration to view its log information.

○ Enhanced password security by encryption for integration configurations

No password required when browsing the integration configurations. Password required for updating an integration configuration.

○ Easy integration data mapping with default field mapping lists. Configuration no longer requires explicit deployment of the field mapping file to ALM or to PPM.

Default field mapping lists are available for both ALM defect and requirement modules. The default field mapping lists can be used without any additional configuration. Users can also customize field mappings easily according to their business requirements.

○ New fields added for ALM requirement type support and optional synchronization to ALM requirement or defect

- ALM requirement type support

New field **QC/ALM Requirement Type** is added to the QC/ALM Info field group. Users can use this field to control which requirement type they want to create in ALM.

- Synchronization to ALM requirement or defect is optional.

For requirement or defect synchronization, the **Synchronize to QC/ALM Defect** field and **Synchronize to QC/ALM Requirement** field are introduced to control whether to synchronize a request to ALM.

> **Note:** For integration configurations upgraded from QC 10, to add the new fields to your existing request types, you can run the `kUpgradeIntegrationRequests.sh` script provided with PPM version 9.20 to update the existing request types.
>
> To do so,
>
>   i. Navigate to the `<PPM_Home>/bin` directory.
>
>   ii. Run the following command:
>
>   **`sh kUpgradeIntegrationRequests.sh`**
>
>   You should be able to see that the new fields are added. At the same time, you can still continue to use the existing data.

○ An integration tool (PPM Center–ALM Integration Tool) is provided to enable hidden fields in ALM for the integration and to deploy the workflow scripts in ALM

> **Note:** The PPM Center-ALM Integration Tool is applicable for integration with ALM version 11.50 or later.
>
> For integration with ALM version 11.00 or 11.20, or for an integration configuration upgraded from QC 10, you need to run the scripts provided by HPE Software Support manually:
>
> - For instructions on how to enable PPM* fields in ALM project, go to https://softwaresupport.hpe.com/group/softwaresupport/search-result/-/facetsearch/document/KM1352699.
>
> - For instructions on how to activate workflow script in ALM project, go to https://softwaresupport.hpe.com/group/softwaresupport/search-result/-/facetsearch/document/KM1352700.

○ Enhanced retry mechanism for failed synchronization.

When ALM server is unreachable due to the server being down or incorrect username or password provided, the automatic synchronization retry attempts are not subject to the standard retry limit until the ALM server is accessible.

Administrators need to add a **Retry** workflow step to enable this enhanced retry mechanism. An example:

- ○ Integration configurations can be exported or imported to simplify maintenance and configuration work. For more information, see "Importing or Exporting an Integration Configuration" on page 214.

- ○ Creating a defect in ALM can create a request in PPM.

- ○ Email notification options are enhanced—you can optionally have notifications sent as follows:

  - • When the integration creates or updates ALM defects as well as other entities.

  - • When integration errors occur.

  - • In a daily consolidation of notifications, as an alternative to individual notifications.

  The information in the notifications matches the information in the event and error log.

- ○ The integration supports both HTTP and HTTPS. (The integration with QC version 10.00 supports HTTP only.)

- ○ When a PPM request is created, the integration automatically populates the request fields related to ALM—the ALM server instance (URL), domain, and project—with the values PPM retains from initial configuration of the integration.

Other Added Functionality to the Integration

- • Synchronize QC/ALM text fields containing 4000 or more bytes to PPM text fields

When QC/ALM to PPM synchronization is performed, the content from QC/ALM text fields with text length greater than 4000 bytes (in UTF-8 encoding) is automatically truncated down to 4000 bytes when populating corresponding PPM text fields.

To prevent the original content in the QC/ALM text fields from being truncated in future synchronizations, set the text field mapping control to **QC/ALM**. This ensures that truncated content in the PPM text fields not to be synchronized back to QC/ALM text fields.

- Synchronize QC/ALM defect or requirement status to PPM request status. Status changes of QC/ALM defect or requirement can also trigger corresponding PPM request workflow steps or actions, thus changing status of the PPM requests.

- Synchronize multi-value fields (for example, drop-down list values, QC/ALM list field, and PPM auto-complete list) bi-directionally.

- Synchronize PPM usernames, instead of full names, to QC/ALM.

- Synchronize QC/ALM requirement types to PPM requests. Changing the requirement type field value in QC/ALM updates the requirement type field value in PPM requests.

- Automatically submit the PPM request created as a result of creating a defect in QC/ALM.

- For integration of PPM request type and QC/ALM defect, a third option for bi-directional integration is available in the **Integration Options** section on the Configure QC/ALM Integration for Request Type page: **Creating a PPM request automatically creates an associated QC/ALM entity and vice versa**.

- Special characters and HTML tags contained in QC/ALM fields can display properly on the Configure QC/ALM Integration for Request Type page, including, but not limited to, the follows:

  `~ ! @ # $ % ^ & * ( ) _ + { } | " : ? < > / . , ; ' \ ` [ ] - =`

  However, for security concern, HPE highly recommends you NOT use angle brackets and square brackets (`<>`, `[]`).

ALM Content Bundle Entities Used by the Integration

The ALM content bundle entities that are used by the integration with QC or ALM are described in the following sections.

ALM - Defect Template with Quality Center Integration Request Type

The PPM request type that ALM provides for integration with
QC/ALM project defects is the ALM - Defect Template with Quality Center Integration request type. This request type uses only the ALM - Defect Template with Quality Center Integration workflow. To prevent you from inadvertently using a workflow that is not enabled for this integration, you cannot choose a different workflow. This request type includes the QC/ALM Defect Information field group.

> **Note:** Although you can create new request types from scratch, HPE recommends that you use the provided ALM request type as a template to create new request types. In the PPM Workbench, you can copy the provided request type and modify the copy. If you create your own request types to integrate PPM with QC/ALM, make sure you use workflows that are enabled for integration.
>
> You can also create a new request type with a customized request header type.

Figure 6-1 shows the Create New ALM - Defect Template with Quality Center Integration page that appears when you create a request and select the ALM - Defect Template with Quality Center Integration request type. Table 6-1 describes the fields in the request, including some fields that do not appear until the request is submitted for creation or until other conditions are met.

The fields in the QC/ALM Defect Information section of the request are defined by the QC/ALM Defect Information field group in the request header type for the request. For more information about these fields, see "QC/ALM Defect Information Field Group" on page 140.

Figure 6-1. ALM - Defect Template with Quality Center Integration request

Table 6-1. ALM - Defect Template with Quality Center Integration request fields

Field Name (*Required)	Description
Summary section	
Request No.	(Added after the request is created) Number of the request

Table 6-1. ALM - Defect Template with Quality Center Integration request fields, continued

Field Name (*Required)	Description
Created On	(Added after the request is created) Date the request was created
*Summary	Summary description of the request
Department	Department to which the user belongs
Created By	User who created the request
*Severity	Severity of the defect
Assigned To	Developer assigned to work on the defect
Detected in Version	Version of the application in which the defect was detected
Assigned Group	Group responsible for addressing the defect
Defect Priority	Priority of the defect
Application	Application in which the defect was discovered
Reproducible	Option to indicate whether the defect is reproducible
Request Status	Status of the request
QC/ALM Defect Information section [a]	
QC/ALM Instance	URL of the QC/ALM instance with the project used for the integration
QC/ALM Domain	Domain of the project in QC/ALM
QC/ALM Project	QC/ALM project that is integrated with this request type
Synchronize to QC/ALM Defect	Select Yes or No radio button to indicate whether synchronize to QC/ALM defect or not.
QC/ALM Defect Number	(Added after the defect is created in QC/ALM) Defect number in QC/ALM
QC/ALM Defect Status	(Added after the defect is created in QC/ALM) Status of the defect in QC/ALM
QC/ALM Integration Message	(Read-only. Added after the defect is created in QC/ALM.) QC/ALM status message indicating success or error in the most recent operation
QC/ALM Assigned To User	Assigned owner of the defect in QC/ALM
QC/ALM Attachment URL	URL of the list of attachments to the QC/ALM requirement or defect
Detected in QC/ALM by	User who detected the defect in QC/ALM
Planned vs. Actual section	

Table 6-1. ALM - Defect Template with Quality Center Integration request fields, continued

Field Name (*Required)	Description
Planned Closing Version	Version of the application targeted to have the defect fix
Closed in Version	Version of the application that has the defect fix
Estimated Fix Time (days)	Original estimate of the number of days it would take to fix the defect
Actual Fix Time (days)	Actual number of days it took to fix the defect
Closed on	Date the defect was closed in QC/ALM
Defect Information section	
Detailed Description	Detailed description of the defect
Developer Comments	Developer comments regarding the defect
a. Fields in the QC/ALM Defect Information section remain visible by default but are not used if PPM is not integrated with QC or ALM.	

Note: The administrator can remove the **QC/ALM Defect Information** section from the request type by removing the QC/ALM Defect Information field group from the ALM - Defect Template with Quality Center Integration request header type.

For details about request header and field groups, see the *Demand Management Configuration Guide*.

How to submit an ALM - Defect Template with Quality Center Integration request

1. Log on to PPM.

2. From the menu bar, select **Create > Request**.

   The Create New Request page appears.

3. On the Create New Request page, in the **Request Type** field, select **ALM - Defect Template with Quality Center Integration** and click **Create.**

   The Create New ALM - Defect Template with Quality Center Integration page appears, displaying the appropriate request fields.

   Required fields have a red asterisk. All other fields are optional, but are often helpful when others are reviewing an open request. For information concerning a specific field, click the Help icon next to the field (if available).

4. Complete the fields in all sections as appropriate.

   The **Notes** section contains fields where notes and information concerning the request can be

entered and stored. Typically, when you create a request, you do not need to add a note to it. However, add a note if you want to convey additional information to the reviewers and processors of the request.

In the **References** section of the request, you can add useful references such as a Web-accessible file or a document or file attached from a local machine. For more information about adding references, see the *Demand Management User's Guide*.

5. On the Create New Request page, click **Submit.**

The request is submitted. The Request Creation Confirmed page appears.

> **Note:** PPM can be configured to allow you to save the request before you submit it. To have this feature enabled, see your PPM administrator.

After submitting the request, on the Request Creation Confirmed page you can click the link for the particular request number in the **Request #** field to view the detail page of the newly generated request.

When the request is submitted, it is assigned an initial status, such as New. The request is then routed along the ALM - Defect Template with Quality Center Integration workflow (see "ALM - Defect Template with Quality Center Integration Workflow" on page 141).

Request Header Types

By default, when you create a new request type from a provided one, the new request type uses the same request header type as the provided request type uses.

When creating a new request type, you can do the following:

- Use the supplied request header type as is.
- Copy the request header type, customize the copy, and use the customized copy in the new request type.
- Create a completely new request header type.

> **Note:** Your request header type must include the QC/ALM fields that appear in the provided request header type. If you customize a request header type, make sure you do not delete the QC/ALM fields.

If you create a new request header type, add the required integration-related fields by selecting the appropriate field group, as follows:

1. Log on to PPM.

2. From the menu bar, select **Open > Administration > Open Workbench.**

   The PPM Workbench opens.

3. From the shortcut bar, select **Demand Mgmt > Request Header Types**.

   The Request Header Type Workbench opens.

4. In the Request Header Type Workbench, click **New Request Header Type**.

   The Request Header Type window opens.

5. Click **Field Groups**.

   The Field Groups window lists the available field groups.

6. Select the appropriate field group:

   ○ For a defect, select the **QC/ALM Defect Information** field group.

   ○ For a requirement, select the **QC/ALM Info** field group.

QC/ALM Defect Information Field Group

The fields in the **QC/ALM Defect Information** section of the ALM - Defect Template with Quality Center Integration request type (see Table 6-1) are defined by the QC/ALM Defect Information field group. These fields support integration with QC/ALM defects and should not be modified (except for their Field Prompts, as desired). The table below provides more information about these fields.

Table 6-2. Fields in QC/ALM Defect Information field group

Field Name	Field Database ID	Field Type
QC/ALM Instance	KNTA_QC_DEFECT_INSTANCE	Autocomplete List
QC/ALM Domain	KNTA_QC_DEFECT_DOMAIN	Autocomplete List
QC/ALM Project	KNTA_QC_DEFECT_PROJECT	Autocomplete List
Synchronize to QC/ALM Defect	SYNC_TO_QC_DEFECT	Radio Buttons (Yes / No)
QC/ALM Defect Number	KNTA_QC_DEFECT_NO	Numeric Text (10 digits)
QC/ALM Defect Status	KNTA_QC_DEFECT_STATUS	Text Field (300)

Table 6-2. Fields in QC/ALM Defect Information field group, continued

Field Name	Field Database ID	Field Type
QC/ALM Integration Message	KNTA_QC_DEFECT_INT_MSG	Text Field (300)
QC/ALM Attachment URL	KNTA_QC_DEFECT_ATT_URL	Link
QC/ALM Assigned To User	KNTA_QC_DEFECT_ASSIGNED_ TO	Autocomplete List

ALM - Defect Template with Quality Center Integration Workflow

The PPM workflow that ALM provides for integration with QC/ALM project defects is the ALM - Defect Template with Quality Center Integration workflow, which includes execution steps to create a defect or requirement in QC/ALM. You can use this workflow as a template for creating your own workflows.

Although you can create new workflows from scratch, HPE recommends that you create them from this template. To create a new workflow, use the PPM Workbench to create a copy of a provided workflow, and then modify the copy (add, delete, or change its steps) to suit your needs.

The integration also uses the ALM - Release Request workflow to create a release entity that includes several requests for change. This workflow can be used for ITIL purposes whether or not PPM is integrated with other applications. For more information, see "ALM - Release Request Workflow" on page 81.

The ALM - Defect Template with Quality Center Integration workflow is used to create a defect and to track how the defect is resolved. See the following figure and workflow step 5, Defect Verification.

Figure 6-2. ALM - Defect Template with Quality Center Integration workflow
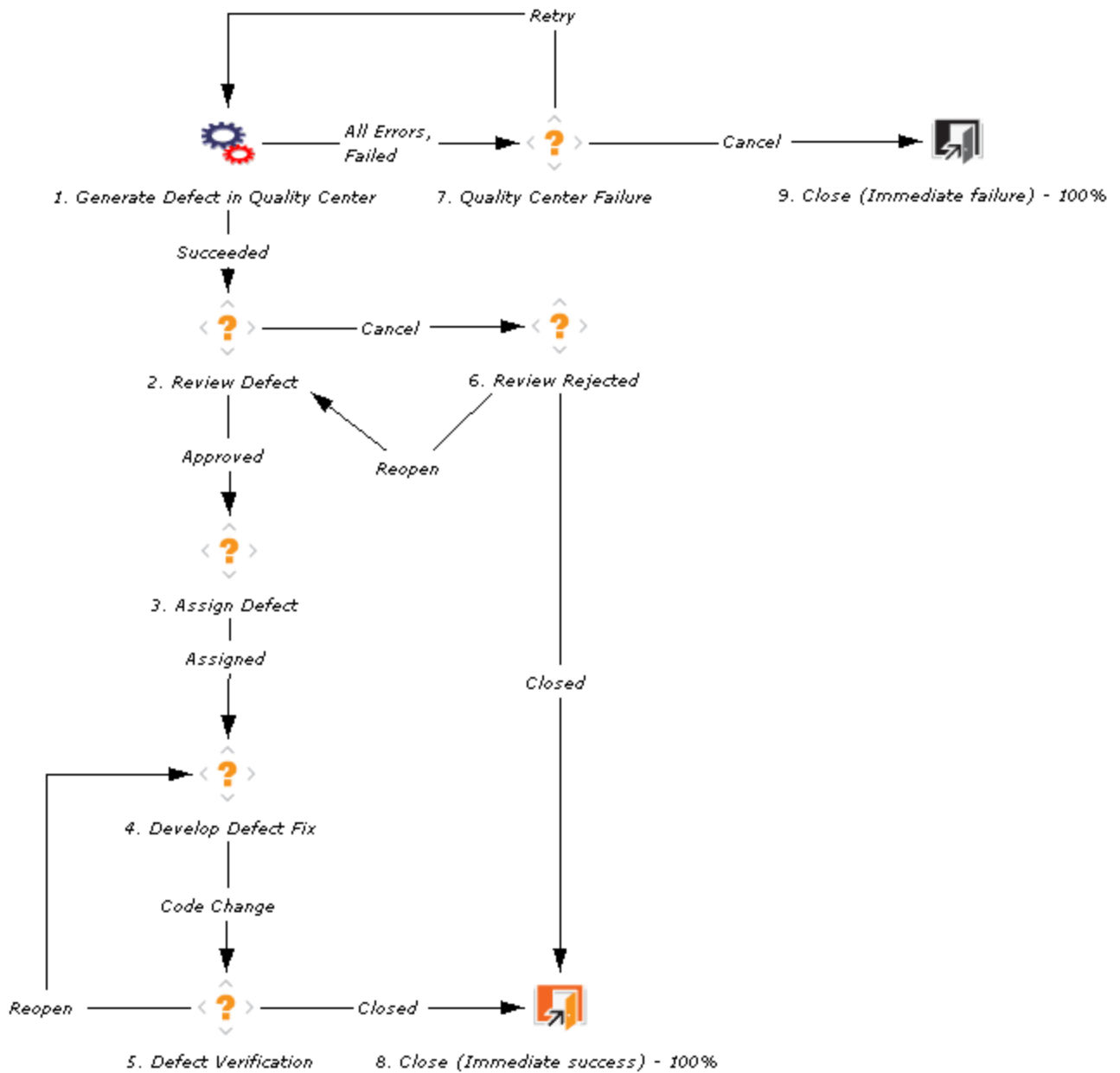
When a request of type ALM - Defect Template with Quality Center Integration is created in PPM, the first step in the workflow creates a defect in QC/ALM. Subsequent steps cause the defect status in QC/ALM to change to Open, Reopen, Fixed or Closed, depending on the stage in the PPM workflow.

Types of Workflow Steps

As with any PPM workflow, the ALM - Defect Template with Quality Center Integration workflow can contain the following types of steps:

- **Decision steps.** Steps that require action from the user in order to proceed. When the user reaches the decision step, the user sees a set of choices. Each choice causes the workflow to proceed in a different manner. For example, at one decision step in a workflow, a project manager might be offered the choice of either deploying a package or sending it back to QA for more testing.

- **Condition steps.** Steps that determine the direction that the workflow takes.

- **Execution steps.** Steps that are automated through PPM. For example, an execution step might create a requirement or defect in QC/ALM, execute a script, or run a build.

ALM provides two execution steps that can be used in PPM to build a workflow for integration of PPM with QC/ALM. The execution steps create a defect or a requirement in QC/ALM.

ALM Request Types Used for Integration with QC/ALM Requirements

Integration with QC/ALM project requirements can use the following request types and workflows:

- ALM - Request for Change (RFC) request type for requests for change, which are used in multiple HPE products (see "ALM - Request for Change (RFC) Request Type" on page 33).

  This request type is used in conjunction with the ALM - Request for Change workflow (see "ALM - Request For Change Workflow" on page 43).

- ALM - Release Management request type for requirements (see "ALM - Release Management Request Type" on page 73).

  This request type is used in conjunction with the ALM - Release Request workflow (see "ALM - Release Request Workflow" on page 81).

These request types include the QC/ALM Info field group (see "QC/ALM Info Field Group" on the next page), and their associated workflows can generate requirements in QC/ALM.

Note: Although you can create new request types from scratch, HPE recommends that you use the provided ALM request type as a template to create new request types. In the PPM Workbench, you can copy the provided request type and modify the copy. If you create your own request types to integrate PPM and QC/ALM, make sure you use workflows that are enabled for integration.

You can also create a new request type with a customized request header type.

QC/ALM Info Field Group

The fields in the **QC/ALM Info** section of the ALM - Request for Change (RFC) request type or the ALM - Release Management request type are defined by the QC/ALM Info field group. These fields support integration with QC/ALM requirements and should not be modified (except for their Field Prompts, as desired). Table 6-3 provides more information about these fields.

Table 6-3. Fields in QC/ALM Info field group

PPM Field Name	PPM Field Database ID	PPM Field Type	Description
QC/ALM Instance	KNTA_QC_ INSTANCE	Autocomplete List	URL of the QC/ALM instance with the project that will integrate with the PPM request.
QC/ALM Domain	KNTA_QC_ DOMAIN	Autocomplete List	Domain of the working QC/ALM project.
QC/ALM Project	KNTA_QC_ PROJECT	Autocomplete List	QC/ALM project that is integrated with this request type.
Assigned To	KNTA_QC_ ASSIGNED_TO	Autocomplete List	Developer the QC/ALM Requirement is assigned to.
Requirement No.	KNTA_QC_ REQUIREMENT_ NO	Numeric Text (10 digits)	Requirement number in QC/ALM.
Requirement Status	KNTA_QC_ REQUIREMENT_ STATUS	Text Field (300)	Status of the requirement in QC/ALM.
Requirement Integration Message	KNTA_QC_ REQUIREMENT_ INT_MSG	Text Field (300)	QC/ALM status message indicating success or error in the most recent operation.
Synchronize to QC/ALM Requirement	SYNC_TO_QC_ RFC	Radio Buttons (Yes / No)	Select Yes or No radio button to indicate whether to synchronize to QC/ALM requirement or not. New field introduced in PPM version 9.20.
Requirement Attachment URL	KNTA_QC_ REQUIREMENT_ ATT_URL	URL	URL of the list of attachments to the QC/ALM requirement.
QC/ALM Requirement Type	QC_ REQUIREMENT_ TYPE	Autocomplete List	Quality Center/ALM requirement type. New field introduced in PPM version 9.20.

Table 6-3. Fields in QC/ALM Info field group, continued

PPM Field Name	PPM Field Database ID	PPM Field Type	Description
QC/ALM Dashboard Subject	KNTA_QC_ DASHBOARD_ SUBJECT	Text Field (200)	QC/ALM dashboard subject name (not in use)
QC/ALM Requirements Coverage	KNTA_QC_ REQUIREMENT_ COVERAGE	Numeric Text (10 decimals)	QC/ALM requirements coverage (not in use)
QC/ALM Open Defects	KNTA_QC_ OPEN_DEFECTS	Numeric Text (10 decimals)	Number of QC/ALM open defects (not in use)

Selecting the Appropriate Integration Procedure

If multiple QC or ALM servers (instances) are integrated with PPM, some of the QC or ALM instances can be at QC 10.00 and others can be at ALM 11.x.

Caution: The PPM-Quality Center Integration Tool must be used only for integration with QC version 10.00. It cannot detect integrations that were previously established with ALM, and any existing integration with an ALM instance prevails if you try to use the integration tool for that integration.

Caution: If you upgrade your PPM from an earlier version to version 9.40,

- And your QC instance is still at version 10.00, if you have no plan to upgrade your QC instance, you can continue to use existing integration of PPM requests and QC entities (defects or requirements) on that instance you have established previously.

- And you upgrade your QC instance from version 10.00 to ALM version 11.x or later, existing integrations of PPM requests and QC entities (defects or requirements) on that instance are not upgraded. They show as read-only entries on the new Integration Configurations landing page (with an **Upgrade** icon available for each entry). You need to upgrade the exiting integration configurations one by one so that they will continue to work properly with all the existing data.

HPE does NOT recommend you re-establish the integrations after you have upgraded both PPM Server and QC server to the latest versions. If you re-establish the integrations, they will operate as new integrations. You risk corrupting the existing data or data loss on both servers.

Note: In configuring an ALM integration, make sure you establish a one-to-one, unique mapping

between each integrated PPM request type and its associated ALM entity (defect or requirement).

The procedure to configure integration of PPM with ALM is significantly different from the procedure to configure integration with QC 10.00. You must use only the appropriate procedure, as follows:

- For integration with QC version 10.00, go to "Configuring Integration with Quality Center Version 10.00" below.

- For integration with ALM, go to "Configuring Integration with ALM Version 11.00 and Later" on page 172.

Configuring Integration with Quality Center Version 10.00

The procedures in this section apply to integrating PPM with Quality Center version 10.00 only. If you are integrating PPM with ALM, go to "Selecting the Appropriate Integration Procedure" on the previous page.

Integration with Quality Center version 10.00 requires installing the PPM-Quality Center Integration Tool on any Windows machine that can open HTTP connections to the PPM Server and to the Quality Center server. This tool enables Quality Center projects for integration and maps PPM fields to Quality Center fields.

> **Caution:** The PPM-Quality Center Integration Tool is used only for integration with Quality Center version 10.00. Do *not* attempt to use the procedures in this section to configure integration with ALM.

Before proceeding, verify that QC version 10.00 is installed and running on the Quality Center server to be integrated.

Overview of Installation and Configuration Process

The procedures for configuring integration of PPM and QC version 10.00 are described in detail in the following sections, and are summarized as follows:

- If an earlier version of the PPM-Quality Center Integration Tool was installed, uninstall it.

- Install the latest version of the PPM-Quality Center Integration Tool (see the *System Requirements*

and Compatibility Matrix). This tool enables a Quality Center project for integration and maps PPM fields to Quality Center fields in an XML mapping file.

- Using the PPM-Quality Center Integration Tool, configure integration with a Quality Center project as follows:

  - Enable a Quality Center project for integration.

  - If you are integrating a PPM request type with a Quality Center requirement, specify the following:

    - Whether email notifications are to be sent when requirements are created and/or updated

    - Whether the requirement hierarchy is to be synchronized with (driven by) the request hierarchy

  - Create a mapping between PPM fields and Quality Center fields, including their value lists.

  - Map the Notes field in PPM to an existing Quality Center project.

  - Deploy the mapping file to PPM and Quality Center.

  Note: If you want to integrate PPM requests with both existing and new defects in a Quality Center project, create a new Quality Center project with integration enabled and copy the existing defects (or the desired subset) from the existing project to the new project.

- Configure PPM for the integration, including specifying server configuration parameters.

Changes to Quality Center Value Lists and Workflows Made by the Integration Tool

When the user enables a Quality Center project for integration using the PPM-Quality Center Integration Tool, the tool performs one of the following actions:

- If the project is new, the tool creates Quality Center value lists and adds Quality Center workflow scripts to the user's existing scripts.

- If the project already exists, the tool updates some of the Quality Center value lists so that those lists contain the same values as they would for a new project.

The lists and Quality Center scripts create a Quality Center project that can work as is with the provided ALM request types and workflows.

As described in the following sections, the integration tool changes the value lists and workflow enforcement in a Quality Center project to enable the project for integration.

Changes to Value Lists

As part of enabling a Quality Center project for integration and establishing a field mapping, the integration tool adds two new value lists and adds a new value to an existing default value list in Quality Center, as follows:

- New Requirement Status value list, with the following values:

  - New

  - Cancelled

  - Closed

  - 1-Requirements Setup Completed

  - 2-Test Plan Setup Completed

  - 3-Test Lab Setup Completed

  - 4-Running Tests in Quality Center

  - 5-Test Execution Completed

  - 6-Running Sanity Tests in Quality Center

  - 7-Sanity Testing Completed

- New Test Level value list, with the following values:

  - Functional

  - Integration

  - Regression

  - Sanity

- New default value of Deleted for the existing Bug Status value list

Workflow Enforcement

As part of enabling a Quality Center project for integration and establishing a field mapping, the integration tool updates the Quality Center project workflow to enforce the following constraints on Quality Center entities:

- **Defect.** The user can make only the following status changes:

  - Fixed to Closed

  - Rejected to Closed

  - Fixed to Reopen

- **Requirement.** The user can make only the following status changes:

  - New to 1-Requirements Setup Completed

  - 1-Requirements Setup Completed to 2-Test Plan Setup Completed

  - 2-Test Plan Setup Completed to 3-Test Lab Setup Completed

  - 4-Running Tests in Quality Center to 5-Test Execution Completed

  - 6-Running Sanity Tests in Quality Center to 7-Sanity Testing Completed

Installing the PPM-Quality Center Integration Tool

Integration requires installing the PPM-Quality Center Integration Tool on a Windows machine that can open HTTP connections to the PPM Server and to the Quality Center server.

To install this tool:

1. If an earlier version of the PPM-Quality Center Integration Tool was installed, uninstall it. See "Uninstalling the Integration Tool" on the next page.

2. Copy the PPM-Quality Center Integration Tool `setup.exe` file to a Windows machine that can open HTTP connections to the PPM Server and to the Quality Center server. This file is located in the PPM Server at:

   <PPM_Home>`/integration/mac/ppmqcintegrationtool`

   where *<PPM_Home>* represents the path where your PPM instance is installed. For example: `xyzserver/E/PPMServer`.

3. Double-click the `setup.exe` file to launch the InstallShield wizard.

   By default, the tool gets installed in the `C:/Program Files/Hewlett-Packard/PPM Center-Quality Center Integration Tool` directory.

4. Follow the instructions in the wizard. When the installation completes, do not launch the PPM-

Quality Center Integration Tool yet. Proceed to configure a Quality Center project for the integration. For more information, see "Configuring a Quality Center Project for the Integration" below.

Uninstalling the Integration Tool

If you want to uninstall the PPM-Quality Center Integration Tool (for example, to be able to run the tool from another Windows machine only), select **Start > Settings > Control Panel > Add/Remove Programs** and follow the instructions on the screen.

You use the tool to create an initial mapping between PPM fields and Quality Center fields (or to revise an existing mapping). Unless you need to install a later version of the tool, before you uninstall the tool, remember to use it to deploy the mapping to both PPM and Quality Center simultaneously.

Enabling Web Services

> **Note:** For more information about the steps in this procedure, see the *Installation and Administration Guide*.

Verify that Web services in PPM are enabled, as follows:

1. Stop the PPM Server.

2. Run the script `sh ./kConfig.sh`.

3. Verify that the ENABLE_WEB_SERVICES parameter in the PPM server.conf configuration file is set to true.

4. Restart the PPM Server.

> **Note:** If PPM uses a cluster configuration, repeat this procedure to enable Web services on all nodes of the cluster.

Configuring a Quality Center Project for the Integration

As documented in the following sections, you establish integration of PPM with Quality Center independently for each Quality Center project as needed, using wizards in the PPM-Quality Center Integration Tool to do the following:

- Enable a Quality Center project for integration

- Create the XML mapping file between PPM and Quality Center fields

- Deploy this mapping file to PPM and Quality Center

You might also need to manually map the **Notes** field in a PPM request to the Quality Center project.

Enabling a Quality Center Project for the Integration

Use the PPM-Quality Center Integration Tool to enable a Quality Center project for the integration as described in this section.

> **Note:** To enable a Quality Center project for integration: HPE strongly recommends that you enable a project only once, otherwise problems might occur in the integration. There is no reason to enable a project twice. You can use the integration tool to change the integration of a project at any time.

To enable a Quality Center project for integration:

1. From the Windows Start menu, select **All Programs > Hewlett-Packard Enterprise > PPM-Quality Center Integration Tool**.

   The main PPM-Quality Center Integration Tool window opens.

2. Select **File > PPM Settings**.

   The PPM Settings window opens.

3. In the **Host** field, type the URL of the PPM Server.

   > **Note:** If the PPM Server is installed in a WAN, use the IP address for the PPM Server, for example `http://192.60.28.01:8080,` rather than its host name, for example `http://ppmhost:8080`.

4. In the **Administrator User Name** and **Administrator Password** fields, type the PPM user name and password.

5. Click **Test Connection** to test the connection with PPM.

6. If a message appears stating that connection was successful, click **OK** to close the PPM Settings window, otherwise resolve the connection issue.

7. In the main PPM-Quality Center Integration Tool window, select **Action > Enable Project.**

The Enable Quality Center Project wizard opens.

8. Click **Next** to continue.

9. In the **Host** field, type the URL of the Quality Center server.

10. Click **Next** to continue.

11. In the **User Name** and **Password** fields, specify the user name and password of the Quality Center project administrator.

12. Click **Test Connection** to test the connection with Quality Center.

13. If a message appears stating that connection was successful, click **OK** on the message, then click **Next.** Otherwise resolve the connection issue.

14. Select a Quality Center domain and project, and specify whether you want to map Quality Center defects, requirements, or both.

    If you are enabling defects but not requirements for integration, skip to step 17.

15. Click **Next** to continue.

    If you are enabling requirements for integration, the following window appears.

    You can use this window to do the following:

    ○ Request notification by email when a PPM request creates a new Quality Center requirement.

    ○ Request notification by email when a PPM request updates a field in an existing Quality Center requirement.

    > **Note:** To send any notifications, Quality Center must be configured to automatically send them, using email addresses established in Quality Center. See "Available PPM Integrations " on page 18 for information about accessing the Quality Center documentation.

    ○ Synchronize the PPM request hierarchy with the Quality Center requirement hierarchy. For information about this synchronization, see "Request Hierarchy Synchronization " on page 221.

16. Select the desired check boxes.

17. Click **Next** to continue.

    The wizard displays user-defined fields related to the PPM request that the integration tool will enable in the Quality Center project, for defects, requirements, or both, as specified in step 14.

> **Note:** Prior to version 7.0, PPM was known as Mercury IT Governance Center or ITG. Field names in Quality Center version 10.00 and in the integration tool in the context of Quality Center still use ITG when referring to PPM.

18. Click **Next** to continue.

    The listed fields are added to the Quality Center project, and the Quality Center workflow script is updated to support integration with PPM.

    > **Note:** If the Quality Center project already contains one or more fields with the same names, a dialog box asks you whether you want to change the names of the added fields. If you select **Yes,** the added fields will be given different names. If you select **No,** the procedure to enable the project for integration is aborted.

    > **Note:** If the Quality Center project does not have a workflow script in its repository, a dialog box asks you whether you want to create a new one. If you select **Yes,** a new script with support for the integration is added to the Quality Center repository. If you select **No,** the script file is not created.
    >
    > If the project is not new and its script files have been previously customized, select **No** and manually merge the customized scripts with the scripts that the wizard adds automatically, which are located in the installation directory of the PPM-Quality Center Integration Tool.

19. Click **Finish** to complete enabling the project for integration and close the wizard.

Creating the Mapping Between PPM and Quality Center Fields

After you enable a Quality Center project for the integration, you use the PPM-Quality Center Integration Tool to map a particular PPM request type to a Quality Center defect or requirement and to specify the desired mapping between the PPM fields and the Quality Center fields.

> **Note:** For information about the provided default mappings, see "Default Field Mappings for PPM and Quality Center Version 10.00 " on page 166.

To create the mapping:

1. In the left pane of the PPM-Quality Center Integration Tool, expand the tree until the child node at the lowest level (**defect**, **requirement**, or both) is visible.

2. Click the child node you want to map and select **Action > Add Mapping** (or right-click the node you want to map and click **Add Mapping**).

The Map Fields wizard opens.

3. Click **Next** to continue.

4. In the **User Name** and **Password** fields, type the user name and password of the Quality Center administrator.

5. Click **Next** to continue.

   The **PPM Request Type** field appears with a drop-down list.

6. Click the arrow of the drop-down list to display a list of all the PPM request types that can be mapped to a Quality Center defect or requirement (depending on which you selected in step 2).

   For information about the default PPM request types that are available to map to Quality Center defects and requirements, see "Using the Integration of PPM with Quality Center/ALM" on page 215.

7. Select the request type in PPM that is to be mapped to the Quality Center defect or requirement.

8. Click **Next** to continue.

   From the integration tool, the wizard displays the default mapping between the Quality Center fields and PPM fields for the request type you selected. In this Map Fields window, the fields map either a PPM request type for defects to a Quality Center defect, or a PPM request type for changes to a Quality Center requirement.

   The **Quality Center Field** column displays Quality Center field names, followed by their DB field names (the names of the columns in the database) in parentheses.

   The **PPM Field** column displays the PPM request field names, followed by their tokens in parentheses.

   The **Override** column specifies which field, if any, is the "dominant" field for a pair. The **Override** column can have one of the following values for any pair of mapped fields (row):

   ○ **QC.** In this case, Quality Center is said to be dominant for the mapped pair. When the Quality Center entity is created or modified in any way, then the integration updates all the mapped PPM fields for which Quality Center is dominant.

      When Quality Center is dominant for a mapped pair and the value in the PPM field is changed, the value in the associated Quality Center field is not affected.

      When the integration creates a Quality Center entity, the PPM request fields have no effect on the fields for which Quality Center is dominant.

   ○ **PPM.** In this case, PPM is said to be dominant for the mapped pair. When the PPM request is created or modified in any way, then the integration updates all the mapped Quality Center

fields for which PPM is dominant.

When PPM is dominant for a mapped pair and the value in the Quality Center field is changed, the value in the associated PPM field is not affected.

When the integration creates a PPM request, the Quality Center entity fields have no effect on the fields for which PPM is dominant.

○ **BIDIRECTIONAL.** In this case, both the PPM fields and their mapped Quality Center fields operate as though they are dominant—when either the integrated PPM request or Quality Center entity is created or modified, the integration updates all the associated fields in the Quality Center entity or PPM request respectively, as specified by the mappings.

9. If you want to change which field, if any, is dominant for a field mapping, click the value in the **Override** column for the field mapping of interest, and select another value from the list that appears.

10. If you want to add a pair of fields to the mapping, do the following:

   a. Click **Add.**

      The Add Field window opens.

   b. In **Quality Center Field** and **PPM Field**, select the fields you want to map to each other.

   c. Click **OK**.

      The pair of fields is added to the mapping and appears in the list.

   > **Note:** If the mapped fields in a pair have different sets of valid values, you must resolve the differences so that a change to one field can update the other. See "Resolving Lists of Valid Values" on the next page.

11. If you want to remove a pair of fields from the mapping, do the following:

   a. Click a line to select the pair.

   b. Click **Remove.**

   c. When the application asks whether you want to delete the selected line or lines, click **Yes.**

12. Click **Next** to continue.

13. Click **Finish** to save the mapping and close the wizard.

   The new mapping is displayed in the PPM-Quality Center Integration Tool. (In the example, expand the **defect** list.)

14. By default, a local copy of the mapping file (ITGQCIntegration.xml) is saved in the Windows

directory in which you installed the PPM-Quality Center Integration Tool. As discussed later, you will use the tool to deploy the mapping file to the PPM Server and the Quality Center server. Then the integration can operate without any dependency on the local Windows machine on which the integration tool is installed.

If you want to save the XML mapping file elsewhere, select **File > Save To** and specify the location.

If you want to open an XML mapping file stored in another location, click **File > Open.**

15. Click **File > Exit** to close the PPM-Quality Center Integration Tool.

Resolving Lists of Valid Values

This section describes how to resolve differences between the sets of valid values for a pair of mapped fields, so that a change to one field correctly updates the other.

To open the mapping file and access the Map Value Lists window:

1. Open the mapping file as described in "Viewing and Changing a Mapping" on page 164.

2. Select the row for the pair of fields of interest and click **Map Value Lists.**

   > **Note:** Some field pairs are automatically mapped and their mappings cannot be changed.

   The Map Value Lists window opens, displaying the lists of valid values for each field in the pair.

Resolve the differences between the sets of valid values based on which of the following three cases applies.

Case One

If you created a new user-defined field in Quality Center and mapped the field to a PPM field that has a list of values, create a new Quality Center list of valid values from the PPM list, as follows:

1. Select the **Create a new list in Quality Center...** option in the Map Values List window.

2. In the **List Name** field, specify a name for the list or accept the default.

3. Click **OK.**

   A new Quality Center value list containing the PPM values is created and associated with the Quality Center field.

Case Two

If the fields in the pair have different sets of values and some records in the Quality Center project already use the current list of values, but from now on you want to use the values that appear in the PPM list, add the PPM values to the Quality Center list, as follows:

1. Select the **Add the PPM Center values to the existing Quality Center list** option in the Map Values List window.

2. Click **OK.**

   The Quality Center value list now includes the PPM values as well as the original Quality Center values.

Case Three

If both fields in the pair already have lists of values that you need to map or remap, do the following:

1. Select the **Map Values** option in the Map Values List window.

2. Select a value in the Quality Center list, select the value in the PPM list to which you want to map it, and click the right arrow button.

   The pair of mapped values appears in the right pane of the Map Value Lists window. For example, if the value representing lowest impact in the Quality Center field is Enhancement and the value representing lowest impact in the PPM field is 1-Low, map the two values as shown in the following figure.
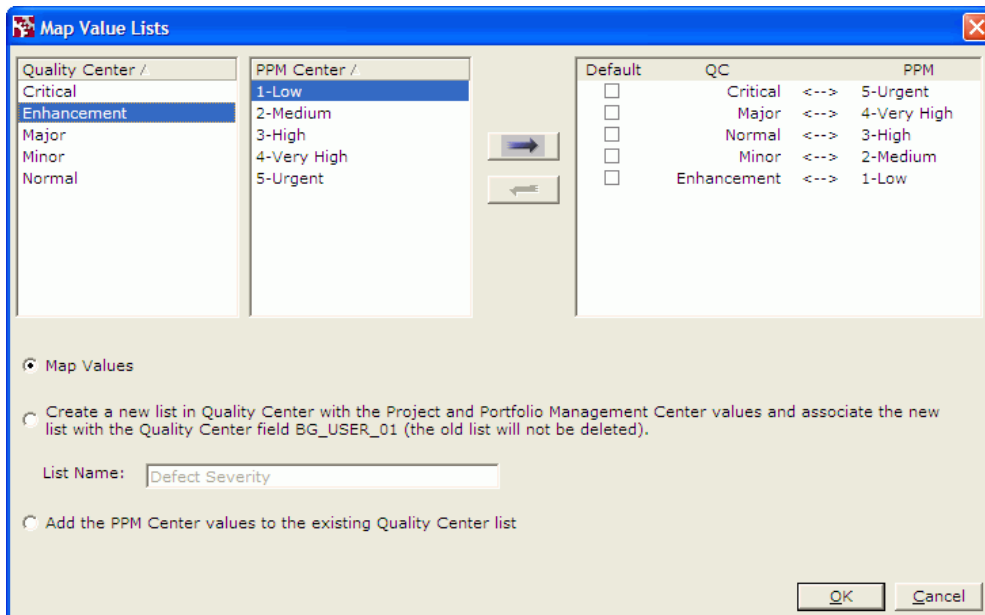
Between the **QC** and **PPM** columns in the right pane, the mapping displays one of the following:

- **-->** if the **Override** column for the pair of fields is set to **QC**

- **<--** if the **Override** column for the pair of fields is set to **PPM**

- **<-->** if the **Override** column for the pair of fields is set to **BIDIRECTIONAL**

3. Repeat for all the values that require mapping.

   - You must map all the values in the Quality Center list if Quality Center is dominant for that field pair, as indicated by **QC** in the **Override** column in the Map Fields window and by **-->** between the **QC** and **PPM** columns in the right pane of the Map Value Lists window.

   - Similarly, you must map all the values in the PPM list if PPM is dominant for that field pair, as indicated by **PPM** in the **Override** column in the Map Fields window and by **<--** between the **QC** and **PPM** columns in the right pane of the Map Value Lists window.

   - You must map all the values in both lists if neither field of the pair is dominant, as indicated by **BIDIRECTIONAL** in the **Override** column in the Map Fields window and by **<-->** between the **QC** and **PPM** columns in the right pane of the Map Value Lists window.

     If you map two or more values in one list to one value in the other list, you must select a check box in the **Default** column to indicate which mapping prevails.

     Consider the example in the following figure, where the two value lists are of different lengths and the mapping is **BIDIRECTIONAL.**
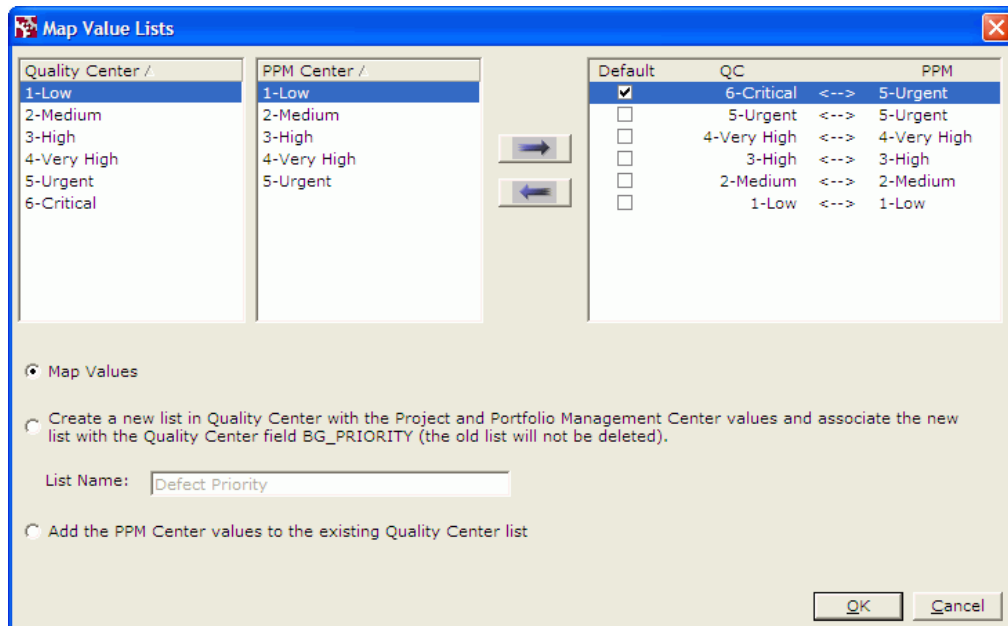
In this example, you must do the following:

  i. Map two or more values in one list to a single value in the other list. Here, both the values 6-Critical and 5-Urgent for the field in Quality Center have been mapped to a value of 5-Urgent for the field in PPM.

  ii. Select the appropriate **Default** check box to eliminate ambiguity as to which pair will be used to map the values. In this example, if the field in PPM changes to a value of **5-Urgent,** the value of the field in Quality Center becomes **6-Critical,** based on the selected **Default** pair of values. If the second check box, for which the PPM value is also **5-Urgent,** is chosen as the default instead, then if the field in PPM changes to a value of **5-Urgent,** the value of the field in Quality Center becomes **5-Urgent.**

 4. Click **OK.**

   The PPM values and Quality Center values for the field become mapped as you have specified.

Mapping the Notes Field in PPM to an Existing Project

When you enable a new Quality Center project for integration, the integration tool attempts to create new fields in the Quality Center project to correspond to the **Notes to be added on save** field for the PPM request. When you update the content of this field in a PPM request, the associated field is updated in Quality Center.

When you enable a new Quality Center project, the new fields are named as follows:

- For a defect, the added field in the Defects table is called **BG_USER_<XX>**, for example **BG_USER_25**.

- For a requirement, the added field in the Requirements table is called **RQ_USER_<XX>**, for example **RQ_USER_26**.

> **Note:** If you enable an existing project that already includes some user-defined memo fields, the added fields may have different names.

