HP ALM

Software Version: 11.52

Administrator Guide

Document Release Date: May 2013

Software Release Date: May 2013

Legal Notices

Warranty

The only warranties for HP 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. HP 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 HP 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

© Copyright 2002 - 2013 Hewlett-Packard Development Company, L.P.

Trademark Notices

Adobe® is a trademark of Adobe Systems Incorporated.

Intel® is a trademark of Intel Corporation in the U.S. and other countries.

Java is a registered trademark of Oracle and/or its affiliates.

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

Oracle® is a registered trademark of Oracle and/or its affiliates.

UNIX® is a registered trademark of The Open Group.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to:

http://h20230.www2.hp.com/selfsolve/manuals

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

Or click the **New users - please register** link 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 HP sales representative for details.

Support

Visit the HP Software Support Online web site at:

http://www.hp.com/go/hpsoftwaresupport

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support web site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

Contents

Administrator Guide 1

Contents 5

Welcome to this Guide 18

How This Guide is Organized 18

ALM Help 19

ALM Help Guides 19

Additional Online Resources 22

ALM Extension Guides 23

Part 1: Site Administration 24

Chapter 1: Site Administration at a Glance 26

Starting Site Administration 27

Understanding Site Administration 28

Defining Site Administrators 31

Chapter 2: Creating Projects 34

About Creating Projects 35

Understanding the Project Structure 35

Creating Domains 36

Creating Projects 37

Copying Projects 44

Importing Projects 50

Creating Template Projects 54

Creating a Template Project 55

Creating a Template from an Existing Template 59

Creating a Template from an Existing Project 64

Importing a Template Project 68

Linking a Template to Projects 72

Updating Project Details 73

Assigning Users to Projects 77
Assigning Project Administrators 78
Enabling Extensions for a Project 79
Chapter 3: Managing the Optimized Project Repository 82
Project Repository Cleanup 83
Browsing the Project Repository 83
Realigning Repositories 85
Chapter 4: Managing Projects 88
About Managing Projects 89
Querying Project Tables 89
Exporting Projects 90
Deactivating and Activating Projects 91
Enabling and Disabling Version Control for a Project 91
Pinging Projects 92
Renaming Projects 92
Converting Projects to Unicode 93
Removing Projects 95
Deleting Projects 96
Deleting Domains 96
Editing the Connection String 97
Restoring Access to Projects 97
Renaming the Defects Module for a Project 99
Limiting Records Displayed in Grids 100
Chapter 5: Upgrading Projects to a New Version 102
About Upgrading Projects 103
Understanding the Version/Patch Numbering Scheme 103
About Upgrading Projects for Major and Minor Versions 103
Verifying Domains and Projects 107
Verifying a Project 107
Verifying a Domain 108
Repairing Domains and Projects 110
Repairing a Project 110

Repairing a Domain ... 111
Upgrading Domains and Projects ... 113
Upgrading a Project ... 113
Upgrading a Domain ... 115
Defining an Exception File ... 116
Backing Up Projects ... 118
Restoring Projects ... 119
Restoring Projects from a Microsoft SQL Database Server ... 119
Restoring Projects from an Oracle Database Server ... 119
Restoring a Repository from the File System ... 120
Restoring LAB_PROJECT ... 120
Migrating the Repository ... 121
Repository Migration Status Window ... 122
Configure Migration Priority ... 125
LAB_PROJECT Post Upgrade Steps ... 125
About Upgrading Projects for Minor-Minor Versions ... 126
Setting Project Priorities for Upgrading Minor-Minor Versions ... 128
Chapter 6: Managing ALM Users ... 130
About Managing Users ... 131
Adding a New User ... 131
Importing Users from LDAP ... 132
Defining LDAP Settings for Importing Users ... 134
Handling User Name Conflicts ... 137
Updating User Details ... 138
Deactivating and Activating Users ... 139
Creating and Changing Passwords ... 139
Enabling LDAP Authentication for Users ... 140
Assigning Projects to Users ... 141
Exporting User Data ... 143
Deleting Users ... 144
Chapter 7: Managing User Connections and Licenses ... 146
About Managing User Connections and Licenses ... 147

Monitoring User Connections 147
Sending Messages to Connected Users 148
Managing Licenses 149
Modifying Licenses 149
Assigning Licenses to Domains and Projects 151
Assigning Licenses to Users 152
PPU Licenses History 154
Chapter 8: Configuring Servers and Parameters 156
About Configuring Servers and Parameters 157
Configuring Server Information 157
Defining New Database Servers 159
Modifying Database Server Properties 161
Configuring Text Search 163
Enabling Text Search on Database User Schemas 163
Enabling Text Search in ALM 163
Selecting a Text Search Language for a Project 164
Defining Searchable Fields 165
Setting ALM Configuration Parameters 167
Default ALM Parameters 167
Optional ALM Parameters 171
Setting ALM Parameters 196
Setting the ALM Mail Protocol 196
Chapter 9: Analyzing Site Usage 198
About Analyzing Site Usage 199
Monitoring Site Usage 199
Filtering Site Usage 200
Exporting Site Analysis Data to a File 201
Customizing the Site Analysis Line Chart Graph 201
Chapter 10: Scheduling Calculations for Project Planning and Tracking (PPT) 204
About Scheduling Calculations for PPT 205
Scheduling Calculations for a Site 205
Enabling or Disabling Automatic Calculations for a Project 205

Launching Calculations for a Project Manually 206
Project Planning and Tracking Tab 206
Chapter 11: ALM Assessment 210
About ALM Assessment 211
Collecting and Assessing ALM Data 216
Chapter 12: QC Sense 218
About QC Sense 219
QC Sense Configuration 219
QC Sense Monitors 220
Configuring QC Sense 221
QC Sense Server Configuration Window 221
Connection String Builder Dialog Box 224
Generating and Viewing QC Sense Reports 226
QC Sense Schema 227
PERF_CLIENT_OPERATIONS 227
PERF_CLIENT_METHODS_CALLS 229
PERF_CLIENT_REQUESTS 230
PERF_SERVER_THREADS 232
PERF_SERVER_SQLS 233
PERF_SERVER_GENERAL_MEASURES 234
PERF_SERVER_THREAD_TYPES 235
Part 2: Project Customization 238
Chapter 13: Project Customization at a Glance 240
Starting Project Customization 241
Understanding the Project Customization Window 243
Saving Customization Changes 245
Chapter 14: Managing Users in a Project 248
About Managing Users in a Project 249
Adding a User to a Project 249
Assigning Users to a User Group 250
Removing a User from a Project 251
Chapter 15: Managing User Groups and Permissions 252

About Managing User Groups and Permissions 253
Adding User Groups 253
Assigning Users to Groups 254
Setting User Group Permissions 255
Setting Transition Rules 257
Owning ALM Objects 258
Hiding Data for a User Group 260
Renaming User Groups 261
Deleting User Groups 261
Understanding Permission Settings 262
Administration Permission Levels 263
Build Verification Suite Permission Levels 266
Business Models Permission Levels 267
Business Process Testing Permission Levels 269
Dashboard Permission Levels 273
Defects Permission Levels 276
Lab Resources Permission Levels 277
Lab Settings Permission Levels 278
Libraries Permission Levels 278
Releases Permission Levels 280
Requirements Permission Levels 283
Resources Permission Levels 286
Test Lab Permission Levels 287
Test Plan Permission Levels 292
Timeslots Permission Levels 295
Customizing Module Access for User Groups 296
Chapter 16: Customizing ALM Projects 298
About Customizing ALM Projects 299
Customizing Project Entities 299
Adding User-Defined Fields 302
Modifying System and User-Defined Fields 303
Deleting User-Defined Fields 304

Defining Input Masks 304
Customizing Project Requirement Types 306
Creating Requirement Types 307
Customizing Requirement Types 307
Renaming Requirement Types 309
Deleting Requirement Types 310
Customizing Project Lists 310
Creating Lists 311
Renaming Lists, Items, or Sub-Items 311
Deleting Lists, Items, or Sub-Items 312
Chapter 17: Configuring Automail 314
About Setting Automail 315
Designating Automail Fields and Conditions 315
Customizing the Subject of Defect Mail 317
Chapter 18: Customizing Risk-Based Quality Management 318
About Customizing Risk-Based Quality Management 319
Customizing Risk-Based Quality Management Criteria 319
Customizing Criteria and Values 320
Customizing Weight Boundaries 321
Customizing Risk Calculations 324
Customizing Risk-Based Quality Management Constants 324
Chapter 19: Activating Alert Rules 328
About Activating Alert Rules 329
Setting Alert Rules 330
Chapter 20: Cross Project Customization 332
About Cross Project Customization 332
Cross Project Customization Overview 333
Updating Linked Projects 334
Updating Linked Project Details 334
Verifying Cross Project Customization 336
Applying Template Customization to Linked Projects 337
The Cross Project Customization Report 339

Updating Linked Template Details ... 340
Chapter 21: Customizing Project Planning and Tracking KPIs ... 342
About Customizing PPT KPIs ... 343
Project Planning and Tracking Page ... 343
Project Planning and Tracking - General Tab ... 344
Configure Transitions Dialog Box ... 346
Project Planning and Tracking - KPI Analysis Tab ... 347
Chapter 22: Project Report Templates ... 350
About Project Report Templates ... 351
Managing Project Report Templates ... 351
Creating a New Report Template ... 352
Editing a Report Template ... 353
Duplicating a Report Template ... 353
Deleting a Report Template ... 354
Designing Report Templates ... 354
About Designing Report Templates ... 354
Designing Document Templates ... 355
Designing Style Templates ... 356
Designing History Templates ... 356
Designing Section Templates ... 357
Guidelines for Creating Full-Page and Tabular Templates ... 359
Full-page templates ... 359
Tabular templates ... 360
Template Creator Tab ... 360
Chapter 23: Business Views ... 362
Business Views Overview ... 363
Creating and Managing Business Views ... 363
Working with DQL ... 365
Business Views User Interface ... 387
Business Views Page ... 388
Link Properties Dialog Box ... 393
Add Related Entity Dialog Box ... 394

Chapter 24: Configuring Business Process Testing 396
About Configuring Business Process Testing 397
Business Process Test Page 397
Chapter 25: Configuring Sprinter 398
About Configuring Sprinter 399
Sprinter Page 399
Chapter 26: Generating Workflow Scripts 402
About Generating Workflow Scripts 403
Customizing Defects Module Field Lists 404
Customizing Defects Module Dialog Boxes 406
Chapter 27: Managing Analysis Menus 410
Part 3: Workflow Customization 412
Chapter 28: Workflow Customization at a Glance 414
Chapter 29: Working with the Workflow Script Editor 416
About Working with the Workflow Script Editor 417
The Script Editor 417
Understanding the Script Editor Commands 419
Creating a Workflow Script 420
Adding a Button to a Toolbar 422
Setting the Properties of the Script Editor 424
Chapter 30: Workflow Event Reference 428
About ALM Events 429
Naming Conventions for ALM Event Procedures 429
Entity 430
Event 431
Reference for ALM Events 431
ActionCanExecute 433
AddComponentToTest 434
AfterPost 434
Attachment_CanDelete 435
Attachment_CanOpen 436
Attachment_CanPost 436

Attachment_New 436
CanAddComponentsToTest 437
CanAddFlowsToTest 437
CanAddTests 437
CanCustomize 437
CanDelete 438
CanDeleteGroupsFromTest 441
CanLogin 441
CanLogout 441
CanPost 441
CanReImportModels 443
CanRemoveComponentsFromTest 443
CanRemoveFlowsFromTest 443
CanRemoveTests 443
DefaultRes 444
DialogBox 444
EnterModule 444
ExitModule 445
FieldCanChange 445
FieldChange 447
GetDetailsPageName 449
GetNewBugPageName 449
GetNewReqPageName 450
GetReqDetailsPageName 450
MoveTo 451
MoveToComponentFolder 453
MoveToFolder 453
MoveToSubject 453
New 454
RemoveComponentFromTest 455
RunTests 456
RunTests_Sprinter 456

RunTestSet ... 456
RunTestsManually ... 457
Chapter 31: Workflow Object and Property Reference ... 458
About ALM Objects and Properties ... 459
Actions Object ... 460
Action Object ... 460
Fields Objects ... 461
Field Object ... 463
Lists Object ... 464
TDConnection Object ... 464
User Object ... 465
ALM Properties ... 465
ActiveModule Property ... 466
ActiveDialogName Property ... 466
Chapter 32: Workflow Examples and Best Practices ... 468
About the Workflow Examples ... 469
Best Practices for Writing Workflow Scripts ... 469
Checking Value Types Before Use ... 470
Recommendations ... 470
Anticipating Full Evaluation of Logical Expressions ... 471
Recommendations ... 471
Defining Default Behavior for Select Case and If-Then-Else Statements ... 472
Recommendations ... 472
Setting Return Values in Functions ... 473
Recommendations ... 473
Making Sure that Entity Properties Are Set Before an Entity Comes into Focus ... 473
Recommendations ... 473
Check if a Dialog Box is Open ... 474
Recommendations ... 474
Avoid Defining Duplicate Subroutines ... 475
Recommendations ... 475
Example: Customizing a Defects Module Dialog Box ... 476

SetFieldApp ... 476
FieldCust_AddDefect ... 477
Example: Changing Tab Names ... 478
Example: Adding a Template to a Memo Field ... 479
Example: Changing One Field Based on Another Field ... 480
Example: Changing a Field Based on the User Group ... 480
Example: Object Validation ... 481
Example: Field Validation ... 481
Example: Presenting a Dynamic Field List ... 482
Example: Changing Field Properties when a Field Changes ... 483
Example: Controlling User Permissions ... 484
Example: Adding Button Functionality ... 484
Example: Error Handling ... 485
Example: Obtaining Session Properties ... 486
Example: Sending Mail ... 486
Sending Mail when a Defect is Submitted ... 486
Sending Mail when a Test Plan Module Field Value Changes ... 487
Example: Storing the Last Values Entered ... 487
Example: Copying Field Values to Another Object ... 489
Part 4: Appendix ... 492
Chapter A: Upgrade Preparation Troubleshooting ... 494
Overview ... 495
Quick Warning Reference ... 495
General Issues ... 495
Schema Issues ... 496
Data Issues ... 500
General Validation ... 501
Supported Database Version ... 501
Valid Database User Schema Name ... 501
Mixed Table Ownership ... 501
Repository over Database Feature ... 502
Version Control Validation ... 502

Database Permissions 502
Text Search Configuration 503
Schema Validation 504
Tables 505
Columns 506
Indexes and Constraints 508
Triggers 511
Sequences 511
Internal Quality Center Changes 512
Data Validation 514
Duplicate Values 515
Duplicate IDs 515
Tree Inconsistencies 516
Views 516
Orphaned Entities 517
Missing Entities 517
Missing Lists and/or List Values 517
Encrypted Values 518
Changing the Database User Schema 519
Missing Database Objects 519
Missing List Warning 520
Sequences Warning 520
Changed Database Objects 520
Extra Database Objects 521

Welcome to this Guide

Welcome to HP Application Lifecycle Management (ALM). ALM empowers IT to manage the core application lifecycle, from requirements through deployment, granting application teams the crucial visibility and collaboration needed for predictable, repeatable, and adaptable delivery of modern applications.

Throughout the application lifecycle management process, ALM projects are accessed by many users—including developers, testers, business analysts, and quality assurance managers. To protect, maintain, and control information in a project, users are assigned to groups with different access privileges. Only an ALM project administrator (belonging to the TDAdmin user group) has full privileges in an ALM project.

As an ALM site administrator, you use Site Administration to create and maintain domains and projects; manage users, connections, and licenses; define database servers; and modify configurations.

As an ALM project administrator, you use Project Customization to customize project entities and lists, set up user groups and permissions, configure mail, set alert rules, and configure the workflow in the ALM modules. You use Cross Project Customization to standardize customization across projects in your organization.

ALM is shipped without any passwords defined. To protect your data from unauthorized access, set your password early in the ALM process.

How This Guide is Organized

The HP Application Lifecycle Management Administrator Guide contains the following parts:

Part	Description
"Site Administration" on page 24	Describes how the site administrator uses Site Administration to manage ALM projects. This includes maintaining projects, users, connections, licenses, servers, configuration parameters, and site analysis.
"Project Customization" on page 238	Describes how the project administrator uses the Project Customization window to control access to a project by defining the project users and their privileges. It also describes how to customize a project to meet the specific needs of the project users.
"Workflow Customization" on page 412	Describes how to create workflow scripts to customize the ALM user interface and to control the actions that users can perform.
"Appendix" on page 492	The Upgrade Preparation Troubleshooting appendix describes the errors detected while verifying and repairing your projects, and provides information on how to fix these errors before upgrade.

ALM Help

The ALM Help is an online help system that describes how to use ALM. You can access the ALM Help in the following ways:

- Click **Documentation Library** in the ALM Help menu to open the ALM Help home page. The home page provides quick links to the main help topics.
- Click **Help on this page** in the Help menu to open the ALM Help to the topic that describes the current page.

ALM Help Guides

The ALM Help consists of the following guides and references, available online, in PDF format, or both. PDFs can be read and printed using Adobe Reader, which can be downloaded from the Adobe Web site (http://www.adobe.com).

Reference	Description
Using the ALM Help	Explains how to use the Help and how it is organized.
What's New?	Describes the newest features in the latest version of ALM.
Movies	Short movies that demonstrate the main product features. To access, select **Help > Movies**.
Readme	Provides last-minute news and information about ALM.

HP Application Lifecycle Management (ALM) Guides	Description
HP ALM User Guide	Explains how to use ALM to organize and execute all phases of the application lifecycle management process. It describes how to specify releases, define requirements, plan tests, run tests, and track defects.
HP ALM Administrator Guide	Explains how to create and maintain projects using Site Administration, and how to customize projects using Project Customization.
HP ALM Lab Management Guide	Explains how to use Lab Management to manage lab resources used for functional and performance testing on remote hosts.
HP ALM Lab Management Troubleshooting Guide	Provides information for troubleshooting problems while working with HP ALM Lab Management.

HP Application Lifecycle Management (ALM) Guides	Description
HP ALM Tutorial	A self-paced guide teaching you how to use ALM to manage the application lifecycle management process.
HP ALM Installation and Upgrade Guide	Describes the installation and configuration processes for setting up ALM Server, and the project upgrade process.
HP ALM Business Views Microsoft Excel Add-in User Guide	Explains how to install and use the Business Views Microsoft Excel Add-in to create and configure business view Excel reports.
Business Process Testing User Guide	Explains how to use Business Process Testing to create business process tests.

HP ALM Performance Center Guides	Description
HP ALM Performance Center Quick Start	A self-paced guide giving the Performance Center user a high level overview of creating and running performance tests.
HP ALM Performance Center Guide	Explains to the Performance Center user how to create, schedule, run, and monitor performance tests. Explains to the Performance Center administrator how to configure and manage Performance Center projects.
HP ALM Performance Center Installation Guide	Describes the installation processes for setting up Performance Center Servers, Performance Center Hosts and other Performance Center components.
HP ALM Performance Center Troubleshooting Guide	Provides information for troubleshooting problems while working with HP ALM Performance Center.

HP ALM Best Practices Guides	Description
HP ALM Agile Testing Best Practices Guide	Provides best practices for implementing agile testing principles.
HP ALM Business Process Models Best Practices Guide	Provides best practices for working with the Business Models module.

HP ALM Best Practices Guides	Description
HP ALM Database Best Practices Guide	Provides best practices for deploying ALM on database servers.
HP ALM Entities Sharing Best Practices Guide	Provides best practices for sharing entities.
HP ALM Project Planning and Tracking Best Practices Guide	Provides best practices for managing and tracking releases.
HP ALM Project Topology Best Practices Guide	Provides best practices for structuring projects.
HP ALM Upgrade Best Practices Guide	Provides methodologies for preparing and planning your ALM upgrade.
HP ALM Versioning and Baselining Best Practices Guide	Provides best practices for implementing version control and for creating baselines.
HP ALM Workflow Best Practices Guide	Provides best practices for implementing workflows.

HP ALM Performance Center Best Practices Guides	Description
HP Performance Centers of Excellence Best Practices	Provides best practices for successfully building and operating Performance Centers of Excellence.
HP Performance Monitoring Best Practices	Provides best practices for monitoring the performance of applications under test.

HP ALM API References	Description
HP ALM Project Database Reference	Provides a complete online reference for the project database tables and fields.
HP ALM Open Test Architecture API Reference	Provides a complete online reference for the ALM COM-based API. You can use the ALM open test architecture to integrate your own configuration management, defect tracking, and home-grown testing tools with an ALM project.
HP ALM Site Administration API Reference	Provides a complete online reference for the Site Administration COM-based API. You can use the Site Administration API to enable your application to organize, manage, and maintain ALM users, projects, domains, connections, and site configuration parameters.

HP ALM API References	Description
HP ALM REST API Reference (Technology Preview)	Provides an online reference for the ALM REST-based API. You can use the REST API to access and work with ALM data.
HP ALM COM Custom Test Type Developer Guide	Provides a complete online guide for creating your own testing tool and integrating it into the ALM environment using native COM development tools.
HP ALM .NET Custom Test Type Developer Guide	Provides a complete online guide for creating your own testing tool and integrating it into the ALM environment using a combination of DCOM and .NET classes.

Additional Online Resources

The following additional online resources are available from the Help menu:

Resource	Description
Troubleshooting & Knowledge Base	Opens the Troubleshooting page on the HP Software Support Web site where you can search the Self-solve knowledge base. Choose **Help > Troubleshooting & Knowledge Base**. The URL for this Web site is http://h20230.www2.hp.com/troubleshooting.jsp.
HP Software Support	Opens the HP Software Support Web site. This site enables you to browse the Self-solve knowledge base. You can also post to and search user discussion forums, submit support requests, download patches and updated documentation, and more. Choose **Help > HP Software Support**. The URL for this Web site is www.hp.com/go/hpsoftwaresupport. • Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. • To find more information about access levels, go to: http://h20230.www2.hp.com/new_access_levels.jsp • To register for an HP Passport user ID, go to: http://h20229.www2.hp.com/passport-registration.html
HP Software Web site	Opens the HP Software Web site. This site provides you with the most up-to-date information on HP Software products. This includes new software releases, seminars and trade shows, customer support, and more. Choose **Help > HP Software Web site**. The URL for this Web site is www.hp.com/go/software.

Resource	Description
Add-ins	Opens the Add-ins page, which offers integration and synchronization solutions with HP and third-party tools.
Tools	Opens the ALM Tools add-ins page, which offers integration and synchronization solutions with HP and third-party tools that are installed with ALM on the ALM Server.

ALM Extension Guides

Extensions provide added functionality to HP ALM. If you have a license for an ALM extension, you can utilize the added functionality by enabling the extension on a per project basis. For more details on enabling extensions, see "Enabling Extensions for a Project" on page 79.

To view the list of extensions available with ALM 11.52, or to download documentation for ALM extensions, visit the HP ALM Add-ins page, available from **Help > Add-ins**.

Part 1: Site Administration

Chapter 1: Site Administration at a Glance

Using HP Application Lifecycle Management (ALM) Site Administration, you create and maintain projects, users, servers, site connections, license usage, and parameters. You can also define site administrators and change site administrator passwords.

This chapter includes:

Starting Site Administration 27

Understanding Site Administration 28

Defining Site Administrators 31

Starting Site Administration

Using Site Administration, you create and maintain your ALM projects.

To start Site Administration:

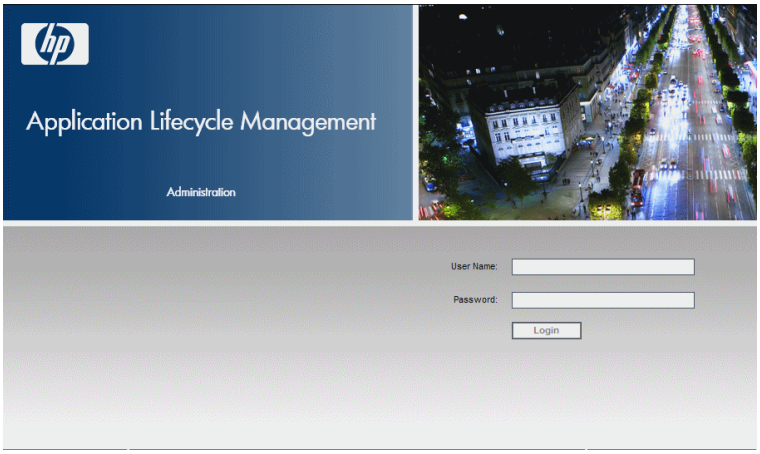
1. Select one of the following:

   - Open your Web browser and type: `http://<ALM server name>[<:port number>]/qcbin`. The HP Application Lifecycle Management Options window opens. Click the **Site Administration** link.
   - Alternatively, open your Web browser and type your Site Administration URL: `http://<ALM server name>[<:port number>]/qcbin/SiteAdmin.jsp`.

   The first time you start Site Administration, files are downloaded to your workstation. ALM then carries out a version check on the client files installed on your workstation. If there is a newer version on the server, updated files are downloaded to your workstation.

   Note: If you run ALM over a virtual environment, such as Citrix or VMware, only the system administrator can install a new version.

   After the ALM version has been checked and updated if necessary, the HP Application Lifecycle Management Site Administration Login window opens.



2. In the **User Name** box, type the name of a user who is defined as a site administrator. The first time you log in to Site Administration, you must use the site administrator name that you specified during the installation of ALM. After you log in to Site Administration, you can define additional site administrators. For more information, see "Defining Site Administrators" on page 31.

3. In the **Password** box, type your site administrator password. The first time you log in to Site Administration, you must use the site administrator password that you specified during the installation of ALM.

   To define or change the site administrator password, see "Creating and Changing Passwords"

on page 139.

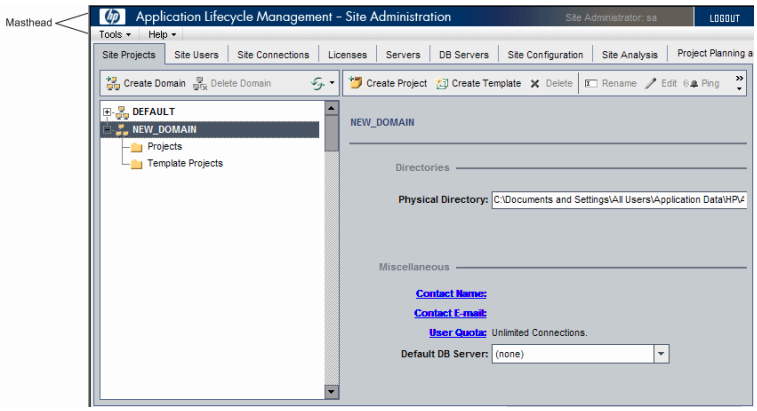
4. Click **Login**. Site Administration opens.

Understanding Site Administration

As an ALM site administrator, you create and maintain projects, users, and servers using Site Administration.

ALM Editions: Some functionality in Site Administration is unavailable for some editions. This includes the following:

- ALM template projects are not available for Quality Center Enterprise Edition.
- Functionality related to project planning and tracking is available for ALM Edition only.



This section includes:

- "Masthead" below
- "Tools Menu" on next page
- "Site Administration Tabs" on page 30

Masthead

The masthead includes options you can use throughout Site Administration.

Option	Description
Tools	Includes commands that you can run from any location in Site Administration. For details, see "Tools Menu" on next page.

Option	Description
Help	Enables you to open the HP Application Lifecycle Management Help and other additional online resources. To display version information for each ALM client component, select **Help > About HP Application Lifecycle Management Software**. **Tip:** To customize the **Help** menu, refer to the *HP Application Lifecycle Management Installation and Upgrade Guide*.
Site Administrator: <user>	The current Site Administrator user.
LOGOUT	**Logout.** Logs you out of Site Administration.

Tools Menu

The **Tools** menu, located on the masthead, contains the following options:

Option	Description
Collect Information	Creates the **ALM_CollectedInfo_<number>.html** file. This file contains diagnostic information about the ALM system. It is useful for when you contact ALM support. The **ALM_CollectedInfo_<number>.html** file is stored in the temp folder on the ALM server machine. To determine the folder location, open the Site Administration log file and locate the **java.io.tmpdir** field.
Repository Migration Status	Displays the status of project migration to the optimized project repository. For more information, see "Migrating the Repository" on page 121.
Set Project Upgrade Priorities	Enables you to set priorities before upgrading projects to new minor-minor versions. For information, see "About Upgrading Projects for Minor-Minor Versions" on page 126.
Update Test Types	Updates custom test type definitions in active projects. This is required after registering custom test types with ALM. If your site contains many active projects, this may take some time. For more information, refer to the *HP ALM Custom Test Types Guide*. **Note:** When you activate a project, custom test type definitions are updated automatically.
ALM Assessment	Collects environment details to assess the current ALM deployment and recommend usage improvements. For more information, see "ALM Assessment" on page 210.

Option	Description
QC Sense	Includes the following options for working with QC Sense, the internal ALM monitoring tool: • **Reports.** Enables you to generate reports based on collected data. • **Configuration.** Enables you to configure QC Sense monitors to define the scope of data collected. For more information, see "QC Sense" on page 218.

Site Administration Tabs

Site Administration contains the following tabs:

Tab	Description
Site Projects	Manage your ALM projects and templates. This includes adding new domains and projects, enabling extensions for projects, querying project data, restoring projects, renaming projects, and activating or deactivating projects. For more information, see "Managing Projects" on page 88. You can also upgrade projects from a previous Quality Center version to the current ALM version. For more information, see "Upgrading Projects to a New Version" on page 102.
Lab Management	Manage the LAB_PROJECT details and define Lab Management administrators. For more information, refer to the *HP ALM Lab Management Guide*.
Site Users	Add new users and define user properties, including changing passwords. For more information, see "Managing ALM Users" on page 130. You can also define site administrators. For more information, see "Defining Site Administrators" on next page.
Site Connections	Monitor the users currently connected to an ALM server. For more information, see "Managing User Connections and Licenses" on page 146.
Licenses	Monitor the total number of ALM licenses in use and to modify the license key. For more information, see "Managing User Connections and Licenses" on page 146.
Servers	Modify ALM server information, such as the log file. For more information, see "Configuring Servers and Parameters" on page 156.
DB Servers	Manage your database servers. This includes adding a new database server, editing a database's connection string, and changing a database's default administrator user name and password. For more information, see "Configuring Servers and Parameters" on page 156.

Tab	Description
Site Configuration	Modify ALM configuration parameters, such as the mail protocol. For more information, see "Configuring Servers and Parameters" on page 156.
Site Analysis	Monitor the number of licensed ALM users connected to your projects at specific points over a period of time. For more information, see "Analyzing Site Usage" on page 198.
Project Planning and Tracking	Schedule project planning and tracking calculations for an ALM site. For more information, see "Scheduling Calculations for Project Planning and Tracking (PPT)" on page 204.

Defining Site Administrators

You can define ALM users as site administrators.

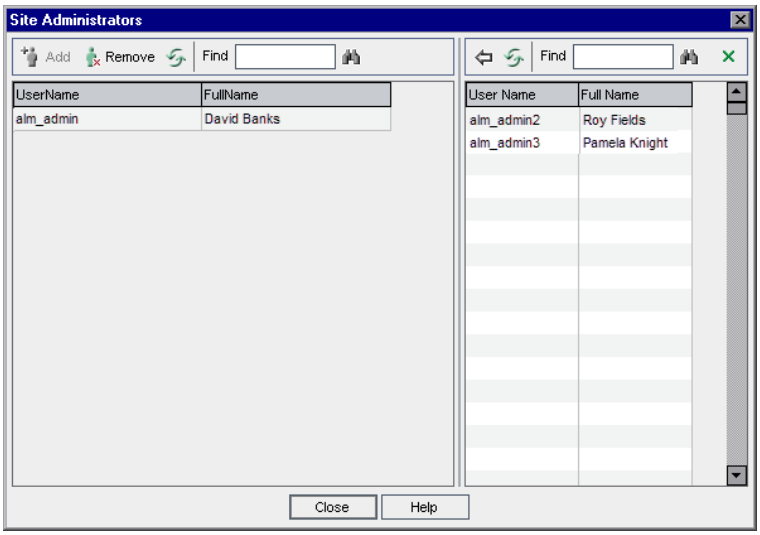
To secure the information in Site Administration, ensure that each user you add as a site administrator has a password defined. For more information, see "Creating and Changing Passwords" on page 139.

To define site administrators:

1. In Site Administration, click the **Site Users** tab.
2. Click the **Site Administrators** button . The Site Administrators dialog box opens displaying the Site Administrators list.

   To change the sort order of the Site Administrators list from ascending to descending, click the **UserName** or **FullName** column heading. Click the column heading again to reverse the sort order.

   You can search for a user in the Site Administrators list by typing the name of a user in the **Find** box, and clicking the **Find** button .
3. Click the **Add Site Administrators** button. The Users list is displayed in the right pane.



4. Select the users that you want to assign as site administrators. You can search for users by typing a search string in the **Find** box above the Users list, and clicking the **Find** button .
5. Click the **Add Selected Users** button . Alternatively, double-click a user. The selected users are moved to the Site Administrators list in the left pane.
6. To remove a site administrator from the Site Administrators list, select the user and click the **Remove Selected Site Administrators** button. Click **OK** to confirm. The user is removed from the Site Administrators list.
7. To refresh the Site Administrators list or Users list, click the **Refresh** button above the appropriate list.

Chapter 2: Creating Projects

You can create and configure HP Application Lifecycle Management (ALM) domains and projects in Site Administration.

For information on managing existing projects, see "Managing Projects" on page 88. This includes querying project data, restoring projects, renaming projects, exporting projects, and activating or deactivating projects.

For information on upgrading projects from a previous Quality Center/HP Application Lifecycle Management version, see "Upgrading Projects to a New Version" on page 102.

This chapter includes:

About Creating Projects 35

Understanding the Project Structure 35

Creating Domains 36

Creating Projects 37

Copying Projects 44

Importing Projects 50

Creating Template Projects 54

Linking a Template to Projects 72

Updating Project Details 73

Assigning Users to Projects 77

Enabling Extensions for a Project 79

About Creating Projects

To start working in ALM, you need to create a **project**. A project collects and stores data relevant to the application management process. You can select from the following:

- Create an empty project
- Create a project based on a template project
- Copy the contents of an existing project to a new project
- Import data from an exported project file

You can also create a template project. You can link a template project to other projects to enable cross project customization. For more information, see "Cross Project Customization" on page 332.

After you create a project, you can add and remove users from the project.

Note: Users who are already logged into **Lab Management** need to re-enter the application to see changes you made in Site Administration.

Projects are grouped by **domain**. A domain contains a group of related projects and assists you in organizing and managing a large number of projects. Each domain contains a **Projects** folder and a **Template Projects** folder to organize your projects and template projects.

ALM Editions: Some functionality in Site Administration is limited per edition. For example, ALM template projects are not available with Quality Center Enterprise Edition.

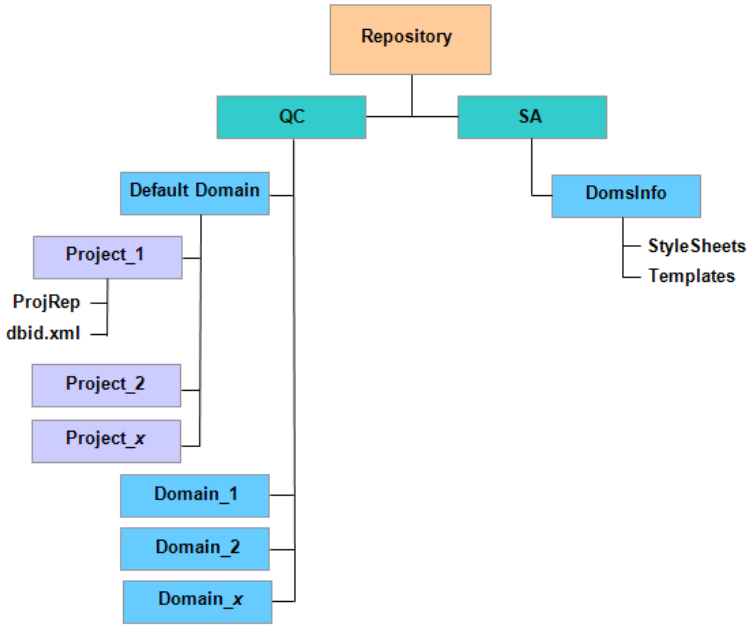
Understanding the Project Structure

When you install ALM, the installation program creates a **project repository** on the application server's file system. By default, the project repository is located under the application deployment directory, at **C:\ProgramData\HP\ALM\repository** on Windows machines, and at **/var/opt/HP/ALM/repository** on UNIX machines.

The project repository contains the **sa** and **qc** sub-folders. The **sa** directory stores global XML files, style sheets, templates, and reports to be used by all projects in the project repository.

The **qc** directory is a working area for a group of domains that are shared by multiple users. Each domain stores projects. When you create a new project, you can add it to the default domain or to a user-defined domain.

The following diagram shows the structure of the repository.



Under the **qc** directory, each project directory contains the following subdirectories:

- **ProjRep.** A subdirectory that contains a repository of all project files, such as test scripts, reports, and attachments. For more information on the project repository, see "Managing the Optimized Project Repository" on page 82.
- **dbid.xml.** An initialization file that stores project information required for restoring a connection to a project. For more information on restoring a connection to a project, see "Restoring Access to Projects" on page 97.

Under the **sa** directory, the **DomsInfo** subdirectory contains the following information:

- **StyleSheets.** A subdirectory for storing global style sheets.
- **Templates.** A subdirectory for storing database templates used when creating new projects.

Creating Domains

You can add new domains to Site Administration. You organize projects in the Projects list by domain.

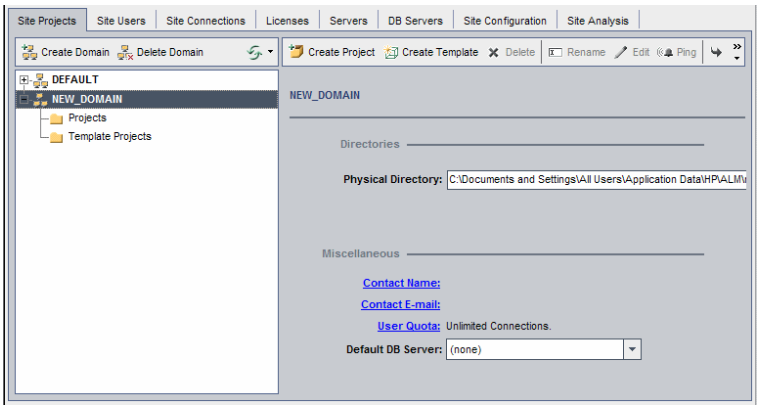
To create a domain:

1. In Site Administration, click the **Site Projects** tab.
2. Click the **Create Domain** button. The Create Domain dialog box opens.
3. Type a **Domain Name** and click **OK**.

   The domain name cannot be longer than 30 characters and can include only letters, digits, and underscores.

> **Note:** Support of non-English national characters depends on the database settings used by the server. We do not recommend using non-English national characters in domain names.

The new domain is added to the Projects list in alphabetical order. In the right pane, under **Directories**, you can view the location of the domain.



4. To add a person's name as a contact when there are questions or problems with the domain or its projects, click the **Contact Name** link. In the Set Contact Name dialog box, type the name of the contact person and click **OK**.
5. To add the email address of the contact person for the domain, click the **Contact Email** link. In the Set Contact Email dialog box, type the email address and click **OK**.
6. To change the number of users allowed to connect concurrently to the domain, click the **User Quota** link. The Domain User Quota dialog box opens.

   Choose **Maximum Connections** and type the maximum number of concurrent connections allowed. Click **OK**.

   > **Note:** In addition to changing the number of users allowed to connect concurrently to a domain, you can also change:
   >
   > - the number of users allowed to connect concurrently to a project. For more information, see "Updating Project Details" on page 73.
   > - the number of concurrent connections that can be opened on the database server by ALM for each project. For more information, see "Configuring Server Information" on page 157.

7. To select a default database server when creating projects in the domain, select a default database server from the **Default DB Server** list.

Creating Projects

You can create ALM projects in Oracle or Microsoft SQL. You can create a project in any of the following ways:

- Create an empty project.
- Create a project from a template. This option copies the customization of an existing template project. It does not copy project data from the template project. **ALM Editions:** This option is not available for **Quality Center Enterprise Edition**.
- Copy the contents of an existing project. For more information, see "Copying Projects" on page 44.
- Import data from an exported project file. For more information, see "Importing Projects" on page 50.

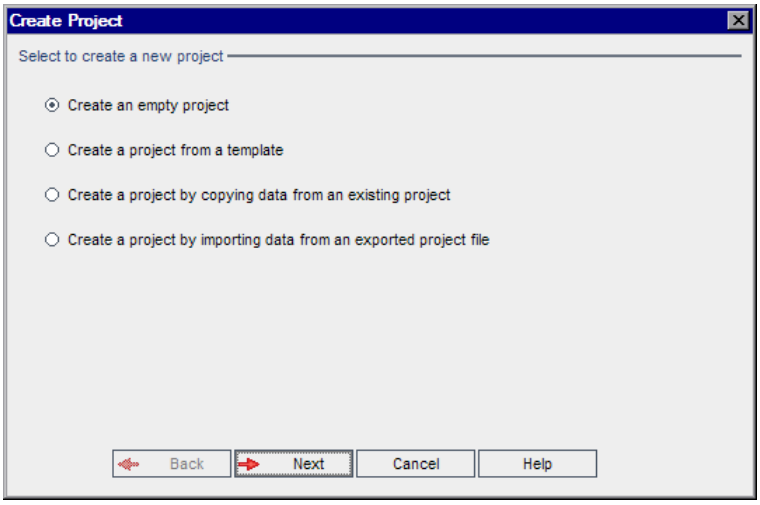
For information on creating template projects, see "Creating Template Projects" on page 54.

Note:

- For information on the Oracle or Microsoft SQL permissions required by ALM, refer to the *HP Application Lifecycle Management Installation and Upgrade Guide*.
- If your database server is connected to ALM over a WAN, the project creation process can be very time-consuming.

To create a project:

1. In Site Administration, click the **Site Projects** tab.
2. Select the domain in which you want to create the project.
3. Click the **Create Project** button. The Create Project dialog box opens.



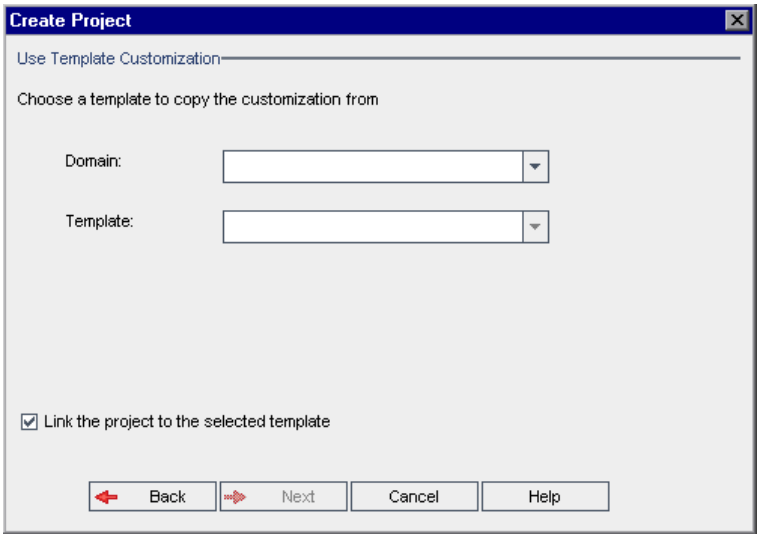
4. Select one of the following options:
   - **Create an empty project**. Creates a new project.
   - **Create a project from a template**. Creates a new project by copying the customization of an existing template project, but not the project data. **ALM Editions:** This option is not available for **Quality Center Enterprise Edition**.
   - **Create a project by copying data from an existing project**. For more information, see "Copying Projects" on page 44.

- **Create a project by importing data from an exported project file**. For more information, see "Importing Projects" on page 50.

5. Click **Next**.

   If you selected **Create a project from a template**, the **Use Template Customization** dialog box opens. **ALM Editions:** This dialog box is not available for **Quality Center Enterprise Edition**.

   If you did not select **Create a project from a template**, proceed to step **7**.



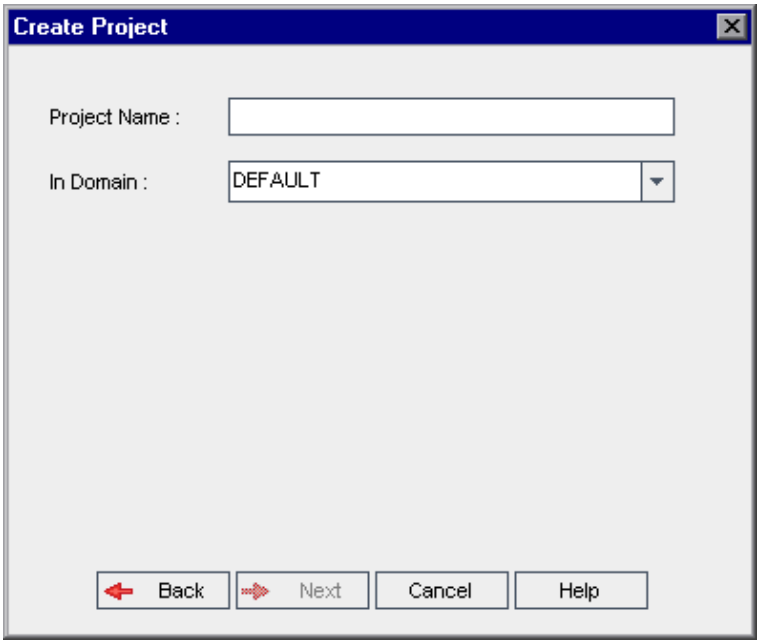
   Select the **domain** and **template** you want to use to create the project.

6. Select **Link the project to the selected template** to link the new project to the template. **ALM Editions:** This option is not available for for **Quality Center Enterprise Edition**.

   Note: After you link a project to a template, the template administrator must apply template customization to the project. This applies the customization from the template to the linked project, and sets the applied customization to read-only in the project. For more information, see "Applying Template Customization to Linked Projects" on page 337.

   Click **Next**.

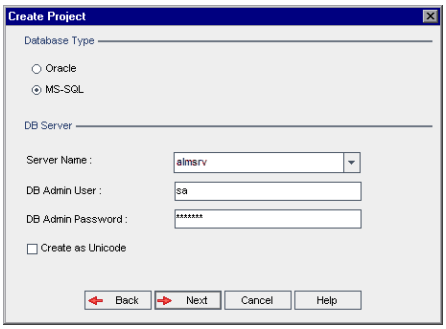
7. The following dialog box opens.



8. In the **Project Name** box, type a name for your project. The project name cannot be longer than 30 characters and can include only letters, digits, and underscores.

   Note: Support of non-English national characters depends on the database settings used by the server. We do not recommend using non-English national characters in project names.

9. In the **In Domain** box, select a domain.
10. Click **Next**. The following dialog box opens.



11. Under **Database Type**, select **Oracle** or **MS-SQL**.
12. By default, the default values defined for the domain are displayed for **Server Name**, **DB Admin User**, and **DB Admin Password**. If additional database servers are defined, you can select another name from the **Server Name** list.
13. Click **Create as Unicode** to create the project as Unicode.

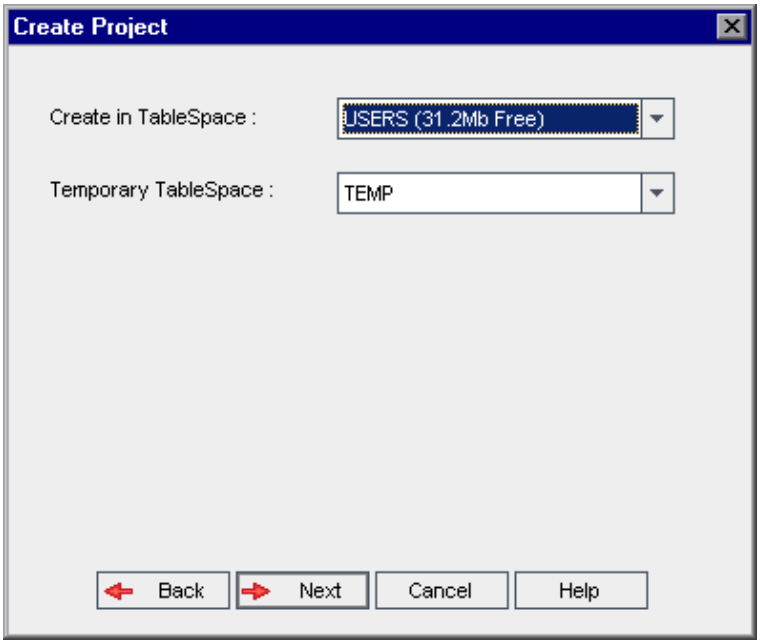
    Note: The **Create as Unicode** checkbox is only displayed when creating a new project from an empty project in an MS-SQL server. Unicode is a feature of MS-SQL that allows

multi-language support. In Oracle, multi-language support is defined when installing the server.

14. Click **Next**.

    If your selected database server does not have the text search feature enabled, a message box opens. The message indicates that after this process completes, you can enable the text search feature. For more information on enabling the text search feature, see "Configuring Text Search" on page 163.

15. If you are creating a Microsoft SQL project, proceed to step **16**. For an Oracle project, the following dialog box opens.

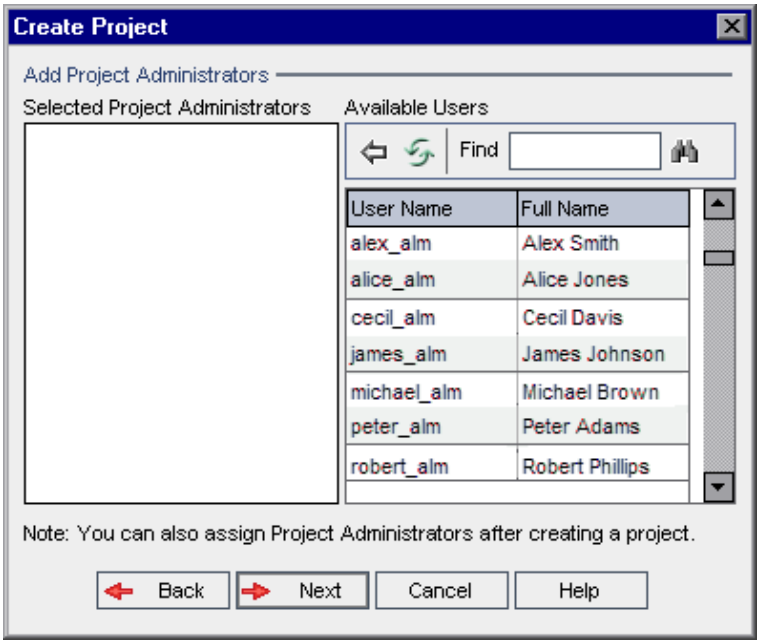


    In the **Create in TableSpace** box, select a storage location that has sufficient space to store the new project. You should not use **UNDO** as the storage location.

    In the **Temporary TableSpace** box, select a temporary storage location that has sufficient space to store the new project.

    Click **Next**.

16. The Add Project Administrators dialog box opens.

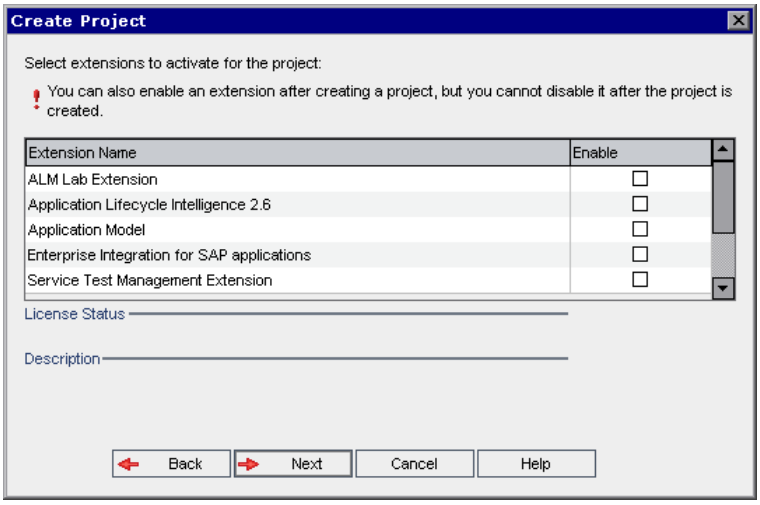


Selected Project Administrators lists users that are assigned as project administrators. **Available Users** lists users available in the project. When you assign project administrators, they are moved from the Available Users list to the Selected Project Administrators list. Project administrator users can add and administer other users in the project.

- **Refresh.** Click the **Refresh** button to refresh the list of available users.
- **Find.** Type the name of a user in the **Find** box, and click the **Find** button to search the Available Users list.
- **Add Selected Users.** Select the users you want to assign as project administrators, and click the **Add Selected Users** button. Alternatively, double-click a user name. The selected users are moved to the Selected Project Administrators list.
- **Delete.** To remove a user from the Selected Project Administrators list, right-click the user name and click **Delete**.

You can also assign project administrators after you have created the project. For more information, see "Assigning Project Administrators" on page 78.

17. Click **Next**. You can enable extensions that are available for your site's ALM edition.



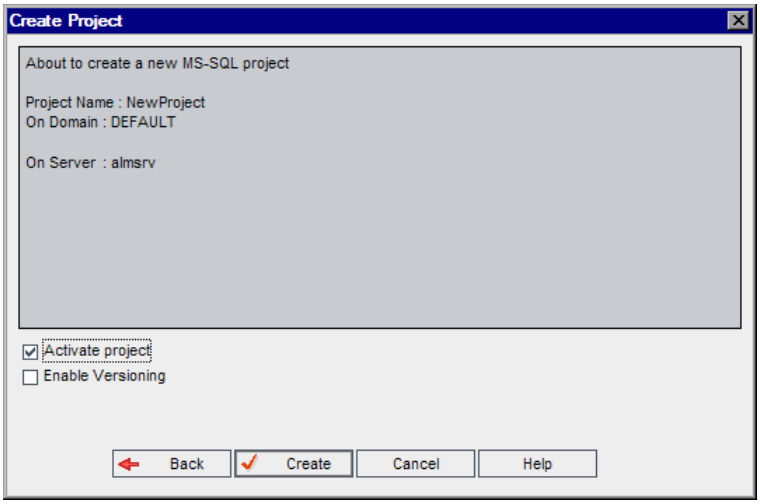
Select the **Enable** check box for the extensions you want to enable.

Note

- You cannot disable an extension for a project after you enable it. We recommend enabling only the extensions you need. Enabling extra extensions can affect performance and consumes additional disk space.
- As you navigate from one extension to another, relevant license status information is displayed in the **License Status** section.

  Extensions for which there is no license on the server are displayed in gray. It is possible to enable an extension for which you do not yet have a license. You can utilize the extension's added functionality later, after receiving the license.
- To work with **Lab Management** or **Performance Center**, select **ALM Lab Extension**. For more information, refer to the *HP ALM Lab Management Guide*.
- You can also enable extensions for a project after you have created the project. For more information, see "Enabling Extensions for a Project" on page 79.

18. Click **Next**. The following dialog box opens.



Verify the project details. To change any of the details, click **Back**.

19. Select **Activate Project** to activate the new project. Only activated projects are available to users in the HP Application Lifecycle Management Login window when they log in to a project. For more information, see "Deactivating and Activating Projects" on page 91.
20. Select **Enable Versioning** to enable version control for the project. You can also enable version control after you have created the project. For more information, see "Enabling and Disabling Version Control for a Project" on page 91.
21. Click **Create**. The new project is added to the Projects list.

Copying Projects

You can create a new project by copying the contents of an existing project.

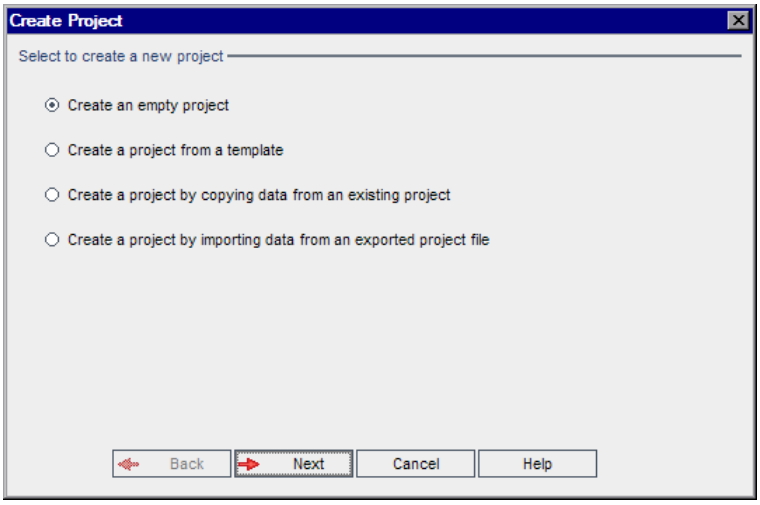
Version Control: If you copy a version control enabled project, the new project is created with version control enabled. Version history is also copied. Entities that are checked out in the source project are checked out in the new project. An administrator in the new project can undo those checkouts. For more information, see the *HP Application Lifecycle Management User Guide*.

Note:

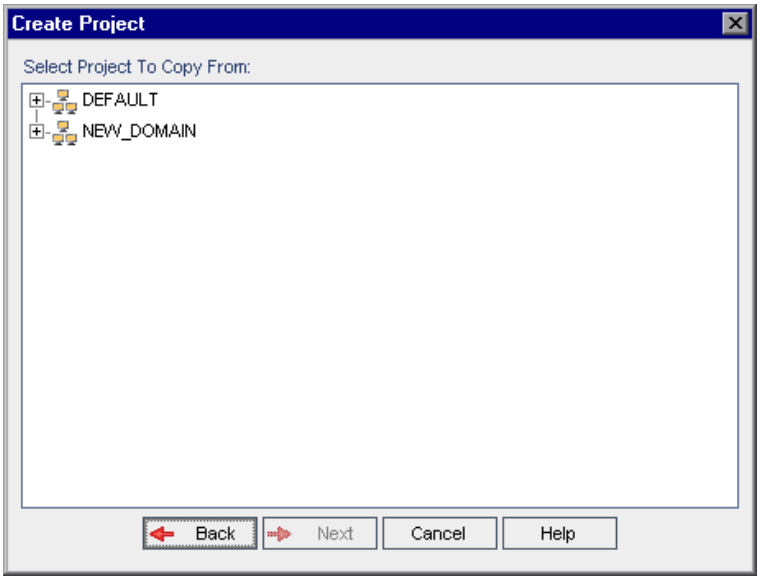
- If your ALM server becomes unavailable while copying, you can resume the copying process at a later stage. To resume copying, reopen Site Administration and select the project from the Projects list. In the right pane, click the **Click Here** link.
- The new project inherits the Unicode or ASCII definition of the copied project.
- Test run details are not be included in the Usage Reports in the copied project.
- Timeslot information and project settings are not copied.
- Result files associated with the original project are not copied.

To copy a project:

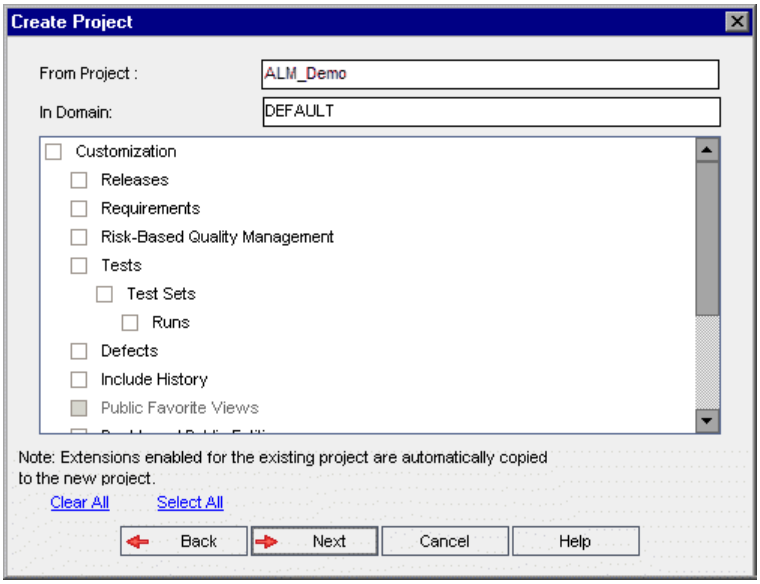
1. Deactivate the project you want to copy. For more information, see "Deactivating and Activating Projects" on page 91.
2. In Site Administration, click the **Site Projects** tab.
3. Select the domain in which you want to create the project.
4. Click the **Create Project** button. The Create Project dialog box opens.