Since the Quality Center fields are memo fields and there can be no more than three user-defined memo fields per table, the attempt to map Notes to memo fields may fail. If the attempt fails, you cannot use the integration tool to map the **Notes to be added on save** field. Instead, you must add the memo fields to the project manually, and edit the XML mapping file by adding one of the following to the appropriate mapping file, using the example:

- For defects:

  `<param name="BUG_ITG_NOTES">BG_USER_25</param>`

- For requirements:

  `<param name="REQ_ITG_NOTES">RQ_USER_26</param>`

Deploying the Mapping File to PPM and Quality Center

After you have completed the mapping, use the PPM-Quality Center Integration Tool to deploy the XML mapping file (ITGQCIntegration.xml) to PPM and to Quality Center.

> **Note:** At certain times, you might need to manually edit or back up the XML mapping file. When you created the XML mapping file, it was stored either in the directory in which you installed the integration tool or in another directory that you specified (see step 14). If you do not know where the file is located, check the Windows registry. The path in the registry is `HKEY_LOCAL_MACHINE\SOFTWARE\Hewlett-Packard\PPM-Quality Center Integration Tool.`
>
> Make sure you redeploy the mapping file to both PPM and Quality Center

To deploy the XML mapping file to PPM and Quality Center:

1. Verify that the PPM server.conf parameters are as specified in step 2. Set the `ENABLE_QUALITY_CENTER_INTEGRATION` parameter to `true`.

2. In the main window of the PPM-Quality Center Integration Tool, select **File > Deploy to Centers**.

The Deploy Configuration File to Centers window opens, listing the URLs of the PPM and Quality Center servers to which the mapping file will be deployed.

3. Click **Deploy**.

The tool starts the deployment process. If any errors occur during deployment, a message is displayed in the window.

4. When the deployment completes, a message indicating that deployment completed successfully is added to the Deploy Configuration File to Centers window.

5. Click **OK** to close the window.

On the PPM Server, the mapping file is deployed to the *<PPM_Home>*/conf directory, where <PPM_Home> represents the path where the PPM instance is installed.

> **Note:** In a clustered PPM environment, you must manually deploy the ITGQCIntegration.xml mapping file to all nodes that do not use a shared *<PPM_Home>*/conf directory. The integration tool cannot deploy the mapping file to multiple nodes.
>
> As needed, copy the mapping file from the *<PPM_Home>*/conf directory on the computer on which the integration tool is installed to a convenient location or device (or select **File > Save To** in the integration tool and specify the location). Then copy the file to the *<PPM_Home>*/conf directory in the other nodes in the cluster.

On the Quality Center server (or servers), the mapping file is deployed to the *<QC_Home>*/repository/sa/DomsInfo/BTO directory, where *<QC_Home>* represents the path where Quality Center is installed.

For information about changing and otherwise maintaining existing mappings, see "Managing Existing Mappings" on page 163.

This completes creating and deploying the mapping file. To configure PPM for the integration, proceed to configure PPM Center for the integration. For more information, see "Configuring PPM for the Integration " below.

Configuring PPM for the Integration

Before beginning to configure the integration as described in the following sections, make sure that ALM bundles have been installed and initially configured as described in "Installing and Setting Up ALM Content Bundle" on page 26.

Connecting PPM Server with Quality Center Servers

Make sure that an HTTP port is open between the PPM Server and each Quality Center server having projects that are to be integrated.

Configuring server.conf Parameters in PPM

> **Note:** For more information about the steps in this procedure, see the *Installation and Administration Guide*.

Add (if not present) and specify the parameters related to Quality Center integration to the PPM server.conf configuration file, as follows:

1. Stop the PPM Server.

2. Run the script sh ./kConfig.sh.

   Set the parameters and values as shown in the following table. (All parameter names begin with com.kintana.core.server. but that is not shown in the table.)

Parameter	Value
ENABLE_QUALITY_CENTER_INTEGRATION	Set this parameter to `false` if an XML mapping file has not been generated and deployed to PPM and Quality Center. Set this parameter to `true` if an XML mapping file has been generated and deployed to PPM and Quality Center, so that integration can be enabled. If a mapping file has not been deployed and you set this parameter to `true,` the PPM Server will not restart. This parameter controls whether PPM attempts to send information to Quality Center. (Even if this parameter is set to `false,` Quality Center sends information to PPM.)
BASE_URL (already present in `server.conf`)	The URL of the PPM Server. By default, contains the host name of the PPM Server, for example, `http://ppmhost:8080.` However, if the PPM Server is installed in a WAN, use the IP address of the PPM Server, for example, `http://192.60.80.01:8080,` rather than its host name.

Parameter	Value
ENABLE_QUALITY_CENTER_ METRICS_SYNC	Always set this parameter to `false`. It does not apply to ALM entities.

3. Restart the PPM Server.

For information about using the integration, see "Using the Integration of PPM with Quality Center/ALM" on page 215.

Managing Existing Mappings

After you have configured PPM and Quality Center for integration, you can use the integration tool to make changes to the configuration.

You can change the configuration for a request type mapping or for an entire Quality Center project. When you change a mapping for a project, the changes apply to all the request types mapped to the project. For example, if you delete the mapping for a project, the mapping for all of the project's mapped request types is also deleted.

As described in the following sections, you can do the following:

- Delete a mapping

- Disable a mapping

- Re-enable a previously disabled mapping

- View and change a mapping

- Enable and disable request hierarchy synchronization

- Enable and disable email notification on requirement creation

- Enable and disable email notification on requirement update

Tip: Remember to redeploy the mapping file to PPM and Quality Center after any mapping revision described in the following sections. See "Deploying the Mapping File to PPM and Quality Center" on page 160.

Deleting a Mapping

When you delete a mapping, the connection between the associated fields in PPM and Quality Center is removed, and updating a field in one application no longer causes an update in the other.

> **Note:** If you later want the applications to update each other, you will need to create a new mapping.

To delete a mapping:

1. Right-click the project or request type of interest and select **Delete Mapping**.

   The tool asks whether you want to delete the mapping.

2. Click **Yes.**

Disabling and Re-Enabling a Mapping

When you disable a mapping, the mapping is not deleted, but creating a request in PPM does not create a defect or requirement in the Quality Center project. In addition, updating a field in one application does not update the field to which it is mapped in the other application.

To disable the mapping between a request type and a defect or requirement:

1. Right-click the project or request type of interest and select **Disable Mapping**.

   The tool asks whether you want to disable the mapping.

2. Click **Yes.**

To enable a mapping that was previously disabled, right-click the project or request type and select **Enable Mapping**.

Viewing and Changing a Mapping

To view a mapping and make changes to the mapping:

1. Right-click the request type of interest and select **Mappings**.

   The Map Fields window opens.

2. If the **Mapping** tab is not already selected, select it to display the mapping.

3. Use this tab to change the mapping in the same way you created the original mapping. See "Creating the Mapping Between PPM and Quality Center Fields" on page 153.

Enabling and Disabling Request Hierarchy Synchronization

You can enable or disable the request hierarchy synchronization between a PPM request and a Quality Center requirement.

To enable the synchronization:

1. Right-click **requirement**.

2. If the **Synchronize Request Hierarchy** option is not selected (has no check mark), click it to select the option.

To disable the synchronization:

1. Right-click **requirement**.

2. If the **Synchronize Request Hierarchy** option is selected (has a check mark), click it to clear the option.

For information about the effect of request hierarchy synchronization, see "Request Hierarchy Synchronization " on page 221.

Enabling and Disabling Email Notification on Requirement Creation

> **Note:** To send any notifications, Quality Center must be configured to automatically send them, using email addresses established in Quality Center. See "Available PPM Integrations " on page 18 for information about accessing the Quality Center documentation.

You can enable or disable sending an automatic email notification when a requirement is created by the integration.

To enable the email notification for requirement creation:

1. Right-click **requirement**.

2. If the **Notify When Created** option is not selected (has no check mark), click it to select the option.

To disable the email notification for requirement creation:

1. Right-click **requirement**.

2. If the **Notify When Created** option is selected (has a check mark), click it to clear the option.

Enabling and Disabling Email Notification on Requirement Update

> **Note:** To send any notifications, Quality Center must be configured to automatically send them, using email addresses established in Quality Center. See "Available PPM Integrations " on page 18 for information about accessing the Quality Center documentation.

You can enable or disable sending an automatic email notification when a requirement is updated by the integration.

To enable the email notification for requirement update:

1. Right-click **requirement**.

2. If the **Notify When Update** option is not selected (has no check mark), click it to select the option.

To disable the email notification for requirement update:

1. Right-click **requirement**.

2. If the **Notify When Update** option is selected (has a check mark), click it to clear the option.

Default Field Mappings for PPM and Quality Center Version 10.00

The following sections apply to integration with Quality Center version 10.00 only, and they describe the default field mappings that are available for integration with Quality Center defects and requirements.

Default Field Mappings for Quality Center Version 10.00 Defects

"Default Field Mappings for Quality Center Version 10.00 Defects" above describes the default defect mappings that can be modified for the integration between the Defects Module in Quality Center and the ALM - Defect Template with Quality Center Integration request type in PPM. The Override column indicates which field is dominant by default—PPM (if set to PPM), Quality Center (if set to QC), or neither (if set to BIDIRECTIONAL). For more information, see "Creating the Mapping Between PPM and Quality Center Fields" on page 153.

Table 6-6. Default defect mappings you can modify

Quality Center Field Name, Database ID, and Field Type	PPM Field Name, Database ID [a], and Field Type	Override
Reproducible BG_REPRODUCIBLE Y/N	Reproducible REPRODUCIBLE Y/N	QC
Project BG_PROJECT List	Application APPLICATION_CODE List	QC
Actual Fix Time BG_ACTUAL_FIX_TIME Date	Actual Fix Time (days) ACTUAL_FIX_TIME Date	QC
Closed in Version BG_CLOSING_VERSION List	Closed in Version CLOSING_VERSION Text (40)	QC
Severity BG_SEVERITY Enumeration	Severity SEVERITY Drop-down list	BIDIRECTIONAL
Closing Date BG_CLOSING_DATE Date	Closed on CLOSING_DATE Date	QC
Detected in Version BG_DETECTION_VERSION List	Detected in Version DETECTION_VERSION Text (40)	BIDIRECTIONAL
Planned Closing Version BG_PLANNED_CLOSING_VER List	Planned Closing Version PLANNED_CLOSING_VER List	BIDIRECTIONAL

Table 6-6. Default defect mappings you can modify, continued

Quality Center Field Name, Database ID, and Field Type	PPM Field Name, Database ID [a], and Field Type	Override
Estimated Fix Time BG_ESTIMATED_FIX_TIME Number	Estimated Fix Time (days) ESTIMATED_FIX_TIME Numeric Text (10 digits)	BIDIRECTIONAL
Comments BG_DEV_COMMENTS Memo	Developer Comments DEV_COMMENTS Text (1800)	PPM
Detected on Date BG_DETECTION_DATE Date	Created On CREATION_DATE Date	PPM
Priority BG_PRIORITY Enumeration	Priority DEFECT_PRIORITY_CODE Drop-down list	BIDIRECTIONAL
Description BG_DESCRIPTION Memo	Detailed Description DEFECT_DESCRIPTION Text (1800)	BIDIRECTIONAL
Detected By BG_DETECTED_BY Quality Center users list	Detected in Quality Center by QC_DETECTED_BY Text (40)	QC
Status [b, c] BG_STATUS Enumeration	Request Status [b] STATUS_ID List	PPM [b]
	Quality Center Defect Status [c] KNTA_QC_DEFECT_STATUS Text (300)	QC [c]
Summary BG_SUMMARY Text (255)	Summary DESCRIPTION Text (200)	BIDIRECTIONAL

a. The listed PPM database IDs are the same as in the PPM Workbench. The exact database IDs are displayed by the integration tool.

b. When the Request Status field is updated in PPM, the new status is sent to Quality Center. If the new status matches one of the defect status in Quality Center, the Status field in Quality Center is updated; if not, the update is ignored by Quality Center.

c. When the status of a defect is updated in Quality Center, the associated Quality Center Defect Status field in PPM is updated accordingly if the value sent by Quality Center is a valid workflow step transition in PPM.

Default Field Mappings for Quality Center Version 10.00 Requirements

The table below describes the default requirement mappings that can be modified for the integration between the Requirements Module in Quality Center and the ALM - Request for Change (RFC) request type or the ALM - Release Management request type in PPM. The Override column indicates which field is dominant by default—PPM (if set to PPM), Quality Center (if set to QC), or neither (if set to BIDIRECTIONAL). For more information, see step 8.

Table 6-7. Default requirement mappings you can modify

Quality Center Field Name, Database ID, and Field Type	PPM Field Name, Database ID [a], and Field Type	Override
Priority RQ_REQ_PRIORITY Enumeration	RFC Priority PRIORITY_CODE Drop-down list	PPM
Author [b] RQ_REQ_AUTHOR User list	Created By CREATED_BY User list	PPM
ITG Request Status [c, d] RQ_USER_XX [e] Enumeration	RFC Status [c] STATUS_ID List	PPM [c]
	Quality Center Status [d] KNTA_QC_REQUIREMENT_ STATUS Text (300)	QC [d]
Name RQ_REQ_NAME Text (255)	RFC Summary DESCRIPTION Text (200)	PPM
ITG Request Description RQ_USER_XX [e] Memo	RFC Description RFC_DESCRIPTION Text (1800)	PPM
Assigned To RQ_USER_XX [e] User list	Quality Center Assigned To User KNTA_QC_ASSIGNED_TO	PPM

a. The listed PPM database IDs are the same as in the PPM Workbench. The exact database IDs are displayed by the integration tool.

b. The Author field in Quality Center displays the name of the user who created the request in PPM. Quality Center can accept any name, but PPM cannot. If you configure this field to be bidirectionally updateable and a user selects a user name in

Table 6-7. Default requirement mappings you can modify, continued

Quality Center Field Name, Database ID, and Field Type	PPM Field Name, Database ID [a], and Field Type	Override
Quality Center that does not exist in PPM, the operation will fail. User name lists must therefore be synchronized.		
c. When the RFC Status field is updated in PPM, the new status is sent to Quality Center. If the new status matches one of the requirement statuses in Quality Center, the ITG Request Status field in Quality Center is updated; if not, the update is ignored by Quality Center.		
d. When the status of a requirement is updated in Quality Center, the associated Quality Center Status field in PPM is updated accordingly if the value sent by Quality Center is a valid workflow step transition in PPM.		
e. The Quality Center fields with the database ID of RQ_USER_XX are user fields that are added to Quality Center when using the integration tool to enable a project. The value of XX is determined when the user field is added to Quality Center.		

Fields the Integration Enables in Quality Center Version 10.00 Entities

The following sections apply to integration with Quality Center version 10.00 only. They describe the initially disabled fields in Quality Center defects and requirements that the PPM-Quality Center Integration Tool enables to support the integration with PPM request types.

The integration automatically populates these Quality Center defect or requirement fields from data in PPM. These are not mappings you establish between fields in PPM and Quality Center. The integration updates some Quality Center fields on an ongoing basis as indicated.

In addition, the Attachments tab in Quality Center lists URLs for all of the attachments to the PPM request. Clicking one of the links opens the PPM request (after login).

Fields the Integration Enables in Quality Center Version 10.00 Defects

The table below describes the fields that the PPM-Quality Center Integration Tool enables in Quality Center defects when the project is enabled (in step 17), and their PPM data source. These fields should not be modified (except for their Labels, as desired).

Table 6-8. Fields the integration enables in Quality Center version 10.00 defects

PPM Data Source	Quality Center Field Name	Quality Center Field Database ID	Quality Center Field Type	Description
PPM Server	ITG Server	BG_USER_XX [a]	Text (120)	User field containing PPM

Table 6-8. Fields the integration enables in Quality Center version 10.00 defects, continued

PPM Data Source	Quality Center Field Name	Quality Center Field Database ID	Quality Center Field Type	Description
base URL				URL.
Name of integrated request type	ITG Request Type	BG_USER_XX [a]	Text (40)	User field containing the PPM request type of the associated PPM request. Used for field mapping.
Notes (formatted)	ITG Notes [b]	BG_USER_XX [a]	Memo	Quality Center memo field. Stores PPM notes. Always overridden by PPM. Added to Quality Center if the user chooses to synchronize the Notes field.
Request ID	ITG Request Id	BG_REQUEST_ ID	Integer	System field containing PPM request ID.

a. The Quality Center fields with the database ID of BG_USER_XX are user fields that are added to Quality Center when using the integration tool to enable a project. The value of XX is determined when the user field is added to Quality Center.

b. Updated on an ongoing basis.

Fields the Integration Enables in Quality Center Version 10.00 Requirements

The table below describes the fields that the PPM-Quality Center Integration Tool enables in Quality Center requirements when the project is enabled (in "Enabling a Quality Center Project for the Integration" on page 151), to support integration, and their PPM data source. These fields should not be modified (except for their Labels, as desired).

Table 6-9. Fields the integration enables in Quality Center version 10.00 requirements

PPM Data Source	Quality Center Field Name	Quality Center Field Database ID	Quality Center Field Type	Description
PPM Server base URL	ITG Server	RQ_USER_XX [a]	Text (120)	User field containing the PPM URL.
Name of integrated request type	ITG Request Type	RQ_USER_XX [a]	Text (40)	User field containing the PPM request type of the associated PPM request. Used for field

Table 6-9. Fields the integration enables in Quality Center version 10.00 requirements, continued

PPM Data Source	Quality Center Field Name	Quality Center Field Database ID	Quality Center Field Type	Description
				mapping.
Notes (formatted)	ITG Notes [b]	RQ_USER_XX [a]	Memo	Quality Center memo field. Stores PPM notes. Always overridden by PPM. Added to Quality Center if the user chooses to synchronize the Notes field.
Request ID	ITG Request Id	RQ_REQUEST_ID	Integer	System field containing PPM request ID.
PPM request status	ITG Request Status [b]	RQ_USER_XX [a]	Text (40)	Status of the PPM request.
"Updated by PPM at <timestamp>" (translated as needed)	ITG Updates [b]	RQ_USER_XX [a]	Text (120)	User field that shows integration message for success or error in most recent operation.
PPM Server base URL	ITG Request Description	RQ_USER_XX [a]	Memo	Description of the PPM request.

a. The Quality Center fields with the database ID of RQ_USER_XX are user fields that are added to Quality Center when using the integration tool to enable a project. The value of XX is determined when the user field is added to Quality Center.

b. Updated on an ongoing basis.

Configuring Integration with ALM Version 11.00 and Later

The procedures in this section apply to integrating PPM with HPE Application Lifecycle Management (ALM) version 11.00 and later. Supported ALM versions include 11.00, 11.20, 11.50, and 12.00.

If you are integrating PPM with Quality Center version 10.00, go to .

Caution: The PPM-Quality Center Integration Tool must be used only for integration with Quality Center version 10.00. It cannot detect integrations that were previously established with ALM version 11.x and later instances, and any existing integration with a version 11.x and later instance prevails if you try to use the integration tool for that integration. If the PPM-Quality Center Integration Tool was previously installed, do not try to use it to configure integration with ALM 11.x and later.

Caution: If PPM is at version 9.40 and you upgrade a Quality Center version 10.00 instance to ALM version 11.x, existing integrations of PPM requests and Quality Center entities (defects or requirements) on that instance will not be upgraded. If the integrations are reestablished, they will operate as new integrations.

Note: In configuring an ALM integration, make sure you establish a one-to-one, unique mapping between each integrated PPM request type and its associated ALM entity (defect or requirement).

Note: If you integrate PPM version 9.40 with ALM version 11.50 or later, you can use the PPM-ALM Integration Tool (download link available on the Integration Configurations page) to enable ALM project fields for the integration. This tool can be used on any Windows machine that can open HTTP connections to the PPM Server and to the ALM server.

For important information about integrating PPM request types with defects and requirements on multiple QC/ALM servers, where some servers can be at Quality Center version 10.00 and others can be at ALM version 11.x, see .

Before proceeding, verify that ALM version 11.x or later is installed and running on the ALM server to be integrated.

Overview of Installation and Configuration Process

The procedures for configuring integration of PPM and ALM 11.x and later require the Configuration license and are performed almost entirely by using the PPM standard interface. They are described in detail in the following sections and are summarized as follows:

- Verify that the PPM Server is running and that each ALM server to be integrated is running version 11.00 or later.

- In ALM, enable fields related to PPM as needed.

- Modify value lists in ALM to support the integration with PPM requests, as needed.

- Based on the eligible request type you select in PPM, configure the integration details, including:

○ Field mappings between PPM fields and ALM fields, including value mappings

○ Email notification options for errors and creation or update of entities.

○ Whether creating a PPM request creates an ALM entity (defect or requirement), or creating an ALM entity creates a PPM request

○ If you are integrating a PPM request type with an ALM requirement:

○ Default folder in ALM for new requirements

○ Whether the requirement hierarchy in ALM is to be synchronized with (driven by) the request hierarchy in PPM

Changes to ALM Value Lists and Workflows Made by the Integration Tool

When the user enables an ALM project for integration using the PPM-ALM Integration Tool, the tool performs one of the following actions:

- If the project is new, the tool creates ALM value lists and adds ALM workflow scripts to the user's existing scripts.

- If the project already exists, the tool updates some of the ALM value lists so that those lists contain the same values as they would for a new project.

The lists and ALM scripts create an ALM project that can work as is with the provided ALM content bundle request types and workflows.

As described in the following sections, the integration tool changes the value lists and workflow enforcement in an ALM project to enable the project for integration.

Changes to Value Lists

As part of enabling an ALM project for integration and establishing a field mapping, the integration tool adds two new value lists and adds a new value to an existing default value list in ALM, as follows:

- New Requirement Status value list, with the following values:

○ New

○ Cancelled

○ Closed

○ 1-Requirements Setup Completed

- 2-Test Plan Setup Completed

- 3-Test Lab Setup Completed

- 4-Running Tests in Quality Center

- 5-Test Execution Completed

- New Test Level value list, with the following values:

  - Functional

  - Integration

  - Regression

  - Sanity

- New default value of Deleted for the existing Bug Status value list

Workflow Enforcement

As part of enabling an ALM project for integration and establishing a field mapping, the integration tool updates the ALM project workflow to enforce the following constraints on ALM entities:

- Defect. The user can make only the following status changes:

  - Fixed to Closed

  - Rejected to Closed

  - Fixed to Reopen

- Requirement. The user can make only the following status changes:

  - New to 1-Requirements Setup Completed

  - 1-Requirements Setup Completed to 2-Test Plan Setup Completed

  - 2-Test Plan Setup Completed to 3-Test Lab Setup Completed

  - 4-Running Tests in Quality Center to 5-Test Execution Completed

  - 6-Running Sanity Tests in Quality Center to 7-Sanity Testing Completed

Fields the Integration Enables in ALM Entities

The following sections apply to integration with ALM 11.00 and later. They describe the initially disabled fields in ALM defects and requirements that you may need to enable to support the integration with PPM request types. Fields in ALM that are related to PPM are not initially enabled.

The integration automatically populates these ALM defect or requirement fields from data in PPM. These are not mappings you establish between fields in PPM and ALM. The integration updates some ALM fields on an ongoing basis as indicated.

In addition, the Attachments tab in ALM lists URLs for all of the attachments to the PPM request. Clicking one of the links opens the PPM request (after login).

To enable the fields associated with the integration in ALM defects or requirements,

- For ALM version 11.00 or 11.20

  Run the scripts provided by HPE Software Support. For instructions, visit
  https://softwaresupport.hp.com/group/softwaresupport/search-result/-
  /facetsearch/document//KM1352699.

- For ALM version 11.50 or 12.00, enable these fields by using the PPM Center – ALM Integration Tool.

Fields Associated with the Integration in ALM Defects

The table below describes the fields in ALM defects that you may need to enable for the integration, and their PPM data source. These fields should not be modified (except for their Labels, as desired).

Table 6-12. Fields you may need to enable in ALM 11.x defects

PPM Data Source	ALM Field Name	ALM Field Database ID	ALM Field Type	Description
PPM Server base URL	PPM Server URL	BG_ REQUEST_ SERVER	Text (120)	PPM URL.
Name of integrated request type	PPM Request Type	BG_ REQUEST_ TYPE	Text (120)	PPM request type of the associated PPM request. Used for field mapping.

Table 6-12. Fields you may need to enable in ALM 11.x defects, continued

PPM Data Source	ALM Field Name	ALM Field Database ID	ALM Field Type	Description
Notes (formatted)	PPM Request Note [a]	BG_ REQUEST_ NOTE	Memo	ALM memo field. Stores PPM notes. Always overridden by PPM. Added to ALM if the user chooses to synchronize the **Notes** field.
Request ID	PPM Request Id	BG_ REQUEST_ID	Integer	System field containing PPM request ID.
a. Updated on an ongoing basis				

Fields Associated with the Integration in ALM Requirements

The table below describes the fields in ALM requirements that you may need to enable for the integration, and their PPM data source. These fields should not be modified (except for their Labels, as desired).

Table 6-13. Fields you may need to enable in ALM 11.x requirements

PPM Data Source	ALM Field Name	ALM Field Database ID	ALM Field Type	Description
PPM Server base URL	PPM Server URL	RQ_REQUEST_ SERVER	Text (120)	PPM URL.
Name of integrated request type	PPM Request Type	RQ_REQUEST_TYPE	Text (120)	PPM request type of the associated PPM request. Used for field mapping.
Notes (formatted)	PPM Request Note [a]	RQ_REQUEST_ NOTE	Memo	ALM memo field. Stores PPM notes. Always overridden by PPM. Added to ALM if the user chooses to synchronize the **Notes** field.
Request ID	PPM Request Id	RQ_REQUEST_ID	Integer	System field containing PPM request ID.

Table 6-13. Fields you may need to enable in ALM 11.x requirements, continued

PPM Data Source	ALM Field Name	ALM Field Database ID	ALM Field Type	Description
PPM request status	PPM Request Status [a]	RQ_REQUEST_ STATUS	Lookup List	Status of the PPM request.
"Updated by PPM at <timestamp>" (translated as needed)	PPM Synchronization Data [a]	RQ_REQUEST_ UPDATES	Text (120)	Integration message for success or error in most recent operation.

a. Updated on an ongoing basis

Installing the PPM-ALM Integration Tool (for ALM Version 11.50 and later)

> **Note:** For integration with ALM version 11.00 or 11.20, you need to run the scripts provided by HPE Software Support manually. For more information, see "Fields the Integration Enables in ALM Entities" on page 176.

Requirements for the PPM-ALM Integration Tool

The following are the PPM and ALM requirements that should be met before installing the PPM-ALM Integration Tool.

Server-Side Requirements

In order to install and use the PPM-ALM Integration Tool, users need the following:

- HPE Application Lifecycle Management (ALM) version 11.50 or later

Client-Side Requirements

The following software is required on the client:

- Microsoft .NET Framework version 3.5 or later

- ALM Connectivity Add-in

  To download and install the ALM Connectivity Add-in,

    a. Launch HPE Application Lifecycle Management version 11.50 from your Internet Explorer browser.

    b. Click **Add-Ins Page**.

       The Application Lifecycle Management - Add-ins page displays.

    c. Click **HPE ALM Connectivity**.

       The HPE ALM Connectivity Add-in page displays.

    d. Click **Download Add-in**.

    e. Run `TDConnect.exe` to install the add-in.

Downloading and Installing the PPM-ALM Integration Tool

1. Log on to PPM.

2. From the menu bar, select Open > Administration > Integrations.

   The Manage Application Changes tab page of the Integration Configurations opens.

3. Click the Download PPM Center-ALM Integration Tool link.

   Save the setup.exe file to a local folder.

4. Double-click the setup.exe file to launch the InstallShield wizard.

   By default, the tool gets installed in the `C:\Program Files\Hewlett-Packard\PPM Center-Application Lifecycle Management Integration Tool` directory.

5. Follow the instructions in the wizard. When the installation completes, do not launch the PPM-ALM Integration Tool yet. Proceed to "Configuring an ALM Project for the Integration (for ALM 11.50 or Later)" on the next page.

Uninstalling the Integration Tool

If you want to uninstall the PPM-Application Lifecycle Management Integration Tool (for example, to be able to run the tool from another Windows machine only), select **Start > Settings > Control Panel > Add/Remove Programs** and follow the instructions on the screen.

Configuring an ALM Project for the Integration (for ALM 11.50 or Later)

As documented in the following sections, you establish integration of PPM with ALM independently for each ALM project as needed, using wizards in the PPM-Application Lifecycle Management Integration Tool to do the following:

- Enable PPM integration with ALM by enabling PPM related fields in ALM

Note: For ALM versions 11.00 and 11.20, you need to manually enable fields associated with the integration in ALM. For details, see "Fields the Integration Enables in ALM Entities" on page 176.

Enabling an ALM Project for the Integration

Use the PPM-Application Lifecycle Management Integration Tool to enable an ALM project for the integration as described in this section.

Note: HPE strongly recommends that you enable a project only once, otherwise problems might occur in the integration. There is no reason to enable a project twice.

1. From the Windows Start menu, select **All Programs > Hewlett-Packard Enterprise > PPM-Application Lifecycle Management Integration Tool.**

   The main PPM-Application Lifecycle Management Integration Tool window opens.

2. Select **Action > Enable Project.**

   The Enable Application Lifecycle Management Project wizard opens.

3. Click **Next** to continue.

4. In the **Host** field, type the URL of the ALM server.

5. Click **Next** to continue.

6. In the **User Name** and **Password** fields, specify the user name and password of the ALM project administrator.

7. Click **Test Connection** to test the connection with ALM.

8. If a message appears stating that connection was successful, click **OK** on the message, then

click **Next.** Otherwise resolve the connection issue.

9. Select an ALM domain and project, and specify whether you want to map ALM defects, requirements, or both.

10. Click **Next** to continue.

    The wizard displays user-defined fields related to the PPM request that the integration tool will enable in the ALM project, for defects, requirements, or both, as specified in .

11. Click **Next** to continue.

    The listed fields are added to the ALM project, and a dialog pops up asking you whether you want to the ALM workflow script is updated to support integration with PPM.

    > **Note:** If the ALM project already contains one or more fields with the same names, a dialog box asks you whether you want to change the names of the added fields. If you select **Yes,** the added fields will be given different names. If you select **No,** the procedure to enable the project for integration is aborted.

    > **Note:** If the ALM project does not have a workflow script in its repository, a dialog box asks you whether you want to create a new one. If you select **Yes**, a new script with support for the integration is added to the ALM repository. If you select **No**, the script file is not created.
    >
    > If the project is not new and its script files have been previously customized, select **Yes** and later manually merge the customized scripts with the scripts that the wizard adds automatically, which are located in the installation directory of the PPM-Application Lifecycle Management Integration Tool.

12. Click **Finish** to complete enabling the project for integration and close the wizard.

Default Field Mappings for PPM and ALM

The following sections apply to integration with ALM only, and they describe the default field mappings that are available for integration with ALM defects and requirements.

Default Field Mappings for ALM Defects

The table below describes the default defect mappings that can be modified for the integration between the Defects Module in ALM and the ALM - Defect Template with Quality Center Integration request type in PPM. The Control column indicates which field is dominant by default—PPM (if set to PPM),

ALM (if set to QC/ALM), or neither (if set to BIDIRECTIONAL). For more information, see "Configuring Field Mappings" on page 193.

Table 6-10. Default defect mappings you can modify

ALM Field Name, Database ID, and Field Type	PPM Field Name, Database ID [a], and Field Type	Control
Detected on Date BG_DETECTION_DATE Date	Created On CREATION_DATE Date	PPM
Summary BG_SUMMARY Text (255)	Summary DESCRIPTION Text (200)	BIDIRECTIONAL
Severity BG_SEVERITY Enumeration	Severity SEVERITY Drop-down list	BIDIRECTIONAL
Detected By BG_DETECTED_BY Quality Center users list	Created by CREATED_BY Text (40)	QC/ALM
Priority BG_PRIORITY Enumeration	Defect Priority DEFECT_PRIORITY_CODE Drop-down list	PPM
Actual Fix Time BG_ACTUAL_FIX_TIME Date	Actual Fix Time (days) ACTUAL_FIX_TIME Date	BIDIRECTIONAL
Estimated Fix Time BG_ESTIMATED_FIX_TIME Number	Estimated Fix Time (days) ESTIMATED_FIX_TIME Numeric Text (10 digits)	BIDIRECTIONAL
Reproducible BG_REPRODUCIBLE Y/N	Reproducible REPRODUCIBLE Y/N	QC/ALM
Detected in Version BG_DETECTION_VERSION List	Detected in Version DETECTION_VERSION Text (40)	QC/ALM
Closed in Version BG_CLOSING_VERSION List	Closed in Version CLOSING_VERSION Text (40)	QC/ALM
Description BG_DESCRIPTION Memo	Detailed Description DEFECT_DESCRIPTION Text (1800)	BIDIRECTIONAL

Table 6-10. Default defect mappings you can modify, continued

ALM Field Name, Database ID, and Field Type	PPM Field Name, Database ID [a], and Field Type	Control
Assigned To RQ_USER_XX [b] User list	Quality Center Assigned To User KNTA_QC_ASSIGNED_TO	QC/ALM
Closing Date BG_CLOSING_DATE Date	Closed on CLOSING_DATE Date	QC/ALM

a. The listed PPM database IDs are the same as in the PPM Workbench. The exact database IDs are displayed by the integration tool.

b. The ALM fields with the database ID of RQ_USER_XX are user fields that are added to ALM when using the integration tool or running the script to enable a project. The value of XX is determined when the user field is added to ALM.

Default Field Mappings for ALM Requirements

The table below describes the default requirement mappings that can be modified for the integration between the Requirements Module in Quality Center and the ALM - Request for Change (RFC) request type or the ALM - Release Management request type in PPM. The Control column indicates which field is dominant by default—PPM (if set to PPM), Quality Center (if set to QC/ALM), or neither (if set to BIDIRECTIONAL). For more information, see "Configuring Field Mappings" on page 193.

Table 6-11. Default requirement mappings you can modify

ALM Field Name, Database ID, and Field Type	PPM Field Name, Database ID [a], and Field Type	Control
Name RQ_REQ_NAME Text (255)	RFC Summary DESCRIPTION Text (200)	BIDIRECTIONAL
Description RQ_USER_XX [b] Memo	RFC Description RFC_DESCRIPTION Text (1800)	BIDIRECTIONAL
Comments DEV-COMMENTS Text (255)	Effect of no change RFC_EFFECT_NO_CHANGE Text Area - 1800	BIDIRECTIONAL

a. The listed PPM database IDs are the same as in the PPM Workbench. The exact database IDs are displayed by the integration tool.

b. The ALM fields with the database ID of RQ_USER_XX are user fields that are added to ALM when using the integration tool to enable a project. The value of XX is determined when the user field is added to ALM.

Modifying Value Lists in ALM

As needed, create or modify the value lists in ALM defects and requirements to support the integration with PPM requests. See "Available PPM Integrations " on page 18 for information about accessing the ALM documentation.

The following configuration procedures include configuring value mappings when both fields in a field mapping are value lists.

Configuring the Integration

To integrate a PPM request type with an ALM version 11.x or later entity:

1. Log on to PPM.

2. From the menu bar, select **Open > Administration > Integration**.

   The Integration Configurations screen of the Manage Application Change opens.

The Manage Application Change page lists all the PPM request types that are "eligible," that is, the request types that are already integrated or can be integrated with ALM defects or requirements. A request type is eligible if it includes the QC/ALM Defect Information field group (for defects) or the QC/ALM Info field group (for requirements).

For integration with ALM 11.x or later, the eligible request types have links that you can click to configure their integrations.

Request types that are integrated with Quality Center version 10.00 (displayed as **QC 10** in the **Version** column) are read only, with no links for configuration in the standard interface—those request types must be configured using the PPM-Quality Center Integration Tool, which works only with Quality Center version 10.00 (see "Selecting the Appropriate Integration Procedure" on page 145).

The status for each request type on the Integration Configurations screen is one of the following:

- **Enabled** . The integration of the request type and ALM entity (defect or requirement) is enabled and operable.

- **Disabled** . The integration of the request type and ALM entity is disabled and not operable.

    > **Tip:** You can easily toggle the status of an integration by clicking the status icon. That is, clicking disables the integration, and clicking enables the integration.

- **Read-Only** . The integration of the request type and QC entity is operable but not configurable. You can upgrade the integration configuration to ALM version 11.00 or later. For more information, see "Upgrading from Integration with Quality Center 10.00 to Integration with ALM" on page 204.

The QC/ALM data displayed for each request type also includes the URL of the QC/ALM server and the QC/ALM version, domain, project, and entity type (defect or requirement).

3. Either modify an existing integration configuration or add a new integration configuration.

    - To modify an existing integration configuration (with ALM 11.00 or later), in the **Request Type** column, click the desired eligible request type.

    - To add a new integration configuration, in the Server Configuration Details section, select a request type from the drop-down list of available request types and click **Add Integration Configuration**.

The Configure QC/ALM Integration for Request Type: <*Request Type*> section opens to the **Step 1: Configure Server Details** tab page.

In this example, the selected request type ALM – Default Template with Quality Center Integration includes the QC/ALM Defect Information field group, so it is associated with an ALM defect, as indicated in the **Entity Type** field.

For a new integration, the **Status** is **Disabled** by default.

If the request type includes the QC/ALM Info field group instead, the request type is associated with an ALM requirement, as indicated in the **Entity Type** field. In this case, the **Integration Options** are different, as described later.

If the request type includes both the QC/ALM Defect Information field group and the QC/ALM Info field group, you will select whether a QC/ALM defect or requirement will be the associated **Entity Type**.

If you add an integration configuration for Requirement entity type, this page looks like the follows:
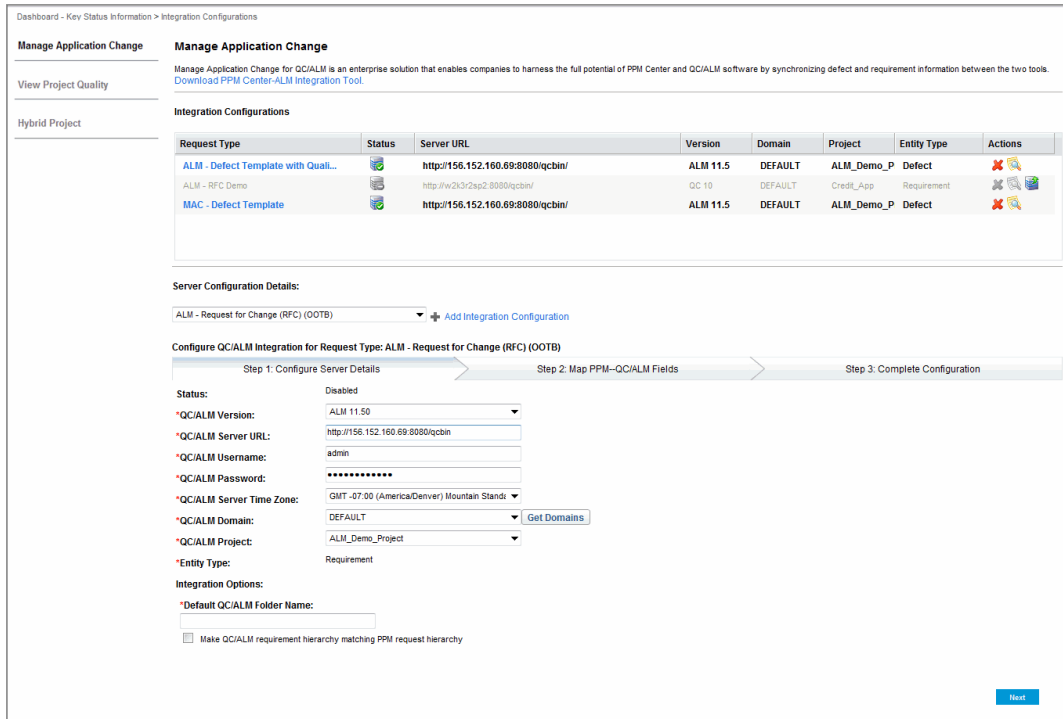
4. Modify or complete the fields described in the following table.

Field Name (*Required)	Description
Status	(Read only) Status of the integration of this request type. After you finish configuring the integration, you enable or disable the integration by clicking the Status icon in the Integration Configurations list.
*QC/ALM Version	Select an ALM version from the drop-down list. Valid values include: ○ `ALM 11.00` ○ `ALM 11.20` (The complete version number is ALM 11.00 SP2) ○ `ALM 11.50` ○ `ALM 12.00` **Caution:** There is a known issue in the ALM 12.00 side: PPM related fields are not returned in the REST response when querying for the requirement entities. This issue blocks the requirement synchronization. Make sure your ALM version includes the fix of this issue before you use ALM 12.00 or later.

Field Name (*Required)	Description
*QC/ALM Server URL	URL of the ALM server, in either of the following format: ○ `http://<ALM_Server_Host>:<Port>/qcbin/`, or ○ `https://<ALM_Server_Host>:<Port>/qcbin/`
*QC/ALM Username	Username used to access ALM.
*Password	Password for the QC/ALM Username.
*QC/ALM Server Time Zone	Time zone of the ALM server. Required to ensure that the integration correctly manages updates between fields mapped as bidirectional between PPM requests and associated ALM defects or requirements. Default is the time zone of the PPM Server. **Note:** For some of the GMT time zones, only the "Daylight Savings Time Not Used" option is available, time zones names that include city names are not available due to the changes with a third-party product. For example, you may see the "GMT +9:00 Daylight Savings Time Not Used" option only, the "GMT +9:00 (Asia/Yakutsk) Yakutsk Time" option is not available.
*QC/ALM Domain	Domain on the ALM server to use for the integration. To retrieve the set of domains, click **Get Domains.**
*QC/ALM Project	ALM project to use for the integration. (List is populated when **QC/ALM Domain** is selected.)
*Entity Type	ALM entity type to be used for integration—**Defect** or **Requirement.** Available option(s) depend on whether the QC/ALM Defect Information field group (for defects) or the QC/ALM Info field group (for requirements) or both field groups are configured in the request header type for the request type. If only one of the field groups is configured in the request header type, this field is read only.
Integration Options	Options that change based on whether the request type is integrated with a defect or a requirement. See "Configuring Integration Options" on page 192.

5. Click **Next**.

The Configure QC/ALM Integration for Request Type: <Request Type> section moves to the **Step 2: Map PPM–QC/ALM Fields** tab page.

If you add an integration configuration for Requirement entity type, this page looks like the follows:

6. Modify or complete the fields described in the following table.

Field Name	Description
QC/ALM Synchronization Control Field	Specifying a value in this field allows the QC/ALM end users to control whether they want to create a defect in PPM Center as a result of creating a defect in ALM.
	The values in the drop-down list are fields (with Y/N values) retrieved from the Defect entity type of the ALM project specified in the **Project** field.

Field Name	Description
	The field is read-only if you selected the **Creating a PPM Center request automatically creates an associated QC/ALM entity** integration option on the **Step 1: Configure Server Details** tab page.
	Limitation: If the field you specified in this option is set as a required field in ALM, errors appear no matter whether or not you build field mapping for the field. Therefore, do not set the field you selected in this option as required in ALM.
Field Mapping	Separate lists of unmapped **QC/ALM Entity Fields** and **PPM Request Fields,** followed by a table of the pre-configured **Current Field Mappings.** See "Configuring Field Mappings" on page 193.

7. Click **Next**.

The Configure QC/ALM Integration for Request Type: <Request Type> section moves to the **Step 3: Complete Configuration** tab page.

If you add an integration configuration for Requirement entity type, this page looks like the follows:

8. (Optional) Specify Notification Options on the tab.

Field Name	Description
Notification Options	Options for email notification when integration errors occur, and when integration creates or updates entities.

You can optionally send emails to addresses you specify when integration errors occur, either as they occur or as a daily consolidation.

Separately, you can optionally send emails to addresses you specify when the integration creates or updates entities, either as these changes occur or as a daily consolidation.

Event logs provide the same information as email notifications. See "Viewing Event Logs" on page 197.

This concludes the configuration procedure for integrating one PPM request type with one ALM defect or requirement.

9. Click **Save** to save the configuration.

    If you configure a new integration, you could click **Save & Enable** to both save the configuration and enable the integration for the request type.

    If you modify an existing and enabled configuration, you would see the **Save & Disable** button instead.

10. As needed, configure integration for other eligible request types, starting at step 3.

Configuring Integration Options

The **Integration Options** are different for a PPM request that is integrated with an ALM defect than for a request that is integrated with a PPM requirement, as described in the following sections.

Options for Integration with an ALM Defect

For an integrated PPM request type and ALM defect, in the **Integration Options**, select one of the following options as needed:

- **Creating a PPM request automatically creates an associated QC/ALM entity.**

  Selecting this option causes the following:

  - Creating or updating a request in PPM creates or updates the associated entity in ALM. Previously configured field mappings and value mappings, if controlled by PPM or bidirectional, apply to the entity in ALM.

  - Creating the PPM request automatically populates the request fields related to ALM.

  - If an ALM user deletes the entity, the integration re-creates the entity in ALM.

  - If a PPM user deletes the request, the association with the entity in ALM is removed but the entity is not deleted.

- **Creating a QC/ALM entity automatically creates an associated PPM request.**

  Selecting this option causes the following:

  - Creating or updating a defect in ALM creates or updates the associated request in PPM. Previously configured field mappings and value mappings, if controlled by ALM or bidirectional, apply to the request in PPM.

  - If a PPM user deletes the request, the integration re-creates the request in PPM.

  - If an ALM user deletes the defect, the association with the request in PPM is removed but the request is not deleted.

- **Creating a PPM Center request automatically creates an associated QC/ALM entity, and vice versa**.

  Selecting this option causes all results you see in the first two options.

Options for Integration with an ALM Requirement

For integration with an ALM requirement, the **Integration Options** at the bottom of the Configure QC/ALM Integration for Request Type page are different than for integration with an ALM defect.

To configure the Integration Options for an integrated PPM request type and ALM requirement:

1. Complete the Default QC/ALM Folder Name field for the ALM requirements associated with the PPM requests.

   If request hierarchy synchronization is not enabled for a request type (see step 2 below), all new requirements that the integration creates when users create requests will be placed in the ALM folder you specify in this field.

2. Specify the Make QC/ALM requirement hierarchy match PPM request hierarchy option for this request type. This option is also known as request hierarchy synchronization.

   For information about the effect of request hierarchy synchronization, see "Request Hierarchy Synchronization " on page 221.