5. Choose the **Create a project by copying data from an existing project** option and click **Next**. The following dialog box opens.



6. Under **Select Project To Copy From**, select the domain and project you want to copy and click **Next**. The following dialog box opens.



7. Select **Customization** to copy project lists, host data, system and user-defined fields, module access, workflow, and transition rules to the new project. If this option is selected, you can also choose to copy any of the following:

Option	Description
Releases	Copies release data from the project.
Requirements	Copies requirement data from the project.
Risk-Based Quality Management	Copies risk-based quality management customization settings from the project. For more information, see "Customizing Risk-Based Quality Management" on page 318.
Tests	Copies test data and test resources from the project. If this option is selected, you can also choose the following option: ▪ **Test Sets.** Copies test set data from the project. If this option is selected, you can also choose to copy the following option: ○ **Runs.** Copies test run data from the project.
Defects	Copies defect data from the project.
Include History	Copies history data for the options that are selected.
Public Favorite Views	Copies public favorite view data from the project. For more information, refer to the *HP Application Lifecycle Management User Guide*.
Dashboard Public Entities	Copies public analysis items and dashboard pages from the project. For more information, refer to the *HP Application Lifecycle Management User Guide*.

Option	Description
Users and Groups	Copies user and group information and permission settings. If this option is selected, you can also choose to copy the following options: ▪ **Dashboard Private Entities.** Copies private analysis items and dashboard pages from the project. For more information, refer to the *HP Application Lifecycle Management User Guide*. ▪ **Private Favorite Views.** Copies private favorite view data and Excel report definitions from the project. For more information, refer to the *HP Application Lifecycle Management User Guide*. ▪ **Mail Conditions.** Copies the mailing configuration data. For more information, see "Configuring Automail" on page 314. ▪ **Alerts and Follow up Flags.** Copies alerts and follow up flags. For more information, refer to the *HP Application Lifecycle Management User Guide*.
Sprinter	Copies Sprinter data. This cannot be unselected.
Analysis Extension	Copies Analysis data. This cannot be unselected.
Quality Center	Copies Quality Center data. This cannot be unselected.

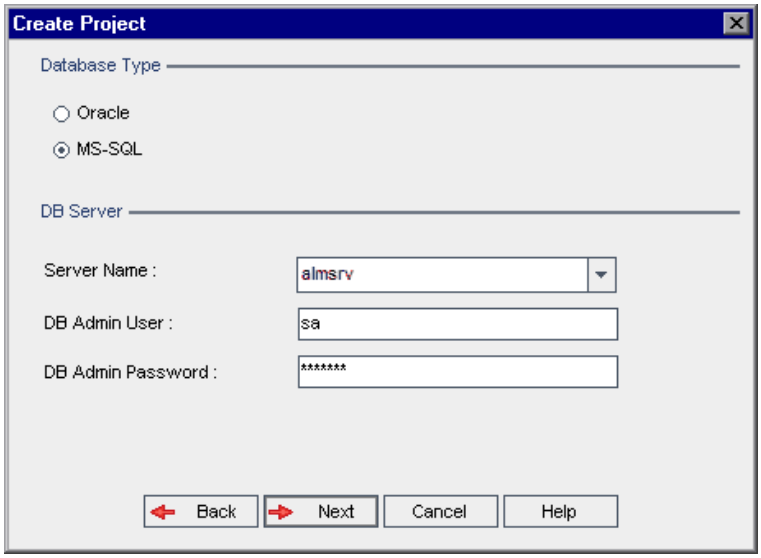
Note:

- If the project from which you are copying has extensions enabled, the extensions and their associated data are also copied to the new project.
- If the project from which you are copying contains libraries, the libraries are not copied to the new project. For information on importing libraries, see the *HP Application Lifecycle Management User Guide*.

To clear all options, click **Clear All**.

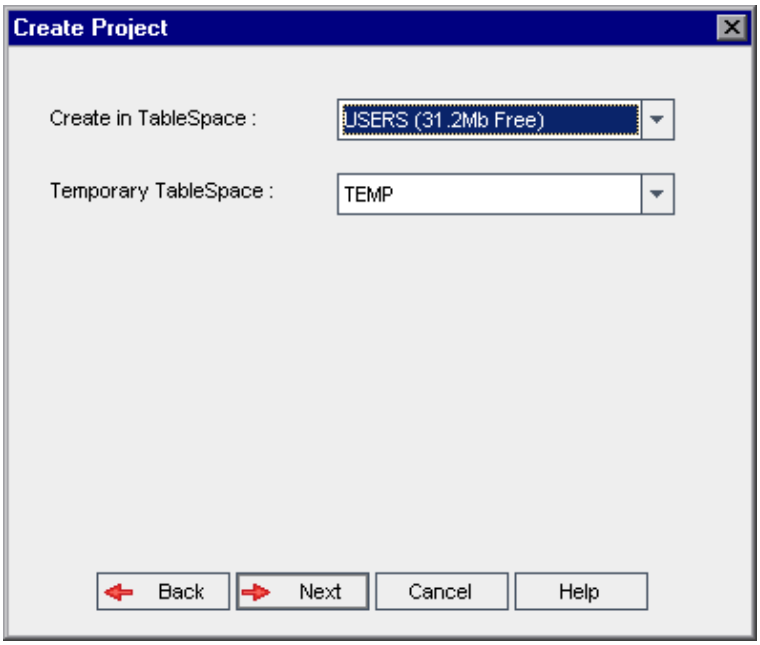
To select all options, click **Select All**.

8. Click **Next**.
9. In the **Project Name** box, type a name for your project. The project name cannot be longer than 30 characters and cannot include any of the following characters: **= ~ ` ! @ # $ % ^ & * ( ) + | { } [ ] : ' ; " < > ? , . / \ -**
10. In the **In Domain** box, select a domain.
11. Click **Next**. The following dialog box opens.



12. Under **Database Type**, select **Oracle** or **MS-SQL**.
13. By default, the default values defined for the domain are displayed for **Server Name**, **DB Admin User**, and **DB Admin Password**. If additional database servers are defined, you can select another name from the **Server Name** list.
14. Click **Next**.

    If your selected database server does not have the text search feature enabled, a message box opens. The message indicates that after this process completes, you can enable the text search feature. For more information on enabling the text search feature, see "Configuring Text Search" on page 163.
15. If you are creating a Microsoft SQL project, proceed to step **16**. For an Oracle project, the following dialog box opens.

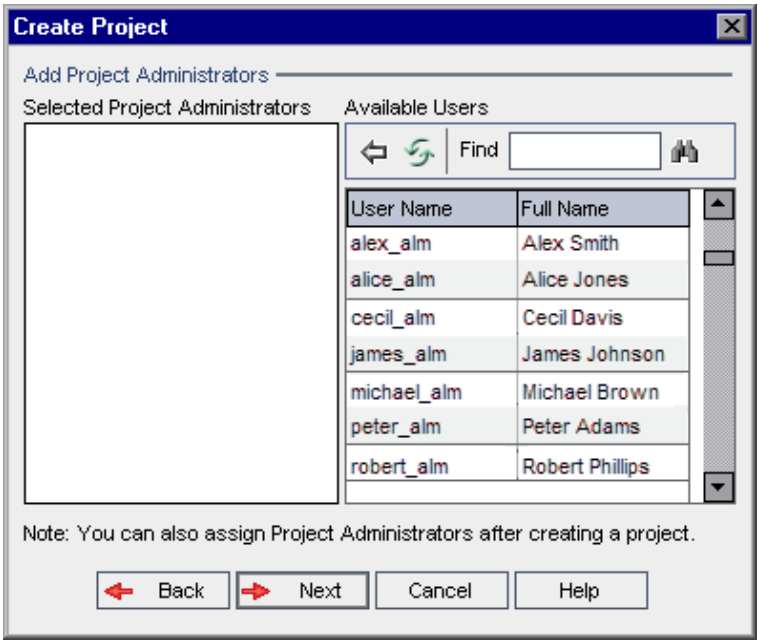


In the **Create in TableSpace** box, select a storage location that has sufficient space to store the new project. You should not use **UNDO** as the storage location.

In the **Temporary TableSpace** box, select a temporary storage location that has sufficient space to store the new project.

Click **Next**.

16. The Add Project Administrators dialog box opens.

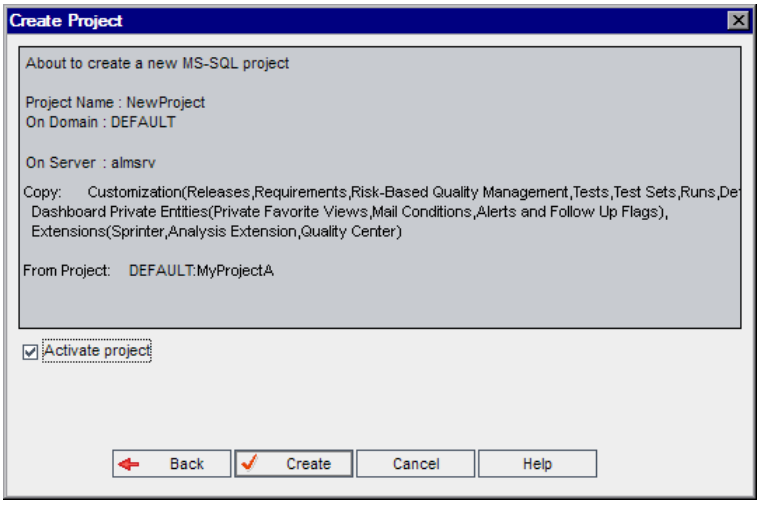


Selected Project Administrators lists users that are assigned as project administrators. **Available Users** lists users available in the project. When you assign project administrators, they are moved from the Available Users list to the Selected Project Administrators list. Project administrator users can add and administer other users in the project.

- **Refresh.** Click the **Refresh** button to refresh the list of available users.
- **Find.** Type the name of a user in the **Find** box, and click the **Find** button to search the Available Users list.
- **Add Selected Users.** Select the users you want to assign as project administrators, and click the **Add Selected Users** button. Alternatively, double-click a user name. The selected users are moved to the Selected Project Administrators list.
- **Delete.** To remove a user from the Selected Project Administrators list, right-click the user name and click **Delete**.

You can also assign project administrators after you have created the project. For more information, see "Assigning Project Administrators" on page 78.

17. Click **Next**. The following dialog box opens.



Verify the project details. To change any of the details, click **Back**.

18. Select **Activate Project** to activate the new project. Only activated projects are available to users in the HP Application Lifecycle Management Login window when they log in to a project. For more information, see "Deactivating and Activating Projects" on page 91.
19. Click **Create**. The new project is added to the Projects list.

Importing Projects

You can import data from exported ALM project files created in the same ALM version. You can also import data from customized projects created by content providers. For example, you can import customized tests, requirements, and test sets for SAP testing, Siebel testing, and SOX compliance testing created by HP content providers.

If you import a project that was previously exported from the same server, ALM recognizes that the same project already exists on the server, based on the project ID. You can choose to replace the existing project, or cancel the import process.

Note: The new project inherits the Unicode or ASCII definition of the imported project.

Version Control: If you import an exported version control enabled project, the project is imported with version control enabled. Version history is also copied.

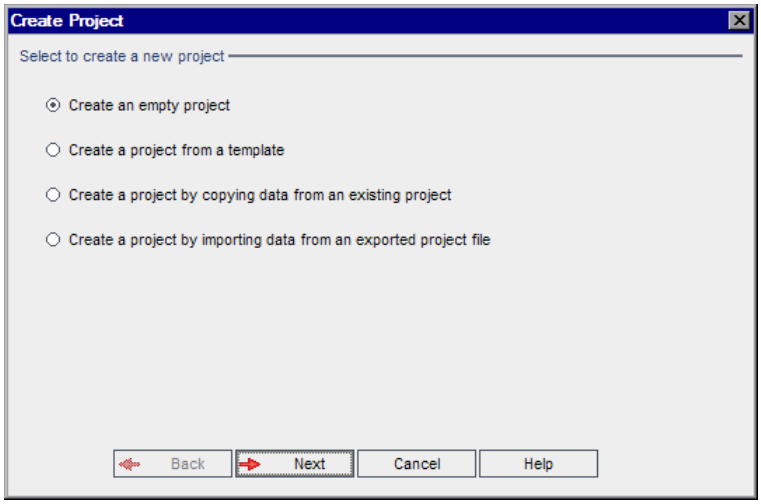
You can also import data from template projects. For more information, see "Importing a Template Project" on page 68.

For more information on exporting projects, see "Exporting Projects" on page 90.

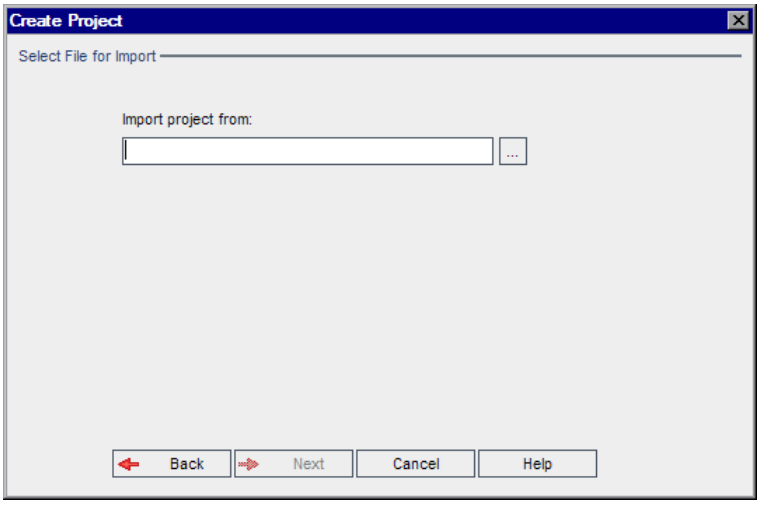
To import an ALM project:

1. In Site Administration, click the **Site Projects** tab.
2. You can do one of the following:
   - Select the domain to which you want to import a project, and click the **Import Project from Project File** button . Alternatively, right-click the domain and choose **Import Project**.

- Click the **Create Project** button. The Create Project dialog box opens.



3. Choose the **Create a project by importing data from an exported project file** option, and click **Next**. The Select File for Import dialog box opens.



4. Click the browse button to the right of the **Import project from** box to locate the project that you want to import. The Open dialog box opens.
5. Locate the directory and select the ALM Project Export file that you want to import. Click **Open**. The selected file is displayed in the **Import project from** box.

> **Note:** If the selected file is an ALM template project file, a new template project is created. The template project is added to the Projects list under **Template Projects**. **ALM Editions:** Template projects are not available for **Quality Center Enterprise Edition**.

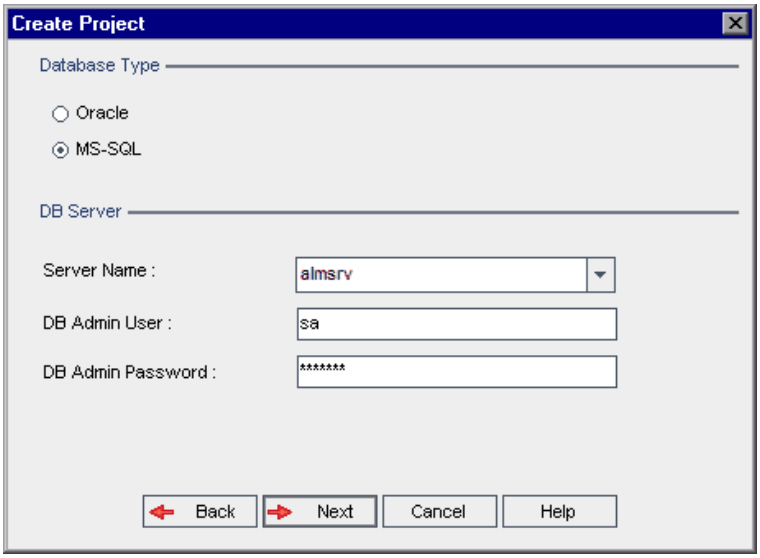
Click **Next**.

6. In the **Project Name** box, type a name for your project. The project name cannot be longer than 30 characters and cannot include any of the following characters: **= ~ ` ! @ # $ % ^ & * ( ) + | { } [ ] : ' ; " < > ? , . / \ -**

7. In the **In Domain** box, select a domain.

   Tip: After the project has been created, you can move it to a different domain in the Projects list using a drag-and-drop operation.

8. Click **Next**. The following dialog box opens.



9. Under **Database Type**, select **Oracle** or **MS-SQL**.

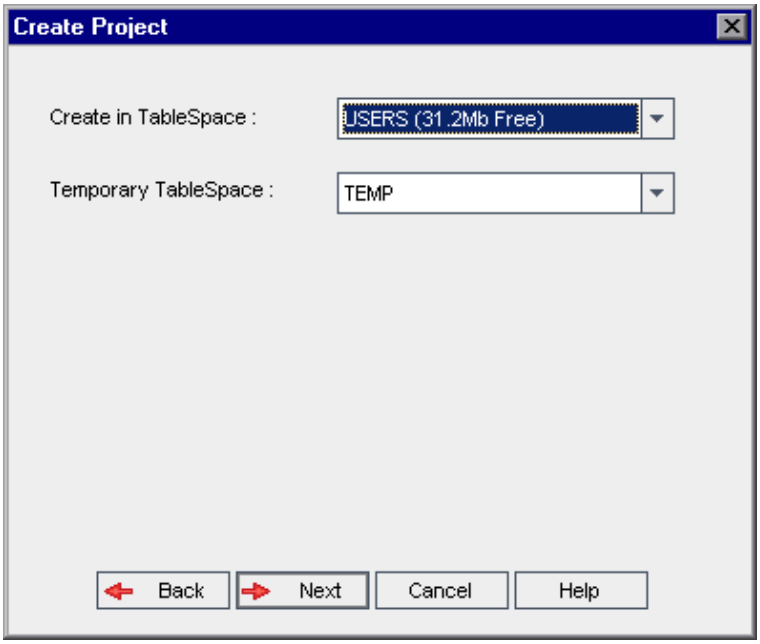
   By default, the default values defined for the domain are displayed for **Server Name**, **DB Admin User**, and **DB Admin Password**. If additional database servers are defined, you can select another name from the **Server Name** list.

   Note: For more information on defining database servers, see "Defining New Database Servers" on page 159.

10. Click **Next**.

    If your selected database server does not have the text search feature enabled, a message box opens. The message indicates that after this process completes, you can enable the text search feature. For more information on enabling the text search feature, see "Configuring Text Search" on page 163.

11. If you are creating a Microsoft SQL project, proceed to step **12**. For an Oracle project, the following dialog box opens.

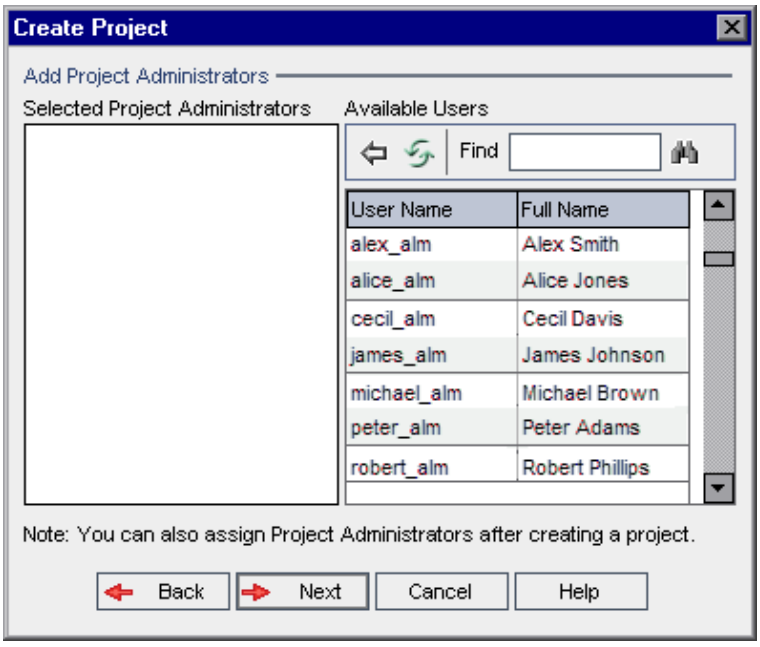


In the **Create in TableSpace** box, select a storage location that has sufficient space to store the new project. You should not use **UNDO** as the storage location.

In the **Temporary TableSpace** box, select a temporary storage location that has sufficient space to store the new project.

Click **Next**.

12. In the Add Project Administrators dialog box, select project administrators.



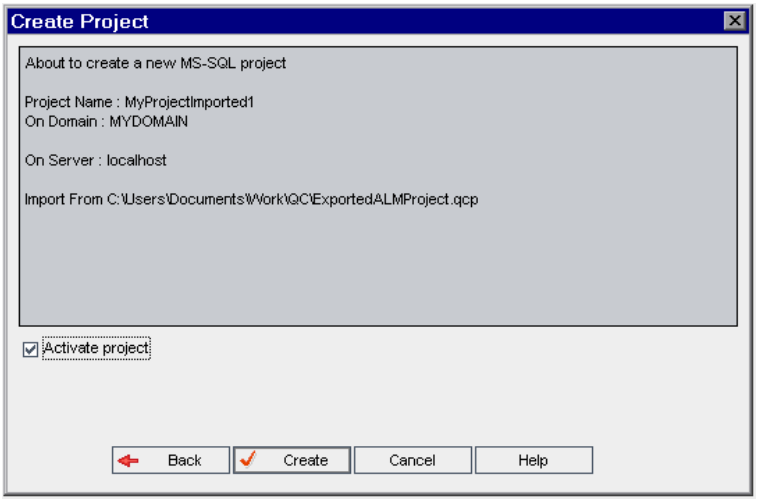
Selected Project Administrators lists users that are assigned as project administrators. **Available Users** lists users available in the project. When you assign project administrators,

they are moved from the Available Users list to the Selected Project Administrators list. Project administrator users can add and administer other users in the project.

- **Refresh.** Click the **Refresh** button to refresh the list of available users.
- **Find.** Type the name of a user in the **Find** box, and click the **Find** button to search the Available Users list.
- **Add Selected Users.** Select the users you want to assign as project administrators, and click the **Add Selected Users** button. Alternatively, double-click a user name. The selected users are moved to the Selected Project Administrators list.
- **Delete.** To remove a user from the Selected Project Administrators list, right-click the user name and click **Delete**.

You can also assign project administrators after you have created the project. For more information, see "Assigning Project Administrators" on page 78.

13. Click **Next**. The following dialog box opens.



Verify the project details. To change any of the details, click **Back**.

14. Select **Activate Project** to activate the new project. Only activated projects are available to users in the HP Application Lifecycle Management Login window when they log in to a project. For more information, see "Deactivating and Activating Projects" on page 91.
15. Click **Create**. The new project is added to the Projects list.

Creating Template Projects

Template projects enable you to define and maintain a common set of project customizations for multiple projects. When you create a template, you can link it to projects. This enables the template administrator to apply template customization changes to the linked projects.

You create a new template project by creating an empty template, by copying an existing template or project, or by importing a template.

ALM Editions: Template projects are not available for **Quality Center Enterprise Edition**.

This section includes:

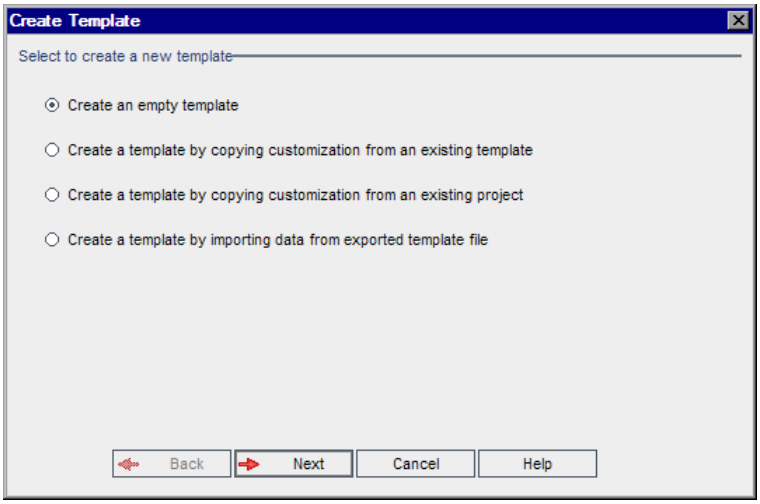
Creating a Template Project 55
Creating a Template from an Existing Template 59
Creating a Template from an Existing Project 64
Importing a Template Project 68

Creating a Template Project

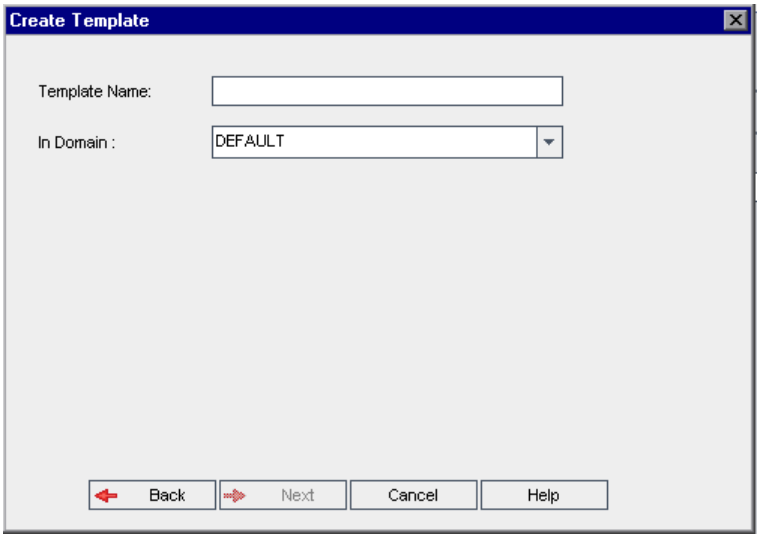
You can create a new template project in Oracle or Microsoft SQL.

To create a template:

1. In Site Administration, click the **Site Projects** tab.
2. Select the domain in which you want to create the template.
3. Click the **Create Template** button. The Create Template dialog box opens.



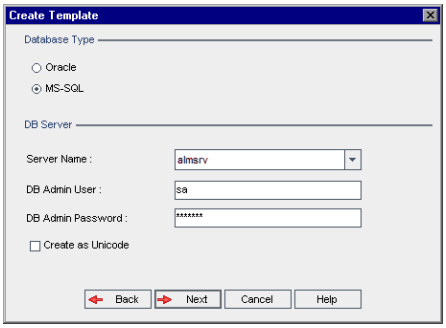
4. Select **Create an empty template** and click **Next**. The following dialog box opens.



5. In the **Template Name** box, type a name for the template. The template name cannot be longer than 30 characters and cannot include any of the following characters: **= ~ ` ! @ # $ % ^ & * ( ) + | { } [ ] : ' ; " < > ? , . / \ -**
6. In the **In Domain** box, select a domain.

> **Tip:** After the template has been created, you can move it to a different domain in the Projects list using a drag-and-drop operation.

7. Click **Next**. The following dialog box opens.



8. Under **Database Type**, select **Oracle** or **MS-SQL**.

   The default values defined for the domain are displayed for **Server Name**, **DB Admin User**, and **DB Admin Password**. If additional database servers are defined, you can select another name from the **Server Name** list.

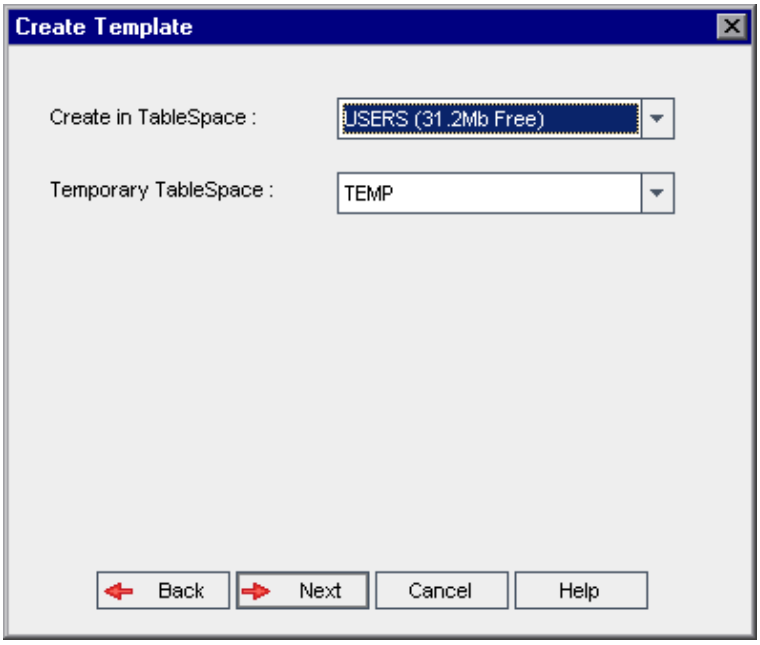
9. Click **Create as Unicode** to create the template as Unicode.

> **Note:** The **Create as Unicode** checkbox is only displayed when creating a new template from an empty template in an MS-SQL server. Unicode is a feature of MS-SQL that allows multi-language support. In Oracle, multi-language support is defined when installing the server.

10. Click **Next**.

If your selected database server does not have the text search feature enabled, a message box opens. The message indicates that after this process completes, you can enable the text search feature. For more information on enabling the text search feature, see "Configuring Text Search" on page 163.

11. If you are creating a Microsoft SQL template, proceed to step **12**. For an Oracle template, the following dialog box opens.

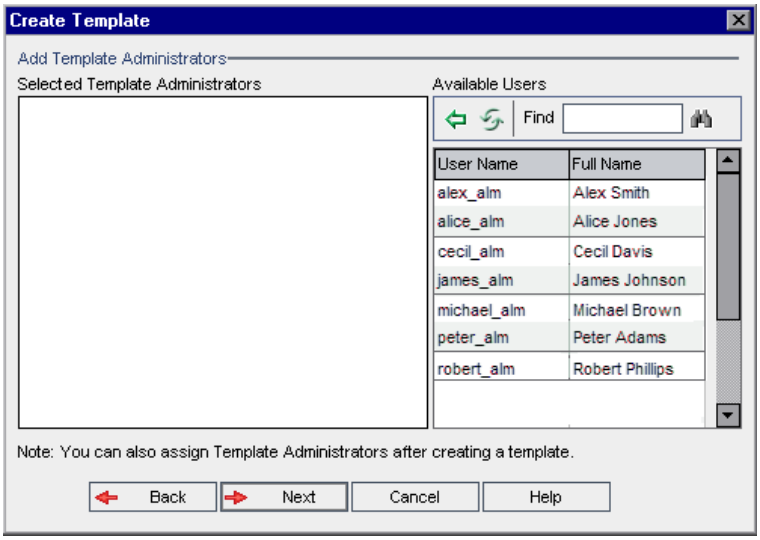


In the **Create in TableSpace** box, select a storage location that has sufficient space to store the new template. You should not use **UNDO** as the storage location.

In the **Temporary TableSpace** box, select a temporary storage location that has sufficient space to store the new template.

Click **Next**.

12. The Add Template Administrators dialog box opens.

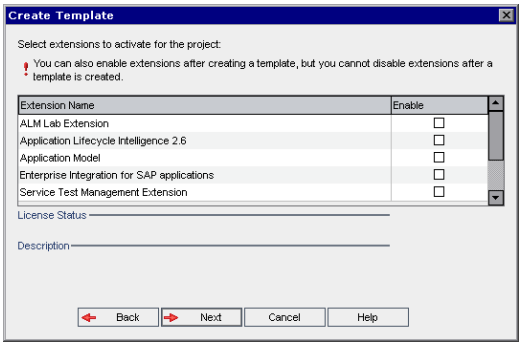


Selected Template Administrators lists users that are assigned as template administrators. **Available Users** lists users available in the template. When you assign template administrators, the users are moved from the Available Users list to the Selected Template Administrators list. Template administrator users can customize template projects and apply template customization to linked projects. For more information, see "Cross Project Customization" on page 332.

- **Refresh.** Click the **Refresh** button to refresh the list of available users.
- **Find.** Type the name of a user in the **Find** box, and click the **Find** button to search the Available Users list.
- **Add Selected Users.** Select the users you want to assign as template administrators, and click the **Add Selected Users** button. Alternatively, double-click a user name. The selected users are moved to the Selected Template Administrators list.
- **Delete.** To remove a user from the Selected Template Administrators list, right-click the user name and click **Delete**.

You can also assign template administrators after you have created the template. For more information, see "Assigning Project Administrators" on page 78.

13. Click **Next**. You can enable extensions that are available for your site's ALM edition.



Select the **Enable** check box for the extensions you want to enable.

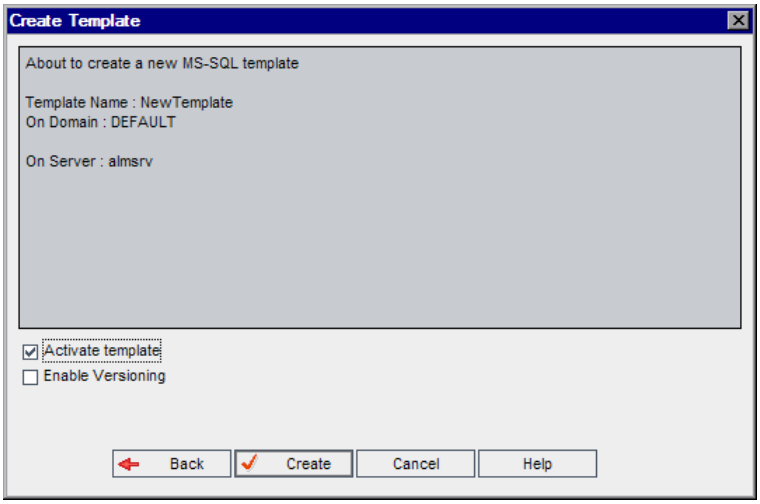
Note

- You cannot disable an extension for a project after you enable it. We recommend enabling only the extensions you need. Enabling extra extensions can affect performance and consumes additional disk space.
- As you navigate from one extension to another, relevant license status information is displayed in the **License Status** section.

  Extensions for which there is no license on the server are displayed in gray. It is possible to enable an extension for which you do not yet have a license. You can utilize the extension's added functionality later, after receiving the license.
- To work with **Lab Management** or **Performance Center**, select **ALM Lab Extension**. For more information, refer to the *HP ALM Lab Management Guide*.

- You can also enable extensions for a project after you have created the project. For more information, see "Enabling Extensions for a Project" on page 79.

14. Click **Next**. The following dialog box opens.



Verify the template details. To change any of the details, click **Back**.

15. Select **Activate template** to activate the template. Only activated templates are available in the HP Application Lifecycle Management Login window. For more information, see "Deactivating and Activating Projects" on page 91.
16. Select **Enable Versioning** to enable version control for the template. You can also enable version control after you have created the template. For more information, see "Enabling and Disabling Version Control for a Project" on page 91.
17. Click **Create**. The new template is added to the Projects list under **Template Projects**.

Creating a Template from an Existing Template

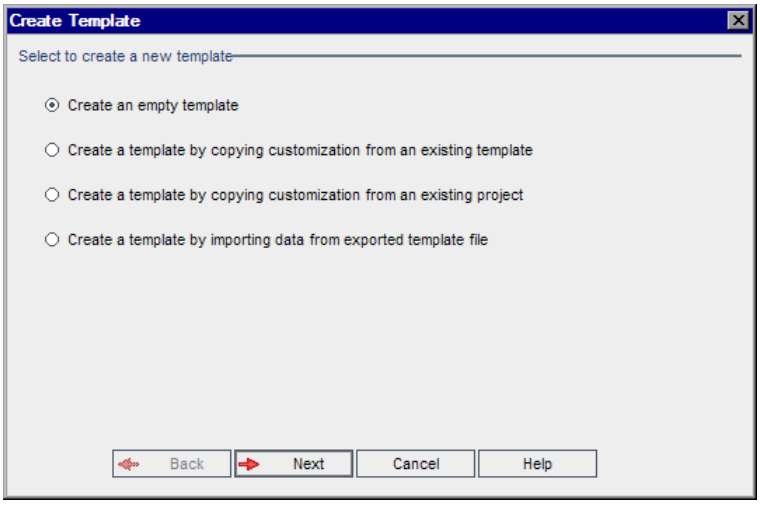
You can create a template project by copying an existing template. This option copies both customization and project data from the source template.

Note: The new template inherits the Unicode or ASCII definition of the copied project.

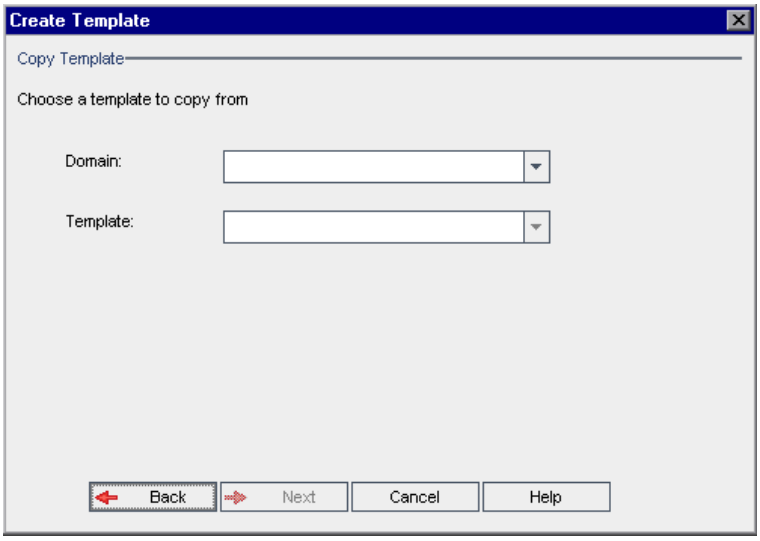
Version Control: If you copy a version control enabled template, the new template is created with version control enabled.

To create a template from an existing template:

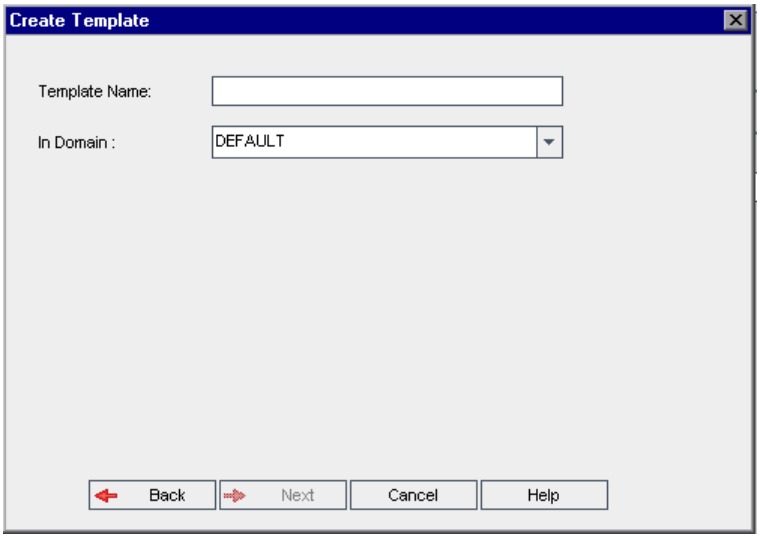
1. In Site Administration, click the **Site Projects** tab.
2. Select the domain in which you want to create the template.
3. Click the **Create Template** button. The Create Template dialog box opens.



4. Select **Create a template by copying customization from an existing template** and click **Next**. The Copy Template dialog box opens.



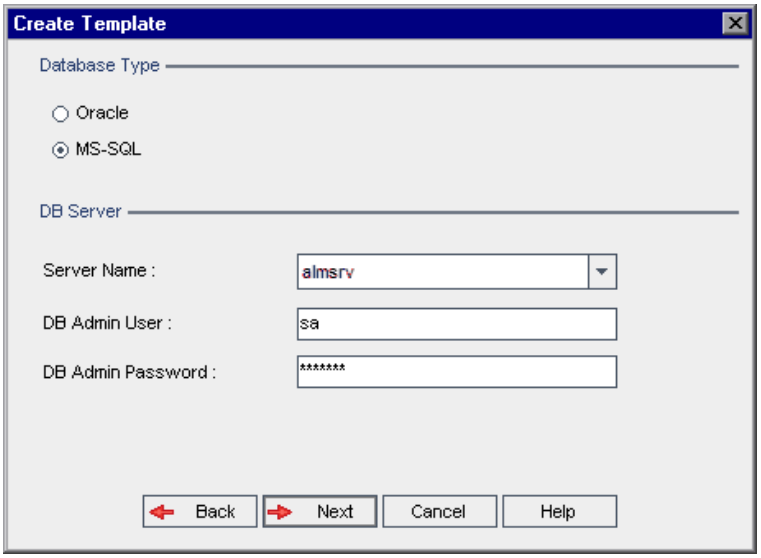
5. In the **Domain** box, select the domain where the template you want to copy is located.
6. In the **Template** box, select the template you want to copy.
7. Click **Next**. The following dialog box opens.



8. In the **Template Name** box, type a name for the template. The template name cannot be longer than 30 characters and cannot include any of the following characters: **= ~ ` ! @ # $ % ^ & * ( ) + | { } [ ] : ' ; " < > ? , . / \ -**
9. In the **In Domain** box, select a domain.

> **Tip:** After the template has been created, you can move it to a different domain in the Projects list using a drag-and-drop operation.

10. Click **Next**. The following dialog box opens.



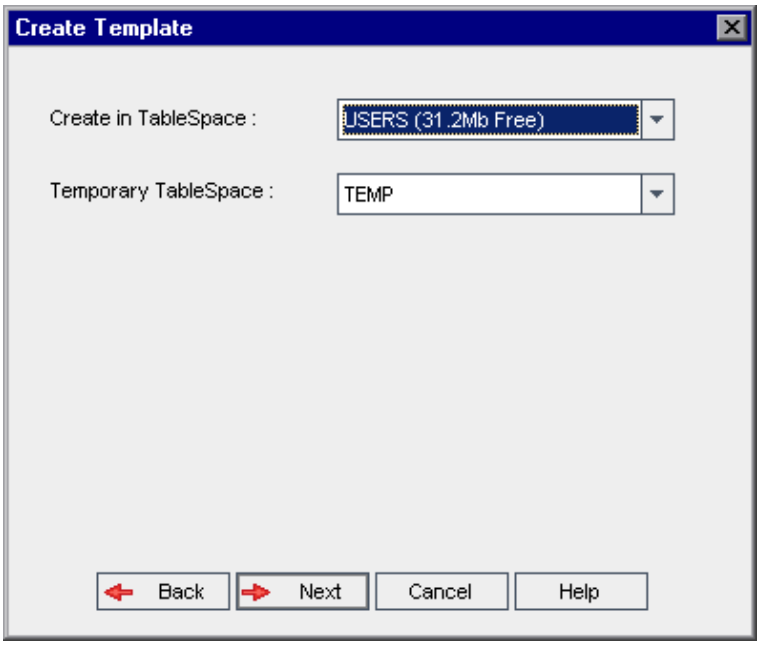
11. Under **Database Type**, select **Oracle** or **MS-SQL**.

    The default values defined for the domain are displayed for **Server Name**, **DB Admin User**, and **DB Admin Password**. If additional database servers are defined, you can select another name from the **Server Name** list.

12. Click **Next**.

Note: If your selected database server does not have the text search feature enabled, a message box opens. The message indicates that after this process completes, you can enable the text search feature. For more information on enabling the text search feature, see "Configuring Text Search" on page 163.

If you are creating a Microsoft SQL template, proceed to step **13**. For an Oracle template, the following dialog box opens.



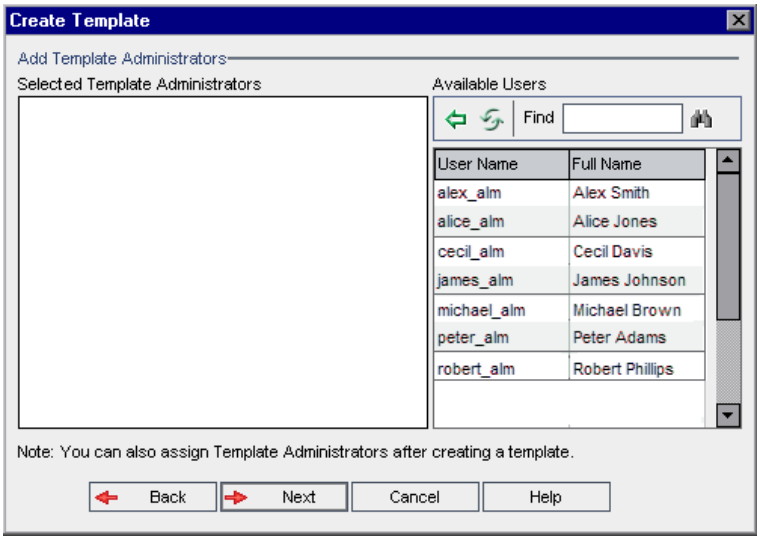
In the **Create in TableSpace** box, select a storage location that has sufficient space to store the new template.

You should not use **UNDO** as the storage location.

In the **Temporary TableSpace** box, select a temporary storage location that has sufficient space to store the new template.

Click **Next**.

13. The Add Template Administrators dialog box opens.

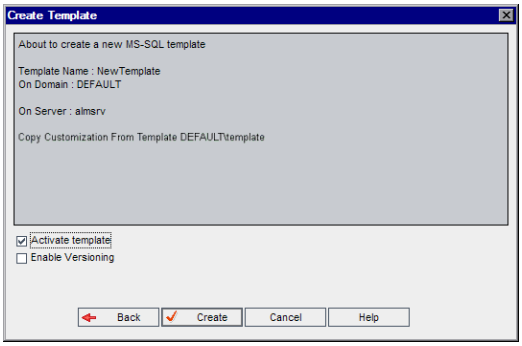


Selected Template Administrators lists users that are assigned as template administrators. **Available Users** lists users available in the template. When you assign template administrators, the users are moved from the Available Users list to the Selected Template Administrators list. Template administrator users can customize template projects and apply template customization to linked projects. For more information, see "Cross Project Customization" on page 332.

- **Refresh.** Click the **Refresh** button to refresh the list of available users.
- **Find.** Type the name of a user in the **Find** box, and click the **Find** button to search the Available Users list.
- **Add Selected Users.** Select the users you want to assign as template administrators, and click the **Add Selected Users** button. Alternatively, double-click a user name. The selected users are moved to the Selected Template Administrators list.
- **Delete.** To remove a user from the Selected Template Administrators list, right-click the user name and click **Delete**.

You can also assign template administrators after you have created the template. For more information, see "Assigning Project Administrators" on page 78.

14. Click **Next**. The following dialog box opens.



Verify the template details. To change of the details, click **Back**.

15. Select **Activate template** to active the template. Only activated templates are available in the HP Application Lifecycle Management Login window. For more information, see "Deactivating and Activating Projects" on page 91.
16. Click **Create**. The new template is added to the Projects list under **Template Projects**.

Creating a Template from an Existing Project

You can create a template project by copying the customization of an existing project. This option copies customization from the project but does not copy project data.

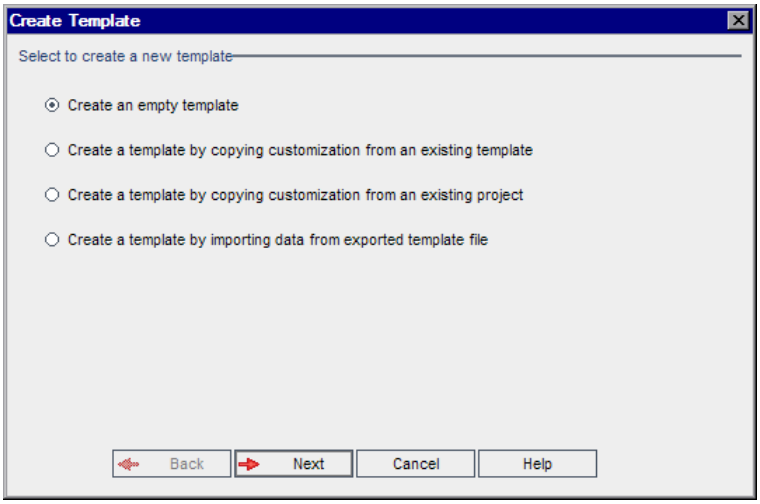
You can choose to link the newly created template to the project from which it is copied. This enables the template administrator to apply template customization changes to the linked project.

Note:

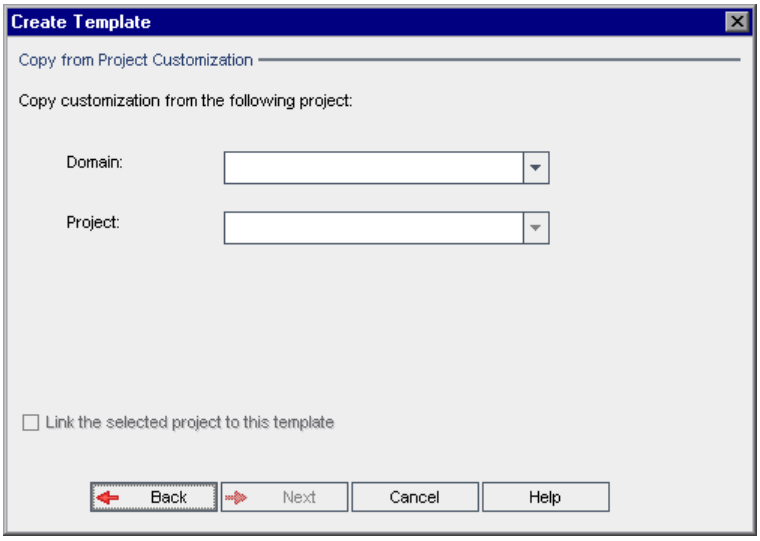
- If the project from which you create the template contains workflow scripts, the scripts must be converted after the template is created. This enables the template administrator to apply template workflow customization to linked projects. For more information, see HP Software Self-solve knowledge base article KM494331 (http://h20230.www2.hp.com/selfsolve/document/KM494331).
- **Version Control:** If you copy a version control enabled project, the new template is created with version control enabled.

To create a template from an existing project:

1. In Site Administration, click the **Site Projects** tab.
2. Select the domain where you want to create the template.
3. Click the **Create Template** button. The Create Template dialog box opens.



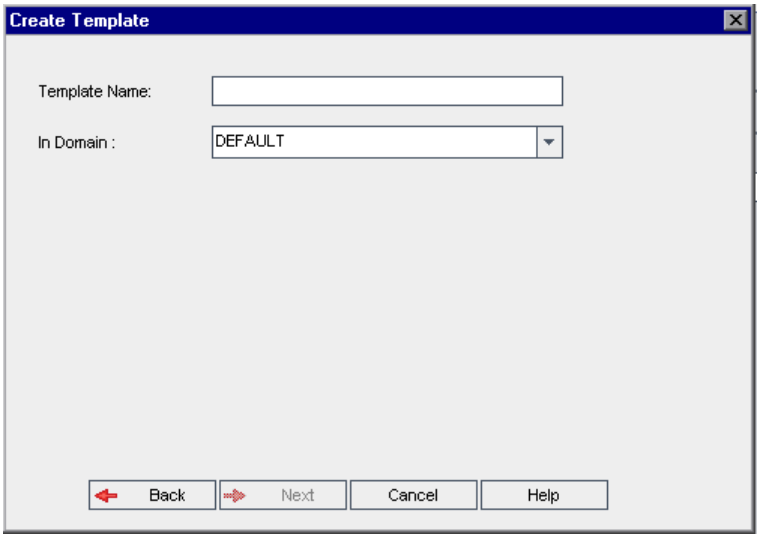
4. Select **Create a template by copying customization from an existing project** and click **Next**. The Copy from Project Customization dialog box opens.



5. In the **Domain** box, select the domain where the project you want to copy is located.
6. In the **Project** box, select the project you want to copy.
7. Select **Link the selected project to this template** to link the project to the newly created template. This enables the template administrator to apply template customization changes to the linked project.

   Note: After you link a project to a template, the template administrator must apply template customization to the project. This applies the customization from the template to the linked project, and sets the applied customization to read-only in the project. For more information, see "Applying Template Customization to Linked Projects" on page 337.

8. Click **Next**. The following dialog box opens.

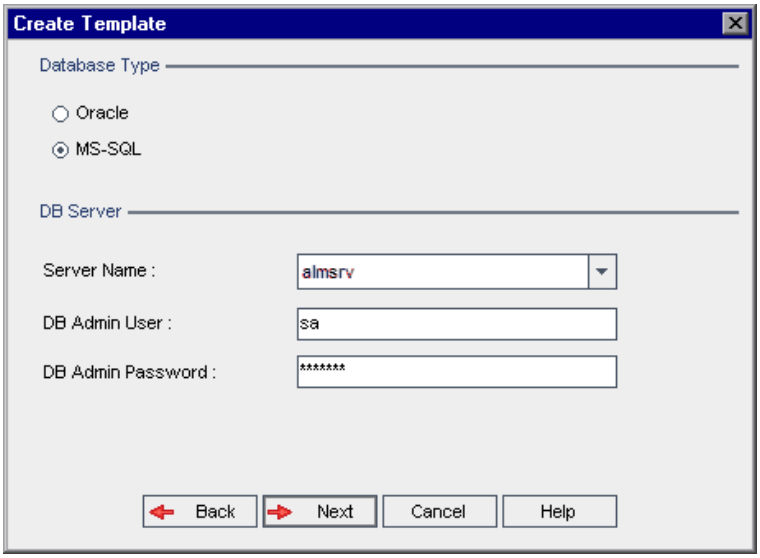


9. In the **Template Name** box, type a name for the template. The template name cannot be longer than 30 characters and cannot include any of the following characters: **= ~ ` ! @ # $ % ^ & * ( ) + | { } [ ] : ' ; " < > ? , . / \ -**

10. In the **In Domain** box, select a domain.

    Tip: After the template has been created, you can move it to a different domain in the Projects list using a drag-and-drop operation.

11. Click **Next**. The following dialog box opens.



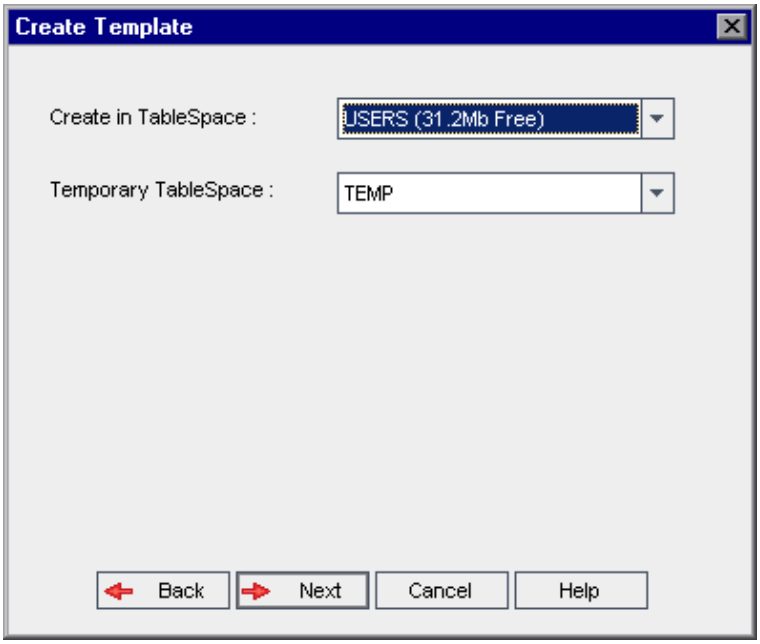
12. Under **Database Type**, select **Oracle** or **MS-SQL**.

    The default values defined for the domain are displayed for **Server Name**, **DB Admin User**, and **DB Admin Password**. If additional database servers are defined, you can select another name from the **Server Name** list.

13. Click **Next**.

    If your selected database server does not have the text search feature enabled, a message box opens. The message indicates that after this process completes, you can enable the text search feature. For more information on enabling the text search feature, see "Configuring Text Search" on page 163.

14. If you are creating a Microsoft SQL template, proceed to step **15**. For an Oracle template, the following dialog box opens.

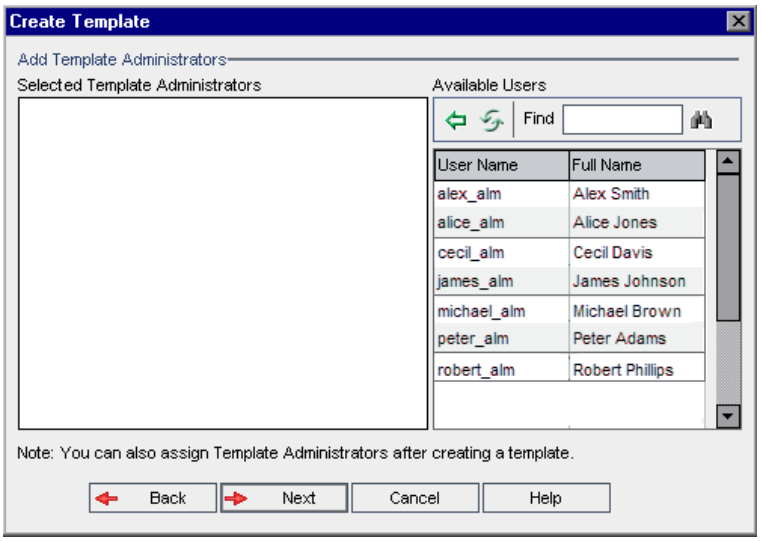


In the **Create in TableSpace** box, select a storage location that has sufficient space to store the new template. You should not use **UNDO** as the storage location.

In the **Temporary TableSpace** box, select a temporary storage location that has sufficient space to store the new template.

Click **Next**.

15. The Add Template Administrators dialog box opens.

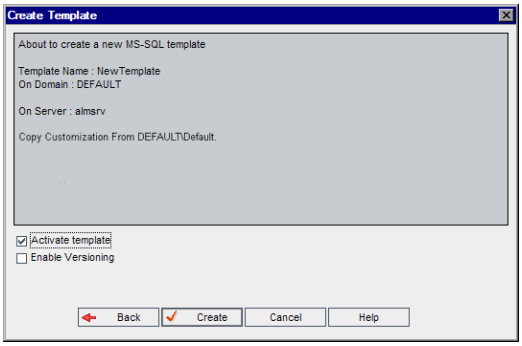


Selected Template Administrators lists users that are assigned as template administrators. **Available Users** lists users available in the template. When you assign template administrators, the users are moved from the Available Users list to the Selected Template Administrators list. Template administrator users can customize template projects and apply template customization to linked projects. For more information, see "Cross Project Customization" on page 332.

- **Refresh.** Click the **Refresh** button to refresh the list of available users.
- **Find.** Type the name of a user in the **Find** box, and click the **Find** button to search the Available Users list.
- **Add Selected Users.** Select the users you want to assign as template administrators, and click the **Add Selected Users** button. Alternatively, double-click a user name. The selected users are moved to the Selected Template Administrators list.
- **Delete.** To remove a user from the Selected Template Administrators list, right-click the user name and click **Delete**.

You can also assign template administrators after you have created the template. For more information, see "Assigning Project Administrators" on page 78.

16. Click **Next**. The following dialog box opens.



Verify the template details. To change any of the details, click **Back**.

17. Select **Activate template** to activate the template. Only activated templates are available in the HP Application Lifecycle Management Login window. For more information, see "Deactivating and Activating Projects" on page 91.
18. Click **Create**. The new template is added to the Projects list under **Template Projects**.

Importing a Template Project

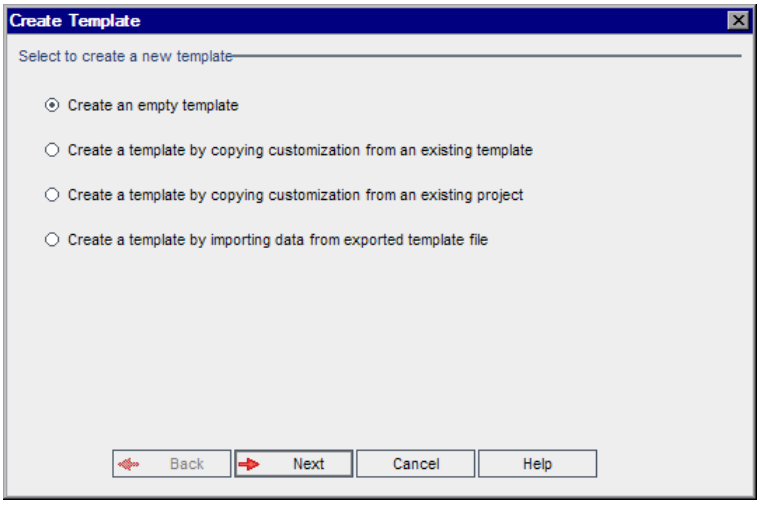
You can create a template project by importing data from an exported template project file created in the current version. For more information on exporting projects, see "Exporting Projects" on page 90.

If you import a template that was previously exported from the same server, ALM recognizes that the same template already exists on the server, based on the template ID. You can choose to replace the existing template, or cancel the import process. If you choose to replace the existing template when prompted, ALM overwrites the template, but does not overwrite connections to linked projects. The new template remains linked to the same projects.

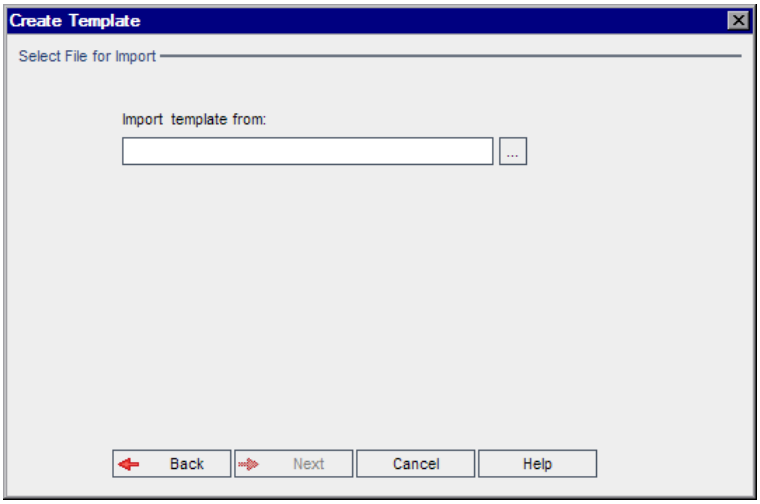
Note: The new template inherits the Unicode or ASCII definition of the imported template.

To import a template project:

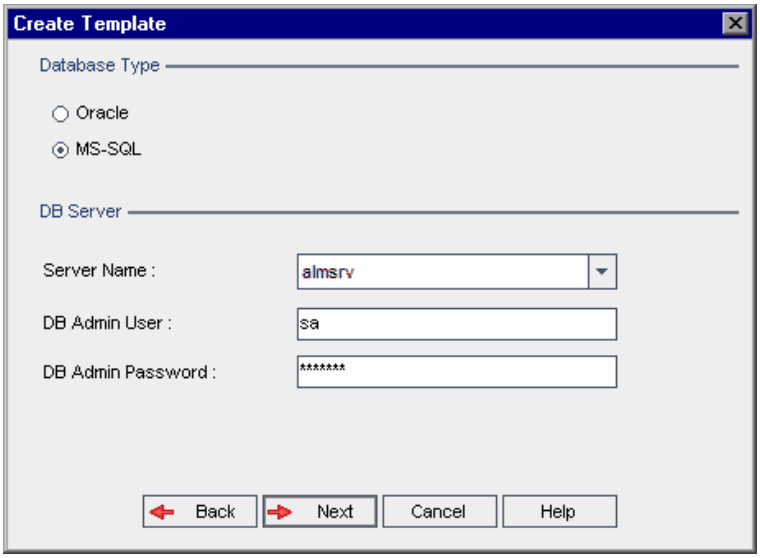
1. In Site Administration, click the **Site Projects** tab.
2. Select the domain where you want to create the template.
3. Click the **Create Template** button. The Create Template dialog box opens.



4. Select **Create a template by importing data from exported template file**. The Create Template: Select File for Import dialog box opens.



5. Click the browse button to the right of the **Import template from** box to locate the template project that you want to import. The Open dialog box opens.
6. Locate the directory and select the ALM Project Export file that you want to import. Click **Open**. The selected file is displayed in the **Import template from** box.
7. Click **Next**. The following dialog box opens.



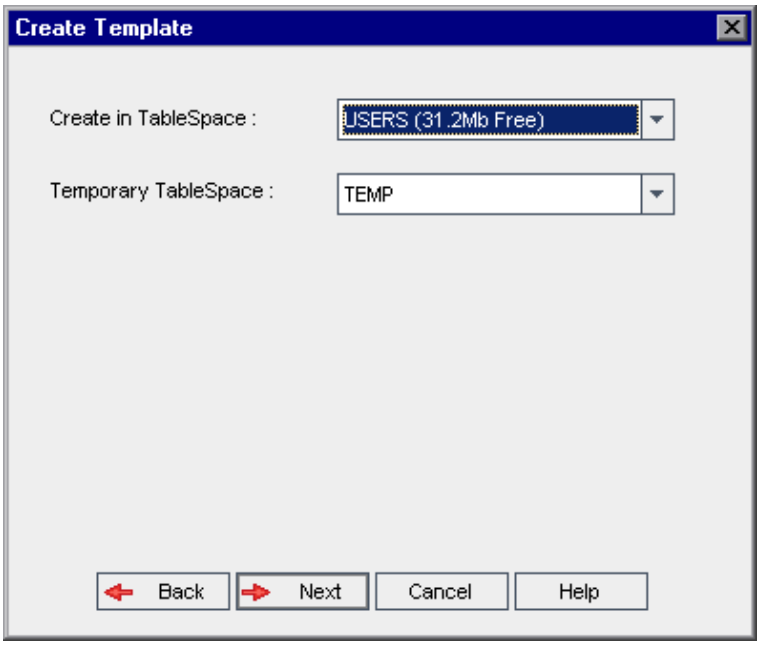
8. Under **Database Type**, select **Oracle** or **MS-SQL**.

   The default values defined for the domain are displayed for **Server Name**, **DB Admin User**, and **DB Admin Password**. If additional database servers are defined, you can select another name from the **Server Name** list.

9. Click **Next**.

   If your selected database server does not have the text search feature enabled, a message box opens. The message indicates that after this process completes, you can enable the text search feature. For more information on enabling the text search feature, see "Configuring Text Search" on page 163.

10. If you are creating a Microsoft SQL template, proceed to step **11**. For an Oracle template, the following dialog box opens.

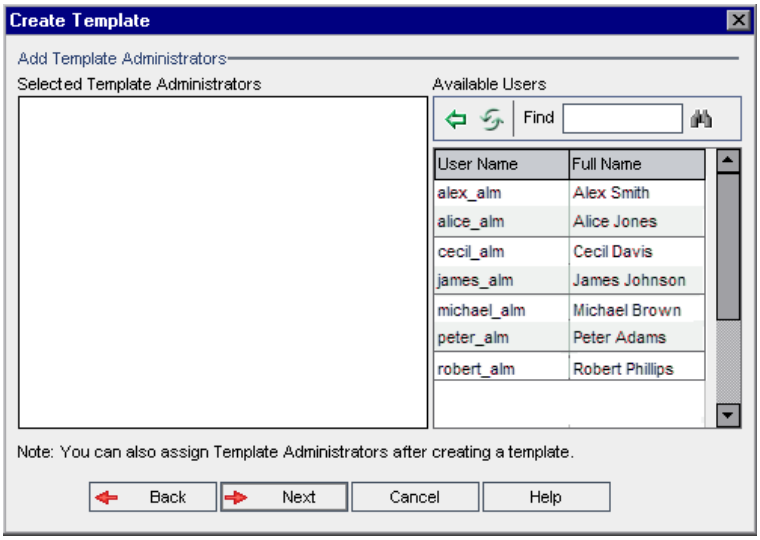


In the **Create in TableSpace** box, select a storage location that has sufficient space to store the new template. You should not use **UNDO** as the storage location.

In the **Temporary TableSpace** box, select a temporary storage location that has sufficient space to store the new template.

Click **Next**.

11. The Add Template Administrators dialog box opens.

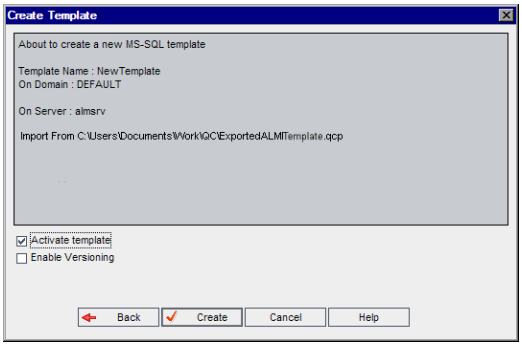


Selected Template Administrators lists users that are assigned as template administrators. **Available Users** lists users available in the template. When you assign template administrators, the users are moved from the Available Users list to the Selected Template Administrators list. Template administrator users can customize template projects and apply template customization to linked projects. For more information, see "Cross Project Customization" on page 332.

- **Refresh.** Click the **Refresh** button to refresh the list of available users.
- **Find.** Type the name of a user in the **Find** box, and click the **Find** button to search the Available Users list.
- **Add Selected Users.** Select the users you want to assign as template administrators, and click the **Add Selected Users** button. Alternatively, double-click a user name. The selected users are moved to the Selected Template Administrators list.
- **Delete.** To remove a user from the Selected Template Administrators list, right-click the user name and click **Delete**.

You can also assign template administrators after you have created the template. For more information, see "Assigning Project Administrators" on page 78.

12. Click **Next**. The following dialog box opens.



Verify the template details. To change any of the details, click **Back**.

13. Select **Activate template** to activate the template. Only activated templates are available in the HP Application Lifecycle Management Login window. For more information, see "Deactivating and Activating Projects" on page 91.
14. Select **Enable Versioning** to enable version control for the template. You can also enable version control after you have created the template. For more information, see "Enabling and Disabling Version Control for a Project" on page 91.
15. Click **Create**. The new template is added to the Projects list under **Template Projects**.

Linking a Template to Projects

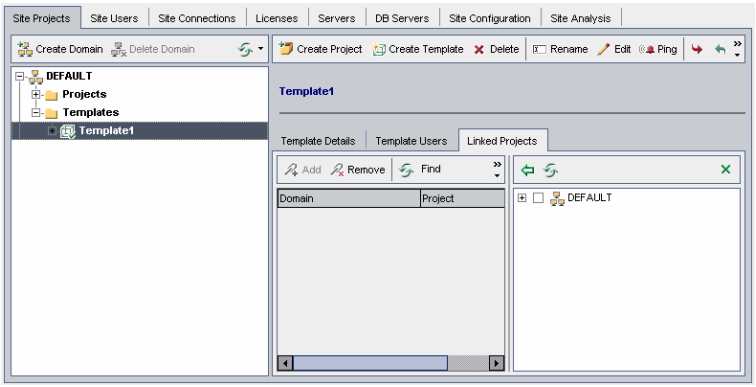
You link a template to projects as part of cross project customization. The template administrator uses cross project customization to apply template customization to the linked projects. You can link a template to multiple projects, but you can link a project to only one template. For more information, see "Cross Project Customization" on page 332.

Note: After you link a template to a project, the template administrator must apply template customization to the project. This applies the customization from the template to the linked project, and sets the applied customization to read-only in the project. For more information, see "Applying Template Customization to Linked Projects" on page 337.

You can also link a template to a project when you create a project. For more information, see "Creating Projects" on page 37. To link a template to a project when you create the template from an existing project, see "Creating a Template from an Existing Project" on page 64.

To link a template to projects:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a template project. In the right pane, click the **Linked Projects** tab. The Linked Projects list is displayed.
3. Click the **Add** button. The Projects list is displayed in the right pane.



4. Select projects from the Projects list, and click the **Add Selected Projects** button. The selected projects are displayed in the Linked Projects list.
5. You can search for a project in the Linked Projects list by typing the name of a project in the **Find** box, and clicking the **Find** button. You can also click on a column heading to change the sort order of the projects in the Linked Projects list.
6. To remove a project from a template, in the Linked Projects list, select the project. To remove more than one project, press the CTRL key and select the projects. Click **Remove**. Click **OK** to confirm. This removes the project from the Linked Projects list, and the project is no longer linked to the template.
7. To refresh the Linked Projects list or the Projects list, click the **Refresh** button above the appropriate list.

Updating Project Details

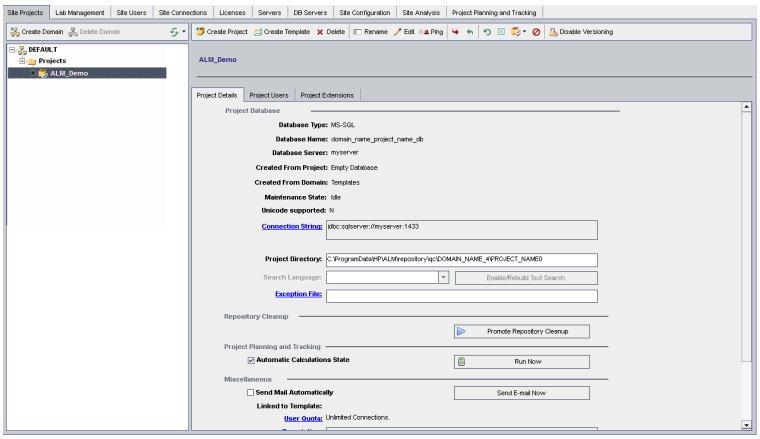
You can view project details such as database type and project directory from the Project Details tab. You can also edit various settings for the project. For example, you can edit the connection string, change the number of users allowed to connect concurrently to the project, and enable the automatic sending of defect email. Updated project details are written to the **dbid.xml** file, so that if a project is restored, the updated project data is used. For more information, see "Restoring Access to Projects" on page 97.

> **Tip:** You can move a project to a different domain in the Projects list using a drag-and-drop operation. This does not change the physical location of the project.

Cross Project Customization: If you are working with a template project, you update template details from the Template Details tab. **ALM Editions:** Template projects are not available for **Quality Center Enterprise Edition**.

To update project details:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a project. In the right pane, select the **Project Details** tab. The project's details are displayed.



Note: If a project is inactive, the project icon is displayed in red. To activate, see "Deactivating and Activating Projects" on page 91.

3. Under **Project Database**, view the following project details:

Field	Description
Database Type	The database type can be MS-SQL or Oracle.
Database Name	The project name, as defined in the database.
Database Server	The name of the database server on which the database is located.
Created From Project	The project was copied from this project. An **Empty Database** value indicates that the project was not copied. For more information, see "Copying Projects" on page 44.
Created From Template	The project was copied from this template.
Restored From Project	The project was restored from this project. For more information, see "Restoring Access to Projects" on page 97.
Created From Domain	The project was copied from this domain.
Restored From Domain	The project was restored from this domain. For more information, see "Restoring Access to Projects" on page 97.

Field	Description
Maintenance State	Indicates whether a maintenance task is being performed on this project. Tasks include verifying, repairing, upgrading, and realigning a project. Possible values are: ▪ **Idle.** No maintenance is being performed on this project. ▪ **Corrupted.** Maintenance cannot be completed because the project is corrupted. To resume, a backup copy of this project must be restored. ▪ **Under maintenance task.** Maintenance is being performed on this project. For more information on maintaining projects,see "Upgrading Projects to a New Version" on page 102.
Unicode Supported	Indicates whether this project supports Unicode.
Connection String	The connection string. To modify the connection string, see "Editing the Connection String" on page 97.
DB User Password	The user password for the Oracle server on which the database is located. To modify this password, see "Modifying Database Server Properties" on page 161.
Project Directory	The location of the project repository in the file system.
Search Language	Indicates the search languages for performing a text search. For more information, see "Selecting a Text Search Language for a Project" on page 164.
Exception File	Indicates the location of the exception file to be used when running the upgrade process. For more information, see "Upgrading Domains and Projects" on page 113.

4. Under **Repository Cleanup**, promote or delay the scheduled project repository cleanup. For more details on the project repository cleanup, see "Project Repository Cleanup" on page 83.

   Click the available button:

   - **Promote Repository Cleanup.** Instructs ALM to cleanup the repository of the current project as soon as possible.
   - **Postpone Repository Cleanup.** Instructs ALM to postpone the repository cleanup of the current project, or to stop a cleanup that is in progress.

5. Under **Project Planning and Tracking**, view the following project details:

Field	Description
Automatic Calculations State	Indicates whether the project is included in the automatic daily project planning and tracking calculations of your site. For more information, see "Enabling or Disabling Automatic Calculations for a Project" on page 205.
Run Now	Enables you to manually trigger project planning and tracking calculations for a project in order to refresh its results without waiting for the next scheduled calculation. For more information, see "Launching Calculations for a Project Manually" on page 206.

ALM Editions: Functionality related to project planning and tracking is available for ALM Edition only.

6. Under **Miscellaneous**, select **Send mail automatically** to enable the mail configuration settings for a project. This sends email to specified users every time set defect fields are updated. If this check box is not selected, mail configuration settings for the project have no effect and email is not sent. For more information on configuring mail, see "Configuring Automail" on page 314.

   The defect messages are sent automatically, at specified time intervals. You can edit the time interval using the "MAIL_INTERVAL" on page 169 parameter in the **Site Configuration** tab. You can also specify if you want the email to include attachments and/or history. For more information, see "Setting ALM Configuration Parameters" on page 167.

   To manually send the defect messages that have accumulated during the current time interval, click the **Send Email Now** button. If the **Send mail automatically** check box is not selected, defect messages do not accumulate, so this button has no effect.

7. If you enable the **Text Search** link in the **DB Servers** tab after you have added a project to the Projects list in the Site Projects tab (for example, after creating, upgrading, or migrating a project), you must also click the **Enable/Rebuild Text Search** button. For more information, see "Enabling Text Search in ALM" on page 163.

8. The **Linked to Template** field displays the name of the template that the project is linked to. For more information on linked templates, see "Updating Linked Template Details" on page 340.

9. To change the number of users allowed to connect concurrently to the project, click the **User Quota** link. The Project User Quota dialog box opens.

   Choose **Maximum connections** and type the maximum number of concurrent connections allowed. Click **OK**.

   Note: The maximum number of users allowed to connect concurrently to the project should not exceed the number of users allowed to connect to its domain. For more information, see "Creating Domains" on page 36.

10. To add a description for the project, click the **Description** link. In the Edit Project Description dialog box, type your description and click **OK**. By default, the project creation date is displayed.

11. Click the **Refresh Projects List** button to refresh the projects in the selected domain. To refresh projects in all domains, click the **Refresh Projects List** arrow and choose **Refresh All Domains**.
12. To assign users to a project, see "Assigning Users to Projects" below.

Assigning Users to Projects

As a site administrator, you can control access to projects or template projects by defining the users that can log on to the project. You can assign users to projects from the Users list, or copy users from existing projects. You can also assign users as project administrators. For more information on assigning project administrators, see "Assigning Project Administrators" on next page.

When a user is no longer working on a project, remove the user from the project to ensure project security. Removing a user from a project does not delete the user from the Users list. To remove the user from the Users list, you must delete the user from the Site Users tab, as described in "Deleting Users" on page 144.

Note:

- As a project administrator, you can assign and remove users from projects, and change user privileges from the Project Customization window. For more information, see "Managing Users in a Project" on page 248.
- You can assign projects to users from the Site Users tab. For more information, see "Assigning Projects to Users" on page 141.
- Automatic email notification is sent to project administrators when users are assigned or removed from a project in Site Administration. You can make automatic notification unavailable by adding the "AUTO_MAIL_USER_NOTIFICATION" on page 173 parameter in the Site Configuration tab.

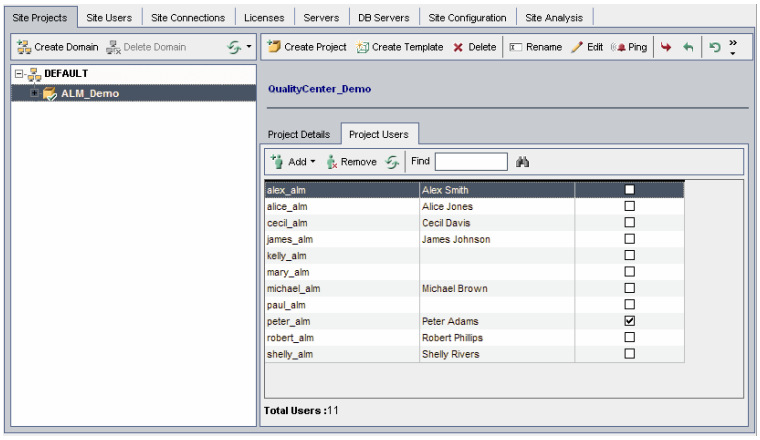
Cross Project Customization: If you are working with a template project, you assign users from the Template Users tab.

ALM Editions: Template projects are not available for **Quality Center Enterprise Edition**.

To assign users to a project:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a project. In the right pane, select the **Project Users** tab.

The users for the selected project are displayed.



You can click the **User Name** or **Full Name** column headers to change the sort order of user names or full names in the Project Users list from ascending to descending. You can also click the **Project Administrator** column header to group users by project administrators.

3. Click the **Add** button, and choose one of the following options:
   - **Add From The Users List.** The Users list is displayed to the right of the Project Users tab. Select the users that you want to assign to the project. You can search for users by typing a search string in the **Find** box above the Users list, and clicking the **Find** button .
   - **Copy From Another Project.** The Projects list is displayed to the right of the Project Users tab. To copy a user, click a project to expand the project directory, and select the user name check box. To copy all users from a project, select the project's check box. To clear all selected users, click **Clear All**.
4. Select users from the Users list or Projects list, and click the **Add Selected Users** button . Alternatively, double-click a user. The selected users are displayed in the Project Users list.
5. To remove a user from a project, select the user in the Project Users list and click the **Remove** button. Click **Yes** to confirm. The user is removed from the Project Users list.
6. To refresh the Project Users list or Users list, click the **Refresh** button above the appropriate list.

Assigning Project Administrators

After you add users to projects you can assign users as project administrators (belonging to the **TDAdmin** user group). Project administrators have full privileges in the project from the Project Customization window. For more information, see "Managing User Groups and Permissions" on page 252.

When you copy users from other projects, they are added with the same user group privileges they had in the project from which they were copied, provided the user group exists in this project. If the user group does not exist in this project, the users are added with **Viewer** group privileges. If you copy a user from another project in which the user is a project administrator, the user is automatically assigned as a project administrator in this project.

When you add users to the project from the Users list, those users are added with Viewer group privileges (read-only privileges).

Note: You can also assign project administrators when you create a new project. For more information, see "Creating Projects" on page 37.

Cross Project Customization: If you are working with a template project, you assign users as template administrators from the **Template Users** tab.

ALM Editions: Template projects are not available for **Quality Center Enterprise Edition**.

To assign Project Administrator privileges to a user:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a project. In the right pane, select the **Project Users** tab.
3. In the Project Users list, select the **Project Administrator** check box for each user you want to assign as a project administrator.
4. To remove a user from the Project Administrator group, clear the **Project Administrator** check box, and confirm you want to remove the user from the group.

Enabling Extensions for a Project

Extensions provide added functionality to ALM. If you have a license for an ALM extension, you can utilize its added functionality after enabling the extension on a per project basis.

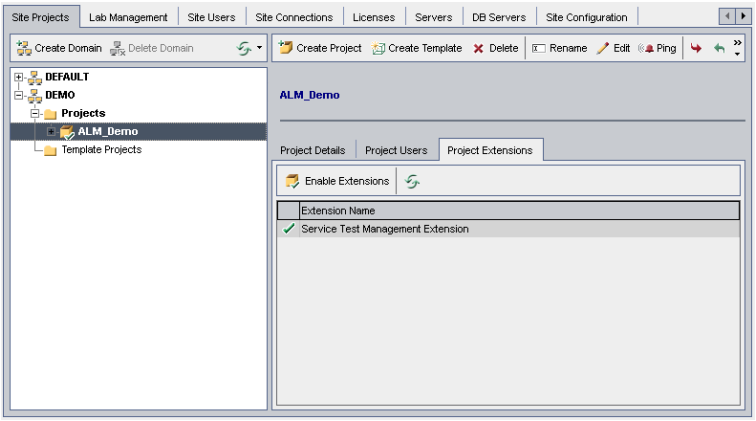
To view the list of extensions available with ALM 11.52, or to download documentation for ALM extensions, visit the Add-ins page, available from **Help > Add-ins**.

Cross Project Customization: If an extension is enabled for a template project, the extension must also be enabled for the template's linked projects. Linked projects can have additional extensions enabled. **ALM Editions:** Template projects are not available for **Quality Center Enterprise Edition**.

You can also enable extensions for a project when you create a project. For more information, see "Creating Projects" on page 37.

To enable extensions for a project:

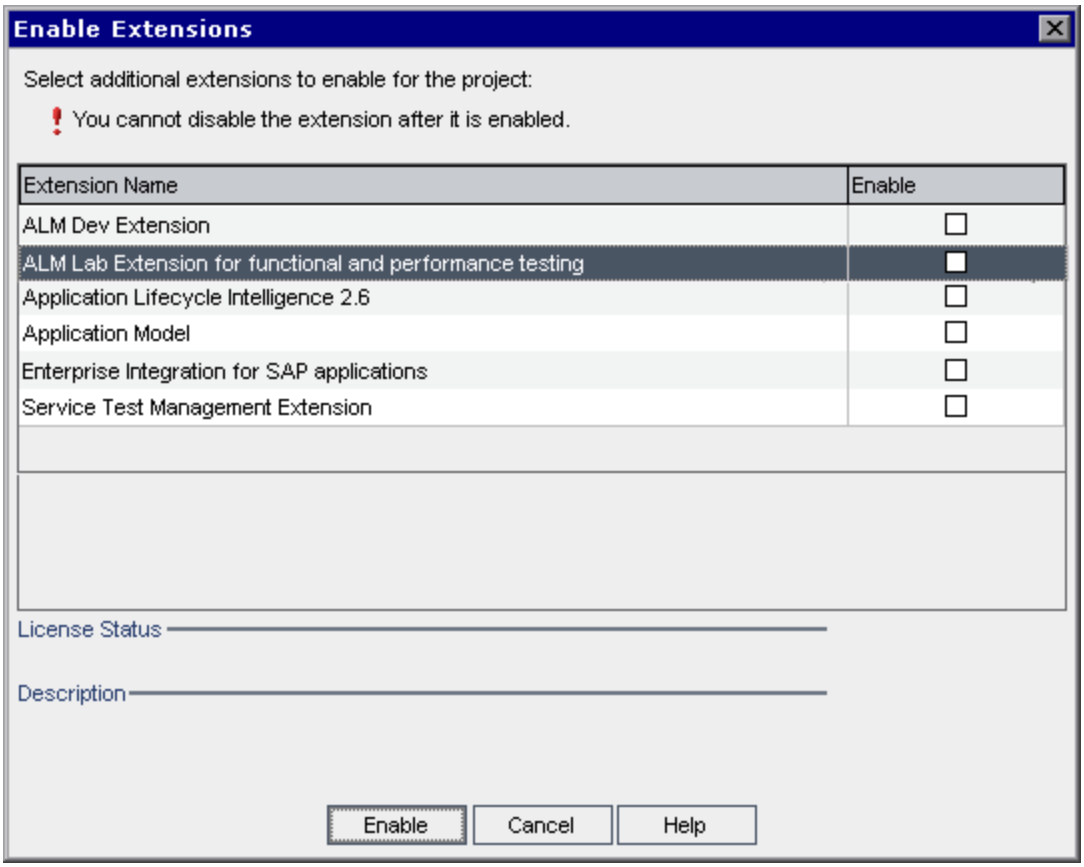
1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a project. In the right pane, click the **Project Extensions** tab.



The Extensions list displays the extensions that are enabled for the selected project.

3. To enable one or more extensions for the project, click the **Enable Extensions** button. The Enable Extensions dialog box opens, displaying the list of extensions available for your ALM edition on the ALM server.

> **Tip:** Extensions for which there is no license on the server are displayed in gray. It is possible to enable an extension for which you do not yet have a license. You can utilize the extension's added functionality later, after receiving the license.

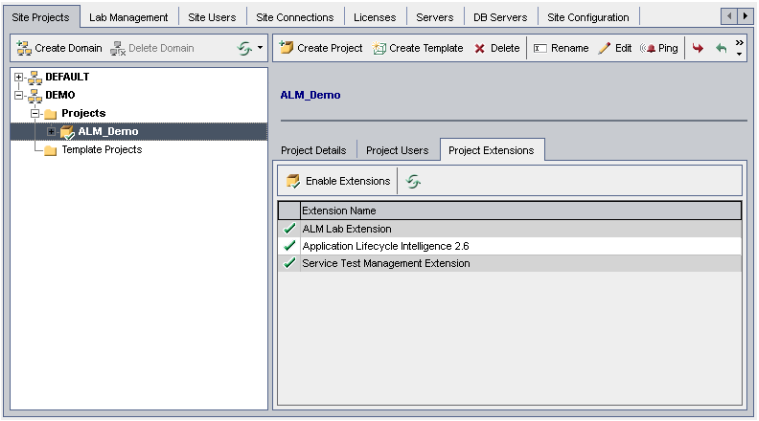


4. Select the **Enable** check box for the extensions you want to enable.

> **Note:** You cannot disable an extension for a project after you enable it. We recommend enabling only the extensions you need. Enabling extra extensions can affect performance and consumes additional disk space.

As you navigate from one extension to another, relevant license status information is displayed in the **License Status** section.

5. Click **Enable**. The selected extensions are enabled for the project, and the extension names are displayed in the Extensions list.



6. To refresh the Extensions list, click the **Refresh** button .

Chapter 3: Managing the Optimized Project Repository

ALM stores all project files in a project repository located under the **ProjRep** directory. The files in this directory are stored in an optimized folder structure that allows maximum storage space. In addition, any two files with identical content are stored only once in the **ProjRep** directory. For example, if you attach the same file to several ALM records, the file is stored only once in the project repository. This results in a significant reduction in disk space, and reduced time in copy operations.

This section includes:

Project Repository Cleanup 83

Browsing the Project Repository 83

Realigning Repositories 85

Project Repository Cleanup

When you add a file to an entity, ALM checks whether an identical file exists in the project repository. If an identical file is located, no physical file is added to the repository.

When you delete a file from an entity, the file is not immediately deleted from the project directory, as it may still be used by other entities.

At regular intervals, the project repository is scanned for obsolete files that are no longer referenced by any entity. If the files remain unreferenced for a specified period, they are removed from the project repository. These intervals are set by default at seven days each. You can configure the intervals using the site configuration parameters below.

You can promote or postpone a project's repository cleanup. For more details, see "Updating Project Details" on page 73.

You can define the following site configuration parameters to regulate the project repository cleanup process:

- **REPOSITORY_GC_PROJECT_CLEANUP_INTERVAL.** Defines the time interval between cleanup processes of each project repository. For details, see "REPOSITORY_GC_PROJECT_CLEANUP_INTERVAL" on page 190.
- **REPOSITORY_GC_DELAY_CANDIDATE_TIME.** Defines the time that elapses after obsolete files are detected in a scan, and before the obsolete files are removed. For details, see "REPOSITORY_GC_DELAY_CANDIDATE_TIME" on page 190.
- **REPOSITORY_GC_JOB_PRIORITY.** Defines the speed at which the cleanup process is performed. For details, see "REPOSITORY_GC_JOB_PRIORITY" on page 190.
- **SUSPEND_REPOSITORY_GC.** Enables you to stop the project repository cleanup process. For details, see "SUSPEND_REPOSITORY_GC" on page 194.

Browsing the Project Repository

You can browse and edit files in the project repository using an FTP client. Most standard FTP clients that support UTF-8 character encoding are compatible. The following clients are verified:

- Filezilla
- Total Commander Built-in FTP client
- FlashFXP

You can configure the FTP service for a secure connection. For more information, see "Enabling Secure FTP Connections" on next page.

Caution: Making changes to the folders, files, or file content directly in the **ProjRep** directory, not via an FTP client, will cause irreversible damage to the project repository.

When making changes to repository files via an FTP client, consider the following:

- It is safe to edit repository files via an FTP client. This will not damage the integrity of the optimized repository.

- Deleting a file will result in missing content in ALM entities.
- Renaming folders and files may result in missing content in ALM entities.

To browse the project repository:

1. **Start the FTP service.** Add and configure the **FTP_PORT** parameter in the **Site Configuration** tab. For more information, see "FTP_PORT" on page 182.
2. **Connect to the FTP server.** In an FTP client, use the following connection values:

Field	Value
Host	The name or IP of the ALM server to which you want to connect.
Port	The FTP port. This should be the same value that you used for the FTP_PORT site parameter.
User	The username of an ALM site administrator.
Password	The password of the ALM site administrator.

3. **Browse and edit repository files.** After you connect to the FTP service, a list of the site domains is displayed. Select a domain, and then select a project. The FTP client displays the project repository directories.

Enabling Secure FTP Connections

You can connect to the FTP server using a secure connection.

To enable a secure FTP connection:

1. On the HP Application Lifecycle Management server machine, generate a **keystore** file. Using the command line, go to **C:\Program Files\HP\HP Application LifeCycle Management 11.50\java\bin**. Type `keytool-genkey-keystore keystore.jks` to run the keystore utility, then follow the instructions.
2. Create an XML file, using the following format:

```
<ssl>
<keystore file="<keystore file path>" password="<keystore password>"/>
</ssl>
```

   where `<keystore file path>` is the directory and file name of the keystore file, and `<keystore password>` is the password you defined for the keystore.

   Save the XML file as **sslkeystore.xml**.
3. Place the **sslkeystore.xml** file in the following directory: **C:\ProgramData\HP\ALM\webapps\qcbin**.
4. Restart the FTP server by restarting the ALM service, or reconfiguring the **FTP_PORT** site parameter.
5. In the FTP client, select the FTPS or SSL option. If you are using FileZilla, go to **File > Site**

Manager and click **New Site**. For **Protocol**, select FTP-File Transfer Protocol, and for **Encryption**, select Require implicit FTP over TLs, and click connect.

Realigning Repositories

ALM project repositories consist of the physical files that are stored in the file system, and database tables that index the files. When you create a backup of a project that is active, the time gap between the database and file system backups can cause discrepancies between the database file index and the physical files.

For details on backup,see "Backing Up Projects" on page 118.

If you restore a project for online work from a backup that was performed while the project was active, you must realign the file system and database tables.

The realigning process performs the following actions:

- If a file is indexed in the database but is not located in the file system, the index of the file is removed from the database.
- If a file is stored in the file system but is not indexed in the database, the file is deleted from the file system.

In addition, the realigning process verifies the integrity of the relationship between the logical and physical database tables.

If an irrecoverable problem is detected, the project **Maintenance State** is changed to `Corrupted`. Check the log for details, and examine the database tables.

By default, the realigning process runs in non-silent mode. When running the process in non-silent mode, ALM may pause and prompt you for input when an error occurs. Instead, you can choose to run the process in silent mode. When an error occurs, ALM will abort the process without prompting you for input.

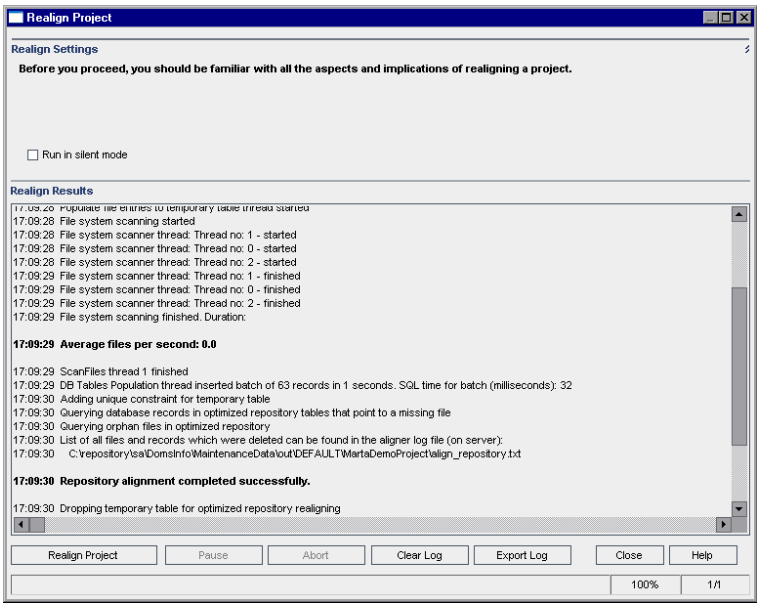
Note: Projects are deactivated while the realignment is in progress, and activated again after the realignment is complete.

Realigning a Project

This section describes how to realign a single project.

To realign a project:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a project.
3. Click the **Maintain Project** button and choose **Realign Repository**. The Realign Project dialog box opens.



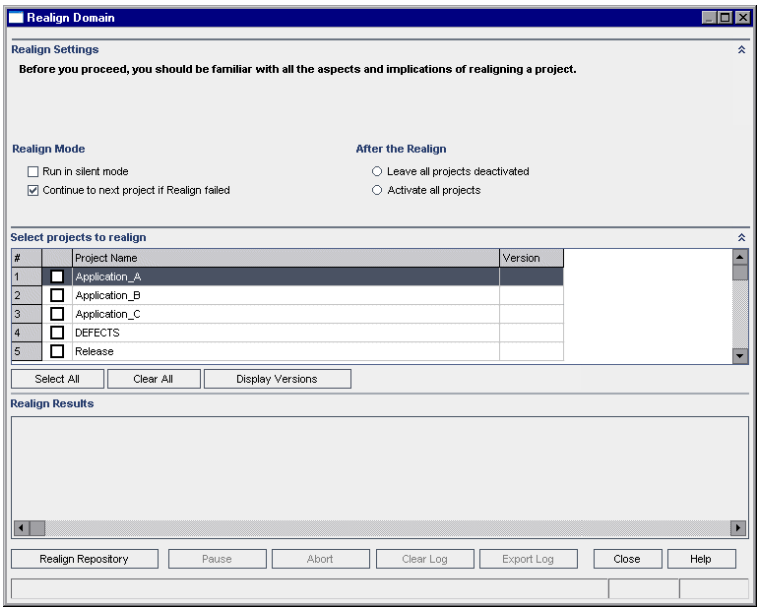
4. To run the realigning process without any user interaction, select **Run in Silent Mode**.
5. To start the realigning process, click the **Realign Project** button. If the project is active, you are prompted to deactivate it. For more information, see "Deactivating and Activating Projects" on page 91.
6. To save the messages displayed in the Realign Results pane to a text file, click the **Export Log** button. In the Export Log to File dialog box, choose a location and type a name for the file. Click **Save**.
7. To clear the messages displayed in the Realign Results pane, click the **Clear Log** button.
8. Click **Close** to close the Realign Project dialog box.

Realigning a Domain

This section describes how to realign all projects in a domain.

To realign a domain:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a domain.
3. Click the **Maintain Domain** button and choose **Realign Repository**. The Realign Domain dialog box opens.



4. In the **Realign Settings** area, under **Realign Mode**, you can select the following options:
   - **Run in Silent Mode.** Runs the process without any user interaction.
   - **Continue to next project if realign failed.** Proceeds to the next project if the realign process fails. This is the default option.
5. In the **Realign Settings** area, under **After the Realign**, you can select one of the following options:
   - **Leave all projects deactivated.** Leaves all projects deactivated after the realign process completes.
   - **Activate all projects.** Activates all projects after the realign process completes.
6. To view the current version numbers of your projects, select the project names, or click **Select All** to view version numbers for all projects. Click the **Display Versions** button.

   The project version number is displayed in the **Version** column.
7. To realign your projects, select the project names, or click **Select All** to realign all projects. Click the **Realign Repository** button.
8. To save the messages displayed in the Realign Results pane in a text file, click the **Export Log** button. In the Export Log to File dialog box, choose a location and type a name for the file. Click **Save**.
9. To clear the messages displayed in the Realign Results pane, click the **Clear Log** button.
10. Click **Close** to close the Realign Domain dialog box.

Chapter 4: Managing Projects

Site Administration enables you to manage and maintain HP Application Lifecycle Management (ALM) domains and projects.

This chapter includes:

About Managing Projects 89

Querying Project Tables 89

Exporting Projects 90

Deactivating and Activating Projects 91

Enabling and Disabling Version Control for a Project 91

Pinging Projects 92

Renaming Projects 92

Converting Projects to Unicode 93

Removing Projects 95

Deleting Projects 96

Deleting Domains 96

Editing the Connection String 97

Restoring Access to Projects 97

Renaming the Defects Module for a Project 99

Limiting Records Displayed in Grids 100

About Managing Projects

You manage ALM projects and template projects using Site Administration. After you create a project, you can export the project, query the contents of the project by defining and running SQL statements, deactivate/activate access to the project, and enable or disable version control for the project. You can also remove a project, and restore access to an existing project.

Note: Users who are already logged into **Lab Management** need to re-enter the application to see changes you made in Site Administration.

For more information on creating projects, see "Creating Projects" on page 34.

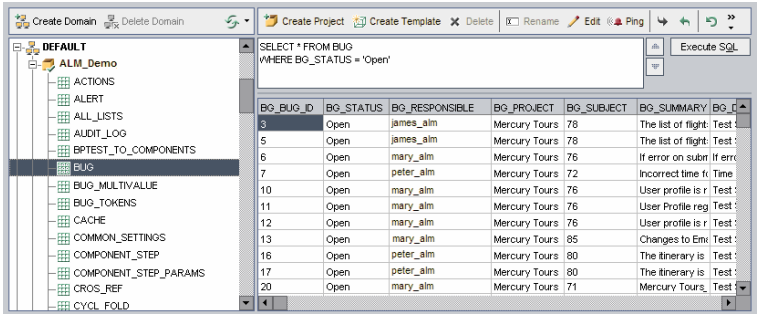
ALM Editions: ALM template projects are not available for **Quality Center Enterprise Edition**.

Querying Project Tables

You can query specific data that is stored in your project or template project. You query a project by defining and running SQL queries. The following examples show SQL queries and the results that they return.

Query	Results
`select * from BUG where BG_STATUS = 'Open'`	All defects that are open.
`select * from BUG where BG_RESPONSIBLE = 'james_alm' or BG_RESPONSIBLE = 'mary_alm'`	All defects assigned to either James or Mary.
`select count (*) from BUG where BG_RESPONSIBLE = 'mary_alm'`	The number of defects assigned to Mary.
`select * from BUG where BG_RESPONSIBLE='james_alm' and BG_STATUS='open'`	All open defects assigned to James.

Using the first query example, the SQL query returns the following:



To query a project:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, double-click a project.
3. Select a table. ALM automatically runs the "SELECT *" query for this table and displays all the data for the table in the SQL Query Results grid.
4. Define a query by typing an SQL statement in the SQL pane.

   To navigate back to your previous SQL statement in the SQL pane, click the **Up** button .

   To navigate forward to your next SQL statement in the SQL pane, click the **Down** button .
5. Click the **Execute SQL** button. The data returned by the query appears in the SQL Query Results grid.

Tip: To export query results, your database administrator can run the same queries on the project database and export the results for you. For details, see "Exporting Projects" below.

Exporting Projects

Exporting ALM projects or template projects enables you to take project data from an ALM server, and back it up to another location or another media device. For example, you may want to create self-contained project image files that are backed up on a USB storage device or DVD. You can send the media device to an ALM server in another location, and import the project files. When you export a project file, it is saved and exported in ZIP format.

Before exporting a project, consider the following guidelines:

- If you export an ALM project that has extensions installed, all data from the project is exported, including data for the extensions. You can only import such an exported project to a server that has the relevant extensions installed.
- You can only import ALM project files created in the same ALM version. For more information on importing projects, see "Importing Projects" on page 50.
- The project database schema and project file system repository combined should not exceed 4 gigabytes.
- There must be adequate disk space available in your home directory on the ALM client machine to temporarily store the exported project file, even if you choose an alternate location for saving the file.
- You will not be able to import the project onto the original server, if a project with the same PUID exists on that server.
- If the project was not part of Lab Management, when restoring access to the project:
  - Details about test runs will not be included in the Usage Reports.
  - Timeslot information and project settings information are lost.

To export a project:

1. In Site Administration, click the **Site Projects** tab.

2. In the Projects list, select a project, and click the **Export Project to Project File** or **Export Template to Project File** button . Alternatively, right-click the project and choose **Export Project** or **Export Template**. If the project is active, you are prompted to deactivate it. For more information, see "Deactivating and Activating Projects" below.
3. The Save As dialog box opens. Select the directory where you want to save the project data. Type a name for the project in the **File name** box. By default, the data is saved as an ALM Project Export file (**.qcp**).
4. Click **Save** to save the project data as an ALM Project Export file.

Deactivating and Activating Projects

You can deactivate or activate a project or template project. When you deactivate a project, the project name is removed from the **Projects** box in the ALM Login window. The project is not deleted from the server. Any users currently connected to the project are forced to log out when you deactivate.

Note: It is recommended that you deactivate a project before you change any data that may cause inconsistency for connected users.

To deactivate a project:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a project.
3. Click the **Deactivate Project** or **Deactivate Template** button . A message box indicates that all connected users will be disconnected.
4. Click **OK** to confirm. The project is deactivated and the project icon is changed in the Projects list.

To activate a project:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a project.
3. Click the **Activate Project** or **Activate Template** button . The project is activated and the project icon is changed in the Projects list.

Enabling and Disabling Version Control for a Project

You can enable version control for a project or template project. For more information on version control, refer to the *HP Application Lifecycle Management User Guide*.

You can also disable version control for a project. When you disable version control for a project, ALM no longer stores previous versions, and deletes all version history for the project. If you enable version control for the project again, previous history is not available.

Note: After enabling version control for a project, you should review all its workflow scripts and make adjustments for each checked in entity. This includes the following entities: **Req**, **Test**, **Resource**, and **Component**. For each checked in entity that includes a **Post** function in its script, you must modify the script. To modify, add a **Checkout** function before every **Post** function. Making this modification prevents the Check Out dialog box from opening each time a call to a **Post** function is made. For more information, see "Workflow Event Reference" on page 428.

To enable version control for a project:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a project.
3. Click the **Enable Versioning** button .
4. If the project is active, click **Yes** to deactivate it. Click **OK** to confirm.
5. When the process completes, click **OK**. Version control is enabled. ALM displays a lock icon next to the project name in the Projects list.

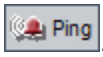
To disable version control for a project:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a project.
3. Click the **Disable Versioning** button .
4. If the project is active, click **Yes** to deactivate it. Click **OK** to confirm.
5. A message displays, indicating that when you disable version control, ALM deletes all version history. Click **OK** to confirm.
6. Click **Yes** to disable version control. Version control is disabled. ALM removes the lock icon next to the project name in the Projects list.

Pinging Projects

You can check whether a project database or template project database is accessible from Site Administration.

To ping a project:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a project.
3. Click the **Ping Project** or **Ping Template** button 

.
4. Click **OK** when prompted with a message that the ping was successful.

Renaming Projects

You can rename a project or template project in the Projects list.

To rename a project:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a project.
3. Click the **Rename Project** or **Rename Template** button. If the project is active, you are prompted to deactivate it. For more information, see "Deactivating and Activating Projects" on page 91.
4. In the Rename Project dialog box, type the new name for the project and click **OK**.

   The project name cannot be longer than 30 characters and can include only letters, digits, and underscores.

   Note: Support of non-English national characters depends on the database settings used by the server. We do not recommend using non-English national characters in project names.

   The project is renamed in the Projects list.

Converting Projects to Unicode

This section describes how to convert projects to Unicode. You can select a specific project to convert, or you can select a domain and convert its projects to Unicode.

Unicode is a feature of MS-SQL that allows multi-language support. In Oracle, multi-language support is defined when installing the server.

You can convert the following types of projects to Unicode:

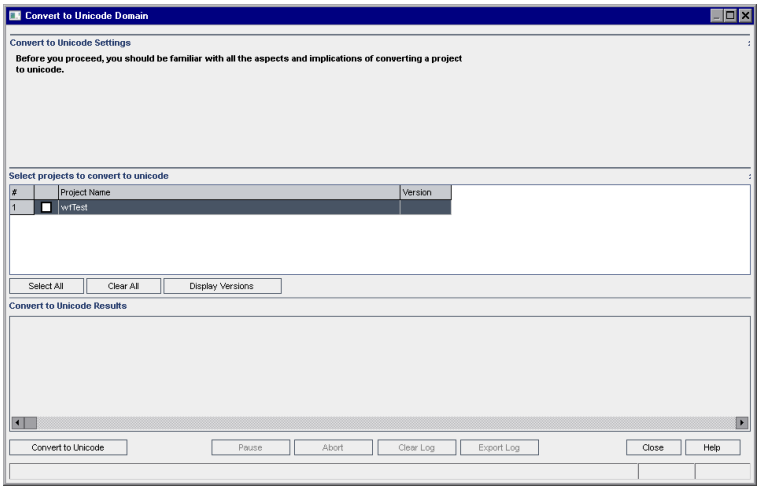
- Projects that are version 11.5 or above.
- Projects created on an MS SQL server.
- Projects created in ASCII.

Caution: Unicode increases the memory required for the database server.

To convert projects in a domain to Unicode:

1. In Site Administration, click the **Site Projects** tab and select the domain whose projects you want to convert.
2. Click the **Maintain Domain** button and select **Convert to Unicode in Domain**. The Convert to Unicode Domain dialog box opens.

   Note: Convert to Unicode in Domain is only available if projects that can be converted to Unicode exist in the selected domain.



3. Select the project or projects you want to convert and click **Convert to Unicode**. Under Convert to Unicode Results, a log is displayed.

   Tip: Click **Display Versions** to display the versions of the selected projects.

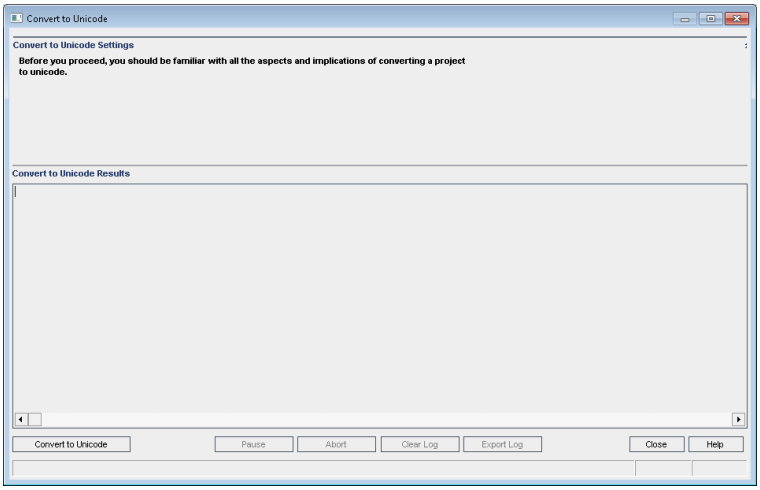
   Note: Projects with columns whose length is greater than 4000 cannot be converted to Unicode. The validation process fails and an error message is generated.

   To fix the problem, either reduce the column length to 4000 or add the columns to the alm_i18n_exceptions.xml exception file before converting. System defined columns and system defined user fields must be reduced and cannot be added to the exception file.

4. To pause the conversion process, click the **Pause** button. To continue, click the **Resume** button.
5. To abort the conversion process, click the **Abort** button. Click **Yes** to confirm.
6. After the process has completed successfully or stopped due to failure, you can select one of the following:
   - **Clear Log.** Clears the text displayed in the Convert to Unicode Results area. This disables the **Export Log** button.
   - **Export Log.** Enables you to save the log as a **.txt** file. In the Export Log to File dialog box, select a location and type a name for the file. Click **Save**.
7. Click **Close** to close the Convert to Unicode Domain dialog box.

To convert a project to Unicode:

1. In Site Administration, click the **Site Projects** tab and select the project you want to convert.
2. Click the **Maintain Project** button and select **Convert Project to Unicode**. The Convert to Unicode dialog box opens.



3. Click **Convert to Unicode**. Under Convert to Unicode Results, a log is displayed.

   Note: Projects with columns whose length is greater than 4000 cannot be converted to Unicode. The validation process fails and an error message is generated.

   To fix the problem, either reduce the column length to 4000 or add the columns to the alm_i18n_exceptions.xml exception file before converting. System defined columns and system defined user fields must be reduced and cannot be added to the exception file.

4. To pause the conversion process, click the **Pause** button. To continue, click the **Resume** button.
5. To abort the conversion process, click the **Abort** button. Click **Yes** to confirm.
6. After the process has completed successfully or stopped due to failure, you can select one of the following:
   - **Clear Log.** Clears the text displayed in the Convert to Unicode Results area. This disables the **Export Log** button.
   - **Export Log.** Enables you to save the log as a **.txt** file. In the Export Log to File dialog box, select a location and type a name for the file. Click **Save**.
7. Click **Close** to close the Convert to Unicode dialog box.

Removing Projects

You can remove a project or a template project from the Projects list in Site Administration. This does not delete the project from the server and you can restore the project if necessary. For more information on restoring access to a project, see "Restoring Access to Projects" on page 97.

Note: If the project is currently in use, it cannot be removed. For information about how to manually remove a project, see HP Software Self-solve knowledge base article KM1457081 (http://h20230.www2.hp.com/selfsolve/document/KM1457081).

To remove a project from the Projects list:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a project.
3. Click the **Remove Project** or **Remove Template** button .
4. Click **OK** to confirm. If the project is still active, you are prompted to deactivate it. For more information, see "Deactivating and Activating Projects" on page 91.
5. Click **OK**.

Deleting Projects

You can delete a project or template project from the Projects list in Site Administration. This deletes the contents of the project from the server and you cannot restore the project.

Note: If the project is currently in use, it cannot be deleted. For information about how to manually delete a project, see HP Software Self-solve knowledge base article KM1457081 (http://h20230.www2.hp.com/selfsolve/document/KM1457081).

To delete a project:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a project.
3. Click the **Delete Project** or **Delete Template** button.
4. Click **OK** to confirm. If there are active users connected to the project, you are prompted to disconnect them.

   The Database Admin Password dialog box opens. If you did not specify a database administrator user name or password, enter the database administrator's user name and password and click **OK**. If you previously specified a database administrator user name or password, these credentials are already entered in the dialog box.
5. Click **OK**.

Deleting Domains

You can delete a domain. It is removed from the Projects list, and its contents are deleted from the server.

Note: You cannot delete a domain if it contains projects or template projects. To delete the domain, you must first delete the projects. For more information, see "Deleting Projects" above.

To delete a domain:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a domain.

3. Click the **Delete Domain** button.
4. Click **Yes** to confirm.

Editing the Connection String

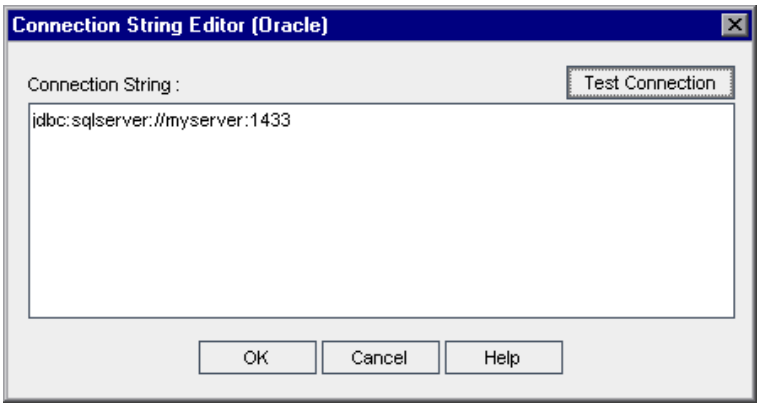
You can edit a project's or a template project's connection string.

To edit the connection string:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a project.
3. Click the **Edit Connection String** button

or the **Connection String** link. If the project is still active, you are prompted to deactivate it. For more information, see "Deactivating and Activating Projects" on page 91.

   The Connection String Editor dialog box opens.



4. In the **Connection String** box, modify the attributes of the connection string, such as the database server name and port number.
5. To test the connection string, click **Test Connection**. In the Ping Database Server dialog box, type the database administrator user name and password and click **OK**. If the connection is successful, a confirmation message displays. Otherwise, an error message displays.
6. Click **OK** to save your connection string modification and close the Connection String Editor.

Restoring Access to Projects

You can restore access to an ALM project or template project that is not in your current Projects list in Site Administration. For example, you may want to access a project from another server. After you restore access to a project, it is added to the Projects list in Site Administration.

Note:

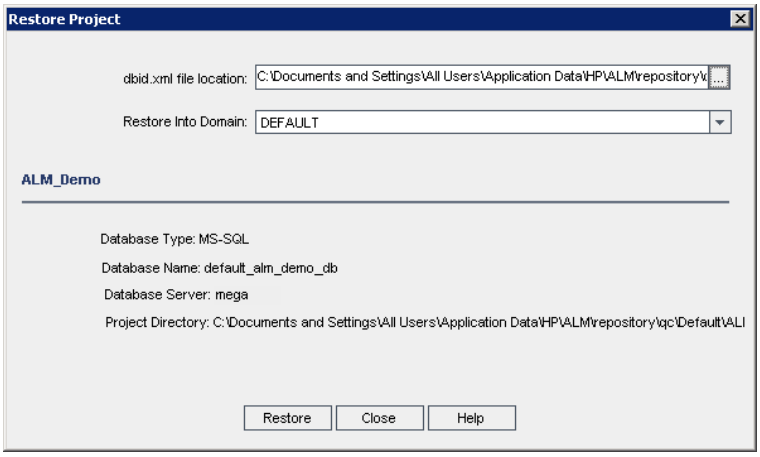
- Before restoring the project, make sure that the database where the project resides exists in the **DB Servers** tab in Site Administration on your ALM server. The ALM server needs to access the contents of the restored project from the project's database. For more

information, see "Upgrading Projects to a New Version" on page 102.

- When restoring a project, you should select the **dbid.xml** file located in the project repository. This ensures that the project retains its original ID. If a project does not have its original ID, the following cross project features may not function properly: cross project customization, importing and synchronizing libraries, and cross project graphs.
- If you are restoring your project from a different directory, or if you renamed your schema or restored it to a different database, you must update the **dbid.xml** file accordingly. For details, see "Updating the dbid.xml file" on next page.
- If you were previously working with **Performance Center 11.00**, you must first restore and upgrade LAB_PROJECT, and then any Performance Center template projects, before restoring and upgrading other Performance Center projects. For details, see "Restoring LAB_PROJECT" on page 120.

To restore access to an ALM project:

1. In Site Administration, click the **Site Projects** tab.
2. Click the **Restore Project** or **Restore Template** button . The Restore Project dialog box opens.
3. To locate the file that includes the project that you want to restore, click the browse button to the right of the **dbid.xml file location** box. The Open File dialog box opens.
4. Locate the file. For information on the location of the **dbid.xml** file, see "Understanding the Project Structure" on page 35.
5. Select the **dbid.xml** file and click **Open**. The Restore Project dialog box opens and displays the database type, name, server, and the directory path of the project.