Configuring Field Mappings

The **Step 2: Map PPM—QC/ALM Fields** tab page of the Configure QC/ALM Integration for Request Type section displays a drop-down list of the unmapped **QC/ALM Entity Fields**, a drop-down list of the unmapped **PPM Request Fields**, and the **Current Field Mappings** table. Following is an example of an integration that has some field mappings configured.
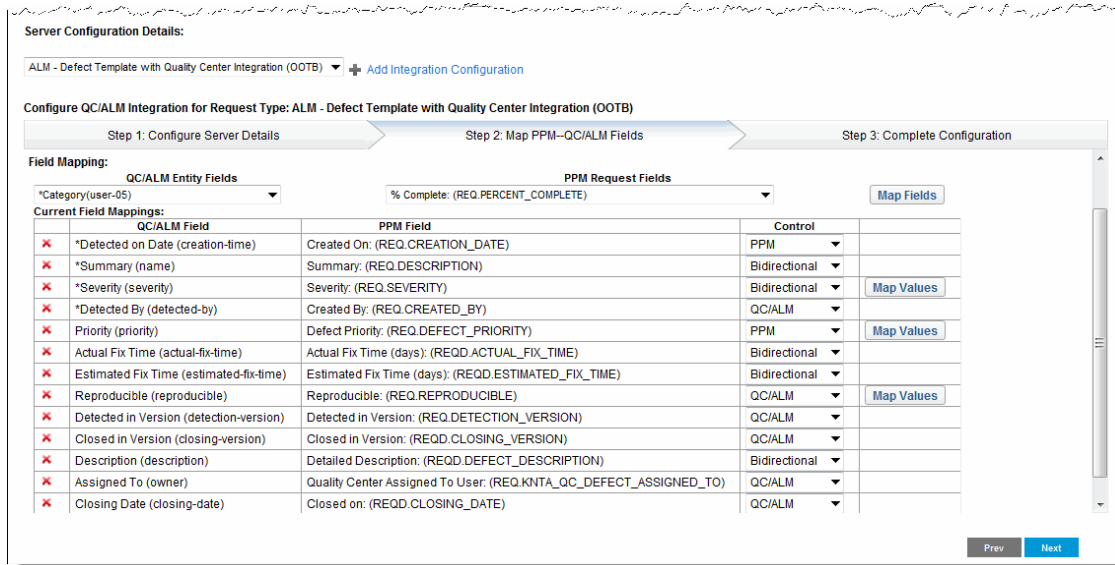
In the **Current Field Mappings** table:

- The **QC/ALM Field** column displays ALM field names, followed by their DB field names (the names of the columns in the database) in parentheses.

- The **PPM Field** column displays the PPM request field names that are mapped to (that is, associated with) the ALM fields, followed by their tokens in parentheses.

- The **Control** column specifies which field in the mapped field pair controls the other, or that a change to either field changes the other. (QC/ALM refers to the controlling field as the dominant field.) The **Control** column can have one of the following values and meanings for any pair of mapped fields (row):

  ○ **QC/ALM**. The ALM entity field controls the mapped PPM request field. When the ALM entity is created or modified in any way, then the integration updates all the mapped PPM request fields that are controlled by ALM.

    When ALM controls a mapped pair and the value in the PPM field is changed, the value in the associated ALM field is not affected.

    When the integration creates an ALM entity, the PPM request fields have no effect on the fields controlled by ALM.

  ○ **PPM.** The PPM request field controls the mapped ALM entity field. When the PPM request is created or modified in any way, then the integration updates all the mapped ALM entity fields that are controlled by PPM.

    When PPM controls a mapped pair and the value in the ALM field is changed, the value in the associated PPM field is not affected.

When the integration creates a PPM request, the ALM entity fields have no effect on the fields controlled by PPM.

- **Bidirectional.** In effect, both the PPM fields and their mapped ALM fields operate as if they are in control—when either the integrated PPM request or ALM entity is created or modified, the integration updates all the associated fields in the ALM entity or PPM request respectively, as specified by the mappings.

How to create a new field mapping:

1. From the **QC/ALM Entity Fields** drop-down list, select the ALM entity field that you want to map to a PPM request field.

   Each ALM entity field that you must map to a PPM request field is indicated by an asterisk (*) to the beginning of the field name in the **QC/ALM Entity Fields** list.

2. From the **PPM Request Fields** drop-down list, select the PPM request field that you want to map to the ALM entity field you selected.

3. Click **Map Fields**.

   The field pair is added to the **Current Field Mappings** table.

4. Repeat this procedure for all the ALM entity fields that must be mapped and for other pairs of fields you want to map.

How to configure or modify a field mapping:

- If you want to change which field, if any, controls a current field mapping (row), change the value in the **Control** column for that field mapping.

- If you want to remove a pair of fields from the mapping, click the Delete icon (✖) in the right end of the row for the field pair you want to delete.

- If both fields in the mapping are lists of values and you want to map the values, see "Configuring Value Mappings" below.

Configuring Value Mappings

When both fields in a field mapping are lists of values, the **Map Values** button appears to the right of the row in the **Current Field Mappings** table. In this case, to map the values of one field to the values of the other field:

1. Click **Map Values**.

   The Map Values window opens.

   In this example, some **QC/ALM Values** and **PPM Values** have been mapped and others have not.
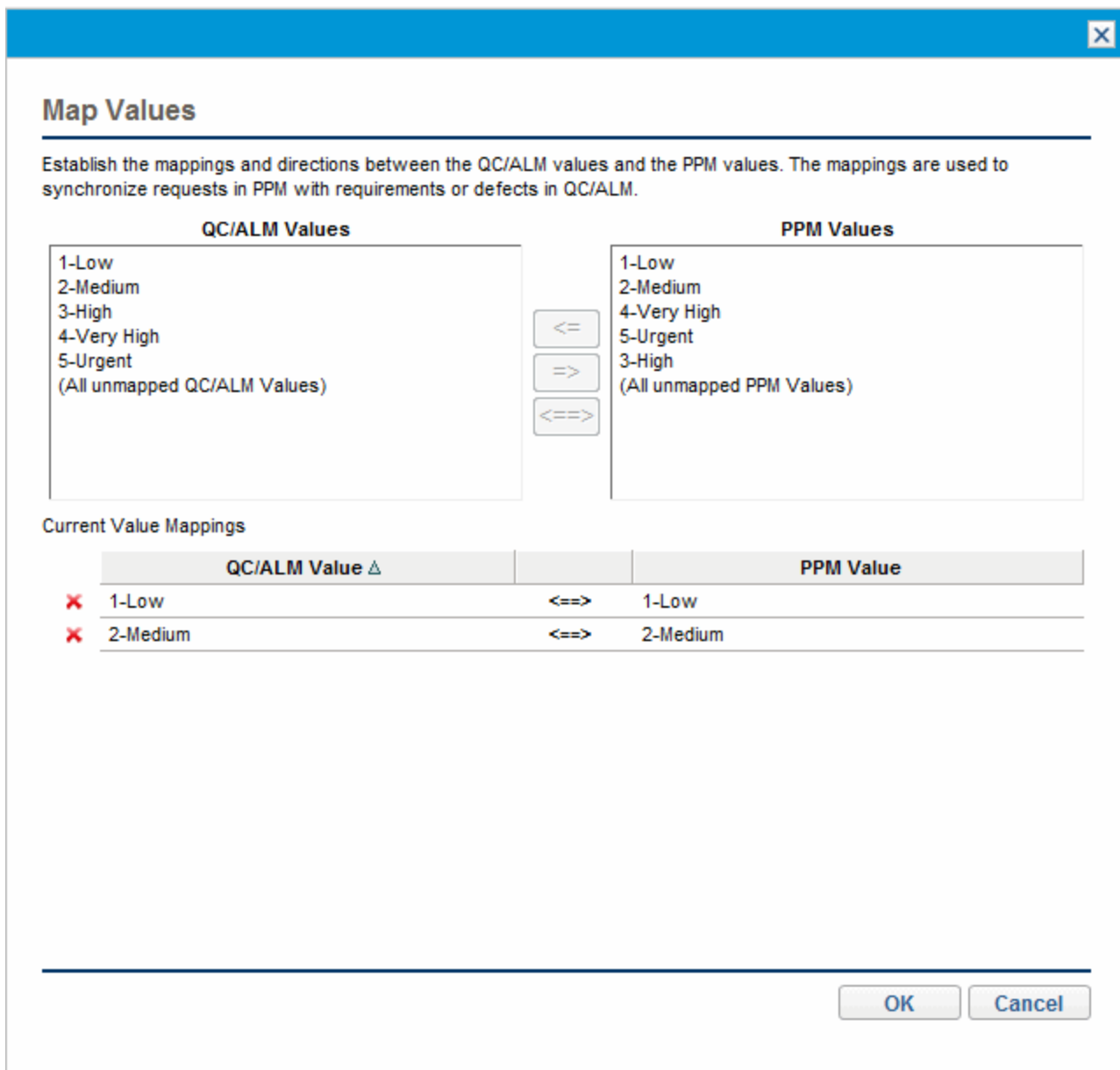
2. Select a value from the **QC/ALM Values** column and a value from the **PPM Values** column. Then select an enabled **<=**, **=>**, or **<=>** button between the columns to specify the direction of the value mapping.

The integration does not allow you to configure invalid value mappings. For example, if the field mapping specified in the **Control** column is **QC/ALM** or **PPM**, not **Bidirectional**, then only the corresponding direction for value mapping is available.

If the field mapping specified in the **Control** column is **PPM**, you can use the **(All unmapped PPM Values)** option to map all the remaining unmapped **PPM Values** to one of the **QC/ALM Values**. Conversely, if the field mapping specified in the **Control** column is **QC/ALM**, you can use the **(All unmapped QC Values)** option to map all the remaining unmapped **QC/ALM Values** to one of the **PPM Values**.

If the field mapping specified in the **Control** column is **Bidirectional**, you can map either all the remaining **PPM Values** to one of the **QC/ALM Values** or all the remaining **QC/ALM Values** to one of the **PPM Values**.

Make sure that the mapping you specify will not result in invalid values in terms of PPM or QC/ALM processes.

3. Click **OK**.

Viewing Event Logs

For integration with ALM version 11.00 or later, an event log, including error information, is maintained in PPM. The Integration Configurations page in PPM has a **View Log** icon for each request type (row) that has a deployed integration. If you click one of these icons, you can search the log for all the events that are logged for that request type or for only their error events, over the date range you specify.

The information provided in the search results includes the following:

- Nature of the event, that is, whether a PPM request, ALM defect, or ALM requirement was created or updated

- Date and time of the event

- Whether or not the event was an error

- Other details

Email notifications, when enabled, provide the same information as event logs. See step 8.

Example: Setting Up Integration Between PPM 9.2x and ALM 11.52 from Scratch

This section provides information for setting up a full integration between PPM version 9.2x and ALM version 11.52 from scratch.

Prerequisites

Make sure the following prerequisites are met:

- PPM version 9.2x is installed, with ALM content bundle deployed.
- The Oracle datatbase is already there
- ALM version 11.52 is installed

Tasks

Task 1: Update the OOTB ALM - Request For Change Workflow

1. Log on to PPM as administrator.
2. From the menu bar, select **Open > Administration > Open Workbench**.

   The PPM Workbench opens.
3. From the shortcut bar, select **Configuration > Workflows**.

   The Workflow Workbench opens to the Query tab.
4. Click **List** and then, on the **Results** tab, double-click **ALM - Request For Change**.

   The Workflow windows opens to the Layout tab.
5. Go to step **23. Non Release Sub WF**, right click and select **Edit Source** from the context menu.

   The Workflow: Non Release Sub WF window opens to the Layout tab.
6. Right click step **4. Quality Process Mode**, and select **Edit** from the context menu.

   The Workflow Step window opens to the Properties tab.
7. Update the value for **Request Status** to **Approved**.

8. Click **OK**.

9. Click **Save** and exit the workflow completely.

Task 2: Create and configure a new project in ALM

1. Log on to Site Administration of ALM.

2. On the Site Projects tab, navigate to the desired location, and provide a path for your project in the **Physical Directory** field, and click **Create Project**.

3. In the Create Project dialog, select **Create an empty project**, and then click **Next**.

4. Enter a project name in the **Project Name** field, select a domain for the project from the **In Domain** drop-down list, and then click **Next.**

5. Select **Oracle** as Database Type, and then provide credentials for the DB Server in the **Server Name**, **DB Admin User**, and **DB Admin Password** fields, and then click **Next.**

6. Specify tablespaces, and then click **Next.**

7. Select project administrators, and then click **Next**.

8. Select extensions to activate for the project, and then click **Next**.

9. Review summary of project information, select **Activate project** , and then click **Create**.

10. Click **OK** when the confirmation message box pops up.

11. Run the following SQL script to enable hidden fields in the ALM project for the requirement module for the integration:

```
update  <db_name>.SYSTEM_FIELD set sf_is_active = 'Y', SF_IS_VISIBLE_IN_NEW_
BUG='111111', sf_is_system = 'N', sf_udf_version=1,  sf_can_change_
permissions='Y', sf_is_transition_logic='Y', sf_grant_modify='111001' where
sf_table_name = 'REQ' and SF_COLUMN_NAME like 'RQ_REQUEST%';

commit;
```

> **Note:** This script applies to ALM with Oracle as database.
>
> For instructions on how to enable hidden fields in the ALM project for the defect module for the integration, go to KM1352699.

> **Note:** The configuration information provided here is based on the out-of-the-box ALM - Request For Change workflow. If your organization has customized any out-of-the-box workflow, make sure you adjust the SQL script accordingly.

12. In Site Administration, deactivate the project, repair the project, and then activate the project.

13. Check in ALM groups and permissions if the fields are active.

14. Go to **Project Lists > Requirement Status** to check if any status is missing.

15. Add request statuses that PPM workflow (ALM - Request For Change) contains but ALM requirement does not.

    a. Log on to the integrated ALM project.

    b. From the menu bar, click **Tools > Customize > Project Lists > Requirement Status > New Item**.

    c. Set **Item Name** as the following PPM request status name, and add them one by one:

        - Build

        - Pending Approval

        - Approved

        - Test

        - Resource Assessment

        - Pending Rework

        - Run Impact Analysis

        - Impact Assessment

        - Ready for Review

        - 6-Running Sanity Tests in Quality Center

        - Backout

        - Pending Requestor Sign-off

        - Review Results Post Deployment

        - Closed - Successful

        - Closed – Failed

16. Return to the ALM project schema database. Update the TRAN_RULES table by running the following script.

```
update <database schema name of your QC/ALM project>.tran_rules

set tr_rules='New,1-Requirements Setup Completed' || chr(10) || '1-
Requirements Setup Completed,2-Test Plan Setup Completed' || chr(10) || '2-
Test Plan Setup Completed,3-Test Lab Setup Completed' || chr(10) || '4-
Running Tests in Quality Center,5-Test Execution Completed' || chr(10) ||
'3-Test Lab Setup Completed,Build' || chr(10) || 'Build,Pending Approval' ||
chr(10) || 'Pending Approval,Approved' || chr(10) || 'Approved,4-Running
Tests in Quality Center' || chr(10) || 'Approved,Test' || chr(10) || '4-
Running Tests in Quality Center,Pending Rework' || chr(10) || 'Test,Pending
Rework' || chr(10) || '4-Running Tests in Quality Center,Resource
Assessment' || chr(10) || 'Test,Resource Assessment' || chr(10) || 'Resource
Assessment,Run Impact Analysis' || chr(10) || 'Run Impact Analysis,Impact
Assessment' || chr(10) || 'Impact Assessment,Ready for Review' || chr(10) ||
'Ready for Review,Run Impact Analysis' || chr(10) || 'Ready for
Review,Pending Approval' || chr(10) || 'Approved,6-Running Sanity Tests in
Quality Center' || chr(10) || 'Pending Rework,Closed - Failed' || chr(10) ||
'Pending Rework,Build' || chr(10) || '6-Running Sanity Tests in Quality
Center,Closed - Failed' || chr(10) || 'Pending Rework,Test' || chr(10) ||
'6-Running Sanity Tests in Quality Center,Test' || chr(10) || 'Test,Pending
Requestor Sign-off' || chr(10) || 'Test,Backout' || chr(10) ||
'Backout,Pending Requestor Sign-off' || chr(10) || 'Build,Pending Requestor
Sign-off' || chr(10) || 'Pending Requestor Sign-off,Review Results Post
Deployment' || chr(10) || 'Review Results Post Deployment,Closed -
Successful' || chr(10) || 'Review Results Post Deployment,Closed - Failed'

WHERE tr_field_name = 'RQ_REQUEST_STATUS';

commit;
```

> **Note:** The configuration information provided here is based on the out-of-the-box ALM - Request For Change workflow. If your organization has customized any out-of-the-box workflow, make sure you adjust the SQL script accordingly.

17. In Site Administration, deactivate the project, repair the project, and then activate the project.

18. Check again for TRAN_RULES in the permissions of the PPM Request Status.

Task 3: Add scripts to perform the operation on ALM server side

> **Note:** The configuration information provided here is based on the out-of-the-box ALM - Request For Change workflow. If your organization has customized any out-of-the-box workflow, make sure you adjust the scripts accordingly.

1. Add scripts to perform the operation on ALM server side.

   a. Log on to ALM as administrator.

   b. From the menu bar, click **Tools > Customize > Workflow > Script Edit**.

      The Script Editor opens.

   c. From the navigation pane, select **Requirement module script**, then go to the last script of this category, copy and append the content of `add requirement script.txt` to the end of the script in the editing pane.

      > **Note:** To obtain the `add requirement script.txt`, go to KM00630244 and click **workflow scripts** to download the zip package.

   d. Click **Save**.

      The new script is added to the **Requirement module script**.

   e. From the navigation pane, select **Defect module script**, then go to the last script of this category, copy and append the content of `add defect script.txt` to the end of the scrip in the editing pane.

      > **Note:** To obtain the `add defect script.txt`, go to KM00630244 and click **workflow scripts** to download the zip package.

   f. Click **Save**.

      The new script is added to the **Defect module script**.

   g. Exit and log off.

2. Run the SQL scripts provided by HPE Software Support manually.

   For instructions on how to activate workflow script in ALM project, go to https://softwaresupport.hp.com/group/softwaresupport/search-result/-/facetsearch/document/KM1352700

Task 4: Add an integration configuration entry in PPM

1. Log on to PPM.

2. From the menu bar, click **Open > Administration > Integrations**.

   The Integration landing page opens to the Manage Application Change tab.

3. In the Server Configuration Details section, select **ALM – Request for Change (RFC)** from the drop-down list and click **Add Integration Configuration**.

> **Note:** The **Default QC/ALM Folder Name** is the folder that will store the requirements created with this integration.

4. Click **Next.**

5. Provide values for the fields as desired or leave as is until the wizard has finished.

Task 5: Create a request in PPM

1. Create a request in PPM.

> **Note:** The **QC/ALM Instance** and **QC/ALM Domain** fields are prepopulated. This also applies to the **QC/ALM Project** field if there IS only one project. If you have associated your Request Type with several projects, this field is not populated. Use a rule or populate the field manually.

2. Follow the workflow until the request is pushed to ALM.

3. Follow the request to closure.

Upgrading Integration with Quality Center to Integration with ALM

This section provides detailed instructions on upgrading your existing integration configurations with Quality Center version 10.00.

See "Upgrading from Integration with Quality Center 10.00 to Integration with ALM" on the next page for detailed instructions.

Quality Center version 9.2 is not officially supported since PPM version 9.20. If you still have PPM integration with Quality Center version 9.2, go to the following sections for more information:

- "Upgrading from Integration with Quality Center 9.20 to Integration with ALM" on page 213

- "Upgrading from Integration with Quality Center 9.20 to Integration with Quality Center 10.00" on page 212

Upgrading from Integration with Quality Center 10.00 to Integration with ALM

You can upgrade your PPM integration with Quality Center version 10.00 to integration with ALM version 11.00 or later. You can upgrade from any combination of QC 10.00 with PPM 9.1x/8.0x/7.x to the latest combination of PPM and ALM.

> **Note:** While you prepare to perform the upgrade process, make sure that:
>
> - Both your PPM Server and Quality Center server are running.
>
> - No users are accessing either PPM or Quality Center products.

The upgrade solution involves the following tasks:

- "Preparation: Workflow Checking" below

- "Task 1: Upgrading PPM to Version 9.40" on page 207

- "Task 2: Uninstalling Quality Center 10.00, Installing ALM 11.00 or Later, and Upgrading Quality Center Database" on page 208

- "Task 3: Upgrade QC Integration Configuration" on page 208

Preparation: Workflow Checking

HPE recommends you check, and modify if necessary, your PPM workflows against the Quality Center workflow before you upgrade the integration configuration.

 Figure 6-3. An example of PPM request workflow steps statuses matching QC requirement transition rules

PPM Request workflow

Logged

↓ Approved

Execution
Create requirement in QC

↓ Approved

New

↓ 1-Requirements Setup Completed

1-Requirements Setup Completed

↓ 2-Test Plan Setup Completed

2-Test Plan Setup Completed

↓ 3-Test Lab Setup Completed

3-Test Lab Setup Completed

↓ 4-Running Tests in Quality Center

4-Running Tests in Quality Center

↓ 5-Test Execution Completed

5-Test Execution Completed

↓ Closed

Closed

Legend

(Request Status Name) — PPM Requst Workflow Step Status Name

(Transition Name) — PPM Request Workflow Transition Name

(QC Transition Rule) — QC Requirement Transition Rule

QC Requirement Transition Rules

New

↓

1-Requirements Setup Completed

↓

2-Test Plan Setup Completed

↓

3-Test Lab Setup Completed

↓

4-Running Tests in Quality Center

↓

5-Test Execution Completed

↓

Closed

For an existing integration with Quality Center version 10.00, you can use the PPM-Quality Center Integration Tool to perform integration configuration. As a result, for example, the workflow related hardcoded configuration can overwrite the workflow script in Quality Center. Then, you can set the control mode to **PPM**, so that PPM request workflow execution drives the synchronization of PPM request status to Quality Center requirement status.

However, with ALM 11.x, due to its policy changes, you are no longer allowed to overwrite ALM's workflow script to enforce the PPM workflow execution.

To achieve bidirectional status synchronization between PPM request and QC defect or requirement, go to PPM Workbench to check and make sure that your PPM request workflow steps are exactly consistent with the transition rules defined in Quality Center.

For example, you have an integration configuration and you create a request in PPM to create a requirement in QC. The PPM request follows the workflow illustrated in the left part of Figure 6-3, and the requirement follows the default transition rules in QC (right part of Figure 6-3).

You need to ensure the following for status synchronization between PPM request and QC/ALM defect or requirement to work properly:

- For successful status synchronization from PPM to QC/ALM, the statuses and sequence of PPM request workflow steps shall be exactly the same as those of transition rules in QC/ALM.

  For example, in PPM request workflow of Figure 6-3, status of the step following the **Execution Create a requirement in QC** step shall be **New**, as the first requirement rule in QC is **New**. If you provide a different status description, such as **New1**, the status synchronization from PPM to QC would be lost.

  > **Note:** For a defect workflow, status of the step following the **Execution Create a defect in QC** step shall be **New**.

  For detailed descriptions and sequence of transition rules defined in QC/ALM, check the TRAN_ RULES table in the QC/ALM project schema database.

- For successful status synchronization from QC/ALM to PPM, in addition to ensuring that the statuses and sequence of PPM request workflow steps are exactly the same as corresponding transition rules in QC, you also need to ensure that in PPM request workflow, the description of a transition is exactly the same as the status of the subsequent request workflow step.

  If, for example, you provided a transition description that is different from its subsequent step status, you would lose synchronization of status from QC to PPM after upgrading your integration configuration to PPM Center 9.14 and ALM 11. However, you can still click **Save** to update other fields, such as Summary and Description, though with error message displayed.

> **Note:** For defect workflow, by default the setting in the TRAN_RULES table of QC is $ANY to $ANY. If there are custom rules instead, see the example on requirement in this section.

Note that if your workflow is the same as the example illustrated in Figure 6-3, HPE recommends you manually deploy the configuration file (ITGQCIntegration.xml) again by using the PPM-Quality Center Integration Tool. For detailed instructions, see "Deploying the Mapping File to PPM and Quality Center" on page 160.

If you are using customized workflow in QC, modify the PPM Center request workflow in the PPM Workbench before you perform the upgrade process.

For answers to any other potential questions you may have regarding the upgrade process, see "Frequently Asked Questions and Answers" on page 211.

For known problems and limitations, see the Release Notes.

Task 1: Upgrading PPM to Version 9.40

1. If you have not upgraded to PPM version 9.40, follow the supported upgrade paths and upgrade your PPM instance from an earlier version to 9.40.

    For supported upgrade paths and detailed upgrade instructions, see the *Upgrade Guide*.

2. Check whether all your QC integration configurations are listed in the Integration Configurations Summary page.

    a. Log on to PPM 9.40 as an administrator.

    b. From the menu bar, select **Open > Administration > Integrations.**

        The Integration Configurations Summary page displays.

    c. QC integration configurations are read-only, with an **Upgrade** icon available in the right end of the entries.

Task 2: Uninstalling Quality Center 10.00, Installing ALM 11.00 or Later, and Upgrading Quality Center Database

1. Stop Quality Center 10.00 service and uninstall Quality Center 10.00 with keeping all the QC DB schema. Make sure that you have noted down the Quality Center JBoss port number.

   For detailed instructions, see *Quality Center Installation Guide*.

2. Install ALM version 11.00 or later. Make sure that you upgrade QC database from the existing one. Also make sure that the QC JBoss port is using the same one as QC 10.00 settings.

   For detailed instructions, see the *Quality Center Installation Guide* and the *Application Lifecycle Management Installation Guide*.

3. Start ALM version 11.00 or later, and log in to Site Administration using DB administrator account.

4. From the navigation pane, locate the project for which you want to upgrade the database.

5. Right click the project and select **Deactivate Project** from the popup menu, then click **OK**.

6. Right click the project again, select **Maintain Project > Upgrade Project**.

   The Upgrade Project dialog opens.

7. Click **Upgrade Project**.

8. Click **Close** when the project is successfully upgraded.

9. Right click the project and select **Activate Project** from the popup menu, then click **OK**.

Task 3: Upgrade QC Integration Configuration

1. Go back to PPM, check that background service **QC Integration Migration Service** is up.

2. On the Integration Configurations Summary page, click the **Upgrade** icon for the QC integration configuration entry you want to upgrade.

   The QC username and password dialog opens.

3. Provide values for the **QC/ALM Username** and **QC/ALM Password** fields.

4. Click **Verify QC/ALM Connection**.

   Wait for the verification to complete, and click **OK** when the verification success prompt pops up.

5. Click **OK** in the QC/ALM username and password dialog.

   The Configure QC/ALM Integration for Request Type section for upgrade purpose appears.

   For sample tab pages for requirement request type or defect request type, see "Configuring Integration with ALM Version 11.00 and Later" on page 172.

6. Check and reconfigure field mappings if necessary.

   By default, the Configure QC/ALM Integration for Request Type section loads all settings from the old configuration automatically, including all field mappings.

   On the **Step 2: Map PPM—QC/ALM Fields** tab page, a Map Values button in the **Current Field Mappings** table indicates that this mapping contains mappings of multi-value fields or drop-down list fields. Click Map Values to open the Map Values dialog, check and make sure the mappings and directions between **QC/ALM Values** and the **PPM Values** are correct.

7. Click **OK** to save your changes and close the Map Values dialog.

8. Click **Next** on the **Step 2: Map PPM—QC/ALM Fields** tab page.

9. Click **Save** or **Save & Migrate**.

   If you click **Save & Migrate**, the Configure QC/ALM Integration for Request Type section jumps back to the **Step 1: Configure Server Details** tab page.

   Wait for the saving and migration process to complete. The **Migration Status** line disappears when all records are successfully migrated and all previously editable fields become read-only. In average it takes around one minute to migrate 1,000 records.

   When the **Migration Status** line disappears, move to the **Step 3: Complete Configuration** tab page, the **Cancel** button to the right end changes to **Done** (during the migration, the button changed to **Back**).

10. Click **Done**.

    The Integration Configurations page refreshes.

    Now the status for the integration configuration you just upgraded already changed. By default, the migrated integration is **Disabled**.

11. You can now click the **Disabled** icon 🗄 to enable the integration configuration that you just successfully migrated.

    > **Tip:** For a migrated integration configuration, clicking the **Delete** icon ✖ removes the new integration configuration only. The original integration configuration still exists.

12. After you have successfully upgraded all integration configurations with QC 10 to with ALM 11.x, make sure you comment out the following line from the `server.conf` file:

    `com.kintana.core.server.ENABLE_QUALITY_CENTER_INTEGRATION=true`

    The `ENABLE_QUALITY_CENTER_INTEGRATION` parameter is not needed anymore after you upgrade the integration configuration to ALM 11.x.

13. Go to PPM database, you should find some new records in the PPM_INT_QC_ENTITY_MAP table.

Best Practices on Mapping Quality Center Field "Detected By"

When you upgrade the integration from Quality Center version 10 to ALM version 11.00 or later, keeping the same configuration mapping settings, you may encounter an error with the QC/ALM field **Detected By**. In this case, you may try one of the following approaches for mapping the QC/ALM field **Detected By**:

- Mapping QC/ALM field **Detected By** with PPM request field **Created By**. When PPM user list and QC/ALM user list are the same, this approach would be the best choice.

- Mapping QC/ALM field **Detected By** with PPM custom field (text field). This approach would require you to enter correct information into PPM custom field.

  For example, you create a text type field in PPM and map it to QC/ALM field **Detected By**. If you set the control mode to **QC**, then synchronization populates the new PPM field automatically with populated the **Detected By** field with the QC Username field.

- Mapping QC/ALM field **Detected By** with PPM custom field (using validation **QC/ALM Defect Assigned User**). If you want to get user list from QC/ALM directly, this approach would be the best choice.

If you leave this mapping field blank when creating a PPM request in order to create a defect in QC/ALM, you can set the control mode to **QC**, then the system populates **Detected By** field in QC/ALM with the value you specified in **QC/ALM Username** field on the integration configuration page.

Frequently Asked Questions and Answers

Q: Does the latest integration of PPM Center 9.40 and ALM 11.x support ALM Request Types and Workflows?

A: If you integrated QC 10 with PPM 8.0x/9.10 using ALM request types and workflows, you can keep using the same request types and workflows after upgrading to integration of PPM 9.20 and ALM 11.x. However, you may want to check the workflows again as ALM changed its rules. For details, see "Preparation: Workflow Checking" on page 204.

Q: Does the latest integration of PPM Center 9.40 and ALM 11.x support the existing ALM bundle?

A: If you already deployed the ALM bundle for QC/ALM integration with a previous version of PPM, there is no need to deploy the new ALM bundle after upgrading PPM to version 9.20.

Q: Does the integration with ALM 11.x support version controlled ALM projects?

A: Yes.

Q: Does the upgrade process update records on both PPM and QC/ALM products?

A: The upgrade process does not update records on either of the PPM and QC/ALM products, it only tries to update the integration configuration data in PPM.

Q: Is the Integration Tool for QC 10 still necessary?

A: No. When you perform the upgrade process, it loads all settings of PPM integration with a QC 10. This means that after upgrade, the integration tool for configuring PPM integration with QC 10 is no longer necessary. You can now modify any integration configuration settings directly from PPM.

If you integrate PPM with ALM version 11.50 (or later), you can run the PPM-ALM Integration Tool available from the
Integration Configurations page to enable PPM related fields in ALM project.

Q: Does upgrading integration configuration (with QC 10) also upgrade the field groups used for the integration and their field names?

A: No. Although the Quality Center Defect Information field group and the Quality Center Info field group are renamed to QC/ALM Defect Information and QC/ALM Info respectively in PPM version 9.20, upgrading an existing integration configuration does not upgrade the names of the field groups and their field names. They still remain the same as before.

The new field groups and their fields apply to new integration configurations you add in PPM version 9.20 only.

Q: How should I manually activate PPM fields in ALM project and workflow script in ALM?

A: With ALM version 11.00 or 11.20, you need to manually activate PPM fields in ALM project and workflow script in ALM. With ALM 11.50 or 12.00, no need to do so manually.

- For ALM version 11.00 or 11.20

  With PPM 9.12 (or later) and ALM 11.00 (or later), when you set up a new integration between PPM request and ALM project, you need to manually activate PPM* fields and add workflow script in ALM project in order to make integration work. This is still necessary with PPM 9.40 and ALM version 11.00 or 11.20. Even if you have upgraded from QC 10, you still need to execute the scripts manually in the database for ALM version 11.00 or 11.20.

  For instructions on how to enable PPM* fields in ALM project, go to KM1352699.

  For instructions on how to activate workflow script in ALM project, go to KM1352700.

- For ALM version 11.50

  If you integrate PPM with ALM version 11.50 (or later), you can run the PPM-ALM Integration Tool available from the Integration Configurations page to enable PPM* fields in ALM project. For details, see "Configuring an ALM Project for the Integration (for ALM 11.50 or Later)" on page 180.

Q: Before upgrade, I had already activated PPM fields in Quality Center 10 using the PPM Center-Quality Center Integration Tool. Then I upgraded from QC 10 instance to ALM 11.50 instance. Can I use the PPM Center-ALM Integration Tool to activate the PPM fields in ALM again?

A: No. After upgrading from QC 10 instance to ALM 11.50 instance, the PPM Center fields related to the integration that you enabled in QC 10 using the PPM Center-Quality Center Integration Tool are rolled back to their default values. You MUST run the scripts manually by following the instructions provided in KM1352699.

If using the PPM Center-ALM Integration Tool to do so, requirements related fields may fail or not work.

Use the PPM Center-ALM Integration Tool for newly installed instances of ALM version 11.50 or later only.

Upgrading from Integration with Quality Center 9.20 to Integration with Quality Center 10.00

To use your existing integration configurations with Quality Center version 9.20 in PPM version 9.20, upgrade your integration configurations by doing the following,

1. Stop PPM Servers.

2. Uninstall Quality Center version 9.20, install Quality Center 10.00, and upgrade Quality Center database.

   For detailed instructions, see *Quality Center Installation Guide*.

3. If you have not upgraded to PPM version 9.40, follow the supported upgrade paths and upgrade your PPM instance from an earlier version to 9.40.

4. Back up the XML mapping file (`ITGQCIntegration.xml`).

5. Upgrade the PPM-Quality Center Integration Tool by following the steps below:

   a. Uninstall the existing PPM-Quality Center Integration Tool.

   b. Download the new version of the PPM-ALM Integration Tool for QC 10.00 from the PPM Server file system (*<PPM_HOME>*`/integration/mac/ppmqcintegrationtool/setup.exe`).

   c. Install the new version PPM-ALM Integration Tool.

6. Open the XML mapping file (`ITGQCIntegration.xml`) you backed up earlier using the new PPM-Quality Center Integration Tool, and deploy the file to both PPM Server and Quality Center 10.00 server.

7. Make sure the `com.kintana.core.server.ENABLE_QUALITY_CENTER_INTEGRATION=true` setting is still present in the `server.conf` file.

8. Restart PPM Servers.

Upgrading from Integration with Quality Center 9.20 to Integration with ALM

To upgrade your existing integration configurations with Quality Center version 9.20 to integration with ALM, do the following:

1. Stop PPM Server.

2. Back up the XML mapping file (`ITGQCIntegration.xml`).

3. Uninstall Quality Center version 9.20, install ALM 11.00 or later, and upgrade Quality Center database.

   For detailed instructions, see the *ALM Installation Guide*.

4. If you have not upgraded to PPM version 9.40, follow the supported upgrade paths and upgrade your PPM instance from an earlier version to 9.40.

5. Copy and paste the XML mapping file (`ITGQCIntegration.xml`) you backed up earlier to the `<PPM_Home>`/`conf`/ directory.

6. Follow the instructions in "Upgrading from Integration with Quality Center 10.00 to Integration with ALM" on page 204 to upgrade the integration configurations.

Importing or Exporting an Integration Configuration

For integration with ALM, it is now possible for administrators to import or export an integration configuration using the `integrationMACConfigurations` REST API introduced with PPM version 9.20. The exported integration configuration can be used for maintaining or migrating the integration configurations. The imported integration configuration can be used to create a new integration configuration or update existing integration configurations.

To import or export an integration configuration,

1. Open the `<PPM_Home>`\kintana\deploy\itg.war\WEB-INF\conf\wink_ws.app file and make sure the following line is not commented out:

   `com.mercury.itg.rest.integration.resource.IntegrationMACConfigurationResource`

2. Access the REST API: `http://<host>:<port>/itg/rest/integrationMACConfigurations` and do the following as necessary:

   ○ To export integration configurations, use `get` as http method.
   Note that the password is encrypted.

   ○ To create an integration configuration, use `post` as http method, `application/xml` or `application/json` as Content-Type. If you want to use a browser to do this, you can use RestClient on firefox. Make sure that `configurationKey` is empty in the content.

   ○ To update an integration configuration, use `post` as http method, `application/xml` or `application/json` as Content-Type. If you want to use a browser to do this, you can use RestClient on firefox. Make sure `configurationKey` in the content is the key of configuration.

For more details about the `integrationMACConfigurations` REST API, see the *RESTful Web Services Guide*.

Using the Integration of PPM with Quality Center/ALM

As necessary, you use the provided ALM request types and workflows as templates to create your own PPM request types and workflows enabled for integration of PPM with Quality Center.

As provided by HPE, the only PPM request type with a request header type that includes the QC/ALM Defect Information field group, and thus the only request type that is available to map to a QC/ALM defect is the ALM - Defect Template with Quality Center Integration request type.

As provided by HPE, the only PPM request types with a request header type that includes the QC/ALM Info field group, and thus the only request types that are available to map to a QC/ALM requirement, are the following:

- ALM - Release Management

- ALM - Request for Change (RFC)

This section describes how the request types and workflows provided in ALM support integration of PPM with QC/ALM, and this section provides guidelines for modifying those request types and workflows.

Steps in PPM Workflows that Involve Integration with QC/ALM

Several workflows and subworkflows provided in ALM software have steps that are related to the integration of PPM with Quality Center, as follows:

- The ALM - Defect Template with Quality Center Integration workflow generates a defect or requirement in QC/ALM as soon as a request of the associated type is created. See "ALM - Defect Template with Quality Center Integration Workflow" on page 141.

- The ALM - Request For Change workflow (see "ALM - Request For Change Workflow" on page 43) calls the following subworkflows at the indicated steps:

  - At step 17, the ALM - Request For Change workflow calls the Plan Tests Sub WF (see "ALM - Plan Tests Sub WF Subworkflow" on page 53).

In this subworkflow, step 1, Quality Process Mode, determines whether the test planning will be done manually or using integration of PPM with QC/ALM. If integration is to be used, the following additional subworkflow steps relate to the integration:

- Step 4, Quality Process Entry

- Step 5, Generate Requirement in Quality Center

- Step 6. Quality Center Failure

- Step 7, Quality Center Requirements Setup

- Step 8, Quality Center Test Plan Setup

- Step 9, Quality Center Test Lab Setup

  ○ At step 22, the ALM - Request For Change workflow calls the Deploy and Test Changes Sub WF (see "ALM - Deploy and Test Changes Sub WF Subworkflow" on page 56).

  - Step 6. Quality Center Failure

  - Step 7, Quality Center Requirements Setup

  - Step 8, Quality Center Test Plan Setup

  - Step 9, Quality Center Test Lab Setup

  ○ At step 22, the ALM - Request For Change workflow calls the Deploy and Test Changes Sub WF (see "ALM - Deploy and Test Changes Sub WF Subworkflow" on page 56).

  In this subworkflow, step 3, Quality Process Mode, determines whether the testing will be done manually or using integration of PPM with QC/ALM. If integration is to be used, the following additional subworkflow step relates to the integration:

  - Step 5, Quality Center Test Execution

- The ALM - Release Request workflow (see "ALM - Release Request Workflow" on page 81), includes the following steps that relate to integration of PPM with QC/ALM for the release management process:

  ○ Step 3, Integrate with Quality Center?

  ○ Step 4, Quality Process Entry

  ○ Step 5, Create Release Requirement in Quality Center

  ○ Step 6, Quality Center Failure

Configuring Request Types and Workflows for the Integration

This section provides guidelines on how to build request types and workflows that support the integration of PPM with QC/ALM.

To enable integration between PPM requests and QC/ALM projects, you must ensure that the request types and projects have the necessary matching fields and that the workflows use steps that support integration.

ALM bundles provide request types and workflows that use QC/ALM capabilities, and the workflows include the required steps. You can create integration-enabled request types and workflows in any of the following ways:

- By using the PPM Workbench to copy the provided ALM request types and workflows, which already contain the request type fields and workflow special commands required for integration, and changing the copies as needed. This is often the easiest approach.

- By creating new request types and workflows that incorporate the fields and special commands required for integration.

- By customizing your existing PPM request types and workflows by adding the fields and special commands required for integration.

After you configure the required request types and workflows, you map the PPM fields and their valid values to the QC/ALM fields and their valid values.

For detailed information about configuring request types and workflows, see the Demand Management Configuration Guide as necessary.

Configuring Request Types

The guidelines to configure a request type for integration are as follows:

- Make sure the request header type for the request type to be associated with a QC/ALM defect includes the QC/ALM Defect Information field group. Only request types with this field group can be mapped to QC/ALM defects.

> **Note:** By default, the QC/ALM Defect Information field group is included in only the ALM - Defect Template with Quality Center Integration request header type.

Make sure the request header type for the request type to be associated with a QC/ALM requirement includes the QC/ALM Info field group. Only request types with this field group can be mapped to QC/ALM requirements. See "Request Header Types " on page 139.

> **Note:** By default, the QC/ALM Info field group is included in only the ALM - Request for Change (RFC) request header type and the ALM - Release Management request header type.

- Decide which request type to map to each QC/ALM project, then make sure that the request type and project have the required mapping of associated fields.

- Make sure that each pair of mapped fields includes the required valid values. For example, if a QC/ALM field contains a lookup list, make sure that the associated field in the PPM request accepts the mapped values. If you update a field in one application with a value that does not have a valid mapped value in the other application, the field in that other application will not be updated.

  For details when integrating with Quality Center version 10.00, see "Resolving Lists of Valid Values" on page 156.

> **Caution:** QC/ALM workflows can limit the changes a user can make to a field. For example, a script might specify that at a certain point in the QC/ALM workflow, the user cannot change the status from Open to Closed.
>
> For integration with QC/ALM with a valid field mapping, if a PPM field is updated, the associated QC/ALM field is updated even if a QC/ALM workflow script specifies the update as invalid.

Configuring Workflows

The guidelines to configure a workflow for integration are as follows (see the *Demand Management Configuration Guide* for details as necessary):

- Make sure the workflow includes execution steps and decision steps that enable the integration of PPM with QC/ALM. The workflows provided by ALM bundles include such steps.

> **Note:** A PPM decision step that depends on QC/ALM (that is, a PPM request that is supposed to be updated by a QC/ALM status change) can have its status changed in QC/ALM only by a user who has Administrator rights.

- If you need to customize a workflow to create a defect or a requirement, HPE recommends using the execution steps that are included in the ALM - Defect Template with Quality Center Integration workflow, instead of building the steps yourself.

  To create defects or requirements in QC/ALM, your workflow must include one of the following execution steps that uses the stated special command:

  - To create defects, the ALM - Create QC Defect execution step with the ksc_create_defect_in_ QC special command

  - To create requirements, the ALM - Create QC Requirement execution step with the ksc_create_ requirement_in_QC special command

  In the **Properties** tab of the workflow execution step, specify a value in the Request Status field that is valid for QC/ALM, for example, 1-Requirements Setup Completed when creating a requirement.

  Once an execution step has created a requirement or defect in QC/ALM, then every time the PPM request status changes, the QC/ALM requirement or defect status also changes if the same PPM status exists for the QC/ALM requirement or defect. For example, if the PPM request status changes to Open, the QC/ALM requirement or defect status will also change to Open, as long as Open is one of the values allowed in the status field of the requirement or defect.

  For more information about configuring workflow steps, see the *Demand Management Configuration Guide*.

  > **Note:** After QC/ALM sends an update to PPM, QC/ALM waits for a response, and the QC/ALM record remains locked until QC/ALM receives the response. Meanwhile, if PPM advances to the next workflow step and attempts, for example, to update (synchronize) QC/ALM with a new request status, QC/ALM rejects the update since the record is locked.
  >
  > Therefore, a PPM workflow should not contain successive steps such that the first causes a PPM request to advance based on a change in QC/ALM status, and the second causes PPM to attempt to update QC/ALM. Make sure there is an intervening step between two such steps.

- When a PPM request is integrated with a QC/ALM project, you can use a change in the QC/ALM status to cause the PPM request to advance through an active decision step to the next step in the associated PPM workflow.

  For example, when the QA manager sets the status of a QC/ALM project to indicate that test planning is complete, the associated request in PPM can automatically advance from the step in the PPM workflow that is awaiting that notification.

Conversely, whenever the status of a PPM request changes, PPM notifies QC/ALM, and (assuming the new status is valid in QC/ALM) QC/ALM users can take appropriate action such as starting tests.

To enable this functionality, you must design the workflow decision steps such that the following three items have the same values:

○ QC/ALM status that will trigger the advancement in the PPM workflow.

○ Transition name (which is specified as the **Meaning** field of the validation value for the workflow step source) for the active decision step in the PPM workflow.

  HPE recommends that you give the **Meaning** field of the transition a value that is unique to this transition, that is, a value that does not exist anywhere else in the workflow. When this value becomes assigned to the **ITG Status** field in Quality Center version 10.00 or the **PPM Status** field in ALM version 11.00 or later, the PPM workflow advances if the value matches a valid transition in an active workflow step. If the workflow has more than one active step and the **Meaning** is not unique, the workflow could advance to an unintended step.

○ **Request Status** field in the destination step in the PPM workflow.

If the QC/ALM status does not appear in the list of valid request status values in PPM, PPM sends an error message to QC/ALM, the QC/ALM status reverts to its previous value, and the PPM workflow does not advance.

For example, in the following portion of the ALM - Plan Tests Sub WF subworkflow, the transition from step 7 to step 8 is called **1-Requirements Setup Completed**.

If you double-click step 8 (the destination step), the **Properties** tab of the Workflow Step window shows that the value in the **Request Status** field is also **1-Requirements Setup Completed**.

All three items—the QC/ALM status, the transition, and the **Request Status** field of the destination step—have the same value. Therefore, if the QA team changes the QC/ALM status to **1-Requirements Setup Completed** when step 7 is active, the workflow will advance to step 8.

If you need to change the value of the **Request Status** field of the destination step, in the **Properties** tab of the Workflow Step window for that step, specify the new value in the **Request Status** field and click OK.

If you need to change the **Meaning** field that defines the transition name, right-click the preceding decision step and select **Edit Source**; in the **Validation** section of the Decision window, click **Open**; click the validation value (row) of interest and click **Edit**; specify the new value in the **Meaning** field; click **OK** to close all open windows. For more detailed information, see the *Demand Management Configuration Guide*.

Synchronization

This section provides details regarding synchronization:

- Request Hierarchy Synchronization

- Disable Status Synchronization and Enable Status Field Mapping

- Synchronize Memo Fields from ALM Defects or Requirements as HTML Reference Attachments to PPM Requests

Request Hierarchy Synchronization

Note: For information about enabling request hierarchy synchronization with requirements in Quality Center**version 10.00:**

- If a mapping has not been created between the Quality Center requirement and the PPM request type, see "Enabling a Quality Center Project for the Integration" on page 151.