6. In the **Restore Into Domain** box, select the domain in which you want the restored project to be located.
7. Click **Restore**.
8. If your database server does not have the text search feature enabled, a message box opens. You can enable the text search feature before or after this process completes.

- Click **Yes** to continue this process. After the process completes, you can enable the text search feature.
- Click **No** to stop this process. Enable the text search feature and then restart the process.

For more information on enabling the text search feature, see "Configuring Text Search" on page 163.

9. When the restore process completes, click **OK**.
10. Click **Close** to close the Restore Project dialog box and view the restored project in the Projects list.

Updating the dbid.xml file

If you are restoring your project from a different directory, or if you renamed your schema or restored it to a different database, for example, as part of the upgrade process, you must update the following values:

- **DB_NAME**. Update to the database schema name as it appears in the database server.
- **DB_CONNSTR_FORMAT**. Update if the project is restored on a new database server.
- **DBSERVER_NAME**. This is the name of the database server as defined in the **DB Servers** tab in Site Administration.
- **DB_USER_PASS**. Update if the encrypted passphrase differs between ALM 11.00 and ALM 11.52. If you are upgrading from QC 10.00 this value is not relevant.
- **PHYSICAL_DIRECTORY**. Update to the new location of the project repository.It must contain a backslash (\) at the end of the path.

Note:

- In order to identify the values of **DB_CONNSTR_FORMAT** and **DB_USER_PASS**, it is recommended to create a new, empty project in ALM 11.52 Site Administration, open the project's **dbid.xml** file, and copy these values. You can later delete the empty project.
- Make sure not to copy/paste or change the value for **PR_SMART_REPOSITORY_ENABLED**.
- If you are restoring **LAB_PROJECT** or Performance Center projects as part of the upgrade process, make sure not to edit the **PROJECT_UID** value. You must restore these projects with their original **PROJECT_UID** value in order to maintain the links between **LAB_PROJECT** and its associated Performance Center projects, important for shared data, such as timeslots, runs, and so on.

Renaming the Defects Module for a Project

You can rename the Defects module for a specific project or template project. For example, you can change the name of the Defects module from `Defects` to `Bugs`. You rename the Defects module by adding a parameter to the **DATACONST** table of the project. For more information on modifying project tables, see "Querying Project Tables" on page 89.

Note: You can rename any ALM module for all your projects by adding the "REPLACE_TITLE"

on page 189 parameter in the **Site Configuration** tab.

To rename the Defects module for a project:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, double-click the project for which you want to rename the Defects module.
3. Select the **DATACONST** table.
4. In the SQL pane, type an SQL INSERT INTO statement to insert a row into the table with the following values:
   - In the **DC_CONST_NAME** column, insert the parameter name REPLACE_TITLE.
   - In the **DC_VALUE** column, insert a string that defines the new name for the Defects module, in the following format:

     ```
     original title [singular];new title [singular];original title [plural];new title [plural]
     ```

     For example, to change the name of the module from `Defects` to `Bugs`, type the following SQL statement into the SQL pane:

     ```
     insert into dataconst values ('REPLACE_TITLE', 'Defect;Bug;Defects;Bugs')
     ```
5. Click the **Execute SQL** button. The new row is added to the **DATACONST** table. The ALM project displays the new Defects module name.

Limiting Records Displayed in Grids

To optimize performance, the number of records retrieved and displayed in ALM grids is limited. The limits include:

- Maximum number of records displayed in a grid.
- Maximum number of records displayed for each group when a **group by** filter is applied to a grid.

To bypass the limit and display all relevant records, users can click the **Retrieve all <x> results** link in the window or dialog box.

You can change the default limits for all site projects, or individually for each project. When you change the limits for a project, they override default limits, or the values defined by the "FETCH_LIMIT" on page 180 or "GROUP_FETCH_LIMIT" on page 183 parameters.

To change the default number of records displayed in grids for all site projects:

Add and configure the "FETCH_LIMIT" on page 180 and "GROUP_FETCH_LIMIT" on page 183 parameters in the **Site Configuration** tab.

To change the default number of records displayed in grids per project:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, double-click the project for which you want to change the default limits for

records displayed in a grid.

3. Select the **DATACONST** table.
4. In the SQL pane, type an SQL INSERT statement to insert a row into the table with the following values:
   - In the **DC_CONST_NAME** column, insert the parameter name FETCH_LIMIT or GROUP_FETCH_LIMIT.
   - In the **DC_VALUE** column, insert a value for the parameter.

   For example, to change the value of the FETCH_LIMIT parameter to 50, type the following SQL statement into the SQL pane:

   ```
   insert into dataconst values ('FETCH_LIMIT', '50')
   ```
5. Click the **Execute SQL** button. The new row is added to the **DATACONST** table.

Chapter 5: Upgrading Projects to a New Version

To work in HP Application Lifecycle Management (ALM) 11.52 with projects created in previous Quality Center and ALM versions, you must upgrade your projects to align them with the required configurations of the current version of ALM.

Performance Center: To work with projects created in Performance Center versions 9.52 or earlier, you must migrate your projects to align them with the required configurations of ALM. For details, see the *HP ALM Performance Center Installation Guide*.

This chapter includes:

About Upgrading Projects 103
Understanding the Version/Patch Numbering Scheme 103
About Upgrading Projects for Major and Minor Versions 103
About Upgrading Projects for Minor-Minor Versions 126

About Upgrading Projects

This section describes the process required for performing:

- Full upgrades of:
  - ALM projects from previous major and minor versions
  - Performance Center projects from ALM/Performance Center 11.00
- Upgrades of ALM projects from previous ALM 11.5*x* minor-minor versions

For details on understanding whether a version is major, minor, or minor-minor, see the information about understanding version numbering in "Understanding the Version/Patch Numbering Scheme" below.

This chapter includes:

- "Understanding the Version/Patch Numbering Scheme" below
- "About Upgrading Projects for Major and Minor Versions" below
- "About Upgrading Projects for Minor-Minor Versions" on page 126

Understanding the Version/Patch Numbering Scheme

Version numbering follows the format:

`(Major)(Major).(Minor)(Minor-Minor).(Build)(Build)(Build)(Build)`

Examples

Major version 11: `11.00.0000`

Minor version 11.5: `11.50.0000`

Minor-minor version 11.52: `11.52.0000`

Patch 3 for major version 11: `11.00.0003`

For details on this format, see the information about the Obsolescence Policy on the HP Support Site. To access the site from within ALM, choose **Help > HP Software Support**. Alternatively, access the URL: www.hp.com/go/hpsoftwaresupport.

About Upgrading Projects for Major and Minor Versions

This section describes the process required for working with:

- Quality Center or ALM projects from previous versions
- Performance Center projects from ALM/Performance Center 11.00

Before upgrading, verify and repair your projects to detect and correct errors in your database user schema and data.

Note:

- **Upgrade Methodology:** To upgrade from a previous Quality Center version with the minimum possible interruption to your system operations, you should be familiar with the considerations and recommendations involved in the upgrade process. For information on upgrade methodology, see the *HP Application Lifecycle Management Upgrade Best Practices Guide*.
- **Product Feature Movie:** To view a movie that presents the methodologies for upgrading to HP ALM from a previous version, choose **Help > Movies** in the ALM main window.
- This process describes upgrading projects for major and minor version upgrades. For details on understanding whether a version is major or minor, see the information about understanding version numbering in the "Understanding the Version/Patch Numbering Scheme" on previous page.
- For details on minor-minor version upgrades, see "About Upgrading Projects for Minor-Minor Versions" on page 126.

This section includes:

- "Upgrade Versions" below
- "Project Upgrade Considerations" on next page
- "Repository Migration Considerations" on next page
- "Upgrade Steps" on page 106

Upgrade Versions

The following table describes how to upgrade projects from previous Quality Center and ALM versions:

From version:	To ALM 11.52:
Quality Center 10.00 and ALM 11.00	Upgrade projects directly to ALM 11.52.
Performance Center 11.00	Upgrade projects directly to ALM 11.52. **Note**: You must first upgrade LAB_PROJECT, and then any Performance Center template projects, before upgrading Performance Center projects.
Performance Center versions 9.52 and earlier	Projects must be migrated to align them with the required configurations of ALM, using the HP ALM Performance Center 11.50 Migration Installation wizard. For details, see the *HP ALM Performance Center Installation Guide*.

From version:	To ALM 11.52:
Quality Center 9.2	Projects must first be upgraded to Quality Center 10.00 or ALM 11.00.
Quality Center 9.0	Projects must first be upgraded to Quality Center 10.00.

Note: If you are upgrading to HP Application Lifecycle Management 11.50 from Quality Center 10.00, 9.2, or 9.0, you must convert the rich text mechanism from Microsoft Word to HTML. For more information, see the HP Self-Solve knowledge base article KM1116588 http://h20230.www2.hp.com/selfsolve/document/KM1116588. (Requires HP Passport sign-in credentials)

Project Upgrade Considerations

Review the following before you upgrade projects to ALM 11.52:

- **Version Control:**
  - **Upgrading Quality Center 10.00 and ALM 11.00 version control enabled projects.** Version control enabled projects from Quality Center 10.00 or ALM 11.00 (including Performance Center projects) cannot be upgraded to ALM 11.52 while there are checked out entities. All entities must be checked in in the corresponding version of Quality Center or ALM.
  - **Upgrading from legacy version control projects.** To work with projects from Quality Center 9.0 or Quality Center 9.2 that use version control, you must first upgrade to Quality Center 10.00, migrate legacy version control data, and then upgrade to ALM 11.52. For information on migrating legacy version control data to Quality Center 10.00, see http://h20230.www2.hp.com/selfsolve/document/KM632120.
- **Server Locales:** Before upgrading a project, ensure that the system locale on the ALM server, database, and file servers, all match.
- **Performance Center**: If you were working with Performance Center 11.00**,** before upgrading other Performance Center projects, you must first upgrade LAB_PROJECT, and then any Performance Center **template** projects. For details, see "Restoring LAB_PROJECT" on page 120.

Repository Migration Considerations

When upgrading projects from Quality Center 10.00, the project file repositories are migrated to a new optimized folder structure. Review the following to ensure that the file repositories are migrated successfully to the new structure:

- Allow for approximately twenty percent more space on the database.
- Before running the upgrade tool, make sure that all project files are saved in the default project directory.

  To find out whether project files, such as tests and test resources, are outside the default project directory, log in to Site Administration. In the **Site Projects** tab, expand each project and click

the **DATACONST** table. For each of the ***_directory** entries in the **DC_CONST_NAME** column, verify that the corresponding **DC_VALUE** is a folder name in the default project directory, and not a path to another directory. For example, verify that the **DC_VALUE** for `tests_directory` is set as **tests**, and the **DC_VALUE** for `resources_directory` is set as **resources**.

- Move any folders or files from the project repository that are not connected to ALM (for example, backup folders, and scripts that are not part of workflow) to a directory outside the repository. Any unrelated files remaining in the repository will be accessible via FTP only after the migration is completed. For more details, see "Browsing the Project Repository" on page 83.
- Make sure the ALM server has full permissions to the file server.
- For information on configuring the resources allocated to building the index, see http://h20230.www2.hp.com/selfsolve/document/KM862600.
- Make sure there is no folder on the file system named **ProjRep**.
- During and after the repository migration, no direct access to the file system is allowed. After the migration is completed, you can browse and edit the optimized file system using an FTP client. For more details, see "Browsing the Project Repository" on page 83.
- To backup a project during and after the repository migration,follow the guidelines in http://h20230.www2.hp.com/selfsolve/document/KM1373517.
- When restoring a project to the new version of ALM, be sure not to change the **<PR_SMART_REPOSITORY_ENABLED>** property in the dbid.xml file.
- The space saving and performance benefits of the optimized repository are realized only after the repository is fully migrated. Therefore, you should resolve any errors or warnings that occur during the migration, and ensure the migration is complete.
- If the ALM, database or file servers are over WAN, the migration process takes many times longer than over LAN.

Upgrade Steps

The upgrade workflow consists of the following steps:

1. **Verify project.** Detects problems in your environment, schema structure, and data integrity that could cause the project upgrade to fail.

   The verification process generates a report which indicates problems that can be repaired by ALM and problems that you should repair manually. For more information, see "Verifying Domains and Projects" on next page.

2. **Repair project.** Fixes data and schema issues found by the verification process. If the verification process finds problems that can cause data loss, the repair process does not fix them automatically. You need to repair these problems manually. For more information, see "Repairing Domains and Projects" on page 110.

   Before you start the repair process, you should back up your project. For more information, see "Backing Up Projects" on page 118.

   In the event that the repair fails, you must restore backed up projects before trying the repair process again. For more information, see "Restoring Projects" on page 119.

3. **Upgrade project.** Upgrades your project to the current version of ALM. For more information,

see "Upgrading Domains and Projects" on page 113.

Before you upgrade the project, back up your project. For information, see "Backing Up Projects" on page 118.

In the event that the upgrade fails, you must restore backed-up projects before trying the upgrade process again. For information, see "Restoring Projects" on page 119.

4. **Manage project repository migration.** After upgrading a project from Quality Center 10.00 to ALM 11.52, ALM migrates the project repository directories to a new file structure in the default project repository location. If the migration process fails, you must fix the problems manually in the project repository. You can also configure the speed at which the migration is performed. For more information, see "Migrating the Repository" on page 121.

For detailed information on the problems detected and fixed by the verification and repair processes, and help with repairing problems that cannot be fixed by ALM, see "Upgrade Preparation Troubleshooting" on page 494.

Verifying Domains and Projects

Before you upgrade a project, you run the verification process to check the correctness of your database user schema and data. Although your database user schema and data may be correct for your previous Quality Center or ALM version, they may not be consistent with the specifications for the current version of ALM.

The verification process detects problems in your environment, settings, schema structure, and data integrity that could cause the upgrade to fail. It generates a verification report which alerts you to problems that can be repaired by ALM and problems that you should manually repair.

By default, the verification report is saved on the ALM server machine. To change this default location, see "VERIFY_REPORT_FOLDER" on page 195.

After the project has been verified, you can still use it with a previous version of Quality Center or ALM.

For detailed information on the problems detected by the verification process, see "Upgrade Preparation Troubleshooting" on page 494.

You can define an exception file to instruct ALM to ignore errors detected while running the verification, repair, or upgrade process. For more information, see "Defining an Exception File" on page 116.

This section includes:

Verifying a Project 107

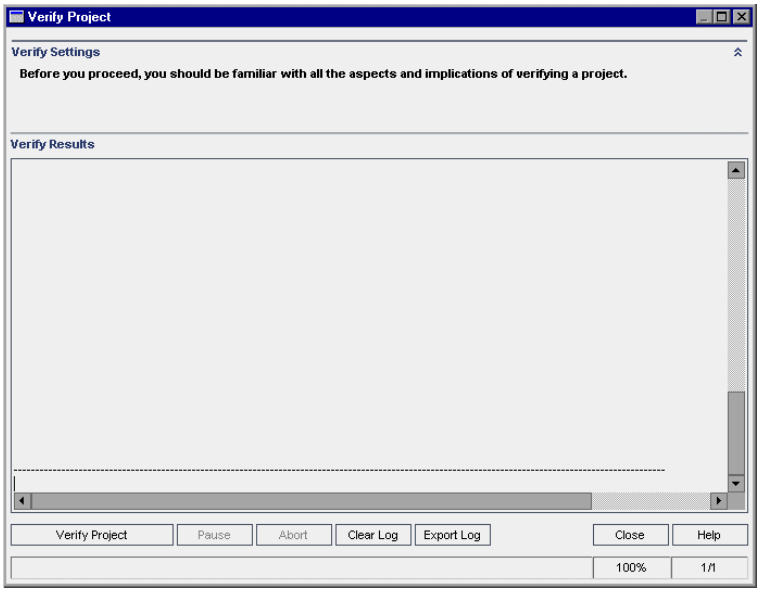
Verifying a Domain 108

Verifying a Project

This section describes how to verify a single project.

To verify a project:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a project.
3. Click the **Maintain Project** button and choose **Verify Project**. The Verify Project dialog box opens.



4. Click the **Verify Project** button to start the verification process. In the Verify Results pane, log messages are displayed.

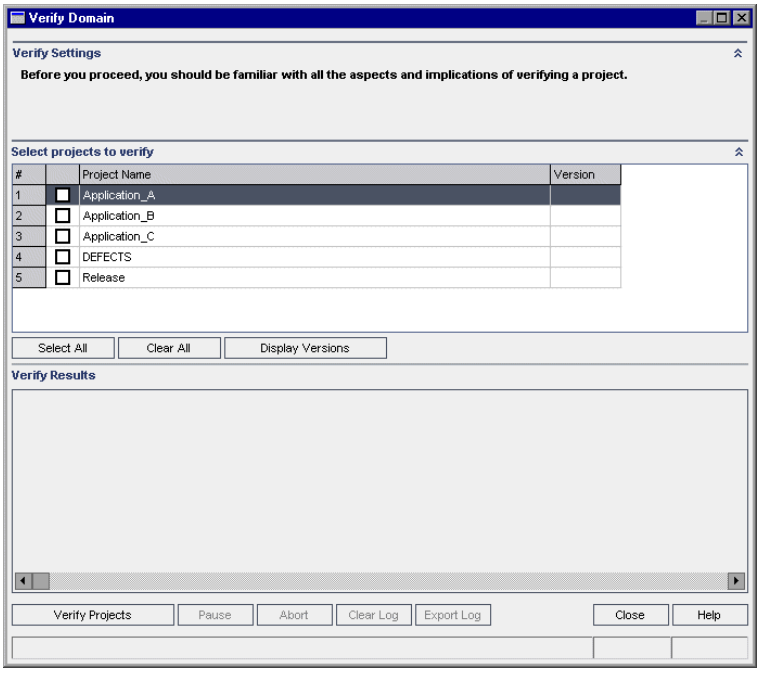
   If an error occurs while running the process, a message box opens. Click the **Abort** or **Retry** buttons accordingly.
5. To pause the verification process, click the **Pause** button. To continue, click the **Resume** button.
6. To abort the verification process, click the **Abort** button. Click **Yes** to confirm.
7. To save the messages displayed in the Verify Results pane to a text file, click the **Export Log** button. In the Export Log to File dialog box, choose a location and type a name for the file. Click **Save**.
8. To clear the messages displayed in the Verify Results pane, click the **Clear Log** button.
9. When the verification process completes, the Verify Results pane displays the location of the verification report. By default, the file is located in the following directory: **<ALM Repository Path>\sa\DomsInfo\MaintenanceData\out\<Domain Name>\<Project Name>**.
10. Analyze the verification report. The report indicates both problems that can be repaired by ALM automatically, and the problems that you need to repair manually.
11. Click **Close** to close the Verify Project dialog box.

Verifying a Domain

This section describes how to verify all projects in a domain.

To verify a domain:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a domain.
3. Click the **Maintain Domain** button and choose **Verify Domain**. The Verify Domain dialog box opens.



4. To view the current version numbers of your projects, select the project names, or click **Select All** to view version numbers for all projects. Click the **Display Versions** button.

   The project version number is displayed in the **Version** column.

5. To verify your projects, select the project names, or click **Select All** to verify all projects. Click the **Verify Projects** button.

   If an error occurs while running the process, a message box opens. Click the **Abort** or **Retry** buttons accordingly.

6. To pause the verification process, click the **Pause** button. To continue, click the **Resume** button.
7. To abort the verification process, click the **Abort** button. Click **Yes** to confirm.
8. To save the messages displayed in the Verify Results pane to a text file, click the **Export Log** button. In the Export Log to File dialog box, choose the location and type the name for the file. Click **Save**.
9. To clear the messages displayed in the Verify Results pane, click the **Clear Log** button.
10. When the verification process completes, the Verify Results pane displays the location of each verification report. By default, the files are located in the following directory: **<ALM Repository Path>\repository\sa\DomsInfo\MaintenanceData\out\<Domain Name>\<Project Name>**.
11. Analyze the verification report. The report indicates problems that can be repaired by ALM and the problems that you need to repair manually.

12. Click **Close** to close the Verify Domain dialog box.

Repairing Domains and Projects

The repair process fixes most data and schema issues found by the verification process. If the verification process finds problems that can cause data loss, the repair process does not fix these automatically. You need to repair these problems manually. To find out whether a particular issue is handled automatically or manually, refer to the verification report.

By default, the repair process runs in non-silent mode. When running the process in non-silent mode, ALM may pause and prompt you for input when an error occurs. Instead, you can choose to run the process in silent mode. When an error occurs, ALM aborts the process without prompting you for input.

After the project has been repaired, you can still use it with a previous Quality Center or ALM version.

For detailed information on the problems fixed by the repair process, and help with repairing problems that cannot be fixed by ALM, see "Upgrade Preparation Troubleshooting" on page 494.

This section includes:

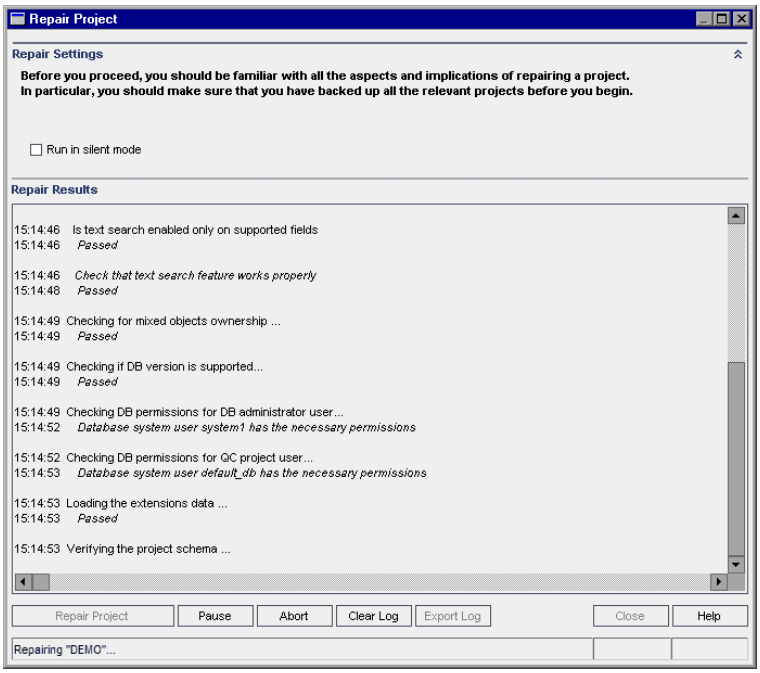
Repairing a Project 110
Repairing a Domain 111

Repairing a Project

This section describes how to repair a single project.

To repair a project:

1. Back up your project. For more information, see "Backing Up Projects" on page 118.
2. Repair problems that cannot be fixed by ALM, as indicated in your verification report (see Step **9** of "Verifying a Project" on page 107).
3. In Site Administration, click the **Site Projects** tab.
4. In the Projects list, select a project.
5. Click the **Maintain Project** button and choose **Repair Project**. The Repair Project dialog box opens.



6. To run the repair process without any user interaction, select **Run in Silent Mode**.
7. To start the repair process, click the **Repair Project** button. If the project is active, you are prompted to deactivate it. For more information, see "Deactivating and Activating Projects" on page 91.

   If an error occurs while running the process in non-silent mode, a message box opens. Click the **Abort** or **Retry** buttons accordingly.
8. To pause the repair process, click the **Pause** button. To continue, click the **Resume** button.
9. To abort the repair process, click the **Abort** button. Click **Yes** to confirm.
10. To save the messages displayed in the Repair Results pane to a text file, click the **Export Log** button. In the Export Log to File dialog box, choose a location and type a name for the file. Click **Save**.
11. To clear the messages displayed in the Repair Results pane, click the **Clear Log** button.
12. Click **Close** to close the Repair Project dialog box.

Repairing a Domain

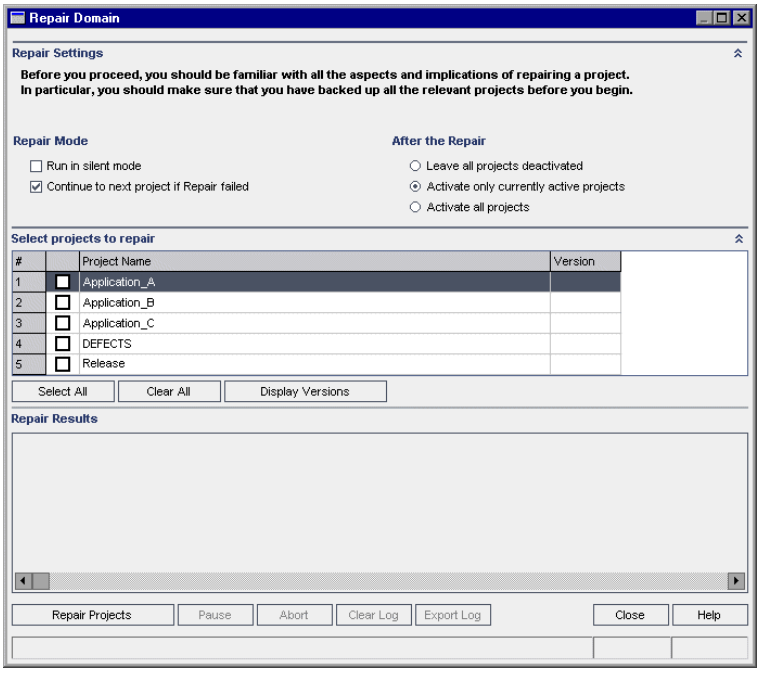
This section describes how to repair all projects in a domain.

To repair a domain:

1. Back up your projects. For more information, see "Backing Up Projects" on page 118.
2. Repair problems that cannot be fixed by ALM, as indicated in your verification report (see Step **10** of "Verifying a Domain" on page 108).
3. In Site Administration, click the **Site Projects** tab.

4. In the Projects list, select a domain.

5. Click the **Maintain Domain** button and choose **Repair Domain**. The Repair Domain dialog box opens.



6. In the **Repair Settings** area, under **Repair Mode**, you can select the following options:
   - **Run in Silent Mode.** Runs the process without any user interaction.
   - **Continue to next project if repair failed.** Proceeds to the next project if the repair process fails. This is the default option.

7. In the **Repair Settings** area, under **After the Repair**, you can select one of the following options:
   - **Leave all projects deactivated.** Leaves all projects deactivated after the repair process completes.
   - **Activate only currently active projects.** Reactivates previously-activated projects after the repair process completes. This is the default option.
   - **Activate all projects.** Activates all projects after the repair process completes.

8. To view the current version numbers of your projects, select the project names, or click **Select All** to view version numbers for all projects. Click the **Display Versions** button.

   The project version number is displayed in the **Version** column.

9. To repair your projects, select the project names, or click **Select All** to verify all projects. Click the **Repair Projects** button.

   If an error occurs while running the process in non-silent mode, a message box opens. Click the **Abort** or **Retry** buttons accordingly.

10. To pause the repair process, click the **Pause** button. To continue, click the **Resume** button.

11. To abort the repair process, click the **Abort** button. Click **Yes** to confirm.
12. To save the messages displayed in the Repair Results pane in a text file, click the **Export Log** button. In the Export Log to File dialog box, choose a location and type a name for the file. Click **Save**.
13. To clear the messages displayed in the Repair Results pane, click the **Clear Log** button.
14. Click **Close** to close the Repair Domain dialog box.

Upgrading Domains and Projects

After the project has been verified and repaired, you can proceed to upgrade your project to the current version of ALM.

For more information on the overall upgrade process, including prerequisites, see "About Upgrading Projects" on page 103.

By default, the upgrade process runs in non-silent mode. When running the process in non-silent mode, ALM may pause and prompt you for input when an error occurs. Instead, you can choose to run the process in silent mode. When running the process in silent mode, ALM aborts the process without prompting you for input.

After the project has been upgraded, you can no longer use the project with a previous version of Quality Center.

Note:

- During the upgrade process, the project directory must be accessible. For example, if your project directory is located on a file server, ensure that the server is running.
- **Version Control:** Version control enabled projects from Quality Center 10.00 or ALM 11.00 cannot be upgraded to ALM 11.52 while there are checked out entities. All entities must be checked in in the corresponding version of Quality Center or ALM.
- **Performance Center**: If you were working with Performance Center 11.00**,** before upgrading other Performance Center projects, you must first upgrade LAB_PROJECT, and then any Performance Center **template** projects, For details on LAB_PROJECT, refer to the *HP ALM Lab Management Guide*.

This section includes:

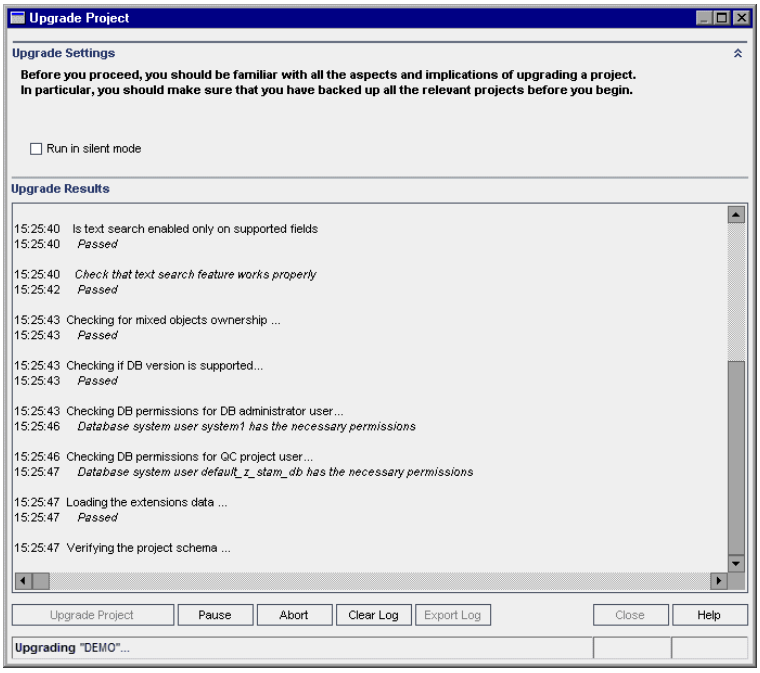
Upgrading a Project 113
Upgrading a Domain 115

Upgrading a Project

This section describes how to upgrade a single project.

To upgrade a project:

1. If you have backed up your project during the repair process (see "Repairing Domains and Projects" on page 110), proceed to step **3**.
2. Back up your projects. For more information, see "Backing Up Projects" on page 118.
3. In Site Administration, click the **Site Projects** tab.
4. In the Projects list, select a project.
5. Click the **Maintain Project** button and select **Upgrade Project**. The Upgrade Project dialog box opens.



6. To run the upgrade process without any user interaction, select **Run in silent mode**.
7. To start the upgrade process, click the **Upgrade Project** button. If the project is active, you are prompted to deactivate it. For more information, see "Deactivating and Activating Projects" on page 91"Deactivating and Activating Projects" on page 91.

   If an error occurs while running the process in non-silent mode, a message box opens. Click the **Abort** or **Retry** buttons accordingly.

   If the upgrade fails, ALM displays an error message with reasons for the failure and refers you to the log file. You must restore the backed up project before you try to upgrade again. For more information, see "Restoring Projects" on page 119.
8. To pause the upgrade process, click the **Pause** button. To continue, click the **Resume** button.
9. To abort the upgrade process, click the **Abort** button. Click **Yes** to confirm.
10. To save the messages displayed in the Upgrade Results pane to a text file, click the **Export Log** button. In the Export Log to File dialog box, choose a location and type a name for the file. Click **Save**.
11. To clear the messages displayed in the Upgrade Results pane, click the **Clear Log** button.

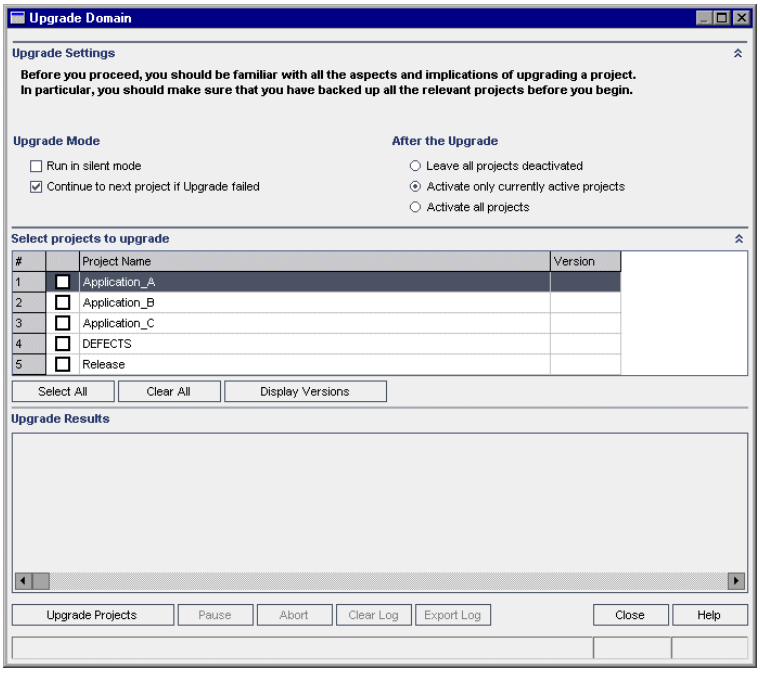
12. Click **Close** to close the Upgrade Project dialog box.

Upgrading a Domain

This section describes how to upgrade all projects in a domain.

To upgrade a domain:

1. If you have backed up your project during the repair process (see "Repairing Domains and Projects" on page 110), proceed to step **3**.
2. Back up your projects. For more information, see "Backing Up Projects" on page 118.
3. In Site Administration, click the **Site Projects** tab.
4. In the Projects list, select a domain.
5. Click the **Maintain Domain** button and select **Upgrade Domain**. The Upgrade Domain dialog box opens.



6. In the **Upgrade Settings** area, under **Upgrade Mode**, you can select the following options:
   - **Run in Silent Mode.** Runs the process without any user interaction.
   - **Continue to next project if upgrade failed.** Proceeds to the next project if the upgrade process fails. This is the default option.
7. In the **Upgrade Settings** area, under **After the Upgrade**, you can select one of the following options:
   - **Leave all projects deactivated.** Leaves all projects deactivated after the upgrade process completes.

- **Activate only currently active projects.** Reactivates previously-activated projects after the upgrade process completes. This is the default option.
- **Activate all projects.** Activates all projects after the upgrade process completes.

8. To view the current version numbers of your projects, select the project names, or click **Select All** to view version numbers for all projects. Click the **Display Versions** button.

   The project version number is displayed in the **Version** column.

9. To upgrade your projects, select the project names, or click **Select All** to verify all projects. Click the **Upgrade Projects** button.

   If an error occurs while running the process in non-silent mode, a message box opens. Click the **Abort** or **Retry** buttons accordingly.

   If the upgrade fails, ALM displays an error message with reasons for the failure and refers you to the log file. You must restore the backed up projects before you try to upgrade again. For more information, see "Restoring Projects" on page 119.

10. To pause the upgrade process, click the **Pause** button. To continue, click the **Resume** button.
11. To abort the upgrade process, click the **Abort** button. Click **Yes** to confirm.
12. To save the messages displayed in the Upgrade Results pane in a text file, click the **Export Log** button. In the Export Log to File dialog box, choose a location and type a name for the file. Click **Save**.
13. To clear the messages displayed in the Upgrade Results pane, click the **Clear Log** button.
14. Click **Close** to close the Upgrade Domain dialog box.

Defining an Exception File

If you have made changes to the schema such as the addition of tables or columns, the upgrade process may fail. You can define an exception file for objects that are added manually to the database user schema, and are not defined in the schema configuration file. This instructs ALM to ignore these changes during the upgrade process.

You can use the exception file to ignore warnings for extra tables, views, columns, and sequences. For any other problem that requires manual repair, consult with your database administrator.

You must use the same exception file when running the verification, repair, or upgrade process.

You can set an exception file for a single project or for all projects in Site Administration.

Caution: Using the exception file to ignore warnings for objects that are added manually to the schema may compromise the stability of your project upgrade and the validity of the database user schema.

To define an exception file:

1. Copy the **SchemaExceptions.xml** file from the ALM installation directory. By default, the file is located in **<ALM installation path>\data\sa\DomsInfo\MaintenanceData**.
2. Create the exception file, for example, **my_exceptions.xml**, and define the exceptions as

follows:

Examples

- For an extra table:

```
<TableMissing>
         <object pattern="MY_Table" type="extra"/>
</TableMissing>
```

- For an extra view:

```
<ViewMissing>
         <object pattern="MY_VIEW" type="extra"/>
</ViewMissing>
```

- For an extra column:

```
<ColumnMissing>
         <object pattern="MY_COLUMN" type="extra"/>
</ColumnMissing>
```

- For an extra sequence:

```
<SequenceMissing>
         <object pattern="MY_SEQUENCE" type="extra"/>
</SequenceMissing>
```

3. To set an exception file for a single project:
   a. In Site Administration, click the **Site Projects** tab.
   b. In the Projects list, select a project. In the right pane, select the **Project Details** tab. The project's details are displayed.
   c. Under **Project Database**, click **Exception File**. The Edit Exception File dialog box opens.
   d. Type the file location. The file is located under **<ALM deployment path>\sa\DomsInfo\MaintenanceData**.
4. To set an exception file for all projects:
   a. In Site Administration, click the **Site Configuration** tab.
   b. Add the **UPGRADE_EXCEPTION_FILE** parameter to the list of parameters and define the exception file location. The file is located under **<ALM deployment path>\sa\DomsInfo\MaintenanceData**.

      For more information on setting parameters, see "Setting ALM Configuration Parameters" on page 167.

      For more information on this parameter, see "UPGRADE_EXCEPTION_FILE" on page 195.

Backing Up Projects

When you run the repair or upgrade process, ALM performs changes on your projects to align them with the specifications for the current version of ALM. ALM also performs backups on a daily basis, regardless of repair or update. You must back up your projects before you start to repair or upgrade them.

We strongly recommend that you deactivate projects before backing them up. If you must back up while your project is still active, you must back up the database before the file system. We also recommend backing up the file system as soon as possible after backing up the database.

To make sure the Garbage Collector does not delete obsolete files while the file system is pending backup, there is a one week grace period before deletion. For details, see "REPOSITORY_GC_DELAY_CANDIDATE_TIME" on page 190.

Note:

- The repair process makes changes to the project database schema only. Before running the repair process, you should back up the project database schema on the database server, but you do not need to back up the project data in the file system.
- Before you run the upgrade process, perform a full backup of your projects that includes the project database schema and the project data.

To back up the project database schema on the database server:

- **Microsoft SQL database.** For information on backing up a schema on a Microsoft SQL database, see HP Software Self-solve knowledge base article KM169526 (http://h20230.www2.hp.com/selfsolve/document/KM169526).
- **Oracle database.** For information on backing up a schema on an Oracle database, see HP Software Self-solve knowledge base article KM205839 (http://h20230.www2.hp.com/selfsolve/document/KM205839).

To back up the project data in the file system:

- In the file system, ensure that all data, including automated tests scripts and results, and attachments, is saved in the project directory under the repository defined during the installation of ALM. Make a copy of this project directory including all subdirectories and files.
- **For upgrade from Quality Center 10.00:** If your automated tests are stored outside the project directory, make a copy of them.

Tip: To find out whether your tests are outside the project directory, log in to Site Administration. In the **Site Projects** tab, expand the project you want to back up and click the **DATACONST** table. Select the **DC_CONST_NAME** parameter with the value **tests_directory** and verify the corresponding **DC_VALUE**. If it is not set as **tests** but is set instead to some other location, then your tests are outside of the project folder.

Restoring Projects

If the repair or upgrade process fails, you must restore the backed up projects before trying the process again. You can restore projects that were backed up on an Oracle or Microsoft SQL database server, and in the file system. A project you restore can be used only in the ALM/Quality Center version from which it was backed up.

If you are restoring projects as part of your overall upgrade process, for example -- if you are moving to a new server or database, you must update the **dbid.xml** file accordingly. For details, see "Restoring Access to Projects" on page 97.

If you were previously working with Performance Center 11.00, see "Restoring LAB_PROJECT" on next page.

This section includes:

Restoring Projects from a Microsoft SQL Database Server 119
Restoring Projects from an Oracle Database Server 119
Restoring a Repository from the File System 120
Restoring LAB_PROJECT 120

Restoring Projects from a Microsoft SQL Database Server

This section describes how to restore a project backed up on a Microsoft SQL database server.

For more information, see HP Software Self-solve knowledge base article KM169526 (http://h20230.www2.hp.com/selfsolve/document/KM169526).

To restore a project from a Microsoft SQL database server:

1. From the SQL Server Enterprise Manager, navigate to the database and select **Tools > Restore Database**.
2. Navigate to the backup file, and follow the restore procedure to complete the data restore process.
3. In Site Administration, restore the project. If you are restoring your project from a different directory, or if you renamed your schema, you must update the **dbid.xml** file accordingly. For details, see "Restoring Access to Projects" on page 97
4. If the backup was performed while the project was active, realign the project repository. For more information, see "Realigning Repositories" on page 85.

Restoring Projects from an Oracle Database Server

This section describes how to restore a project backed up on an Oracle database server.

For more information, see HP Software Self-solve knowledge base article KM205839 (http://h20230.www2.hp.com/selfsolve/document/KM205839).

To restore a project from an Oracle database server:

1. Copy the backup file to the Oracle server machine.
2. Using the SQL*Plus utility, log in to the Oracle server using the **system** account.
3. Create a user for the ALM project. Make sure you create it with the same name as the project name (or the Oracle user name) when the project was exported.

   Use these SQL statements:

   ```
   CREATE USER [<project name>] IDENTIFIED BY tdtdtd DEFAULT
   TABLESPACE TD_data TEMPORARY TABLESPACE TD_TEMP;

   GRANT CONNECT,RESOURCE TO [<project name>];
   ```

4. On the ALM installation DVD, locate the **\Utilities\Databases\Scripts** directory. Open the **qc_project_db___oracle.sql** file and follow the instructions.
5. Using the command line, type `imp` to run the import utility.
6. Follow the prompt, and log in to the Oracle server using the **system** account. Make sure you import all the dump files.

   After all tables have been successfully imported, a confirmation message displays.
7. In Site Administration, restore the project. If you are restoring your project from a different directory, or if you renamed your schema, you must update the **dbid.xml** file accordingly. For more information on restoring access to projects, see "Restoring Access to Projects" on page 97.
8. If the backup was performed while the project was active, realign the project repository. For more information, see "Realigning Repositories" on page 85.

Restoring a Repository from the File System

This section describes how to restore a repository backed up in the file system.

To restore a repository from the file system:

1. Copy the backed up repository to the ALM repository.
2. In Site Administration, restore the project. If you are restoring your project from a different directory, or if you renamed your schema, you must update the **dbid.xml** file accordingly. For more information on restoring access to projects, see "Restoring Access to Projects" on page 97.
3. If the backup was performed while the project was active, you must restore the database and the file system from a backup that was created after the database backup was created, realign the project, and then activate the project. This procedure must always be performed during disaster recovery. For more information, see "Realigning Repositories" on page 85. If this process is done in order to recover certain files or directories, you can skip the realignment.

Restoring LAB_PROJECT

If you were previously working with Performance Center 11.00, and ALM 11.52 is installed on a new server, you must restore your backed up projects on the new server. Before restoring and

upgrading other Performance Center projects, you must first restore and upgrade LAB_PROJECT, and then any Performance Center **template** projects.

You restore LAB_PROJECT from the Lab Management tab in Site Administration. For details, refer to the *HP ALM Lab Management Guide*.

Restoring Performance Center 11.00 projects includes the following steps:

1. If a new Site Administration schema was created during installation, a new LAB_PROJECT was created. To restore your Performance Center 11.00 LAB_PROJECT, you must first remove the new LAB_PROJECT from the ALM 11.50 server.
2. In ALM 11.52 Site Administration, restore your Performance Center 11.00 LAB_PROJECT.
3. Proceed with the verify, repair, and upgrade of your LAB_PROJECT. For details, see "About Upgrading Projects" on page 103.

   If the verification process fails and you receive an error message that project fields cannot be decrypted using the Confidential Data Passphrase, see "Encrypted Values" on page 518 in "Upgrade Preparation Troubleshooting" on page 494.
4. Restore, verify, repair, and upgrade any Performance Center 11.00 **template** projects.

   Note: Performance Center 11.00 **template** projects must be restored to the **DEFAULT** domain.
5. Restore, verify, repair, and upgrade your Performance Center 11.00 projects.

Migrating the Repository

ALM versions 11.00 and later use a new project repository architecture, that is optimized to allow maximum storage space. For details on the new repository, see "Managing the Optimized Project Repository" on page 82.

When upgrading projects from Quality Center 10.00 to ALM 11.50, the project repository is automatically upgraded to the optimized repository format. This is carried out in two stages:

1. The first stage is performed during the upgrade of the project. In this stage, all files in the repository are scanned, and their names are stored in a project table.
2. After upgrade is completed, the project is reactivated. The repository files are gradually migrated to the new system. In this stage, the files are moved from their old location to their new location in the optimized repository. Depending on various factors, such as the size of the repository and the network speed, the file migration may take up to several days.

   This second phase of the repository migration is carried in the background. Users can work in the project even while it is in progress. New files that you add to a project after upgrade are saved in the new project repository.

Note:

- Until the migration process for a project is complete, you cannot export or copy the project.
- To back up a project before its migration is complete, you must suspend the migration process. For more details, see "Configure Migration Priority" on page 125.

Monitor the migration progress in the Repository Migration Status window, and troubleshoot any problems that may occur.

In Site Administration, you can track the status of file migration for each project, and configure the number of resources allocated to performing the migration.

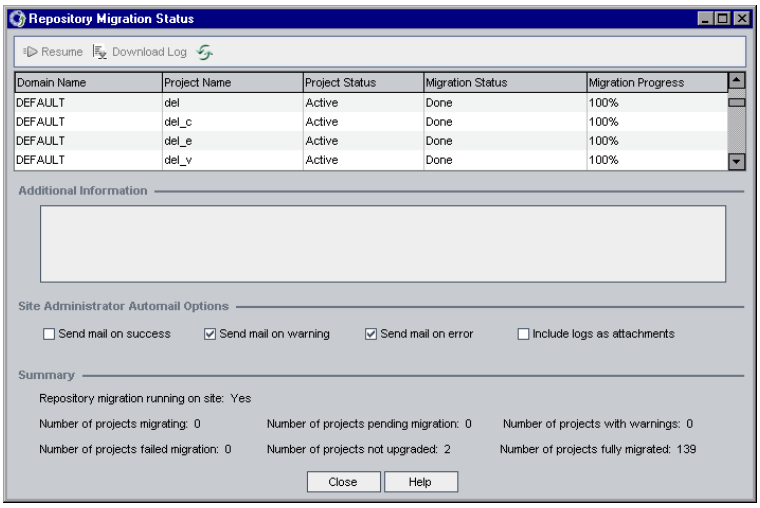
This section includes:

Repository Migration Status Window 122

Configure Migration Priority 125

Repository Migration Status Window

This window lists all the site projects and displays the status of their migration to the optimized project repository.



To access	In Site Administration, select **Tools** > **Repository Migration Status**.
See also	• "Migrating the Repository" on previous page • "Configure Migration Priority" on page 125 • "Managing the Optimized Project Repository" on page 82

User interface elements are described below:

UI Element	Description
	Instructs ALM to resume the migration of the selected project. If an error or warning was detected during the migration of the selected project, fix the problem as described in the **Additional Information** field, and click **Resume**. **Note:** If the migration process stopped due to missing files, after clicking **Resume**, you can no longer restore missing files.
	Downloads a log of the migration events associated with the selected project.
	Refresh. Refreshes the display with the most up-to-date information. **Note:** The grid updates automatically after the migration of every 1000 files.
Domain Name	The domain to which the selected project belongs.
Project Name	The name of the selected project.
Project Status	Indicates the selected project's status in Site Administration. For example, `Active` or `Deactivated`. **Note:** Deactivating a project does not affect its repository migration.

UI Element	Description
Migration Status	The migration status of a project can be one of the following: • **None.** Project is not upgraded to ALM 11.52, and will not be migrated. • **Pending.** File migration is pending. • **Migrating.** File migration is in progress. • **Done.** File migration is complete. • **Error.** An error occurred during file migration, and migration could not be completed. See the cause of the error in the **Additional Information** panel. Fix the error, and click **Resume**. • **Warning.** A warning occurred during file migration. For details of the warning, and the actions you must take to resolve the problem, download the log files listed in the **Additional Information** panel. Resolve the problems as necessary, and click **Resume** to complete the migration. **There are several possible causes for warnings:** • One or more project files were not found in the project repository. This can result from missing or renamed files. • Redundant files are found in the repository. The migration cannot complete until the legacy repository is empty of files. Redundant files can be one of the following: ▪ Duplicate project files that could not be deleted. This can result from insufficient permissions. ▪ Files unrelated to ALM that were manually saved in the project repository. ▪ Unidentified project files.
Migration Progress	The number of project files migrated to the new repository, as a percentage of the total number of project files.
Additional Information	If a problem was detected, displays the cause of the problem, and links to log files. The log files describe the actions you must take to resolve the problem.

UI Element	Description
Site Administrator Automail Options	ALM sends automail to Site Administrators upon events connected to repository migration. Select the following options: • **Send mail on success.** Sends mail when the migration of a project repository completes successfully. • **Send mail on warning.** Sends mail when a warning is detected during the migration of a project repository. • **Send mail on error.** Sends mail when an error is detected during the migration of a project repository. • **Include logs as attachments.** Attaches detailed log files to automail messages.
Summary	Summary information of the migration status of all site projects.

Configure Migration Priority

While the migration process does not interfere with your work on projects, the process may affect system performance as a whole. Use the following site configuration parameters to control the amount of system resources used by the migration process.

- **REPOSITORY_MIGRATION_JOB_PRIORITY.** Determines the speed at which files are copied from the old to the new project repository. For more details, see "REPOSITORY_MIGRATION_JOB_PRIORITY" on page 191.
- **SUSPEND_REPOSITORY_MIGRATION.** Stops the repository migration on the entire site. Use this parameter temporarily, and only in special circumstances. For example, if you suspect that the migration process disrupts your system. For more details, see "SUSPEND_REPOSITORY_MIGRATION" on page 194.

Additional parameters are available for configuring the resources allocated to the migration process. For more information, see HP Software Knowledge base article KM862600 (http://h20230.www2.hp.com/selfsolve/document/KM862600).

Consider the following when configuring the migration priority:

- Allocating more resources to the migration process may slow down other processes.
- Allocating less resources extends the time in which the process is completed.
- Projects that are pending migration or in the process of migration cannot be exported or copied.

LAB_PROJECT Post Upgrade Steps

If you have upgraded a Performance Center 11.00 project, perform the following steps in the staging and production environments:

For more information regarding these steps, refer to the *HP ALM Lab Management Guide*.

1. Log into **Lab Management**.
2. Configure the external URL:
   a. On the Lab Management sidebar, under **Servers**, select **PC Servers**.
   b. On the toolbar, click the **ALM Connection** button. The ALM Connection dialog box opens.
   c. Enter external URL information.
3. Add a Performance Center Server:
   a. On the **PC Servers** toolbar, click the **New PC Server** button. The New PC Server dialog box opens.
   b. Define new server information.
   c. Click **OK**.
4. Update license details:
   a. On the Lab Management sidebar, under **Performance Center**, select **PC Licenses**.
   b. Define new license information.
5. Add, remove, or reconfigure hosts:
   a. On the Lab Management sidebar, under **Lab Resources**, select **Hosts**.
   b. Add, remove, or reconfigure hosts as needed.

About Upgrading Projects for Minor-Minor Versions

This section describes the process for upgrading projects for minor-minor versions, including setting project priorities. Minor-minor versions can include changes to database schema.

Note: For details on understanding whether a version is major, minor, or minor-minor, see .

Automatic upgrades for minor-minor versions start when the ALM Server is restarted after installation of the new minor-minor version.

While a project is being upgraded, users cannot access the project.

This section includes:

- "Candidates for Automatic Upgrades" below
- "Project Priority" on next page
- "How to Upgrade Minor-Minor Versions" on next page

Candidates for Automatic Upgrades

The automatic upgrade processes projects that meet all of the following criteria:

- Lab Management projects
- Projects whose major and minor version are identical to the major and minor version installed on

the server, yet whose minor-minor version is less than the minor-minor version installed on the server.

Examples

Assume ALM version 15.25 is installed on the server:

- Version 15. 21, 15.22, 15.23, and 15.24 projects are automatically upgraded.
- Version 15.25 and 15.26 projects are not automatically upgraded.
- Version 14.00 projects are not upgraded.

Project Priority

The upgrade processes projects in the following order:

- Template and Lab Management projects
- Projects with the highest priority, as set by the user
- Most active projects
- Most recent modification date
- Highest number of users

How to Upgrade Minor-Minor Versions

1. Before upgrading the minor-minor version on the ALM Server, you can optionally set priorities to determine the order in which projects are upgraded using the Project Update Prioritization dialog box in Site Administration. For user interface details, see "Setting Project Priorities for Upgrading Minor-Minor Versions" on next page.
2. Back up the database.
3. Install the new minor-minor version on the ALM Server by following the instructions:
   - On the HP Software Support Web site. Choose **Help > HP Software Support**. The URL for this Web site is www.hp.com/go/hpsoftwaresupport.
   - In the *HP Application Lifecycle Management Installation and Upgrade Guide.*

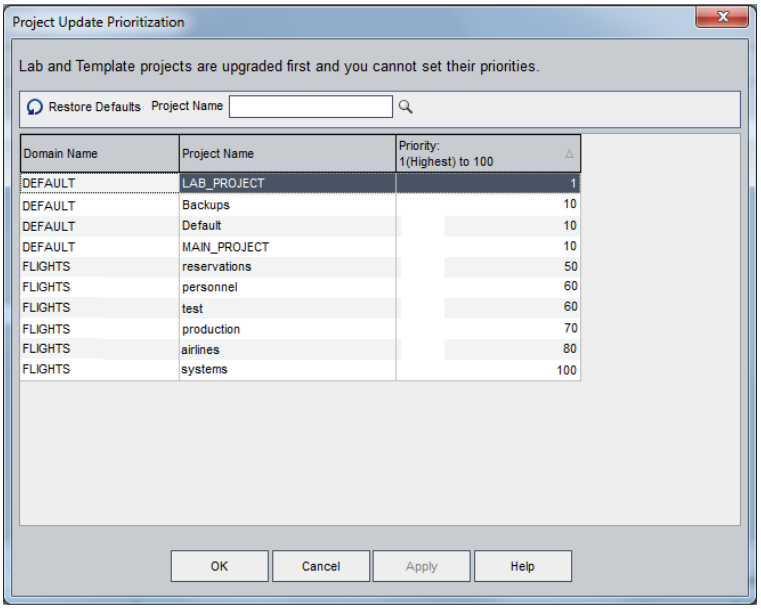
After the ALM Server is restarted, projects are upgraded automatically according to the defined priorities.

This section also includes:

Setting Project Priorities for Upgrading Minor-Minor Versions 128

Setting Project Priorities for Upgrading Minor-Minor Versions

The Project Update Prioritization dialog box enables you to set project priorities for upgrading minor-minor versions.



To access	In Site Administration, select **Tools > Set Project Upgrade Priorities**.
Important information	• Projects are listed in order of priority. Highest priority projects are at the top of the list. • Lab and Template projects are always highest priority. You cannot change the priority for these projects. • You can set priorities at any time. The changes affect the next minor-minor version upgrade. If such an upgrade is in progress, the changes affect only projects with **Pending Upgrade** status.
See also	• "Upgrading Projects to a New Version" on page 102 • *HP Application Lifecycle Management Installation and Upgrade Guide*

User interface elements are described below:

UI Element	Description
Restore Defaults	**Restore Defaults**. Resets project priorities to the default value **100**. **Note: Restore Defaults** does not affect the priority of Lab or Template projects.

UI Element	Description
	Find. Enables you to navigate to a project in the grid using the adjacent **Project Name** text box.
Domain Name	Lists the domains.
Project Name	Lists the projects.
Priority	Displays the priority for the project for the next upgrade. **1** is highest priority. **100** is lowest priority.

Chapter 6: Managing ALM Users

You manage HP Application Lifecycle Management (ALM) users in Site Administration. You can add new users, define user details, change user passwords, and define site administrators. You can also import users from LDAP and enable LDAP authentication for users. After you add users, you can assign projects to users.

This chapter includes:

About Managing Users 131

Adding a New User 131

Importing Users from LDAP 132

Updating User Details 138

Deactivating and Activating Users 139

Creating and Changing Passwords 139

Enabling LDAP Authentication for Users 140

Assigning Projects to Users 141

Exporting User Data 143

Deleting Users 144

About Managing Users

You use Site Administration to manage the users connected to your ALM projects. You begin by adding or importing new users to the Users list in Site Administration. You can then define user details and change or override a user's password. You can also enable users to log in to ALM using their LDAP passwords.

For each ALM user, you can select projects that the user can access. You can also define ALM users as site administrators. For more information, see "Defining Site Administrators" on page 31.

Note: You can monitor the users currently connected to an ALM server. For more information, see "Managing User Connections and Licenses" on page 146.

Adding a New User

You can add new users to the Users list in Site Administration. After the user is added, you can view users and define user details. For more information on updating user details, see "Updating User Details" on page 138.

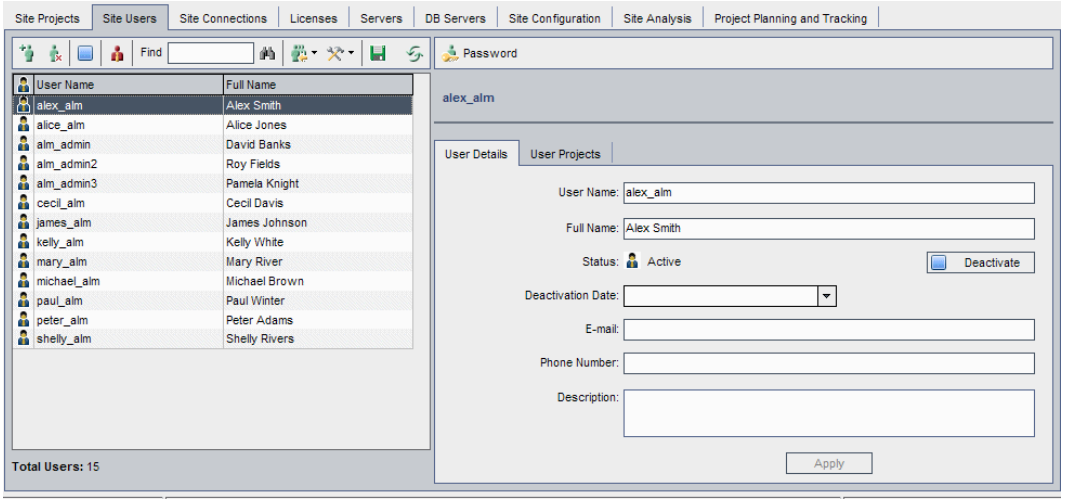
You can also import new users from LDAP directories. For more information, see "Importing Users from LDAP" on next page.

Note: Creating a new user for an ALM project consists of two steps:

- Adding the user to the Users list in Site Administration (as described in this section).
- Assigning the user to a user group using Project Customization. Each user group has access to certain ALM tasks. For more information, see "Managing Users in a Project" on page 248 and "Managing User Groups and Permissions" on page 252.

To add a new user:

1. In Site Administration, click the **Site Users** tab.



2. Click the **New User** button . The New User dialog box opens.
3. Type a **User Name** (maximum length 60 characters) and **Full Name**. A user name cannot include the following characters: **( ) @ \ / : * ? " ` < > | + = ; , %**
4. Type additional user information: **Full Name**, **E-mail**, **Phone Number**, and a **Description**. The email information is important, as it enables users to receive project information directly to their mailboxes.

   Note: You can update user information in the User Details tab. For more information, see "Updating User Details" on page 138.

5. Click **OK**. The new user is added to the Users list.
6. Assign the new user a password. For task details, see "Creating and Changing Passwords" on page 139.

   Note: A new user is created with a blank password.

Importing Users from LDAP

You can import users from an LDAP directory to the Users list in Site Administration.

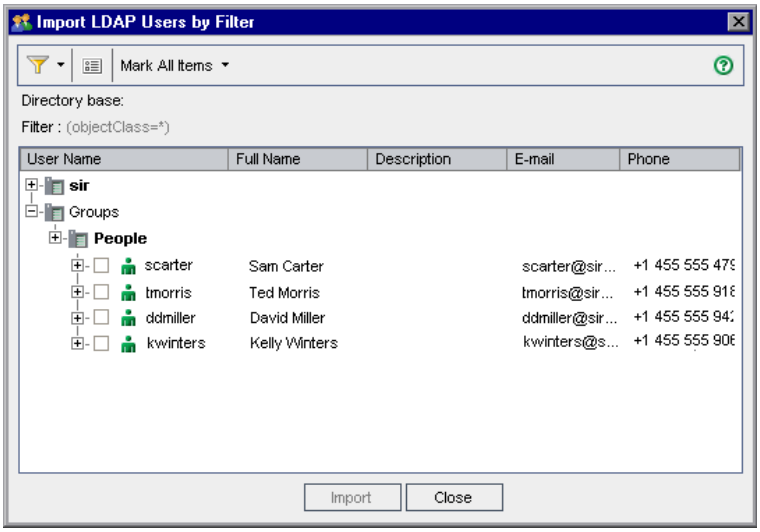
Notes:

- Verify that the LDAP import settings are defined. For more information, see "Defining LDAP Settings for Importing Users" on page 134.
- Working with LDAP through SSL requires that you perform additional steps. For more information, see HP Software Self-solve knowledge base article KM188096 (http://h20230.www2.hp.com/selfsolve/document/KM188096).
- The **LDAP_TIMEOUT** parameter enables you to define a connection time-out between ALM and an LDAP server. By default, the value is set to 10 minutes. For more information, see "LDAP_TIMEOUT" on page 184.

You select users either by filtering and browsing the LDAP directory base, or by searching for users by keyword.

To browse the LDAP directory base:

1. In Site Administration, click the **Site Users** tab.
2. Click the **Import LDAP Users** button , and select **Import LDAP Users by Filter**. The Import LDAP Users by Filter dialog box opens.



3. To filter the LDAP directory base, click the **Filter All** button . If you have pre-selected users, a warning message box opens. Click **OK** to clear all selections and continue. The Filter dialog box opens. Type a filter condition to display specific records from your LDAP directory base and click **OK**.

4. To view LDAP details for a user, select an item and click the **Show LDAP Details** button . The LDAP User Details dialog box opens and displays the user attributes.

5. You can use the following options to import users:

   - To import a user, expand a directory and mark the user name by selecting the check box.
   - To import a group of users, use CTRL or SHIFT to highlight users to be included. Click the **Mark All Items** arrow and choose **Mark Selected Items** to select the check boxes of the highlighted users.
   - To import all users, click **Mark All Items**.

6. To clear the check boxes of highlighted users, click the **Mark All Items** arrow and choose **Clear Selected Items**. To clear all check boxes, click the **Mark All Items** arrow and choose **Clear All**.

7. Click **Import**. A confirm message box opens. Click **Yes** to continue.

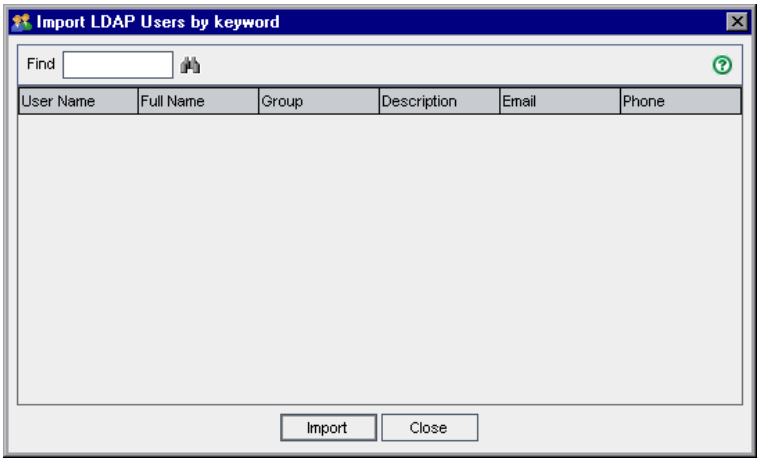
   - If the users were imported successfully, a message box opens. Click **OK** and then click **Close** to close the Import LDAP Users dialog box.
   - If the same user names exist in the Users list, the Handle Conflict dialog box opens. For more information, see "Handling User Name Conflicts" on page 137.

8. Click **Close** to close the Import LDAP Users dialog box.

To search for users by keyword:

1. In Site Administration, click the **Site Users** tab.

2. Click the **Import LDAP Users** arrow , and select **Import LDAP Users by Keyword**. The Import LDAP Users by Keyword dialog box opens.



3. In the **Find** box, type a keyword, and click the **Find** button .

   ALM searches for the keyword in the following fields: **User Name**, **Full Name**, **Group**, **Description**, **Email**, **Phone**.

   > **Tip:** To broaden your search, enter partial values. For example, enter `Mi` to search for `Michael` and `Mikhael`.

4. Click **Import**. A confirm message box opens. Click **Yes** to continue.
   - If the users were imported successfully, a message box opens. Click **OK**and then click **Close** to close the Import LDAP Users dialog box.
   - If the same user names exist in the Users list, the Handle Conflict dialog box opens. For more information, see "Handling User Name Conflicts" on page 137.
5. Click **Close** to close the Search LDAP Users dialog box.

Defining LDAP Settings for Importing Users

To enable you to import users from an LDAP directory to the Users list in Site Administration, you must define your LDAP import settings.

When you import users from an LDAP directory, ALM copies attribute values from an LDAP directory into ALM. For each imported user, the following attribute values are copied:

- **Distinguished name (DN).** A unique name that is made up of a sequence of relative distinguished names (RDN) separated by commas.

  Example

  `CN=John Smith, OU=QA, O=HP`

  `CN` is the common name; `OU` is the organizational unit; and `O` is the organization.

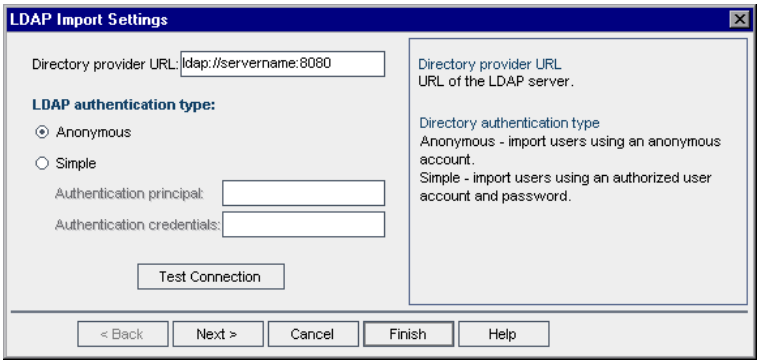
- **Userid (UID).** The name that identifies a user as an authorized user. The UID attribute value is mapped to the **User Name** field in ALM.

- **Full Name, Description, Email and Phone.** Optional attributes that are used to populate the Full Name, Description, Email, and Phone Number fields for each user imported from an LDAP directory.

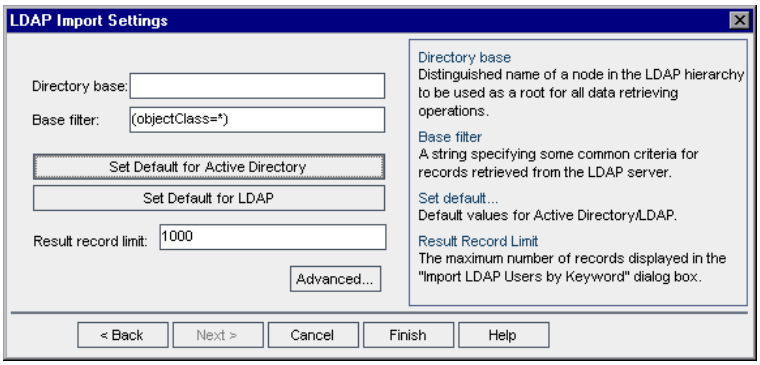
Note: The optional **LDAP_IMPORT_ATTRIBUTE_MASK** parameter enables you to define a regular expression that can be used to distinguish between different values for an LDAP attribute. For details, see "LDAP_IMPORT_ATTRIBUTE_MASK" on page 184.

To define LDAP settings for importing users:

1. In Site Administration, click the **Site Users** tab.
2. Click the **User Settings** button and select **LDAP Import Settings**. The LDAP Import Settings dialog box opens.



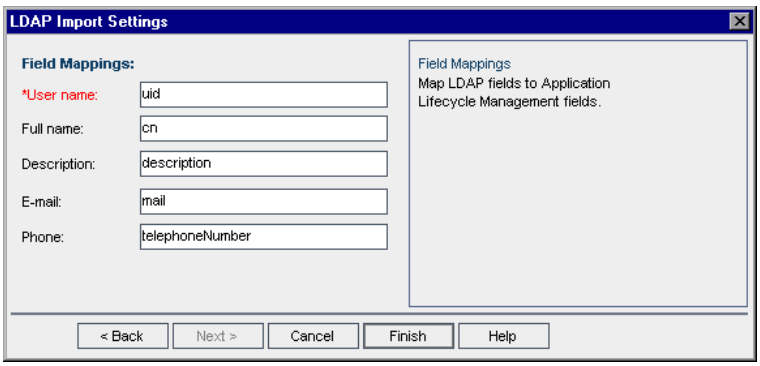
3. In the **Directory provider URL** box, type the URL of the LDAP server (`ldap://<server name>:<port number>`).
4. Under **LDAPauthentication type**:
   - Select **Anonymous** to enable you to import users from the LDAP server using an anonymous account.
   - Select **Simple** to enable you to import users from the LDAP server using an authorized user account and password.
5. If you select **Simple**, the following options are enabled:
   - In the **Authentication principal** box, type the authorized user name.
   - In the **Authentication credentials** box, type the password.
6. Click the **Test Connection** button to test the URL of the LDAP server.
7. Choose one of the following options:
   - To define additional LDAP settings, proceed to the next step.
   - To close the LDAP Import Settings dialog box, click **Finish**.
8. To define additional LDAP settings, click **Next**. The following dialog box opens:



9. In the **Directory base** box, type the LDAP directory name.

   Note: The **Directory base** is a distinguished name of a node in the LDAP hierarchy and is used as a root for operations retrieving data. If this field is left empty, it dramatically increases the search time of a user in the LDAP tree.

10. In the **Base filter** box, define filter criteria.
11. To set the default values for the Active Directory, click the **Set Default for Active Directory** button.
12. To set the default values for LDAP, click the **Set Default for LDAP** button.
13. In the **Result record limit** box, enter the maximum number of records to display in the **Import LDAP Users by Keyword** dialog box. The minimum value is 100; the default value is 1000.
14. Choose one of the following options:
    - To populate optional attributes in ALM for each user imported from an LDAP directory, proceed to the next step.
    - To close the LDAP Import Settings dialog box, click **Finish**.
15. To populate optional attributes in ALM for each user imported from an LDAP directory, click **Advanced**. The following dialog box opens.



16. Define the corresponding LDAP field names. Note that **User Name** is a required field.
17. Click **Finish** to close the LDAP Import Settings dialog box.

Handling User Name Conflicts

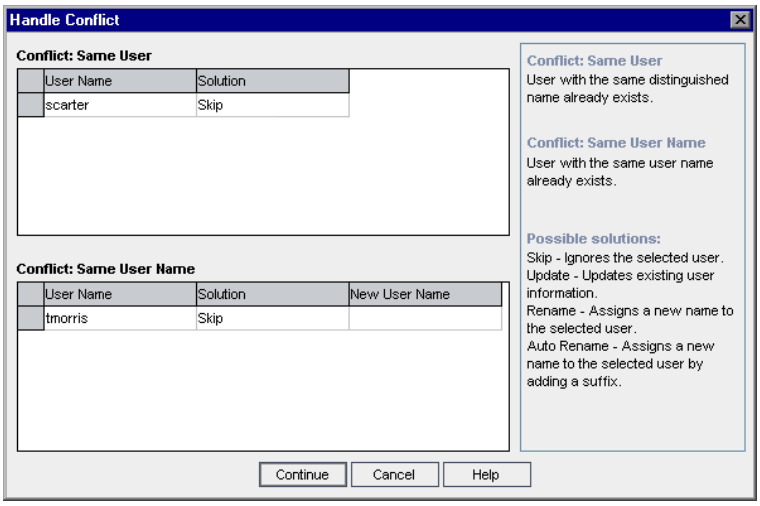
When importing users from an LDAP directory to the Users list in Site Administration, you may encounter the following conflicts:

- **Same user.** A user with the same LDAP distinguished name already exists.
- **Same user name.** A user with the same user name already exists.

To resume the process of importing users, you can choose to skip the user, rename a user name, or update user information.

To handle user name conflicts:

1. Import users (see "Importing Users from LDAP" on page 132). If conflicts occur, the Handle Conflict dialog box opens.



2. If the conflict is listed under **Conflict: Same User**, you can choose one of the following options to resume the process:

Option	Description
Update	Updates existing user information. Click the corresponding **Solution** box. Click the browse button and choose **Update**.
Skip	Does not import the selected user (default).

3. If the conflict is listed under **Conflict: Same User Name**, you can choose one of the following options to resume the process:

Option	Description
Rename	Assigns a new name to the selected user. Click the corresponding **Solution** box. Click the browse button and choose **Rename**. In the **New User Name** box, type the new name.

Option	Description
Auto Rename	Assigns a new name to the selected user by adding a suffix. Click the corresponding **Solution** box. Click the browse button and choose **AutoRename**. The new name is displayed in the **New User Name** box.
Update	Updates existing user information. Click the corresponding **Solution** box. Click the browse button and choose **Update**.
Skip	Does not import the selected user (default).

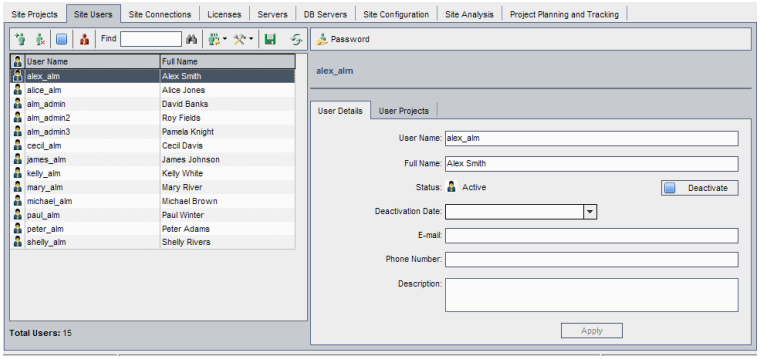
4. Click **Continue**.

Updating User Details

After you add a user, you can update user details. For example, you may need to update a user's full name or contact details. You can also define ALM users as site administrators. For more information, see "Defining Site Administrators" on page 31.

To update user details:

1. In Site Administration, click the **Site Users** tab. In the right pane, click the **User Details** tab.



2. Select a user from the Users list.

> **Tip:** You can search for a user in the Users list by typing the name, or the first letter(s) of the name, of a user in the **Find** box, and clicking the **Find** button . The first user that matches the searched text is highlighted. Click the button again to search for other users containing the searched text.

3. Edit the user detail fields.

> **Note:** If the user was imported from an LDAP directory to Site Administration, the **Domain Authentication** box displays the LDAP authentication properties of the imported user. If the user was not imported, the **Domain Authentication** box is not displayed. For more information, see "Importing Users from LDAP" on page 132.

4. To set the user's status, click the **Deactivate** or **Activate** button. For more information on user status, see "Deactivating and Activating Users" on next page.

5. To assign projects to a user, click the **User Projects** tab. For more information, see "Assigning Projects to Users" on page 141.
6. Click **Apply** to save your changes.

Deactivating and Activating Users

You can deactivate or activate an ALM user. A deactivated user cannot log in to any project. The user is not deleted from the Users list, and all user permissions and settings are saved. This can be useful, for example, for contract workers that work intermittently for a set period of time.

Caution: A deactivated site administrator user cannot log in to Site Administration.

To deactivate a user:

1. In Site Administration, click the **Site Users** tab.
2. Choose one of the following:
   - To deactivate users as of the next attempted login, select one or more Active users from the Users list, and click the **Deactivate** button on the toolbar. The **Status** is set to **Inactive**, and the user icon is changed in the Users list. In addition, the **Deactivation Date** box is hidden.

     If a user is currently logged in to an ALM project, this does not terminate the user session. When the user next attempts to log in to a project, a message box displays stating that the user is deactivated and cannot log in.
   - To deactivate a user on a set date in the future, select an Active user from the Users list. Click the **User Details** tab. In the **Deactivation Date** box, click the drop-down arrow and select a date.

To activate a user:

1. In Site Administration, click the **Site Users** tab.
2. Select one or more Inactive users from the Users list.
3. Click the **Activate** button on the toolbar. The **Status** is set to **Active**, and the user icon is changed in the Users list.

Creating and Changing Passwords

The site administrator can create, change, or override a user's password.

Note:

- You can only change passwords for users that are set to log in to ALM using their ALM passwords. If LDAP passwords are in use, this option is unavailable. For more information on LDAP authentication, see "Defining LDAP Settings for Importing Users" on page 134.
- Users who are not administrators can change their passwords using the **User Properties** link in the Project Customization window. For more information, refer to the *HP Application*

Lifecycle Management User Guide.

- For details on the old connection method to Site Administration for the purposes of backward compatibility, see "BACKWARD_SUPPORT_SA_DEFAULT_USER" on page 173.

To create or change a password:

1. In Site Administration, click the **Site Users** tab.
2. Select a user from the Users list.
3. Click the **Password** button. The Set User Password dialog box opens.
4. In the **New Password** box, type a new password (maximum length 20 characters).
5. In the **Retype Password** box, retype the new password.
6. Click **OK**.

Enabling LDAP Authentication for Users

You can allow users to log in to ALM using their LDAP passwords, instead of ALM passwords.

Working with LDAP through SSL requires that you perform additional steps. For more information, see HP Software Self-solve knowledge base article KM188096 (http://h20230.www2.hp.com/selfsolve/document/KM188096).

This section includes:

- "Extending LDAP Authentication" below
- "Considerations" on next page
- "Enabling LDAP Authentication for Users" on next page

Extending LDAP Authentication

When users attempt to log in to ALM, they are authenticated against LDAP using the distinguished names (DN) that are stored in the Domain Authentication property in the ALM database. When the user attempts to log in, and the DN information in ALM is invalid, the user cannot log in to ALM.

You can enhance the search so that when the DN information is invalid, ALM also searches on the LDAP server, using the LDAP import settings defined in Site Administration. If the user is found, the DN is updated in ALM, and an automatic login attempt is performed.

To set this extended search, define a comma-separated list for the "LDAP_SEARCH_USER_CRITERIA" on page 169 Site Configuration parameter. The possible values are **username**, **email**, **fullname**, **phone**, **description**. The order of the properties defines their priority if multiple results are found.

Example

If the parameter is set to `username` and `email`, and two users are found with the same user name on the LDAP server, their email addresses are checked. If more than one user is found

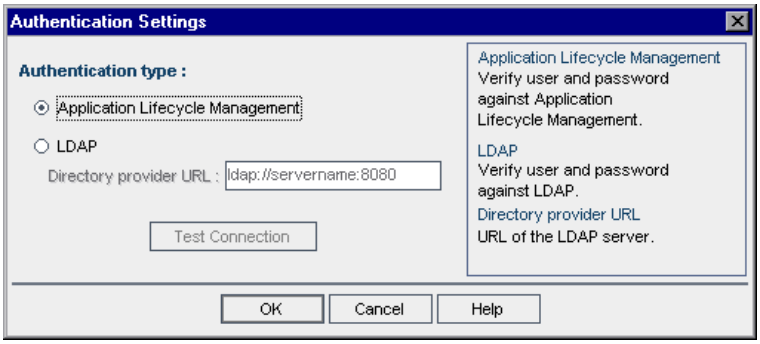
answering the criteria, an error message is returned. If the search for the user succeeds, the user is logged in to ALM.

Considerations

- ALM does not support a multi-server LDAP environment.
- After LDAP authentication is enabled, authentication will be performed against the LDAP server. Make sure that the site administrator is set up as an LDAP user before switching to LDAP authentication, otherwise the site administrator will not be able to log in after the authentication type is switched.
- After you enable LDAP authentication, you need to disable the password reset option, by defining the "PASSWORD_RESET_DISABLE" on page 187 Site Configuration parameter.

Enabling LDAP Authentication for Users

1. In Site Administration, click the **Site Users** tab.
2. Click the **User Settings** button and choose **Authentication Settings**. The Authentication Settings dialog box opens.



3. Under **Authentication type**, select **LDAP** to set the authentication type as LDAP for all users.
4. In the **Directory provider URL** box, type the URL of the LDAP server (`ldap://<server name>:<port number>`).
5. Click the **Test Connection** button to test the URL of the LDAP server.
6. Click **OK**.

Assigning Projects to Users

As an ALM site administrator, you can control user access to ALM projects by defining the projects to which a user can log on. When a user is no longer working on a project, remove the user from the User Projects list.

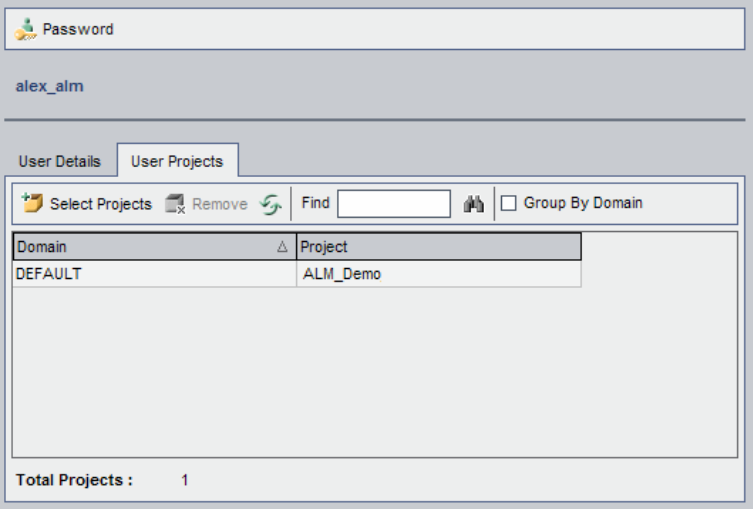
When you add a user to a project, the user is automatically assigned to the project with Viewer privileges. For more information on user groups and group privileges, see "Managing Users in a Project" on page 248 and "Managing User Groups and Permissions" on page 252.

Note:

- You can assign users to projects from the Site Projects tab. For more information, see "Assigning Users to Projects" on page 77.
- ALM sends automatic email notification to project administrators when users are assigned or removed from a project in Site Administration. You can make automatic notification unavailable by adding the "AUTO_MAIL_USER_NOTIFICATION" on page 173 parameter in the Site Configuration tab.

To assign projects to a user:

1. In Site Administration, click the **Site Users** tab. In the right pane, select the **User Projects** tab. The Projects list for the selected user is displayed.



Tip: You can click the **Domain** column to change the sort order from ascending to descending domain names. You can also click the **Project** column to sort according to project instead of domain name.

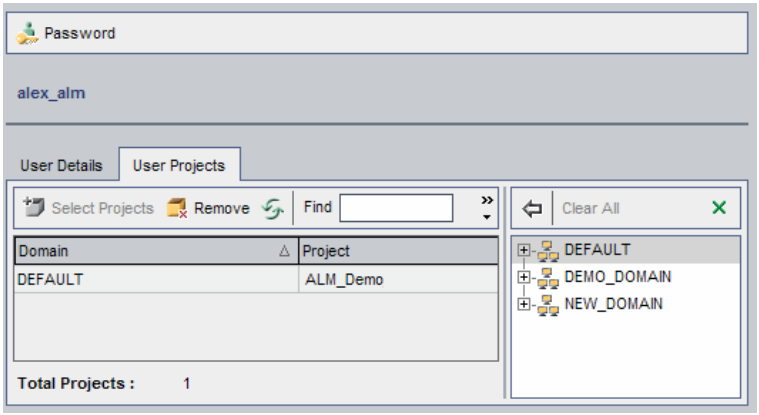
2. In the Users list in the left pane, select a user.

Tip: You can search for a user by typing the name of a user in the **Find** box, and clicking the **Find** button .

The selected user's projects are displayed in the User Projects list.

To group user projects by domain, select **Group By Domain**. Clear the check box to remove the group by settings.

3. In the User Projects tab, click the **Select Projects** button. The ALM Projects list is displayed in a new pane to the right of the User Projects tab.



4. To select projects from the Projects list, expand the directory, and select the projects that you want to assign to the user.

   To clear all selected projects, click **Clear All**.

5. Click the **Add current user to the selected projects** button . The selected projects are added to the User Projects list.

6. To remove a project from the User Projects list, select the project in the User Projects tab and click the **Remove** button. Click **OK** to confirm. The project is removed from the User Projects list.

   Note: This does not delete the project from the server.

7. To refresh the User Projects list, click the **Refresh** button .

Exporting User Data

You can export the user name and full name of all site users from the Users list to a text file.

To export user data:

1. In Site Administration, click the **Site Users** tab.

2. Click the **Export User Data To File** button . A confirm message box opens. Click **Yes** to continue. The Export Data To File dialog box opens.

3. Select the directory where you want to save the parameters, and type a name for the file in the **File name** box.

4. Click **Save** to export the data to a text file.

Deleting Users

You can delete a user from the Users list.

To delete a user:

1. In Site Administration, click the **Site Users** tab.
2. Select a user from the Users list.
3. Click the **Delete User** button .
4. Click **Yes** to confirm.

Chapter 7: Managing User Connections and Licenses

In Site Administration, you can monitor user connections and modify license information.

This chapter includes:

About Managing User Connections and Licenses 147

Monitoring User Connections 147

Managing Licenses 149

About Managing User Connections and Licenses

You use the **Site Connections** tab in Site Administration to monitor and manage the users connected to your HP Application Lifecycle Management (ALM) projects. For more information, see "Monitoring User Connections" below.

You use the **Licenses** tab in Site Administration to view ALM license information and modify the license key. For more information, see "Managing Licenses" on page 149.

Monitoring User Connections

You can use the Site Connections tab to perform the following:

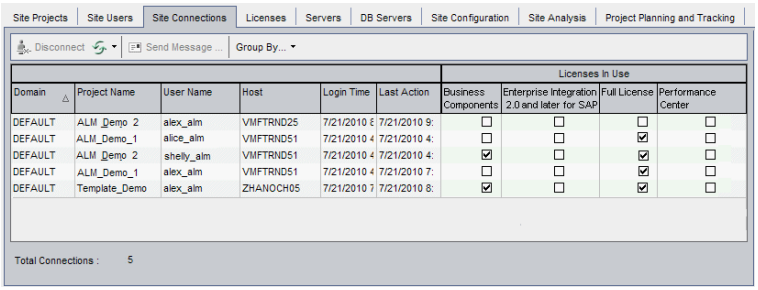
- Monitor the users currently connected to an ALM server. For each user, you can view the domain and project being used, the user's machine name, the time the user first logged in to the project, and the time the most recent action was performed. You can also view the client type connection to the ALM server.
- View the licenses that are used by each user.
- Send messages to users connected to your ALM projects. You can also disconnect users from projects.
- Modify access to an ALM project using the **Module Access** link. For more information, see "Customizing Module Access for User Groups" on page 296.

Note:

- To view the total number of licenses that are in use for each ALM module, click the **Licenses** tab. For more information, see "Managing Licenses" on page 149.
- To view and analyze the number of licensed ALM users connected to your projects at specific points over a period of time, click the **Site Analysis** tab. For more information, see "Monitoring Site Usage" on page 199.
- The ALM client can be inactive for a certain amount of time before it is disconnected from ALM. Disconnecting the client enables the license to be used by another ALM user. For more information, see the site administration configuration parameter "WAIT_BEFORE_DISCONNECT" on page 171.

To monitor user connections:

1. In Site Administration, click the **Site Connections** tab.



> **Tip:** You can click any column heading to change the sort order of the column from ascending to descending.

2. To refresh the Connections list, click the **Refresh Connections List** button .

   To instruct ALM to automatically refresh the Connections list, click the **Refresh Connections List** arrow and choose **Automatic Refresh**. By default, the Connections list is automatically refreshed every 60 seconds. To change the automatic refresh rate, click the **Refresh Connections List** arrow and choose **Set Refresh Rate**. In the Set Refresh Rate dialog box, specify a new refresh rate in seconds.

3. You can group connected users by clicking the **Group By** arrow, and choosing a Group By option. To group connected users by project, choose **Group By Project**. To group connected users by user, choose **Group By User**. To clear the Group By settings, click the **Group By** arrow and choose **Clear Group By**.

4. You can send a message to a connected user or group of users by clicking the **Send Message** button. For more information on sending messages, see "Sending Messages to Connected Users" below.

5. To disconnect a user or group of users from a project, select the row of the user or group and click the **Disconnect Users** button

. Click **Yes** to confirm.

Sending Messages to Connected Users

You can send messages to users connected to your ALM projects. This enables you to routinely inform connected users about important maintenance activities. For example, disconnecting a project, or restarting an ALM server.

When you send a message, a pop-up window automatically opens on the user's machine displaying the message text. The message box is displayed until the user closes it or disconnects from ALM. For more information, refer to the *HP Application Lifecycle Management User Guide*.

To send messages to connected users:

1. In Site Administration, click the **Site Connections** tab.
2. Select the users to whom you want to send a message:
   - To send a message to a user or group of users, select the row of the user or group.
   - To send a message to multiple users, use **Ctrl** or **Shift** to highlight users to include.
3. Click the **Send Message** button. The Send Message dialog box opens.

   The **To** box displays the intended recipients of the message in the format `[DOMAIN:Project Name:User Name]`.

   Example

   ```
   [DEFAULT:ApplicationLifecycleManagement_Demo:peter_alm]
   ```

4. In the **Message Text** box, type a message.
5. Click **Send**. ALM sends the message to user machines within five minutes.

Managing Licenses

In the Licenses tab, you can view the total number of licenses in use, the maximum number of licenses that you have for each ALM project or domain, and the expiration dates for the licenses. When other HP tools, such as UFT, are connected to an ALM project, you can view the total number of licenses in use for these tools. You can also add licenses. In addition, you can view the ALM edition installed on your server.

Within the Licenses tab, there are tabs for viewing and updating licenses:

- **Status**. You can modify licenses and link to the licensing portal to retrieve licenses.
- **License Assignments**. You can assign licenses to the various domains and projects.
- **Named Licenses**. You can assign named licenses to specific users.
- **PPU Licenses History**. You can view the current number of available PPU licenses and the history of their usage.

Note:

- To view the ALM licenses that are currently being used by each user, click the **Site Connections** tab. For more information, see "Monitoring User Connections" on page 147.
- To view and analyze the number of licensed ALM users connected to your projects at specific points over a period of time, click the **Site Analysis** tab. For more information, see "Monitoring Site Usage" on page 199.
- **Performance Center:** You can view additional Performance Center license information in Lab Management. For more information, refer to the *HP ALM Lab Management Guide*.

This section includes:

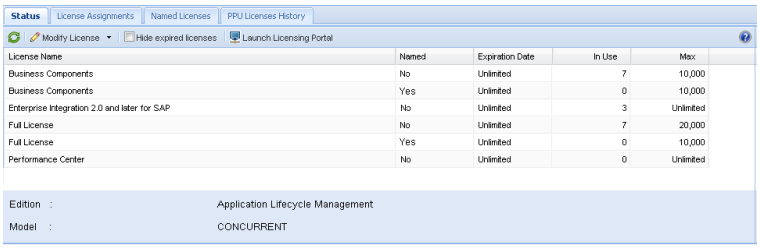
Modifying Licenses 149
Assigning Licenses to Domains and Projects 151
Assigning Licenses to Users 152
PPU Licenses History 154

Modifying Licenses

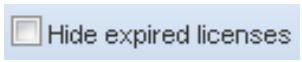
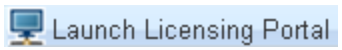
In the Status tab, you can modify licenses and launch the HP Licensing portal to retrieve licenses.

To modify licenses:

1. In Site Administration, click the **Licenses** tab.
2. Click the **Status** tab.



The Status tab includes the following UI elements:

UI Element	Description
	Refresh. Refreshes the screen.
	Enables you to modify the license by selecting one of the following: **Upload License.** Opens the Upload License dialog box, enabling you to browse for and select a license key. **Paste License.** Opens the Paste License dialog box, enabling you to paste the license key into a text box.
	Hides expired licenses in the table.
	Opens the HP Licensing portal in a new browser, enabling you to purchase licenses.
License Name	The ALM module name.
Named	Indicates whether the license is a named license assigned to a specific user.
Expiration Date	The expiration date of the license.
In Use	The total number of licenses in use.
Max	The maximum number of available licenses.
Edition	Indicates the ALM edition installed. For more information, refer to the *HP Application Lifecycle Management User Guide*.
Model	Indicates the license type. Possible values are: **CONCURRENT.** A license that allows perpetual usage of the purchased licenses until the license expiration date. **PPU.** A license that allows a month of actual usage of the purchased licenses. Used licenses expire at the end of the month in which they were used.

3. Click the arrow next to **Modify Licenses.**

4. Select **Upload License** or **Paste License**.
   - Select **Upload License** to open the Upload License dialog box, enabling you to browse for and select a license key.
   - Select **Paste License** to open the Paste License dialog box, enabling you to paste the license key into a text box.
5. To add licenses, click **Launch Licensing Portal**.

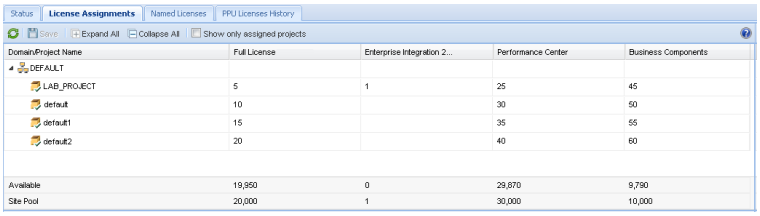
   The HP Licensing portal opens in a new browser, enabling you to purchase licenses.

Assigning Licenses to Domains and Projects

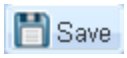
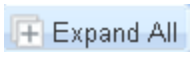
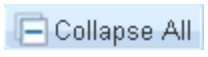
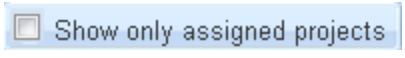
In the License Assignments tab, you can assign licenses to specific domains or projects to allow users assigned to each project to use the assigned licenses. The number of remaining, available licenses is shown towards the bottom of the tab. If a project needs more licenses than the number it is assigned, extra licenses can be taken from the domain or from the available licenses, until all available licenses are used.

To assign licenses:

1. In Site Administration, click the **Licenses** tab.
2. Click the **License Assignments** tab.



The License Assignment tab includes the following UI elements:

UI Element	Description
	Refresh. Refreshes the screen.
	Saves changes.
	Expands the domains and displays all project names.
	Collapses the domains and hides the project names.
	Hides the domains and projects that do not have licenses assigned to them.
Available	Number of available unassigned site level licenses by license type remaining.

UI Element	Description
Site Pool	Total number of available site level licenses by license type.

3. Select the domain or project to which you want to assign licenses. and enter the number of licenses you want to assign in the project or domain row.
4. Enter the number of licenses you want to assign in the entry in the selected row, according to the type of licenses you are assigning.

   The total of available licenses is reduced accordingly.

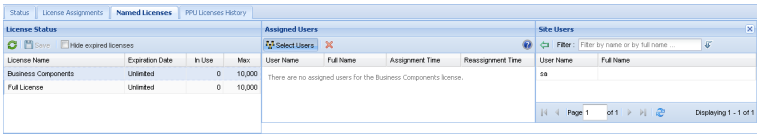
   ALM validates that the total number of licenses assigned to the projects and domains does not exceed the number of licenses in the site pool.
5. Click **Save** to save the changes.

Assigning Licenses to Users

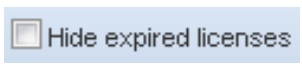
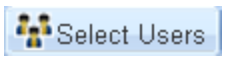
You can assign named licenses to specific users. The user has exclusive access to this license, but cannot use licenses from the Site Pool. The user must use this license for thirty days. After thirty days, the license assignment can be removed from the user. Also, when a named license is assigned to a user, you have one hour to remove the assignment.

To assign a license to a user:

1. In Site Administration, click the **Licenses** tab.
2. Click the **Named Licenses** tab.
3. Click **Select Users** to open the Site Users area to the right of the screen.



The Named Licenses tab includes the following UI elements:

UI Element	Description
	Refresh. Refreshes the screen.
	Saves changes and updates the Assigned Time column.
	Hides any expired licenses in the table.
	Opens the Site Users area to the right of the screen.

UI Element	Description
	Remove selected users. Removes the selected users from the Assigned Users area.
	Add selected users. Adds the selected users to the Assigned Users area.
	Apply filter. Applies the filter typed in the Filter field. To return to the full list, leave the Filter field empty and click **Apply filter** again.
	Pages forward or back in the list of site users.
	Refreshes the list of site users.
License Name	The ALM module name.
Expiration Date	The expiration date of the license.
In Use	The total number of licenses in use.
Max	The maximum number of licenses that you have for each ALM module.
User Name	The name of the user.
Full Name	The full name of the user.
Assignment Time	The date and time that the user was assigned to the license.
Reassignment Time	The date and time that the license assignment can be removed from the user. Once a license is assigned to a user, you have one hour to remove the assignment. After that time, the assignment cannot be removed from the user for thirty days. During the first hour and after thirty days, this field is valued with **Reassignable**.
Page	Displays the current page in the list of site users.

4. In License Status, select the license name.
5. In Site Users, select the user.
6. Click **Add selected users** to assign the user. The user name is displayed in the Assigned Users area.
7. Click **Save** to save the changes.

To remove a license assignment from a user:

Note: A license assignment can only be removed from a user when Reassignment Time has the value Reassignable.

1. In License Status, select the license.
2. In Assigned Users, select the user.
3. Click **Remove selected users** to remove the user. The user name is no longer displayed in the Assigned Users.area.
4. Click **Save** to save the changes.

PPU Licenses History

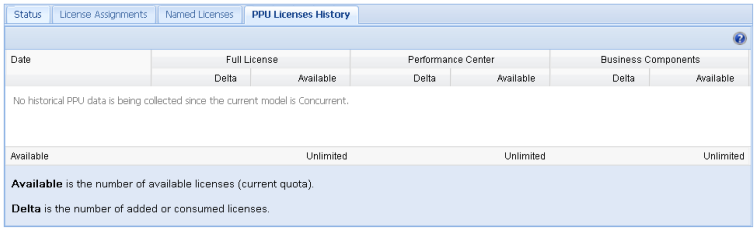
The usage of Pay-Per-Use (PPU) licenses is determined by the peak number of concurrent license usage that is recorded during the month. You can track peak usage in the current month in the Site Analysis tab. For more information, see "Monitoring Site Usage" on page 199.

At the beginning of each month, the number of used PPU licenses is subtracted from the total number of available licenses. If the remaining number of available licenses goes below the number of licenses you require, you can load additional licenses to meet the demand.

In the PPU Licenses History tab, you can view the number of available PPU licenses and the history of PPU license usage.

The information in the tab includes the following transactions:

- New purchases of PPU licenses, increasing the number of available licenses.
- The automatic recalculation of licenses at the beginning of each month, decreasing the number of available licenses.



The PPU Licenses History tab includes the following UI elements:

UI Element	Description
Date	The date on which a change in the number of available licenses is recorded. This occurs in one of the following cases: • Available licenses are calculated at the beginning of a month. • New licenses are added.

UI Element	Description
Delta	The number of licenses added or subtracted from the previous quota of available licenses. A Delta value is displayed separately for each module. • In the case of a calculation of available licenses, the delta displays a negative value, equal to the peak number of licenses that were used concurrently in the previous month. • In the case of adding new licenses, the delta displays a positive value, equal to the number of licenses added.
Available	The number of licenses available for concurrent use during the reported calendar month. An Available value is displayed separately for each module. The number of available licenses is equal to the number of available licenses in the previous row, plus the Delta value in the current row. The Available value can also indicate Unlimited, meaning there is no limit to the number of resources available.
Recommended to buy	A suggested number of additional licenses you should buy, assuming the same level of usage as in the previous month. A Recommended to buy value is displayed separately for each module. This value is equal to the number of licenses used in the previous month (Delta), minus the current amount of available licenses. If the amount is less than zero, nothing is displayed. Recommended to buy is not displayed if the number of licenses is unlimited.

Chapter 8: Configuring Servers and Parameters

You use Site Administration to configure HP Application Lifecycle Management (ALM) servers, define and modify database servers, configure the text search, set configuration parameters, and define the ALM mail protocol.

This chapter includes:

About Configuring Servers and Parameters 157

Configuring Server Information 157

Defining New Database Servers 159

Modifying Database Server Properties 161

Configuring Text Search 163

Setting ALM Configuration Parameters 167

Setting the ALM Mail Protocol 196

About Configuring Servers and Parameters

You use the **Servers** tab to configure ALM server information. You can set the server log files and maximum number of database handles. For more information, see "Configuring Server Information" below.

You use the **DB Servers** tab to define database servers that were not defined during installation. For each database server, you enter the database type, database name, default connection string, and administrator user and password.

You also use the **DB Servers** tab to modify existing database server definitions. For more information, see "Modifying Database Server Properties" on page 161. In addition, you can configure the text search option for a specified database server that has the text search feature installed and configured. For more information, see "Configuring Text Search" on page 163.

You use the **Site Configuration** tab to add and modify ALM configuration parameters. For more information, see "Setting ALM Configuration Parameters" on page 167. In addition, you can set the mail protocol to be used by all the server nodes in your ALM site. For more information, see "Setting the ALM Mail Protocol" on page 196.

Configuring Server Information

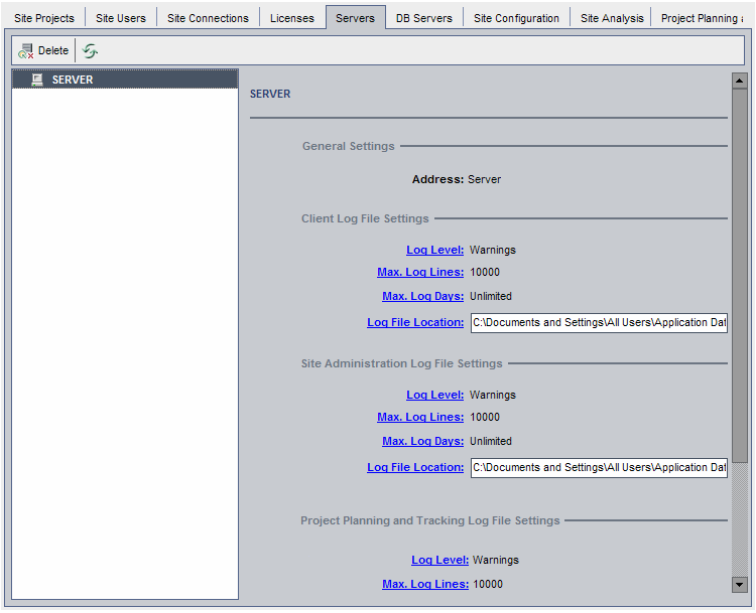
You can configure ALM server information. This includes:

- **Setting the ALM server log files.** ALM can write all ALM and Site Administration events to log files. The log files display the date and time a function was run. This is useful for when you contact ALM support.
- **Setting the maximum number of database connections.** ALM can open a number of connections for each project on a database server. You can set the maximum number of connections per active request that can be opened by ALM for each project.

ALM Editions: Functionality related to project planning and tracking (PPT) is available for ALM Edition only.

To configure ALM server information:

1. In Site Administration, click the **Servers** tab.



2. In the Server list, select a server.

   The **General Settings** area displays the server name.

3. You configure ALM and Site Administration log file settings under the **Client Log File Settings**, **Site Administration Log File Settings** and **Project Planning and Tracking Log File Settings** sections, respectively.

   Click the **Log Level** link to configure the type of log file you want the server to create. Select one of the following options in the Log Level dialog box:

   - **None.** Does not create a log file.
   - **Errors.** Records error events.
   - **Warnings.** Records potentially harmful situations.
   - **Flow.** Records informational messages that highlight the application flow.
   - **Debug.** Records events that are most useful for debugging.

4. Click the **Max. Log Lines** link to open the Maximum Log Lines dialog box and configure the maximum number of lines that ALM can write to the log file. ALM creates a new log file after the log file reaches the maximum number of lines. The default value is **10,000**.

5. Click the **Max. Log Days** link to open the Maximum Log Days dialog box and configure the maximum number of days that the ALM server keeps the log file. ALM automatically deletes the log files once the maximum number of days is reached. The default value is **Unlimited**.

6. Click the **Log File Location** link to change the directory path of the log file. In the Log File Location dialog box, type the new location for the log file.

   Saving logs on a remote server may cause performance issues. It is therefore recommended that you save logs to the local file system. If the local file system has limited space, prepare a script which periodically moves the logs to network storage

7. You can set the maximum number of connections per active request that can be opened on the database server by the ALM server for each project. Click the **Max. Database Connections** link to open the Maximum Database Connections dialog box and set the maximum number of concurrent connections. For information on maximum database connections, see http://h20230.www2.hp.com/selfsolve/document/KM630397.

   Note: In addition to changing the number of connections that can be opened on the database server for each project, you can also change:

   - The number of users allowed to connect concurrently to a domain. For more information, see "Creating Domains" on page 36.
   - The number of users allowed to connect concurrently to a project. For more information, see "Updating Project Details" on page 73.

8. To remove an ALM server from the Server list, select it and click the **Delete Server** button

. Click **Yes** to confirm.

9. Click the **Refresh Servers List** button to refresh the servers list.

Defining New Database Servers

You can define additional database servers that were not defined during the installation process.

Notes:

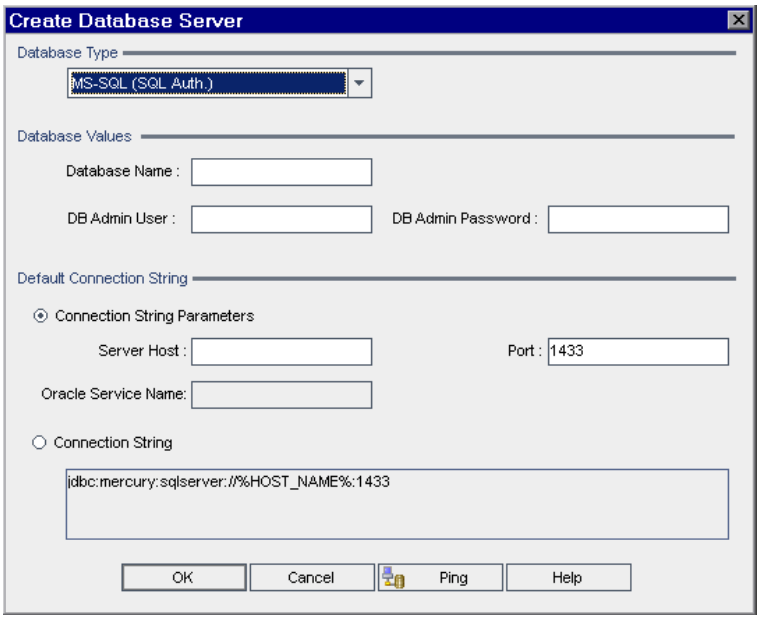
- For information on the Oracle or Microsoft SQL permissions required by ALM, refer to the *HP Application Lifecycle Management Installation and Upgrade Guide*.
- To make text search unavailable on a new database server, you must disable text search on the database server before you define the new database server in ALM.

To define a new database server:

1. In Site Administration, click the **DB Servers** tab.

2. Click the **New Database Server** button

. The Create Database Server dialog box opens.



3. Under **Database Type**, select the type of database server you want to define:
   - **MS-SQL (SQL Auth.).** Uses SQL authentication.
   - **MS-SQL (Win Auth.).** Uses Microsoft Windows authentication.
   - **Oracle.**
4. Under **Database Values**, in the **Database Name** box, type the database name.
5. In the **DB Admin User** box, type the login name of the database administrator.
   - For Oracle database type, the default administrator user account enabling you to create ALM projects is **system**.
   - For MS-SQL (SQL Auth.) database type, the default administrator user account enabling you to create ALM projects is **sa**.
   - For MS-SQL (Win Auth.) database type, the **DB Admin User** box is unavailable. The login name of the database administrator is the Windows user that is set to run ALM as a service.
6. In the **DB Admin Password** box, type the password of the database administrator. This field is unavailable if you selected the **MS-SQL (Win Auth.)** database type.
7. Under **Default Connection String**, you can edit the default connection string parameters or the connection string, as follows:
   - To edit the default connection string parameters, choose **Connection String Parameters** and define the following parameters:

Parameter	Description
Server Host	The server name.
Port	The port number of the database server.
Oracle Service Name	The service name for an Oracle database server.

- To edit the connection string, choose **Connection String** and edit the connection string.
- For Oracle RAC support, enter a connection string, using the following example:

  ```
  jdbc:company:oracle:TNSNamesFile=<ALM server>\tnsnames.ora;
  TNSServerName=OrgRAC
  ```

  - **tnsnames.ora** is a file containing Oracle database addresses. For more details, refer to HP Application Lifecycle Management Installation and Upgrade Guide.
  - **OrgRAC** is the address of the TNS server ALM should refer to.

  Note: To enable Oracle RAC support, you must set the **ORACLE_RAC_SUPPORT** site administration parameter to "Y". For more information, see "Setting ALM Configuration Parameters" on page 167.

- To check whether you can connect to the database server, click the **Ping** button. The DB admin user and password you entered are displayed in the Ping Database Server dialog box. Click **OK**.

8. Click **OK** to close the Create Database Server dialog box.
9. If necessary, click the **Refresh Database Servers List** button to refresh the database servers list.

Modifying Database Server Properties

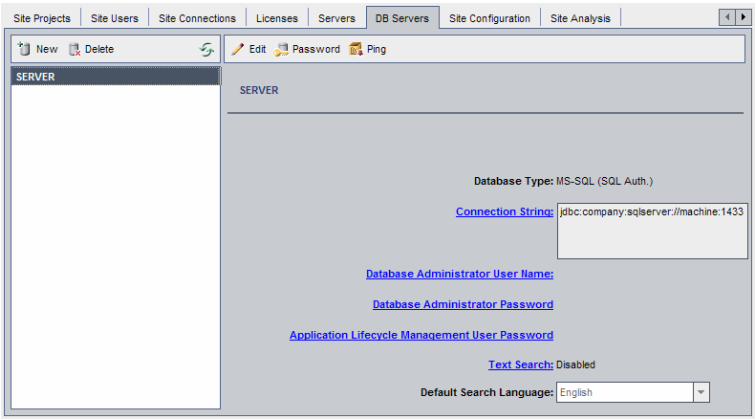
You can modify the database server properties.

Notes:

- For information on the Oracle or Microsoft SQL permissions required by ALM, refer to the *HP Application Lifecycle Management Installation and Upgrade Guide*.
- You can configure the site administration database schema for Oracle RAC support. For more information, refer to the *HP Application Lifecycle Management Installation and Upgrade Guide*.
- For debugging issues related to closure, see "DEBUG_CLOSURE_LOG_DOM_PROJ" on page 176.

To modify database server properties:

1. In Site Administration, click the **DB Servers** tab.



2. Select a database server in the Database Servers list.

3. To modify the connection string, click the **Edit Connection String** button

, or click the **Connection String** link. Edit the connection string in the Connection String Editor and click **OK**. For more information on connection strings, see "Defining New Database Servers" on page 159.

4. To modify the database administrator's login name, click the **Database Administrator User Name** link. In the Database Administrator User Name dialog box, type the new login name and click **OK**.

   For more information on defining a new login name for a database administrator, see step 5 under "Defining New Database Servers" on page 159.

5. To modify the database administrator's password, click the **DatabaseAdministrator Password** button

, or click the **Database Administrator Password** link. In the Database Administrator Password dialog box, type the new password and then retype it. Click **OK**.

6. To modify the default ALM user password for accessing the database schema, click the **Application Lifecycle Management User Password** link. In the User Password dialog box, type the new password and retype it. Click **OK**.

   Note: If you change the ALM user password, you must update the user password in the database server accordingly.

7. To enable text search capabilities in ALM, click the **Text Search** link.

   If the text search is enabled, you can set the default text search language for the database server in the **Default Search Language** list.

   For more information on text search, see "Configuring Text Search" on next page.

8. To check whether you can connect to the database server, click the **Ping Database Server** button. The DB admin user and password you entered are displayed in the Ping Database Server dialog box. Click **OK**.

9. To delete a database server from the Database Servers list, select it and click the **Delete Database Server** button . Click **Yes** to confirm.
10. Click the **Refresh Database Servers List** button to refresh the database servers list.

Configuring Text Search

Text search allows users to enter keywords and search specific project fields in the Requirements, Test Plan, and Defects modules. For information on working with the text search feature, refer to the *HP Application Lifecycle Management User Guide*.

To configure text search:

- Perform setups on each database user schema on which you want to enable text search. For more information, see "Enabling Text Search on Database User Schemas" below.
- In Site Administration, enable text search and define the default search language for a specified database server in the DB Servers tab. For more information, see "Enabling Text Search in ALM" below.
- To specify a different search language for a specific project, change the search language from the Site Projects tab. For more information, see "Selecting a Text Search Language for a Project" on next page.
- For a specific project, define the project fields to be included in the search from Project Customization. For more information, see "Defining Searchable Fields" on page 165.

Enabling Text Search on Database User Schemas

Before you enable text search in ALM, you must perform a set up step on each database user schema on which you want to enable text search.

To enable text search on an Oracle database user schema:

As Admin user, run the following command:

```
GRANT CTXAPP to <database user schema>
```

To enable text search on an SQL database user schema:

Enable full text indexing:

```
EXEC sp_fulltext_database 'enable'
```

Enabling Text Search in ALM

In Site Administration, you can enable text search for a specified database server that has the text search feature installed and configured. You can enable text search on a database server before or after you add projects to your Projects list.

If you enable text search on a database server before you add projects, the projects that you add afterwards are text search enabled. If you enable text search on a database server after you have added projects, you must manually enable text search for each existing project.

After you have enabled the text search for a specified database server, you set the default search language for the database server. You can change the default search language for a specific project from the Site Projects tab. For more information, see "Selecting a Text Search Language for a Project" below.

To enable text search on a database server before adding projects:

1. In Site Administration, click the **DB Servers** tab.
2. In the Database Servers list, select a database server.
3. Click the **Text Search** link.

   Caution: You cannot disable the text search after you have enabled it.

   Click **Yes** to confirm. The **Text Search** value changes from **Disabled** to **Enabled**.
4. In the **Default Search Language** list, set the default text search language for the database server.

To enable text search on a database server after adding projects:

1. In Site Administration, click the **DB Servers** tab.
2. In the Database Servers list, select a database server.
3. Click the **Text Search** link.

   Caution: You cannot disable the text search after you have enabled it.

   Click **Yes** to confirm. The **Text Search** value changes from **Disabled** to **Enabled**.
4. In the **Default Search Language** list, set the default text search language for the database server.
5. Click the **Site Projects** tab, and select a project for which you want to enable text search.
6. In the **Project Details** tab, click the **Enable/Rebuild Text Search** button to enable and rebuild the text search indexes. Click **Yes** to confirm.

   If the operation of enabling and rebuilding the text search indexes times out before it completes, you can change the default timeout value by defining the "TEXT_SEARCH_TIMEOUT" on page 194 parameter.
7. To enable text search for additional projects, repeat the previous two steps.

Selecting a Text Search Language for a Project

For each project, you can specify a search language other than the default search language you set for the database server. For more information on enabling the text search and setting the default search language, see "Enabling Text Search in ALM" on previous page.

Note: Search languages are not available for a project created on a database server that does not have the text search feature enabled.

To select a search language for a project:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a project. In the right pane, click the **Project Details** tab.
3. In the **Search Language** field, select a language for the project. For more information on updating project details in the Project Details tab, see "Updating Project Details" on page 73.

Defining Searchable Fields

For each project, you must define the fields to be included in the search in project customization. The searchable option is only available in the Requirement, Test, Business Component, Business Process Models, and Defect entities.

The following are available as searchable fields:

- User-defined fields with field type **Memo** or **String**
- The following system fields.

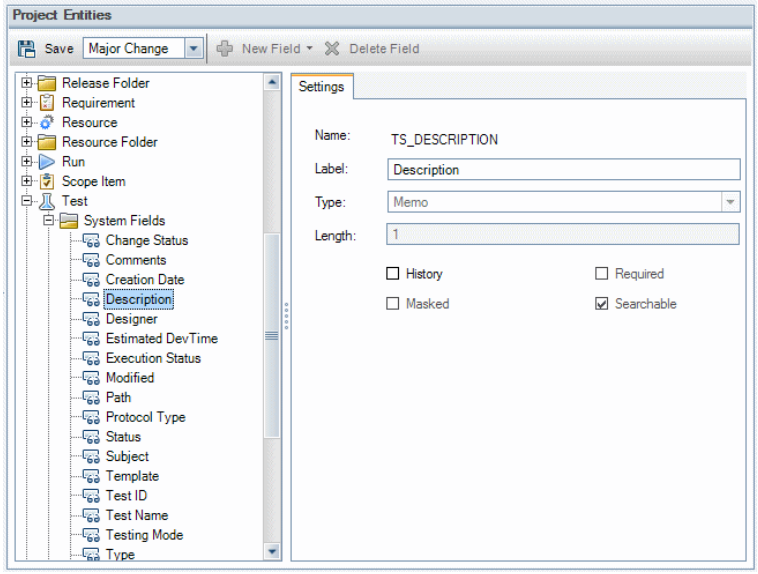
Searchable System Fields

Entity	Searchable System Fields
Business Component	• Comments • Component Folder • Component Name • Description • Original Location • Subtype ID
Business Process Model Activities	• Comments • Description • Text
Business Process Models	• Comments • Description
Defect	• Comments • Description • Summary

Entity	Searchable System Fields
Requirement	• Comments • Description • Name • Rich Text
Run	• Comments
Test	• Comments • Description • Path • Protocol Type • Template • Test Name

To define a searchable field:

1. In the ALM main window, select **Tools > Customize** from the common toolbar. The Project Customization window opens.
2. Click the **Project Entities** link. The Project Entities page opens. For more information on customizing project entities, see "Customizing Project Entities" on page 299.
3. Expand an entity, and select a system or user-defined field that can be made searchable.



4. Select the **Searchable** check box.
5. Click **Save** to save your changes to the Project Entities page.

Setting ALM Configuration Parameters

You can set the default ALM configuration parameters and add optional ones.

This section includes:

Default ALM Parameters 167
Optional ALM Parameters 171
Setting ALM Parameters 196

Default ALM Parameters

You can set the following default site configuration parameters:

Parameter	Description
ADD_NEW_ USERS_FROM_ PROJECT **(formerly CUSTOM_ ENABLE_USER_ ADMIN)**	If this parameter is set to **N**, you can add new ALM users from Site Administration (**Site Users** tab) only. If this parameter is set to **Y** (default), new ALM users can also be added from Project Customization. In the Project Users page, click **Add User**. The Add User to Project dialog box opens. If this parameter is set to **Y**, a **New** button is available for adding new ALM users. For more information, see "Adding a User to a Project" on page 249.
ATTACH_MAX_ SIZE	The maximum size (in kilobytes) of an attachment that can be sent with email from ALM. If the attachment size is greater than the specified value, the email is sent without the attachment. By default, the maximum email attachment size is 3,000 KB. For details, see "About Setting Automail" on page 315.
AUTO_MAIL_ WITH_ ATTACHMENT **(formerly SAQ_ MAIL_ WITH_ ATTACHMENT)**	If this parameter is set to **Y** (default), defect email is sent with attachments. This applies only if you select **Send mail automatically** in the **Site Projects** tab. For more information, see "Configuring Automail" on page 314. **Note:** The former parameter name is supported for purposes of backward compatibility.