- If a mapping has been created, see "Enabling and Disabling Request Hierarchy Synchronization" on page 165.

For information about enabling request hierarchy synchronization with requirements in ALM **version 11.00 or later,** see "Options for Integration with an ALM Requirement" on page 193.

The integration of PPM with QC/ALM allows you to synchronize the hierarchies of requests in PPM and requirements in QC/ALM, that is, to make the QC/ALM requirement hierarchy match the PPM request hierarchy automatically, as in the following example sequence:

1. A PPM request named Request A is created.

2. With integration, an associated Requirement A is automatically created in QC/ALM.

3. A PPM request named Request B is created with a reference to Request A indicating that Request A is the parent of Request B.

4. An associated Requirement B is automatically created in QC/ALM. If request hierarchy synchronization is enabled, since Request B is the child of Request A, Requirement B is automatically created as the child of Requirement A.

   > **Note:** If Requirement A does not exist in QC/ALM, creating a reference in PPM from Request B to Request A has no effect in QC/ALM.

If you later delete the relationship (the reference) between Request A and Request B in PPM, the relationship between Requirement A and Requirement B is automatically deleted in QC/ALM. For integration with ALM version 11.00 or later, instead of being a child of Requirement A, Requirement B is moved to the folder specified in the **Default QC/ALM Folder Name** field.

When request hierarchy synchronization is first enabled, it does not cause immediate reorganization of existing requirements in QC/ALM to retroactively match the hierarchy of associated requests. Thereafter, however:

- Any update to any field in an existing integrated request initiates an update to the hierarchy (folder structure) of the associated requirement to reflect the hierarchy of the request.

- Upon creation of any new PPM request, the integration creates a new QC/ALM requirement in a matching hierarchy.

- Any changes you make to the hierarchy in PPM are automatically reflected in the hierarchy of the QC/ALM requirements. Thus, changes made in QC/ALM to the hierarchy of a requirement can be overridden later by updates to the associated PPM request.

PPM allows a request to have multiple parents, but QC/ALM limits a requirement to only one parent. If a PPM request has multiple parents, QC/ALM does not duplicate that hierarchy in the project, and PPM displays a message indicating a problem with hierarchy synchronization.

Example of Request Hierarchy Synchronization

Each requirement is mapped to a PPM request of type ALM - Request for Change (RFC). The request numbers in PPM are shown in the **ITG Request Id** column (for Quality Center version 10.00) or **PPM Request Id** column (for ALM version 11.00 or later). When the requests were created, their **RFC Summary** fields were specified as **Parent Requirement**, **Child Requirement**, and **Grand-child**

Requirement to indicate their intended hierarchy (first in PPM and then automatically as requirements in QC/ALM) after establishing the relationships among the requests in PPM.

When you open a request, you can add a different request in the **References** section and specify the relationship of the reference request to the request you opened. For example, the reference request can be a child or a parent of the open request.

For the example, you could use any one of the following methods to establish the relationships among requests, which would then automatically synchronize the relationships among the associated QC/ALM requirements:

- Open request 30512 and make it the parent of request 30514, and then open request 30514 and make it the parent of request 30515.

- Open request 30514 and make it the child of request 30512, and then open request 30515 and make it the child of request 30514.

- Open only request 30514 and make it both the child of request 30512 and the parent of request 30515. This method is slightly quicker and is used in the following procedure.

How to create the relationships between the requests in PPM

1. Select and open the request for which you want to define one or more relationships.

   In the example, open request 30514.

2. In the **Reference Additions** section of the request, in the **New Reference** drop-down list, select **Request (Existing)**.

3. Click **Add.**

   The Add Reference: Request window opens.

4. Specify data about the request to be related to the open request, and click **Search**.

   In this example, type `30512` in the **Request #** field and click **Search**.

   If the search is successful, the following window opens.

The window allows you to select from the search results which requests will be references (in the example, 30512 is the only request found using the specified search criteria), and to select an option to specify the relationship the reference request (or requests) will have to the open request (30514 in the example).

5. In the **Request Search Results** section, select the check box for the request that is to be made a reference.

6. In the upper section of the window, select the option for the relationship you want this reference request to have to the open request.

   In this example, select **Parent of this Request** because you want reference request 30512 to be the parent of open request 30514.

7. Click **Add**.

   The reference request and its relationship are added to the **References to be added on Save** text box in the open request.

8. Repeat step 2 through step 7 (or step 3 through step 7) as necessary to create parent and child relationships with additional requests.
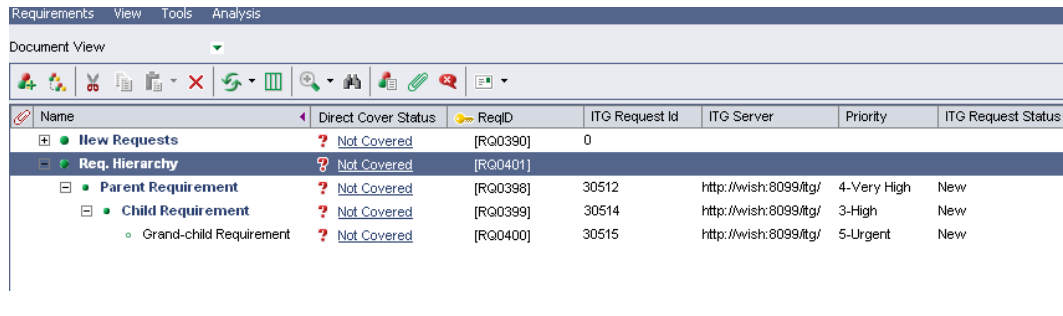
   In this example, follow step 3 through step 7, but in step 4 type 30004 in the **Request #** field to make that request the one to reference, and in step 6 select **Child of this Request** because you want reference request 30004 to be the child of open request 30514.

9. Click **Save**.

   The reference requests with which you have defined relationships are listed in the **References** section, **Requests** subsection of the open request.

In Quality Center, the requirements (rows) are now reorganized and indented to reflect the relationship hierarchy you specified among the PPM requests.

> **Note:** Field names **ITG Request Id, ITG Server,** and **ITG Request Status** are used for Quality Center version 10.00. In ALM version 11.00 or later, these field names are **PPM Request Id, PPM Server,** and **PPM Request Status.**

Disable Status Synchronization and Enable Status Field Mapping

You can disable the PPM defect workflow driven synchronization between PPM request status and ALM entity status by setting the server configuration parameter `REQUEST_LINK_ALM_ENTITY_STATUS` to `false` from the Administration Console. As a result, the **Status** field is available in the **QC/ALM Entity Fields** list in PPM for defect status mapping purpose, allowing you to view the real time status of an ALM entity in PPM.

This feature applies to PPM integration with ALM version 11.00 or later.

However, note that disabling the PPM defect workflow driven status synchronization means that PPM defect request status is no longer synchronized to ALM. Therefore, the status field mapping is controlled by **PPM** by default.

To disable status synchronization and configure the **Status** field mapping for a selected integration configuration,

1. Set the `REQUEST_LINK_ALM_ENTITY_STATUS` to `false` from the Administration Console.

   a. Log on to PPM.

   b. From the menu bar, click **Open > Administration > Open Administration Console**.

   c. In the navigation pane, expand **Administration Task > Application Configuration**.

d. In the Edit PPM Application Parameters pane, search for the `REQUEST_LINK_ALM_ENTITY_ STATUS` parameter.

e. Edit the **Value** for the parameter to `false`.

f. Click **Save**.

2. Configure the **Status** field mapping for a selected integration configuration.

a. From the menu bar, click **Open > Administration > Integrations**.

b. On the Manage Application Change page, from the **Integration Configurations** list, select an entry for which you want to configure the **Status** field mapping.

c. In the Server Configuration Details section, move to **Step 2: Map PPM-QC/ALM Fields** tab page.

The **Status** field is now available in the drop-down list of **QC/ALM Entity Fields** in the Field Mapping section.

d. Select **Status** from the **QC/ALM Entity Fields**, and then select **Request Status** from the **PPM Request Fields** list, and then click **Map Fields.**

The field mapping is added to the Current Field Mappings table.

e. Save the change to this integration configuration.

Synchronize Memo Fields from ALM Defects or Requirements as HTML Reference Attachments to PPM Requests

Starting from version 9.22, for any new defect or requirement that is created in ALM version 11.00 (or later), each of the memo fields (that are mapped to PPM request text fields on the Manage Application Change integration landing page in PPM) in the defect or requirement will be saved as an html reference attachment to the PPM request after you perform ALM to PPM synchronization. You can find all html attachments in the References section of a PPM request.

The HTML file name follows this format: *<ALM_Entity_Field_Name>*.html, and in the description of the html file, a label in the format of "The content of memo field '*<Lable Name>*' in ALM/QC entity". For example, description.html.

	Name	Size	Author	Added By	Date
− Attached Documents					
Details	description.html The content of memo field "Description" in ALM/QC entity	118 B	-	Admin User	10/22/13
Details	user-25.html The content of memo field "NOTE1" in ALM/QC entity	2 KB	-	Admin User	10/22/13

If a memo field is changed or updated in ALM, the html attachment file to PPM request is updated after next synchronization.

This ensures that a full version of the content from each of the ALM defect or requirement memo fields is available for users to view when the corresponding PPM request text fields contain confusing html tags after synchronization. This is especially helpful when ALM defect or requirement memo fields contain 4000+ bytes, because no text will be truncated in the html attachment files after synchronization. All content of memo fields will be synchronized as HTML attachments, and any update from ALM entities will also update the PPM request attachment files after next synchronization.

For existing PPM requests that contain 4000+ bytes in any of the mapped text fields, after you upgrade PPM to version 9.22 or later, when you perform ALM to PPM synchronization, the system compares the content contained in the PPM request text field against the first 4000 bytes of the content in the ALM memo field (stored in ALM database) to see if any update has been made. If yes, an html reference attachment file will be generated for each of the ALM defect or requirement memo fields and synchronized to the PPM request; otherwise no html attachment file will be generated.

Note that the content in mapped PPM request text fields remain the same after you ugrade PPM to version 9.22 or later.

Chapter 8: Integrating PPM Projects with HPE ALM Releases — View Project Quality

This section contains the following topics:

Introduction to the View Project Quality Integration Solution

The View Project Quality integration solution integrates PPM with the Releases module of the Application Lifecycle Management product. The integration allows project management officers, project managers, development managers, and QA managers to have visibility into quality KPIs and ALM scorecards from PPM Center projects, and allows them to optimize the management of project quality by:

- Introducing and enforcing consistent workflows for all major application delivery processes, and,

- Initiating and managing both application and testing projects and ALM Releases.

This integration solution is easy to configure and requires no additional installations.

The integration solution information is stored in the PPM_INT_SOLUTIONS_NLS table, and all ALM server information related to the integration configurations are stored in the PPM_INT_CONFIGURATIONS table. Mapping information about PPM projects and ALM releases are stored in the related requests. ALM quality statistics information are stored in the PPM_INT_QC_KPI table. To view error logs, see the PPM_INT_EVENTS table.

When configuring the integration solution, consider the following:

- The integration solution supports one-one mapping between a PPM project to an ALM release only.

- This integration supports OOTB quality metrics only. Customizable reports and metrics are not

supported yet.

- There is no cleanup service to clean all the outdated data in the KPIs table.

- You can not delete any ALM server configuration.

For information about the ALM versions supported for integration, see the *System Requirements and Compatibility Matrix*.

> **Note:** No software needs to be installed on the ALM server to integrate PPM and ALM. However, see the *System Requirements and Compatibility Matrix*.

For more information about ALM, see its product documentation at the Web site described in "Available PPM Integrations " on page 18.

Entities Introduced with the Integration

The following entities are introduced in PPM version 9.20 for the View Project Quality integration solution:

- The View Project Quality configuration page

  This page allows administrators to register ALM instances for PPM to retrieve KPI information and quality matrices from ALM and to display the data on the Project Overview page in PPM.

  To access the View Project Quality configuration page, from PPM menu bar, click **Open > Administration > Integrations**, and then click the **View Project Quality** tab from the navigation pane.

- QC/ALM Release Information field group

  The QC/ALM Release Information field group can be added to new or existing request types for creating or modifying projects that are to be linked to ALM releases.

  The field group contains the following fields:

Field	Description
QC/ALM Server	Retrieved from the ALM server registered on the View Project Quality configuration page.
QC/ALM Domain	Retrieved from the ALM server registered on the View Project Quality configuration page.

Field	Description
QC/ALM Project	Retrieved from the registered ALM Server.
QC/ALM Release	Retrieved from the ALM Project.
SCORECARD URL	To display ALM scorecard on the Project Overview page in PPM, administrators need to define the scorecard as an analysis item in ALM so that project managers can paste the link into this field. For detailed information about defining the scorecard as an analysis item in ALM, see ALM documentation.

- QC Integration Sync KPI Service

  The QC Integration Sync KPI Service synchronizes quality KPI data from ALM to PPM. The default synchronization interval is 24 hours. You can set the synchronization interval on the Schedule Services page.

  The QC Integration Sync KPI Service gets all valid ALM server configurations and loops through all configurations to retrieve KPI information from ALM and stores them in PPM database.

  If the service found that the quality KPI for current day already exists, it skips the project and continues to process next project.

  Only projects that are NOT cancelled, failed, closed, or complete will be synchronized.

  For details about KPI data to be retrieved from ALM, see "KPI Data Retrieved from ALM" on page 243.

- Four default portlets included in the Project Overview page

  After installation of PPM version 9.40 for newly created project types, the following new portlets are included in the Project Overview page by default and are displayed when there is data in the portlets.

Portlet	Description
Requirement Trends	Displays requirement status trend in the last 10 days. The requirement status includes requirement coverage.
Open Defect Trends	Displays open defects trend in the last 10 days. The open defects include open defects number and total defects number.
Priority Defect Trends	Displays trend of defects of higher severity in the last 10 days. The higher severity defects include defects of priority S1 and S2.

Portlet	Description
Project Quality Scorecard	Displays ALM release scorecard information. The scorecard portlet is displayed only when the ALM scorecard information is configured in the project request field of the Project Planning and Tracking (PPT) module in ALM.

However, if you create a project type by copying from an existing project type created in an earlier version of PPM, you need to manually add the new portlets to the Project Overview page on the Project Overview Layout policy page. For more information, see "(Optional) Configure Project Overview Layout" on page 237.

User Flow

The typical flow of this integration is as follows:

Step 1: A PPM administrator configures the View Project Quality integration solution

Before project managers can use the integration, PPM administrator needs to configure the View Project Quality integration solution, including:

- Identifying whether there is business needs for integration with ALM and configure Project Types accordingly

- Registering ALM instances

- Enabling and scheduling the QC Integration Sync KPI Service

Step 2: A project manager initiates a project in PPM by using a project type that contains the QC/ALM Release Information field section, and provides deliverables to all the assigned teams.

For details on creating a project in PPM, see the Project Management User's Guide.

Step 3: Based on the deliverables, business analysts, development managers, and QA managers build milestones using Project Planning and Tracking (PPT) in ALM.

Step 4: A release manager creates a corresponding release in ALM and defines key performance indicators (KPIs) for milestones and releases.

Step 5: The project manager updates the PPM project by linking it to the corresponding ALM release on the Project Details page.

For details on linking to an ALM instance, see "Linking a PPM Project to an ALM Release" on page 244.

Step 6: ALM tracks release progress against specific milestones, with KPI matrices passed back to PPM to be included in the overall project health status.

Step 7: Project managers and project participants view and monitor quality matrices on the Project Overview page.

For more information, see "Viewing and Monitoring Project Quality Information" on page 245.

Configuring the Integration Solution

PPM administrators need to identify whether there is business needs for any projects for integrating PPM with ALM, so that project quality related information can be retrieved from ALM for project stakeholders and participants.

Then, PPM administrators need to perform the following configuration tasks before end users are able to view and manage project quality related information from ALM:

- "Configuring Project Types" below

- "Registering ALM Instances" on page 238

- "Enabling and Scheduling the QC Integration Sync KPI Service" on page 239

Configuring Project Types

Configuring a project type involves adding the QC/ALM Release Information field group to the request header type associated with the project type associated request type.

For existing users, if you would like to include the new portlets on the Project Overview page of existing projects, you can configure the existing project types. See "Configuring Existing Project Types" below.

For new users, you can create new project types. HPE strongly recommends you to create your own project types with customized project type associated request types, instead of using the system default request types directly. See "Creating New Project Types with Customized Project Type Associated Request Types" on page 235.

Configuring Existing Project Types

1. Select a project type associated request type.

   a. Log on to PPM.

   b. From the menu bar, select **Search > Administrative > Project Types**.

The Manage Project Types page opens.

    c. Select and click a desired project type from the list.

      The Manage Project Type: <Project Type Name> page opens.

    d. Click the **Request Types** policy tab.

      The Request Types policy page opens.

    e. Note down the value displayed in the **Project Process Request Type** field. For example, **Project Details**.

2. Add the QC/ALM Release Information field group to the request header type associated with the request type.

    a. Log on to PPM.

    b. From the menu bar, select **Open > Administration > Open Workbench.**

      The PPM Workbench opens.

    c. From the shortcut bar, select **Demand Mgmt > Request Types.**

      The Request Type Workbench opens.

    d. In the Request Type field, type the request type you noted down in "Configuring Existing Project Types" on the previous page of "Creating New Project Types with Customized Project Type Associated Request Types" on the next page and click **List**. For example, type `Project Details`.

    e. Open the request type displayed on the Results tab. For example, open the **Project Details** request type.

      The Request Type : <Request Type Name> dialog opens.

    f. Click **Open** for the Request Header Type field.

      The Request Header Type dialog opens.

    g. Click **Field Groups**.

      The Field Groups dialog opens.

    h. Select the checkbox for the QC/ALM Release Information field group, and click **OK**.

      The QC/ALM Release Information field group is added to the request header type.

    i. Click **OK**.

Creating New Project Types with Customized Project Type Associated Request Types

1. Select a project type associated request type.

   a. Log on to PPM.

   b. From the menu bar, select **Create > Administrative > Project Type**.

      The Create Project Type page opens.

   c. Click the **Request Types** policy tab.

      The Request Types policy page opens.

   d. Click the selector icon for the **Project Process Request Type** field.

      The pop-up dialog displays a list of all request types associated with project type.

   e. Select a request type from the list that you want to copy as template for your own request type. For example, **Project Details**.

   > **Note:** For a project type you created by copying from an existing project type, the associated request type is also copied over, which is displayed in the **Project Process Request Type** field.

2. Create your own project type associated request type by copying the request type you selected in step e.

   a. From the PPM menu bar, select **Open > Administration > Open Workbench.**

      The PPM Workbench opens.

   b. From the shortcut bar, select **Demand Mgmt > Request Types**.

      The Request Type Workbench opens.

   c. In the **Request Type** field, type the request type you just selected in "Creating New Project Types with Customized Project Type Associated Request Types" above and click **List**. For example, type **Project Details**.

   d. Select the **Project Details** request type displayed on the Results tab and click **Copy**.

      The Copy Request Type dialog opens.

   e. In the **Request Type Name** field, enter a name for your request type, for example, My

`Project Type`; Complete other fields as necessary and click **Copy.**

The Question dialog opens, asking if you would like to edit the request type.

f. Click **Yes.**

The Request Type : <Request Type Name> dialog opens.

g. Note down the Request Header Type field value, for example, `Project Details`.

h. Click **Cancel.**

3. Create your own Request Header Type by copying the one you noted down in step g.

a. Still in PPM Workbench, from the shortcut bar, select **Demand Mgmt > Request Header Types**.

The Request Header Type Workbench opens.

b. In the **Request Header Type Name** field, enter the value you noted down in step g, for example, `Project Details`, click **List**.

c. Select **Project Details** request header type displayed on the Results tab and click **Copy**.

The Copy Request Header Type dialog opens.

d. In the **Request Header Type Name** field, enter a name for your request header type, for example, `Project Details_MyRHT`.

e. Click **Copy.**

The Question dialog opens, asking if you would like to edit the request header type.

f. Click **Yes.**

The Request Header Type : <Request Header Type Name> dialog opens.

4. Add the QC/ALM Release Information field group to the request header type you created in step 3.

a. In the Request Header Type : <Request Header Type Name> dialog, click **Field Groups**.

The Field Groups dialog opens.

b. Select the checkbox for the QC/ALM Release Information field group, and click **OK**.

The QC/ALM Release Information field group is added to the request header type.

c. Click **OK**.

5. Associate the request header type you just configured to the request type you created in step 2.

a. Still in PPM Workbench, from the shortcut bar, select **Demand Mgmt > Request Types**.

The Request Type Workbench opens.

b. Open the request type you just created in step 2.

The Request Type : <Request Type Name> dialog opens.

c. Click the selector for the **Request Header Type** field.

The Validate dialog opens.

d. Select the request header type you configured in step 4 and click **OK**.

e. Click **OK** in the Request Type dialog.

6. Create your project type with the request type you just configured.

a. From the PPM menu bar, select **Create > Administrative > Project Type**.

The Create Project Type page opens.

b. In the **Project Type** field, type a name for your new project type.

c. Click the **Request Types** policy tab.

The Request Types policy page opens.

d. Click the selector icon for the **Project Process Request Type** field.

The pop-up dialog displays a list of all request types associated with project type.

e. From the list, click the request type you configured in step 5.

f. Click **Create.**

(Optional) Configure Project Overview Layout

After installation of PPM version 9.40, for newly created project types, the four new portlets are included in the Project Overview page by default. No additional configuration work is required.

However, if you create a project type by customizing an existing project type created in an earlier version of PPM, to view project quality related information, you need to manually add the new portlets on the Project Overview Layout policy page. To do so,

1. From the PPM menu bar, select **Search > Administrative > Project Types**.

The Manage Project Types page opens.

2. Select the project type you just created.

The Modify Project Type page opens.

3. Click the **Project Overview Layout** tab.

    The Project Overview Layout policy page opens.

4. In the Layout section, click **Add Portlets**.

    The Add Portlets page opens.

5. Select **Project Overview** from the drop-down list for the **Category** field and click **Find Portlets**.

6. Select the checkboxes for the following four portlets:

    Open Defect Trends
    Priority Defect Trends
    Project Quality Scorecard
    Requirements Trends

7. Click **Add**.

8. On the Project Overview Layout policy page, click **Save** or **Done**.

Registering ALM Instances

To register an ALM instance,

1. From the PPM menu bar, select **Open > Administration > Integrations.**

    The Integration Configurations page opens.

2. Click **View Project Quality** in the navigation pane.

    The View Project Quality integration configurations page opens.

3. Click **Add Integration Configuration**.

    The Server Configuration Details section displays.

4. Complete the fields described in the following table.

Field (*Required)	Description
*QC/ALM Server Name	Specify a unique name for the target ALM server. **Note:** The server name shall not contain pound sign (#) or space.

Field (*Required)	Description
*QC/ALM Version	Select ALM server version from the dropdown list of supported versions. Valid values include `ALM 11.00`, `ALM 11.20`, `ALM 11.50`, and `ALM 12.00`.
*QC/ALM Server URL	URL of the ALM server you want to integrate with. For example, `http://ALM11.company.net:8080/qcbin/`
*QC/ALM Username	ALM account username that you use to log on to the ALM server.
*QC/ALM Password	Password of your ALM account.
Enable User Access Control	Specify whether you want to enable LDAP authentication to restrict user access. This option is available only when you have configured LDAP authentication support on both the PPM Server and the target ALM Server. For details, see "Configuring LDAP Authentication Support for User Access Control" on page 242.
Description	Provide a description for the ALM instance.

5. Click **Save.**

The ALM instance you just registered is added to the Integration Configurations summary list.

Enabling and Scheduling the QC Integration Sync KPI Service

To enable and schedule the QC Integration Sync KPI Service:

1. From the PPM menu bar, select **Open >Administration > Schedule Services.**

   The Schedule Services page opens.

2. Click the table row that displays the QC Integration Sync KPI Service.

   The editable fields for that service are enabled.

3. To enable the service, from the **Status** list, select **Enabled.**

4. Leave the type of expression in the **Schedule Type** list to **Simple.**

5. In the **Schedule** column, leave the default value (24 hours).

6. Click **Save.**

Your changes take effect immediately after you save them. There is no need to restart the PPM Server.

Advanced Configuration Tasks

You can perform the following advanced configuration tasks to provide additional support:

-

-

Configuring PPM Server for MLU Support

If both your PPM Server and ALM server are using a language other than English, you need to perform some extra configuration tasks to enable proper display of project quality related information in PPM.

1. Make sure that the `IntegrationResources_<Language_Code>.properties` file is present in the `<PPM_HOME>\WEB-INF\resources\web\` directory.

    The `IntegrationResources_<Language_Code>.properties` file is a localized version of the `IntegrationResources.properties` file, for example, `IntegrationResources_fr.properties`. It is present in the `<PPM_HOME>\WEB-INF\resources\web\` directory if you have applied the PPM language pack for your language.

    For a list of supported languages and the corresponding language codes, see the *System Requirements and Compatibility Matrix*.

2. Modify the `<PPM_Home>/conf/QCKPI.conf` file.

    a. Stop the PPM Server.

    b. Open the `<PPM_Home>/conf/QCKPI.conf` file in a text editor.

    c. Copy the following content and paste it to the end of the file:

    ```
    #Test
    test.passed=Passed

    #Requirement
    requirement.notCovered=Not Covered
    requirement.na=N/A
    requirement.passed=Passed
    requirement.reviewed=Reviewed
    ```

```
#Defect
defect.priority=4-Very High,5-Urgent
defect.new=New
defect.open=Open
defect.closed=Closed
```

    d. Modify the content you just pasted by

        i. Adding the ALM server name you specified in step 4 as prefix to each parameter, and

       ii. Changing each parameter value to the target language, as follows:

```
#Test
<QC/ALM Server Name>.test.passed=<Translation of "Passed">

#Requirement
<QC/ALM Server Name>.requirement.notCovered=<Translation of "Not
Covered">
<QC/ALM Server Name>.requirement.na=N/A
<QC/ALM Server Name>.requirement.passed=<Translation of "Passed">
<QC/ALM Server Name>.requirement.reviewed=<Translation of "Reviewed">

#Defect
<QC/ALM Server Name>.defect.priority=<Translation of "4-Very High,5-
Urgent">
<QC/ALM Server Name>.defect.new=<Translation of "New">
<QC/ALM Server Name>.defect.open=<Translation of "Open">
<QC/ALM Server Name>.defect.closed=<Translation of "Closed">
```

For example, if you want to configure Simplified Chinese language support, and the ALM server name you specified in step 4 is ALMServer4PPM, modify the newly pasted section to the following:

```
#Test
ALMServer4PPM.test.passed=通过

#Requirement
ALMServer4PPM.requirement.notCovered=未覆盖
ALMServer4PPM.requirement.na=N/A
ALMServer4PPM.requirement.passed=通过
ALMServer4PPM.requirement.reviewed=已审阅

#Defect
ALMServer4PPM.defect.priority=4-非常高,5-紧急
ALMServer4PPM.defect.new=新建
ALMServer4PPM.defect.open=打开
ALMServer4PPM.defect.closed=已关闭
```

e. Save the file in Unicode format.

> **Caution:** Make sure you save the file in Unicode format. Otherwise you may encounter errors.

f. Restart the PPM Server.

Configuring LDAP Authentication Support for User Access Control

If you want to ensure that end users can only view the domains that they are allowed to, you can configure LDAP authorization to restrict their access.

1. Configure PPM Server for LDAP authentication support.

   For detailed instructions, see the *Installation and Administration Guide*.

2. Configure LDAP authentication support on your target ALM server, similar to the follows:
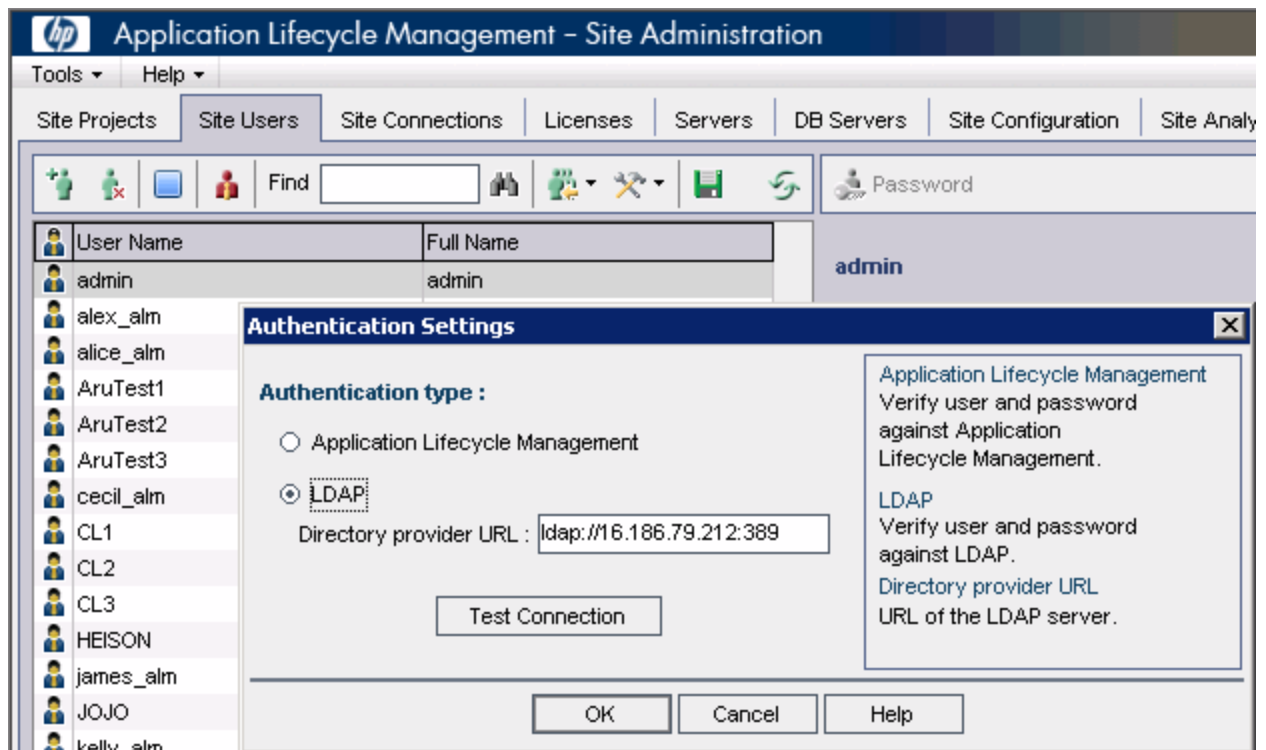
For detailed instructions, see ALM documentation for the current version.

When LDAP authentication support is configured on both the PPM Server and ALM server, the **Enable User Access Control** option is available on the View Project Quality Integration Configurations page.

KPI Data Retrieved from ALM

All KPI data retrieved from ALM are stored in the PPM_INT_QC_KPI table.

The table below describes three types of KPI data that are retrieved from ALM as well as the REST APIs that are used to retrieve KPI data.

KPI data	Description	REST APIs used to retrieve KPI data
Requirements		
# of Total Requirements	Total number of requirements within a specific release	`http://<URL>:<Port>/qcbin/rest/ domains<Domain>/projects/<Project>/ requirements?fields=status, req-reviewed& query= {type-id[<>1];target- rel[<Release_ID>]}`
# of Covered Requirements	Number of the requirements within a specific release, excluding status "Not Covered" or "N/A"	
# of Reviewed Requirements	Number of requirements with status "Reviewed" within a specific release	
# of Passed Requirements	Number of requirements with status of Passed	
Defects		
# of Total Defects	Total number	`http://<URL>:<Port>/qcbin/rest/`

KPI data	Description	REST APIs used to retrieve KPI data

	of defects of release level	`domains/<Domain>/projects/<Project>/`
		`defects/groups/severity,status?`
		`query={target-rel[<Release_ID>]}`

# of Priority Defects	Number of defects with priority level at "4-Very High" or "5-Urgent"	
# of Closed Defects	Number of defects with status of Closed	

Using the View Project Quality Integration Solution

Project managers should not use the integration until PPM administrator has completed the configuration tasks described in "Configuring the Integration Solution" on page 233 .

After initiating a project in PPM as a project manager, when you have obtained the ALM release information, you can update the project by linking it to the corresponding ALM release on the Project Overview page. Then, you can refresh the project to view and monitor the project quality related KPI data retrieved from ALM in PPM.

Linking a PPM Project to an ALM Release

1. Log on to PPM.

2. Open the project you created earlier.

3. Go to the QC/ALM Release Information section on the Project Details tab page.

4. Complete the fields described in the table below:

Field	Description
QC/ALM Server	Click the selector icon and select an ALM server from the list of available ALM servers registered by the PPM system administrator.

Field	Description
QC/ALM Domain	Click the selector icon and select an ALM domain from the list of available ALM domains.
QC/ALM Project	Click the selector icon and select an ALM project from the list of available ALM projects.
QC/ALM Release	Click the selector icon and select a desired ALM release from the list of available ALM releases.
SCORECARD URL	To display ALM scorecard on the Project Overview page in PPM, administrators need to define the scorecard as a analysis item in ALM. You can copy the analysis item URL from ALM and paste it into this field. To do so, a. Log on to ALM and click Dashboard > Analysis View. b. Right click on the desired scorecard analysis item and select Share Analysis Item. c. Select Copy Analysis Item Public URL from the popup dialog and click OK. d. Paste the URL to the SCORECARD URL field in PPM.

A completed QC/ALM Release Information section looks like the follows:

5. Click **Save.**

Viewing and Monitoring Project Quality Information

After you have provided all the necessary information, you can refresh the project, view and monitor project quality information retrieved from ALM on the Project Overview page of PPM.

To view and monitor project quality information,

1. Log on to PPM.

2. From the menu bar, select **Search > Projects**.

3. Provide search criteria in desired fields and click **Search.**

4. Locate your project and double click it.

5. The Project Summary tab page of the Project Overview page shall display project quality related portlets when there is data available.

   Examples of the four project quality related portlets:

   ○ Open Defect Trends
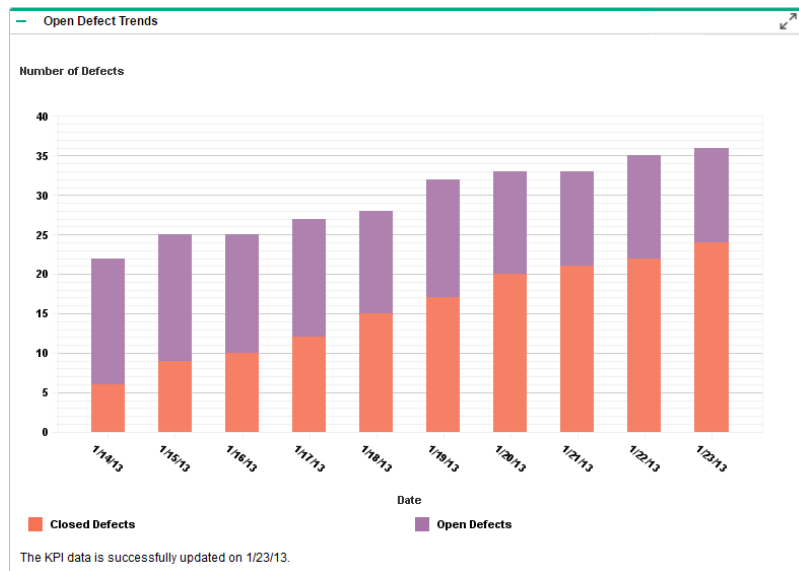
   ○ Priority Defect Trends
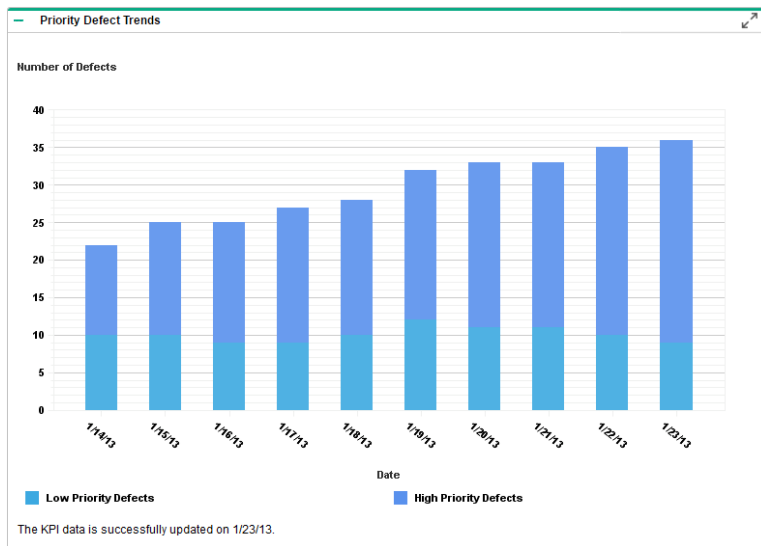
- Requirements Trends

- Project Quality Scorecard

> **Note:** When there is no data available, the portlets are not displayed.

Notes about the Mapping Relationship between PPM Project and ALM Release

The mapping relationships between PPM projects and ALM releases are stored in the related requests. You may use the SQL below to find related project mappings.

```
SELECT p.PRJ_PROJECT_ID, p.project_name, vqm.vqm_server_id, vqm.vqm_domain,
vqm.vqm_project, vqm.vqm_release_id, vqm.vqm_release
FROM KCRT_FG_VQM_INFO vqm, KCRT_FG_PFM_PROJECT p
WHERE p.request_id = vqm.request_id
AND p.prj_project_id is not null AND vqm.vqm_release_id is not null
```

Workaround for Mapping One PPM Project to Multiple ALM Releases

PPM supports ONE-ONE mapping relationship between a PPM project and an ALM release. If your organization has the business need for mapping one PPM project to multiple ALM releases, you can follow the workaround below:

1. Create PPM Project A for linking to one ALM release.

2. Create PPM Master Project B, and then add task for PPM Project A.

3. Set Project A as reference in the task.

4. Create portlet types to display the quality statistics in Master Project B.

Chapter 9: Integrating PPM Tasks with ALM Releases - View Project Quality

This section contains the following:

-

-

-

Introduction to Integrating PPM Tasks with ALM Releases

The integration between PPM tasks and ALM Releases implemented in version 9.40 is an extension of the capability of the View Project Quality integration solution (integration between PPM projects and ALM releases). The integration extension enables project managers to manage multiple ALM releases with one PPM project by integrating project tasks with ALM releases.

The integration of PPM tasks with the Releases module of the Application Lifecycle Management (ALM) product allows project management officers, project managers, development managers, and QA managers to have visibility into quality KPIs and ALM scorecards of multiple ALM releases from PPM Center projects by integrating PPM tasks with ALM releases, and allows them to optimize the management of project quality by:

- Introducing and enforcing consistent workflows for all major application delivery processes.

- Initiating and managing both application and testing projects and ALM releases.

The integration allows project managers to link a task under a project to a specific release managed in ALM. Each PPM Center task is associated with a single release in ALM throughout the task life cycle. This is a one-way one-to-one mapping relationship. This association begins when a project manager maps one task to a specific ALM release.

After the mapping relationship is established, the project managers are able to view quality KPIs and ALM scorecard report for a release retrieved from ALM. In addition, project managers can also view the overall release hierarchy information of a specific work package from within PPM Center.

The following table describes data tables used in this integration solution:

Data table	Description
PPM_INT_SOLUTIONS_NLS	Stores the integration solution information
PPM_INT_CONFIGURATIONS	Stores all ALM server information related to the integration configurations
PPM_INT_QUALITY_TASK_MAPPING	Stores mapping information about PPM tasks and ALM releases
PPM_INT_QUALITY_STATISTICS	Stores ALM quality statistics information
PPM_INT_EVENTS [a]	Stores error logs

a. In version 9.31, the following two columns are added to this table, so that as an administrator, you can parse more effective information about PPM-ALM integration issues from the events logged in the table:

- PPM_REQUEST_ID
- SYNC_PARTNER_ID

For detailed information, see *Data Model Guide*.

When configuring the integration solution, consider the following:

- The integration solution supports one-one mapping between a PPM task to an ALM release only.

- This integration supports OOTB quality metrics only. Customizable reports and metrics are not supported yet.

- There is no cleanup service to clean all the outdated data in the KPIs table.

- You can not delete any existing ALM server configuration entries.

The table below summarizes some of the similarities that this integration solution shares with the View Project Quality solution introduced in PPM version 9.20:

Similarities	Differences	
	View Project Quality solution (available since 9.20)	**Integration of PPM Tasks with ALM Releases (available since 9.21)**
Both integrate with ALM Releases module	Integrates at PPM project level	Integrates at PPM task level
Both support one-one mapping relationship	Between a PPM project and an ALM release	Between a PPM task and an ALM release
Both retrieve similar KPI data and scorecard information	• Displays data retrieved from ALM in four portlets on the	• Displays data retrieved from ALM in graphs on:

Similarities	Differences	
	View Project Quality solution (available since 9.20)	Integration of PPM Tasks with ALM Releases (available since 9.21)
from ALM, including: • Priority Defects Trend • Open Defects Trend • Requirements Trend • Project Quality Scorecard	**Project Summary** tab of the Project Overview page • Stores data retrieved from ALM in the PPM_INT_QC_ KPI table	○ The **Quality** tab of Project Overview page; ○ The **Quality** tab of the Task Details page • Stores data retrieved from ALM in the PPM_INT_ QUALITY_STATISTICS table
Both uses background synchronization services	Uses the QC Integration Sync KPI Service	Uses the Project Quality Sync Service
Both share the same View Project Quality integration configuration landing page		The **Enable User Access Control** option is removed after deploying the integration bundles.

These two integration solutions are independent of each other. You can still use the View Project Quality solution after upgrading PPM to version 9.40. However, HPE recommends you use the integration of PPM tasks with ALM Releases.

For information about the ALM versions supported for integration, see the *System Requirements and Compatibility Matrix*.

> **Note:** No software needs to be installed on the ALM server to integrate PPM and ALM. However, see the *System Requirements and Compatibility Matrix*.

For more information about ALM, see its product documentation at the following website:

https://softwaresupport.hpe.com

User Flow

The typical flow of this integration is as follows:

Step 1: A PPM administrator configures the integration solution

Before project managers can use the integration, PPM administrator needs to configure the integration solution, including:

- Identifying whether there is business needs for integrating PPM tasks with ALM releases

- Registering ALM instances

- Enabling and scheduling the Project Quality Sync Service

Step 2: A project manager initiates a project in PPM, and provides deliverables to all the assigned teams.

Step 3: Based on the deliverables, business analysts, development managers, and QA managers build milestones using Project Planning and Tracking (PPT) in ALM.

Step 4: A release manager creates a corresponding release in ALM and defines key performance indicators (KPIs) for milestones and releases.

Step 5: The project manager updates the PPM project by linking its tasks to corresponding ALM releases from the Task Details page.

For details about linking to an ALM release, see "Linking a PPM Task to an ALM Release" on page 261.

Step 6: ALM tracks release progress against specific milestones, with KPI matrices retrieved and sent to PPM to be included in the overall project health status.

Step 7: Project managers and project participants view and monitor quality matrices on the Quality tab of Project Overview page.

For more information, see "Using the Integration of PPM Tasks withALM Releases" on page 260.

Configuring the Integration of PPM Tasks with ALM Releases

- "Downloading and Installing the PPM Plug-in for ALM Integration" on the next page

- "Adding an ALM Integration Configuration" on page 255

- "Enabling and Scheduling the Project Quality Sync Service" on page 257

- "Configuring PPM Server for MLU Support" on page 257

- "KPI Data Retrieved from ALM" on page 259

Downloading and Installing the PPM Plug-in for ALM Integration

To integrate PPM tasks with ALM releases, administrators must download and install the free ALM Integration Plug-in for PPM. This is a one-time action.

Installation Instructions

To download and install the ALM Integration Plug-in for PPM:

1. Obtain the plug-in bundles from the HP Live Network.

    a. Go to the PPM Community on HP Live Network.

    b. Click **ALM Integration Plug-in for PPM**.

       The ALM Integration Plug-in for PPM - Downloads page opens.

    c. Click **Download** in the **ALM Integration Plug-in for PPM 9.40** section.

       The `ppm-940-ALMPlugin.zip` package is downloaded.

       This package contains the following two bundles:

Bundle name	Description
`ppm-940-PluginQuality.jar`	Contains the plug-in basic structure for integrating PPM with ALM release management tools.
`ppm-940-PluginQualityVPQ.jar`	Contains the plug-in files specifically for integrating PPM tasks with the ALM Releases module.

2. Stop PPM Server.

3. Unzip the package and copy the two bundles to the *<PPM_Home>* directory.

4. Deploy the bundles by running the following commands from the *<PPM_Home>*/bin directory.

    `sh ./kDeploy.sh -i PluginQuality`

    `sh ./kDeploy.sh -i PluginQualityVPQ`

5. Repeat step 3 and step 4 for each of the server nodes in your cluster.

6. Restart PPM Server.

> **Note:** HPE recommends you restart the PPM Server to make sure that the integration bundles are properly deployed before you continue to deploy bundles for another integration solution.

Entities Installed by the HP PPM Plug-in for ALM Release Integration

The following entities are installed:

- **Quality** tab added to the Task Details page

  The Quality tab on the Task Details page allows project managers to link the task to a specific ALM release.

  After the mapping relationship is established, the tab displays the following quality graphs and scorecard report retrieved from ALM for the linked release:

  ○ **Priority Defects Trend:** Displays trend of defects of higher severity in the last 10 days. The higher severity defects include defects of priority S1 and S2.

  ○ **Open Defects Trend:** Displays open defects trend in the last 10 days. The open defects include open defects number and total defects number.

  ○ **Requirements Trend:** Displays requirement status trend in the last 10 days. The requirement status includes requirement coverage.

  ○ **Scorecard Report:** Displays scorecard report for the linked ALM release.

  > **Note:** The scorecard report is displayed only when the ALM scorecard information is configured in the project request field of the Project Planning and Tracking (PPT) module in ALM.

- **Quality** tab added to the Project Overview page

  The Quality tab on the Project Overview page offers a centralized view of the following quality graphs for each of the ALM releases linked to tasks of the current project:

  ○ **Priority Defects Trend:** Displays trend of defects of higher severity in the last 10 days. The higher severity defects include defects of priority S1 and S2.

  ○ **Open Defects Trend:** Displays open defects trend in the last 10 days. The open defects include open defects number and total defects number.

○ **Requirements Trend:** Displays requirement status trend in the last 10 days. The requirement status includes requirement coverage.

Clicking a specific release in the navigation tree brings users to the Task Details page of the linked task.

- Project Quality Sync Service

The Project Quality Sync Service synchronizes quality KPI data from ALM to PPM. The default synchronization interval is 24 hours. You can set the synchronization interval on the Schedule Services page.

The Project Quality Sync Service gets all valid ALM server configurations and loops through all configurations to retrieve KPI information from ALM and stores the data in PPM database.

If the service found that the quality KPI for current day already exists, it skips the task and continues to process the next task.

Synchronization rules are as follows:

○ When a project or task is deleted, the mapping and records are removed from the database. This means that users will lose all the data related to the corresponding mapping.

○ When a task is canceled or completed, synchronization stops.

○ When a project is canceled or closed, synchronization stops.

○ If an ALM release is deleted on ALM end or cannot be found when the service tries to synchronize data, the status is logged in the PPM_INT_QUALITY_TASK_MAPPING table.

○ If the end date of an ALM release is ahead of the user's current date, for example, ALM release R1 is scheduled to end on 11/11/2012 and today is 05/01/2013, then the service stops synchronization next time.

For details about KPI data to be retrieved from ALM, see "KPI Data Retrieved from ALM" on page 259.

Adding an ALM Integration Configuration

Administrators need to add integration configurations with ALM server on the View Project Quality configuration page before project managers can link their project tasks to releases managed in the ALM Release module..

To add an integration configuration,

1. Log on to PPM.

2. From the menu bar, select **Open > Administration > Integrations**.

   The integration configuration landing page opens.

3. Click **View Project Quality** in the navigation pane.

   The View Project Quality integration configurations page opens.

4. In the Server Configuration Details section, click **Add Integration Configuration**.

   The Server Configuration Details section displays.

5. Complete the fields described in the following table.

Field (*Required)	Description
*QC/ALM Server Name	Specify a unique name for the target ALM server. **Note:** The server name shall not contain pound sign (#) or space.
*QC/ALM Version	Select ALM server version from the dropdown list of supported versions. Valid values include `ALM 11.00`, `ALM 11.20`, `ALM 11.50`, and `ALM 12.00`.
*QC/ALM Server URL	URL of the ALM server you want to integrate with. For example, `http://ALM11.company.net:8080/qcbin/`
*QC/ALM Username	ALM account username that you use to log on to the ALM server.
*QC/ALM Password	Password of your ALM account.
Description	Provide a description for the ALM server.

6. Click **Save.**

   The ALM server you just configured is added to the Integration Configurations summary list.

Enabling and Scheduling the Project Quality Sync Service

To enable and schedule the Project Quality Sync Service:

1. From the PPM menu bar, select **Open >Administration > Schedule Services.**

   The Schedule Services page opens.

2. Click the table row that displays the Project Quality Sync Service.

   The editable fields for that service are enabled.

3. To enable the service, from the **Status** list, select **Enabled.**

4. Leave the type of expression in the **Schedule Type** list to **Simple.**

5. In the **Schedule** column, leave the default value (24 hours).

6. Click **Save.**

Your changes take effect immediately after you save them. There is no need to restart the PPM Server.

Configuring PPM Server for MLU Support

If both your PPM Server and ALM server use a language other than English, you need to perform some extra configuration tasks to enable proper display of project quality related information in PPM.

1. Make sure that the `IntegrationResources_<Language_Code>.properties` file is present in the `<PPM_Home>\WEB-INF\resources\web\` directory.

   The `IntegrationResources_<Language_Code>.properties` file is a localized version of the `IntegrationResources.properties` file, for example, `IntegrationResources_fr.properties`. It is present in the `<PPM_Home>\WEB-INF\resources\web\` directory if you have applied the PPM language pack for the desired language.

   For a list of supported languages and the corresponding language codes, see the *System Requirements and Compatibility Matrix*.

2. Modify the `<PPM_Home>\conf\hpALMKPI.conf` file. The `hpALMKPI.conf` file is added into the `<PPM_Home>\conf\` directory automatically after you deploy the `ppm-940-PluginQualityALM.jar` bundle.

To do so,

   a. Stop the PPM Server.

   b. Open the *<PPM_Home>*\conf\hpALMKPI.conf file in a text editor.

   c. Copy and paste the following content to the end of the file:

```
#Test
test.passed=Passed

#Requirement
requirement.notCovered=Not Covered
requirement.na=N/A
requirement.passed=Passed
requirement.reviewed=Reviewed

#Defect
defect.priority=4-Very High,5-Urgent
defect.new=New
defect.open=Open
defect.closed=Closed
```

   d. Modify the content you just pasted by doing the follows:

      i. Add the ALM server name you specified in "Adding an ALM Integration Configuration" on page 255 as prefix to each parameter, and

     ii. Change each parameter value to the target language, as follows:

```
#Test
QC/ALM Server Name>.test.passed=<Translation of "Passed">

#Requirement
<QC/ALM Server Name>.requirement.notCovered=<Translation of "Not
Covered">
<QC/ALM Server Name>.requirement.na=N/A
<QC/ALM Server Name>.requirement.passed=<Translation of "Passed">
<QC/ALM Server Name>.requirement.reviewed=<Translation of "Reviewed">

#Defect
<QC/ALM Server Name>.defect.priority=<Translation of "4-Very High,5-Urgent">
<QC/ALM Server Name>.defect.new=<Translation of "New">
<QC/ALM Server Name>.defect.open=<Translation of "Open">
<QC/ALM Server Name>.defect.closed=<Translation of "Closed">
```

For example, if you want to configure Simplified Chinese language support, and the ALM server name you specified in "Adding an ALM Integration Configuration" on page 255 is

`ALMServer4PPM`, modify the newly pasted section to the following:

```
#Test
ALMServer4PPM.test.passed=通过

#Requirement
ALMServer4PPM.requirement.notCovered=未覆盖
ALMServer4PPM.requirement.na=N/A
ALMServer4PPM.requirement.passed=通过
ALMServer4PPM.requirement.reviewed=已审阅

#Defect
ALMServer4PPM.defect.priority=4-非常高,5-紧急
ALMServer4PPM.defect.new=新建
ALMServer4PPM.defect.open=打开
ALMServer4PPM.defect.closed=已关闭
```

e. Save the file in Unicode format.

> **Caution:** Make sure you save the file in Unicode format. Otherwise you may encounter errors.

f. Restart the PPM Server.

KPI Data Retrieved from ALM

All KPI data retrieved from ALM are stored in the PPM_INT_QUALITY_STATISTICS table.

The table below describes three types of KPI data that are retrieved from ALM as well as the REST APIs that are used to retrieve KPI data.

KPI Data	Description	REST APIs used to retrieve KPI data
Requirements		

# of Total Requirements	Total number of requirements within a specific release	`http://<URL>:<Port>/qcbin/rest/domains/<` 	
		Domain
		`>/projects/<`
		Project`>/requirements?fields=status,`
		`req-reviewed& query={type- #`
		`of Covered id[<>1];release.id[<`*Release_*
		ID`>]}`

# of Covered Requirements	Number of the requirements within a specific release, excluding status "Not Covered" or "N/A"	

KPI Data	Description	REST APIs used to retrieve KPI data
# of Reviewed Requirements	Number of requirements with status "Reviewed" within a specific release	
# of Passed Requirements	Number of requirements with status of Passed	
Defects		

# of Total Defects	Total number of defects of release level	`http://<` 	
		`URL`
		`>:<`
		`Port`
		`>/qcbin/rest/domains/<`
		`Domain`
		`>/projects/<`
		`Project>/defects/groups/severity,`
		`status?query={target-rel`
		`[<Release_ID>]}`

# of Priority Defects	Number of defects with priority level at "4-Very High" or "5-Urgent"	
# of Closed Defects	Number of defects with status of Closed	

Using the Integration of PPM Tasks withALM Releases

Project managers can start using the integration after PPM administrators have completed the configuration tasks described in "Configuring the Integration of PPM Tasks with ALM Releases" on page 252.

After a project manager initiates a project in PPM, and adds tasks and assigns them to sub-project managers, with the ALM release information available, you can do the follows:

- As a sub-project manager, you can link your task to a corresponding ALM release on the Task Details page for your task. See "Linking a PPM Task to an ALM Release" on the next page.

  Note: If a task is already mapped to an ALM release, you cannot map it to another ALM release; If a task has progress information that is synchronized from Agile Manager, you cannot map it to an ALM release as well.

Then, after one-one mapping relationship between a PPM task and an ALM release is established:

- As a sub-project manager, you can view and monitor the quality related KPI data retrieved from ALM for your task on the **Quality** tab of your Task Details page.

- As a project manager, you can view and monitor the project quality related KPI data retrieved from ALM for all releases that are linked to your project tasks on the **Quality** tab of the Project Overview page, and drill down to details for each task.

Linking a PPM Task to an ALM Release

To link a PPM task to an ALM release,

1. Log on to PPM.

2. From the menu bar, select **Search > Projects**.

   The Search Projects page opens.

3. Locate and open the desired project.

4. On the Project Summary tab of the Project Overview page, click **Edit Work Plan** in the Work Plan portlet.

   The Work Plan for <*Project_Name*> page opens.

5. Open the Task Details page for the desired task that you want to link to anALM release, and go to the **Quality** tab.

6. Click **Start Mapping**.

   The **Link to an HPE ALM release** section displays Step 1: General.

7. From the **Server** drop-down list, select a target ALM server that you want to link to.

8. Click **Next.**

   The mapping process move to Step 2: Authentication.

9. In the **Username** and **Password** fields, enter the user name and password that you use to log on to the target server.

10. Click Next.

    The mapping process moves to Step 3: Project Server and loads domain, project, and release information based on the account information you provided.

11. Select a value from the drop-down list for each of the **Domain**, **Project**, and **Release** fields.

12. Click **Confirm**.

    The mapping process moves to Step 4: Confirmation.

13. Verify the information and click **Done.**

14. The mapping is established and starts retrieving data from the linked ALM release.

    When the synchronization completes, the following graphs display on the **Quality** tab:

    ○ Priority Defects Trend

    ○ Open Defects Trend

    ○ Requirements Trend

    Note: The Scorecard Report is not yet available. For instructions about configuring the Scorecard Report, see "Configuring Scorecard Report " below.

Configuring Scorecard Report

1. Get Scorecard URL for the Scorecard Report.

    a. Log on to the target ALM server and click **Dashboard > Analysis View**.

    b. Right click on the desired scorecard analysis item and select **Share Analysis Item**.

    c. Select **Copy Analysis Item Public URL** from the popup dialog and click **OK**.

    Note: Make sure that the scorecard analysis item you configured is related to the ALM release you linked, otherwise it may mess up the display of quality related data on the Quality tab.

2. Click **Edit Link** on the **Quality** tab of the Task Details page.

3. Paste the URL to the field and click **Save**.

    The Quality tab page refreshes to retrieve scorecard information.

4. Scroll down the view the scorecard report.

An example of the scorecard report:
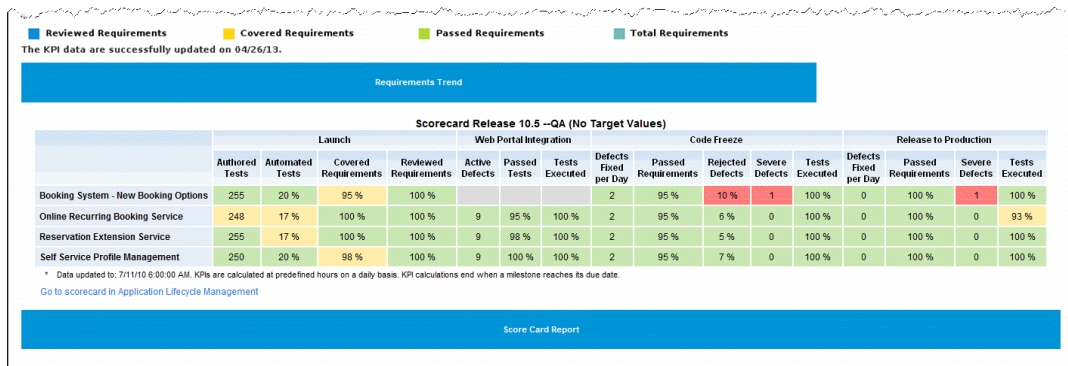
Deleting a Mapping from the Task Details Page

To delete a mapping from the Quality tab of the Task Details page,

1. Log on to PPM.

2. From the menu bar, select **Search > Projects**.

   The Search Projects page opens.

3. Locate and open the desired project.

4. Go to the Task Details page for the desired task that you want to remove the mapping you already established for it, and go to the **Quality** tab.

5. Click ✖ .

   A prompt dialog pops up.

6. Click **OK.**

   The mapping is removed and the **Quality** tab page returns to its original state.

   You can click **Start Mapping** to link the task to another ALM release as needed.

Viewing and Monitoring Project Quality KPIs and Scorecard Report

Once the one-one mapping relationship between a PPM task and an ALM release is established, the Quality tab of the Task Details displays the following quality information retrieved from ALM:

- Priority Defects Trend

- Open Defects Trend

- Requirements Trend

- Scorecard Report (if you provided scorecard URL)

You can do the follows:

- As a sub-project manager, you can view and monitor the quality related KPI data retrieved from ALM for your task on the **Quality** tab of the Task Details page. See "View Quality Data from the Task Details page " below.

- As a project manager, you can view and monitor the project quality related KPI data retrieved from ALM for all releases linked to your project tasks on the **Quality** tab of the Project Overview page, and drill down to details for each task. See "View Quality Data from the Project Overview Page " on the next page.

View Quality Data from the Task Details page

To do so,

1. Log on to PPM.

2. From the menu bar, select **Search > Projects**.

3. Provide search criteria in desired fields and click **Search**.

4. Locate your project and open it.

5. On the Project Summary tab page, click **Edit Work Plan** in the Work Plan portlet.

6. Locate your task and open it.

   The Task Details page displays.

7. Go to the **Quality** tab.

   The Quality tab displays quality KPI data graphs as well as scorecard report.

Note: If the Scorecard Report is not available, you need to configure the Scorecard Report link first. For instructions, see "Linking a PPM Task to an ALM Release" on page 261.

View Quality Data from the Project Overview Page

To view quality related KPI data from the Project Overview page,

1. Log on to PPM.

2. From the menu bar, select **Search > Projects**.

3. Provide search criteria in desired fields and click **Search**.

4. Locate your project and open it.

The Project Overview page displays.

5. Go to the **Quality** tab.

   ○ If no tasks are linked to any ALM releases, a promotion picture is displayed.
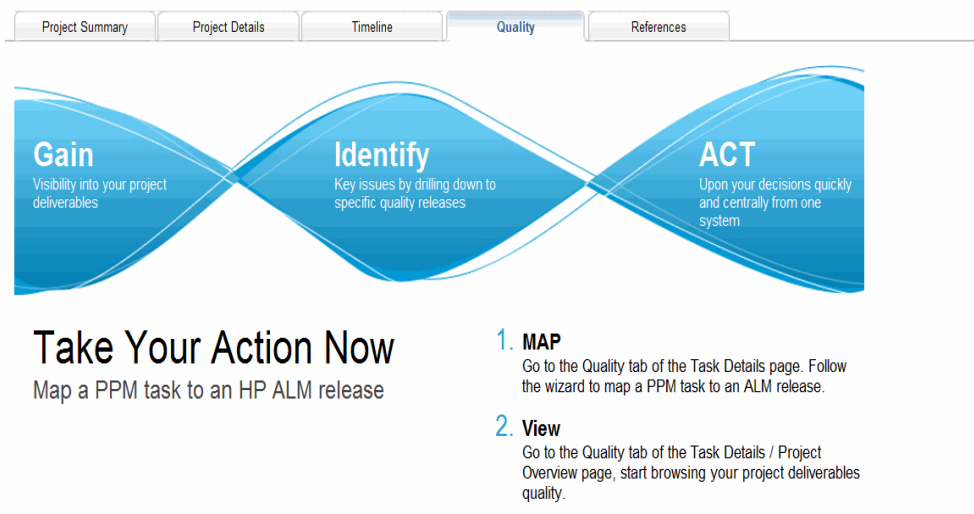
   ○ If sub-project managers have linked their tasks to ALM releases, the **Quality** tab displays quality KPI data aggregated for the master project by default.

6. Click [icon] or [icon] to cycle the display of the quality graphs.

7. To browse quality KPI data for a linked ALM release, click the desired release in the left navigation tree.
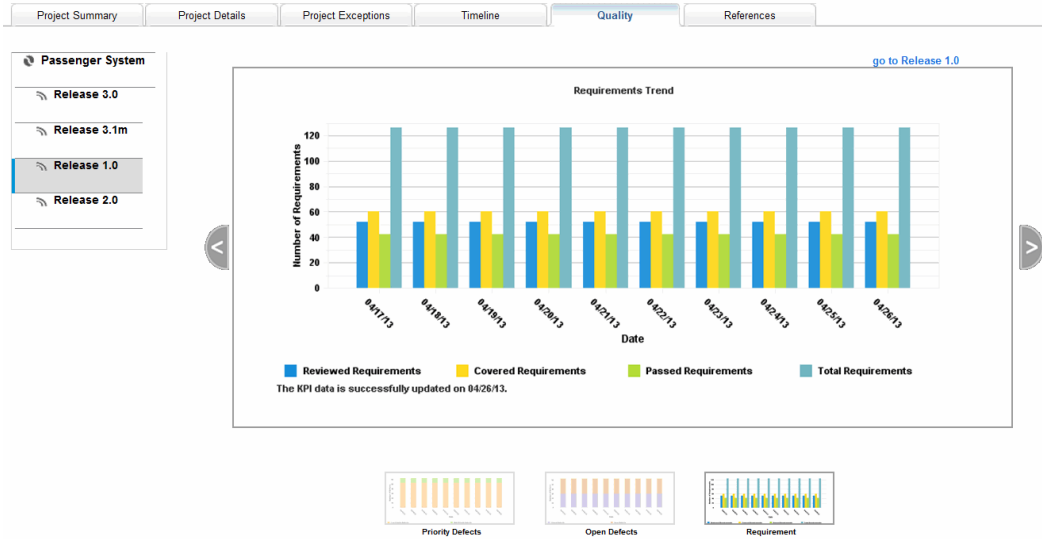
8. To view scorecard report for the release, click the **Go to <*Release_Name*>** link at the top right corner.

   The Task Details page for the task lined to the release displays.

9. Go to the **Quality** tab and scroll down to view the scorecard report.

Part 4: Integration with HPE Release Control

This part includes the following solution integration:

- Integrating PPM with HPE Release Control, Using ALM

Chapter 10: Integrating PPM with HPE Release Control, Using ALM

Integrating PPM with Release Control enables you to link directly from a change request in PPM to associated impact analysis data in Release Control. Based on the information provided in Release Control, you can then decide whether to approve or reject the deployment of the change request.

For more information about the benefits of this integration, see "Integration of PPM with Release Control, Using ALM" on page 21.

For information about the versions supported for integration, see the *System Requirements and Compatibility Matrix.*

> **Note:** No software needs to be installed on the Release Control server to integrate PPM with Release Control. However, see the *System Requirements and Compatibility Matrix*.

For more information about Release Control, see its product documentation at the Web site described in "Available PPM Integrations " on page 18.

Configuring Release Control for the Integration

To configure Release Control for integration, you must do the following:

- Configure the PPM Web Services adapter in Release Control

- Configure the JavaScript files in Release Control

> **Note:** For more information about the steps in this procedure, see the *Installation and Administration Guide.*

Before beginning this configuration, verify that Web services are enabled for use with PPM, as follows:

1. Stop the PPM Server.

2. Run the following script:

   `sh ./kConfig.sh`

3. Verify that the ENABLE_WEB_SERVICES parameter in the PPM `server.conf` configuration file

is set to `true`.

4. Restart the PPM Server.

Configuring the PPM Web Services Adapter

To establish integration, you must configure the PPM Web Services adapter in Release Control to convert change requests that come from PPM to generic requests that Release Control can process, as follows:

1. Configure the PPM Web Services connector settings as described in the *Release Control Installation and Configuration Guide.*

   > **Note:** Before version 7.0, PPM was known as Mercury IT Governance Center or ITG. Release Control software and documentation might still refer to PPM as IT Governance Center or ITG.

2. In the `itg-ws-adapter.settings` file, under `<request-type level="1">`, set the `requestTypeName` to the name of the PPM request type representing a PPM release, for example, **ALM - Release Management.**

3. In the `itg-ws-adapter.settings` file, under `<request-type level="2">`, set the `requestTypeName` to the name of the PPM request type representing a PPM change, for example, **ALM - Request for Change (RFC),** and set the `parentRequestTypeName` to the value of the `requestTypeName` in step 2.

Configuring the JavaScript Files

Configure the JavaScript files in Release Control for the integration with PPM.

> **Note:** If PPM is (or will be) integrated with HPEUniversal CMDB as well, additional configuration steps may be required in Release Control before configuring the JavaScript.

Refer to documentation for Release Control and contact HPERelease Control Support as necessary.

Configuring PPM for the Integration

Before beginning to configure the integration as described in this section, make sure that ALM has been installed and initially configured as described in "Installing and Setting Up ALM Content Bundle" on page 26.

Establishing Server Connections for Supported Versions

Make sure that the HTTP port is open between the PPM Server and the Release Control machines.

Verify that a supported version of Release Control is installed and running (see the *System Requirements and Compatibility Matrix*).

Configuring the server.conf Parameter in PPM

> **Note:** For more information about the steps in this procedure, see the *Installation and Administration Guide.*

To be able to open Release Control from PPM as part of the integration, add and specify the parameter related to Release Control integration to the PPM `server.conf` configuration file, as follows:

1. Stop the PPM Server.

2. Run the following script:

   `sh ./kConfig.sh`

   Set the parameter and value as shown in the following table. (All parameter names begin with `com.kintana.core.server.` but that is not shown in the table.)

Parameter	Value
CCM_MACHINE_ URL	URL of the Release Control server: `http://<RC_Host>:<Port>/ccm/` where `<RC_Host>` represents the host machine on which Release Control is running

3. Verify that the `ENABLE_WEB_SERVICES` parameter in the `server.conf` file is set to `true`.

4. Restart the PPM Server.

Using the Integration of PPM with Release Control

ALM provides the ALM - Releases portlet to facilitate the release request process (see "Release Management Portlets to Display KPIs" on page 84). If PPM and Release Control are integrated, for each release request you can click the **Click to View** link in the **View Impact** column in the portlet to log in to Release Control. When you log in, Release Control displays the **Overview** tab and other tabs. The information displayed for the selected change request includes the following, for example:

- The "service desk application" from which the request originated. From the perspective of Release Control, PPM is a service desk application in this context.

- On the **Request Details** tab, the request ID number of the original change request, with a link to open the change request in PPM.

- On the **Request Details** tab, the planned and actual start and end times for execution of the request.

- On the **Overview** tab, the number of configuration items (CIs) and applications that are affected by the request, with links that access the **Impact Analysis** tab, where details of the affected CIs and applications are displayed.

- On the **Collaboration** tab, the communication among users regarding action items.

If PPM is integrated with Release Control *and* with Universal CMDB, you can click the **Launch HPE Release Control** button in the **Impacted Configuration Items** section of an ALM - Request for Change (RFC) request in PPM to access Release Control.

Part 5: Integration with HPE Service Manager

This part includes the following solution integrations:

- Integrating PPM Requests with HPE Service Manager Changes, Using ALM

- Integrating PPM Tasks with HPE Service Manager RFCs

Chapter 11: Integrating PPM Requests with HPE Service Manager Changes, Using ALM

This section includes the following:

-

-

-

-

-

-

-

-

-

-

-

-

-

Introduction to Integrating PPM Requests with Service Manager Changes, Using ALM

For an overview of the integration of PPM with Service Manager, see .

The integration is enabled by a configurable Service Manager adapter file in PPM along with the request types and workflows provided by ALM, so that PPM acts as a single repository for application-related requests for change (RFCs). The adapter converts changes (tickets) in Service Manager to requests for change (RFCs), and then imports those RFCs into PPM for processing.

Conversely, using a configurable PPM adapter file that also resides in PPM, the integration can optionally *update* Service Manager changes based on subsequent changes made in PPM to the RFCs.

If the Service Manager adapter file and the associated PPM adapter file are both configured, such that data can be sent in both directions between Service Manager and PPM, the integration is said to be "bidirectional."

A separate adapter file is required for each mapping between a Service Manager change and a PPM request type.

ALM provides two default adapter files—a Service Manager adapter file and an associated default PPM adapter file.

The integration runs as a service in the PPM Server. As described later, the configuration of an adapter file controls various aspects of its import process.

This section describes how to configure the adapters in PPM and the Service Manager application for integration. This section is intended for Service Manager administrators or for PPM system administrators who are also familiar with Service Manager.

For information about the supported versions of Service Manager, see the *System Requirements and Compatibility Matrix*.

> **Caution:** To integrate PPM requests with changes in Service Manager version 9.20, you must modify the ChangeManagement WSDL in Service Manager as described in this section. These WSDL modifications may interfere with the operation of Service Manager integrations with other products. Before integrating PPM with Service Manager version 9.20, review the WSDL modifications described in the following sections to determine whether they could affect Service Manager integrations with other products, and proceed accordingly:
>
> - step d
>
> - "Associating New Display Actions to the New Processes Loaded" on page 282
>
> - "Configuring Browsing of Service Manager Changes from a URL" on page 286
>
> The integration of PPM*tasks* with changes in Service Manager version 9.20, as described in "Integrating PPM Tasks with HPE Service Manager RFCs" on page 332, does *not* require changing the Service Manager WSDL.

> **Note:** No software needs to be installed on the Service Manager server for integration with PPM. However, see the *System Requirements and Compatibility Matrix.*

For information about the ALM - Request for Change (RFC) request type in PPM that is used to establish integration of PPM with Service Manager, see "ALM - Request for Change (RFC) Request Type" on page 33.

For more information about Service Manager, see its product documentation at the Web site described in "Available PPM Integrations " on page 18.

Converting Service Manager Changes to PPM RFCs

The following figure depicts the flow for converting a change in Service Manager to an ALM request for change (RFC), and importing the RFC into PPM.

Figure 4-1. Using the Service Manager adapter to import changes from   Service Manager into PPM
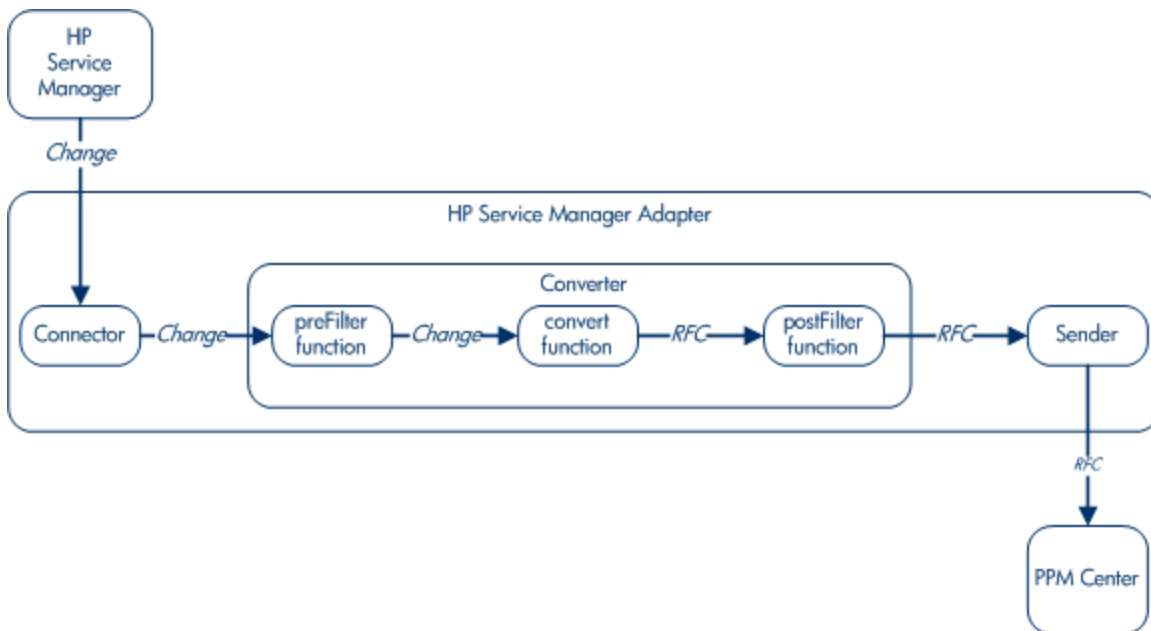
The Service Manager adapter consists of the following three components:

- **Connector.** Collects changes from the Service Manager system.

- **Converter.** Uses field mapping to convert the changes from the Service Manager data model in which the changes were created to RFCs for the PPM data model.

    The converter also contains two optional filters to control which changes are imported into PPM. The preFilter filters out categories of changes you specify in the Service Manager data model before those changes are converted. After the Service Manager changes are converted to RFCs, the postFilter filters out categories of requests you specify before those requests are presented to the sender.

- **Sender.** Using the RFC data from the converter, creates the RFCs in PPM.

Converting PPM RFCs to Service Manager Change Updates

Similar to the Service Manager adapter, the PPM adapter consists of connector, converter, and sender components. In this case, these components allow Service Manager to import updates from PPM.

The following figure depicts the flow for converting an ALM RFC (request for change) in PPM to an update to a change in Service Manager, and importing the update into Service Manager.

Figure 4-2. Using the PPM adapter to import change updates   from PPM into Service Manager
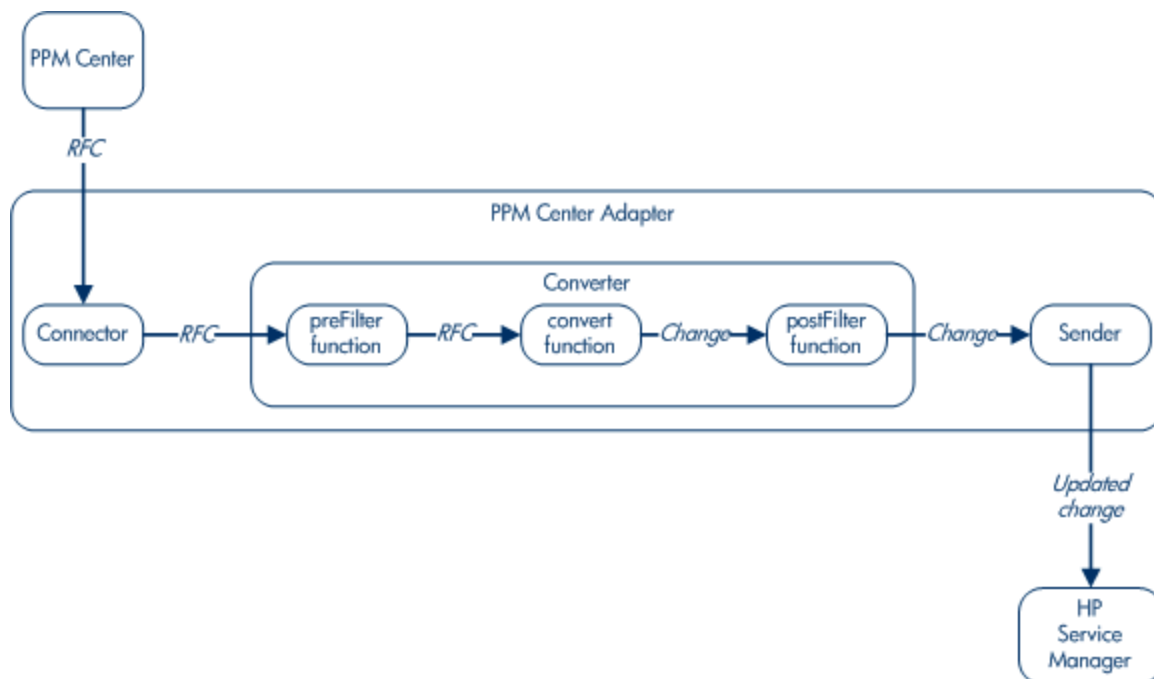
Overview of Configuring the Service Manager Integration

Before configuring the integration, you must identify the Service Manager data attributes that will be integrated with PPM, including which of the attributes of Service Manager changes to send to PPM, and, for bidirectional integration, which of the attributes to receive back from PPM.

Before you can begin importing changes from Service Manager into PPM, you must configure Service Manager and configure the Service Manager adapter in PPM to enable integration. The procedures are summarized as follows:

- Configure Service Manager in the particular ways required for integration of PPM with Service Manager.

- Generate Service Manager Web service stubs for Service Manager. PPM needs the Service Manager Web service stubs to connect to Service Manager.

- Configure the Service Manager adapter configuration file on the PPM Server, to support converting a change in Service Manager to an ALM request for change (RFC) and then importing the RFC into PPM. ALM provides an Service Manager default adapter configuration file.

- If you want the integration to be bidirectional, that is, to also send updates that are made in PPM RFCs back to Service Manager as change updates, configure the PPM adapter configuration file on the PPM Server. ALM provides a default PPM adapter configuration file associated with Service Manager.

- Configure the required `server.conf` parameter in PPM.

- Restart the PPM Server in normal mode and verify that the ALM Startup service has started.

- Configure logging of errors and events.

Configuring Service Manager for Integration with PPM

Before starting these configuration procedures, make sure that the Service Manager server, Service Manager client, and Service Manager Webtier are installed and running.

Note: PPM can be integrated with multiple instances of Service Manager, if those instances are at the same version. (See the *System Requirements and Compatibility Matrix*.) However, the procedures in this section must be performed identically on the multiple Service Manager instances.

Note: If you already integrated Service Manager with PPM, and you upgrade the integration solution to PPM 9.12 or later, copy and rename two web service objects in Service Manager as follows to make sure that the integration will continue to work:

- Change -> ChangePPMIntALM

- ChangeTask -> ChangeTaskPPMIntALM

How to copy and rename the web service objects in Service Manager

1. Log on to Service Manager as system administrator.

2. Select Menu **Navigation > Tailoring > Web Services > Web Service Configuration**.

3. Type `Change` in the **Service Name** field, and click **Search**.

4. Select **Change**.

5. Modify the **Object Name** field from `Change` to **ChangePPMIntALM**.

6. Click **Add**.

   This will create or add a new object name called **ChangePPMIntALM** in the Change Web Service based on the OOTB object `Change`.

7. Repeat the above steps to create or add a new object called **ChangeTaskPPMIntALM** based on the OOTB object `ChangeTask`.

Fields related to the integration are exposed in the ChangePPMIntALM web service when using PPM Center version 9.12 and later.

Integration of Service Manager with PPM requires specific configuration of Service Manager, as described in the following sections.

1. "Configuring the Change Management Module" on the next page

2. "Associating New Display Actions to the New Processes Loaded" on page 282

3. "Reconfiguring the ChangeManagement Module" on page 283

4. "Configuring Browsing of Service Manager Changes from a URL" on page 286

5. "Modifying the cm3r.pre.add Trigger" on page 288

6. "Performing the Mass Update Procedure for the Existing Records" on page 289

7. "Updating Service Manager Application Server with Idle Session Timeout and Modifying the falcon User Profile" on page 290

Before starting this procedure, make sure you review the warning in the section "Introduction to Integrating PPM Requests with Service Manager Changes, Using ALM" on page 274.

Configuring the Change Management Module

1. Add **sysmodtime** and **orig.date.entered** fields to the **cm3r** table.

   > **Note:** If you are on Service Manager version 9.20 or later, these two fields may already exist in your OOTB sample data. You can skip adding these two fields.

   a. Type `dbdict` in the Service Manager command line and press **Enter**.

      The Database Dictionary window opens.

   b. Type `cm3r` in the File Name field and click **Search**.

   c. Select **cm3r** in the File Name section.

      The `cm3r` dbdict displays.

   d. Click the **Keys** tab.

   e. Scroll down to the bottom of the form and place your cursor in the empty key.

   f. Click **New Field/Key**.

      The key.window page opens.

   g. Add a new key to the database table. The key should be of the type "Not Null" and should contain the **sysmodtime** field. If the Service Manager system uses another field to contain the last modified date/time of the record, use that field instead.

      i. In the key.window page, from the **Type** drop-down list, select **no nulls**.

      ii. Type `sysmodtime` in the Fields textbox.

      iii. Click **Add Key**.

   h. Add a new key to the database table. The key should be of the type "Not Nulls" and should contain the **orig.date.entered** field, for use in the initial load mode (for more information, see the **initial-load-state** adapter attribute in "Configuring the Service Manager Adapter Attributes" on page 305).

      i. In the key.window page, from the **Type** drop-down list, select **no nulls**.

      ii. Type `orig.date.entered` in the Fields textbox.

      iii. Click **Add Key**.

2. Make sure that the required fields are exposed through the Change Management Web Services:

a. For Service Manager version 7.01, select

Menu Navigation > Tailoring > WSDL Configuration.

For Service Manager version 7.10, version 9.20 or later, select

Menu Navigation > Tailoring > Web Services > WSDL Configuration.

For Service Manager version 9.40 or later, select

Menu Navigation > Tailoring > Web Services > Web Service Configuration.

b. Type `Change` in the Service Name field and click **Search**.

For PPM version 9.10 or 9.11, select **Change** in the **Object Name** section.

For PPM version 9.12 or later, select **ChangePPMIntALM** in the **Object Name** section.

c. Click the **Fields** tab and add fields with captions and types to the table as follows:

New Field	New Caption	Type
sysmodtime	sysmodtime	DateTimeType
header,orig.date.entered	origDateEntered	DateTimeType

d. For Service Manager version 9.20 or later, change existing WSDL captions as follows:

Field	Original Caption	New Caption
header,number	ChangeID	changeNumber
header,brief.description	Title	briefDescription

3. Import the unload file provided with PPM to set up the Change Management Web service for the integration with PPM.

a. Type `db` in the Service Manager command line and press **Enter**.

The Database Manager window opens.

b. Select **More > Import/Load**.

The Service Manager File Load/Import screen opens.

c. For the File Name field, browse to select the file to load.

- For integration with PPM version 9.10 and 9.11, the load file is located at:

  `<PPM_Home>\conf\sdi\serviceManagerFiles\sm_operations`

- For integration with PPM version 9.12 or later, the load file is located at:

  `<PPM_Home>\conf\sdi\serviceManagerFiles\PPMIntALMWebService.unl`

Where *<PPM_Home>* represents the path where the PPM instance is installed. For example: `xyzserver/E/PPMServer`.

d. Click **Load FG**.

A message stating that the unload file is loaded displays.

e. Click **Back**.

Associating New Display Actions to the New Processes Loaded

1. Select **Menu Navigation > Tailoring > Document Engine > Objects.**

   The Object Definition screen opens.

2. Type `cm3r` in the File Name field and press **Enter**.

3. Select **cm3r** from the Object Name list.

   The Object Definition for `cm3r` opens.

4. Place your cursor in the Default State field (its value should be `change.view`) and click the **Find** icon.

   The State Definition screen opens.

5. Scroll down to the bottom of the table, add two new Display Actions to the table as follows:

New Display Action	New Process Name	Condition	Save First
checkretract [a]	ccm.check.retract	true	
checkapproval	ccm.check.approval	true	

a. After adding the checkretract Display Action, click Save. Another available line is added.

6. Click **Save**.

7. Click **OK**.

   The screen returns to the `cm3r` Object Definition screen.

8. Click **Cancel**.

   The screen returns to the Object Definition search page.

9. Type `cm3t` in the File Name field and click **Search**.

○ If the `cm3t` Default State does not have the same value as the `cm3r` default state, then repeat "Associating New Display Actions to the New Processes Loaded" on the previous page through "Associating New Display Actions to the New Processes Loaded" on the previous page for the `cm3t` object.

○ If they share the same Default State, click **Cancel**, then click **Back** to leave the Object Definition screen.

Reconfiguring the ChangeManagement Module

Follow instructions in this section to reconfigure the ChangeManagement Web service, then update the ChangeManagement external access record.

Reconfigure the ChangeManagement WSDL

Do the following,

1. For Service Manager version 7.01, select

   Menu Navigation > Tailoring > WSDL Configuration.

   For Service Manager version 7.10, version 9.20 or later, select

   Menu Navigation > Tailoring > Web Services > WSDL Configuration.

2. In the **Service Name** field, type `Change` and click **Search.**

3. For PPM version 9.10 or 9.11, select **Change**.

   For PPM version 9.12 or later, select **ChangePPMIntALM**.

4. Add the following two new actions to the table for the `cm3r` object:

New Allowed Action	New Action Name
checkapproval	CanApprove
checkretract	CanRetract

5. Click **Save**.

6. Click **OK** to return to the External Access Definition screen.

7. Clear all data, type `cm3t` in the Name field and press **Enter.**

8. For PPM version 9.10 or 9.11, select **ChangeTask**.

For PPM version 9.12 or later, select **ChangeTaskPPMIntALM**.

9. Repeat step 4 through step 6 for the `cm3t` object.

Add ppmFields Structure to the cm3r dbdict Table

To add ppmFields structure to the `cm3r` dbdict,

1. Type `dbdict` in the Service Manager command line and press **Enter**.

   The Database Dictionary screen opens.

2. Type `cm3r` in the File Name field and click **Search**.

3. Select **cm3r** in the File Name section.

4. Click on the Name column of the **descriptor** row.

5. Click **New Field/Key**.

   The field.window pops up.

6. Type `ppmFields` in the Name field.

7. Select **structure** from the Type drop-down list.

8. Click **Add**.

   The **ppmFields** structure is added to the `cm3r` table.

9. Locate the **ppmFields** structure row you just added, with your cursor on the row, click **New Field/Key**.

   The field.window pops up again, with the Structure field value of `ppmFields`.

10. Type `ppmURL` in the Name field.

11. Select **character** from the Type drop-down list.

12. Click **Add**.

13. Locate the **ppmFields** structure row, with your cursor on the row, click **New Field/Key**.

14. Type `requestModifiedDate` in the Name field.

15. Select **date/time** from the Type drop-down list.

16. Click **Add**.

17. Double click the **ppmFields** structure row and provide values for the ppmFields structure as

follows, then click **OK**.

> **SQL Name:**
> **SQL Type:**
> **SQL Table:** a3 (**Note:** This may change depending on how many other complex data types are pushed out as a separate array.)

18. Double click the **ppmURL** field row and provide values for the **ppmURL** field as follows, then click **OK**.

> **SQL Name:** PPMURL
> **SQL Type:** VARCHAR2(400)
> **SQL Table:** a3 (**Note:** This may change depending on how many other complex data types are pushed out as a separate array.)

19. Double click the **requestModifiedDate** field row and provide values for the **requestModifiedDate** field as follows, then click **OK**.

> **SQL Name:** RequestModifiedDate
> **SQL Type:** DATE
> **SQL Table:** a3 (**Note:** This may change depending on how many other complex data types are pushed out as a separate array.)

20. Click on the **SQL Tables** tab, and on the next available row, enter a new entry with the following values:

> **Alias:** a3
> **Name:** CM3RA3
> **Type:** oracle10

21. Click **OK**.

22. Select **SM Alters** when prompted.

Update the ChangeManagement External Access Record

Do the following:

1. For Service Manager version 7.01, select
   Menu Navigation > Tailoring > WSDL Configuration.

   For Service Manager version 7.10, version 9.20 or later, select
   Menu Navigation > Tailoring > Web Services > WSDL Configuration.

The External Access Definition screen opens.

2. Type Change in the **Service Name** field and click **Search.**

3. For PPM version 9.10 or 9.11, select **Change** in the object.name section.

   For PPM version 9.12 or later, select **ChangePPMIntALM.**

4. Click the **Fields** tab in the External Access Definition section.

5. Scroll down to the bottom of the form, and add two new fields with the following values:

New Field	New Caption	Type
ppmFields,ppmURL	ppmURL	StringType
ppmFields,requestModifiedDate	reqModDate	DateTimeType

6. Click **Save**.

7. Click **OK.**

Configuring Browsing of Service Manager Changes from a URL

This procedure enables RFCs in PPM to be updated with URL links to the corresponding changes in Service Manager, so that PPM users can easily jump directly to those changes.

To enable this capability, configure Service Manager as follows:

1. Log on to Service Manager as system administrator.

2. Confirm the **WebServer URL** field in the System Definition Information record.

   a. Select **Menu Navigation > System Administration > Base System Configuration > Miscellaneous > System Information Record.**

      The System Information Definition screen opens.

   b. In the form, click the **Active Integrations** tab.

   c. Confirm that the **WebServer URL** field in the WebServer Information section defines the Service Manager Web Client URL.

If it does not, type the URL of the Service Manager Web server configured for Web access in the **WebServer URL** field. For example:

```
http://<Host>:<Port>/sm/index.do
```

d. Click **OK** to save the System Information Definition.

e. Click **Cancel** to exit.

3. Add the **record.url** field to the cm3r table.

a. Type dbdict in the Service Manager command line and press **Enter**.

The Database Dictionary window opens.

b. Type cm3r in the File Name field and click **Search**.

c. Select **cm3r** in the File Name section.

d. Click on the Name column of the **descriptor** row.

e. Click **New Field/Key**.

The field.window pops up.

f. Type record.url in the Name field.

g. Select **character** from the Type drop-down list.

h. Click **Add**.

The **record.url** field is added to the cm3r table.

i. Locate the new **record.url** field row and double click it, in the field.window that pops up, enter the following:

Field Name	Value
SQL field name	RECORD_URL
SQL data type	VARCHAR
SQL data length	400

j. Click **OK**.

4. Add **record.url** to the Change Management Web service.

a. For version 7.01, select **Menu Navigation > Tailoring > WSDL Configuration.**

For version 7.10, version 9.20 or later, select
Menu Navigation > Tailoring > Web Services > WSDL Configuration.

The External Access Definition screen opens.

b.   Type Change in the **Service Name** field and click **Search.**

c.   For PPM version 9.10 or 9.11, select **Change** in the object.name section.

For PPM version 9.12 or later, select **ChangePPMIntALM** in the object.name section.

d.   Click the **Fields** tab in the External Access Definition section.

e.   On the next available line, add a field with caption and type to the table as follows:

New Field	New Caption	Type
record.url	recordUrl	StringType

f.   Click **Save**.

g.   Click **OK**.

Modifying the cm3r.pre.add Trigger

To modify the cm3r.pre.add trigger,

1.   Type db in the Service Manager command line and press **Enter**.

     The Database Manager window opens.

2.   Type triggers in the Table field and click **Search**.

3.   Type cm3r.pre.add in the Trigger Name field and click **Search**.

4.   Copy and paste the following JavaScript into the Script text box:

```
_rec_number = record.number;
var _filename = "cm3r";
var _query = "number=\"" + _rec_number+ "\"";
var _title = "Change number " + _rec_number;
var _link=system.library.urlCreator.getURLFromQuery (_filename, _query, _
title);
record.record_url =_link;
```

5.   Click **Save**.

6.   Click **Compile** to check for any errors and compile. No errors should occur.

Now, any new record will have the **record.url** field populated. However, as described in the following steps, you will need to perform the Mass Update procedure to populate the **record.url** field for existing records.