Parameter	Description
AUTO_MAIL_ WITH_ HISTORY **(formerly SAQ_ MAIL_ WITH_HISTORY)**	If this parameter is set to **Y** (default), defect email is sent with the history. This applies only if you select **Send mail automatically** in the **Site Projects** tab. For more information, see "Configuring Automail" on page 314. **Note:** The former parameter name is supported for purposes of backward compatibility.
BASE_ REPOSITORY_ PATH	The base repository path. The ALM and Site Administration repositories are sub-folders of this repository. If you change this parameter value, new projects you create are stored in this location. After you change the value of this parameter, you must restart all servers in the cluster. The initial repository path is set during ALM server configuration. For more information, refer to the *HP Application Lifecycle Management Installation and Upgrade Guide*.
COMMUNICATION_ SECURITY_ PASSPHRASE	Communication between HP ALM and other HP BTO applications is enabled after authentication by a Single Sign-On (SSO) token through REST API. This parameter contains the passphrase that ALM uses to encrypt the SSO token. The initial parameter value is the SSO communication security passphrase that was entered during the ALM server configuration. For details, refer to the *HP Application Lifecycle Management Installation and Upgrade Guide*. Changing the COMMUNICATION_SECURITY_PASSPHRASE parameter on ALM requires that the equivalent value on other servers will be updated as well, for example, on Performance Center server and Host machines.
CREATE_HTTP_ SESSION	You can use this parameter if you are working with load balancing over a cluster of application servers. If the parameter is set to **Y**, ALM creates an HTTP session. This causes the load balancer to operate in sticky mode, meaning that after a request sent by a client is directed to a particular node in the cluster, all subsequent requests sent by that client are directed to the same node. By default, this parameter is set to **N**. For details, see the *HP Application Lifecycle Management Installation and Upgrade Guide*.
DISABLE_ VERBOSE_ ERROR_ MESSAGES	This parameter is a security feature that controls the level of detail that error messages display. If the parameter is set to **N**, the user can view system details connected to the error. By default, this parameter is set to **Y**.

Parameter	Description
EVENT_LOG_ PURGE_ PERIOD_DAYS	**Performance Center:** The time interval in days that deletable events remain in the EVENT_LOG database table. By default, the value is set to 60. If you set the value to **-1**, the events period is unlimited. For details, see the *HP ALM Performance Center Guide*.
LDAP_SEARCH_ USER_ CRITERIA	A comma-separated list of ALM user properties to be used as LDAP search criteria, if the Domain Authentication property does not contain the user's distinguished name (DN). The order of the properties defines their priority if multiple results are found. The following are the possible values: **username**, **email**, **fullname**, **phone**, **description**. For more information on LDAP, see "Enabling LDAP Authentication for Users" on page 140.
LIBRARY_FUSE	The parameter value indicates the base number for calculating the maximum number of entities for a library, in order to maintain optimal performance. By default, the value is 2500. Calculation is performed as follows: • Maximum number of tests in a library = LIBRARY_FUSE * 1 (2500 by default) • Maximum number of resources in a library = LIBRARY_FUSE * 0.25 (625 by default) • Maximum number of business components in a library = LIBRARY_ FUSE * 0.25 (625 by default) A validation of this value is performed when you create baselines, import libraries, or synchronize libraries. For information on a related parameter, see "REQUIREMENTS_ LIBRARY_FUSE" on page 191.
LOCK_TIMEOUT	The maximum number of hours that ALM objects can remain locked. After this time, the lock is removed. By default, the value is set to **10** hours.
MAIL_FORMAT	The format ALM uses to send email. By default, the format is set to **HTML**. To instruct ALM to send email as plain text, change the value to **Text**.
MAIL_INTERVAL	The time interval in minutes for sending a defect email according to your mail configuration settings. By default, the value is set to 10 minutes. Note that this applies only if you select **Send mail automatically** in the **Site Projects** tab. For more information, see "Updating Project Details" on page 73 and "Configuring Automail" on page 314.
MAIL_MESSAGE_ CHARSET	The character set used by ALM to send email to users. By default, the value is set to **UTF-8**.

Parameter	Description
MAIL_PROTOCOL	Displays the mail service used to send email messages to users. To configure the mail protocol, use the **Settings** button. For more information, see "Setting the ALM Mail Protocol" on page 196.
MAIL_SERVER_ HOST	Displays the server name used by the SMTP mail service. To configure the server name, use the **Settings** button. For more information, see "Setting the ALM Mail Protocol" on page 196. For information on a related parameter, see "Optional ALM Parameters" on next page.
MAIL_SHOW_ SITE_NAME	Indicates whether the site name is displayed in the subject of the mail. This parameter can be either a project parameter or a site parameter. If the parameter is defined in both site and project tables, the project value is considered. The default value is **N**. For more information, see "Customizing the Subject of Defect Mail" on page 317.
REPORT_QUERY_ RECORDS_LIMIT	The maximum number of records that can be retrieved from the database for an Excel report. If you set the value to **-1**, the number is unlimited. For more information, see the *HP Application Lifecycle Management User Guide*.
REPORT_QUERY_ TIMEOUT	The maximum length of time in seconds that ALM waits for an SQL query for an Excel report to be executed. If the query takes longer than this time to be executed, the query is canceled. For more information, see the *HP Application Lifecycle Management User Guide*.

Parameter	Description
RESTRICT_ SERVER_ FOLDERS	This parameter enables you to access restricted-access server directories using the OTA **ExtendedStorage.ServerPath** property. If this parameter does not exist, or is set to **Y**, you can only use the **ExtendedStorage.ServerPath** property to access the following directories: • the Site Administration (**SA**) directory • the root directory for a project • the **attach** subdirectory for a project • the **baseline** subdirectory for a project • the **checkouts** subdirectory for a project • the **components** subdirectory for a project • the **hist** subdirectory for a project • the **resources** subdirectory for a project • the **StyleSheets** subdirectory for a project • the **tests** subdirectory for a project If this parameter is set to **N**, you can access all server directories using the **ExtendedStorage.ServerPath** property. For more information on this property, refer to the *HP ALM Open Test Architecture API Reference*. For more information about ALM project structure, see "Understanding the Project Structure" on page 35.
SITE_ANALYSIS	If this parameter is set to **Y** (default), you can track ALM license usage over time from the **Site Analysis** tab. If this parameter is set to **N**, the Site Analysis tab is unavailable. For more information, see "Analyzing Site Usage" on page 198.
SUPPORT_ TESTSET_END	If this parameter is set to **Y** (default), UFT closes automatically when a test set is finished running.
WAIT_BEFORE_ DISCONNECT	The time interval in minutes that the ALM client can be inactive before it is disconnected from the ALM server machine. Disconnecting the client enables the license to be used by another ALM user. By default, the value is set to **600** minutes. For performance reasons, it is recommended to set a value of at least 60 minutes. If you set the value to **-1**, ALM is not disconnected, regardless of how long the client is inactive. For more information, see "Monitoring User Connections" on page 147 and "AUTO_LOGOUT_ON_SERVER_DISCONNECT" on next page.

Optional ALM Parameters

You can add the following optional site configuration parameters:

Parameter (A – Z)	Description
ALLOW_MULTIPLE_ VALUES	This parameter determines whether the **Allow Multiple Values** check box is visible in the Project Entities page in Project Customization. If this parameter is set to **N**, then the **Allow Multiple Values** check box is unavailable. If this parameter does not exist or is set to "Y", then the **Allow Multiple Values** check box is available. For more information on the **Allow Multiple Values** check box, see "Customizing Project Entities" on page 299.
ALLOW_UPDATE_ USER_ PROPERTIES_ FROM_ CUSTOMIZATION	User details are set in Site Administration. A project administrator cannot change details of project users in Project Customization. If this parameter is set to **Y**, the project administrator is able to change the details of project users in Project Customization. This option may cause a security risk, as it allows the project administrator to replace a user's e-mail address with his own. By doing so, the project administrator can then use the Forgot Password link to reset and change a user's password. If this parameter does not exist, or is set to **N**, only the user can change his details in Project Customization. For details, see "Understanding the Project Customization Window" on page 243.
ASYNC_MAIL_ ENABLED	If this parameter is set to **Y** (default), emails are sent asynchronously. This means that the email is queued to be sent, and you can continue working. If an email is undeliverable: • An email notification is sent to you, if the mail server is up. • A warning is added to the Site Administration log. If this parameter is set to **N,** emails are sent synchronously—meaning, emails are sent immediately and you continue working only when the mail is sent successfully.
AUTO_LOGOUT_ON_ SERVER_ DISCONNECT	The ALM server can disconnect an ALM client session. This occurs if: • The site administrator disconnects the session. • The session is automatically disconnected, according to the inactivity time interval setting. For more information on setting a timeout, see "WAIT_BEFORE_DISCONNECT" on previous page. The ALM client machine displays a message, informing the user that the session has been disconnected. If this parameter is set to **Y**, the client machine also automatically performs logout actions and returns the user to the ALM Login window. This ensures that the user does not continue to work in a session which is no longer connected to the server. If this parameter is set to **N**, no logout action is performed on disconnect.

Parameter (A – Z)	Description
AUTO_MAIL_ SUBJECT_ FORMAT **(formerly SAQFORMAT)**	This parameter enables you to customize the subject line of defect email sent automatically to users. For example, you can define a subject line such as `Defect no. 4321 has changed` by providing the value `Defect no. ?BG_BUG_ID has changed` , where **Defect no.** and **has changed** are strings, and **BG_BUG_ID** is an ALM field name. To customize the subject line for a specific project, see "Customizing the Subject of Defect Mail" on page 317. **Note:** The former parameter name is supported for purposes of backward compatibility.
AUTO_MAIL_USER_ NOTIFICATION	This parameter enables you to prevent ALM sending automatic email notification to project administrators when users are assigned or removed from a project in Site Administration. If this parameter is set to **N**, then automatic notification is not sent to project administrators. If this parameter does not exist, is empty, or is set to **Y**, then automatic notification is sent. For more information on assigning users to projects, see "Assigning Users to Projects" on page 77.
BACKWARD_ SUPPORT_ ALL_DOMAINS_ PROJECTS	This parameter enables the use of DomainsList and ProjectsList properties for the purposes of backward compatibility. If this parameter is set to **Y**, then the DomainsList and ProjectsList properties are supported. If the parameter does not exist or is empty, the default value is **N**, and these properties are not supported.
BACKWARD_ SUPPORT_ SA_DEFAULT_USER	This parameter enables the use of the old connection method to Site Administration for the purposes of backward compatibility. To work with scripts that use the old connection method (where the site administrator only required a password to log in), a user should be defined, and this user's password is used during login. The value of this parameter is a user name, whose password is used. If the parameter does not exist or is empty, an empty string is used. For details, see "Creating and Changing Passwords" on page 139.

Parameter (A – Z)	Description
BPT_WRAPPER_ TEST_ AUDIT	**Business Process Testing:** By default, ALM does not save the Business Process Testing wrapper tests it creates for running automated business process tests (or flows) with QuickTest Professional. This parameter enables you to save the Business Process Testing wrapper tests as attachments to the test or flow run for auditing purposes. The attachment is named **BPTWrapperTest.zip**. If this parameter is set to **N**, is empty, or does not exist, then Business Process Testing wrapper tests are not saved (default). If this parameter is set to **Y**, then Business Process Testing wrapper tests are saved. **Note:** Business Process Testing wrapper tests are not created under the following circumstances, even if this parameter is set to "Y": • When running a test or a flow from the Test Plan module. • When the test or flow has no components. • When running QuickTest Professional version 11.00 or earlier, if the test or flow contains at least one component which is not automated using QuickTest Professional (keyword driven or scripted). • When running automated business process tests or flows with UFT version 11.50 or later. For details, see the information about wrapper tests in the *HP Business Process Testing User Guide*.
BV_EXCEL_ REPORT_MAX_ ROWS	This parameter defines the maximum number of rows that can be included in a Business View Excel Report. The default value is **100,000**.
BV_EXCEL_ REPORT_SYNC_ MODE_MAX_ROW_ LIMIT	This parameter defines the maximum number of rows that can be generated in synchronous mode for a Business View Excel Report. The default value is **30**.
CLEAN_ORPHAN_ ANALYSIS_DATA_ JOB_SLEEP_ INTERVAL	This parameter defines how often orphan analysis data is cleaned up off the file repository. The data is cleaned up by the Orphan Analysis Data File Cleanup job. Define a value in minutes where the minimum is **10080** (one week). The default value is **43200** (one month).

Parameter (A – Z)	Description
COPY_CHANGES_USER_FIELDS **(formerly COPY_PASTE_CHANGES_OWNER)**	This parameter enables you to specify that the user who copies a record is listed in the specified User List fields of the copy. For more information on fields that have User List as their Field Type, see "Customizing Project Entities" on page 299. The value of this parameter is a comma-separated list of User List fields. For example, set the value of the parameter to `BG_DETECTED_BY`. Assume defect `10` is detected by user `Cecil_qc`, and user `Shelly_qc` copies defect `10`. ALM creates a copy of the defect with `Shelly_qc` as the user who detected the defect, not `Cecil_qc`.
CUSTOM_HELP_MENU_LINK	This parameter enables you to add a custom entry to the Help menu that links to a URL address. For example, if you want to allow users local access to ALM product movies, you can save the movies on the server, and create a link to a movies index page. Use the following syntax to enter a parameter value: `<link alias>;<URL>`, where the values of both `<link alias>` and `<URL>` are surrounded by quotation marks, and separated by a semicolon. For example, set the value of the parameter to `"MyBusiness - Online Help Page";"http://mybusiness/ALMHelp"`. The above example adds the `MyBusiness - Online Help Page` entry to the Help menu. Clicking the entry opens a custom Web page located at `http://mybusiness/ALMHelp`.
CUSTOM_PREREQUISITES_PAGE_URL	This parameter enables you to handle missing prerequisites during the deployment phase of starting the ALM client. The value of this parameter is either: • A valid URL to a page that contains links for downloading alternate prerequisites. • **NO_URL** or blank. The deployment phase opens the default URL. The default value is blank. **Note:** Setting a separate URL for each prerequisite is not supported. The page must contain information for all prerequisites.

Parameter (A – Z)	Description
DASHBOARD_ PAGE_ ITEM_LIMIT	By default, dashboard pages can include up to four graphs. This parameter enables you to set a different number of maximum graphs that dashboard pages can include. Increasing the number of graphs may reduce the system's performance. For more information on dashboard pages, refer to the *HP Application Lifecycle Management User Guide*.
DEBUG_CLOSURE_ LOG_ DOM_PROJ	This parameter enables the creation of a server log for debugging issues related to closure. This can be helpful when investigating missing or redundant results of closure queries, or suspected performance related issues. For details, see "Modifying Database Server Properties" on page 161. **Caution:** This parameter should not be enabled unless debugging is required by support for specific reasons. Delete this parameter if not in use for a specific debugging purpose. The closure log can be enabled per project, per domain, or per site. Use one of the following formats for the parameter value: • **Per project**: <domain name>;<project name> For example, `DEFAULT;project1` For more than one project in a domain, separate the projects with a semi-colon. For example, for two projects named project1 and project2 in the DEFAULT domain, use: `DEFAULT;project1; DEFAULT;project2` • **Per domain:** <domain name>;`___DEBUG_ALWAYS___` to indicate all projects in the specified domain For example, for the DEFAULT domain, use: `DEFAULT;___ DEBUG_ALWAYS___` For more than one domain, separate the domains with a semi-colon. • **Per site:**`___DEBUG_ALWAYS___;___DEBUG_ALWAYS___` to indicate all domains, and all projects in each domain **Note:** The `DEBUG_ALWAYS` value uses three underscores as a prefix and suffix. The log files are saved to the location for server log files, as specified in the Site Administration **Server** tab > **Client Log File Settings**. The log files are not purged automatically. You must delete the closure log files manually when the closure log is no longer required.

Parameter (A – Z)	Description
DISABLE_ COMMAND_ INTERFACE	If this parameter is set to **Y** (default), only users belonging to the TDAdmin group can use the OTA **Command** object. If it is set to **N**, any user can use it. If it is set to **ALL**, no users can use it. For more information, refer to the *HP ALM Open Test Architecture API Reference*.
DISABLE_ CONSOLE_ DEBUG_INFO	This parameter enables you to allow access to the ALM debug info console page (access to it is disabled by default). If this parameter exists and is set to **N**, the debug info console page can be accessed.
DISABLE_DEFAULT_ VALUES	If this parameter is set to **Y**, default values for certain entities (such as defects, tests, and test configurations) cannot be set per user per project. For more information, see the *HP Application Lifecycle Management User Guide*.
DISABLE_ EXTENDED_ STORAGE	This parameter controls user access to the OTA **ExtendedStorage** object. This is a security feature that can be used to limit access to the file system of the project. If this parameter is set to **Y** (default), the ExtendedStorage object cannot be accessed from TDConnection. Users can access the object from a specific entity for read-only, but no changes can be made. If it is set to **N**, the ExtendedStorage object can be accessed by all users, from a specific entity or from TDConnection. For more information about the ExtendedStorage object, refer to the *HP ALM Open Test Architecture API Reference*. **Note:** This parameter can impact how certain add-ins run, such as the HP Screen Recorder. For details, see the documentation for the individual add-ins, available from the HP Application Lifecycle Management Add-ins page (**Help > Add-ins**) or the HP Application Lifecycle Management Tools page (**Help > Tools**).
DISABLE_GET_ CHILDREN_LISTS_ WITH_VERSIONING	If this parameter is set to **Y**, this parameter disables certain performance enhancements that improve the speed at which various modules display data (such as entity trees). This parameter is relevant for version controlled projects. The default is **N**.

Parameter (A – Z)	Description
DISABLE_HTTP_ COMPRESSION	By default, the data transferred from the ALM server to clients is compressed to improve performance. If this parameter exists and is set to **Y**, the data compression is disabled.
DISABLE_ PASSWORD_ OTA_ENCRYPTION	By default, the OTA **TDConnection.Password** property is encrypted. If this parameter exists and is set to **Y**, encryption for this property is disabled. **Note:** Setting this parameter has no effect on password encryption during transport to the server machine.
DISPLAY_LAST_ USER_ INFO	This parameter enables you to add additional security to the client ALM Login window. By default, ALM displays the last user login information (user name, domain and project). If this parameter is set to **N**, the last user login information is not saved on the client machine and is not displayed in the ALM Login window. To activate this parameter, you must log in to ALM, log out, and log in again. If this parameter is set to **Y** or does not exist, the last user information is displayed. For details, see "Saving Customization Changes" on page 245.
DOWNLOAD_REST_ ATTACHMENTS	This parameter determines whether attachments are downloaded automatically in REST. If the value is **Y**, attachments are downloaded only after the browser prompts the user to open or save the attachment. If the value is **N**, the browser downloads the attachment automatically The default is **Y**.
EI_DELETE_ INTERVAL	This parameter defines the interval, in hours, between jobs that delete execution items from the database. If an execution item was marked as **Deleted** for more than 10 hours, it is deleted by the job. The default is 10.
ENABLE_CREATE_ DOCGEN_FAVORITE	This parameter defines whether new project documents can be added to your favorites list. The available options are: • **Y**. Project documents can be added to the favorites list. • **N**. Project documents cannot be added to the favorites list. By default, the parameter is set to **N**.

Parameter (A – Z)	Description
ENABLE_CREATE_LEGACY_EXCEL_REPORT	This parameter defines whether Microsoft Excel reports can be created in the Analysis View module. The available options are: • **Y**. Reports can be created. • **N**. Reports cannot be created. The default setting is **Y**.
ENABLE_CREATE_STANDARD_REPORT	This parameter defines whether standard reports can be created in the Analysis View module. The available options are: • **Y**. Reports can be created. • **N**. Reports cannot be created. Only existing reports are available. The default setting is **N**.
ENABLE_ENTITY_SELECTION_TREE_REQ_MOVE_TO	By default, the MoveTo event is not triggered when navigating up/down in a tree. However, in Quality Center version 10.00, the MoveTo event was triggered when navigating up/down in a Requirements tree. For backward compatability, this parameter enables you to define whether the MoveTo event is triggered in when navigating up/down in a Requirements tree. The available options are: • **Y**. The MoveTo event is triggered. • **N**. The MoveTo event is not triggered. The default setting is **N**.
ENTITY_LINK_HOST	This parameter enables you to set the mail server host name used in the link to an entity when ALM mails an entity. By default, ALM uses the default host name specified during installation. For details, see MAIL_SERVER_PORT.
ENTITY_LINK_PORT	This parameter enables you to set the mail server port number used in the link to an entity when ALM mails an entity. By default, ALM uses the default port number specified during installation.
EXTENDED_MEMO_FIELDS	This parameter extends the maximum number of memo-type user-defined fields per entity to 15 instead of five. The default value is **N**. To extend the number of memo-type fields, set the parameter to **Y**. For more information, see the Settings Tab in "Customizing Project Entities" on page 299, and "Adding User-Defined Fields" on page 302.

Parameter (A – Z)	Description
FAST_RECONNECT_MODE	This parameter defines options for reconnecting after a user session expires. Values include: **0.** Disables the reconnect option that bypasses reload of customization if no major change has been made. Users must manually log out and log in again when the session expires. **100** (default)**.** Password authentication is required. The user must enter a password to reconnect and continue working in ALM. **200.** The user does not need to enter password information to reconnect to ALM. User authentication is performed using the current password. If the user's password has been changed since the last log in, the user cannot reconnect. The user must log out and log in again using the new password. **Note:** If the user has been removed from the ALM Users list, the user cannot reconnect. For more information, refer to the *HP Application Lifecycle Management User Guide*.
FAVORITES_DEPTH	Defines the number of most recently used favorite views displayed on the **Favorites** menu. By default, ALM displays the four most recently used views on the menu. To hide the list of recently used views completely, set the parameter to **0**. For more information on favorite views, refer to the *HP Application Lifecycle Management User Guide*.
FETCH_LIMIT	To optimize performance, the maximum number of records retrieved and displayed in ALM grids is limited. This parameter enables you to change the default limit. If this parameter does not exist, the maximum number of records displayed is limited to 500. If the value of this parameter is set to **0**, all results are displayed. For information on a related parameter, see "GROUP_FETCH_LIMIT" on page 183. For information on configuring this value per project, see "Limiting Records Displayed in Grids" on page 100.

Parameter (A – Z)	Description
FILE_EXTENSION_ BLACK_LIST_ DOWNLOAD	The value is the semicolon delimited string with the file extensions that are not allowed to be downloaded via open attachments, REST API, or FTP Explorer. **Note:** • Imported or restored projects are not blocked even if their repository contains files of blocked file types. • Projects upgraded from previous versions still contain files of blocked file types that already exist in the repository. • During optimized repository migration, files of blocked file types are not blocked.
FILE_EXTENSION_ BLACK_LIST_ UPLOAD	The value is the semicolon delimited string with the file extensions that are not allowed to be uploaded via open attachments, extended storage, REST API, or FTP Explorer. **Note:** • Imported or restored projects are not blocked even if their repository contains files of blocked file types. • Projects upgraded from previous versions still contain files of blocked file types that already exist in the repository. • During optimized repository migration, files of blocked file types are not blocked.
FORCE_KEY_ AUTHENTICATION	If this parameter is set to **Y**, ALM uses secure session keys. If this parameter is empty or set to **N** (default), ALM does not enforce the secure mode. For example, this enables you to keep using existing LoadRunner scripts for load testing ALM.
FORCE_LOGIN_SSL_ MODE	If this parameter is set to **Y**, only the login process is sent over SSL (HTTPS). All other communication is sent without SSL (using HTTP). **Note:** ALM must be configured to work with SSL. For more information, refer to the *HP Application Lifecycle Management Installation and Upgrade Guide*. For information on a related parameter, see "LOGIN_SSL_PORT" on page 184.

Parameter (A – Z)	Description
FROM_EMAIL_ ADDRESS	If a user clicks the **Forgot Password** link in the ALM Login window, an email notification is sent to the user with a link to specify a new password. This parameter enables you to change the email address in the email **From** field. For more information on resetting passwords, refer to the *HP Application Lifecycle Management User Guide*.
FTP_PORT	The port number of the FTP service that enables you to browse ALM project repositories. For more information, see "Browsing the Project Repository" on page 83. Recommended values are **21** or **2121**. **Note:** • Defining the parameter starts the FTP service on the specified port. • Reconfiguring the parameter restarts the FTP service. • Deleting the parameter stops the FTP service.
GET_COVERAGE_ FROM_BL_FOR_ PINNED_TESTSET	This parameter enables you to get coverage from the baseline (and not from current view) when selecting tests for a requirement to be added to a pinned test set. The default value is **N**.
GRAPH_RESULTS_ LIFESPAN	When a graph is shared via the **Share Analysis Item** command in the Analysis View module, the graph retrieves data from a cache. By default, the cache information is updated every 60 minutes. This parameter enables you to configure the frequency, in minutes, at which the cache is updated. The minimum value you can assign to the parameter is 5 minutes. The maximum value is 60 minutes. For more details about sharing graph data, refer to the *HP Application Lifecycle Management User Guide*.

Parameter (A – Z)	Description
GROUP_FETCH_ LIMIT	To optimize performance, the maximum number of records retrieved and displayed per group when a **group by** filter is applied to ALM grids is limited. This parameter enables you to change the default limit. If this parameter does not exist, the maximum number of records displayed per group is limited to 100. If the value of this parameter is set to **0**, all results are displayed for each group. For information on a related parameter, see "FETCH_LIMIT" on page 180. For information on configuring this value per project, see "Limiting Records Displayed in Grids" on page 100.
INHERIT_MODULE_ ACCESS_TO_VIEWS	Some modules are logically connected to views in other modules. For example, the Business Models module is related to business linkage in the Requirements module and the Test Plan module. Project managers can hide access to modules for certain users/groups. This parameter indicates whether such related views should be hidden if their "parent" modules are hidden. **Note:** This parameter affects only views related to the Business Models module. Default: N (meaning, related views are not hidden)
LAB_MAX_DB_ HANDLERS	This parameter enables you to define the maximum number of connections to the database allowed for LAB_PROJECT. The default value is **100**.

Parameter (A – Z)	Description
LDAP_IMPORT_ ATTRIBUTE_MASK	This parameter enables you to define a regular expression that can be used to distinguish between different values for an LDAP attribute when importing users from an LDAP directory. When importing users, ALM chooses a value for the attribute that matches the regular expression. The parameter should be of the format: **<LDAP attribute name> = <regular expression>**, where **<LDAP attribute name>** is the name of the LDAP attribute whose value you want to choose, and **<regular expression>** is a regular expression. This regular expression should conform to the standard Java syntax for regular expressions. For example, a parameter value `uid=^\D\w+$` would choose values for the LDAP attribute `uid` consisting of a non-digit followed by any number of word characters (letters, numerals or the underscore character). For more information, see "Defining LDAP Settings for Importing Users" on page 134. For more information on importing users from an LDAP directory, see "Importing Users from LDAP" on page 132.
LDAP_RESULT_ SIZE_LIMIT	The maximum number of results that LDAP returns for a filtered query. Valid values are **100** or **10000**. The default value is **1000**. For more information about using LDAP, see "Importing Users from LDAP" on page 132.
LDAP_TIMEOUT **(formerly DIRECTORY_TIME_ LIMIT_ CONSTRAINT)**	The length of time, in milliseconds, that ALM waits before canceling an LDAP operation. The time limit on LDAP operations prevents a situation where LDAP encounters a problem and causes ALM to wait indefinitely. The default timeout value is 10 minutes. For more information about using LDAP, see "Importing Users from LDAP" on page 132.
LOGIN_SSL_PORT	If the FORCE_LOGIN_SSL_MODE parameter exists and is set to **Y**, this parameter enables you to configure the port used for SSL login. By default, the value is 443. For information on a related parameter, see "FORCE_LOGIN_SSL_ MODE" on page 181.

Parameter (A – Z)	Description
LR_ DIRECTFILEACCESS	This parameter applies if you are integrating with HP LoadRunner. If set to **Y**, it enables the direct accessing of scripts located within the same LAN as your ALM client/server. **Note:** In a UNIX or Linux environment, you must also set the UNIX_SERVER parameter.
MAIL_SERVER_ PORT	The SMTP server port used by ALM to send mail. By default, the value is set to 25. For information on a related parameter, see "MAIL_SERVER_HOST" on page 170.
MAX_ CONCURRENT_ REPORTS	This parameter defines the maximum number of project reports that can be generated concurrently. Define any value where the minimum is **1**. Where the maximum number has been reached, and you choose to generate an additional report, the report generation only begins once one of the original reports finishes generating. For example, if the maximum number is **5**, and you try generate an additional report, the sixth report only begins generating once one of the original five concludes generating. The default maximum is **3**.
MAX_GRAPH_ RESULT_DATA_ TABLE_VOLUME	This parameter enables you to change the maximum size of the graph results, as calculated by the number of cells in the graph's Grid View. The size of a cell is 8 bytes. The default maximum is **100** megabytes.
MAX_KPIS_PER_ MILESTONE	This parameter determines the maximum number of KPIs that can be defined per milestone. The default number is 30. For more information on adding KPIs to milestones, refer to the *HP Application Lifecycle Management User Guide*.
MAX_MILESTONES_ PER_RELEASE	This parameter determines the maximum number of milestones that can be defined per release. The default number is 20. For more information on defining milestones, refer to the *HP Application Lifecycle Management User Guide*.

Parameter (A – Z)	Description
MAX_QUERY_ LENGTH	This parameter enables you to change the maximum query length (number of letters). It includes the parameters and is valid for Oracle and for SQL. The default value is 1000000. Some reports are too large for the default limit query size to handle. In such cases, this parameter should be set to 100000000. For details, see "The Cross Project Customization Report" on page 339.
MAX_SCOPE_ ITEMS_PER_ RELEASE	This parameter determines the maximum number of scope items that can be defined per release. The default number is 20. For more information on defining scope items, refer to the *HP Application Lifecycle Management User Guide*.
MAX_TEST_ INSTANCES	This parameter enables you to limit the number of test instances handled by the server per call. The default value is 1000000. If there is a call with more test instances than this value, an error message will appear and the calls will be rejected causing the operation to fail.
MAX_TESTS_IN_ TEST_SET	This parameter enables you to limit the number of tests in a test set for copy/paste or delete functions. The default value is 20000. If there is a test set with more tests than this value, an error message will appear and the functions will not be performed.
MAX_THRESHOLD_ VALUES_PER_KPI	This parameter determines the maximum number of threshold values that can be defined per KPI. The default number is 12. For more information on defining thresholds, refer to the *HP Application Lifecycle Management User Guide*.
NLS_SEARCH_ LOCALE	The language used by the **Find Similar Defects** command to tokenize the defect summary. This parameter is needed only if the default locale on the server does not match the language in which the defect summary is written, in terms of whether spaces are used to separate words. The value should be a string value that matches a language code listed in ISO 639 (http://www.w3.org/WAI/ER/IG/ert/iso639.htm). For example, if the default locale is English and the text is in Japanese, which does not use spaces to separate words, set `NLS_ SEARCH_LOCALE=ja`. If this parameter is not defined or is invalid, the default locale of the server is used.
ORACLE_RAC_ SUPPORT	This parameter must be set to **Y** to enable RAC support on Oracle database servers. For more information, refer to the *HP Application Lifecycle Management Installation and Upgrade Guide*.

Parameter (A – Z)	Description
ORPHAN_ ANALYSIS_DATA_ FILE_INACTIVE_ PERIOD	The parameter defines how old an analysis data file must be in order to be considered an orphan. The file is considered to be an orphan if it's last modified property is older than the time frame you define as part of this parameter. Define a value in milliseconds where the minimum is **3600000** (one hour). The default value is **86400000** (one day).
PASSWORD_ RESET_ DISABLE	This parameter determines whether ALM users can reset their passwords using the **Forgot Password** link in the ALM Login window. If this parameter is not defined or if the parameter is set to 'N', users can reset their passwords using the **Forgot Password** link. If LDAP authentication is enabled, you must set this parameter to 'Y'. For more information, see "Enabling LDAP Authentication for Users" on page 140. For more information on resetting passwords, refer to the *HP Application Lifecycle Management User Guide*.
PASSWORD_ RESET_ ELAPSED_TIME	If a user clicks the **Forgot Password** link in the ALM Login window, by default 24 hours have to elapse before the same user can make another password reset request. This parameter allows you to change the length of time, in minutes, that has to elapse before a user can make another password reset request. For more information on resetting passwords, refer to the *HP Application Lifecycle Management User Guide*.
PASSWORD_ RESET_ SERVER	If a user clicks the **Forgot Password link** in the ALM Login window, an email notification is sent to the user with a link to specify a new password. This parameter allows you to override the default URL, or parts of the URL, that is embedded in the reset link. Use one of the following syntaxes: • **<server>:<port>.** Overrides both the default server and port. • **<server>.** Overrides the default server. • **<port>.** Overrides the default port. For more information on resetting passwords, refer to the *HP Application Lifecycle Management User Guide*.

Parameter (A – Z)	Description
PASSWORD_ RESET_ VALID_PERIOD	If a user clicks the Forgot Password link in the ALM Login window, an email notification is sent to the user with a link to specify a new password. By default, the link is valid for 24 hours. This parameter allows you to change the length of time, in minutes, that the link is valid. For more information on resetting passwords, refer to the *HP Application Lifecycle Management User Guide*.
PPT_KPI_ FAILURES_ PERCENTAGE_PER_ RELEASE_FUSE	This parameter determines the number of seconds ALM pauses before retrying a failed KPI calculation. The default value is 5 seconds. For more information on scheduling project planning and tracking calculations, see "Scheduling Calculations for Project Planning and Tracking (PPT)" on page 204.
PPT_KPI_ FAILURES_ PERCENTAGE_PER_ RELEASE_FUSE	By default, if 10% or more of the KPI calculations within the release fail, ALM aborts project planning and tracking calculations on a release and skips to the next release in a project. The default number is 10. For more information on scheduling project planning and tracking calculations, see "Scheduling Calculations for Project Planning and Tracking (PPT)" on page 204.
PPT_RECENTLY_ USED_PROJECTS_ THRESHOLD_ MINUTES	This parameter sets a threshold for controlling which projects are included in PPT calculations. PPT calculations are not performed for projects to which no one has logged in for the number of minutes specified The default is 10080 minutes (7 days). For more information on scheduling project planning and tracking calculations, see "Scheduling Calculations for Project Planning and Tracking (PPT)" on page 204.
PROJECT_ SELECTION_ MAX_PROJECTS	By default, cross-project graphs can include up to six projects. This parameter enables you to set a different number of maximum projects that cross-project graphs can include. Increasing the number of projects may reduce the system's performance. For more information on cross-project graphs, refer to the *HP Application Lifecycle Management User Guide*.
QC_SENSE_ AUTHORIZATION_ DISABLED	This parameter allows you to disable the authorization check for users that want access to QC Sense reports. The default value is **N** (meaning, the authorization check is enabled).

Parameter (A – Z)	Description
QC_SENSE_ REPORTS_ USERS	By default, only site administrator users have access to QC Sense reports. This parameter enables you to grant permissions to additional users who are not site administrators. Enter user names using the following syntax: `<user1>;<user2>;<user3>`

REPLACE_TITLE	This parameter enables you to change the names of ALM modules across all your projects.
	
	Rename one or more modules by entering the following parameter value:
	`<original title1 [singular]>;<new title1 [singular]>;`
	`<original title1 [plural]>;<new title1 [plural]>;`
	`<original title2 [singular]>;<new title2 [singular]>;...`
	
	For example, if you want to change the name of the Defects module to **Bugs**, and the Requirements module to **Goals**, enter the following:
	`Defect;Bug;Defects;Bugs;Requirement;Goal; Requirements;Goals`
	
	Renaming the Releases module does not change the module name in the following locations:
	• The **Releases** command in the Releases module menu bar.
	• The **New Release Folder** menu command and dialog box.
	• The **New Release** menu command and dialog box.
	
	Note: To rename the Defects module for a specific project only, see "Renaming the Defects Module for a Project" on page 99.

REPORT_MAX_ ALLOWED_SIZE	This parameter defines the estimated maximum number of pages that can be included in a project report. The default value is **1000**. **Note:** While a report it being generated, ALM can only estimate the number of pages. Therefore it cannot be guaranteed that the final generated report will not exceed this number of pages. To leave the maximum number of pages as unlimited, set the parameter to **-1**. However, this is not recommended as it may negatively impact on your server's performance.

Parameter (A – Z)	Description
REPORT_RESULTS_ LIFESPAN	When a project report is shared via the **Share Analysis Item** command in the Analysis View module, the report retrieves data from a cache. By default, the cache information is updated every 60 minutes. This parameter enables you to configure the frequency, in minutes, at which the cache is updated. The minimum value you can assign to the parameter is 5 minutes. The maximum value is 60 minutes. For more details about sharing graph data, refer to the *HP Application Lifecycle Management User Guide*.
REPOSITORY_GC_ DELAY_ CANDIDATE_TIME	This parameter enables you to create a delay between the time of the cleaning processes of each project repository and the time the files without references are actually deleted. Set a value in days between **0** and **28**. If the parameter does not exist, ALM delays the deletion of obsolete files in the file system for seven days. For details on the project repository cleanup process, see "Project Repository Cleanup" on page 83.
REPOSITORY_GC_ JOB _PRIORITY	The parameter determines the speed at which the repository cleanup process is performed. Set a value between **0** (fastest) and **10** (slowest). If the parameter does not exist, the speed is set to **3**. For more details on the project repository cleanup process, see "Project Repository Cleanup" on page 83.
REPOSITORY_GC_ PROJECT_ CLEANUP_ INTERVAL	This parameter defines the time interval in days between cleaning processes of each project repository. Set a value in days between **1** and **28**. If the parameter does not exist, ALM scans project repositories once every seven days. For more details on the project repository cleanup process, see "Project Repository Cleanup" on page 83.

Parameter (A – Z)	Description
REPOSITORY_ MIGRATION _JOB_PRIORITY	The parameter determines the speed at which files are copied from the old to the new project repository. Set a value between **0** (fastest) and **10** (slowest). If the parameter does not exist, the speed is set to **3**. For more details on the project repository migration process,see "Migrating the Repository" on page 121.
REQUIREMENT_ REVIEWED_FIELD_ AUTOMATIC_ UPDATE	If this parameter is set to **Y** (default), then any change to a requirement field automatically sets the **Reviewed (RQ_REQ_REVIEWED)** field to **Not Reviewed**. If it is set to **N**, then a change to a requirement field does not affect the value of the Reviewed field. For details on RQ_REQ_REVIEWED, see "Schema Issues" on page 496.
REQUIREMENTS_ LIBRARY_FUSE	This parameter indicates the maximum number of requirements for a library in order to maintain optimal performance. The default value for this parameter is 3500. A validation of this value is performed when you create baselines, import libraries, or synchronize libraries. For information on a related parameter, see "LIBRARY_FUSE" on page 169.
REST_API_ DEFAULT_PAGE_ SIZE	The number of entities per page returned by default with each GET operation on a collection using the REST API (unless the API consumer specifies a different page size for retrieval). The default is 100 entities.
REST_API_HTTP_ CACHE_ENABLED	This parameter enables REST API support for server side HTTP caching. If enabled, the server supports caching using an ETag for the following resources: • customization/entities • customization/relations • customization/used-lists • customization/users The default is **Y**.
REST_API_MAX_ BULK_SIZE	The maximum number of entities allowed for a single bulk operation. A bulk operation over REST means to be able to POST, PUT, or DELETE a collection of entities of the same type. The default is 2000 entities.

Parameter (A – Z)	Description
REST_API_MAX_ PAGE _SIZE	The maximum number of entities per page that can be returned with a single GET operation on a collection using the REST API. The default is 5000 entities.
REST_SESSION_ MAX_ IDLE_TIME	This parameter sets the REST API session maximum idle time (in minutes). The idle time is the amount of time a REST API session token remains valid if there is no activity in the session. After this period, the session (including the licenses and locks that it holds) expire. On the next call, the REST API recreates a new session. The default value is 60 minutes.
SD_RUN_AMOUNT	The amount of runs taken into account when calculating the suggested duration of a particular test set or build verification suite execution. The default value for this parameter is 10. The recommended threshold value is 100. A higher value could potentially affect performance.
SECURED_QC_URL	When ALM generates email, it includes a link to ALM in the email. If this parameter is set to **Y**, the ALM URL uses an SSL connection (starting with `https:`). If it is set to **N** (default), SSL is not used.
SEND_EXCEPTION_ DEFAULT_TO	This parameter defines the default email address for sending an error report in ALM. For information on a related parameter, see "SEND_EXCEPTION_ ENABLED" below. For more information on sending error details, see the *HP Application Lifecycle Management User Guide*.
SEND_EXCEPTION_ ENABLED	If this parameter is set to **Y**, the **Send Error Details** option is available on the **Help** menu in ALM, enabling you to send details of an error as attachments to an email. For information on a related parameter, see "SEND_EXCEPTION_ DEFAULT_TO" above. For more information on sending error details, see the *HP Application Lifecycle Management User Guide*.

Parameter (A – Z)	Description
SKIP_CLIENT_ PREREQUISITES_ CHECK	This parameter enables you to bypass the prerequisites check performed in the deployment phase of starting the ALM client. The default value is **N**. **Tip:** For similar functionality in MSI Generator, check the **Skip Prerequisites Check** checkbox.
SMTP_SSL_ SUPPORT	This parameter enables an SSL tunnel to the SMTP server. The default value is **N**.
SQL_QUERY_ VALIDATION_ BLACK_LIST	By default, ALM checks that SQL queries for an Excel reports do not include any of the following commands: **INSERT**, **DELETE**, **UPDATE**, **DROP**, **CREATE**, **COMMIT**, **ROLLBACK**, **ALTER**, **EXEC**, **EXECUTE**, **MERGE**, **GRANT**, **REVOKE**, **SET**, **INTO**, or **TRUNCATE**. This ensures that you do not inadvertently modify or delete records in the project database. You can modify which commands are on this list by adding this parameter. The parameter's value must be a comma-separated list of SQL commands that ALM should verify are not included in SQL queries for an Excel report. Note that this verification is not performed if the SQL_QUERY_ VALIDATION_ENABLED parameter exists and is set to **N**. For details, see "SQL_QUERY_VALIDATION_ENABLED" below.
SQL_QUERY_ VALIDATION_ ENABLED	By default, ALM checks SQL queries in Excel reports to ensure that they are valid and do not alter the project database. For more information on this validation, see the *HP Application Lifecycle Management User Guide*. If this parameter is set to **N**, this validation is not performed. If this parameter does not exist, is empty, or is set to **Y**, this validation is performed. For information on a related parameter, see "SQL_QUERY_ VALIDATION_BLACK_LIST" above.
SSO_EXPIRATION_ TIME	Expiration time of the LWSSO token (the authentication token of the REST API) in minutes. After this period of inactivity, the REST API consumer is required to re-authenticate. The default value is 60 minutes.
SUPPORT_ TESTSET_END	If this parameter is set to "Y", the Automatic Runner dialog box can signal a remote agent when a test set execution starts and ends. The events are passed using the Remote Agent's **Set_Value** method.

Parameter (A – Z)	Description
SUSPEND_ REPOSITORY_ GC	This parameter relates to the project repository cleaning process. For more details, see "Project Repository Cleanup" on page 83. The parameter stops the cleanup process on the entire site. Use this only in special circumstances. For example, to check if the cleanup process is impacting system performance. Define this parameter and set its value to **Y** to temporarily stop the project repository cleaning process. To restart the cleaning process, set the parameter to **N**.
SUSPEND_ REPOSITORY_ MIGRATION	This parameter relates to the project repository migration process. For more details, see "Configure Migration Priority" on page 125. The parameter stops the migration process on the entire site. Use this only in special circumstances. For example, to back up projects, or to check if the migration is affecting system performance. Define this parameter and set its value to **Y** to temporarily stop the migration of project files. To restart the migration, set the parameter to **N**.
STATIC_CONTENT_ MAX_CACHED_ FILE_SIZE	The maximum file size in MB that can be cached in the Jetty static content cache. You can reduce the cache file size, or cancel the cache by setting the parameter value to **0**.
TEXT_SEARCH_ TIMEOUT	The length of time, in seconds, that ALM waits before canceling the operation of enabling and rebuilding the text search indexes. This operation is activated by clicking the **Enable/Rebuild Text Search** button in the Site Projects tab of the Site Administration. The default timeout value is 20 minutes. For more information on configuring text search, see "Configuring Text Search" on page 163.

Parameter (A – Z)	Description
UNIX_SERVER	If this parameter is set to **Y**, it enables direct file access from a testing tool on a Windows machine to a UNIX based repository. You must then add a new parameter for each directory on the UNIX server machine you want to be able to access externally and specify the corresponding Windows path, as follows: • **Parameter name** is `FOLDER_MAPPING_`*n* where *n* is an identifying number. For example, `FOLDER_MAPPING_1` • **Parameter value** is in the format *UNIXpath*`->`*Windowspath* For example, `/opt/Mercury/repository/qc/->\\netapp\qc\repository\` **Note:** This parameter applies to HP LoadRunner.
UPGRADE_ EXCEPTION_ FILE	This parameter defines the location of the global exception file to be used when upgrading projects. This file defines exceptions for the ALM database user schema. By default, the **SchemaExceptions.xml** file is saved in the directory. For more information on upgrading projects, see "Upgrading Domains and Projects" on page 113.
UPLOAD_ATTACH_ MAX_SIZE	This parameter prevents the upload of attachments whose size is greater than the integer value specified for this parameter. The size is in kilobytes. **Note:** This parameter does not affect uploads of attachments in integration tools. The default value is blank (meaning, all attachments are uploaded regardless of size).
VERIFY_REPORT_ FOLDER	This parameter determines where the verification report is saved when the project verification process completes. By default, the output is saved in **<ALM repository path>\sa\DomsInfo \MaintenanceData\out** on your ALM server machine. For more information on verifying projects, see "Verifying Domains and Projects" on page 107.

Setting ALM Parameters

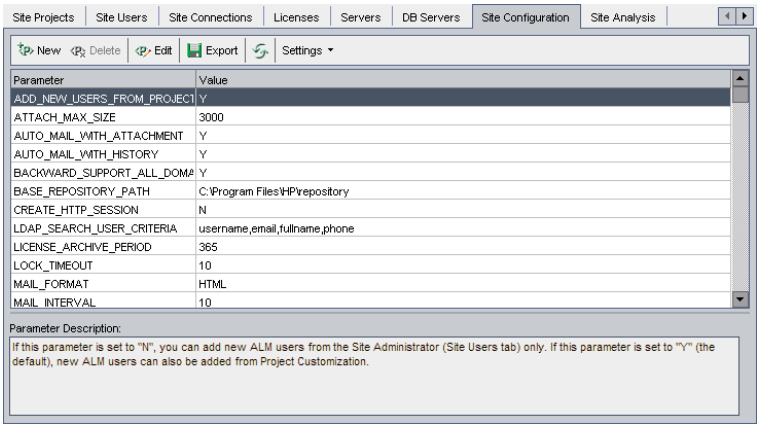
You can add, modify, and delete parameters in the Site Configuration tab. You can also export parameters to a text file.

Note:

- You cannot add or delete default parameters. You can only modify them.
- You must reconnect to any open projects to work with the new settings.

To set ALM parameters:

1. In Site Administration, click the **Site Configuration** tab.



2. To add a new parameter to the list, click the **New Parameter** button. The New Parameter dialog box opens. Type a name, value, and description for the parameter you want to add. Click **OK**.
3. To delete a parameter from the list, select it and click the **Delete Parameter** button. Click **Yes** to confirm.
4. To edit a parameter, select it from the list and click the **Edit Parameter** button. The Edit Parameter dialog box opens. Type a new value and value description, and click **OK**.
5. To export parameters from the site configuration grid to a text file, click the **Export** button. The Export Data To File dialog box opens. Select the directory where you want to save the parameters, and type a name for the file in the **File name** box. Click **Save**.
6. You can click the **Refresh Parameters List** button to refresh the parameter list.

Setting the ALM Mail Protocol

ALM uses email to send project information to users. You can select the mail service to be used by all the server nodes in your ALM site. ALM supports the SMTP mail protocols.

For more information on setting the ALM mail protocol, refer to the *HP Application Lifecycle Management Installation and Upgrade Guide*. To view the protocol currently configured, see the "MAIL_PROTOCOL" on page 170 parameter in the **Site Configuration** tab in Site Administration.

To set the ALM mail protocol:

1. In Site Administration, click the **Site Configuration** tab.
2. Click the **Settings** button and select **Set Mail Protocol**. The Set Mail Protocol dialog box opens.
3. Select one of the following options:
   - **None.** ALM does not send an email.
   - **SMTP Server.** ALM sends an email from an SMTP server on the network. Type the address of an SMTP server available on your local area network. For details, see the "MAIL_SERVER_HOST" on page 170 parameter.
   - **Microsoft IIS SMTP Service.** ALM sends an email from the ALM server machines. This option is available if you installed Microsoft IIS SMTP Service on your ALM server machines during IIS installation.
4. Click **Test** to send a test email to your mailbox. The Test Mail dialog box opens. Type an email address and click **Send**. A pop-up message confirms whether the mail was sent successfully.
5. Click **OK** to close the Set Mail Protocol dialog box.

Chapter 9: Analyzing Site Usage

In Site Administration, you can track the number of licensed users that have connected to your HP Application Lifecycle Management (ALM) site at specific points over a period of time. You can also analyze ALM usage by filtering the number of licensed users by projects, users or license types.

This chapter includes:

About Analyzing Site Usage 199

Monitoring Site Usage 199

Filtering Site Usage 200

Exporting Site Analysis Data to a File 201

Customizing the Site Analysis Line Chart Graph 201

About Analyzing Site Usage

You use the **Site Analysis** tab in Site Administration to monitor license usage for each time interval displayed. You can specify the time interval displayed along the x-axis. You can also specify what information appears in the graph by filtering the graph content by projects, users, or license types.

Example

You may want to charge each department in your organization according to license usage. You can filter by projects in a specific department to view license usage for the department. You can also view license usage for a specific group of users by filtering according to selected users.

If the **Site Analysis** tab is not displayed, you can make it available by editing the SITE_ANALYSIS parameter in the Site Configuration tab in Site Administration. For details, see "SITE_ANALYSIS" on page 171.

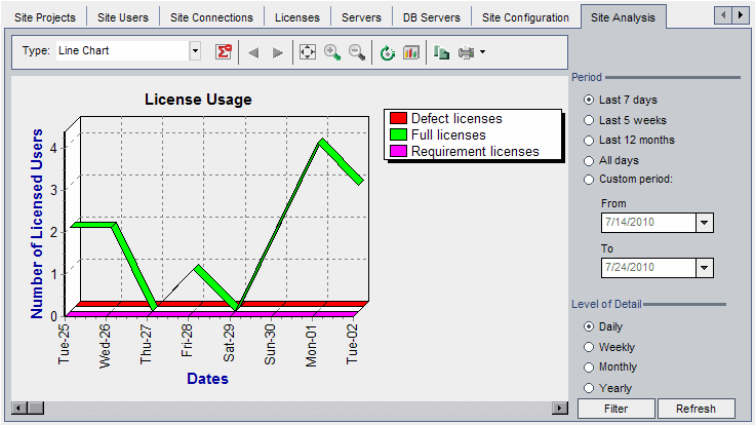
Monitoring Site Usage

You can monitor the number of licensed users that have connected to an ALM site over a selected period of time. You can analyze the usage level of each module and extension, and track the peak usage level in the current month. The data can be displayed in line graphs or data grids. In addition, you can filter records by projects, users, or license type, and save data to a file.

Note: You can also monitor the users currently connected to an ALM server. For more information, see "Managing User Connections and Licenses" on page 146.

To monitor site usage:

1. In Site Administration, click the **Site Analysis** tab.



2. In the **Type** box, select a display type:
   - **Line Chart.** Displays the data as a line graph.
   - **Data Grid.** Displays the data as a grid.
3. In the right pane under **Period**, select a set or custom period of time you want the line graph or

data grid to show.

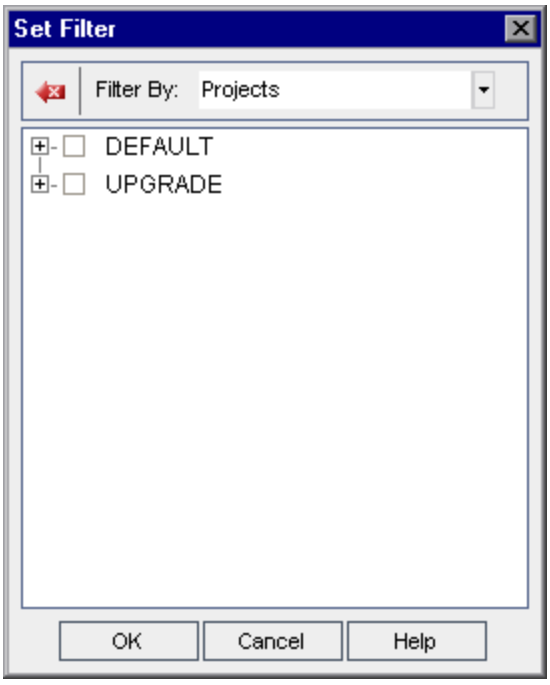
4. Under **Level of Detail**, select the time period between each measurement.
5. Click the **Filter** button to open the Set Filter dialog box and filter the graph contents. For more information, see "Filtering Site Usage" below.
6. To customize the appearance of a Line Chart graph, see "Customizing the Site Analysis Line Chart Graph" on next page.
7. If you chose Data Grid, you can save the contents of a data grid as a text file, Microsoft Excel spreadsheet, Microsoft Word document, or HTML document. To save, click the **Save As** button. For more information, see "Exporting Site Analysis Data to a File" on next page.
8. To refresh data in the graph, click the **Refresh** button.

Filtering Site Usage

You can analyze the number of users that have connected to your ALM site at specific points over time by filtering by projects, users, or license types.

To filter site usage:

1. In Site Administration, click the **Site Analysis** tab.
2. Click the **Filter** button in the bottom right corner. The Set Filter dialog box opens.



3. Under **Filter By**, select the category that you want to filter:
   - **Projects.** Displays all the ALM domains and projects.
   - **Users.** Displays all the ALM site users.
   - **License Types.** Displays all the available license types.

4. Click the items you want to include in the filter.
   - For **Projects**, double-click the domain folder to display the domain's projects, and select the projects you want to include. To filter all projects in the domain, select the domain folder.
   - For **Users**, select the users you want to include.
   - For **License Types**, select the licenses you want to include.
5. To clear the selected filter conditions, click the **Clear** button .
6. Click **OK** to apply the filter and close the Set Filter dialog box. The new line chart or data grid is displayed.

Exporting Site Analysis Data to a File

You can export site analysis data in a Data Grid as a text file, Microsoft Excel spreadsheet, Microsoft Word document, or HTML document.

To export Site Analysis data to a file:

1. In Site Administration, click the **Site Analysis** tab.
2. In the **Type** field, select the **Data Grid** display type.
3. Select the analysis period and define a filter.
4. Click **Save as**, and select one of the following formats:
   - **Text Format.** Saves the data as a Text file.
   - **Excel Sheet.** Saves the data as an Excel sheet.
   - **Word Document.** Saves the data as a Word document.
   - **HTML Document.** Saves the data as an HTML document.
5. In the **Save in** box, choose a location for the file.
6. In the **File name** box, type a name for the file.

   The **Save as type** box is automatically filled according to the format you selected.
7. Click **Save**.

Customizing the Site Analysis Line Chart Graph

You can determine how information appears in the Line Chart graph using the line chart toolbar. The toolbar includes the following buttons:

Tool Element	Description
	Show Total Values. Toggles between displaying and hiding a total value in the graph.

Tool Element	Description
	Scroll to the Left. Scrolls the graph to the left. (This button is enabled when the Zoom In and Zoom Out buttons are in use.)
	Scroll to the Right. Scrolls the graph to the right. (This button is enabled when the Zoom In and Zoom Out buttons are in use.)
	Show All. Returns the graph to its normal size. (This button is enabled when the Zoom In and Zoom Out buttons are in use.)
	Zoom In. Increases the magnification of the selected portion of the graph.
	Zoom Out. Decreases the magnification of the selected portion of the graph.
	Rotate Bottom Labels. Toggles between displaying the text on the x-axis vertically and horizontally.
	Set 2D/3D Graph. Toggles the graph from two to three dimensions.
	Copy Graph to Clipboard. Copies the graph to the Clipboard.
	Print Graph. You can choose to print the graph in portrait or landscape view.

Chapter 10: Scheduling Calculations for Project Planning and Tracking (PPT)

In Site Administration, you can schedule project planning and tracking (PPT) calculations for your HP Application Lifecycle Management (ALM) projects.

ALM Editions: Functionality related to PPT is available for ALM Edition only.

This chapter includes:

About Scheduling Calculations for PPT 205

Scheduling Calculations for a Site 205

Enabling or Disabling Automatic Calculations for a Project 205

Launching Calculations for a Project Manually 206

Project Planning and Tracking Tab 206

About Scheduling Calculations for PPT

PPT tracks application readiness and displays the status of your release in the form of a scorecard. The scorecard monitors and tracks how well each milestone is being met on a daily basis.

To view progress in the scorecard, you must run PPT calculations for your project. You can schedule calculations for your ALM site, and enable scheduled calculations for specific projects and include them in your daily progress calculations. In addition, you can manually trigger calculations for a selected project in order to refresh its results without waiting for an upcoming scheduled calculation.

For more details on PPT, refer to the *HP Application Lifecycle Management User Guide*.

Note: For information on configuring PPT log files, see "Configuring Server Information" on page 157.

Scheduling Calculations for a Site

This section describes how to schedule PPT calculations for an ALM site.

To schedule calculations for a site:

1. In Site Administration, click the **Project Planning and Tracking** tab.
2. Schedule calculations in the Project Planning and Tracking tab. For user interface details, see "Project Planning and Tracking Tab" on next page.
3. Enable projects for automatic calculations. For more details, see "Enabling or Disabling Automatic Calculations for a Project" below.

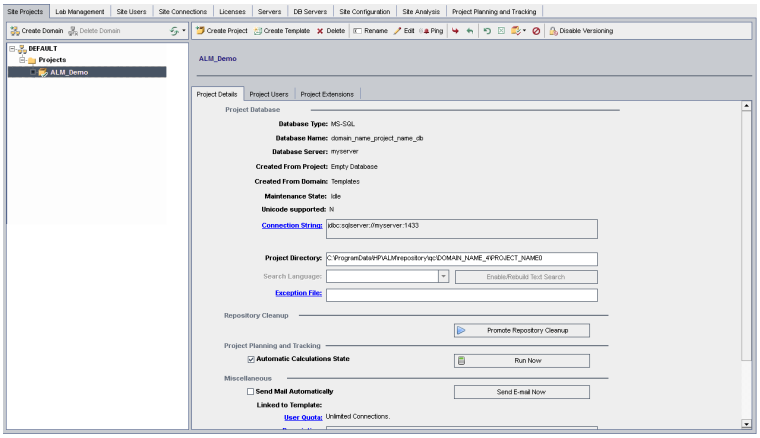
Enabling or Disabling Automatic Calculations for a Project

This section describes how to enable PPT calculations for a project in order to include it in the automatic daily calculations of your site. When your business needs change, you can disable calculations for a project.

Note: PPT is enabled by default when creating a new project.

To enable or disable automatic calculations for a project:

1. In Site Administration, click the **Site Projects** tab.



2. In the Projects list, select a project.
3. In the Project Details tab, under **Project Planning and Tracking**, click **Automatic Calculations State**. Click **OK** to confirm.