7. Click **OK**.

Performing the Mass Update Procedure for the Existing Records

To perform the Mass Update procedure for the existing records,

1. Select **Menu Navigation > Change Management > Search Changes.**

2. Click **Search** to display a list of change records.

3. Select the records you want to update.

4. Perform the Mass Update procedure:

    a. Click **Mass Update** in the toolbar to start updating the listed records.

       Database Manager displays the initial form again, but with different options (buttons).

    b. Do *not* specify values in any field. Click **Complex Update.**

       Database Manager displays the Instructions screen.

    c. Type the following expression in the **Instructions for action on EACH RECORD** input field:

       ```
       record.url in $file=jscall("urlCreator.getURLFromQuery","cm3r","number=\"" +
       number in $file + "\"","Change number " + number in $file)
       ```

5. Click **Execute**.

   For each record updated, this step sets the **record.url** field based on the Web server URL entered in System Information Record. Then this step returns terminal control to you and displays the following message:

   ```
   <n> records updated in the cm3r file.
   ```

   where <n> is the number of records updated.

6. Once the Mass Update procedure is complete, confirm that the **record.url** field is populated for the existing Change tickets.

7. (Optional) To browse directly from the Web server using a URL that includes a ticket ID, disable

the `querySecurity` parameter by using the instructions in the following Service Manager Help topic:

"Web parameter: sc.querySecurity"

8. (Optional) Add the following line to the conversion script file in PPM for converting Service Manager changes to PPM requests:

```
ppmRFC.addURLReference(serviceManagerRFC.get ("record.url"), <display text for
URL reference>);
```

where *<display text for URL reference>* represents the text to be displayed for the link to the URL.

For information about copying the sample conversion file provided with ALM and revising the copy, see "Configuring the Service Manager Adapter Converter Property (Script)" on page 308.

For information about the addURLReference function, see "ppmRFC Object" on page 310.

Updating Service Manager Application Server with Idle Session Timeout and Modifying the falcon User Profile

How to update Service Manager Application Server with idle session timeout

1. Update the `sm.ini` file with the following parameter to drop idle connections:

   ```
   Webservices_sessiontimeout:300
   ```

2. Restart the Service Manager Windows services.

If you integrate Service Manager with PPM version 9.14 or later, make sure you modify the falcon user profile in Service Manager.

1. Type `operator` in the Service Manager command line and press **Enter**.

   The Search Operator Record screen opens.

2. Type `falcon` in the Login Name field and click **Search**.

   The Operator Record screen displays.

3. Go to the **Login Profiles** tab.

4. In the Login Profile section, set Time Zone and Date Format as follows:

- Set **Time Zone** to your location (optional)

- Click pull-down button for the **Date Format** field and select **mm/dd/yyyy**.

5. Click **Save.**

6. Click **OK.**

Configuring PPM for Integration with Service Manager

To configure PPM for integration with Service Manager, do the following:

1. Install and configure ALM module.

   a. (Recommended but not required) Set your PPM Server to **Restricted** mode.

      i. Stop your PPM server.

      ii. Run the following command:

      ```
      cd c:\ppm\bin
      sh ./setSErverMode.sh RESTRICTED
      ```

      iii. Restart your PPM server.

      In cluster environment, make sure there is only one administrator performing the configuration tasks.

   b. (Optional) Deploy the ALM bundle by running the following command:

      ```
      cd c:\ppm\bin
      sh ./kDeploy.sh -i ALM
      ```

      The "`Deployment ALM has been successfully installed.`" message displays.

      Note: You can use the old version of ALM bundle even if PPM is upgraded to a newer version.

   c. Modify the Contact User Data user data.

      i. From the menu bar, select **Open > Administration > Open Workbench.**

      The PPM Workbench opens.

ii. From the shortcut bar, select **Configuation > User Data**.

The User Data Workbench opens.

iii. Click **List**, locate Contact User Data in the **Results** tab, and click **Open**.

iv. On the **Fields** tab, click **New**.

The Field: New dialog pops up.

v. Provide values for the new field as follows:

Field Prompt: Location
Token:LOCATION
Description: Added for ALM integration
Validation: Text Field - 40
User Data Col:USER_DATA1
Display:Yes

vi. Click **OK.**

vii. Click **Save.**

d. Modify the CRT - Priority - Enabled validation.

i. From the PPM Workbench shortcut bar, select **Configuration > Validations**.

The Validation Workbench opens.

ii. Click **List**, locate CRT - Priority - Enabled in the **Results** tab, and click **Open**.

The Validation : CRT - Priority - Enabled dialog pops up.

iii. Click **New**.

The Add Validation Value dialog pops up.

iv. Add new validation values as follows:

New Code	New Meaning
MEDIUM	Medium
IMMEDIATE	Immediate
PLANNING	Planning

v. Click **OK.**

vi. Click **Save.**

e. Create ALM related security groups.

  i. From the PPM Workbench shortcut bar, select **Sys Admin > Security Groups**.

   The Security Group Workbench opens.

  ii. Click **New Security Group.**

  iii. Type `ALM - Application Developer` in the Name field and provide other values as necessary.

  iv. Click **OK.**

  v. Repeat step *iii* and step *iv* to create the following security groups:

   ALM - Applications Development Manager
   ALM - CAB group (Change Advisory Board)
   ALM - Change Builder
   ALM - Change Manager
   ALM - Customer
   ALM - IT Executive Board
   ALM - Independent Tester
   ALM - Operations Manager
   ALM - QA Manager
   ALM - Release Manager
   ALM - SOX - System Owner

f. Remove unnecessary Request Header Type Field Groups.

  i. From the PPM Workbench shortcut bar, select **Demand Mgmt > Request Header Type.**

   The Request Header Type Workbench opens.

  ii. Click **List**, locate ALM - Request for Change (RFC) in the **Results** tab, and click **Open**.

  iii. Click **Field Groups.**

  iv. Uncheck the following field groups:

   CMDB Application
   MAM Impact Analysis
   QC/ALM Info

  v. Click **OK.**

  vi. Click **Save.**

  vii. Click **OK.**

g. Create a PPM integration user account.

      i. From the PPM Workbench shortcut bar, select **Sys Admin > Users.**

      The Users Workbench opens.

     ii. Click **New User** and provide values for the new user as follows:

      Username: SM_PPM_INT
      First Name:SM_PPM
      Last Name: Interface
      Password: <*Your_Password*>
      New password on login:No
      Application Licenses:Demand Management

    iii. Click **Save.**

    iv. Go to the **Security Groups** tab, click **New** and add the following security groups:

      PPM User
      PPM Demand Manager

     v. Click **Save.**

    vi. Click **OK** twice.

  h. Set your PPM server back to Normal mode.

     i. Stop your PPM Server.

    ii. Run the following command:

```
cd c:\ppm\bin
sh ./setServerMode.sh NORMAL
```

 i. Modify the server.conf file.

     i. Open the server.conf file in Notepad.

    ii. Append the following directives and save the file:

```
com.kintana.core.server.MAC_LOG_SEVERITY=0
```

    iii. Run the following command:

```
cd bin
sh ./kUpdateHtml.sh
```

2. Encrypt the SM_PPM_INT user's password.

  a. Run the following command:

```
cd c:\ppm\bin
sh ./kEncrypt.sh <PPM_User_Password>
```

b. Save the resulting encrypted string.

You would need to specify the PPM Integration user password (enclosed with the `#!#` character delimiters) in the "`*.settings`" files later.

3. Setup configuration files for Service Manager=>PPM direction.

a. Run the following command:

```
cd c:\ppm\conf\sdi
cp serviceManger-adapter.settings1 serviceManager-adapter.settings
```

b. Edit the `serviceManager-adapter.settings` file.

   i. In the `<connector>` section, set the following:

   - `timzeZone=<Same_As_PPM_SM_TIME_Zone_settings>`

     For example, `timeZone=US/Pacific; preferred GMT-08:00 format`.

   - `userName=<SM_USER_ID>`

     For example, `userName=SM_PPM_INT`

     Make sure this account exists on Service Manager application.

   - `password=<CLEAR_TEXT_SM_USER_ID_PASSWORD>`

     For example, `password=<plain SM password, or encrypted SM password>`

     HPE recommends that you use encrypted SM password.

   - `serviceUrl=http://<SMC_WEB_SERVICES_URL>:<PORT>/sc62server/PWS/`

   ii. In the `<convertor>` section, change `scripts=convertSMToPPM.js.sample` to `scripts=convertSMToPPM.js`.

   iii. In the `<sender>` section, set the following:

   - `userName=<SMC_USER_ID>`

     For example, `userName=SM_PPM_INT`

     Make sure this account exists on PPM application.

   - `password=<![CDATA[insert encrypted password string]]>`

     For example, `password=<![CDATA[<encrypted password`

     > **Note:** Do not enclose the password with `#!#`.

   - `requestType=ALM - Request For Change (RFC)`

> **Note:** If you changed the request type for this integration, make sure to change the request type name here accordingly.

4. Setup configuration files for PPM=>Service Manager direction.

   a. Run the following command:

   ```
   cd c:\ppm\conf\sdi
   cp ppm-sm-adapter.settings1 ppm-sm-adapter.settings
   ```

   b. Edit the `ppm-sm-adapter.settings` file.

      i. In the `<connector>` section, set the following:

         - `requestType=ALM - Request For Change (RFC)`

           > **Note:** If you changed the request type for this integration, make sure to change the request type name here accordingly.

         - `userName=<SMC_USER_ID>`

           For example, `userName=SM_PPM_INT`

           Make sure this account exists on PPM application.

         - `password=<![CDATA[insert encrypted password string]]>`

           For example, `password=<![CDATA[<>encrypted password string>]>`

           > **Note:** Do not enclose the password with "#!#".

         - `requestStatusNames=<List_of_comma-separated_PPM_Request_statuses>`

           The value is the statuses of requests to be synchronized to SM. The statuses are separated by commas.

           For example, `requestStatusNames=new, in progress, on hold, complete`

      ii. In the `<convertor>` section, change `scripts=convertPPMtoSM.js.sample` to `scripts=convertPPMtoSM.js`.

      iii. In the `<sender>` section, set the following:

         - `userName=<SMC_USER_ID>`

           For example, `userName=SM_PPM_INT`

           Make sure this account exists on PPM application.

         - `password=<CLEAR_TEXT_SM_USER_ID_PASSWORD>`

           For example, password=<plain SM password>

- timzeZone=<*Same_As_PPM_SM_TIME_Zone_settings*>

  For example, `timeZone=US/Pacific;` preferred GMT-08:00 format.

- serviceUrl=http://<*SMC_WEB_SERVICES_URL*>:<*PORT*>/sc62server/PWS/

c. Edit the js files in the `c:\ppm\sdi\ppm-sm-adapter.ext` directory: `convertPPMToSM.js` and `convertSMToPPM.js`.

Add the following value:

i. Find the following line:

```
serviceManagerRFC.setField(PPMURL,"http://localhost:<PPM_
PORT>/itg/web/knta/crt/RequestDetail.jsp?REQUEST_ID=" + ppmRFC.get
("REQ-REQUEST_ID"));
```

ii. Replace <*PPM_PORT*> with your port number.

iii. Save the modified file.

d. Compile the Web Services stub.

> **Caution:** You must recompile the Web Services stub if you modified the `*.settings`, `convertRequest.js` file, or tailored Service Manager web services schema.

i. Open a Command Prompt window by typing `cmd` in the **Start > Run** dialog and click **Open**.

ii. Run the following command:

```
cd c:/ppm/bin/sdi
```

iii. Run the following command:

kGenerateServiceCenterStub.bat http://<*SMC_WEB_SERVICES_ URL*>:<*PORT*>/sc62server/PWS/ChangeManagement?wsdl <KINTANA_SERVER_NAME>

For example,

```
kGenerateServiceCenterStub.bat http://sm28p-
01.infra.mms:13087/sc62server/PWS/ChangeManagement.wsdl app
```

The following message displays after the stub is generated:

```
Stub generated successfully
```

5. (Optional) For cluster setup, do the following:

a. Compress the entire `c:\PPM` folder into a zip file and copy it onto other members the first time.

b. For subsequent changes, copy the following folders to all cluster members at the same

directory location:

- `c:\ppm\sdi-persistency` folder

- `c:\ppm\conf\sdi` directory

c. If there are subsequent changes, do not forget to regenerate the Web service stubs on all cluster members.

For more information about generating Web service stubs, see "Generating Web Service Stubs " on page 301.

6. Restart your PPM Servers.

7. Validate the following:

a. The PPM ALM Startup Service is running.

i. In PPM Center, from the menu bar, select **Open > Administration > Schedule Services.**

ii. The Schedule Services page opens.

iii. Check that the ALM Startup service is running.

b. Check the `c:\ppm\sdi-persistency` folder.

i. You should see `.ser`, `.properties`, `.log`, `.data` files related to `serviceManager-adapter`.

ii. You should see `.ser`, `.properties`, `.log`, `.data`, files related to `ppm-sm-adapter`.

iii. If files are missing, the possible error is with the "`.settings`" file.

iv. Date for these files should be recent.

> **Tip:** This folder is generated automatically by the system after the configuration completes. It stores the real configuration files and JS files to be called by the system later. If you modified any configuration file or JS file in the *<PPM_Home>*`/conf/` folder after this `sdi-persistency` folder is created, for the changes you made to take effect, make sure you delete the `sdi-persistency` folder before restarting your PPM Servers.

(Optional) Setting Logs for Debugging Purpose

1. Go to the *<PPM_Home>*`/conf/sdi/ppm-sm-adapter.ext` directory. Change the file name of `convertPPMToSM.js.sample` to `convertPPMToSM.js`.

2. Open the `convertPPMToSM.js` file in a text editor and edit it.

    a. Comment out the line beginning with "`var REQMODDATE`".

    b. Add the following line right after it:

```
var SYNCSTATUS = "syncStatus"; //The syncStatus field
```

> **Note:** You may need to add the syncStatus field and expose it in the WSDL before adding the above line.

    c. In the function `convert`, add the following line:

```
logger.error("PPM -> SM Convert " + ppmRFC.get("REQD-SD_TICKET_ID"));
```

    d. Comment out the line beginning with
       "`serviceManagerRFC.setField(REQMODDATE`"

    e. Add the following line right after it:

```
serviceManagerRFC.setField(SYNCSTATUS,ppmRFC.get("REQ-DESCRIPTION"));
```

    f. In the function `preFilter`, add the following line:

```
logger.error("PPM -> SM preFilter " + ppmRFC.get("REQD-SD_TICKET_ID"));
```

    g. In the function `postFilter`, add the following line:

```
logger.error("PPM -> SM postFilter ");
```

    h. In the function `convertDate`, replace the following line:

```
format.setTimeZone(java.util.TimeZone.getTimeZone ("PST"));
```

       with the following:

```
format.setTimeZone(java.util.TimeZone.getDefault());
```

    i. Save the file.

3. Go to the *<PPM_Home>*`/conf/sdi/serviceManager-adapter.ext` directory, change the file name of `convertSMToPPM.js.sample` to `convertSMToPPM.js`.

4. Open the `convertSMToPPM.js` file in a text editor and edit it.

    a. In the function `convert`, add the following line:

```
logger.error("SM -> PPM convert " + serviceManagerRFC.get
("header.changeNumber"));
```

    b. In the function `preFilter`, add the following line:

```
logger.error("SM -> PPM preFilter " + serviceManagerRFC.get
("header.changeNumber"));
```

   c. In the function `posteFilter`, add the following line:

```
logger.error("SM -> PPM postFilter");
```

   d. Save the file.

Validating the Integration is working

To validate whether the integration is working,

1. Log on to Service Manager via the Service Manager thick client or the Service Manager URL. (You may need Service Manager App owner or TSC to help with this.)

   a. On Service Manager side, create a Change Request using the integration Service Manager user ID, for example, `SM_PPM_INT`.

   b. Wait about 5 minutes.

2. Log on to PPM on a different browser.

   a. On PPM side, search for change request of the ALM - Request for Change (RFC) request type.

   b. If the Service Manager=>PPM direction works, you should see the corresponding PPM RFC request.

   c. Populate the required fields on the above RFC request and click **Submit.**

   d. Wait about 10 minutes.

3. From the Service Manager client,

   a. If the PPM=>Service Manager direction works, you should see the following field/values that are mapped by default in the `c:\ppm\conf\sdi\ppm-sm-adapter.ext\convertRequest.js` file.

   By default, these fields are not visible on the Service Manager request page. So, you may need to make the fields visible or login to the DB schema and look for them.

   - PPMURL — http://localhost:8080/itg/web/knta/crt/RequestDetail.jsp?REQUEST.jsp

   - REQMODDATE — Today's date

   - There may be other custom mappings defined in that file as well.

   b. Alternatively, you can use SQL client to check the Service Manager DB schema.

Generating Web Service Stubs

> **Note:** Perform this procedure for new installations of ALM content bundle and after any upgrade of PPM.

In Service Manager, you can modify which fields are available through Web services. Each time you modify these settings, a new Web Services Description Language (WSDL) descriptor is created. In the PPM Server, you must regenerate the Web service stubs from the new descriptor.

To generate the stubs:

1. Navigate to the `<PPM_Home>/bin/sdi` directory on the PPM Server.

   where `<PPM_Home>` represents the path where the PPM instance is installed. For example: `xyzserver/E/PPMServer.`

2. Run the following script:

   `./kGenerateServiceCenterStub.sh <wsdl-url> <PPM_Server_Name>`

   where

`<wsdl-url>`	represents the Service Manager WSDL URL for Change Management.
`<PPM_Server_Name>`	represents the subdirectory of `<PPM_Home>/server` specified during installation as the value for the `KINTANA_SERVER_NAME` parameter in the `server.conf` file. (This value is not necessarily the actual host name of the server.) When generated, the stubs will be placed in this subdirectory.

   The specific script is in the following format:

   `./kGenerateServiceCenterStub.sh http://<Host>:<Port>/sc62server/PWS/ChangeManagement?wsdl <PPM_Server_Name>`

   For example:

   `./kGenerateServiceCenterStub.sh http://ServManager:13080/sc62server/PWS/ChangeManagement?wsdl kintana`

3. Verify that the stubs are now in the following directory:

   `<PPM_Home>/server/<PPM_Server_Name>/deploy/itg.war/WEB-INF/classes/`

> **Note:** The stub generation might fail if the host is not resolvable by name. To resolve the host by

name, make an entry in the `hosts` file for the machine you want to connect. See the documentation for your operating system for the location of the `hosts` file.

Configuring the Service Manager Adapter Configuration File

The Service Manager adapter configuration file is an XML file in PPM that enables integration of Service Manager with PPM and converts Service Manager changes to PPM requests. The configuration file consists of the following components, each with its own attributes or properties (see Figure 4-1):

- General settings for the adapter itself, such as its name and the name of the Service Manager application in which the changes are created.

- Connector between Service Manager and the adapter.

- Converter of changes in the Service Manager data model to generic requests in the PPM data model. The converter calls the scripts that define the field mapping and filter functions.

- PPM sender, which sends the converted and filtered requests to PPM.

  Note: If PPM is operating in a clustered server configuration, share or copy the *<PPM_Home>*/sdi-persistency directory and the *<PPM_Home>*/conf/sdi directory among all the servers in the cluster.

The following sections describe how to configure the Service Manager adapter configuration file and the scripts called by its converter, and how to modify copies of the provided files while preserving the originals.

Location, Naming, and Structure of Service Manager Adapter Configuration Files

Each Service Manager adapter configuration file must follow specific conventions for its location, naming, and structure, as described in the following sections.

Location and Naming of the Service Manager Adapter Configuration File

The adapter configuration files are located in the `<PPM_Home>`/conf/sdi directory of the PPM Server. This directory contains the following:

- Configuration file for the adapter (or one for each adapter if there are multiple adapters).

  In this configuration file, you define a name for the adapter. The configuration file must have a `.settings` file extension.

- Subdirectory (or one for each adapter configuration file if there are multiple files).

  The subdirectory contains the conversion scripts, which are called by the converter to map the fields between and to filter the changes and requests. The name of the subdirectory must be the same as the `<adapter name>` (for a description, see "Structure of the Service Manager Adapter Configuration File" below and "Configuring the Service Manager Adapter Attributes" on page 305) followed by `.ext`.

  For example, if the adapter name is `serviceManager-adapter,` the `<PPM_Home>`/conf/sdi directory must contain a subdirectory named `serviceManager-adapter.ext` to hold all the conversion script files for the adapter.

Structure of the Service Manager Adapter Configuration File

ALM provides, as a template, a default Service Manager adapter configuration file named `serviceManager-adapter.settings1.` The adapter file you configure and use must have a `.settings` file extension.

Copy the default adapter file (to preserve the original), and rename the copy with a `.settings` file extension and, if desired, a different file name.

As detailed in subsequent sections, the adapter file has the following basic structure, including adapter attributes, and properties for its connector, converter, and sender:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<settings>

<adapter adapter-name="<adapter name>">
    <service-desk-application><SD application>
        </service-desk-application>
    <number-of-tickets><number of tickets>
```

```
        </number-of-tickets>
    <polling-schedules><schedule></polling-schedules>
    <polling-frequency><frequency></polling-frequency>
    <initial-load-state><date></initial-load-state>

    <request-types>
    <request-type level="1">
    <polling-operation>

    <connector>
    <connector-type>serviceManagerChange</connector-type>
        <properties>
            idProperty=
            lastUpdatedPropertyForQuery=
            creationDatePropertyForQuery=
            lastUpdatedPropertyForResult=
            creationDatePropertyForResult=
            keyMethodName=
            timeZone=
            wsDateFormatPattern=
            queryDateFormatPattern=
            userName=
            password=
            serviceUrl=
        </properties>
    </connector>

    <converter>
    <converter-type>scriptConverter</converter-type>
        <properties>
            scripts=<convert1>.js,<convert2>.js,...
        </properties>
    </converter>

    </polling-operation>
    </request-type>
    /request-types>
    <sender>
    <sender-type>PPMSender</sender-type>
        <properties>
            userName=
            password=
            requestType=
            updateRequest=
            ticketIdFieldName=
            sdSystemFieldName=
            staticFieldNames=
        </properties>
    </sender>

</adapter>
```

```
</settings>
```

> **Caution:** Do not delete or change the values provided for `<connector-type>`, `<converter-type>`, or `<sender-type>`.

The following sections describe how to configure the adapter attributes, the connector properties, the converter property (`scripts`), and the sender properties.

Configuring the Service Manager Adapter Attributes

Specify the adapter attributes of the Service Manager adapter configuration file, such as the adapter name and the service desk application, as described in the table below.

Table 4-1. Service Manager adapter attributes

Attribute Name (*Required)	Description	Default Value
*adapter-name	Logical name that represents the adapter name on the client machine. For example: serviceManager-adapter This name is also used for the scripts (`.ext`) directory. (See "Location and Naming of the Service Manager Adapter Configuration File" on page 303.)	(None)
*service-desk-application	Unique logical name for the service desk system you are using. For example: Service Manager	(None)
number-of-tickets	Number of changes that the adapter processes at a time.	50
polling-schedules	Times of day that the adapter polls Service Manager for changes, formatted as a list of cron expressions separated by the new line character. For example: 30 * * * * <new line> 0 * * * *	(None)
polling-frequency	Frequency (in seconds) that the adapter polls Service Manager for changes.	If polling-schedules and polling-frequency are unspecified, then the

Table 4-1. Service Manager adapter attributes, continued

Attribute Name (*Required)	Description	Default Value
		default polling-frequency is 30 seconds, starting when you restart the PPM Server.
initial-load-state	Earliest creation date and time of changes the adapter retrieves from Service Manager, in the following format: MM/dd/yy HH:mm:ss z For example: 10/19/08 21:30:00 EST After completion, the adapter does not retrieve any new or updated changes (per the polling-schedules and polling-frequency attributes) until this attribute is commented out.	(None)

Configuring the Service Manager Adapter Connector Properties

Specify the properties for the connector section of the Service Manager adapter configuration file as described in the table below.

Table 4-2. Service Manager adapter connector properties

Property Name (*Required)	Description	Default Value
*idProperty	Property name of the ID field in the instance returned from the Service Manager Web service.	(None)
*lastUpdatedPropertyForQuery	Property name of the **last-update** field used to query the Service Manager Web service (the field name used in an expert search on the Service Manager client machine).	(None)
*creationDatePropertyForQuery	Property name of the **creation-date** field used to query the Service Manager Web service.	(None)

Table 4-2. Service Manager adapter connector properties, continued

Property Name (*Required)	Description	Default Value
*lastUpdatedPropertyForResult	Property name of the **last-update** field in the instance returned from the Service Manager Web service (usually the field name exposed as API).	(None)
*creationDatePropertyForResult	Property name of the **creation-date** field in the instance returned from the Service Manager Web service.	(None)
*keyMethodName	Name of the method for request keys (usually the ID field name).	(None)
*timeZone	Time zone, used for converting the last updated time of a request from Service Manager. Use the same time zone as the Service Manager server. The format can be GMT+<X> or GMT–<X>, where <X> is the offset in hh:mm format from GMT. For example, GMT–07:00. However, to handle Daylight Saving Time, use an area time zone instead of specifying a time relative to GMT.	(None)
*wsDateFormatPattern	Date format used in the Service Manager Web service answer. For available formats, see the following URL: http://java.sun.com/j2se/1.4.2/docs/api/java/text/SimpleDateFormat.html	(None)
*queryDateFormatPattern	Date format used for querying the Service Manager system (as used in the UI expert search). Make sure this value is consistent with the Service Manager server date pattern. For available formats, see the following URL: http://java.sun.com/j2se/1.4.2/docs/api/java/text/SimpleDateFormat.html	(None)
*userName	User name in the Service Manager system that PPM uses to connect to Service Manager. This user name must include only single-byte characters. This user must have full access to the Change Management module in Service Manager.	(None)
*password	Password in the Service Manager system that PPM uses to connect to Service Manager.	(None)

Table 4-2. Service Manager adapter connector properties, continued

Property Name (*Required)	Description	Default Value
*serviceUrl	Web service URL of Service Manager. The format is as follows: `http://<Service_Manager_Host>:<Port>/sc62server/PWS/` where *<Service_Manager_Host>* represents the host machine where Service Manager is running.	(None)

Configuring the Service Manager Adapter Converter Property (Script)

The converter section of the Service Manager adapter configuration file contains the `scripts` property. The script file is written in the JavaScript language, and it maps the fields from the Service Manager data model to the PPM data model and filters the requests.

The `scripts` property is a script file name in the following format:

`scripts=<convert1>.js`

This file must reside in the same directory as the adapter, as follows:

<PPM_Home>`/conf/sdi/`*<adapter name>*`.ext`

where *<adapter name>* is as defined in Table 4-1.

> **Note:** Make sure that no line in a script exceeds 256 characters.

> **Note:** Multiple scripts are supported, using a comma-separated list, in the following format:
>
> `scripts=<convert1>.js,<convert2>.js,...`
>
> The adapter searches for these conversion script files in the adapter directory.

The conversion script is responsible for field mapping during the conversion of changes in the Service Manager data model to requests in the PPM data model, and for filtering the changes and requests.

The script must contain the `convert` function and can contain the optional `preFilter` and `postFilter` functions, as follows:

- **preFilter.**

  The following function filters the changes before they are converted to the PPM data model, so that no unnecessary requests are converted:

  ```
  preFilter(smChange)
  ```

  For example, the following `preFilter` function specifies that Service Manager changes with a Low priority will not be converted and that all other requests will be converted:

  ```
  function preFilter(smChange){
      if (smChange.get("Request Urgency")==SM_PRIORITY_LOW)
         return false;
      else
         return true;
     }
  ```

- **convert.**

  After identifying the PPM request attributes that are required for Service Manager changes, use the `convert` function of the conversion script to map fields of Service Manager changes to fields of PPM requests.

  The following convert function uses the mapping you specify to convert the fields of the change in Service Manager to the fields of the request in PPM:

  ```
  convert(smChange, ppmRFC)
  ```

- **postFilter.**

  The following function filters the converted requests, so that only the desired requests will be imported into PPM:

  ```
  postFilter(ppmRFC)
  ```

  For example, the following `postFilter` function specifies that only PPM requests with a status of Approved will be sent to the PPM Server:

  ```
  function postFilter(ppmRFC){
      ppmStatus=ppmRFC.getField("status");
      if (ppmStatus==STATUS_APPROVED)
         return true;
      else
         return false;
  }
  ```

ALM provides a sample conversion script file named `ConvertSMToPPM.js.sample` in the *PPM_Home*`/conf/sdi/serviceManager-adapter.ext` directory.

Copy the sample file, delete the `.sample` extension in the copy, and revise the copied conversion script as needed. Use the syntax described in the following sections for the conversion script APIs.

smChange Object

The smChange object represents the Service Manager change. For the `preFilter` and `convert` script functions, use one of the following functions to retrieve fields from the Service Manager change:

- `value=SMRFC.get(String fieldName);`

- `SMRFC.get("Request Urgency")=SM_PRIORITY_LOW;`

ppmRFC Object

The ppmRFC object represents the PPM request. For the `convert` and `postFilter` script functions, use the following functions to modify the PPM request fields:

- Reference ID

  You must use the following function to track the Service Manager change ID in the PPM request:

  `setRefId(String referenceId);`

- Time Stamp

  You must use the following function to set the last update time in the PPM request:

  ```
  /**
  * Set the time stamp in long format—that is, the number of
  * milliseconds since January 1, 1970, 00:00:00 GMT
  **/
  setUpdatedTimeStamp(long updatedTimeStamp);
  /**
  * Set the time stamp in the Java simple date format, which is
  * described at the following location:
  * http://java.sun.com/j2se/1.4.2/docs/api/java/text/SimpleDateFormat.html
  **/
  setUpdatedTimeStamp(String updatedTimeStamp, String format);
  ```

- General Field

  Use the following function to set the value of a general field in the PPM request:

  `setField (String fieldName, String value);`

- Date

Use the following function to set the value of a date field in the PPM request:

```
/**
* Set the date in long format—that is, the number of
* milliseconds since January 1, 1970, 00:00:00 GMT
**/
setDateValue(String fieldName, long date);
/**
* Set the date in the Java simple date format which is
* described in the following location:
* http://java.sun.com/j2se/1.4.2/docs/api/java/text/SimpleDateFormat.html
**/
setDateValue(String fieldName, String date, String format);
```

- Notes to be added upon creation of a PPM request

  Use the following function to add a note upon creation of a PPM request:

  ```
  addUserNoteOnCreate(String content, String addedBy, long time;
  ```

- Notes to be added upon update of a PPM request

  Use the following function to add a note upon update of a PPM request:

  ```
  addUserNoteOnUpdate(String content, String addedBy, long time;
  ```

- URL reference creation

  If you have configured Service Manager to expose the ticket URL as the `record.url` attribute (see "Configuring Browsing of Service Manager Changes from a URL" on page 286), you can use the following function to create a URL reference to an Service Manager change:

  ```
  addURLReference(String attachmentURL, String attachments);
  ```

Configuring the Service Manager Adapter Sender Properties

Specify the properties for the sender section of the Service Manager adapter configuration file as described in "Configuring the Service Manager Adapter Sender Properties" above. The sender controls creating RFCs in PPM with the converted data.

Table 9-3. Service Manager adapter sender properties

Property Name (*Required)	Description	Default Value
*userName	User name in PPM by whose credentials requests are	(None)

Table 9-3. Service Manager adapter sender properties, continued

Property Name (*Required)	Description	Default Value
	created. This user name must include only single-byte characters.	
*password	Password of the userName. You must encrypt this password by using the `kEncrypt.sh` script, which is located in the `bin` directory of the PPM Server. Encrypted passwords must be created in a `CDATA` section.	(None)
*requestType	PPM request type that should be created for the converted changes. For example: ALM - Request for Change (RFC)	(None)
updateRequest	If set to `true,` enables *updates* made to Service Manager changes to be automatically sent to existing PPM requests. HPE recommends retaining the default value of `false` because usually, after Service Manager changes are converted to PPM requests, processing takes place entirely in PPM.	false
*ticketIdFieldName [a]	Field in PPM containing the Service Manager ticket ID. This field is presented in the PPM request as the **Ticket Id** field in the **Service Desk Info** section.	(None)
*sdSystemFieldName [a]	Field in PPM containing the Service Manager system name. This field is presented in the PPM request as the **System Name** field in the **Service Desk Info** section.	(None)
*staticFieldNames (Applicable and required only for bidirectional integration)	List of PPM request fields (separated by semicolons) that are *not* to be updated when changes are made to their mapped Service Manager change fields. This list is used to prevent inappropriate update of Service Manager tickets for bidirectional integration. For example, to prevent a ticket from being updated when the last update time in PPM changes, specify the following: REQD-SD_LAST_UPDATE	(None)
a. If PPM is integrated with multiple Service Manager servers, the combination of values in the		

Table 9-3. Service Manager adapter sender properties, continued

Property Name (*Required)	Description	Default Value
	ticketIdFieldName and sdSystemFieldName properties ensures that all the tickets from all the Service Manager servers are uniquely identified in PPM.	

The sender is the last section of the adapter configuration file. Make sure the file ends with the following lines:

```
</adapter>
</settings>
```

This completes the configuration of the Service Manager adapter configuration file.

Note: If PPM is operating in a clustered server configuration, share or copy the *<PPM_Home>*/sdi-persistency directory and the *<PPM_Home>*/conf/sdi directory among all the servers in the cluster.

If you want to establish bidirectional integration, proceed to "Configuring the PPM Adapter Configuration File" below. Otherwise, go to "Configuring the server.conf Parameter in PPM" on page 322.

Configuring the PPM Adapter Configuration File

Note: This procedure is optional. It establishes bidirectional integration. (See "Introduction to Integrating PPM Requests with Service Manager Changes, Using ALM" on page 274.) If you do not want to establish bidirectional integration at this time, proceed to "Configuring the server.conf Parameter in PPM" on page 322.

The PPM adapter configuration file is an XML file in PPM that enables integration of PPM with Service Manager and then converts PPM RFCs to Service Manager change updates. The configuration file consists of the following components, each with its own attributes or properties (see Figure 4-2):

- General settings for the adapter itself, such as its name.

- Connector between PPM and the adapter.

- Converter of RFCs in the PPM data model to change updates in the Service Manager data model. The converter calls the scripts that define the field mapping and filter functions.

- Sender, which sends the converted and filtered requests to Service Manager.

> **Note:** If PPM is operating in a clustered server configuration, share or copy the `<PPM_Home>/sdi-persistency` directory and the `<PPM_Home>/conf/sdi` directory among all the servers in the cluster.

The following sections describe how to configure the PPM adapter configuration file and the scripts called by its converter, and how to modify copies of the provided files while preserving the originals.

Location, Naming, and Structure of PPM Adapter Configuration Files

Each PPM adapter configuration file must follow specific conventions for its location, naming, and structure, as described in the following sections.

Location and Naming of the PPM Adapter Configuration File

The adapter configuration files are located in the `<PPM_Home>/conf/sdi` directory of the PPM Server. This directory contains the following:

- Configuration file for the adapter (or one for each adapter if there are multiple adapters).

  In this configuration file, you define a name for the adapter. The configuration file must have a `.settings` file extension.

- Subdirectory (or one for each adapter configuration file if there are multiple files).

  The subdirectory contains the conversion scripts, which are called by the converter to map the fields between and to filter the tickets and requests. The name of the subdirectory must be the same as the *<adapter name>* (for a description, see "Structure of the PPM Adapter Configuration File" below and Table 4-4) followed by `.ext`.

  For example, if the adapter name is `ppm-adapter,` the `<PPM_Home>/conf/sdi` directory must contain a subdirectory named `ppm-adapter.ext` to hold all the conversion script files for the adapter.

Structure of the PPM Adapter Configuration File

ALM provides, as a template, a default PPM adapter configuration file named `ppm-sm-adapter.settings1.` The adapter file you configure and use must have a `.settings` file extension.

Copy the default adapter file (to preserve the original), and rename the copy with a `.settings` file extension and, if desired, a different file name.

As detailed in subsequent sections, the adapter file has the following basic structure, including adapter attributes, and properties for its connector, converter, and sender:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<settings>

<adapter adapter-name="<adapter name>">
    <service-desk-application><SD application>
        </service-desk-application>
    <number-of-tickets><number of tickets></number-of-tickets>
    <polling-schedules><schedule></polling-schedules>
    <polling-frequency><frequency></polling-frequency>

    <request-types>
    <request-type level="1">
    <polling-operation>

    <connector>
    <connector-type>PPMRequest</connector-type>
        <properties>
            sdSystemName=
            requestType=
            datePattern=
            userName=
            password=
            sdSystemFieldName=
            idProperty=
            updateTimeField=
            createTimeField=
            requestStatusNames=
        </properties>
    </connector>

    <converter>
    <converter-type>scriptConverter</converter-type>
        <properties>
            scripts=<convert1>.js,<convert2>.js,...
        </properties>
    </converter>

    </polling-operation>
    </request-type>
    </request-types>

    <sender>
    <sender-type>serviceManagerSender</sender-type>
        <properties>
            userName=
            password=
```

```
                  queryDateFormatPattern=
                  timeZone=
                  keyMethodName=
                  serviceUrl=
                  staticFieldNames=
                  idProperty=
            </properties>
        </sender>

    </adapter>

    </settings>
```

> **Caution:** Do not delete or change the values provided for `<connector-type>`, `<converter-type>`, or `<sender-type>`.

The following sections describe how to configure the adapter attributes, the connector properties, the converter property (`scripts`), and the sender properties.

Configuring the PPM Adapter Attributes

Specify the adapter attributes of the PPM adapter configuration file, such as the adapter name and the service desk application, as described in the table below.

Table 4-4. PPM Center adapter attributes

Attribute Name (*Required)	Description	Default Value
*adapter-name	Logical name that represents the adapter name on the client machine. For example: ppm-adapter This name is also used for the scripts (`.ext`) directory. (See "Location and Naming of the PPM Adapter Configuration File" on page 314.)	(None)
*service-desk-application	Unique, logical name for the PPM system you are using. For example: PPM	(None)
number-of-tickets	Number of tickets that the adapter processes at a time.	50
polling-schedules	Times of day that the adapter polls PPM for	(None)

Table 4-4. PPM Center adapter attributes, continued

Attribute Name (*Required)	Description	Default Value
	changes, formatted as a list of cron expressions separated by the new line character. For example: 30 * * * * *<new line> 0 * * * *	
polling-frequency	Frequency (in seconds) that the adapter polls PPM for changes.	If polling-schedules and polling-frequency are unspecified, then the default polling-frequency is 30 seconds, starting when you restart the PPM Server.

Configuring the PPM Adapter Connector Properties

Specify the properties for the connector section of the PPM adapter configuration file as described in the table below.

Table 4-5. PPM Center adapter connector properties

Property Name (*Required)	Description	Default Value
*sdSystemName	Name of the adapter from which changes are imported into PPM as requests. Must be the same value as specified for the service-desk-application property in the Service Manager adapter (see Table 4-1). For example: Service Manager	(None)
*requestType	PPM request type that should be created for the converted changes. For example: ALM - Request for Change (RFC)	(None)
datePattern	Date format for the date field. Use the Java™ simple date format. See the following URL: http://java.sun.com/j2se/1.4.2/docs/api/java/text/SimpleDateFormat.html	yyyy-MM-dd HH:mm:ss

Table 4-5. PPM Center adapter connector properties, continued

Property Name (*Required)	Description	Default Value
*userName	User name in PPM by whose credentials requests are created. This user name must include only single-byte characters.	(None)
*password	Password of the userName. This password should be encrypted using the PPM script `kEncrypt.sh,` which is located in the `bin` directory of the PPM Server. Encrypted passwords must be created in a `CDATA` section.	(None)
*sdSystemFieldName	Field in PPM containing the Service Manager system name. This field is presented in the PPM request as the **System Name** field in the **Service Desk Info** section.	(None)
*idProperty	Property name of the ID field in the instance returned from the Service Manager Web service.	(None)
*updateTimeField	Field in PPM that represents the time the request was updated in PPM.	(None)
*createTimeField	Field in PPM that represents the time the request was created in PPM.	(None)
requestStatusNames	List of PPM request status, separated by semicolons (;). Only requests with the status you specify are retrieved from PPM for processing. However, if you do not specify any status, all requests are retrieved.	(None)

Configuring the PPM Adapter Converter Property (Script)

The converter section of the PPM adapter configuration file contains the `scripts` property. The script file is written in the JavaScript language. The script maps the fields from the PPM data model to the Service Manager data model, and filters the requests.

The `scripts` property is a script file name in the following format:

`scripts=<convert1>.js`

This file must reside in the same directory as the adapter, as follows:

`<PPM_Home>/conf/sdi/<adapter name>.ext`

where *<adapter name>* is as defined in .

> **Note:** Make sure that no line in a script exceeds 256 characters.

> **Note:** Multiple scripts are supported, using a comma-separated list, in the following format:
>
> `scripts=<convert1>.js,<convert2>.js,...`
>
> The adapter searches for these conversion script files in the adapter directory.

The conversion script is responsible for field mapping during the conversion of requests in the PPM data model to change updates in the Service Manager data model, and for filtering the requests and change updates.

The script must contain the `convert` function and can contain the `preFilter` and `postFilter` functions, as follows:

- **preFilter.**

  The following function filters the changes before they are converted to the Service Manager data model, so that no unnecessary requests are converted:

  `preFilter(ppmRFC)`

- **convert.**

  After identifying the Service Manager change attributes that need to be updated from the PPM requests, use the `convert` function of the conversion script to map fields of PPM requests to fields of Service Manager changes.

  The following convert function uses the mapping you specify to convert the fields of the request in PPM to the fields of the change in Service Manager:

  `convert(ppmRFC, smChange)`

- **postFilter.**

  The following function filters the converted change updates, so that only the desired updates will be applied to the corresponding change in Service Manager:

  `postFilter(smChange)`

  ALM provides a sample conversion script file named `ConvertPPMToSM.js.sample` in the `<PPM_Home>`/conf/sdi/serviceManager-adapter.ext directory.

  Copy the sample file, delete the `.sample` extension in the copy, and revise the copied conversion script as needed. Use the syntax described in the following sections for the conversion script APIs.

ppmRFC Object

The ppmRFC object represents the PPM request. For the `preFilter` and `convert` script functions, use the following function to retrieve request fields from PPM:

```
get(String fieldName);
```

smChange Object

The smChange object represents the Service Manager ticket. For the `convert` and `postFilter` script functions, use the following functions to modify the ticket fields:

- Reference ID

  You must use the following function to track the Service Manager change ID in the PPM request:

  ```
  setRefId(String referenceId);
  ```

- Time Stamp

  You must use the following function to set the last update time in the PPM request:

  ```
  /**
  * Set the time stamp in long format—that is, the number of
  * milliseconds since January 1, 1970, 00:00:00 GMT
  **/
  setUpdatedTimeStamp(long updatedTimeStamp);
  /**
  * Set the time stamp in the Java simple date format, which is * described at the
  following location:
  * http://java.sun.com/j2se/1.4.2/docs/api/java/text/SimpleDateFormat.html
  **/
  setUpdatedTimeStamp(String updatedTimeStamp, String format);
  ```

- General Field

  Use the following function to set a value of a general field in the PPM request:

  ```
  setField (String fieldName, String value);
  ```

- Date

  Use the following function to set a value of a date field in the PPM request:

  ```
  /**
  * Set the date in long format—that is, the number of
  * milliseconds since January 1, 1970, 00:00:00 GMT
  ```

```
**/
setDateValue(String fieldName, long date);
/**
* Set the date in the Java simple date format which is
* described in the following location:
* http://java.sun.com/j2se/1.4.2/docs/api/java/text/SimpleDateFormat.html
**/
setDateValue(String fieldName, String date, String format);
```

Configuring the PPM Adapter Sender Properties

Specify the properties for the sender section of the PPM adapter configuration file as described in the table below. The sender controls updating changes in Service Manager with the converted data.

Table 4-6. PPM Center adapter sender properties

Property Name (*Required)	Description	Default Value
*userName	User name in the Service Manager system that PPM uses to connect to Service Manager. This user name must include only single-byte characters. This user must have full access to the Change Management module in Service Manager.	(None)
*password	Password in the Service Manager system that PPM uses to connect to Service Manager.	(None)
*queryDateFormatPattern	Date format used for querying the Service Manager system (as used in the UI expert search). Make sure this value is consistent with the Service Manager server date pattern. For available formats, see the following URL: http://java.sun.com/j2se/1.4.2/docs/api/java/text/SimpleDateFormat.html	(None)
*timeZone	Time zone, used for converting the last updated time of a request from Service Manager. Use the same time zone as the Service Manager server. The format can be GMT+<X> or GMT–<X>, where <X> is the offset in hh:mm format from GMT. For example, GMT–07:00. However, to handle Daylight Saving Time, use an area time zone instead of specifying a time relative to GMT.	(None)

Table 4-6. PPM Center adapter sender properties, continued

Property Name (*Required)	Description	Default Value
*keyMethodName	Name of the method for request keys (usually the ID field name).	(None)
*serviceUrl	Web service URL of Service Manager. The format is as follows: http://<*Service_Manager_Host*>:<*Port*>/sc62server/PWS/ where <*Service_Manager_Host*> represents the host machine where Service Manager is running.	(None)
*staticFieldNames (Applicable and required only for bidirectional integration)	List of Service Manager change fields (separated by semicolons) that are *not* to be updated when changes are made to their mapped PPM request fields. This list is used to prevent inappropriate update of PPM requests for bidirectional integration. For example, to prevent a request from being updated when the last update time in Service Manager changes, specify the following: sysmodtime	(None)
idProperty	Property name of the ID field in the instance returned from the Service Manager Web service.	(None)

The sender is the last section of the adapter configuration file. Make sure the file ends with the following:

```
</adapter>
</settings>
```

This completes the configuration of the PPM adapter file.

> **Note:** If PPM is operating in a clustered server configuration, share or copy the <*PPM_Home*>/sdi-persistency directory and the <*PPM_Home*>/conf/sdi directory among all the servers in the cluster.

Configuring the server.conf Parameter in PPM

> **Note:** For more information about the steps in this procedure, see the *Installation and Administration Guide.*

Add (if not present) and specify the parameter related to Service Manager integration to the PPM `server.conf` configuration file, as follows:

1. Stop the PPM Server.

2. Run the following script:

   `sh ./kConfig.sh`

   Set the parameter and value as shown in the following table. (All parameter names begin with `com.kintana.core.server.` but that is not shown in the table.)

Parameter	Value
MAC_LOG_SEVERITY	Specify the level of logging to be used. When set to `0` (the default), only integration exceptions (errors) and a summary are logged. When set to `1,` non-error events related to the processing of changes are also logged. See "Error and Non-Error Logging" on the next page.