Launching Calculations for a Project Manually

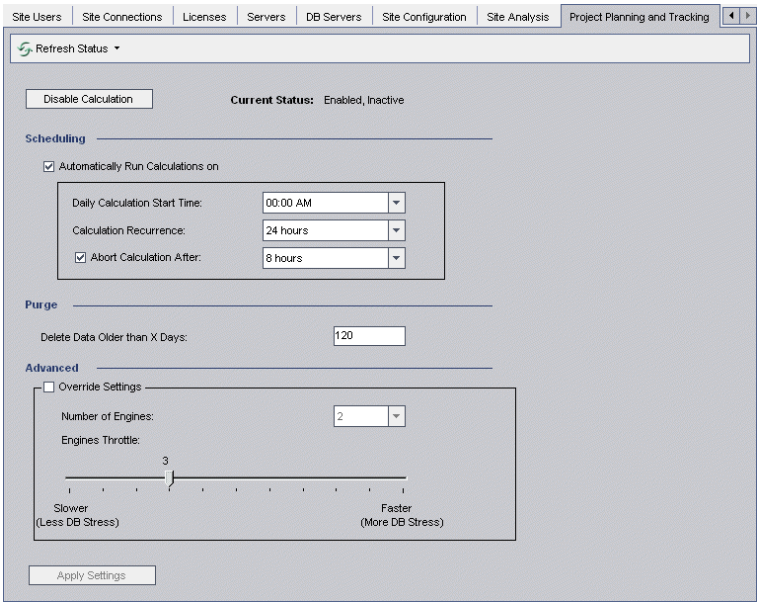
This section describes how to manually trigger PPT calculations for a project in order to refresh its results without waiting for the next scheduled calculation.

To launch calculations for a project manually:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, select a project.
3. In the Project Details tab, under **Project Planning and Tracking**, click the **Run Now** button.

Project Planning and Tracking Tab

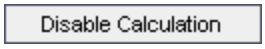
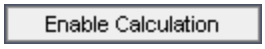
This tab enables you to manage PPT calculations for your entire site.



To access	In Site Administration, click the **Project Planning and Tracking** tab.
Important information	• The database server time displayed on the bottom right-side of the Project Planning and Tracking tab is used when scheduling calculations. • By default, ALM performs calculations on a project that has been in use in the past 7 days. If a project has not been in use in the past 7 days, calculations are not performed. To change the number of days, edit the **PPT_RECENTLY_USED_PROJECTS_THRESHOLD_MINUTES** parameter in the Site Configuration tab in Site Administration. For more information, see "PPT_RECENTLY_USED_PROJECTS_THRESHOLD_MINUTES" on page 188. • By default, if 10% or more of the KPI calculations within the release fail, ALM aborts project planning and tracking calculations on a release and skips to the next release in a project. To change the percentage value, edit the **PPT_KPI_FAILURES_PERCENTAGE_PER_RELEASE_FUSE** parameter in the Site Configuration tab in Site Administration. For more information, see "PPT_KPI_FAILURES_PERCENTAGE_PER_RELEASE_FUSE" on page 188.

Miscellaneous Elements

User interface elements are described below:

UI Elements	Description
	This button includes the following options: **Refresh Status.** Refreshes the Project Planning and Tracking tab so that it displays the most up-to-date information. **Automatic Refresh.** Instructs ALM to automatically refresh the Project Planning and Tracking tab. By default, the tab is automatically refreshed every 60 seconds. **Set Refresh Rate.** Opens the Set Refresh Rate dialog box, enabling you to change the automatic refresh rate in seconds.
	Terminates all PPT activities.
	Enables PPT for your site.
Current Status	Includes the following options: **Enabled/Disabled.** Indicates whether PPT is enabled for your site. **Active/Inactive.** Indicates whether the scheduled calculation is currently running.
Apply Settings	Applies scheduling changes.

Scheduling Area

This area enables you to schedule PPT calculations for your entire site.

User interface elements are described below:

UI Elements	Description
Automatically Run Calculations On	Indicates whether or not scheduled calculations are performed on your site.
Daily Calculation Start Time	Start time for scheduling PPT calculations.
Calculation Recurrence	Runs calculations periodically at a specified time.
Abort Calculation After	Ends scheduled calculations at a specified time.

Purge Area

This area enables you to purge calculations after a specific time period.

User interface elements are described below:

UI Elements	Description
Delete data older than X days	Deletes data older than the defined value. The default value is set to 120 days. **Note:** To retain the scorecards, the last five days of the milestone's results are not purged.

Advanced Area

This area enables you to increase the number of parallel calculations that are being performed on an entire site. You can also change the speed of scheduled calculations.

User interface elements are described below:

UI Elements	Description
Override Settings	Enables the advanced settings.
Number of Engines	Sets the number of parallel calculations that are being performed concurrently on an entire site.
Engines Throttle	Changes the speed that it takes for ALM to calculate the KPI data. **1** indicates a slower processing speed and minimum database stress. **10** indicates a faster processing speed and maximum database stress.

Chapter 11: ALM Assessment

This chapter describes the ALM Assessment process for collecting and analyzing environment details in order to assess the current ALM deployment and to recommend usage improvements.

This chapter includes:

About ALM Assessment 211

Collecting and Assessing ALM Data 216

About ALM Assessment

As an ALM site administrator, you can collect environment details to assess the current ALM deployment and recommend usage improvements. The collected data can also be helpful when contacting support.

The ALM Assessment process collects environment details, which are stored in a file in the following location: **C:\repository\productData\assessment\results\AssessmentInfo_<current date>.zip**.

Once these details are collected, you can then access the ALM Assessment website to assess the deployment and recommend usage improvements.

Note: Some of the SQL Server properties require the site administrator for that database to have advanced options permissions. To ensure these permissions, run the following:

```
sp_configure 'show advanced options', 1
GO
RECONFIGURE
GO
```

The following table describes the collected details.

Property Name	Description
DB Common Data	
Name	DB Server name
Version	DB Server version
SQL Server Data	
productLevel	The level of the version of the SQL Server instance
edition	The installed product edition of the SQL Server instance
engineEdition	Database Engine edition of the SQL Server instance installed on the server
editionId	The edition ID of the SQL Server instance
isFullTextInstalled	Indicates whether the full-text and semantic indexing components are installed on the current SQL Server instance
collation	Name of the default collation for the server

Property Name	Description
osMachineName	Windows computer name on which the server instance is running
Oracle DB Data	
sga_max_size	Hard limit up to which sga_target can dynamically adjust sizes
processes	The maximum number of operating system user processes that can simultaneously connect to Oracle
sessions	Maximum number of sessions that can be created in the system
cpu_count	The number of CPUs available for use by the Oracle database
db_cache_size	The size of the DEFAULT buffer pool for buffers with the primary block size
log_buffer	The amount of memory (in bytes) that Oracle uses when buffering redo entries to a redo log file
shared_pool_size	The size of the shared pool
large_pool_size	The size of the large pool allocation heap
java_pool_size	The size of the Java pool, from which the Java memory manager allocates most Java state during runtime execution
pga_aggregate_target	The target aggregate PGA memory available to all server processes attached to the instance
optimizer_mode	Establishes the default behavior for choosing an optimization approach for the instance
optimizer_features_enable	Acts as an umbrella parameter for enabling a series of optimizer features based on an Oracle release number
compatible	Allows using a new release of Oracle, while at the same time guaranteeing backward compatibility with an earlier release
cursor_sharing	Determines what kind of SQL statements can share the same cursors
open_cursors	Specifies the maximum number of open cursors (handles to private SQL areas) a session can have at once

Property Name	Description
session_cached_cursors	Specifies the number of session cursors to cache
memory_max_target	Specifies the maximum value to which a DBA can set the MEMORY_TARGET initialization parameter
sga_target	Specifies the total size of all SGA components
NLS Parameters (NLS_ CHARACTERSET, NLS_LANGUAGE , NLS_TERRITORY, NLS_LENGTH_ SEMANTICS)	Character set related information
Site Usage Data	
usersCount	Number of site users
activeUsersCount	Number of active site users
operativeUsersCount	Number of active site users who opened a session in the last 30 days
projectsCount	Number of projects
activeProjectsCount	Number of active projects
operativeProjectsCount	Number of active projects that had a session in the last 30 days
Application Server Data	
maxDbConnectionsAllowed	The maximum concurrent db connections allowed
connectorName	Jetty connector name
maxConnectionsRequest	The maximum number of connections requests
maxConnectionsOpen	The maximum number of open connections
maxConnectionsDuration	The maximum duration of connections
Architecture Data	
activeNodes	Number of active nodes- application servers
totalNodes	Total number of nodes
serverName	Application server name
isActive	Application server status
General Site Data	

Property Name	Description
almEdition	ALM edition
almEditionDisplayName	ALM edition display name
informationCollectionTime	Information collection time in milliseconds
extensionName	Extension name
extensionDisplayName	Extension display name
version	Extension version
Cache Data	
cacheManagerName	Name of the EHCache manager
cacheName	Name of the cache
objectCount	Number of items found in cache
maxElementsInMemory	Sets the maximum number of objects created in memory
maxElementsOnDisk	Sets the maximum number of objects maintained in the DiskStore
memoryStoreEvictionPolicy	Policy enforced upon reaching the maxElementsInMemory limit
eternal	Determines whether an element is eternal, meaning that timeouts are ignored and the element is never expired
timeToIdleSeconds	Sets the time for an element to idle before it expires
timeToLiveSeconds	Sets the time for an element to live before it expires
overflowToDisk	Determines whether elements can overflow to disk when the memory store reaches the maxElementsInMemory limit
diskPersistent	Determines whether the disk store persists between restarts of the Virtual Machine
diskSpoolBufferSizeMB	The size to allocate the DiskStore for a spool buffer
diskExpiryThreadIntervalSeconds	The number of seconds between runs of the disk expiry thread

Property Name	Description
statisticsAccuracy	The accuracy of the cache statistics at the time the statistic was computed
cacheHits	The number of times a requested item was found in the cache
onDiskHits	The number of times a requested item was found in the Disk Store
inMemoryHits	The number of times a requested item was found in the Memory Store
misses	The number of times a requested element was not found in the cache
averageGetTime	The average get time
evictionCount	Gets the number of cache evictions since the cache was created or statistics were cleared
Jvm Data	
jvmVersion	Version of JVM being used
usedHeapSize	The total amount of memory in the Java virtual machine
maxHeapSize	The maximum amount of memory that the Java virtual machine attempts to use
freeHeapSize	The amount of free memory in the Java virtual machine
nonHeapUsedSize	The amount of used memory in bytes
nonHeapInitSize	The amount of memory in bytes that the Java virtual machine initially requests from the operating system for memory management
nonHeapCommittedSize	The amount of memory in bytes that is committed for the Java virtual machine to use
nonHeapMaxSize	The maximum amount of memory in bytes that can be used for memory management
name	The name representing this memory manager
collectionElapseTime	The approximate accumulated collection elapsed time in milliseconds
innerCollectionCount	The total number of collections that have occurred

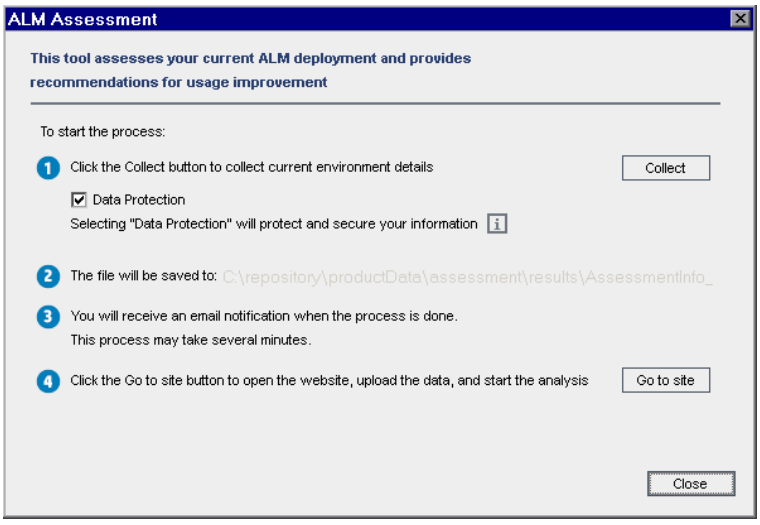
Property Name	Description
memoryPoolNames	The names of memory pools that this memory manager manages
architecture	Architecture of the JVM being used
is64Bit	Whether the JVM being used has 64 bits architecture
numberOfProcessorsAvailable	Number of processors available for this JVM
Machine Specification Data	
totalMemoryMB	Total memory installed on the machine
QCSense Data	
startGroupingTime	Segment starting time
endGroupingTime	Segment ending time
nodeName	Name of the node for which the data was collected
measurementName	Name of the measurement collected during the segment
weightedMeanAverageValue	Weighted mean of the average value recorded during the segment
minimumValue	Minimum value recorded during the segment
weightedMeanMinimumValue	Weighted mean of the minimum value recorded during the segment
maximumValue	Maximum value recorded during the segment
weightedMeanMaximumValue	Weighted mean of the maximum value recorded during the segment
measurementsInspectedNumber	Number of records (measurement values) recorded during the segment

Collecting and Assessing ALM Data

You can assess the current ALM deployment and recommend usage improvements based on the environment details collected.

To collect environment details:

1. In Site Administration, select **Tools > ALM Assessment**. The ALM Assessment dialog box opens.



2. Select **Data Protection** to protect sensitive data. Click for more information on the data protection process.
3. Click **Collect** to begin the data collection process. An email is sent when the process is complete.

   Note: The email may indicate that the data collection process did not complete successfully. In this case, ensure that all nodes are active and rerun the data collection process.

4. Once the data collection process is complete, log in to the ALM Assessment website. The link to the website is found in the email sent when the data collection process is complete, or click **Go to site** in the ALM Assessment dialog box.
5. Follow the instructions on the website to upload the collected data and to complete the assessment process.

Chapter 12: QC Sense

This chapter describes QC Sense, the internal monitoring tool for collecting and analyzing HP Application Lifecycle Management (ALM) usage and performance data.

This chapter includes:

About QC Sense 219

QC Sense Configuration 219

Generating and Viewing QC Sense Reports 226

QC Sense Schema 227

About QC Sense

As an ALM site administrator, you can utilize usage and performance data collected by QC Sense to analyze ALM performance behaviors from the user perspective. For example, you can look at the time it takes from the user's click of a button until the desired response is received.

QC Sense collects data on user actions performed in the ALM user interface, and monitors all client and server activity resulting from those actions. You can examine a single user activity and its resulting server and database activities, as well as compare average system response analyzed by user action, server transaction, and many other parameters.

You can configure QC Sense to collect data based on a wide variety of actions and measures. This enables you to examine performance as it relates to different aspects of ALM, such as projects, users, types of actions, and workflow impact. You can examine and compare performance between components of the system, including application servers, database servers, the network, and the file system.

In addition to data on single user activity, you can configure QC Sense to collect information on the ALM servers in your site. QC Sense can collect data on server and network activity such as server threads, memory usage, active sessions, database access time, and file system access time.

QC Sense includes client and server monitors. Each monitor collects usage and performance data on a specific area within ALM. As an ALM site administrator, you can configure each monitor to customize the scope of data collected. All data collected by QC Sense is stored centrally in a site database so you can easily connect between client activity and the resulting server activity. For more information, see "QC Sense Configuration" below.

QC Sense reports enable you to examine and compare performance to help identify the source of a problem. You can compare data between users who are experiencing different levels of performance, or you can clarify actions or behaviors that may be causing a sudden slowing in system responsiveness. For example, you can identify especially long operations, or examine which long operations occurred just before the general slowness on your site. For more information, see "Generating and Viewing QC Sense Reports" on page 226.

The data collected by the QC Sense monitors is stored in the database according to the QC Sense schema. For more information, see "QC Sense Schema" on page 227.

QC Sense Configuration

To configure QC Sense, access the QC Sense Server Configuration window from Site Administration.

The configuration options enable you to:

- set filters for each client and server monitor to define the scope of data that QC Sense collects
- enable or disable a QC Sense monitor
- define the location for storing QC Sense data
- define the maximum number of records stored per monitor

- set the frequency for transferring updated data to the database
- define the frequency at which the QC Sense tables are purged

This section includes:

QC Sense Monitors 220
Configuring QC Sense 221
QC Sense Server Configuration Window 221
Connection String Builder Dialog Box 224

QC Sense Monitors

The following table lists the QC Sense monitors and describes the data that each monitor collects by default. For more information on the QC Sense database tables that store the collected data, see "QC Sense Schema" on page 227.

Monitor Name	Database Table Description	Default Configuration
Client Operation	Contains raw data for user operations, such as submitting a defect, updating a requirement, or clicking the **Login** button. For details, see "PERF_CLIENT_OPERATIONS" on page 227.	**Partial**. Collects data on the following: • **Login** operations that exceed two minutes. • **Create Entity** operations that exceed two minutes. • **Paste** operations that exceed two minutes. • All operations that exceed five minutes.
Client Method Call	Contains raw data for ALM client methods monitored by QC Sense. For details, see "PERF_CLIENT_METHODS_CALLS" on page 229.	**Partial.** Collects data on the following: • Calls to generate Analysis Items that exceed two minutes. • Calls to workflow events that exceed two minutes.
Client Request	Raw data for requests sent by a client operation to the ALM server. For details, see "PERF_CLIENT_REQUESTS" on page 230.	**Partial.** Collects data on all requests that were sent to the server in the context of a filtered client operation or a filtered client method.

Monitor Name	Database Table Description	Default Configuration
Server General	Aggregated data based on several server measures. For details, see "PERF_SERVER_GENERAL_MEASURES" on page 234.	**On**
Server Thread Type	Aggregated data on threads running on the server. For details, see "PERF_SERVER_THREAD_TYPES" on page 235.	**On**
Server Thread	Raw data for each thread run on the server. For details, see "PERF_SERVER_THREADS" on page 232.	**Off**
Server SQL	Raw data for each SQL statement run by the ALM server. For details, see "PERF_SERVER_SQLS" on page 233.	**Off**

Configuring QC Sense

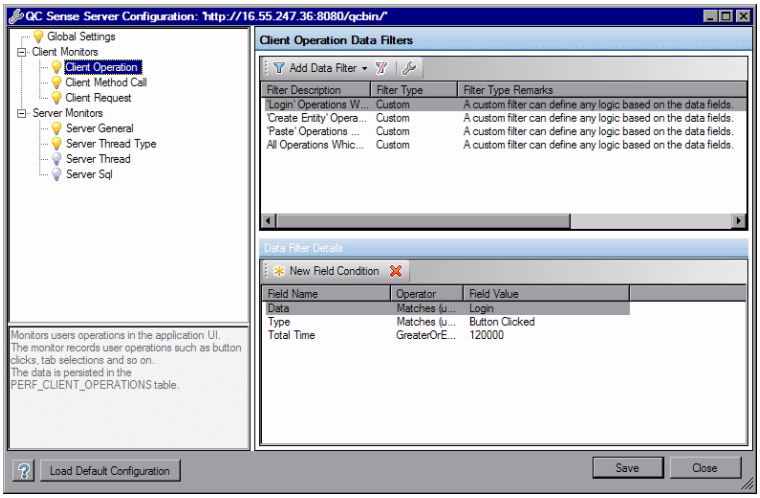
You can configure the global settings for QC Sense, such as specifying an alternate location for the QC Sense schema. You can also configure settings for each monitor to define the scope of data collected, and the maximum number of records stored for the monitor.

To configure QC Sense:

1. In Site Administration, select **Tools > QC Sense > Configuration**. The Login to <server> dialog box opens.
2. Enter login credentials for a site administrator user and click **OK**. The QC Sense Server Configuration window opens. For user interface details, see "QC Sense Server Configuration Window" below.

QC Sense Server Configuration Window

This window enables you to configure QC Sense settings for an ALM server.

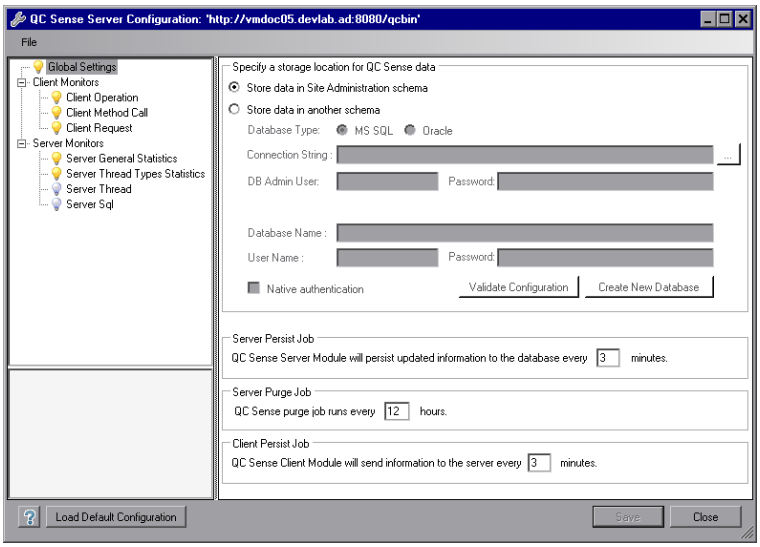


QC Sense configuration includes global settings, as well as settings you can configure for each monitor individually.

To access	In Site Administration, select **Tools > QC Sense > Configuration**. In the Login to <server> box, enter your site administrator password.
Important information	For details on the default configuration settings, see "QC Sense Monitors" on page 220.
See also	• "About QC Sense" on page 219 • "Generating and Viewing QC Sense Reports" on page 226

Global Settings

Enables you to define general settings for QC Sense.



To access	In the monitor list located on the left side of the window, select **Global Settings**.

User interface elements are described below:

UI Element	Description
Specify a storage location for QC Sense data	Includes the following options: • **Store data in Site Administration schema.** • **Store data in another schema** (recommended)**.** Specify an alternate location and connection information for storing QC Sense schema tables. To define the Connection String, click the browse button ( ). For details, see the " Connection String Builder Dialog Box" on next page. You can also enter all the necessary data manually. • **Native authentication.** For an SQL server, use Windows Authentication and not SQL Server authentication. For details on the QC Sense schema, see "QC Sense Schema" on page 227.
Server Persist Job	Defines the time interval, in minutes, at which updated information from QC Sense server monitors is written to the database.
Server Purge Job	Defines the time interval, in hours, at which QC Sense tables are cleaned.
Client Persist Job	Defines the time interval, in minutes, at which updated information from QC Sense client monitors is sent to the server.
Load Default Configuration	Restores default QC Sense configuration settings. Located in the bottom left of the configuration window.

Monitor Settings

Enables you to define settings for QC Sense monitors. For details on the monitors, see "QC Sense Monitors" on page 220.

To access	In the monitor list located on the left side of the window, select a monitor to configure.

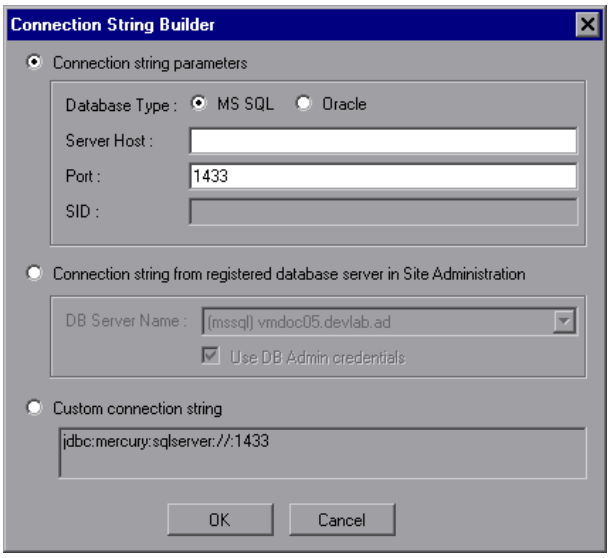
User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
<Monitor list>	Located on the left side of the window, displays the QC Sense client and server monitors. Select a monitor to configure its options and filters. indicates an active monitor. indicates an inactive monitor. Data is not collected for this monitor. To activate or deactivate a monitor, right-click the monitor and select **Turn Monitor OFF/ON**.

UI Element	Description
	Enables you to set filter conditions to define the scope of data collected by the filter. Select a filter from the menu to add and configure. There are different filters available for each QC Sense monitor.
	Delete Data Filter. Deletes the selected data filter. If there are no filters defined, QC Sense collects all data for the selected monitor. **Note:** An SQL server must have at least one filter.
	Monitor Settings. Enables you to define settings for the selected monitor. Includes the following settings: • **Maximum number of records in monitor database table:** Defines the maximum number of records for the monitor that can remain in the database after the database is cleaned by the server purge job. • **Time frame length**. Defines the time frame for which some measures are calculated. For example, the average server CPU time used for processing a thread, measured over a 15 minute period. Available for the following monitors: Server General, Server Thread Type • **Excluded Fields.** Monitor data is not saved for the selected fields. Available for the following monitors: Server SQL, Server Thread
Data Filters pane	Lists the filters for the selected monitor.
Data Filter Details	Displays details and enables you to set conditions for the selected data filter. For more information on the available fields for each monitor, see "QC Sense Schema" on page 227.
Monitor Description	Describes the selected monitor. Indicates the QC Sense schema table that stores data for the monitor.

Connection String Builder Dialog Box

This dialog box enables you to build custom connection strings. It also enables you to use connection strings that are already defined in Site Administration.



To access	From the **Global Settings** window, under **Specify a storage location for QC Sense data**, click **Store data in another schema**, and then click ...
See also	"QC Sense Server Configuration Window" on page 221

User interface elements are described below:

UI Element	Description

Connection string parameters	This option enables you to build the connection string by selecting a database type and defining parameters.
	• **Database Type.** The database type can be **MS SQL** or **Oracle**.
	• **Server Host.** The server name.
	• **Port.** The port number of the database server. The default ports are:
	  ▪ Oracle: 1521
	  ▪ MS SQL: 1433
	• **SID.** The service ID for an Oracle database server. The SID field can be edited only for Oracle database type.

UI Element	Description
Connection string from registered database server in Site Administration	This option enables you to select connection strings from registered database servers in Site Administration. • **DB Server Name.** The database server name. • **Use DB Admin credentials.** Use the database administrator credentials of the selected database server.
Custom connection string	This option enables you to define complex and non-standard connection strings directly here.

Generating and Viewing QC Sense Reports

Note: The QC Sense reports are for debugging purposes only, and are disabled by default. If you generate QC Sense reports, it is recommended to disable the reports immediately after usage.

You can generate reports based on the data collected by QC Sense. For example, to examine the user experience, you can generate reports on the following:

- all operations performed by a specific user
- transactions of a specific type for all users, such as creating a requirement or clicking the **Login** button
- transactions longer than a set amount of time
- compare users who are experiencing different levels of performance

After you generate a report, you can print the report.

The following types of reports are available:

- **Client reports.** Based on data collected by the QC Sense client monitors. Reports provide information from the user experience perspective, representing user operations in the ALM user interface.
- **Server reports.** Based on data collected by the QC Sense server monitors.
  - Represent server activities invoked by user operations.
  - Server reports on general server activity.
- **Database table reports.** Enable you to access information in the QC Sense schema tables using standard SQL syntax.

To generate and view reports:

1. In Site Administration, select **Tools > QC Sense > Reports**. The Login to <server> dialog box opens.

2. Enter login credentials for a site administrator user and click **OK**. The QC Sense Report page opens.
3. Click a report link. In the login window that opens, enter login credentials for a site administrator user.

   Note: You can grant permissions to additional users who are not site administrators by configuring the QC_SENSE_REPORTS_USERS parameter in the Site Configuration tab in Site Administration. For information, see "QC_SENSE_REPORTS_USERS" on page 189.
4. In the report viewer, enter parameters to define the scope of the report you want to generate.

QC Sense Schema

QC Sense has a single database schema per ALM site. The schema is created during the ALM installation, and stored by default in the Site Administration schema. However, the QC Sense schema is independent of the ALM Site Administration schema, and the tables can be stored in any database schema. For enterprise sites, we recommend storing the schema on another database server. For details on defining an alternate location for the QC Sense schema, see "QC Sense Configuration" on page 219.

The QC Sense schema consists of the following tables:

Table Name	Data Source	Data Type
"PERF_CLIENT_OPERATIONS" below	Client	Raw
"PERF_CLIENT_METHODS_CALLS" on page 229	Client	Raw
"PERF_CLIENT_REQUESTS" on page 230	Client	Raw
"PERF_SERVER_THREADS" on page 232	Server	Raw
"PERF_SERVER_SQLS" on page 233	Server	Raw
"PERF_SERVER_GENERAL_MEASURES" on page 234	Server	Aggregated
"PERF_SERVER_THREAD_TYPES" on page 235	Server	Aggregated

PERF_CLIENT_OPERATIONS

This table stores the data collected by the **Client Operation** monitor.

The table contains one record for each client operation. A **client operation** is an action performed by the user in the ALM user interface. For example:

- clicking the **Authenticate** button in the Login window
- selecting the **Attachments** tab in the Defects module
- expanding a folder in the test plan tree

A client operation type is described by **type**, **data**, and **context**. The column prefix for the table is `PCO`. For example, `PCO_OPERATION_ID`.

COLUMN NAME	DESCRIPTION
OPERATION_ID	A unique GUID assigned to the operation.
CLIENT_MACHINE_NAME	The client host name in which the operation was performed.
LOGIN_SESSION_ID	The login session ID.
PROJECT_SESSION_ID	The project session ID.
PROJECT	The domain and project name, in the format `<domain name>/<project name>`.
USER_NAME	The user name.
OPERATION_TYPE	The type of the operation. For example: • Button Clicked • Tab Selected • Tree Node Expanded
OPERATION_DATA	The operation's data. For example: • The clicked button label, such as **Login** • The selected tab label, such as **Attachments**
OPERATION_CONTEXT	The path to the window in which the operation was performed. For example: • **Module**: Business Components .Net; **View**: EntityTypeViewControl; **View**: ComponentStepsViewControl; **View**: DesignStepsViewControl • **Form**: Component Step Details
CLIENT_START_TIME	The start time of the operation as a date time data type.
CLIENT_END_TIME	The end time of the operation as a date time data type.
CLIENT_START_TIME_MS	The start time of the operation as the number of milliseconds since **01/01/1970**.
CLIENT_TOTAL_TIME	The total milliseconds that passed from the start of the operation until the end of the operation.

PERF_CLIENT_METHODS_CALLS

This table stores the data collected by the **Client Method Call** monitor.

The table contains one record for each method call monitored by QC Sense. The methods that are monitored are predefined by QC Sense. Each method call is linked to the following related records:

- **Owner Operation.** The active operation when this method call was ran.
- **Owner Method Call.** The active monitored method when this method call ran. It can be null.
- **Called Methods.** Another monitored method called from this method call, directly or indirectly.
- **Requests.** Requests sent to the server from this method call, directly or indirectly.

The column prefix for the table is `PCMC`. For example, `PCMC_METHOD_CALL_ID`.

COLUMN NAME	DESCRIPTION
METHOD_CALL_ID	A unique GUID assigned to the method call.
LOGIN_SESSION_ID	The login session ID.
PROJECT_SESSION_ID	The project session ID.
PROJECT	The domain and project name, in the format `<domain name>/<project name>`.
USER_NAME	The user name.
SEQUENCE	The sequence of the method call in the context of its owner operation.
OWNER_OPERATION_ID	The active client operation ID when the method call started.
OWNER_OPERATION_TYPE	The type of the owner operation.
OWNER_OPERATION_DATA	The data of the owner operation.
OWNER_OPERATION_CONTEXT	The context of the owner operation.
OWNER_METHOD_CALL_ID	The active method call ID when the method call started (can be null).
METHOD_NAME	The method name, for example, **Login**.
CLASS_NAME	The class name, for example, **ConnectionManagementService**.
MODULE_NAME	The module/assembly name, for example, **QCClient.Library.dll**.
ADDITIONAL_DATA	Additional data added by the method call.

COLUMN NAME	DESCRIPTION
CLIENT_START_TIME	The start time of the method call as a date time data type.
CLIENT_END_TIME	The end time of the method call as a date time data type.
CLIENT_START_TIME_MS	The start time of the operation as the number of milliseconds since **01/01/1970**.
CLIENT_TOTAL_TIME	The total milliseconds that passed from the start of the operation until the end of the operation.

PERF_CLIENT_REQUESTS

This table stores the data collected by the **Client Request** monitor.

The table contains one record for each request sent from the client to the server. Each request is linked to the following related records:

- **Owner Operation.** The active operation when the request was sent to the server.
- **Owner Method Call.** The active monitored method when the request was sent to the server. It can be null.

The request record contains the following:

- client performance data, such as the time the client sent the request to the server
- server performance data, such as the time the request arrived at the server

The column prefix for the table is `PCR`. For example, `PCR_REQUEST_ID`.

COLUMN NAME	DESCRIPTION
REQUEST_ID	The request unique GUID. **Note:** This is also the GUID of the thread that handled the request in the server.
SEQUENCE	The sequence of the request in the context of its owner operation.
OWNER_OPERATION_ID	The active client operation ID when the request was sent to the server.
OWNER_METHOD_CALL_ID	The active monitored method when the request was sent to the server.
LOGIN_SESSION_ID	The login session ID.
PROJECT_SESSION_ID	The project session ID.

COLUMN NAME	DESCRIPTION
PROJECT	The domain and project name, in the format `<domain name>/<project name>`.
USER_NAME	The user name.
REQUEST_TYPE	The request type, for example, **PostBug**.
CLIENT_START_TIME	The time the request was sent to the server as a date time data type.
CLIENT_END_TIME	The time the response returned from the server as a date time data type.
CLIENT_START_TIME_MS	The time the request was sent to the server as the number of milliseconds since **01/01/1970**.
CLIENT_TOTAL_TIME	The total milliseconds that passed from the time the request was sent to the server until the response was received.
SERVER_MACHINE_NAME	The ALM server in which the request was processed.
SERVER_START_TIME	The time the server started processing the request as a date time data type.
SERVER_START_TIME_MS	The time the server started processing the request as the number of milliseconds since **01/01/1970**.
SERVER_TOTAL_TIME	The total time, in milliseconds, it took for the server to process the request.
SERVER_CPU_TIME	The total CPU time, in milliseconds, allocated for processing the request.
DB_TIME_AVG	The average time the database processed an SQL statement for this thread.
DB_TIME_MAX	The maximum time the database processed an SQL statement for this thread.
DB_TIME_MIN	The minimum time the database processed an SQL statement for this thread.
DB_TIME_COUNT	The number of SQL statements processed by the database for this thread.
FS_TIME_AVG	The average access time to the file system for this request.
FS_TIME_MIN	The minimum access time to the file system for this request.
FS_TIME_MAX	The maximum access time to the file system for this request.
FS_TIME_COUNT	The number of file system accesses (read/write or delete file) for this request.

PERF_SERVER_THREADS

This table stores the data collected by the **Server Thread** monitor.

The table contains one record for each thread run on the server. There are four thread categories:

- **REQUEST.** Thread processing web-gate requests.
- **JOB.** Threads running ALM jobs.
- **ASYNC_TASK.** Threads running ALM asynchronous tasks.
- **NONE.** All other threads, such as threads running the repository migration process.

The column prefix for the table is `PCT`. For example, `PCT_THREAD_ID`.

COLUMN NAME	DESCRIPTION
THREAD_ID	The thread unique GUID.
SERVER_MACHINE_NAME	The ALM server in which the thread was processed.
THREADY_CATEGORY	The thread category. Available categories: REQUEST, JOB, ASYNC_TASK, NONE.
THREAD_TYPE	The thread type. For example, request type – **PostBug**, Job Name – **CKeepAliveJob**, etc.
SERVER_START_TIME	The time the thread started to run as a date time data type.
SERVER_START_TIME_MS	The time the thread started to run as the number of milliseconds since **01/01/1970**.
LOGIN_SESSION_ID	The login session ID.
PROJECT_SESSION_ID	The project session ID.
PROJECT	The domain and project name, in the format `<domain name>/<project name>`.
USER_NAME	The user name.
SERVER_TOTAL_TIME	The total time, in milliseconds, it took for the server to process the thread.
SERVER_CPU_TIME	The total CPU time, in milliseconds, allocated for processing the thread.
DB_TIME_AVG	The average time the database processed an SQL statement for this thread.
DB_TIME_MAX	The maximum time the database processed an SQL statement for this thread.

COLUMN NAME	DESCRIPTION
DB_TIME_MIN	The minimum time the database processed an SQL statement for this thread.
DB_TIME_COUNT	The number of SQL statements processed by the database for this thread.
FS_TIME_AVG	The average access time to the file system for this thread.
FS_TIME_MIN	The minimum access time to the file system for this thread.
FS_TIME_MAX	The maximum access time to the file system for this thread.
FS_TIME_COUNT	The number of file system accesses (read/write or delete file) for this thread.

PERF_SERVER_SQLS

This table stores the data collected by the **Server SQL** monitor.

You can examine the data in this table as it relates to the user actions that are performed (stored in the PERF_CLIENT_OPERATIONS table) and the resulting requests that are generated (stored in the PERF_CLIENT_REQUESTS table).

The table contains one record for each SQL statement run on the server. The column prefix for the table is `PSS`. For example, `PSS_SQL_ID`.

COLUMN NAME	DESCRIPTION
SQL_ID	The SQL unique GUID.
SERVER_MACHINE_NAME	The ALM server that ran this SQL statement.
THREAD_ID	The ID of the thread in the context of which the SQL statement was run.
THREAD_CATEGORY	The thread category.
THREAD_TYPE	The thread type.
PROJECT	The domain and project name, in the format `<domain name>/<project name>`.
USER_NAME	The user name.
SQL_TYPE	The type of SQL statement, for example: `executeQuery' and `executeUpdate'.
RECORD_COUNT	Number of records added, deleted or etched by this SQL statement.
START_TIME	The SQL statement start time as a date time data type.

COLUMN NAME	DESCRIPTION
START_TIME_MS	The SQL statement start time as the number of milliseconds since **01/01/1970**.
TOTAL_TIME	The total time, in milliseconds, that it took for the server to execute the SQL statement.
SQL_STRING	The actual SQL string.

PERF_SERVER_GENERAL_MEASURES

This table stores the data collected by the **Server General** monitor.

The table stores aggregated data about ALM server behavior. Each record describes a single measure on a single ALM node in a specific time frame. The column prefix for the table is `PSGM`. For example, `PSGM_SERVER_MACHINE_NAME`.

COLUMN NAME	DESCRIPTION
SERVER_MACHINE_NAME	The ALM server from which the data was collected.
START_TIME	The start time of the record's time frame.
END_TIME	The end time of the record's time frame.
MEASURE_NAME	The measure name. Available values: • MEMORY_USAGE • ACTIVE_THREADS • ACTIVE_PROJECT_SESSION • THREAD_TOTAL_TIME • THREAD_CPU_TIME • FREC_REQUEST_CALL_TOTAL_TIME • DB_TIME • FS_TIME
AVG	The average value measured during the time frame.
MIN	The minimum value measured during the time frame.
MAX	The maximum value measured during the time frame.
COUNT	The number of times the measure was calculated during the time frame.

PERF_SERVER_THREAD_TYPES

This table stores the data collected by the **Server Thread Type** monitor.

The table stores aggregated data about server threads. Each record describes a single thread type behavior on a single ALM node in a specific time frame in a specific ALM project. The column prefix for the table is `PSTT`, for example, `PSTT_SERVER_MACHINE_NAME`.

COLUMN NAME	DESCRIPTION
SERVER_MACHINE_NAME	The ALM server from which the data was collected.
START_TIME	The start time of the record's time frame.
END_TIME	The end time of the record's time frame.
THREAD_CATEGORY	The thread category. Available categories: REQUEST, JOB, ASYNC_TASK, NONE.
THREAD_TYPE	The thread type. For example, request type – **PostBug**, Job Name – **CKeepAliveJob**, etc.
PROJECT	The domain and project name, in the format `<domain name>\<project name>`.
SERVER_TOTAL_TIME_AVG	The average time, in milliseconds, it took for the server to process a thread.
SERVER_TOTAL_TIME_MIN	The minimum time, in milliseconds, it took for the server to process a thread.
SERVER_TOTAL_TIME_MAX	The maximum time, in milliseconds, it took for the server to process a thread.
SERVER_TOTAL_TIME_COUNT	The number of threads that run in the server.
SERVER_CPU_TIME_AVG	The average CPU time, in milliseconds, allocated for processing a thread.
SERVER_CPU_TIME_MIN	The minimum CPU time, in milliseconds, allocated for processing a thread.
SERVER_CPU_TIME_MAX	The maximum CPU time, in milliseconds, allocated for processing a thread.
SERVER_CPU_TIME_COUNT	The number of threads that run in the server.
DB_TIME_AVG	The average time for processing an SQL statement.
DB_TIME_MIN	The minimum time for processing an SQL statement.

COLUMN NAME	DESCRIPTION
DB_TIME_MAX	The maximum time for processing an SQL statement.
DB_TIME_COUNT	The number of SQL statements processed by the database.
FS_TIME_AVG	The average access time to the file system.
FS_TIME_MIN	The minimum access time to the file system.
FS_TIME_MAX	The maximum access time to the file system.
FS_TIME_COUNT	The number of file system accesses (read/write or delete file).

Part 2: Project Customization

Chapter 13: Project Customization at a Glance

As an HP Application Lifecycle Management (ALM) project administrator, you use Project Customization to control access to a project by defining the users who can access the project and by determining the types of tasks each user can perform. You can also customize a project to meet the specific requirements of your organization.

Cross Project Customization: As an ALM template administrator, you can customize a template project and apply the customization to one or more ALM projects using cross project customization. This enables you to standardize policies and procedures across projects in your organization. For more information, see "Cross Project Customization" on page 332.

ALM Editions: Cross project customization is not available for Quality Center Enterprise Edition.

This chapter includes:

Starting Project Customization 241

Understanding the Project Customization Window 243

Saving Customization Changes 245

Starting Project Customization

You can customize your ALM projects using the Project Customization window.

Note: Users belonging to the Viewers group cannot view or change any settings in the Project Customization window, except settings on the User Properties page.

To start project customization:

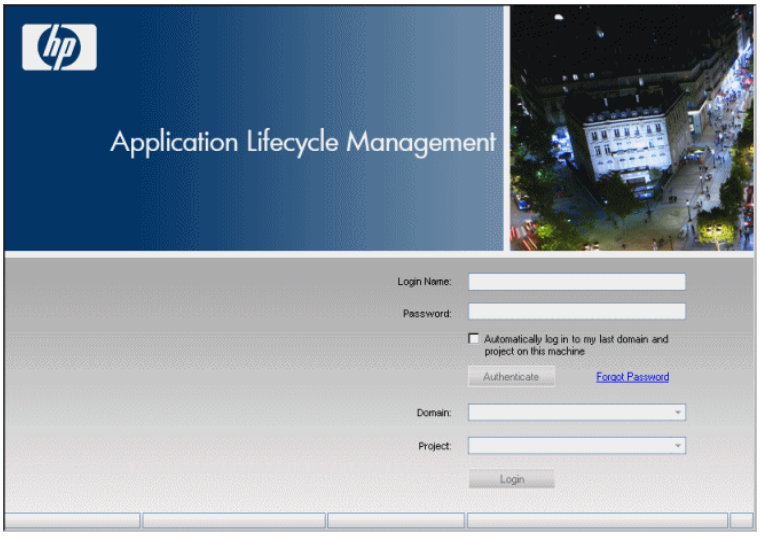
1. Open your Web browser and type your ALM URL `http://<ALM server name>[<:port number>]/qcbin`. The HP Application Lifecycle Management Options window opens.

2. Click the **Application Lifecycle Management** link.

   The first time you run ALM, files are downloaded to your workstation. Subsequently, ALM carries out a version check. If there is a newer version on the server, updated files are downloaded to your workstation.

   - **Windows XP** and **Windows 7**: If you do not have administrator privileges on your machine, and a Security Warning displays, click **Don't Install**. You will be redirected to the Install screen.
   - If file downloads are prohibited through your browser, you can install these files by using the HP ALM Client MSI Generator add-in, available from the HP Application Lifecycle Management Add-ins page (**Help > Add-ins**).
   - If you run ALM over a virtual environment, such as Citrix or VMware, only the system administrator can install a new version.

After the ALM version has been checked and files have been updated if necessary, the ALM Login window opens.



3. In the **Login Name** box, type your user name.

   If you type a user name that does not have administrator privileges for a particular project, you are restricted to the customization functions available for that user group. For more information, see "About Managing User Groups and Permissions" on page 253.

4. In the **Password** box, type your password. If you cannot remember your password, click the **Forgot Password** link. For more information, refer to the *HP Application Lifecycle Management User Guide*.

   After you log in to ALM, you can change your password from the Project Customization window. For more information, refer to the *HP Application Lifecycle Management User Guide*. In addition, site administrators can change a user's password from Site Administration. For more information, see "Creating and Changing Passwords" on page 139.

5. Select the **Automatically log in to my last domain and project on this machine** check box if you want ALM to automatically log in to the last project in which you were working.

6. Click **Authenticate**. ALM verifies your user name and password and determines which domains and projects you may access. If you specified automatic login, ALM opens.

7. In the **Domain** list, select a domain. By default, the last domain in which you were working is displayed.

8. In the **Project** list, select a project. By default, the last project in which you were working is displayed.

9. Click **Login**. ALM opens and displays the module in which you last worked during your previous session.

10. Choose **Tools > Customize**. For more information, see "Understanding the Project Customization Window" on next page.

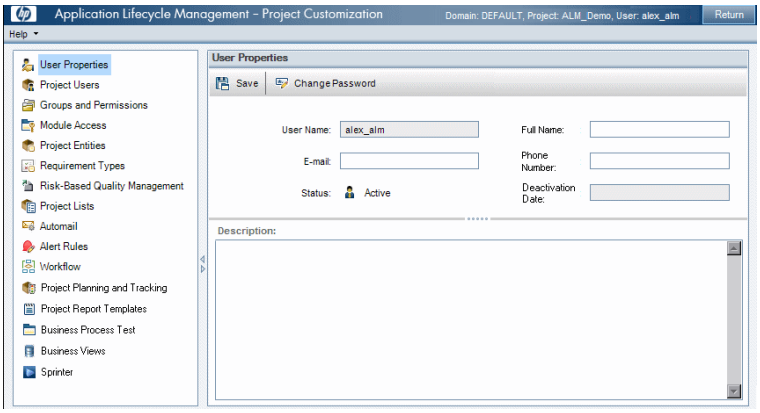
11. To exit the Project Customization window and return to your ALM project, click the **Return** button located on the upper-right corner of the window.

    If you have made changes to Project Customization, the Customization Changes dialog box opens. For more information, see "Saving Customization Changes" on page 245.

Understanding the Project Customization Window

As an ALM project administrator, you can customize a project to meet the specific requirements of your organization in the Project Customization window.

For important considerations when making changes to project customization, see "Saving Customization Changes" on page 245.



The Project Customization window contains the following links.

Note: If you are working in a template project, some links are displayed with **(Shared)** after the name. This indicates that the customization on that page is applied to linked projects. For more information, see "Cross Project Customization" on page 332.

Link	Description
User Properties	All users can use this option to change their user properties and password. For more information, refer to the *HP Application Lifecycle Management User Guide*. In Site Administration, a site administrator can override and change a user's properties and password from the **Site Users** tab. For more information, see "Updating User Details" on page 138, and "Creating and Changing Passwords" on page 139. A project administrator cannot change a user's properties from Project Customization, unless the "ALLOW_UPDATE_USER_PROPERTIES_FROM_CUSTOMIZATION" on page 172 parameter is set to **Y**.

Link	Description
Project Users	You can add and remove users from an ALM project. You can also assign users to user groups to restrict user access privileges. For more information, see "Managing Users in a Project" on page 248. **Note:** You create ALM users and define user properties from Site Administration. For more information, see "Managing ALM Users" on page 130
Groups and Permissions	You can assign privileges to user groups by specifying permission settings. This includes specifying transition rules and hiding data. For more information, see "Managing User Groups and Permissions" on page 252.
Module Access	You can control the modules that each user group can access. By preventing users from accessing unnecessary modules, you can better utilize your ALM licenses. For more information, see "Customizing Module Access for User Groups" on page 296.
Project Entities	You can customize your ALM project to suit your environment. A project can contain system fields and user-defined fields. System fields can be modified. User-defined fields can be added, modified, and deleted. For more information, see "Customizing Project Entities" on page 299.
Requirement Types	You can add requirement types to your ALM project and define which fields are available and which fields are required for each requirement type. For more information, see "Customizing Project Requirement Types" on page 306.
Risk-Based Quality Management	You can customize criteria and criterion values for risk-based testing, and customize default testing efforts and testing levels. For more information, see "Customizing Risk-Based Quality Management" on page 318 **ALM Editions**: The Risk-Based Quality Management link in Project Customization is not available for ALM Essentials Edition.
Project Lists	You can add customized field lists to a project. A field list contains values that the user can enter in system fields or user-defined fields. For more information, see "Customizing Project Lists" on page 310.
Automail	You can set up automatic mail notification rules to inform users via email about defect repair activity. For more information, see "Configuring Automail" on page 314.
Alert Rules	You can activate alert rules for your project. This instructs ALM to create alerts and send email when changes occur in the project. For more information, see "Activating Alert Rules" on page 328.

Link	Description
Workflow	You can generate scripts to perform commonly needed customizations on the fields of the Defects module dialog boxes. For more information, see "Generating Workflow Scripts" on page 402. In addition, you can write scripts to customize dialog boxes in any module, and to control the actions that users can perform. For more information, see "Workflow Customization at a Glance" on page 414.
Project Planning and Tracking	You can create and customize the project planning and tracking (PPT) KPIs. For more information, see "Customizing Project Planning and Tracking KPIs" on page 342. **ALM Editions:** The Project Planning and Tracking link in Project Customization is available for ALM Edition.
Project Report Templates	You can create and customize report templates that project users can assign to template based reports. For more information, see "Project Report Templates" on page 350.
Business Process Testing	You can configure Business Process Testing. For more information, see "Configuring Business Process Testing" on page 396.
Business Views	You can create business views which can be used as a basis for creating reports in the Analysis View module. For more information, see Business Views.
Sprinter	You can configure settings for working with HP Sprinter for manual testing in ALM. For more information, see "Configuring Sprinter" on page 398. **ALM Editions:** The **Sprinter** link in Project Customization is not available for ALM Essentials Edition or Performance Center Edition.

Saving Customization Changes

As a project administrator, you can save a customization change as a major change or a minor change. The option you select determines if customization is reloaded when a user reconnects after a session expires.

- **Major change** (default)**.** When a user session expires and the user reconnects, customization reloads.

  We recommend that you use this option only for critical changes that must be made available to users as soon as possible. Limiting major changes enables users to quickly reconnect without reloading customization. For example, you might use this option when adding a required user-defined field.

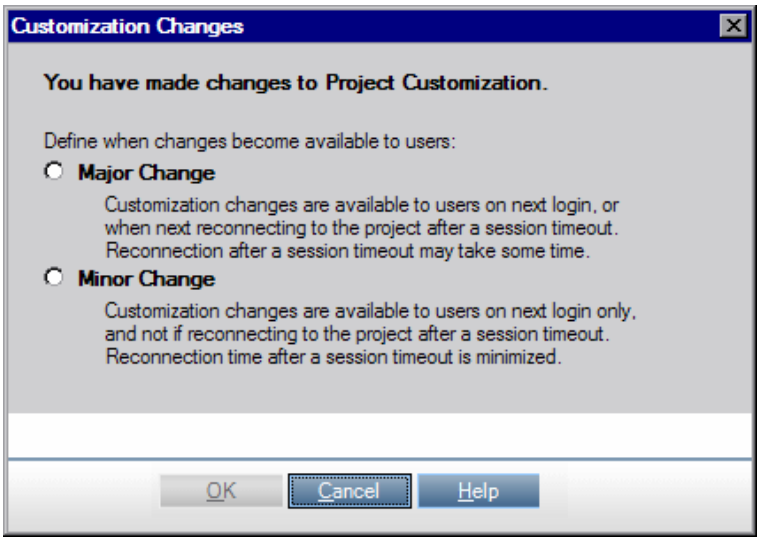
- **Minor change.** When a user session expires and the user reconnects, customization does not reload.

  If at least one major customization change has been made since the last login, customization is reloaded when a user reconnects, for details, see "DISPLAY_LAST_USER_INFO" on page

178. This includes all major and minor customization changes that have been made between the last time the user logged in and the current login time.

To select a save option for a customization change:

1. After making changes to Project Customization, click the **Return** button to exit the Project Customization window. The Customization Changes dialog box opens.



2. Select a save option and click **OK** to exit Project Customization and return to your ALM project.

Chapter 14: Managing Users in a Project

As an HP Application Lifecycle Management (ALM) project administrator, you can control access to a project by defining the users who can log in to the project and by specifying the types of tasks each user may perform.

This chapter includes:

About Managing Users in a Project 249

Adding a User to a Project 249

Assigning Users to a User Group 250

Removing a User from a Project 251

About Managing Users in a Project

For each ALM project, you must select a list of valid users from the overall ALM users list.

Note: The users list is created in Site Administration. For more information, see "Managing ALM Users" on page 130.

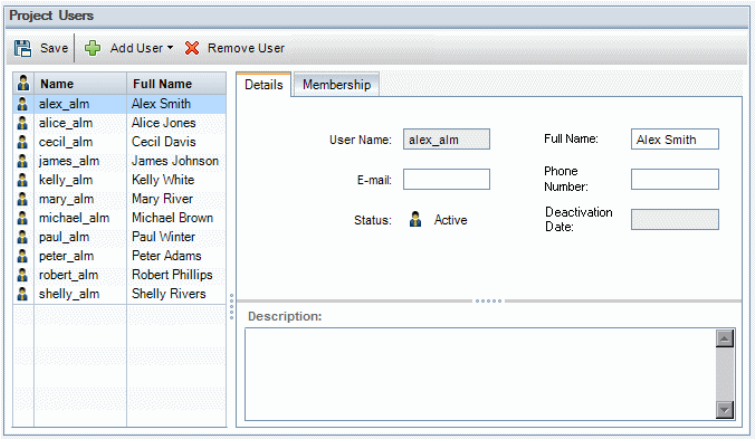
You then need to assign each project user to a user group. Each group has permissions to perform certain ALM tasks.

Adding a User to a Project

You add new users to an ALM project.

To add a user to a project:

1. In the Project Customization window, in the left pane, click **Project Users**. The Project Users page opens.



Tip: You can click the **Name** column to change the sort order from ascending to descending user names. You can also click the **Full Name** column to sort according to full names instead of user names.

2. Click the arrow to the right of the **Add User** button. Add users to the project using one of the following options:

   - To add an existing user by typing a user name, select **Add User By Name**. The Add User dialog box opens. Type the user name of a user that is defined in Site Administration for this project. Click **OK**.
   - To create a new user in the Site Users list and add the user to the project, select **Add New User to Site**. In the Add New User to Site dialog box, type the details for the new user and click **OK**.

     If this option is not available, you can enable it by setting the ADD_NEW_USERS_FROM_PROJECT parameter in Site Administration. For information, see "ADD_NEW_USERS_FROM_PROJECT" on page 167.

- To add existing users from the Site Users list, select **Add Users from Site**. In the Add Users from Site dialog box, select the users you want to add to the project.

  You can refresh the list of users with the **Refresh** button, and you can search for existing users by name with the **Find** button. Click **OK**.

The users are added to the Project Users list and the user details are displayed in the Details tab. User details are defined in Site Administration. For more information, see "Updating User Details" on page 138.

3. Click **Save** to save your changes to the Project Users page.

Assigning Users to a User Group

After you add a user to the project, you can assign the user to one or more user groups. By default, new users are assigned to the project as members of the **Viewer** user group.

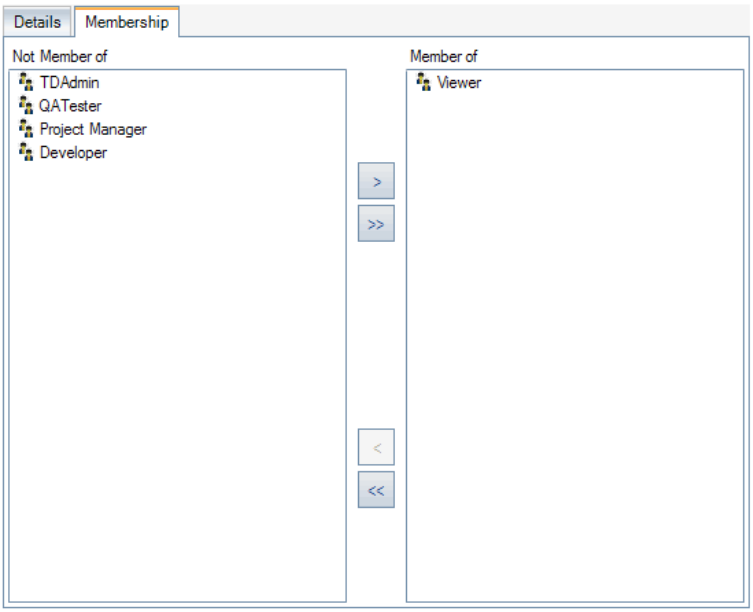
You can assign a user to a default user group, or to a customized user group. For more information on customizing a user group, see "Managing User Groups and Permissions" on page 252. You can change the access privileges for existing users at any time by changing the user group to which they are assigned.

Tip: You can also assign users to user groups from the **Groups and Permissions** page. For more information, see "Assigning Users to Groups" on page 254.

To assign a user to a user group:

1. In the Project Customization window, in the left pane, click **Project Users**. The Project Users page opens.
2. In the **Project Users** list, select the user you want to assign to a user group. The user properties are displayed (name, email, phone, and description) in the Details tab. The email information is important as it enables a user to receive defects, tests, requirements, and test set notifications directly to their mailbox.

   The user details are defined in Site Administration. For more information, see "Updating User Details" on page 138.
3. Select the **Membership** tab.



4. To assign the selected user to a user group, click a user group name in the **Not Member of** list and click the right arrow button .
5. To remove the user from the currently selected user group, click a user group name in the **Member of** list and click the left arrow button .

   Note: The **Member of** list can never be empty. A user must always belong to at least one user group.

6. To move all the user groups from one list to the other, click the double arrow buttons .
7. Click **Save** to save your changes to the Project Users page.

Removing a User from a Project

To ensure the security of a project, remove any users who are no longer working on the project. Removing a user from a project does not delete the user from the ALM users list in Site Administration.

To remove a user from a project:

1. In the Project Customization window, in the left pane, click **Project Users**. The Project Users page opens.
2. In the **Project Users** list, select the user you want to remove and click the **Remove User** button.
3. Click **OK** to confirm. The user is removed from the Project Users list.
4. Click **Save** to save your changes to the Project Users page.

Chapter 15: Managing User Groups and Permissions

You can control access to HP Application Lifecycle Management (ALM) projects and modules by defining the user groups that can enter them, and by determining the types of tasks each user group performs according to permission levels.

This chapter includes:

About Managing User Groups and Permissions 253

Adding User Groups 253

Assigning Users to Groups 254

Setting User Group Permissions 255

Setting Transition Rules 257

Hiding Data for a User Group 260

Renaming User Groups 261

Deleting User Groups 261

Understanding Permission Settings 262

Customizing Module Access for User Groups 296

About Managing User Groups and Permissions

To protect a project from unauthorized access, ALM enables you to assign each user to one or more groups. ALM includes predefined groups with default privileges. Each group has access to certain ALM tasks. The default user groups include: TDAdmin, QATester, Project Manager, Developer, and Viewer.

Performance Center:

- The following additional default user groups are available: Performance Advisor, Performance Tester, Performance Test Specialist.
- For information on permission levels specific to **Performance Center**, refer to the *HP ALM Performance Center Guide*.

When a project requires that certain user groups have privileges that are outside the scope of their default permissions, you can add your own customized user groups and assign each group a unique set of privileges.

After you set user group permissions, you can also define the ALM modules to which you want to give a user group access. When a user group member logs in to a project, only the authorized modules are displayed.

Groups and Permissions when Using Cross Project Customization

ALM template administrators use cross project customization to apply customization from a template project to one or more ALM projects. For information, see "Cross Project Customization" on page 332

ALM Editions: Cross project customization is not available for Quality Center Enterprise Edition.