3. Restart the PPM Server.

Enabling the ALM Startup Service

To enable the ALM Startup service:

1. Log on to PPM.

2. From the menu bar, select **Open > Administration > Schedule Services.**

   The Schedule Services page opens.

3. In the **Service Name** column, select the **ALM Startup** service.

4. In the **Status** column for the service, select **Enabled.**

5. Specify one hour for the interval in the **Schedule** column. This interval has no effect on the polling of Service Manager for tickets.

6. Click **Save.**

7. If a date was specified in the `initial-load-state` Service Manager adapter attribute in order to retrieve existing Service Manager changes from that date forward, those changes will be retrieved, converted, and sent to PPM, but then no new Service Manager changes will be retrieved. In this case, to retrieve Service Manager changes on an ongoing basis, do the following:

> **Note:** For more information about the steps in this procedure, see the *Installation and Administration Guide*.

    a. Stop the PPM Server.

    b. Comment out the `initial-load-state` Service Manager adapter attribute.

    c. Restart the PPM Server in normal mode.

    d. Repeat step 1 through step 6 to verify that the ALM Startup service is running.

Error and Non-Error Logging

As described in the following sections, logging is essential for ensuring correct operation of the integration. You can specify the following two types of logging for Service Manager integration:

- System-level logging using a summary logs table and a log details table. See "System-Level Logging" below.

- Configurable logging for conversion scripts. See "Configurable Logging for Conversion Scripts" on page 326.

System-Level Logging

System-level logging is the only way to administer the integration on an ongoing basis. HPE recommends creating a portlet or a report to regularly query and display the logging tables and help you identify potential problems.

You can configure ALM to log only errors for integration exceptions and a summary, or to also log entries for non-error events related to the processing of changes. For information about setting the `server.conf` parameter that controls the level of logging (MAC_LOG_SEVERITY), see "Configuring the server.conf Parameter in PPM" on page 322.

Setting the MAC_LOG_SEVERITY parameter to `0` causes the summary logs table (`SDI_SUMMARY_LOGS`) to include only a summary of errors. Setting this parameter to `1` causes the summary logs table to also include information about tickets as they are being processed.

The summary logs table maintains only one row for each unique combination of the **TICKET_ID** and **ADAPTER_NAME** columns. See Summary Logs Table. Each row is updated on an ongoing basis as status changes.

The log details table (SDI_LOG_DETAILS) includes details about the errors that have a value of **Y** in the **MESSAGES** column of the summary logs table. See Log Details Table. (The setting of the MAC_ LOG_SEVERITY parameter in the server.conf file does not affect the contents of the log details table.)

Summary Logs Table

The columns in the summary logs table (SDI_SUMMARY_LOGS) that contain useful information are described in the table below.

Table 4-7. Summary logs table (SDI_SUMMARY_LOGS)

Column	Description
LOG_ID	Primary key for this table.
TICKET_ID	Ticket ID imported using the connector. In some cases such as connection errors or authentication failures, the value is **SUMMARY.**
STATUS	Current state of the ticket. Possible values are as follows: • **Retrieved/Not processed.** The ticket was retrieved and has not been processed. • **preFilter Passed.** The ticket passed the preFilter function and was sent to the convert function. • **Rejected in preFilter.** The ticket did not pass preFilter criteria. • **Ticket converted.** The ticket passed the convert function and was sent to the postFilter function. • **postFilter Passed.** The ticket passed the postFilter function and was sent to the sender function. • **Rejected in postFilter.** The ticket did not pass the postFilter function. • **Error occurred in JavaScript.** An exception occurred in the preFilter, convert, or postFilter function when processing the JavaScript conversion file. • **Ticket processed.** The ticket was created in the target system. • **Errors occurred when sending the ticket.** An exception occurred and the ticket could not be sent to the sender.
LAST_UPDATE_DATE	Last time the ticket was updated in PPM.
COMPONENT	Component that logged the message—the connector, converter, or sender.
ADAPTER_NAME	Service desk name in the adapter settings file.
MESSAGES	If the value is **Y,** further details for this log entry are provided in the log

Table 4-7. Summary logs table (SDI_SUMMARY_LOGS), continued

Column	Description
	details table (SDI_LOG_DETAILS). If the value is **N,** no further details are provided.
IN_PROCESS	If the value is **Y,** processing of this ticket is complete. If the value is **N,** processing of this ticket is not yet complete.

Log Details Table

The log details table (SDI_LOG_DETAILS) includes the details described in the following table for the errors that have a value of **Y** in the **MESSAGES** column of the summary logs table.

Table 4-8. Log details table (SDI_LOG_DETAILS)

Column	Description
LOG_DETAIL_ID	Primary key for this table
LOG_ID	Foreign key to the entry in the SDI_SUMMARY_LOGS table
DETAILS	Detailed error message captured when the error occurred

Configurable Logging for Conversion Scripts

Configurable logging is useful for debugging your integration mapping. During testing, you can specify log messages that appear at key points in the conversion script to indicate correct or incorrect conversion. Before deploying the integration to production, you would typically want to comment out the messages for correct operation so they do not quickly accumulate in the logs.

If you want to view log messages describing the activity that occurs during the request conversion process, you can include logging objects in your conversion scripts. During the conversion process, you can view the log messages in the conversion script log files, located in the *<PPM_Home>*/script-logs directory. Each adapter logs messages in a separate log file. The names of the log files are based on the names of the adapters for which the files log errors.

Within any of the script functions, a logging object with the following syntax should be included:

```
logger.<type of message>("<log message>");
```

The following message types can be used:

- info. Records all processing activity that is performed.

- warn. Records warning messages.

- error. Records error messages.

> **Note:** Logging is also controlled by the severity specified in the PPM`logging.conf` file.

For example, you can include a logging object such as the following:

```
logger.info("Processing ticket " + serviceManagerRFC.get("header.changeNumber"));
```

If you want the conversion script log files to display a list of all Service Manager fields, you can include the following logging object in your conversion script:

```
logger.info(BeanUtils.describe(ticket));
```

If you use this logging object, make sure the following line is included at the beginning of the conversion script:

```
importPackage(Packages.org.apache.commons.beanutils);
```

Troubleshooting the Integration

This section contains the following topics:

-

-

-

Troubleshooting Tips

Resources on PPM Side

- File System

  - PPM / Service Manager directories

    - `c:\ppm\conf\sdi` — Configuration / adapter files and folder

    - `c:\ppm\bin\sdi` — Web stub compilation scripts

  - `c:\ppm\sdi-persistency`

    i. When the connection is successfully established, files will be generated in this directory.

      If not, there should be an issue with starting the ALM services, permission issues with configuration files or connection related problem.

    ii. `*.ser` — The last Service Manager ticket updated

    iii. `*.log` — The integration log

- DB tables

    a. `sdi_log_details`

    b. `sdi_summary_logs`

Failure to Compile the Web Services Stub

1. Make sure that the host names can be resolved.

2. Make sure the Service Manager has enabled the legacy URL:

   `http://<SMC_WEB_SERVICES_URL>:<PORT>/sc62server/PWS/ChangeManagement.wsdl`

   Otherwise you will get timeout error.

3. Make sure PPM can access Service Manager Web services port and URL through the firewall, Load Balancer, Proxy, and so on.

Error Types

Unable to read serviceManager-adapter.settings file

Check the folder/file ownership to the `*.settings` file. Must be owned by the process owner, for example, `infra\itg##-user`.

Failure to synchronize data

1. timeZone mismatch

    a. Make sure the `timeZone` values match between the `*.settings` file in PPM, the `server.conf` file on PPM, Service Manager application server timezone setting and the Service Manager `SM_PPM_INT` user's profile.

- Prefer GMT-08:00

- Will take US/Pacific and so on.
  Run Server Admin Timezone to get a list of valid values.

b. `serverLog.txt` message:
   ```
   ":Scheduler_Worker-
   5:com.mercury.onyx.sdi.client.SingleLevelAdapter:2010/06/17-14:18:39.787
   EDT: SingleLevelAdapter name=serviceManager-adapter_1 - collected 0 tickets"
   ```

   Check the time zone setting of the Service Manager integration user `SM_PPM_INT`:

   i. Login to Service Manager with the integration user ID.

   ii. Type `operator` in the Service Manager command line.

   iii. Login Name, Service Manager user name defined by Service Manager administrator. For example, `SM_PPM_INT`.

   iv. Click the **Login Profiles** tab.

   v. Validate Time Zone setting.

2. Make sure the corresponding Service Manager fields are exposed.

   Use the following URL and validate the Service Manager field are listed here.

   ```
   [http://<SMC_WEB_SERVICES_
   URL>:<PORT>/sc62server/PWSChangeManagement.wsdl|http://<SMC_WEB_SERVICES_
   URL>:<PORT>/sc62server/PWSChangeManagement.wsdl]
   ```

3. Make sure the Service Manager fields has a corresponding DB column/table. For example, check the DB schema for the field.

4. Make sure the corresponding fields exist in the Service Manager. For example, structured fields.

Error in Service Manager server log " SELECT m1."NUMBER" FROM CM3RM1 m1 WHERE ((m1."SYSMODTIME" and m1."SYSMODTIME")) ORDER BY m1."NUMBER" ASC "

To troubleshoot the error,

1. Check the Service Manager user account used to perform the integration.

2. Open this operator in Service Manager and change the date format to be the same with the `queryDateFormatPattern`' in the `*.settings` configuration file.

   For example, `queryDateFormatPattern=MM/dd/yyyy HH:mm:ss` in `serviceManager-txp-`

```
adapter.settings.
```

3. After making this change, regenerate the Web Service stubs.

Error in the sdi_log_details table

- `"PPMSender:Failed to create/update a request in PPM for ticket with refId= "`

    ○ Make sure the PPM user ID used for integration is created on PPM side and the password matches.

    ○ Try logging into PPM using the `SM_PPM_INT` user ID and make sure that the Reset Password window pop up is not present. If so, reset to something else and then reset back to "your password" again.

- `"com.mercury.onyx.sdi.SDIException:`
  `java.lang.IllegalArgumentException: Property 'reqModDate'`
  `has no read method"`

    Validate the **reqModDate** field is exposed and has the correct Caption:

- `"org.xml.sax.SAXParseException: The markup in the document`
  `following the root element must be well-formed."`

    Validate there are no spaces in the exposed WSDL caption names.

    For example, "Record URL" is incorrect, "RecordURL" is correct.

Authentication Issues with the sdi_log_details table and serverLog.txt

1. Check to make sure the MAM field group is disabled if not needed. Will see failed authentication error in PPM`serverLog.txt`.

2. Validate the Service Manager user credentials and password matches between the `c:\ppm\conf\sdi\serviceManager.settings` and the `ppm-sm-adapater.settings` files.

Check Service Manager Global Time Zone Setting

1. Check Time Zone and Format settings in Service Manager application.

    a. Log on to the Service Manager thick client.

b. Select **Menu Navigationb > System Administration > Base System Configuration > Miscellaneous > System Information Record**.

c. Click the **Date Info** tab.

d. Check the values for the Time Zone and Format fields.

2. In the DB, look at the corresponding time zone file to see the offset and switchover date.

a. Type `db` in the Service Manager thick client command line and press **Enter**.

The Database Manager screen opens.

b. Select **tzfile** from the Table drop-down list and click **Search**.

c. Double click the correct time zone file. For example, **US/Pacific**.

d. Check the offset and switchover date.

Additional logging.conf Parameters for the ALM Module

Additional logging can be found in the `serverLog.txt` file.

Make sure you change the `SYSTEM_THRESHOLD` setting from `ERROR` to `DEBUG` for the following to take effect:

```
com.kintana.core.logging.PRODUCT_FUNCTION_LOGGING_LEVEL = com.kintana.services,
DEBUG

com.kintana.core.logging.PRODUCT_FUNCTION_LOGGING_LEVEL = com.mercury.onyx, DEBUG

com.kintana.core.logging.PRODUCT_FUNCTION_LOGGING_LEVEL = com.kintana.crt, DEBUG

com.kintana.core.logging.PRODUCT_FUNCTION_LOGGING_LEVEL =
com.mercury.onyx.sdi.client.AdapterHelper, DEBUG
```

Chapter 12: Integrating PPM Tasks with HPE Service Manager RFCs

For an overview of the integration of PPM tasks with Service Manager RFCs, see "Integration of PPM Project Tasks with Service Manager RFCs" on page 23.

> **Note:** This integration does not use the ALM entities and does not require installing the ALM software.

This integration allows PPM project managers to specify which tasks in a project, if any, automatically create corresponding requests for change (RFCs) in Service Manager. As the RFCs are completed in Service Manager, the statuses of the associated PPM tasks are automatically set to Complete (or Cancelled).

> **Caution:** This integration is not available for all projects where MSP Integration is set to **Microsoft controls all shared work plan information**.

The typical flow of this integration is as follows:

- A project manager creates a task in a PPM project, which initiates an operational RFC in Service Manager.

- Key information is copied from the task to the RFC (and is updated when the task is modified).

- A Service Manager user works on the RFC, and the PPM task is automatically updated to reflect the RFC status.

- The project manager can review the status of the Service Manager RFC as reflected in the PPM task.

- Upon completion of the Service Manager RFC, the PPM task status is updated to **Complete** or **Cancelled.**

Configuring the Integration of PPM Tasks and Service Manager RFCs

To configure this integration, perform the following steps in PPM and Service Manager. You must have system administrator privileges in both PPM and Service Manager.

1. Stop the PPM Server.

2. Check the PPM Web services configuration. Make sure the basic authentication mode is enabled.

   a. Open the configuration file located at: *<PPM_Home>*\server*<PPM_Server_Name>*\deploy\itg.war\WEB-INF\conf\axis2.xml, where *<PPM_Server_Name>* is the host name or IP address of your PPM instance.

   b. Make sure the value of InFlowBasicAuth is true.

      Note: If PPM is operating in a cluster configuration, you must update the axis2.xml file for all of the nodes in the cluster.

3. Log on to Service Manager. In the cm3r table in Service Manager, add a new field for the PPM task ID (for example, PPMTaskId).

   The field name you specify here will be used for field mapping in

      Caution: After adding the new field, for example, PPMTaskId, make sure you use the same case-sensitive field name PPMTaskId later when necessary.

   a. In the Service Manager command line, type dbdict and press **Enter**.

      The Database Dictionary screen opens.

   b. In the **File Name** field, enter cm3r and click **Search**.

   c. Select **cm3r** in the File Name section.

   d. Click on the Name column of the **descriptor** row.

   e. Click **New Field/Key**.

      The field.window pops up.

   f. Enter taskId in the **Name** field.

   g. Select **number** from the **Type** drop-down list and click the **Add Field** icon.

      The field is added to the dbdict.

   h. Locate the **TaskId** field and double-click it.

   i. In the field.window that pops up, enter the following:

Table Field	Value
SQL Name	TASKID

Table Field	Value
SQL Type	FLOAT **Note:** You may also choose `VARCHAR2(400)`.
SQL Table	m1

j. Click **OK**.

k. Click **SM Alters** after confirmation of SQL Statements.

4. In PPM, check the field mapping file, which defines the field mapping from PPM to Service
    Manager.

    a. In the `<PPM_Home>`\conf\smrfc directory, copy the field mapping file `sm-rfc-
        mapping.xml.sample` under the same directory and rename the copied file to `sm-rfc-
        mapping.xml`.

    b. In the new mapping file, map the `TASK_ID` field to the Service Manager field that you created in
        step 3. The fields in the mapping file are case-sensitive.

        The sample `sm-rfc-mapping.xml` file in PPM Center 9.1x in fact has `smField->PPMTaskID`
        that maps to `ppmField->TASK_ID`. Such as:

        ```
        <field>
        <smField>PPMTaskId</smField>
        <ppmField>TASK_ID</ppmField>
        <useOnCreate>true</useOnCreate>
        <useOnUpdate>true</useOnUpdate>
        </field>
        ```

    c. In the new mapping file, verify that the Status field in Service Manager is set to `initial`. This
        value will be maintained by Service Manager after the change record is created in Service
        Manager.

        ```
        <field>
                <smField>Status</smField>
                <useOnCreate>true</useOnCreate>
                <useOnUpdate>false</useOnUpdate>
                <defaultValue>initial</defaultValue>
         </field>
        ```

    d. In the new mapping file, configure the default value for the Service field by changing the
        sample value "Applications" to your desired value.

        ```
        <field>
                <smField>Service</smField>
        ```

```
            <useOnCreate>true</useOnCreate>
            <defaultValue>Applications</defaultValue>
      </field>
```

The following is an example of the `sm-rfc-mapping.xml` file:

```xml
<?xml version="1.0" ?>
<PPMSMIntegration>
<SMRFCMapping>
    <field>
        <smField>PPMTaskId</smField>
        <ppmField>TASK_ID</ppmField>
        <useOnCreate>true</useOnCreate>
        <useOnUpdate>true</useOnUpdate>
    </field>
    <field>
        <smField>BriefDescription</smField>
        <ppmField>TASK_NAME</ppmField>
        <useOnCreate>true</useOnCreate>
        <useOnUpdate>true</useOnUpdate>
    </field>
    <field>
        <smField>Description</smField>
        <ppmField>TASK_DESCRIPTION</ppmField>
        <useOnCreate>true</useOnCreate>
        <useOnUpdate>true</useOnUpdate>
        <defaultValue>ppm task does not have a description</defaultValue>
    </field>
    <field>
        <smField>RequestedDate</smField>
        <ppmField>TASK_SCHEDULED_END_DATE</ppmField>
        <useOnCreate>true</useOnCreate>
        <useOnUpdate>true</useOnUpdate>
    </field>
    <field>
        <smField>RequestedBy</smField>
        <useOnCreate>true</useOnCreate>
        <defaultValue>FALCON, JENNIFER</defaultValue>
    </field>
    <field>
        <smField>Status</smField>
        <useOnCreate>true</useOnCreate>
        <useOnUpdate>false</useOnUpdate>
        <defaultValue>initial</defaultValue>
    </field>
    <field>
        <smField>Service</smField>
        <useOnCreate>true</useOnCreate>
```

```
            <defaultValue>Applications</defaultValue>
        </field>
    </SMRFCMapping>
</PPMSMIntegration>
```

Each element in the mapping file is described in the following table:

Element	Description
field	Each field element represents a field mapping between PPM and Service Manager.
smField	Caption name of the Service Manager field. The Service Manager field must be exposed through the ChangeIIA object in the ChangeManagement Web service in Service Manager. For more information, see step 5.
ppmField	Field name of the PPM task. For the tokens available for this element, see the notes following this table.
useOnCreate	Specifies whether or not the field value is sent to Service Manager when an RFC is created.
useOnUpdate	Specifies whether or not the field value is sent to Service Manager when an RFC is updated.
defaultValue	Provides a default value for the Service Manager field. **Note:** If `ppmField` is not specified, or the value of `ppmField` is null or empty, then this default value is used.

Note:

○ You can add more mappings as needed. However, make sure that all the Service Manager fields specified in the field mapping file are exposed through the Service Manager ChangeManagement Web service. For more information, see step 5.

○ The tokens in the following table can be used in the `ppmField` element.

Token	Description
TASK_ID	Task ID
TASK_NAME	Name of the task
TASK_DESCRIPTION	Description of the task
TASK_STATUS	Status meaning of the task
TASK_PRIORITY	Priority of the task
TASK_IS_MILESTONE	Whether the task is a milestone

Token	Description
TASK_IS_MAJOR_MILESTONE	Whether the task is a major milestone
TASK_SCHEDULED_END_DATE	Scheduled finish date of the task
TASK_SCHEDULED_START_DATE	Scheduled start date of the task

TASK_PATH	Path of the task in the following format:
	
	`rootTask > firstLevelTask >`
	`secondLevelTask >...`

TASK_USERDATA_01 through TASK_USERDATA_20	User data fields 1 – 20 of the task
PROJECT_ID	Project ID
PROJECT_NAME	Name of the project
PROJECT_REQUEST_ID	PFM request ID of the project
PROJECT_MANAGER_USER_ID	User ID of the project manager
PROJECT_MANAGER_USERNAME	User name of the project manager
PROJECT_MANAGER_FULLNAME	Full name of the project manager
PROJECT_MANAGER_EMAIL	Email address of the project manager
SYS_USER_ID	User ID of the current user
SYS_USERNAME	user name of the current user
SYS_USER_FULLNAME	Full name of the current user
SYS_USER_EMAIL	Email address of the current user

5. Make sure that the Service Manager fields specified in the field mapping file are exposed through the `ChangeIIA` object in the ChangeManagement Web service in Service Manager.

   In Service Manager, check that all the fields are listed on the **Fields** tab of the `ChangeIIA` object (for the `cm3r` table) in the ChangeManagement Web service. If any field is not listed, add the field name and caption name to the **Field** and **Caption** columns.

   For details on how to expose the fields of a table through a Service Manager Web service, see the Service Manager online help.

   To add `TaskId` to the ChangeManagement WSDL,

   a. Log on to Service Manager as System Administrator.

   b. Select **Menu Navigation > Tailoring > Web Services > WSDL Configuration**.

      The External Access Definition page opens.

   c. In the **Service Name** field, type `ChangeManagement` and click **Search**.

d. In the object.name section, select **ChangeIIA**.

e. In the External Access Definition section, click the **Fields** tab.

f. Scroll down to the bottom of the form, and on the next available line, enter the following:

Table Field	Value
Field	`taskId`
Caption	`TaskId` **Note:** This caption will be exposed to WSDL and must match the smField name defined in the `sm-rfc-mapping.xml` file.
Type	`StringType` **Note:** PPM web service only supports `DateTime` type and `String` type.

g. Click **Save**.

h. Click **OK.**

i. Repeat step 3 through step 5 if you have other new fields added into Service Manager and mapped them with PPM Center in the xml file.

However, HPE srecommends you to start with just `TaskId` field first, once the sample xml file works, then you may consider to customize.

j. To validate whether the field is exposed to WSDL file, launch your browser and enter the following:

`http://<SM_WEB_SERVICES_URL>:<PORT>/sc62server/PWS/ChangeManagement.wsdl`

k. In the WSDL content file, search for TaskId, you shall see it under ChangeIIAKeysType, this indicates the field is exposed properly. If you do not see it exposed, then go back to Service Manager and make sure web service is working.

6. Import the following unload files provided with PPM (present in the `<PPM_Home>`\conf\smrfc directory) into Service Manager:

   o `PPMIntegration.unl` (the integration unload file)

   o `PPMIntegration_Schedule.unl` (the `ppmfailover` schedule object file)

   Caution: If PPM is integrated with Service Manager with a PD patch, you shall also import the following unl file into Service Manager:

○ HPSMPPMIntegration.unl

If you already deployed 9.22.0001 patch, the unl file is also present in the *<PPM_Home>*\conf\smrfc directory. Otherwise, go to KM00786444 to download the unl file.

Tip: If you are using an Oracle database with Service Manager, loading of the PPMIntegration.unl file might fail the first time. Normally, reloading the file could solve the problem.

Tip: If you load the PPMIntegration_Schedule.unl file more than once, duplicate ppmfailover schedules will be created in Service Manager. In this case, delete the redundant ppmfailover schedules.

a. Log on to Service Manager as system administrator.

b. Type **db** in the Service Manager command line and press **Enter**.

    The Database Manager window opens.

c. Right click in the Database Manager window and select **Import/Load** from the context menu.

    The HPE Service Manager File Load/Import screen opens.

d. In the File Name field, specify the file to load:

    <PPM_Home>\conf\smrfc\PPMIntegration.unl

e. Click **Load FG**.

    A message stating that the records have been loaded displays.

f. Click **Back**.

g. Repeat the step c through step e for the PPMIntegration_Schedule.unl file.

7. Configure the integration table in Service Manager.

    a. Type **db** in the Service Manager command line and press **Enter**.

        The Database Manager window opens.

    b. Select **ppmIntegration** from the Table drop-down list and click **Search**.

        The form for the ppmIntegration table displays.

    c. Add the following new record to the table:

Table Field	Value
Id	1
Field to Store TaskId	The value of this field must match the field name you previously added to the cm3r table for the PPM task ID (see step 3). For example, **PPMTaskId.** **Note:** If the case-sensitive field names do not match, the integration will fail.
PPM Server URL	URL of the PPM Web services. For example: `http://<Host_Name>:<Port>/itg/ppmservices/`
PPM Server Username	The user name that Service Manager uses to call the PPM Web services. This user name must include only single-byte characters. HPE recommends that you create a separate user account for this purpose.
PPM Server Password	The password of the user name that Service Manager uses to call the PPM Web services.

8. If necessary, modify the Service Manager processes that will call the PPM Web services to update the RFC status and task status.

   Note: The code is added to the **cm.close, cm.reject, cm.update.save,** and **cm.next.phase** processes when you load the `PPMIntegration.unl` file. The standard **Save, Close, Reject, Next Phase,** and **Reopen** actions for RFCs invoke these processes. If you have added any other actions used to update changes (through tailoring of your Service Manager instance), you must also modify the processes that these self-defined actions will invoke.

   Note: For Service Manager with a PD patch deployed, you only need to modify the **cm.close** and **change.update.save** processes.

   To modify the Service Manager processes,

   a. Log on to Service Manager as system administrator.

   b. Select **Menu Navigation > Tailoring > Document Engine > Processes**.

      The Process Definition search screen displays.

   c. Click **Search**.

   d. In the Process Name field, type `cm.close`.

   e. In the Process Definition section, click the **Final JavaScript** tab.

   f. Add the following code:

```
if(vars.$L_exit!="bad.val")
system.library.HPPPMSMIntegration.integratePPM();
```

g. Repeat step c through step f for the **cm.reject**, **cm.update.save**, and **cm.next.phase** processes.

> **Note:** For Service Manager with a PD patch, repeat step c through step f for the **change.update.save** process only.

h. If you have other processes to call PPM Web services to update the RFC status and task status, make sure to add the code to them as well.

9. Edit Javascript for correct host and port for PPM.

a. In Service Manager, select **Menu Navigation > Tailoring > Script Library**.

b. In the Name field, type `Integration` and click **Search**.

c. Find the line that contains the URL for the PPM instance (should begin with "`this.location = new String`"). Set the URL to

`http://<Host_Name>:<Port>/itg/ppmservices/IntegrationService`

Then the line would look like the follows:

```
this.location = new String("http://<Host_
Name>:<Port>/itg/ppmservices/IntegrationService")
```

d. Click **Save**, then click **Compile**.

e. Click **Cancel** to return to the Search screen.

f. Repeat the step b through step d, search for `Project` and set the URL for it to

`http://<Host_Name>:<Port>/itg/ppmservices/ProjectService`

10. Configure the `ppmfailover` schedule in Service Manager.

By default, the Repeat Interval of the `ppmfailover` schedule is one hour. You can change the default Repeat Interval to another value. However, do not change other field values.

11. Start the `ppmfailover` schedule in Service Manager.

12. Run the following script:

```
sh ./kConfig.sh
```

> **Note:** For more information about the `kConfig.sh` script, see the *Installation and Administration Guide*.

13. Add (if not present) and specify the parameters and values related to Service Manager integration to the PPM`server.conf` configuration file, as shown in the following table. (All parameter names begin with `com.kintana.core.server.` but that is not shown in the table.)

Parameter	Description, Value
SM_RFC_ INTEGRATION_ ENABLED	Setting the parameter to `true` enables SM RFC integration with PPM Center.
SM_ USERNAME	User name that PPM uses to access Service Manager. This user name must include only single-byte characters. For example: `admin`.
SM_ PASSWORD	Password that PPM uses to access Service Manager. You must encrypt this password by using the `kEncrypt.sh` script, which is located in the `bin` directory of the PPM Server. Then remove `#!#` from the beginning and the end of the encrypted password.
SM_URL	Host name or IP address of Service Manager. For example: `http://<Host_Name>:13080`
SM_WEB_URL	Address of Service Manager Web tier. For example: `http://<Host_Name>:<Port>/<WebTier_Package_File_ Name>/index.do` To obtain the Service Manager Web tier URL, a. Log on to Service Manager as system admininistrator. b. Click **Navigation > System Administration > Base System Configuration > Miscellaneous > System Information Record.** c. Click **Active Integrations**, get the WebServer URL value. By default it would be `http://<Host_Name>:13080/sm/index.do.`
ENABLE_WEB_ SERVICES	Setting the parameter to `true` enables web services.

14. Start the PPM Server.

Enabling RFC Creation for a PPM Project Type

In the HPE Service Manager project policy in PPM, you can enable or disable the RFC creation capability at the project type level. The settings of a project type affect all projects of that project type.

However, the policy has an option to allow project managers to override the project type setting for particular projects.

To set the RFC creation capability for a project type:

1. Log on to PPM.

2. From the menu bar, select **Open > Administration > Project Types & Templates > Manage Project Types.**

3. Open the project type you want to configure.

    The Modify Project Type window for that project type opens.

4. In the list of policies, click **HPE Service Manager.**

    The HPE Service Manager policy opens.

    The default settings for the options enable RFC creation for projects of the selected project type and allow project managers to override that setting for particular projects of that type:

5. To enable all projects of this project type for RFC creation and prevent project managers from disabling this capability, make sure that the **Enable RFC creation capability** check box is selected and the **Allow project managers to override these settings** option is set to **No.**

    If you want project managers to control the RFC creation capability for particular projects regardless of the setting of the **Enable RFC creation capability** check box, set the **Allow project managers to override these settings** option to **Yes.**

6. Click **Save.**

Enabling RFC Creation for a PPM Project

If the RFC creation capability is enabled for the project type used by a project, the RFC creation capability is, by default, enabled for the project when it is created.

If the RFC creation capability is not enabled for a project, to enable it:

1. Log on to PPM.

2. Open the project in PPM.

3. On the Project Overview page, click **Project Settings.**

4. In the list of policies, click **HPE Service Manager.**

    The HPE Service Manager policy opens.

5. Use one of the following procedures to verify or enable your ability to create Service Manager RFCs for the project:

   ○ If the **Enable RFC creation capability** check box is selected, you can create Service Manager RFCs for the project.

   ○ If the **Enable RFC creation capability** check box is not selected but you can select it, select it.

   ○ If the **Enable RFC creation capability** check box is not selected and you cannot select it, the setting on the project type used by the project does not currently allow you to enable RFC creation for any projects of that type or to override that setting for particular projects. In this case, see "Enabling RFC Creation for a PPM Project Type" on page 342 and set the override option on the project type to **Yes,** or contact the person who configures these settings. Then return to the HPEService Manager policy for the project and select the **Enable RFC creation capability** check box for the project.

6. Click **Done.**

Creating a Service Manager RFC from a PPM Task

To create an RFC in Service Manager that is associated with a project task in PPM:

1. Log on to PPM, and do the following:

   a. Make sure the RFC creation capability is enabled for the project. See "Enabling RFC Creation for a PPM Project " on the previous page.

   b. On the Project Overview page, click **Edit Work Plan,** and then double-click a task to open it.

      The Task Details page for that task opens.

   c. On the **Operational RFC** tab on the Task Details page, select the **Create an Operational Request for Change (RFC) upon task save** check box.

   > **Known Problem**: If you are PPM 9.40 fresh user, when you select this check box and click Save or Done, the check box is unselected after refreshing the page.
   >
   > **Workaround**: You can either wait for the patch release on top of PPM 9.40 where this issue is fixed or run the following SQL:

```
CREATE SEQUENCE SM_RFC_S INCREMENT BY 1
MINVALUE 1 MAXVALUE 99999999999999999999999999 START WITH 30000
CACHE 20 NOORDER NOCYCLE;

commit;
```

   d. Click **Save.**

After the task is saved, PPM returns to the task page. If Service Manager returns an RFC Change ID, the information on the **Operational RFC** tab is automatically updated with the following information from Service Manager, which is read-only in PPM:

Item	Value
Change Number	Change ID (for example, C10029) of the RFC created in Service Manager, which is displayed as a link. Clicking the link takes you to the Service Manager logon screen.
Change Status	Value from the Status field of the RFC in Service Manager.
Closure Code	Value from the Closure Code field of the RFC in Service Manager.
Closure Comments	Reason entered when the RFC in Service Manager was closed or rejected.
Change Last Update Date	Date (and time) when the RFC was last updated in Service Manager.

In addition, the status line, which is located at the bottom of the task page, indicates the success of the RFC creation.

> **Note:** If Service Manager does not return an RFC Change ID, an error message is displayed on the status line. In most cases, this means the RFC was not created. However, it is possible that the RFC was successfully created but communication problems prevented Service Manager from returning the RFC Change ID to PPM.
>
> To prevent duplicate RFCs from being created for the same task, when the task is saved, PPM searches for an RFC Change in Service Manager to which the task should be attached. If an RFC is found, PPM integrates with it. If PPM does not find an RFC, it creates one.

2. Log on to Service Manager, and do the following to verify that the RFC has been successfully created.

   a. Select **Menu Navigation > Change Management > Changes > Search Changes.**

   b. Click **Search.**

A list of changes opens, where you can find the change (RFC) created for the PPM task with the following fields related to the PPM task:

Field	Value
Change ID	Change ID generated for the task (shown as Change Number in PPM). For example, `C10029`.
Brief Description	Name of the PPM task. For example, `Task 6`.
PPM Task ID	Task ID that PPM assigned to this task. For example, `36002`.

Synchronizing an RFC with its Associated PPM Task

When you update a Service Manager RFC that has an associated PPM task, the information on the PPM task's **Operational RFC** tab is automatically updated.

To update an RFC and then check the status of its associated task:

1. Log on to Service Manager.

2. From the change list, double-click a change (RFC) that has an associated PPM task.

3. Update the RFC in Service Manager.

4. In PPM, refresh the Task Details page for the task to verify that the **Operational RFC** tab displays the updated status of the associated RFC.

Updating the Status of a PPM Task when the Associated RFC is Closed or Rejected

When you close or reject a Service Manager RFC that has an associated PPM task, the status of the PPM task is automatically set to Complete or Cancelled, and the information on the task's **Operational RFC** tab is updated accordingly.

To close or reject an RFC and then check the status of its associated task:

1. Log on to Service Manager.

2. From the change list, double-click a change that has an associated PPM task.

3. Modify the status of the change.

   Setting the status to Reject sets the PPM task status to Cancelled. Setting the status to Close sets the PPM task status to Complete.

4. In PPM, refresh the Task Details page for the associated task to verify that the **Operational RFC** tab is updated and that the task status is also updated as shown in the following table:

Field	Value
Status	One of the following: ○ Complete (if the change was closed) ○ Cancelled (if the change was rejected)
% Complete	One of the following: ○ 100 (if the change was closed) ○ 0 (if the change was rejected)

Error Logging

PPM includes a sample logging file to debug or monitor the integration. Configure logging as in the following example:

1. Open the sample logging file, located at:

   `<PPM Server>/conf/smrfc/logging.conf.sample`

2. Copy the last two lines from the sample logging file into the PPM logging file, which is located at:

   `<PPM Server>/conf/logging.conf`

   These two lines are as follows, and they enable logging for the integration-specific classes SmRfcFieldMap and `SmRfcFieldMapReader`:

   ```
   com.kintana.core.logging.CLASS_LOGGING_
   LEVEL=com.mercury.itg.core.service.util.SmRfcFieldMap,DEBUG
   ```

   ```
   com.kintana.core.logging.CLASS_LOGGING_
   LEVEL=com.mercury.itg.core.service.util.SmRfcFieldMapReader,DEBUG
   ```

3. Revise the copied lines as needed.

4. By default, the copied lines set the logging mode for the integration to `DEBUG`. When you do not need to debug the integration, set the logging mode on the copied lines to `ERROR` or comment out the copied lines.

Part 6: Integration with HPE Universal CMDB

This part includes the following solution integrations:

- Integrating PPM with HPE Universal CMDB, Using ALM

- Integrating PPM with HPE Universal CMDB for Service Portfolio

- Integrating APM with Universal CMDB

Chapter 13: Integrating PPM with HPE Universal CMDB, Using ALM

This section contains the following topics:

- "Introduction to Integrating PPM with Universal CMDB, Using ALM" below
- "Configuring Universal CMDB for the Integration" on the next page
- "Configuring PPM for the Integration" on page 352
- "Using the Integration " on page 355

Introduction to Integrating PPM with Universal CMDB, Using ALM

Universal CMDB consists of a business-service-oriented data model with built-in discovery of the following:

- Configuration items (CIs) and their dependencies
- Visualization and mapping of business services
- Tracking of configuration changes

When you integrate PPM with Universal CMDB, you can select CIs and run impact analysis reports from change requests in PPM, to determine which components of a system will be affected by a software change, and to what extent. The integration assists IT managers and Change Advisory Boards in deciding whether a change request should be approved for development or deployment.

For example, your software change might involve upgrading a database server. Before you can perform the upgrade, you need to stop the server. In some cases this could prevent users from accessing crucial services, or even cause a crash of your production system. Impact analysis determines the effect on the entire system of stopping the server, and gives you a report showing the components that will be impacted. This enables you to plan the change with minimal disturbance to your operations.

For more information about the benefits of this integration, see "Integration of PPM with Universal CMDB for Impact Analysis of Requests, Using ALM" on page 24.

For information about the versions supported for integration, see the *System Requirements and Compatibility Matrix.*

> **Note:** No software needs to be installed on the Universal CMDB server to integrate PPM and Universal CMDB. However, see the *System Requirements and Compatibility Matrix*.

For more information about Universal CMDB, see its product documentation at the Web site described in "Available PPM Integrations " on page 18.

Using Impact Analysis in a Change Request Lifecycle

Since a software change might be developed and deployed over a lengthy period of time, you might want to generate an Impact Analysis Report at the following stages in the lifecycle of the change request:

- **Before you approve or develop the change.** Before you approve or develop the change, it is useful to forecast the effect that introducing the change will have on your production system. You describe the intended change, specify the components that you think will be affected, and run impact analysis to forecast the effect of the change.

  The resulting report gives you an indication of how your system will cope with the change after the change has been developed and deployed, and helps you decide whether to approve the change for development.

- **After the change is approved for deployment to a production system, but before you deploy the change.** While the software change is being developed and then evaluated for quality, modifications may occur in your system infrastructure. For example, servers might be added or removed, or applications might be changed. As a result, the original impact analysis may no longer give an accurate indication of what will happen when you introduce the change. So after the change has been developed, evaluated, and approved for deployment, you perform another impact analysis to give you an up-to-date picture.

Configuring Universal CMDB for the Integration

To configure Universal CMDB for the integration, configure views in Universal CMDB and create a PPM user and password in Universal CMDB. See the Universal CMDB documentation.

Configuring PPM for the Integration

Perform the procedures in this section to configure PPM for the integration.

-

-

Configuring server.conf Parameters in PPM

> **Note:** For more information about the steps in this procedure, see the *Installation and Administration Guide.*

Add (if not present) and specify the parameters related to Universal CMDB integration to the PPM `server.conf` configuration file, as follows:

1. Stop the PPM Server.

2. Run the script sh ./kConfig.sh.

   Set the parameters and values as shown in the following table. (All parameter names begin with `com.kintana.core.server.` but that is not shown in the table.)

Parameter	Value
UCMDB_ GATEWAY_ URL	URL of the Probe Gateway component of the Discovery and Dependency Mapping (DDM) Probe. The Probe Gateway provides communication (HTTP or HTTPS) between the Probe Manager and the Universal CMDB user for processes such as downloading tasks and returning task results. The value for *<UCMDB_Host>*:*<port>* is usually the same as for the UCMDB_ SERVER_URL parameter. `http://`*<UCMDB_Host>*`:`*<port>*`/mam/gateway?`
UCMDB_ MAX_CI_ NUMBER	Maximum number of CIs a request is allowed to have. Valid values are from 1 to 100. If a value greater than 100 is specified, the PPM Server will not restart and displays an error message. The default is 20.
UCMDB_ PASSWORD	Password for Universal CMDB user specified in UCMDB_USER. This password must be encrypted as described in "Encrypting the Password Specified as a server.conf Parameter" on the next page.

Parameter	Value

UCMDB_ SERVER_ URL	URL of the Universal CMDB server:
	
	`http://<UCMDB_Host>:<port>/ucmdb/`
	
	or
	
	`https://<UCMDB_Host>:<port>/ucmdb/`
	
	where *<UCMDB_Host>* represents the host machine on which Universal CMDB is running.
	
	Note: If the Universal CMDB server is configured to support HTTPS, make sure you configure the UCMDB_SSL_KEYSTORE_PATH parameter.

UCMDB_ SERVER_ VERSION	Do not use.
UCMDB_ SSL_ KEYSTORE_ PATH	Universal CMDB keystore path, used only if the UCMDB_SERVER_URL parameter uses HTTPS.
UCMDB_ USER	Universal CMDB user name, for example, admin. This user name must include only single-byte characters.

3. Restart the PPM Server.

Encrypting the Password Specified as a server.conf Parameter

The password that you assigned to the `UCMDB_PASSWORD` parameter must be encrypted, as follows:

1. Navigate to the *<PPM_Home>*/bin/ucmdb directory.

2. Run the `kEncryptUcmdbPassword.sh` utility.

3. Specify the password you want to encrypt.

   The utility encrypts the password and displays the encrypted text.

4. Copy the text in the `Encrypted text` section on the screen and paste the text as the value for `UCMDB_PASSWORD` in the `server.conf` file, making sure that you do not copy any carriage returns into the file.

Configuring a Request Type

The integration requires a request type with the Universal CMDB Impact Analysis field group. The ALM - Request for Change (RFC) request type provided with ALM includes this field group, which is enabled by default.

If necessary, you can add the Universal CMDB Impact Analysis field group to a different request type, as follows:

1. Log on to PPM.

2. From the menu bar, select **Open > Administration > Open Workbench.**

   The PPM Workbench opens.

3. From the shortcut bar, select **Demand Mgmt > Request Types.**

   The Request Types Workbench opens.

4. Select the request type to which you will add the Universal CMDB Impact Analysis field group.

5. Open the request type and its request header type.

6. Click **Field Groups.**

7. In the Field Groups window, scroll down and select the check box to enable the **Universal CMDB Impact Analysis** field group.

8. Click **OK** to add the Universal CMDB Impact Analysis field group to the request header type.

9. Expand **Universal CMDB Impact Analysis** in the **Prompt** column on the **Fields** tab in the request header type.

10. Select the **CI List** field, and click **Edit.**

11. On the **Attributes** tab, select **No** for the **Display Only** option and select **Yes** for the **Display** option.

12. Select the **Security** tab and edit field-level security for the **CI List** field as needed.

13. Click **OK** to save and close the Edit window and click **OK** to save and close the Request Header Type window.

14. Return to the request type, which now has the Universal CMDB Impact Analysis section, including the **CI List** field, based on the changes you made to the request header type.

15. Use the **Layout** tab to reposition the **Impacted Configuration Items** section on the request type,

as desired.

> **Note:** On the **Status Dependencies** tab for the **CI List** field, leave the **Required, Reconfirm,** and **Clear** options set to **No.**
>
> You can change the **Visible** and **Editable** options as desired.

Using the Integration

The Configuration Management Database (CMDB) in Universal CMDB contains information about all your CIs (servers, applications, hosts, and so on) and their relationships. CIs can vary widely in complexity, size, and type, from an entire system (including all hardware, software, and documentation) to a single module or minor hardware component.

For example, the CMDB contains information about which CIs are affected when other CIs stop functioning or have operational problems.

You generally perform impact analysis as follows:

- Select the CIs that you suspect might be affected by the planned software change.

  > **Tip:** HPE recommends using CI names in Universal CMDB that will be meaningful to you in PPM.

- Run impact analysis on the selected CIs. Impact analysis analyzes the relationships among the selected CIs in the CMDB, and generates a report showing the CIs that will be affected by the planned change.

- Run impact analysis at additional points. Examples:

  - Before approving a change, to forecast the effect that the change will have on your system.

  - After the change has been created, in case your system infrastructure has changed since the previous impact analysis was performed.

Selecting CIs in Universal CMDB and Adding the CIs to a Request

To select the desired CIs in Universal CMDB to add to the request:

1. Log on to PPM.

2. From the menu bar, select **Create > Request.**

3. Create a request using a request type that includes the Universal CMDB Impact Analysis field group.

4. In the **Impacted Configuration Items** section of the new request, click **Select Configuration Items.**

   The CI selector applet from Universal CMDB launches, with **Browse** and **Search** tabs.

   > **Note:** If you integrated PPM with Universal CMDB version 10.00, every time the CI selector applet from Universal CMDB launches, it is loaded with a new session cookie even if you set your browser to save session cookies.

5. Use the tabs to select the desired CIs for the request.

   ○ To use the **Browse** tab, select a view such as **Network Topology** from the **View** list.

   Expand the view as desired. You can select multiple items in the tree by using Ctrl + **click** or Shift + **click.**

   ○ To use the **Search** tab, complete the **Search for** field to limit the search results, and select a **CI Type.** Click the search icon for the **CI Name** field and select items as needed. You can select multiple items by using Ctrl + **click** or Shift + **click.**

6. Click **OK** to add the selected CIs to the request, under the **Impacted Configuration Items** heading in the **Universal CMDB Impact Analysis** section. The added CIs are shaded in color. Duplicate CIs are not added.

   For each CI, data appears in the **CI Name** and **CI ID** columns. If you selected CIs using the **Browse** tab in the Universal CMDB applet, data also appears for the **View Name, View Type, View Tree Name,** and **View TQL Name** columns. Data does not appear for these columns if you selected CIs using the **Search** tab, because the same CI can exist in multiple views.

   If adding the CIs to the request would exceed the maximum number allowed by the UCMDB_MAX_ CI_NUMBER parameter in the server.conf file, none of the selected CIs are added.

   When you select and add CIs in Universal CMDB to the PPM request, the CIs are never changed or deleted in Universal CMDB.

7. As necessary, click the **Delete** icon to the left of any CI you want to delete.

   The **Delete** icon becomes an undo icon to allow you to undo the pending deletion.

8. After the desired CIs are listed in the request, click **Submit** (or **Save** for an existing request) to add the list of CIs to the request.

> **Note:** If a request with a list of CIs is copied, the list of CIs is not copied to the new request.

> **Note:** The CIs in a request are available as part of standard Web services for a request. The token is `REQ_VP_KNTA_UCMDB_CI_LIST`. The value of the field represented by this token is a comma-separated list of CI IDs.

Generating Impact Analysis Reports

To generate an Impact Analysis Report for the CIs that have been added to a request:

1. Click the **Launch HPE Universal CMDB Impact Analysis** button in the **Impacted Configuration Items** section of the request. Log in to Universal CMDB if prompted to do so.

   An Impact Analysis Report is generated in Universal CMDB.

2. Add the Impact Analysis Report to the request, using the **Impact Analysis Report** field in the **Impact & Resource Assessment** section of the request.

3. Based on the impact analysis report, make your recommendations in an Impact Assessment Report. Add your report to the request using the **Impact Assessment Report** field in the **Impact & Resource Assessment** section of the request.

The **Launch HPE Release Control** button appears if PPM is also integrated with HPE Release Control, as described in "Integrating PPM with HPE Release Control, Using ALM" on page 269.

Chapter 14: Integrating PPM with HPEUniversal CMDB for Service Portfolio

For service portfolio functionality, services can be associated with the **Service** field in PPM requests, and then labor costs can be tracked for each service. The list of services can be retrieved from Universal CMDB in real time, that is, each time a PPM user needs the list to specify the **Service** field.

The service list does not reside in PPM. After configuring the integration with Universal CMDB, the service list can be retrieved from Universal CMDB.

> **Note:** This integration does not use the ALM entities and does not require installing the ALM software.

Configuring PPM Center and Universal CMDB for the Integration

This section contains the following steps:

1. "(Optional) Configuring Universal CMDB to Support HTTPS for the Integration " below

2. "Configuring server.conf Parameters in PPM" on page 361

3. "Creating a Request Header Type with the Service Field" on page 363

4. "Creating a Request Type that Uses the New Request Header Type" on page 364

5. "Setting up UCMDB CI Type Properties" on page 364

(Optional) Configuring Universal CMDB to Support HTTPS for the Integration

If you need to enable real-time integration with Universal CMDB using HTTPS, perform one of the following procedures, depending on whether or not the Universal CMDB server already supports HTTPS. Then proceed to the rest steps.

Configuring Universal CMDB if the Server Already Supports HTTPS

If the current Universal CMDB server is already configured to support HTTPS, you can use the existing keystore file directly:

1. Locate the keystore file.

2. Locate the certificate in the keystore.

3. Run the script below to export the certificate:

   ```
   keytool -export -alias [Certificate alias name] -keystore <UCMDB_Server_
   HOME>\UCMDBServer\j2f\EJBContainer\server\mercury\conf\<keystore or certs file
   name> -file c:\<Certificate alias name>
   ```

4. Import the certificate to the PPM server:

   ```
   keytool -import -trustcacerts -alias[Certificate alias name] -keystore
   [keystore or certs file path] -file c:\<Certificate alias name>
   ```

Configuring Universal CMDB if the Server Does Not Support HTTPS Yet

If the current Universal CMDB server is not configured to support HTTPS yet, do the following:

1. Generate the keystore file on Universal CMDB.

   Run the following scripts:

   <UCMDB_Server_HOME>\UCMDBServer\j2f\JRE\bin>keytool -genkey -keystore

   *<UCMDB_Server_
   HOME>*\UCMDBServer\j2f\EJBContainer\server\mercury\conf\ucmdb.keystore

   Enter the keystore password and provide other information as necessary when prompted.

2. Enable SSL on the Web server for Universal CMDB Server.

   a. To enable JBoss in SSL, edit the following file:

      <UCMDB_Server_HOME>\j2f\EJBContainer\server\mercury\deploy\jbossweb-
      tomcat55.sar\server.xml

      Uncomment the following entry:

```
<!-- SSL/TLS Connector configuration using the admin devl guide keystore:
<Connector port="8443" address="${jboss.bind.address}"
    maxThreads="100" strategy="ms" maxHttpHeaderSize="8192"
    emptySessionPath="true"
    scheme="https" secure="true" clientAuth="false"
    keystoreFile="${jboss.server.home.dir}/conf/ucmdb.keystore"
    keystorePass="changeit" [keyAlias="myCert"]
    sslProtocol = "TLS" />
-->
```

Make sure that `keystoreFile` and `keystorePass` are set to the correct path of a java keystore and its password respectively.

> **Note:** If the keystore holds more than one certificate, the first one will be used.
>
> (Optional) To choose a specific certificate, use the `keyAlias` attribute with the alias of the certificate.

With the above settings, JBoss will look for the keystore in:

<UCMDB_Server_HOME>\j2f\EJBContainer\server\mercury\conf\ucmdb.keystore

b. Restart the Universal CMDB server.

Now, to access the Universal CMDB web page in SSL you can go to the URL

https://*<UCMDB_Server_HOST>*:8443/ucmdb

3. Generate a certificate in the keystore file with alias `ucmdbCer`, then export the certificate and import it to the PPM Server.

a. Run the script below to generate the certificate:

```
keytool -genkey -alias ucmdbCer -keyalg RSA -keystore <UCMDB_Server_
HOME>\UCMDBServer\j2f\EJBContainer\server\mercury\conf\ucmdb.keystore
```

b. Export the certificate:

```
keytool -export -alias ucmdbCer -keystore <UCMDB_Server_
HOME>\UCMDBServer\j2f\EJBContainer\server\mercury\conf\ucmdb.keystore -file
c:\ucmdbCer
```

c. Import the certificate to the PPM server:

```
keytool -import -trustcacerts -alias ucmdbCer -keystore [keystore or certs
file path] -file c:\ucmdbCer
```

Configuring server.conf Parameters in PPM

> **Note:** For more information about the steps in this procedure, see the *Installation and Administration Guide*.

Add (if not present) and specify the parameters related to Universal CMDB integration for service portfolio to the PPM `server.conf` configuration file, as follows:

1. Stop the PPM Server.

2. Run the script sh ./kConfig.sh.

   Set the parameters and values as shown in the following table. (All parameter names begin with `com.kintana.core.server.` but that is not shown in the table.)

Parameter	Description
SERVICE_LIST_ SOURCE	Specifies the source of the list of available services: ○ `uCMDB` to retrieve the list of services from Universal CMDB each time they are needed for PPM requests ○ `lookup` to retrieve the list of services from PPM for PPM requests and project tasks For integration with Universal CMDB, enter `uCMDB`.
SERVICE_LIST_ UCMDB_CACHE_ TIMEOUT	Length of time (in seconds) the service list remains in PPM cache before it is retrieved again, for example, `300`. For more information, see the *Installation and Administration Guide*.
SERVICE_LIST_ UCMDB_CI_MAPPINGS	Service list mappings between the following pairs of attributes: ○ The name attribute for the Service List uCMDB autocomplete list in PPM, and the CI name attribute in Universal CMDB ○ The description attribute for the Service List uCMDB autocomplete list in PPM, and the CI description attribute in Universal CMDB The mappings are formatted in pairs and separated by commas as follows (with no spaces or new lines): `name:`*`<uCMDB_CI_Name_Attribute>,`* `description:`*`<uCMDB_CI_Description_Attribute>`* For example, the value of this parameter could be:

Parameter	Description

	`name:data_name,`
	
	`description:service_description`
	
	This example maps name in PPM to the CI name attribute data_name in Universal CMDB, and it maps description in PPM to the CI description attribute service_description in Universal CMDB.
	
	Note: All items that are to be mapped must already exist in PPM or Universal CMDB.

SERVICE_LIST_ UCMDB_CI_TYPE	Name of the CI type used to store the service list, for example, `Service.`HPE recommends using the value `Service.` **Note:** You must create this CI type on the Universal CMDB server. For more information about creating a CI type, see the Universal CMDB documentation.
SERVICE_LIST_ UCMDB_MAX_CI_ NUMBER	The maximum number of Universal CMDB configuration items allowed in the service list, for example `1000.` For more information, see the *Installation and Administration Guide*.

UCMDB_SERVER_URL	URL of the Universal CMDB server:
	
	`http://<UCMDB_Host>:<port>/ucmdb/`
	
	or
	
	`https://<UCMDB_Host>:<port>/ucmdb`
	
	where *<UCMDB_Host>* represents the host machine on which Universal CMDB is running.
	
	For example: `http://<UCMDB URL FQDN>:8080/ucmdb/`
	
	Note: If the Universal CMDB server is configured to support HTTPS, make sure you configure the UCMDB_SSL_ KEYSTORE_PATH parameter. For information about enabling HTTPS for Universal CMDB, see "(Optional) Configuring Universal CMDB to Support HTTPS for the Integration " on page 358. You should also configure PPM to use HTTPS.

UCMDB_SERVER_ VERSION	Version of the UCMDB server, for example: `10`
UCMDB_SSL_ KEYSTORE_PATH	Universal CMDB keystore path, used only if the UCMDB_ SERVER_URL parameter uses HTTPS.

Parameter	Description
UCMDB_WS_MAX_ CONNECTION_ NUMBER	Maximum number of connections allowed to connect to the Universal CMDB server using the Web Service API, for example, `10`.
UCMDB_WS_ PASSWORD	Password for Universal CMDB user specified in UCMDB_WS_ USER, logging in through Web service.
	You must encrypt this password by using the `kEncrypt.sh` script, which is located in the `bin` directory of the PPM Server.
UCMDB_WS_USER	Universal CMDB user name for logging in through Web service, for example, `admin`.
	This user name must include only single-byte characters.
UCMDB_GATEWAY_ URL	URL of the Probe Gateway component of the Discovery and Dependency Mapping (DDM) Probe. The Probe Gateway provides communication (HTTP or HTTPS) between the Probe Manager and the Universal CMDB user for processes such as downloading tasks and returning task results.
	For example: `http://<UCMDB_Host>:<port>/mam/gateway?`
	The value for `<UCMDB_Host>:<port>` is usually the same as for the UCMDB_SERVER_URL parameter.

3. Restart the PPM Server.

Creating a Request Header Type with the Service Field

To create a request header type that includes the **Service** field:

1. Log on to PPM.

2. From the menu, click **Open** > **Administrator** > **Open Workbench**.

3. In the workbench, click **Demand Mgmt** > **Request Header Types**.

4. In the Request Header Type Workbench dialog box, click **New Request Header Type**.

5. In the Request Header Type dialog box, do the following:

   a. Enter the **Request Header Type Name**. For example, **Service**.

   b. Click the **Reference Code** field.

   c. Click the **Field Groups** button.

   d. In the Field Groups dialog box, select the **Service** field group.

e. Click **OK**.

f. Double-click the **Service** field in the Request Header Type dialog box and confirm the validation for this field is **Service List UCMDB**.

> **Note:** This validation is an auto-complete list. When users select auto-complete for the Service field, this validation invokes a special command to retrieve the Service list from Universal CMDB.
>
> As with any request field, the administrator can make the Service field optional or required.

g. Click **OK**.

h. Click **OK**.

Creating a Request Type that Uses the New Request Header Type

1. In PPM Center, click **Open** > **Administrator** > **Open Workbench**.

2. In the workbench, click **Demand Mgmt** > **Request Types**.

3. In the Request Type Workbench dialog box, click **New Request Type**.

4. In the New Request Type dialog, do the following:

   a. Enter a name in the **Request Type Name** field. For example, **Service Request**.

   b. Click the selection icon next to the **Request Header Type** field.

   c. In the Validation dialog box, select **Service** and click **OK**.

   d. Confirm the Service field group is added and click **OK**.

> **Note:** When users create a request of that new request type and click the auto-complete for the Service field the service list options are retrieved from Universal CMDB at that time.

Setting up UCMDB CI Type Properties

In UCMDB, the **Name** attribute of the CI Type must be a key attribute. To confirm that it is defined as a key attribute:

1. Open the UCMDB user interface with administrative privileges.

2. Go to the **CI Type Manager**.

3. Select the CI Type you are using for your service list. For example, service.

4. Go to **Attributes** and make sure that the **Name** attribute is marked as the key attribute.

   If it is not defined as a key attribute, do the following:

   a. Go to the **Details** tab.

   b. In the Identification dialog box, change the default identification to **By key attributes**.

      > **Caution:** Note down the identification that is defined before changing it since you will need to change it back to its identification after the procedure.

   c. In the Available Attributes dialog box, select **Name** as the key attribute.

   d. Click **Save**

   e. In the Identification pane, change the identification back to whatever the default was before changing it in step b.

      > **Note:** The original out-of-the-box default value was **By identification rule**.

   f. Click **Save**.

Adding a Service Context from UCMDB to PPM

This section includes the following steps:

Adding a Service Field from UCMDB to a PPM Project

1. Log on to PPM as a user with administrative privileges.

2. From the menu item, select **Open** > **Administrator** > **Open Workbench**.

3. In the workbench, go to **Demand Mgmt** > **Request Header Types**.

4. In the Request Header Type Workbench window, click **List**.

5. In the Results tab, double click **Project Details Request Header Type**.

6. In the Request Header Type: Project Details window, click **Field Groups**.

7. Select the **Service** field and click **OK**.

8. Click **Save**.

9. Click **OK**.

Adding a Service Field from UCMDB to a PPM Task

> **Note:**
>
> - This should be configured for each project separately and does not apply for all projects.
>
> - This can only be configured after "Adding a Service Field from UCMDB to a PPM Project" on the previous page.

1. Log on to PPM as a user with administrative privileges.

2. From the Search Projects window, search and open your project.

3. Within the project, select **Settings**.

4. In the Project Settings window, Project Fields pane, Additional Fields section, select the **Service** option.

5. Click **Save**.

6. Click **Done**.

Verification

> **Note:** These procedures are to verify that the implementations were done successfully.

> **Caution:** These verifications must be done in this order.

To verify that the field was added to the PPM project

1. Log on to PPM as a user with administrative privileges.

2. Navigate to **Search** > **Projects**.

3. From the Search Projects window, search and open your project.

4. Within the project, go to the **Project Details** tab.

5. Confirm that the **Service** field exists and that you can use it to add CIs from UCMDB.

    a. In the **Service** field, click the **Choose Services** button on the right-hand side of the field.

    b. Confirm that you can select CIs from UCMDB.

To verify that UCMDB CIs can be added to a PPM task

1. In PPM, navigate to **Search** > **Projects**.

2. Search for and select your project.

3. In the **Project Summary** tab, click **Edit Work Plan**.

4. Double-click the task to which you want to add the UCMDB Service CI.

5. In the **Service** field, click the **Choose Service** button.

6. Verify that the services in the list are the CIs from UCMDB that you wish to integrate with PPM, and select a service.

7. Click **Save**.

8. Confirm that the service was successfully added to the **Service** field and click **Done**.

Chapter 15: Integrating HPE APM with HPE Universal CMDB

This chapterincludes the following:

-

-

-

-

-

-

-

Overview

The integration between APM for PPM (APM) and HPE Universal CMDB (UCMDB) enables you to share information from UCDMB with APM.

You can use the integration to automate the creation and update of requests in APM, freeing you from repetitive and manual input of information in APM. This also ensures that APM is kept up to date with real, accurate, discovered data in your environment.

The following table provides an overview of the APM integration with UCMDB:

Integration direction	From UCMDB to APM
Integration technology	Push adapter
Pushed data	CIs created in UCMDB are pushed to APM to create requests in APM
HPE Universal CMDB adapter	APM Push Adapter (APMPushAdapter)

How Data is Synchronized Between APM and UCMDB

When referring to the concept of data information, it is important to distinguish between a UCMDB CI (Configuration Item) and an APM Application. Both are defined in a different Data Model, and there must be a conversion before transferring CIs in UCMDB to Applications in APM.

The following graphic shows the high-level components of the integration:
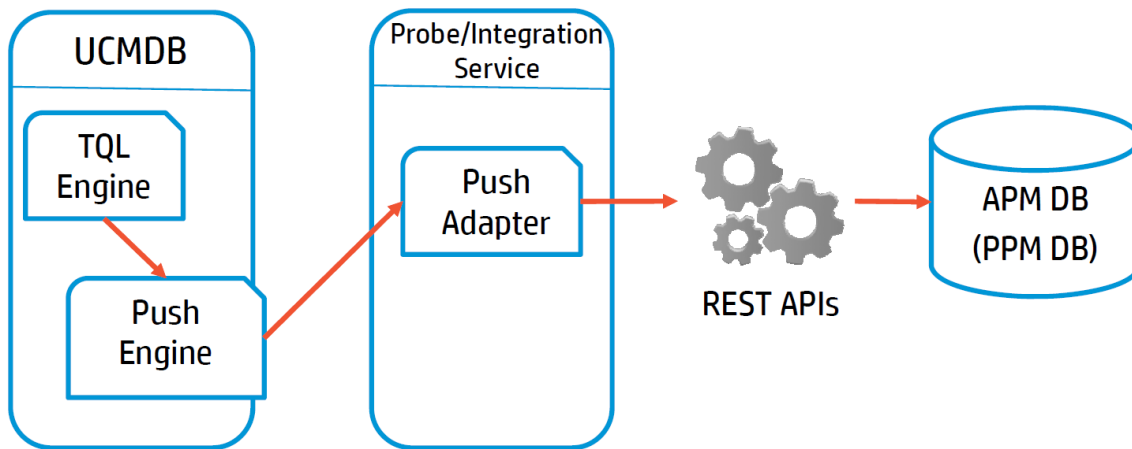
> **Note:** The Push Adapter is executed in the Data Flow Probe/Integration Service process.

UCMDB stores its information using CIs. The integration chooses which data to pull from UCMDB by defining integration TQL queries. Each TQL query defines a superset of data relevant for the integration.

The **UCMDB Push Engine**:

- Retrieves the required data from the UCMDB, using the given TQL query.

- Filters the data to include only the data that has changed since the last execution of this synchronization.

- Splits the data into multiple chunks without breaking consistency.

- Sends the information to the Probe/Adapter.

The Push Adapter is a generic framework for easily configuring push adapters, using only XML and Groovy. It allows easy mapping of the data from the UCMDB data model into the APM data model, and the transfer of this converted data into the APM database through REST APIs called from APM.

For more information about push adapter, see **Developing Push Adapters** in the *Universal CMDB Developer Reference Guide*.

For details about REST APIs that this integration call from APM, see "REST APIs Called in the Integration" on page 391.

For entity mappings and field mappings between APM and UCMDB, see "Default Entity and Field Mappings between APM and UCMDB" on page 386.

Supported Versions

The APM adapter supports the following:

- Universal CMDB version 10.00 and later

- HPE Project and Portfolio Management Center (PPM Center) version 9.22 (or later) where APM for PPM 9.20 (or later) is installed

How to Integrate UCMDB and APM

To set up integration between UCMDB and APM, you must complete the following steps:

- "Deploy the APM Push Adapter" below

- "Create an Integration Point between APM and UCMDB" on the next page

- "(Optional) Push CI Data from UCMDB to APM" on page 375

- "Schedule Data Push Jobs" on page 378

Deploy the APM Push Adapter

To integrate APM with UCMDB, administrators must deploy the APM Push Adapter.

> **Note:** If your UCMDB version is 10.10 or later, you can skip the steps below and proceed to "Create an Integration Point between APM and UCMDB" on the next page. The APM Push Adapter is deployed by default.

To download the APM Push Adapater zip package and deploy it,

1. Obtain the APM Push Adapter from the HPE Live Network.

   a. Go to the HPE Live Network : https://hpln.hpe.com/product/project-and-portfolio-

management/resources/file-repository

    b. If you have not logged in, the HP Passport sign-in page opens.

        Provide your User ID and Password, and click **Sign-in**.

    c. Click **APMPushAdapter** to download the `APMPushAdapter.zip` package.

2. Start the UCMDB server.

3. Open a browser and log on to UCMDB as an administrator.

   Example of the UCMDB login page URL:
   http://<*UCMDBServerName.Domain*>:8080/ucmdb-ui/login_page.jsp

4. From the left navigation bar, click the **Administration** tab.

5. Click **Package Manager**.

6. Click **Deploy packages to server (from local disk)** ⊕.

   The Deploy Packages to Server dialog opens.

7. Click **Add** ➕.

8. Browse to the folder where `APMPushAdapter.zip` is located. Select the zip file, and click **Open**.

9. Click **Deploy**.

10. Click **OK** when the following confirmation message displays: `Resources were deployed successfully.`

Create an Integration Point between APM and UCMDB

1. Log in to UCMDB as an administrator.

2. Navigate to **Data Flow Management > Integration Studio**.

   UCMDB displays a list of existing integration points.

3. Click the **New Integration Point** ✳ button.

   The New Integration Point dialog box opens.

4. Complete the **Integration Properties** and **Adapter Properties** fields as shown in the following table:

Field (*Required)	Description
Integration Properties section	
*Integration Name	Type the name (unique key) of the integration point.
Integration Description	Type a description of the current integration point.
*Adapter	Click the **Select Adapter** ... button and select **HPE Software Products > APM > APM Push Adapter** from the Select Adapter dialog
*Is Integration Activated?	Select this option to indicate the integration point is active.
Adapter Properties section	
*Protocol Type	Select **http** or **https** from the drop-down list
*Hostname/IP	Type the hostname or IP Address of the PPM Server. For example, `16.166.16.16` or `hostname`.
*Port	Type the communication port of the PPM Server. The default value is `80`. Example: `30000`
*Path	Type the path of the PPM Server. The default value is `itg`.

*Credentials ID	Click the **Select Credential Id** ... button, then select **Generic Protocal** or **DefaultDomain > Generic Protocal** in the Protocal pane, and from the Credentials list, select a credential or create a new credential that is to be used by UCMDB to access APM.
	
	To create a credential for this integration point,
	
	a. Click the **Select Credential Id** ... button. The Choose Credentials dialog opens.
	
	b. Click the **Create new connection details for selected protocol type** * button. The Generic Protocol Parameters dialog opens.
	
	c. Provide values for the following fields and click **OK**:
	
	  • **Network Scope:** Use the default value `ALL`.
	  • **User Label:** Type a label for the credential.
	  • **User Name:** Provide the user name for the APM account that is to be used by UCMDB to access APM.
	  • **Password:** Click ... and provide the password for the APM account that is to be used by UCMDB to access APM.

Field (*Required)	Description
	d. Click **OK** twice. **Tip:** Users of any of the following security groups can be used as Credentials ID: ○ APM User ○ APM Administrator ○ APM Analyst
*Data Flow Probe	The name of the Data Flow Probe/Integration service used to execute the synchronization from. Select **IntegrationService** for this integration. **Note:** If the **IntegrationService** option does not exist, consult with your UCMDB administrator for the best selection for your requirements.
Additional Probes	Not required for this integration point.

Below is an example of the completed dialog:

5. Click **Test Connection** to make sure there is a valid connection.

6. Click **OK**.

The integration point is created and its details are displayed. (It is not saved to the server until you click on the **OK** button)

UCMDB creates a default data push job when creating the integration point. The default data push job includes everything, it runs immediately and performs a Full Synchronization.

> **Note:** The first Full Synchronization may take a while to complete.

If needed you may create or edit the existing job. To create or edit and run a customized data push job, see "(Optional) Push CI Data from UCMDB to APM" on the next page.

For instructions about scheduling the data push job, see "Schedule Data Push Jobs" on page 378.

(Optional) Push CI Data from UCMDB to APM

Data push jobs copy or update CI or CI relationship records from the local UCMDB system to your APM system.

To run a customized data push job, complete the following steps:

1. Log in to UCMDB as an administrator.

2. Navigate to **Data Flow Management > Integration Studio**.

   UCMDB displays a list of existing integration points.

3. Select the integration point you created for APM.

4. Select the default data push job **APM Push**.

   Or, if the default data push job does not satisfy your needs, you may add a new data push job as follows:

   a. Click the **New Integration Job** button in the right panel.

   b. In the **Name** field, type a unique name for the job.

   c. Click the **Add Query** button to add existing TQL queries to the job.

      UCMDB creates a default data push job when creating the integration point for APM. The following table lists the Topology Query Language (TQL) queries in the default data push job. If required, you may create, update, or remove TQL queries for the push job. You may also need to update the mapping. See .

      > **Note:** To access these OOTB TQL queries for push, navigate to **Modelling > Modeling Studio > Resources**, select **Queries** from the drop-down list for the **Resource Type** field and then navigate to **Root > Integration > APM Push**.

TQL Query	Description
APM Location Push	Pushes Location CIs. Mapping XML: `pushMappingAPMLocation.xml`
APM Process Push	Pushes BusinessProcess CIs. Mapping XML: `pushMappingAPMProcess.xml`
APM Process	Clears old relations between processes in APM.

TQL Query	Description
Relation Clear Push	This TQL query synchronization must have been run BEFORE the 'APM Process Relation Push' TQL query synchronization.
	Mapping XML: pushMappingAPMProcessRClear.xml
APM Process Relation Push	Pushes relations between Processes (pushed by APM Process Push) to other business elements or to processes.
	BusinessProcess CIs must have been pushed before this TQL query synchronization in the 'APM Process Push' TQL query synchronization.
	Mapping XML: pushMappingAPMProcessR.xml
APM Server Push	Pushes Node CIs (Computers, Network Devices, etc.).
	Mapping XML: pushMappingAPMServer.xml
APM Server Relation Clear Push	Clears old relations between Servers in APM.
	This TQL query synchronization must have been run BEFORE the 'APM Server Relation Push' TQL query synchronization.
	Mapping XML: pushMappingAPMServerRClear.xml
APM Server Relation Push	Pushes relations between Servers (pushed by APM Server Push) to other servers.
	Node CIs must have been pushed before this TQL query synchronization in the 'APM Server Push' TQL query synchronization.
	Mapping XML: pushMappingAPMServerR.xml
APM Application Push	Pushes BusinessApplication CIs.
	Location, BusinessProcess, and/or Node CIs must have been pushed before this TQL query synchronization.
	Mapping XML: pushMappingAPMApplication.xml
APM Application Relation Clear Push	Clears old relations (except for the downstream relations) between Applications in APM.
	This TQL query synchronization must have been run BEFORE the 'APM Application Relation Push' TQL query synchronization.
	Mapping XML: pushMappingAPMApplicationRClear.xml
APM Application Relation Push	Pushes relations (except for the downstream relations) between Applications (pushed by APM Application Push) to other business elements or to nodes.

TQL Query	Description
	BusinessApplication CIs must have been pushed before this TQL query synchronization in the 'APM Application Push' TQL query synchronization. Mapping XML: `pushMappingAPMApplicationR.xml`
APM Application Relation Clear Down Push	Clears old downstream relations between Applications in APM. This TQL query synchronization must have been run BEFORE the 'APM Application Relation Down Push' TQL query synchronization. Mapping XML: `pushMappingAPMApplicationDownRClear.xml`
APM Application Relation Down Push	Pushes downstream relations between Applications (pushed by APM Application Push) to other business elements or to nodes. BusinessApplication CIs must have been pushed before this TQL query synchronization in the 'APM Application Push' TQL query synchronization. Mapping XML: `pushMappingAPMApplicationDownR.xml`

   d. Select the **Allow Deletion** option for each query.

      This allows deletion of synchronized data in APM when data in UCMDB are deleted. Otherwise requests created in APM as a result of synchronization remain even when their original data in UCMDB are deleted.

      > **Note:** For scheduling configuration, see "Schedule Data Push Jobs" on the next page.

   e. Click **OK.**

   f. Save the integration point.

5. Run the job manually to see if the integration job works properly:

   a. To push all the relevant data for the job, click the All Data Synchronization [icon] button (or **Full Synchronization** [icon]).

   b. To push only the changes in the data since the job last executed, click the Changes Synchronization [icon] button (or **Delta Synchronization** [icon]).

6. The job is in **Running** status. Wait for the job to complete; click the **Refresh** [icon] button multiple times as needed until the job is completed.

   You can also look at the Query tab as that gives a progress bar against the query it is running.

7. When the job is completed, the job status becomes one of the following depending on the results:

   ○ Succeeded
   ○ Passed with failures
   ○ Failed

8. Click the **Statistics** tab to view the results; if any errors occur, click the **Query Status** tab and **Job Errors** tab for more information. For more information about errors, see "Troubleshooting and Limitations" on page 391.

Schedule Data Push Jobs

UCMDB allows you to schedule job executions directly from a data push job.

1. Log in to UCMDB as an administrator.

2. Navigate to **Data Flow Management > Integration Studio.**

   UCMDB displays a list of existing integration points.

3. Select the integration point you created for the APM - UCMDB integration.

4. Select the APM Push job.

5. Click the **Edit Integration Job** button.

   The Edit Integration Job dialog opens.

   > **Note:** UCMDB allows you to define two different schedules for two types of data push: **Changes Synchronization** (or **Delta Synchronization**) and **All Data Synchronization** (or **Full Synchronization**). It is recommended to use the Changes Sync schedule to only synchronize changes and avoid synchronizing the entire set of data each time.

6. Define a schedule for Changes Synchronization or Delta Synchronization.

   a. Click on the **Changes Synchronization** tab (or **Delta Synchronization**).

   b. Select the **Scheduler enabled** option.

   c. Select the scheduling options you want to use.

7. Click the **All Data Synchronization** tab (or **Full Synchronization**) and select the scheduling options you want to use.

8. Click **OK**.

9. Save the integration point.

View UCMDB Data in APM

After a push job is successfully completed, you can search for and verify that the pushed CI/relationship data is in APM.

To view UCMDB data in APM,

1. Log on to PPM Center.

2. On the **Open** menu, click **Application Portfolio > Search Entities**.

   The Search Entities page opens.

3. In the **Entities** section, click one of the following entities:

   ○ Application
   ○ Location
   ○ Process
   ○ Server

   The Search: APM - *<Entity>* page opens.

   In this example, click **Application** and the Search: APM - Application page opens.

4. Click **Search.**

   The Search Results page displays request search results.

5. Click any **Application No** to view an APM - *<Entity>* request.

Customize the Integration

This section includes:

- "Overview" below

- "Customize an Existing Mapping" on the next page

- "Add a New Mapping to the Integration" on page 381

Overview

This section contains details about the architecture of the integration.

Data Flow Architecture

1. The Push Engine executes the TQL query.

2. For a differential flow, the data is compared to the last synchronized data, and only the changes are forwarded.

3. Data is converted into Composite CIs (instances of data according to the TQL Root elements).

4. Data is then pushed to the Push Adapter.

5. The Push Adapter loads the correct mapping for the specific TQL query.

6. All dynamic_mappings are executed and saved to maps, to allow usage in the next mapping stage.

   For more information, see **Developing Push Adapters** in the *Universal CMDB Developer Reference Guide*.

7. Data is sent to APM database via REST APIs from APM, where REST APIs converts data to APM compatible data.

Integration TQL Queries

A TQL query used for the integration must contain a root query node.

Any attribute used in the mapping flow of the Push Adapter must be marked in the selected layout of the query node. Each TQL query may only have one mapping.

For more information, see **Data Flow Management > Integration > Integration Studio > Integration Studio User Interface > Integration Jobs Pane**.

Customize an Existing Mapping

This example shows you how to add the NAME attribute to the integration including the TQL query and Push Adapter Mapping. It allows the integration to push the NAME attribute to Location in APM.

After completing the following steps, you may run the job with the customized mapping:

1. Add the NAME attribute to the APM Location Push TQL query layout.

   In this step we add the NAME attribute of the Location to the integration TQL query (APM Location

Push TQL) so that we can use the attribute and value in the mapping.

    a. Navigate to **Modeling > Modeling Studio > Resources** and select the **Queries Resource Type**.

    b. Navigate to **Query: Root > Integration > APM Push > APM Location Push**.

    c. Select **Root**, right-click and select **Query Node Properties**.

    d. Go to the Element Layout tab.

    e. Move the **Name** to the Specific Attributes box.

    f. Click **OK**.

    g. Save the Query.

2. Add the NAME Mapping to the pushMappingAPMLocation.xml push adapter mapping.

   In this step we take the value from the TQL result and remodel it to the APM Data Model.

    a. Navigate to **Data Flow Management > Adapter Management > Packages > APMPushAdapter > Configuration Files > pushMapingAPMLocation.xml.**

    b. Navigate to the **<target_ci_type name="fields">** XML tag.

    c. Below the tag, add the following XML tag to hold the value of the Description:

```
<target_mapping name="REQ.DESCRIPTION" datatype="STRING"
value="APMPushFunctions.subString(Root['name'],200)"/>
```

    where, `"REQ.DESCRIPTION"` is the request Name field token of Location.

    d. Click **OK**.

Add a New Mapping to the Integration

This example shows how to add a new TQL query and push-mapping to the integration. It also shows how to push Locations from UCMDB to APM. It consists of the following steps:

- Step 1: Create a TQL Query

    a. Navigate to **Modeling > Modeling Studio > New > Query**.

    b. From the **CI Types** tab, add a **Location** to the query.

    c. Right-click the **Location Query Node** and select **Query Node Properties**.

    d. Rename the **Element Name** to **Root**.

e. Navigate to the **Element Layout** tab.

f. Select **Select attributes for layout**.

g. In the **Attributes condition** drop down, select **Specific Attributes**, and add the **Name** attribute

h. Click **OK**.

i. Save the query to **Root > Integration > APM Push > APM Location Push**.

- Step 2: Create a Push-Mapping

  a. Navigate to **Data Flow Management > Adapter Management > APMPushAdapter**.

  b. Click the **Create new resource** ✳ button and select **New Configuration File**.

  c. Type the following Name: `APMPushAdapter/mappings/pushMappingAPMLocation.xml`.

  d. Select the **APMPushAdapter** package.

  e. Click **OK**.

  f. Copy the following into the newly created XML file:

```xml
<integration xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
 <info>
    <source name="UCMDB" versions="10.0" vendor="HP"/>
    <target name="APM" versions="2.0" vendor="HP"/>
  </info>
  <import>
    <scriptFile path="mappings.scripts.APMPushFunctions"/>
  </import>
  <targetcis>
    <source_instance_type query-name="APM Location Push" root-element-
name="Root" >
      <target_ci_type name="request">
        <target_mapping name="uuid" datatype="STRING" value="Root
['global_id']"/>
        <target_mapping name="requestType" datatype="STRING" value="'APM
- Location'"/>
        <target_mapping name="description" datatype="STRING" value="Root
['name']"/>
        <target_ci_type name="fields">
          <target_mapping name="REQ.DESCRIPTION" datatype="STRING"
value="APMPushFunctions.subString(Root['name'],200)"/>
        </target_ci_type>
      </target_ci_type>
    </source_instance_type>
  </targetcis
```

```
</integration>
```

In the following line:

```
<target_mapping name="requestType" datatype="STRING" value="'APM -
Location'"/>
```

The value is the request type name in APM. In this example, it is `APM - Location`.

g. Click **OK**.

- Step 3: Create a Job with the New TQL Query

  a. Navigate to **Data Flow Management > Integration Studio**.

  b. Create an Integration Point with APM.

  c. In the **Integration Jobs** tab, click the **New Integration Job** ✳ button .

  d. Insert a job name in the **Name** field.

  e. Click the ➕ button, and choose the **APM Location Push query**.

  f. Click **OK**.

- Step 4: Run the Job

  a. Click on the job created in "Step 3: Create a Job with the New TQL Query " above.

  b. Click the All Data Synchronization 🔄 button (or **Full Synchronization** 🔄).

  c. Wait for the job to finish. You should click the **Refresh** 🔄 button to see progress.

  d. Make sure that the status is **Succeeded**.

- Step 5: View the Results

  a. Log on to PPM.

  b. On the **Open** menu,

    - Click **Application Portfolio > Search Entities**, then in the **Entities** section, click **Location**. Or,

    - Click **Search > Requests**, then from the **Request Type** drop-down list, select **APM - Location**.

  c. Click **Search**.

  The Search Results page displays request search results.

Developer References

This section includes the following:

- "Adapter" on the next page

- "Default Entity and Field Mappings between APM and UCMDB" on page 386

- "REST APIs Called in the Integration" on page 391

Adapter

This integration job uses the adapter called APM Push Adapter. It is displayed as APM Push in the Job Name.

Input CI Type

destination_config

Triggered CI Data

Name	Value
adapterId	${ADAPTER.adapter_id}
attributeValues	${SOURCE.attribute_values}
credentialsId	${SOURCE.credentials_id}
destinationId	${SOURCE.destination_id}

Adapter Parameters

Name	Value
credentialsId	
domain	itg
host	
port	80
probeName	
protocalType	http

Default Entity and Field Mappings between APM and UCMDB

The following sections describe out of the box mappings that are available with the APM Push Adapter for integration with APM and UCMDB.

The following table provides an overview of type mappings between APM entities and UCMDB CI Types:

APM Entity	PPM Center Request Type	UCMDB CI Type	Remarks
Application	APM - Application	BusinessApplication	For detailed mappings, see "Default Field Mappings between APM Application and UCMDB BusinessApplication" below.
Process	APM - Process	BusinessProcess	For detailed mappings, see "Default Field Mappings between APM Process and UCMDB BusinessProcess" on page 388.
Location	APM - Location	Location	For detailed field mappings, see "Default Field Mappings between APM Location and UCMDB Location" on page 389.
Server	APM - Server	Node	For detailed field mappings, see "Default Field Mappings between APM Server and UCMDB Node" on page 390.

Default Field Mappings between APM Application and UCMDB BusinessApplication

The following table describes the default field mappings that can be modified for the integration between the APM entity of **Application** and the UCMDB CI Type of **BusinessApplication**.

APM Field Name and Field Type	UCMDB CI Attribute and Field Type
Name	Name

APM Field Name and Field Type	UCMDB CI Attribute and Field Type
KNTA_PROJECT_NAME Text Field - 300	name string
Updated By [a]	(N/A)
Create On [b] CREATION_DATE Date	(N/A)
Purpose APM_APP_PURPOSE Text Field - 4000	Description description string
Business Criticality [c] APM_RATING_BUSINESS_CRIT Drop Down List	BusinessCriticality business_criticality integer
Created By[a] CREATED_BY Auto Complete List	(N/A)
Supported Processes APM_SUPPORTED_PROCESSES Auto Complete List	Name (of CI Type **BusinessProcess**) name (of CI Type **BusinessProcess**) string
Downstream Applications APM_DOWNSTREAM_APPS Auto Complete List	Name (of downstream CI Type **BusinessApplication**) name (of downstream CI Type **BusinessApplication**) string
Upstream Applications APM_UPSTREAM_APPS Auto Complete List	Name (of upstream CI Type **BusinessApplication**) name (of upstream CI Type **BusinessApplication**) string
Service Level Agreement APM_APP_SLA Text Field - 200	Name (of CI Type **ServiceLevelAgreement**) name (of CI Type **ServiceLevelAgreement**) string
Servers APM_SERVER_LIST Auto Complete List	Name (of CI Type **Node**) name (of CI Type **Node**) string
Database APM_DATABASE_LIST Text Field - 200	Name (of CI Type **Database**) name (of CI Type **Database**) string

a. The APM account you provided when creating the integration point (see "Create an Integration Point between APM and UCMDB" on page 371).

b. Time when the request is created for the first time in APM.

c. The following mapping rule is used for this mapping:

APM Field Name and Field Type	UCMDB CI Attribute and Field Type
<target_mapping name="REQD.APM_RATING_BUSINESS_CRIT" datatype="STRING" value="APMPushFunctions.getPropertyValue('bc', Root['business_ criticality'].toString() , '')"/> where the definition of 'bc' token is defined as follows in the **server.properties** field: bc.0=0 - Least critical bc.1=1 - Slightly critical bc.2=2 - Less than average bc.3=3 - More than average bc.4=4 - Critical bc.5=5 - Highly critical When synchronizing the **Business Criticality** field from UCMDB to APM, if the value is '1' in UCMDB, then the field value will be set to '1 - Slightly critical' in APM.	

Default Field Mappings between APM Process and UCMDB BusinessProcess

The following table describes the default field mappings that can be modified for the integration between the APM entity of Process and the UCMDB CI Type of BusinessProcess.

APM Field Name, Database ID, and Field Type	UCMDB CI Attribute, Database ID, and Field Type
Process Name DESCRIPTION Text Field - 200	Name name string
Parent Process APM_PARENT Auto Complete List	Name of parent **BusinessProcess** name of parent **BusinessProcess** string
Description APM_DESCRIPTION Text Area - 4000	Description description string
Created By [a] CREATED_BY Auto Complete List	(N/A)
Created On [b] CREATION_DATE Date	(N/A)
a. The APM account you provided when creating the integration point (see "Create an Integration	

APM Field Name, Database ID, and Field Type	UCMDB CI Attribute, Database ID, and Field Type
Point between APM and UCMDB" on page 371)	
b. Time when the request is created for the first time in APM.	

Default Field Mappings between APM Location and UCMDB Location

The following table describes the default field mappings that can be modified for the integration between the APM entity of Location and the UCMDB CI Type of Location.

APM Field Name, Database ID, and Field Type	UCMDB CI Attribute, Database ID, and Field Type
Location Name DESCRIPTION Text Field - 200	Name name string
Description APM_DESCRIPTION Text Field - 4000	Description description string
Address APM_LOC_ADDRESS Text Field - 200	StreetAddress+ExtendedStreetAddress street_address+extended_street_address string+string
Postal Code APM_LOC_ZIPCODE Text Field - 20	PostalCode postal_code string
Country APM_LOC_COUNTRY Text Field - 200	CountryOrArea country_or_area string
Region APM_LOC_REGION DDL	Region region string
Longitude APM_LONGITUDE Text Field - 40	Longitude longitude string
Latitude APM_LATITUDE	Latitude latitude

APM Field Name, Database ID, and Field Type	UCMDB CI Attribute, Database ID, and Field Type
Text Field - 40	string
City APM_LATITUDE Text Field - 200	~~City~~[a] ~~city~~ ~~string~~
State/Province APM_STATE Text Field - 200	~~State~~[a] ~~state~~ ~~string~~
a. The CI attribute is removed from UCMDB version 10.x, but exists in earlier versions of UCMDB. For UCMDB instances that upgraded from an earlier version to 10.x, this CI attribute exists but is read-only. If the CI attribute has a value, the value can be synchronized to APM, otherwise the APM field remains empty after you run the synchronization push job in UCMDB.	

Default Field Mappings between APM Server and UCMDB Node

The following table describes the default field mappings that can be modified for the integration between the APM entity of Server and the UCMDB CI Type of Node.

APM Field Name, Token, and Component Type	UCMDB CI Attribute Display Name, Name, and Type
Server Name DESCRIPTION Text Field - 200	Name name string
Description APM_DESCRIPTION Text Area - 4000	Description description string (value size: 1000)
IP Address APM_IP_ADDRESS Text Field, Max Length: 15	IP Address (of **IpAddress**) ip_address (of **IpAddress**) string
OS APM_OS Drop-down List	OsDescription os_description string
Running Software APM_RUNNING_SOFTWARE Text Area - 4000	ProductName:Name (of **RunningSoftware**) product_name:name (of **RunningSoftware**) product_name_enum:string
Location	Name (of **Location**)

APM Field Name, Token, and Component Type	UCMDB CI Attribute Display Name, Name, and Type
APM_LOCATION Auto Complete List	name (of **Location**) string

REST APIs Called in the Integration

The following Demand Management REST APIs are called in this integration to convert data from UCMDB into APM compatible data:

- Get a request

  For more information, see **Getting Details of a Request** section of the *RESTful Web Services Guide* for PPM 9.40.

- Create a request

  For more information, see **Create/Update a Request** section of the *RESTful Web Services Guide* for PPM 9.40.

- Update a request

  For more information, see **Creating/Updating a Request** section of the *RESTful Web Services Guide* for PPM 9.40.

- Delete a request

  Added in version 9.22. For more information, see **Deleting a Request** section of the *RESTful Web Services Guide* for PPM 9.40.

Troubleshooting and Limitations

This section includes the following:

Limitations

- The Data Flow Probe or Integration Service must be installed on a Windows OS.

- For requests created in APM from CIs pushed from UCMDB, any changes made in APM are overwritten when you run the data push job in UCMDB.

- The APM Application request form holds a single value for Location, therefore it is designed to push only one value for Location of the Application from the UCMDB BusinessApplication.

  In the definition of Location of the Application (see the `pushMappingAPMApplicationR.xml` file), the Location of the Server is used for the Location of the Application is used. For an Application that contains multiple servers, select one of the servers and then you can get its Location.

- For value mappings between UCMDB and APM, certain mapping rules are followed.

  For example, when synchronizing the Business Criticality field of Application, the following mapping rule is used for the mapping:

  ```
  <target_mapping name="REQD.APM_RATING_BUSINESS_CRIT" datatype="STRING"
  value="APMPushFunctions.getPropertyValue('bc', Root['business_
  criticality'].toString() , '')"/>
  ```

  where the definition of 'bc' token is defined as follows in the
  <APMPushAdapter.zip>/mappings/scripts/**server.properties** file:

  ```
  bc.0=0 - Least critical
  bc.1=1 - Slightly critical
  bc.2=2 - Less than average
  bc.3=3 - More than average
  bc.4=4 - Critical
  bc.5=5 - Highly critical
  ```

  When synchronizing the **Business Criticality** field from UCMDB to APM, if the value is '1' in UCMDB, then the field value will be set to '1 - `Slightly critical`' in APM.

  If you need to use this value mapping for other fields from UCMDB to APM, make sure you customize the mapping by following the example above.

- This integration does not support synchronizing values in languages that are not supported by UCMDB. For example, Simplified Chinese.

- The "**;**" character is not supported. If a UCMDB CI name contains a "**;**" character, it would be treated as a separator and two entries would show up in APM after synchronization.

Troubleshooting Problems

- **Problem:** Some UCMDB CIs include characters that are not supported in APM Entities. For example, semicolon (;).

  Solution: The suggested solution is to modify the content synchronized from UCMDB to APM. You can use the Replace function to replace the unsupported characters for this field mapping in the XML mapping file. However, note that this may cause inconsistent content between UCMDB and APM.

  An example, in the `pushMappingAPMApplication.xml` file, the semicolon character (;) is replaced with a space:

  ```
  <target_mapping name="REQ.KNTA_PROJECT_NAME" datatype="STRING"
  value="APMPushFunctions.stringReplace(Root['name'], ';', ' ')"/>
  ```

- **Problem:** For some fields, the field value lengths between APM and UCMDB are different, therefore you may need to customize the field mapping. You can follow the example below:

  Example

  Use a substring as illustrated below to limit the field length to 200 characters:

  ```
  <target_mapping name="REQ.DESCRIPTION" datatype="STRING"
  value="APMPushFunctions.subString(APMPushFunctions.stringReplace(Root
  ['name'],';',' '),200)"/>
  ```

Logs

The push adapter framework uses a different logging system for the normal fcmdb.adapters.*.log files.

To change the level of the log files to debug, edit the following file:

- On the Data Flow Probe machine:

  ```
  ..\DataFlowProbe\conf\log\fcmdb.push.properties
  ```

- If using the integration service, on the UCMDB server:

  ```
  ..\UCMDB Server\Integrations\conf\log\fcmdb.push.properties
  ```

Change the log level to DEBUG:

```
loglevel=DEBUG
```

The integration generates `fcmdb.push.*` logs in the following folder:

- On the Data Flow Probe machine:

  `..\DataFlowProbe\runtime\log\`

- If using the integration service, on the UCMDB server:

  `..\UCMDB Server\Integrations\runtime\log\`

Send documentation feedback

If you have comments about this document, you can contact the documentation team by email. If an email client is configured on this system, click the link above and an email window opens with the following information in the subject line:

Feedback on Solution Integrations Guide (Project and Portfolio Management Center 9.40)

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to your_IE_team_PDL@hpe.com.

We appreciate your feedback!