If you are working with cross project customization, consider the following when setting permissions for groups:

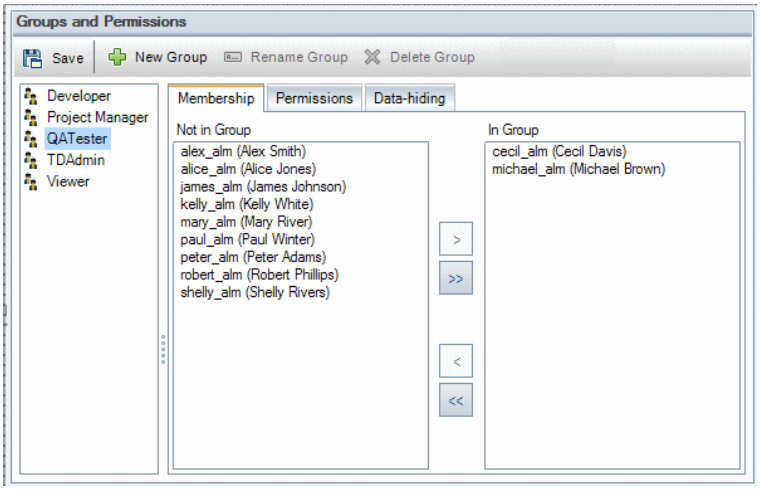
- **Working with a Template Project:** If you are working with a template project, in Project Customization, you use the **Groups (Shared)** link to manage user groups and permissions. User groups created in a template project are created in the linked projects when you apply the template customization. The users assigned to the user group in the template project are not applied to linked projects. For more information on applying template customization, see "Applying Template Customization to Linked Projects" on page 337.
- **Working with a Linked Project:** User groups defined by a template project are displayed with a template icon in the linked project. You can assign users to user groups defined by the template project. You cannot modify, rename, or delete a user group defined by the template project. You can, however, limit the records that the user group can view. For more information, see "Hiding Data for a User Group" on page 260.

Adding User Groups

If you determine that the default user groups do not meet the needs of your project, you can create additional user groups for your project. When you add a new user group, you set the group's permissions according to an existing user group.

To add a user group:

1. In the Project Customization window, in the left pane, click **Groups and Permissions**. The Groups and Permissions page opens.



2. Click the **New Group** button. A confirm message box opens. Click **Yes** to continue. The New Group dialog box opens.
3. In the **New Group Name** box, type a name for the group. A group name cannot include the following characters: **( ) @ \ / : * ? " ` < > | + = ; , %**
4. In the **Set As** list, assign the privileges of an existing user group to the new group.

   Choose an existing user group that has similar access privileges to the new user group you want to create. This minimizes the level of customization you need to do.
5. Click **OK**. The new group name is added to the group list in the Groups and Permissions page.
6. Click **Save** to save your changes to the Groups and Permissions page.

Assigning Users to Groups

After you add a user to the project, you can assign the user to one or more user groups. By default, new users are assigned to the project as members of the **Viewer** user group.

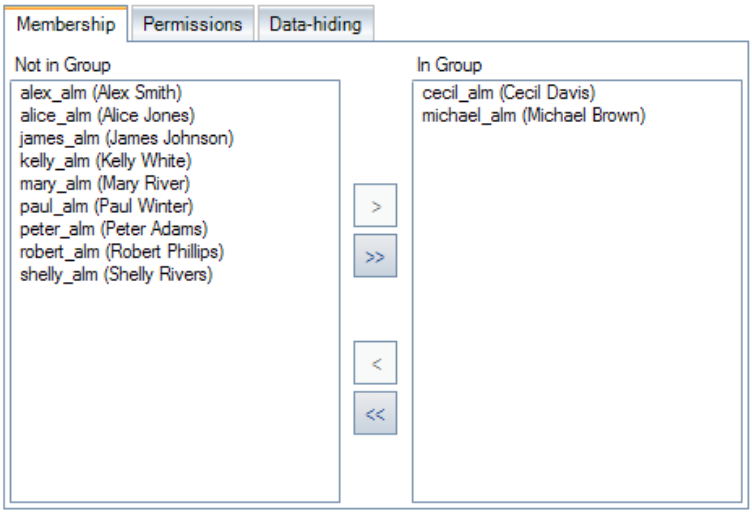
You can assign a user to a default user group, or to a customized user group. You can change the access privileges for existing users at any time by changing the user group to which they are assigned.

Tip: You can also assign users to user groups from the **Project Users** customization module. For more information, see "Managing Users in a Project" on page 248.

To assign a user to a user group:

1. In the Project Customization window, in the left pane, click **Groups and Permissions**. The Groups and Permissions page opens.
2. In the group list, select the group to which you want to assign a user.

3. Select the **Membership** tab to see which users belong to the group.



   The users assigned to the group are displayed in the **In Group** pane of the Membership tab. Users not assigned to the group are displayed in the **Not in Group** pane of the Membership tab.

4. To assign a user to the currently selected user group, select a user in the **Not in Group** list and click the right arrow button .
5. To remove a user from the currently selected user group, select a user in the **In Group** list and click the left arrow button .
6. To move all the user groups from one list to the other, click the double arrow buttons .
7. Click **Save** to save your changes to the Groups and Permissions page.

Setting User Group Permissions

Every user group has a set of privileges, or permissions, which are defined by the ALM project administrator. For example, suppose a group of users called DOC has Viewer permissions. To work more effectively on the project, they need to add, modify, and delete defects. As the ALM project administrator, you can assign these privileges to the DOC group by specifying permission settings.

Note:

- You cannot modify the privileges of a default user group. To view permissions for these groups, in the Groups and Permissions page, select the user group in the group list and click the **Permissions** tab. For more information, see "Understanding Permission Settings" on page 262.
- If you upgrade a project, and the upgraded version contains a permission that was not present in the original version, ALM automatically assigns that permission to all users in the project.

Cross Project Customization - Working with a Linked Project:

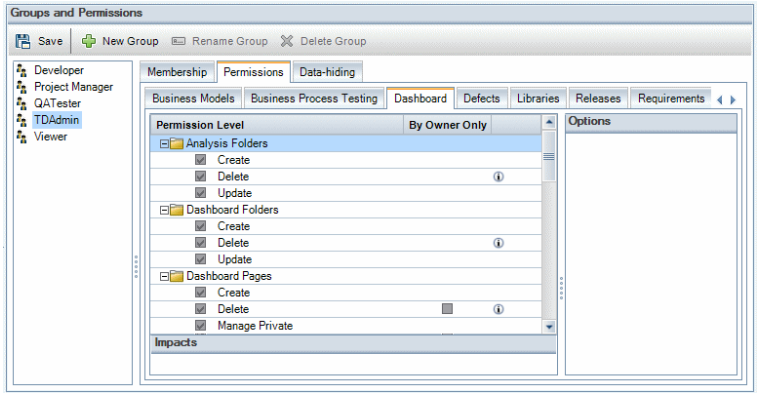
User groups defined by a template project are displayed with a template icon in the linked project. If you are working with a project that is linked to a template project, you cannot modify the permissions of a user group defined by the template project. You can, however, limit the records that the user group can view. For more information, see "Hiding Data for a User Group" on page 260.

ALM Editions: Cross project customization is not available for Quality Center Enterprise Edition.

To set user group permissions:

1. In the Project Customization window, in the left pane, click **Groups and Permissions**. The Groups and Permissions page opens.
2. In the group list, select the user group for which you want to set permissions, and click the **Permissions** tab.

   A set of tabs are displayed in alphabetical order, one tab for each ALM module plus tabs for specific purposes, such as for administration. Permission levels (such as Create, Update, and Delete) for the entities available in each module are listed under the Permission Level column.



3. Click a module tab. If necessary, to see the permission levels for each entity, expand the entity.
   - If the permissions of an entity is dependent on, or impacts, the permissions of another entity, an icon is displayed to the right of the **By Owner Only** column, and the Impacts pane at the bottom of the window displays information about the impact.
   - If additional options are available for a permission level, they are displayed in the Options pane at the right of the window.
   - If an entity's permission levels can only be modified by its owner, a check box appears in the **By Owner Only** column. For more information, see "Owning ALM Objects" on page 258.
4. Select the checkbox for the permission levels that the selected user group should have for each entity. For more information on the available permissions, see "Understanding Permission Settings" on page 262.
5. If a permission level has sublevels, expand the permission level to display the list of associated fields. Then select the fields that the selected user group can use.
6. Limit the capabilities of modifying a field as follows:

- To limit the entities whose permission levels can only be modified by their owners, select the checkbox for the permission level in the **By Owner Only** column. For example, ensure that only the person who owns the record can delete the value by selecting the checkbox in the **By Owner Only** column, next to Delete in the Permission Level column. For more information, see "Owning ALM Objects" on next page.
- To limit the values a user group can select from a lookup list type field, set transition rules of permissible field values in the Options pane. For more information, see "Setting Transition Rules" below.

7. Click the **Data-hiding** tab to hide data from the current user group by module. For more information, see "Hiding Data for a User Group" on page 260.
8. Click **Save** to save your changes to the Groups and Permissions page.

Setting Transition Rules

You can limit a group's modifying privileges by setting transition rules for modifying values in fields. These rules determine the values that the group can modify in fields that you specify. Transition rules can be set only for lookup and user list fields.

Example

When modifying defect information, you can limit the items a user group can select in the Status field of a defect record. You can set a transition rule that only allows a user group to edit the Status field from **Fixed** to **Closed**.

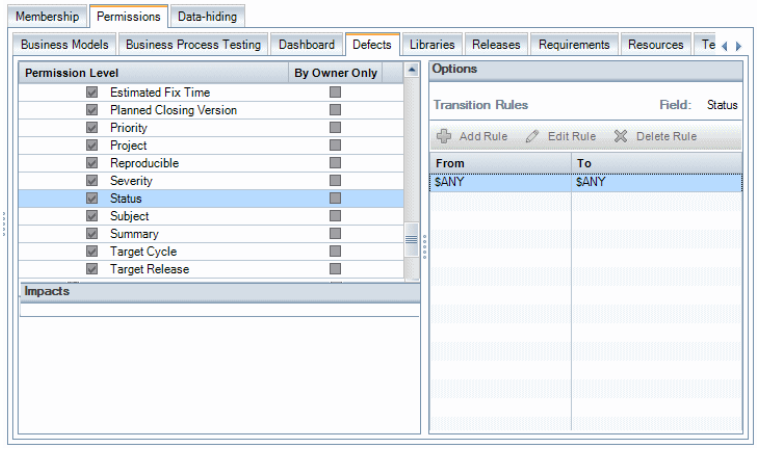
Note: When Workflow has been used to change a list of values for a field that is set with transition rules, the field may only be modified in a way that satisfies both the workflow script and the transition rules. For more information, see "Workflow Event Reference" on page 428.

To set transition rules:

1. In the Project Customization window, in the left pane, click **Groups and Permissions**. The Groups and Permissions page opens.
2. In the group list, choose the user group for which you want to set permissions.
3. Click the **Permissions** tab.
4. Click a module tab, such as **Defects**. The tab displays the entities available in the Defects module and their corresponding permission levels.
5. Expand and select an entity and its permission level. For example, expand **Defect** and then expand **Update**. The permission level expands and lists available fields.

   For more information on the available entities and permission levels, see "Understanding Permission Settings" on page 262.

6. Select a field. For example, select **Status.** The Transition Rules grid appears in the Options pane on the right side of the window.



7. Click **Add Rule** to add a transition rule. The Add Transition Rules dialog box opens.
8. Under **From**, you can:
   - Select **$ANY** to allow a user group to modify the field, irrespective of the currently displayed value.
   - Select a value from the list. A user group is able to modify the selected field only when the field displays the value you select. For example, to allow a user group to edit the Status field of a defect only if "Fixed" is the current value, select **Fixed**.
9. Under **To**, you can:
   - Select **$ANY** to allow a user group to change the field to any value.
   - Select a value from the list. A user group is able to change the value of the selected field to only the value that you specify. For example, to allow a user group to change the value of the Status field only to "Closed", select **Closed**.
10. Click **OK** to save and close the Add Transition Rules dialog box. The new rules are displayed in the Transition Rules grid.
11. To modify a transition rule, select a rule from the Transition Rules grid and click the **Edit Rule** button. In the Edit Transition Rules dialog box, modify the rule. Click **OK**.
12. To delete a transition rule, select a rule from the Transition Rules grid and click the **Delete Rule** button. Click **OK** to confirm if prompted.
13. Click **Save** to save your changes to the Groups and Permissions page.

Owning ALM Objects

When setting group permissions, you can limit the capabilities of modifying or deleting a field value so that only the user who owns the record can change or delete the value. The following table describes the objects in ALM and the users that are defined as the owners of the objects.

ALM Object	Owner
Analysis folder	The **Owner** field displays the user who created the analysis folder. The owner is updated to the current user if the analysis folder is moved (cut and paste) to the user's private folders.
Analysis item	The **Owner** field displays the user who created the analysis item. The owner is updated to the current user if the analysis item is moved (cut and paste) to the user's private folders.
Baselines	The **Created By** field displays the user who captured the baseline.
Business Components	The **Assigned To** field displays the user or user group responsible for the component.
Business Process Model Elements	The **Import By** field displays the user who imported the business process model elements.
Business Process Model Models	The **Created By** field displays the user who created the business process model models.
Business Process Model Paths	The **Created By** field displays the user who created the business process model paths.
Dashboard folder	The **Folder Owner** field displays the user who created the dashboard folder. The owner is updated to the current user if the dashboard folder is moved (cut and paste) to the user's private folders.
Dashboard page	The **Page Owner** field displays the user who created the dashboard page. The owner is updated to the current user if the dashboard page is moved (cut and paste) to the user's private folders.
Defect	The **Assigned To** field displays the user to which the defect has been assigned.
Favorites	The **Owner** field displays the user who created the favorite.
Requirement	The **Author** field displays the user who created the requirement.
Test in the Test Plan module	The **Designer** field displays the user who created the test in the test plan module.
Resource in the Test Resources module	The **Created By** field displays the user who created the resource in the test resources module.
Test Configuration	The **Created By** field displays the user who created the test configuration.

ALM Object	Owner
Test in the Test Lab module	The **Responsible Tester** field displays the user testing the test in the test lab module.
Test run in the Test Lab module	The **Tester** field displays the user testing the test run in the test lab module.

Note: You can change the owner of an ALM object by modifying the value of **TB_OWNER_FIELD_NAME** in the **Tables** table. For more information on the **Tables** table, refer to the *HP ALM Project Database Reference*.

Hiding Data for a User Group

You can instruct ALM to hide specific records that a user group can view. You can hide records related to defects, libraries, requirements, business components, resources, tests, and test sets. Hiding records includes the following options:

- **Filtering Data.** You can set filters for specific fields, limiting the records that the user group can view. For example, you can set the filter for the field **Assigned To** to "[CurrentUser]". This instructs ALM to display only the records that are assigned to the current user.

  Note: If you are filtering requirements for a user group according to the **Req Parent** field, all requirements under the selected parent are displayed in the requirements grid. The requirements tree does not display any requirements.

  For more information on filtering, refer to the *HP Application Lifecycle Management User Guide*.

- **Defining Visible Fields.** You can select which fields in a module the user group can see and which should be hidden. Users belonging to a specific user group need to view only data that relates to their work. For example, you may want to hide the Path field in the Test Plan module from user groups that should not be able to access test scripts from the file system. You cannot hide required fields.

To hide data:

1. In the Project Customization window, in the left pane, click **Groups and Permissions**. The Groups and Permissions page opens.
2. In the group list, choose the user group for which you want to hide data.
3. Click the **Data-hiding** tab.
4. Click the entity for which you want to hide data. For example, click **Defect**. The pane at the right displays the filters currently set and the fields currently visible in the Defects module to the users of the selected group.
5. Click the **Set Filter/Sort** button. The Filter <entity> dialog box opens.
6. Set one or more filters. The filter determines the records that a user group can view in ALM. For

more information, refer to the *HP Application Lifecycle Management User Guide*.

7. Click **OK** to close the Filter <entity> dialog box. The filters you set are displayed.
8. Click the **Set Visible Fields** button. The Select Columns dialog box opens.
9. Click the arrows to hide or display each field.
10. Click **OK** to close the Select Columns dialog box. The fields you set as visible are displayed.
11. Click **Save** to save your changes to the Groups and Permissions page.

Cross Project Customization

ALM Editions: Cross project customization is not available for Quality Center Enterprise Edition.

If you are working with a project that is linked to a template project, you cannot hide data for any field defined by the template project.

For user groups defined by the template project, you can hide the data of user-defined fields defined by the project. In the Groups and Permissions page, select the user group in the group list, and click the **Data-hiding** tab to determine which data is visible.

Renaming User Groups

You can rename a user group. All customization performed on the group remains.

Cross Project Customization - Working with a Linked Project: If you are working with a project that is linked to a template project, you cannot rename a user group defined by the template project.

ALM Editions: Cross project customization is not available for Quality Center Enterprise Edition.

To rename a user group:

1. In the Project Customization window, in the left pane, click **Groups and Permissions**. The Groups and Permissions page opens.
2. In the group list, select a group name.
3. Click the **Rename Group** button. The Rename Group dialog box opens.
4. Type a new name for the group.
5. Click **OK** to save your changes.

Deleting User Groups

You can delete user groups that were added to an ALM project.

ALM Editions: Cross project customization is not available for Quality Center Enterprise Edition.

Cross Project Customization. If you are working with cross project customization, consider the following:

- **Working with a Template Project:** If you delete a user group from a template project, ALM does not delete the group from the linked projects. After you next apply template customization

to the linked projects, the user group is no longer read-only in the project and can be modified, renamed, or deleted by the project administrator.

- **Working with a Linked Project:** If you are working with a project that is linked to a template project, you cannot delete a user group defined by the template project.

To delete a user group:

1. In the Project Customization window, in the left pane, click **Groups and Permissions**. The Groups and Permissions page opens.
2. In the group list, select a group name.
3. Click the **Delete Group** button.
4. Click **Yes** to confirm.

Understanding Permission Settings

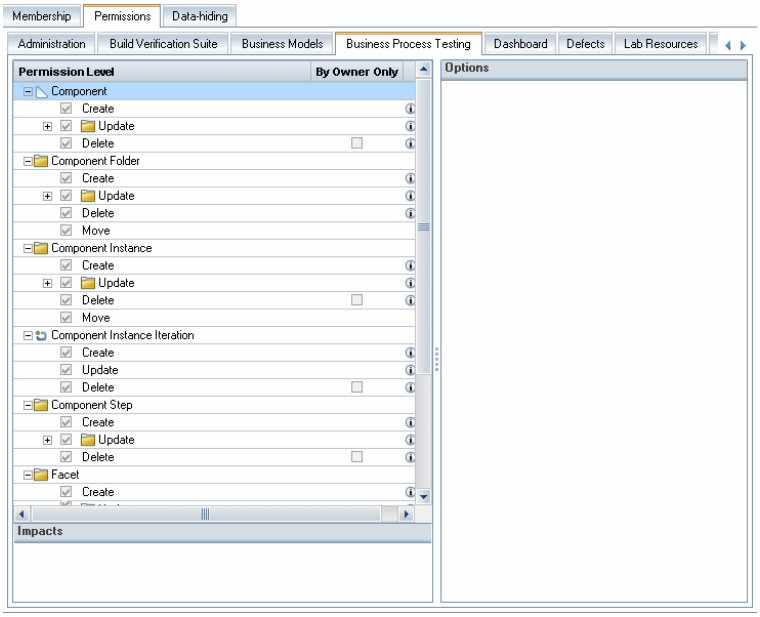
You can display the permissions of user groups in the Permissions tab. You can modify the permissions of custom user groups at any time. You cannot modify the permissions of the default user groups (TDAdmin, QATester, Project Manager, Developer, and Viewer).

Note:

- **ALM Editions:** Some permission settings may not be applicable, according to the available functionality with each edition.
- **Cross Project Customization - Working with a Linked Project:** If you are working with a project that is linked to a template project, you cannot modify the permissions of user groups defined in the template. You can, however, limit the records that the user group can view. For more information, see "Hiding Data for a User Group" on page 260.

  ALM Editions: Cross project customization is not available for Quality Center Enterprise Edition.
- **Performance Center:** The following additional default user groups are available: Performance Advisor, Performance Tester, Performance Test Specialist.
- **Performance Center:** For information on permission levels specific to Performance Center, refer to the *HP ALM Performance Center Guide*.

To display permissions for a user group, in the Groups and Permissions page, select the user group in the group list, and click the **Permissions** tab.



The Permissions tab contains the following tabs:

Administration Permission Levels 263
Build Verification Suite Permission Levels 266
Business Models Permission Levels 267
Business Process Testing Permission Levels 269
Dashboard Permission Levels 273
Defects Permission Levels 276
Lab Resources Permission Levels 277
Lab Settings Permission Levels 278
Libraries Permission Levels 278
Releases Permission Levels 280
Requirements Permission Levels 283
Resources Permission Levels 286
Test Lab Permission Levels 287
Test Plan Permission Levels 292
Timeslots Permission Levels 295

Administration Permission Levels

The Administration tab displays the following administrative tasks available in ALM.

Permission Level	Description
Add Public Favorite View Folders	User group can add public favorite view folders.
Add Public Favorite Views	User group can add public favorite views.
Allow Major Changes	User group can save a customization change as a major change. For more information, see "Saving Customization Changes" on page 245.
Change User Properties & Password	User group can change its members' properties and passwords, using the **User Properties** link in the Project Customization window.
Clear History	User group can clear the information displayed in the History table. For instructions on clearing history, refer to the *HP Application Lifecycle Management User Guide*.
Configure Automail	User group can set up a mailing configuration to routinely inform users about defect repair activity, using the **Automail** link in the Project Customization window.
Customize Module Access	User group can decide the type of access a user group can have for ALM, using the **Module Access** link in the Project Customization window.
Customize Project Entities	User group can customize fields in an ALM project, using the **Project Entities** link in the Project Customization window.
Customize Project Lists	User group can add their own customized lists to a project, using the **Project Lists** link in the Project Customization window.
Customize Report Templates	User group can customize report templates.
Customize Requirement Types	User group can customize requirement types in an ALM project, using the **Requirement Types** link in the Project Customization window.
Customize Risk-Based Quality Management	User group can customize criteria and default settings for risk-based testing using the **Risk-Based Quality Management** link in the Project Customization window.
Customize Sprinter	User group can customize Sprinter for running tests manually in ALM.

Permission Level	Description
Delete Public Favorite View Folders	User group can delete public favorite view folders. **Note:** Selecting this option also selects **Delete Public Favorite Views**.
Delete Public Favorite Views	User group can delete public favorite views. To ensure that only the owner can delete public favorite views, select **By Owner Only**. **Note:** Clearing this option also clears **Delete Public Favorite View Folders**.
Manage Analysis Menus	User group can view and manage analysis items in the Analysis View module's Analysis Menu tab.
Manage Business Views	User group can create and manage business views.
Manage Private Favorite Views	User group can manage private favorite views.
Manage Project Planning and Tracking	User group can manage PPT releases in the Releases module.
Modify Public Favorite View Folders	User group can modify public favorite view folders.
Modify Public Favorite Views	User group can modify public favorite views. To ensure that only the owner can modify public favorite views, select **By Owner Only**.
Set Up Alert Rules	User group can set up alert rules, using the **Alert Rules** link in the Project Customization window.
Set Up Cross Project Customization	User group can manage cross project customization in template and linked projects, using the **Cross Project Customization** link in the Project Customization window.
Set Up Groups	User group can assign privileges to user groups and specify permission settings, using the **Groups and Permissions** link in the Project Customization window.
Set Up Project Users	User group can add and remove users from an ALM project, using the **Project Users** link in the Project Customization window.
Set Up Workflow	User group can write and/or generate scripts that dynamically change the user interface in the ALM modules, using the **Workflow** link in the Project Customization window.

Permission Level	Description
Undo Checkouts	User group can cancel the check out of a versioned entity that was checked out by another user. For more information on version control, refer to the *HP Application Lifecycle Management User Guide*.

Build Verification Suite Permission Levels

The Build Verification Suite tab displays the entities available in the Build Verification module and their corresponding permission levels. The entities are listed below in alphabetical order.

Entity > Permission Level	Description
Build Verification Suite > Create and Update	User group can create and update build verification suites. **Note:** Clearing this option also clears **Build Verification Suite > Move**.
Build Verification Suite > Delete	User group can delete build verification suites. **Note:** Clearing this option also clears **Build Verification Suite Folder > Delete**.
Build Verification Suite > Move	User group can move build verification suites in the tree. **Note:** Selecting this option also selects **Build Verification Suite > Create and Update**. Clearing this option also clears **Build Verification Suite Folder > Move**.
Build Verification Suite Folder > Create and Update	User group can create and update build verification suite folders. **Note:** Clearing this option also clears **Build Verification Suite Folder > Move**.
Build Verification Suite Folder > Delete	User group can delete build verification suite folders. **Note:** Clearing this option also clears **Build Verification Suite > Delete**.
Build Verification Suite Folder > Move	User group can move build verification suite folders in the tree. **Note:** Selecting this option also selects **Build Verification Suite Folder > Create and Update** and **Build Verification Suite > Move**.

Business Models Permission Levels

The Business Models tab displays the entities available in the Business Models module and their corresponding permission levels. The entities are listed below in alphabetical order.

Entity > Permission Level	Description
Model > Create	User group can add business process models. **Note:** Selecting this option also selects **Model Activity > Create**, **Model Path > Create**, **Model > Update**, and **Requirement > Create** in the **Requirements** tab. Clearing this option also clears **Model > Import**. This option is cleared when you clear **Requirement > Create** in the **Requirements** tab.
Model > Delete	User group can delete business process models. To ensure that only the owner can delete business process models, select **By Owner Only**. **Note:** Selecting this option also selects **Model Path > Delete**. Clearing this option also clears **Model > Import** and **Model Folder > Delete**.
Model > Import	User group can import business process models. To ensure that only the owner can import business process models, select **By Owner Only**. **Note:** Selecting this option also selects **Model > Update**, **Model > Create**, **Model > Delete**, **Model Activity > Create**, **Model Activity > Update**, and **Requirement > Create** in the **Requirements** tab. This option is cleared when you clear **Requirement > Create** in the **Requirements** tab.
Model > Update	User group can update business process models. This permission level enables you to specify the fields that the selected user group can modify. To ensure that only the owner can modify the fields, select **By Owner Only**. **Note:** Selecting this option also selects **Model Activity > Update** and **Model Path > Update**. Clearing this option also clears **Model > Create** and **Model > Import**.
Model Activity > Create	User group can add model activities. **Note:** Selecting this option also selects **Model Activity > Update**, and **Requirement > Create** in the **Requirements** tab. Clearing this option also clears **Model > Create** and **Model > Import**. This option is cleared when you clear **Requirement > Create** in the **Requirements** tab.

Entity > Permission Level	Description
Model Activity > Update	User group can update model activities. This permission level enables you to specify the fields that the selected user group can modify. To ensure that only the owner can modify the fields, select **By Owner Only**. **Note:** Clearing this option also clears **Model > Update**, **Model > Import**, and **Model Activity > Create**.
Model Folder > Create	User group can add model folders. **Note:** Selecting this option also selects **Model Folder > Update**.
Model Folder > Delete	User group can delete model folders. **Note:** Selecting this option also selects **Model > Delete**.
Model Folder > Update	User group can update model folders. This permission level enables you to specify the fields that the selected user group can modify. **Note:** Clearing this option also clears **Model Folder > Create**.
Model Linkage > Create	User group can add model linkage. **Note:** Selecting this option also selects **Model Linkage > Update**.
Model Linkage > Delete	User group can delete model linkage.
Model Linkage > Update	User group can update model linkage. **Note:** Clearing this option also clears **Model Linkage > Create**.
Model Path > Create	User group can add model paths. **Note:** Selecting this option also selects **Model Path> Update**, **Model Path > Delete**, and **Requirement > Create** in the **Requirements** tab. Clearing this option also clears **Model > Create**. This option is cleared when you clear **Requirement > Create** in the **Requirements** tab.

Entity > Permission Level	Description
Model Path > Delete	User group can delete model paths. To ensure that only the owner can delete model paths, select **By Owner Only**. **Note:** Clearing this option also clears **Model Path> Create** and **Model > Delete**.
Model Path > Update	User group can update model paths. This permission level enables you to specify the fields that the selected user group can modify. To ensure that only the owner can modify the fields, select **By Owner Only**. **Note:** Clearing this option also clears **Model Path> Create** and **Model > Update**.

Business Process Testing Permission Levels

The Business Components tab displays the entities available in the Business Components module and their corresponding permission levels. The entities are listed below in alphabetical order.

Entity > Permission Level	Description
Component > Create	Enables you to create components in the component tree. **Note:** Selecting this option also selects **Component > Update**, **Implementation > Create**, and **Parameter > Create**. Clearing this option also clears **Test > Create** in the **Test Plan** tab. This option is selected when **Test > Create** in the **Test Plan** tab is selected.
Component > Delete	Enables you to delete components from the component tree. To ensure that only the owner can delete components, select **By Owner Only**. **Note:** Selecting this option also selects **Implementation > Delete** and **Parameter > Delete**. Clearing this option also clears **Component Folder > Delete**, and **Test > Delete** in the **Test Plan** tab. This option is selected when **Test > Delete** in the **Test Plan** tab is selected.

Entity > Permission Level	Description
Component > Update	Enables you to update component fields by indicating the fields that the selected user group can modify. To ensure that only the owner can modify the fields, select **By Owner Only**. **Note:** Selecting this option also selects **Implementation > Update** and **Parameter > Update**. Clearing this option also clears **Component > Create**, and **Test > Update** in the **Test Plan** tab. This option is selected when **Test > Update** in the **Test Plan** tab is selected.
Component Folder > Create	Enables you to add folders to the component tree. **Note:** Selecting this option also selects **Component Folder > Update**.
Component Folder > Delete	Enables you to delete folders from the component tree. **Note:** Selecting this option also selects **Component > Delete**.
Component Folder > Move	User group can move component folders to different folders in the component tree.
Component Folder > Update	Enables you to modify folders in the component tree. This permission level enables you to specify the fields that the selected user group can modify. **Note:** Clearing this option also clears **Component Folder > Create**.
Component Instance > Create	Enables you to add component instances. **Note:** Selecting this option also selects **Component Instance > Update** and **Component Instance Iteration > Create**. Clearing this option also clears **Test > Create** in the **Test Plan** tab. This option is selected when **Test > Create** in the **Test Plan** tab is selected.
Component Instance > Delete	Enables you to delete component instances. To ensure that only the owner of the component instance can delete it, select **By Owner Only**. **Note:** Selecting this option also selects **Component Instance Iteration > Delete**. Clearing this option also clears **Test > Delete** in the **Test Plan** tab. This option is selected when **Test > Delete** in the **Test Plan** tab is selected.

Entity > Permission Level	Description
Component Instance > Move	User group can move component instances.
Component Instance > Update	Enables you to modify component instances. This permission level enables you to specify the fields that the selected user group can modify. To ensure that only the owner can modify the fields, select **By Owner Only**. **Note:** Selecting this option also selects **Component Instance Iteration > Update**. Clearing this option also clears **Component Instance > Create**, and **Test > Update** in the **Test Plan** tab. This option is selected when **Test > Update** in the **Test Plan** tab is selected.
Component Instance Iteration > Create	Enables you to add component instance iterations. **Note:** Selecting this option also selects **Component Instance Iteration > Update**. Clearing this option also clears **Component Instance > Create**.
Component Instance Iteration > Delete	Enables you to delete component instance iterations. To ensure that only the owner of the component instance iteration can delete it, select **By Owner Only**. **Note:** Clearing this option also clears **Component Instance > Delete**.
Component Instance Iteration > Update	Enables you to modify component instance iterations. **Note:** Clearing this option also clears **Component Instance Iteration > Create** and **Component Instance > Update**.
Component Step > Create	Enables you to add design steps to the component. **Note:** Selecting this option also selects **Component Step > Update**. Clearing this option also clears **Implementation > Create**.
Component Step > Delete	Enables you to delete design steps from the component. To ensure that only the owner can delete design steps from the component, select **By Owner Only**. **Note:** Clearing this option also clears **Implementation > Delete**.

Entity > Permission Level	Description
Component Step > Update	Enables you to modify design steps in the component. This permission level enables you to specify the fields that the selected user group can modify. To ensure that only the owner can modify the fields, select **By Owner Only**. **Note:** Clearing this option also clears **Component Step > Create** and **Implementation > Update**.
Implementation > Create	Enables you to create manual and/or automated implementations. **Note:** Selecting this option also selects **Component Step > Create** and **Implementation > Update**. Clearing this option also clears **Component > Create**.
Implementation > Delete	Enables you to delete manual and/or automated implementations. To ensure that only the owner can delete Implementations, select **By Owner Only**. **Note:** Selecting this option also selects **Component Step > Delete**. Clearing this option also clears **Component > Delete**.
Implementation > Update	Enables you to update manual and/or automated implementations by indicating the fields that the selected user group can modify. To ensure that only the owner can modify the fields, select **By Owner Only**. **Note:** Selecting this option also selects **Component Step > Update**. Clearing this option also clears **Component > Update** and **Implementation > Create**.
Parameter > Create	Enables you to create parameters. **Note:** Selecting this option also selects **Parameter > Update**. Clearing this option also clears **Component > Create**.
Parameter > Delete	Enables you to delete parameters. To ensure that only the owner can delete parameters, select **By Owner Only**. **Note:** Clearing this option also clears **Component > Delete**.

Entity > Permission Level	Description
Parameter > Update	Enables you to update parameters by indicating the fields that the selected user group can modify. To ensure that only the owner can modify the fields, select **By Owner Only**. **Note:** Clearing this option also clears **Component > Update** and **Parameter > Create**.

Dashboard Permission Levels

The Dashboard tab displays the entities available in the Dashboard module and their corresponding permission levels.

Entity > Permission Level	Description
Analysis Folders > Create	User group can add public analysis folders.
Analysis Folders > Delete	User group can delete public analysis folders. **Note:** Selecting this option also selects **Graphs > Delete**, **Standard Reports > Delete**, **Excel Reports > Delete**, **Project Reports > Delete**, and **Business Views Graphs > Delete**.
Analysis Folders > Update	User group can modify public analysis folders.
Business Views Excel Reports > Create	User group can add business views Excel reports.
Business Views Excel Reports > Customize DQL Query	User group can customize DQL queries for use in creating business views Excel reports. To ensure that only the owner can delete business views Excel reports, select **By Owner Only**.
Business Views Excel Reports > Delete	User group can delete business views Excel reports. To ensure that only the owner can delete business views Excel reports, select **By Owner Only**. **Note:** Clearing this option also clears **Analysis Folders > Delete**.

Entity > Permission Level	Description
Business Views Excel Reports > Manage Private	User group can manage private business views Excel reports in the analysis tree.
Business Views Excel Reports > Update	User group can update business views Excel reports. To ensure that only the owner can update business views Excel reports, select **By Owner Only**.
Business Views Graphs > Allow Cross Project Graphs	User group can include multiple projects in business views graphs. If this permission level is not selected, the user group can create business views graphs for the current project only. **Caution:** Cross-project analysis items use many system resources. To avoid a reduction in system performance, you should use this permission selectively.
Business Views Graphs > Create	User group can add business views graphs.
Business Views Graphs > Delete	User group can delete business views graphs. To ensure that only the owner can delete business views graphs, select **By Owner Only**. **Note:** Clearing this option also clears **Analysis Folders > Delete**.
Business Views Graphs > Manage Private	User group can manage private business views graphs in the analysis tree.
Business Views Graphs > Share	User group can share business views graphs.
Business Views Graphs > Update	User group can update business views graphs. To ensure that only the owner can update business views graphs, select **By Owner Only**.
Dashboard Folders > Create	User group can add public dashboard folders.
Dashboard Folders > Delete	User group can delete public dashboard folders. **Note:** Selecting this option also selects **Dashboard Pages > Delete**.
Dashboard Folders > Update	User group can modify public dashboard folders.
Dashboard Pages > Create	User group can add public dashboard pages.

Entity > Permission Level	Description
Dashboard Pages > Delete	User group can delete public dashboard pages. To ensure that only the owner can delete public dashboard pages, select **By Owner Only**. **Note:** Clearing this option also clears **Dashboard Folders > Delete**.
Dashboard Pages > Manage Private	User group can manage private dashboard pages.
Dashboard Pages > Update	User group can modify public dashboard pages. To ensure that only the owner can update public dashboard pages, select **By Owner Only**.
Excel Reports > Create	User group can add public Excel reports.
Excel Reports > Delete	User group can delete public Excel reports. To ensure that only the owner can delete public Excel reports, select **By Owner Only**. **Note:** Clearing this option also clears **Analysis Folders > Delete**.
Excel Reports > Generate	User group can generate public Excel reports.
Excel Reports > Manage Private	User group can manage private Excel reports in the analysis tree.
Excel Reports > Update	User group can modify public Excel reports. To ensure that only the owner can update public Excel reports, select **By Owner Only**.
Graphs > Allow Cross Project Graphs	User group can include multiple projects in graphs. If this permission level is not selected, the user group can create graphs for the current project only. **Caution:** Cross-project analysis items use many system resources. To avoid a reduction in system performance, you should use this permission selectively.
Graphs > Create	User group can add public graphs.
Graphs > Delete	User group can delete public graphs. To ensure that only the owner can delete public graphs, select **By Owner Only**. **Note:** Clearing this option also clears **Analysis Folders > Delete**.
Graphs > Manage Private	User group can manage private graphs in the analysis tree.
Graphs > Share	User group can share graphs.

Entity > Permission Level	Description
Graphs > Update	User group can update public graphs. To ensure that only the owner can update public graphs, select **By Owner Only**.
Project Reports > Allow Custom Templates	User group can assign custom templates to project reports.
Project Reports > Create	User group can add public project reports.
Project Reports > Delete	User group can delete public project reports. To ensure that only the owner can delete public project reports, select **By Owner Only**. **Note:** Clearing this option also clears **Analysis Folders > Delete**.
Project Reports > Manage Private	User group can manage private project reports in the tree.
Project Reports > Share	User group can share public project reports.
Project Reports > Update	User group can modify public project reports. To ensure that only the owner can update public project reports, select **By Owner Only**.
Standard Reports > Create	User group can add public standard reports.
Standard Reports > Delete	User group can delete public standard reports. To ensure that only the owner can delete public standard reports, select **By Owner Only**. **Note:** Clearing this option also clears **Analysis Folders > Delete**.
Standard Reports > Manage Private	User group can manage private standard reports in the analysis tree.
Standard Reports > Update	User group can modify public standard reports. To ensure that only the owner can update public standard reports, select **By Owner Only**.

Defects Permission Levels

The Defects tab displays the entities available in the Defects module and their corresponding permission levels. The entities are listed below in alphabetical order.

Entity > Permission Level	Description
Defect > Create	User group can add defects to the Defects Grid. **Note:** Selecting this option also selects **Defect > Update**.
Defect > Update	User group can modify defects in the Defects Grid. This permission level enables you to specify the fields that the selected user group can modify. To ensure that only the owner of the defect can modify each field, select **By Owner Only**. **Note:** Clearing this option also clears **Defect > Create**.
Defect > Delete	User group can delete defects from the Defects Grid. To ensure that only the owner of the defect can delete a defect, select **By Owner Only**.
Link > Create	User group can add defect links to the ALM entities. **Note:** Selecting this option also selects **Link > Update**.
Link > Update	User group can modify defect links. This permission level enables you to specify the fields that the selected user group can modify. To ensure that only the owner of the defect link can modify the fields, select **By Owner Only**. **Note:** Clearing this option also clears **Link > Create**.
Link > Delete	User group can remove defect links from ALM entities. To ensure that only the owner of the defect link can remove a defect link, select **By Owner Only**.

Lab Resources Permission Levels

The Lab Resources tab displays the entities available in the Lab Resources module and their corresponding permission levels.

Entity > Permission Level	Description
Testing Host > Add/Remove Private Host	User group can add or remove a private testing host.
Testing Host > Change Status	User group can change a testing host's status.
Testing Host > Check Host	User group can perform checks on a testing host.
Testing Host > Install Patch	User group can install patches on Performance Center hosts.

Entity > Permission Level	Description
Testing Host > Kill Process	User group can kill processes on Performance Center hosts.
Testing Host > Manage DP Queue	User group can view pending data processing requests.
Testing Host > Reboot	User group can reboot testing hosts.
Testing Host > Reconfigure Host	User group can reconfigure Performance Center hosts.
AUT Environments> Manage AUT Environments	User group can manage AUT Environments entities.

Lab Settings Permission Levels

The Lab Settings tab displays the entities available in the Lab Settings module and their corresponding permission levels.

Entity > Permission Level	Description
Project Settings > Manage Autostart Retries	User group can manage autostart retries information.
Project Settings > Manage Controller Options	User group can define global Controller options.
Project Settings > Manage Timeslot Alerts	User group can manage timeslot alert settings.

Libraries Permission Levels

The Libraries tab displays the entities available in the Libraries module and their corresponding permission levels. The entities are listed below in alphabetical order.

Entity > Permission Level	Description
Baseline > Create	User group can create baselines for libraries. **Note:** Selecting this option also selects **Baseline > Update**.

Entity > Permission Level	Description
Baseline > Delete	User group can delete baselines. To ensure that only the owner of the baseline can delete a baseline, select **By Owner Only**. **Note:** Clearing this option also clears **Library > Delete** and **Library folder > Delete**.
Baseline > Update	User group can modify baselines. This permission level enables you to specify the fields that the selected user group can modify. To ensure that only the owner of the baseline can modify the fields, select **By Owner Only**. **Note:** Clearing this option also clears **Baseline > Create**.
Library > Compare libraries and baselines	User group can compare libraries and baselines in the libraries tree. To ensure that only the owner of the library can compare libraries and baselines, select **By Owner Only**. **Note:** Clearing this option also clears **Library > Synchronize library to baseline**.
Library > Create	User group can add libraries to library folders in the libraries tree. **Note:** Selecting this option also selects **Library > Update**.
Library > Delete	User group can delete libraries from the libraries tree. To ensure that only the owner of the library can delete a library, select **By Owner Only**. **Note:** Selecting this option also selects **Baseline > Delete**. Clearing this option also clears **Library folder > Delete**.
Library > Import Library	User group can import a library to the libraries tree. To ensure that only the owner of the library can import a library, select **By Owner Only**.
Library > Move library	User group can move libraries to different library folders in the libraries tree. To ensure that only the owner of the library can move a library, select **By Owner Only**.

Entity > Permission Level	Description
Library > Synchronize library to baseline	User group can synchronize libraries in the libraries tree. To ensure that only the owner of the library can synchronize the library to a baseline, select **By Owner Only**. **Note:** Selecting this option also selects **Library > Compare libraries and baselines**.
Library > Update	User group can modify libraries in library folders. This permission level enables you to specify the fields that the selected user group can modify. To ensure that only the owner of the library can update the fields, select **By Owner Only**. **Note:** Clearing this option also clears **Library > Create**.
Library folder > Create	User group can add library folders to the libraries tree. **Note:** Selecting this option also selects **Library Folder > Update**.
Library folder > Delete	User group can delete library folders. **Note:** Selecting this option also selects **Library > Delete** and **Baseline > Delete**.
Library folder > Move library folder	User group can move library folders to different library folders in the libraries tree.
Library folder > Update	User group can modify library folders in the libraries tree. This permission level enables you to specify the fields that the selected user group can modify. **Note:** Clearing this option also clears **Library Folder > Create**.

Releases Permission Levels

The Releases tab displays the entities available in the Releases module and their corresponding permission levels. The entities are listed below in alphabetical order.

Entity > Permission Level	Description
Cycle > Create	User group can add cycles to the releases tree. **Note:** Selecting this option also selects **Cycle > Update**. Clearing this option also clears **Release > Create**.
Cycle > Delete	User group can delete cycles from the releases tree. **Note:** Clearing this option also clears **Release > Delete**.
Cycle > Update	User group can modify cycles in the releases tree. This permission level enables you to specify the fields that the selected user group can modify. **Note:** Clearing this option also clears **Release > Update** and **Cycle > Create**.
Milestone > Create	User group can add milestones to the releases tree. **Note:** Selecting this option also selects **Milestone > Update**, **Milestone > Manage KPI Thresholds**, **Milestone > Manage KPIs**, and **Milestone > Manage Milestone Scope Items**. Clearing this option also clears **Release > Create**.
Milestone > Delete	User group can delete milestones from the releases tree. **Note:** Selecting this option also selects **Milestone > Manage KPI Thresholds**, **Milestone > Manage KPIs**, and **Milestone > Manage Milestone Scope Items**. Clearing this option also clears **Release > Delete**.
Milestone > Manage KPI Thresholds	User group can manage threshold values for KPIs. **Note:** Clearing this option also clears **Milestone > Create**, **Milestone > Delete**, **Milestone > Manage KPIs**, and **Milestone > Update**.
Milestone > Manage KPIs	User group can define KPIs for tracking the milestone scope. **Note:** Selecting this option also selects **Milestone > Manage KPI Thresholds**. Clearing this option also clears **Milestone > Delete** and **Milestone > Create**.

Entity > Permission Level	Description
Milestone > Manage Milestone Scope Items	User group can manage scope items for each milestone. **Note:** Clearing this option also clears **Milestone > Create**, **Milestone > Delete**, and **Release > Manage Scope Items**.
Milestone > Update	User group can modify milestones in the releases tree. This permission level enables you to specify the fields that the selected user group can modify. **Note:** Selecting this option also selects **Milestone > Manage KPI Thresholds**. Clearing this option also clears **Release > Update** and **Milestone > Create**.
Release > Create	User group can add releases to release folders in the releases tree. **Note:** Selecting this option also selects **Cycle > Create**, **Milestone > Create**, **Release > Update**, and **Release > Manage Scope Items**.
Release > Delete	User group can delete releases and cycles from the releases tree. **Note:** Selecting this option also selects **Milestone > Delete** and **Cycle > Delete**. Clearing this option also clears **Release folder > Delete**.
Release > Manage Scope Items	User group can manage scope items for the release. **Note:** Selecting this option also selects **Milestone > Manage Milestone Scope Items**. Clearing this option also clears **Release > Create**.
Release > Move	User group can move releases in the releases tree. **Note:** Clearing this option also clears **Release folder > Move**.
Release > Update	User group can modify releases in release folders. This permission level enables you to specify the fields that the selected user group can modify. **Note:** Selecting this option also selects **Milestone > Update** and **Cycle > Update**. Clearing this option also clears **Release > Create**.

Entity > Permission Level	Description
Release folder > Create	User group can add release folders to the releases tree. **Note:** Selecting this option also selects **Release folder > Update**.
Release Folder > Delete	User group can delete release folders, releases, and cycles from the releases tree. **Note:** Selecting this option also selects **Release > Delete**.
Release folder > Move	User group can move release folders in the releases tree. **Note:** Selecting this option also selects **Release > Move**.
Release folder > Update	User group can modify release folders in the releases tree. This permission level enables you to specify the fields that the selected user group can modify. **Note:** Clearing this option also clears **Release folder > Create**.

Requirements Permission Levels

The Requirements tab displays the entities available in the Requirements module and their corresponding permission levels. The entities are listed below in alphabetical order.

Entity > Permission Level	Description
Coverage > Create	User group can add coverage to a requirement. **Note:** Selecting this option also selects **Coverage > Update**. Clearing this option also clears **Requirement > Create**.
Coverage > Delete	User group can remove coverage from a requirement. **Note:** Clearing this option also clears **Requirement > Delete**.

Entity > Permission Level	Description
Coverage > Update	User group can modify coverage for a requirement. **Note:** Clearing this option also clears **Coverage > Create** and **Requirement > Update**.
Coverage Analysis Management> Refresh Coverage Analysis	User group can refresh coverage analysis for a requirement.
Requirement > Create	User group can add requirements to the requirements tree. **Note:** Selecting this option also selects **Coverage > Create** and **Requirement > Update**. Clearing this option also clears **Model > Create**, **Model Activity > Create**, **Model Path > Create**, and **Model > Import** in the **Business Models** tab. This option is selected when you select **Model > Create**, **Model Activity > Create**, **Model Path > Create**, or **Model > Import** in the **Business Models** tab.
Requirement > Delete	User group can delete requirements from the requirements tree. To ensure that only the owner of the requirement can delete it, select **By Owner Only**. **Note:** Selecting this option also selects **Coverage > Delete**.
Requirement > Move	User group can move requirements in the requirements tree. To ensure that only the owner of the requirement can move it, select **By Owner Only**.
Requirement > Update	User group can modify requirements in the requirements tree. This permission level enables you to specify the fields that the selected user group can modify. To ensure that only the owner of the requirement can modify the fields, select **By Owner Only**. **Note:** Selecting this option also selects **Coverage > Update**. Clearing this option also clears **Requirement > Create**.
Risk-Based Quality Management > Analyze	The user group can perform risk-based quality management analysis on a requirement and its children. For more information on risk-based quality management, see the *HP Application Lifecycle Management User Guide*.

Entity > Permission Level	Description
Risk-Based Quality Management > Assess Business Criticality	The user group can assess business criticality and override calculated analysis results of a requirement. For more information on risk-based quality management, see the *HP Application Lifecycle Management User Guide*. **Note:** Selecting this option also selects **RBQM business criticality**, **RBQM custom business criticality**, **RBQM effective business criticality**, and **RBQM use custom business criticality**, which are fields under **Requirement > Update**. Clearing this option clears the same fields. Also, this option is selected or cleared when you select or clear any of these fields.
Risk-Based Quality Management > Assess Failure Probability	The user group can assess failure probability and override calculated analysis results of a requirement. For more information on risk-based quality management, see the *HP Application Lifecycle Management User Guide*. **Note:** Selecting this option also selects **RBQM failure probability**, **RBQM custom failure probability**, **RBQM effective failure probability**, and **RBQM use custom failure probability**, which are fields under **Requirement > Update**. Clearing this option clears the same fields. Also, this option is selected or cleared when you select or clear any of these fields.
Risk-Based Quality Management > Assess Functional Complexity	The user group can assess functional complexity and override calculated analysis results of a requirement. For more information on risk-based quality management, see the *HP Application Lifecycle Management User Guide*. **Note:** Selecting this option also selects **RBQM Functional Complexity**, **RBQM custom Functional Complexity**, **RBQM effective Functional Complexity**, and **RBQM use custom Functional Complexity**, which are fields under **Requirement > Update**. Clearing this option clears the same fields. Also, this option is selected or cleared when you select or clear any of these fields.
Trace > Create	User group can add traceability links to a requirement. **Note:** Selecting this option also selects **Trace > Update**.

Entity > Permission Level	Description
Trace > Delete	User group can remove traceability links from a requirement. To ensure that only the owner of the requirement can remove the traceability link, select **By Owner Only**.
Trace > Update	User group can modify traceability links for a requirement. This permission level enables you to specify whether the user group can modify the comment for a traceability link. To ensure that only the owner of the requirement can modify the comment for a traceability link, select **By Owner Only**. **Note:** Clearing this option also clears **Trace > Create**.

Resources Permission Levels

The Test Resources tab displays the entities available in the Test Resources module and their corresponding permission levels.

Entity > Permission Level	Description
Resource > Copy	User group can copy resources to folders in the resource tree. To ensure that only the owner of a resource can copy it, select the **By Owner Only** check box.
Resource > Create	User group can add resources to the resource tree. **Note:** Selecting this option also selects **Resource > Update**.
Resource > Delete	User group can delete resources from the resource tree. To ensure that only the owner of a resource can delete it, select the **By Owner Only** check box. **Note:** Clearing this option also clears **Resource Folder > Delete**.
Resource > Move	User group can move resources to different folders in the resource tree. To ensure that only the owner of a resource can move it, select the **By Owner Only** check box. **Note:** Selecting this option also selects **Folder ID**, which is a field under **Resource > Update**. Also, this option is cleared when you clear **Folder ID**.

Entity > Permission Level	Description
Resource > Update	User group can modify resources in the resource tree and upload resources to the ALM repository. This permission level enables you to specify the fields that the selected user group can modify. To ensure that only the owner can modify a resource field, select **By Owner Only**. **Note:** Clearing this option also clears **Resource > Create**.
Resource Folder > Copy	User group can copy folders in the resource tree.
Resource Folder > Create	User group can add folders to the resource tree. **Note:** Selecting this option also selects **Resource Folder > Update**.
Resource Folder > Delete	User group can delete folders from the resource tree. **Note:** Selecting this option also selects **Resource > Delete**.
Resource Folder > Move	User group can move folders in the resource tree. **Note:** Selecting this option also selects **Parent folder ID**, which is a field under **Resource Folder > Update**. Also, this option is cleared when you clear **Parent folder ID**.
Resource Folder > Update	User group can modify folders in the resource tree. This permission level enables you to specify the fields that the selected user group can modify. **Note:** Clearing this option also clears **Resource Folder > Create**.

Test Lab Permission Levels

The Test Lab tab displays the entities available in the Test Lab module and their corresponding permission levels.

Entity > Permission Level	Description
Host > Create	User group can add hosts for running tests. **Note:** Selecting this option also selects **Host > Update**.
Host > Delete	User group can delete hosts.
Host > Update	User group can update hosts. **Note:** Clearing this option also clears **Host > Create**.
Host Group > Create	User group can add host groups for running tests. **Note:** Selecting this option also selects **Host Group > Update**.
Host Group > Delete	User group can delete host groups.
Host Group > Update	User group can modify host group information. **Note:** Clearing this option also clears **Host Group > Create**.
Result > Create	User group can add run results from an external testing tool. **Note:** Selecting this option also selects **Result > Update**. Clearing this option also clears **Run > Create**.
Result > Delete	User group can delete run results that were added by an external testing tool. **Note:** Clearing this option also clears **Run > Delete**.
Result > Update	User group can modify run results that were added by an external testing tool. **Note:** Clearing this option also clears **Run > Update** and **Result > Create**.
Run > Create	User group can run tests (meaning, create new test runs). **Note:** Selecting this option also selects **Run > Update**, **Test Instance > Update**, **Result > Create**, and **Run Step > Create**.

Entity > Permission Level	Description
Run > Delete	User group can delete test run information. To ensure that only the owner of the run can delete it, select **By Owner Only**. **Note:** Selecting this option also selects **Test Instance > Update**, **Result > Delete**, and **Run Step > Delete**. Clearing this option also clears **Run > Delete Build Verification Suite \ Test Set Run** and **Test Set > Reset**.
Run > Delete Build Verification Suite \ Test Set Run	User group can delete a Build Verification Suite Run or a Functional Test Set Run from the Test Runs module. To ensure that only the owner of the run can delete it, select **By Owner Only**. **Note:** Selecting this option also selects **Run > Delete**.
Run > Start Build Verification Suite Run	User group can start a Build Verification Suite Run. To ensure that only the owner of the run can start it, select **By Owner Only**. **Note:** Selecting this option also selects **Timeslot > Create and Update** in the **Timeslots** tab. This option is cleared when **Timeslot > Create and Update** in the **Timeslots** tab is cleared.
Run > Stop Build Verification Suite \ Test Set Run	User group can stop a Build Verification Suite Run or a Functional Test Set Run from the Test Runs module. To ensure that only the owner of the run can stop it, select **By Owner Only**. **Note:** Clearing this option also clears **Timeslot > Abort and Free** in the **Timeslots** tab. This option is selected when **Timeslot > Abort and Free** in the **Timeslots** tab is selected.
Run > Start Test Set Run	User group can start a Test Set Run. To ensure that only the owner of the run can start it, select **By Owner Only**. **Note:** Selecting this option also selects **Timeslot > Create and Update** in the **Timeslots** tab. This option is cleared when **Timeslot > Create and Update** in the **Timeslots** tab is cleared.

Entity > Permission Level	Description
Run > Stop Run	User group can stop a run from the Test Runs module. To ensure that only the owner of the run can stop it, select **By Owner Only**. **Note:** Clearing this option also clears **Timeslot > Abort and Free** in the **Timeslots** tab. This option is selected when **Timeslot > Abort and Free** in the **Timeslots** tab is selected.
Run > Update	User group can modify test run information. This permission level enables you to specify the fields that the selected user group can modify. To ensure that only the owner of the run can modify it, select **By Owner Only**. **Note:** Selecting this option also selects **Result > Update** and **Run Step > Update**. Clearing this option also clears **Run > Create**.
Run Step > Create	User group can create test steps. **Note:** Selecting this option also selects **Run Step > Update**. Clearing this option also clears **Run > Create**
Run Step > Delete	User group can delete step information. **Note:** Clearing this option also clears **Run > Delete**.
Run Step > Update	User group can modify step information. This permission level enables you to specify the fields that the selected user group can modify. **Note:** Clearing this option also clears **Run > Update** and **Run Step > Create**.
Test Instance > Create	User group can add test instances to a test set. **Note:** Selecting this option also selects **Test Instance > Update**. Clearing this option also clears **Test Set > Create**.
Test Instance > Delete	User group can remove test instances from a test set. To ensure that only the owner of the test set can remove it, select **By Owner Only**. **Note:** Clearing this option also clears **Test Set > Delete**.

Entity > Permission Level	Description
Test Instance > Update	User group can modify test instances in a test set. This permission level enables you to specify the fields that the selected user group can modify. To ensure that only the owner of the test set can modify it, select **By Owner Only**. **Note:** Clearing this option also clears **Run > Create**, **Test Instance > Create**, **Test Set > Update**, **Test Set > Reset**, and **Run > Delete**.
Test Set > Copy	User group can copy test sets to folders in the test sets tree.
Test Set > Create	User group can add test sets. **Note:** Selecting this option also selects **Test Set > Update** and **Test Instance > Create**.
Test Set > Delete	User group can delete test sets. **Note:** Selecting this option also selects **Test Instance > Delete**. Clearing this option also clears **Test Set Folder > Delete**.
Test Set > Move	User group can move test sets to different folders in the test sets tree. **Note:** Selecting this option also selects **Test Set Folder**, which is a field under **Test Set > Update**. Also, this option is cleared when you clear **Test Set Folder**.
Test Set > Reset	User group can clear all runs in a test set. **Note:** Selecting this option also selects **Run > Delete** and **Test Instance > Update**.
Test Set > Update	User group can modify test sets. This permission level enables you to specify the fields that the selected user group can modify. **Note:** Selecting this option also selects **Test Instance > Update**. Clearing this option also clears **Test Set > Create**.
Test Set Folder > Copy	User group can copy folders in the test sets tree.

Entity > Permission Level	Description
Test Set Folder > Create	User group can add folders to the test sets tree. **Note:** Selecting this option also selects **Test Set Folder > Update**.
Test Set Folder > Delete	User group can delete folders in the test sets tree. **Note:** Selecting this option also selects **Test Set > Delete**.
Test Set Folder > Move	User group can move folders in the test sets tree.
Test Set Folder > Update	User group can modify folders in the test sets tree. This permission level enables you to specify the fields that the selected user group can modify. **Note:** Clearing this option also clears **Test Set Folder > Create**.

Test Plan Permission Levels

The Test Plan tab displays the entities available in the Test Plan module and their corresponding permission levels. The entities are listed below in alphabetical order.

Entity > Permission Level	Description
Design Step > Create	User group can add design steps in the Design Steps tab. **Note:** Selecting this option also selects **Design Step > Update**. Clearing this option also clears **Test > Create**.
Design Step > Delete	User group can delete design steps from the Design Steps tab. To ensure that only the owner can delete design steps from the Design Steps tab, select **By Owner Only**. **Note:** Clearing this option also clears **Test > Delete**.

Entity > Permission Level	Description
Design Step > Update	User group can modify design steps in the Design Steps tab. This permission level enables you to specify the fields that the selected user group can modify. To ensure that only the owner can modify the fields, select **By Owner Only**. **Note:** Clearing this option also clears **Design Step > Create** and **Test > Update**.
Test > Create	User group can add tests to the test plan tree. **Note:** Selecting this option also selects **Design Step > Create**, **Test Configuration > Create**, and **Test > Update**, as well as **Component > Create** and **Component Instance > Create** in the **Business Process Testing** tab. This option is cleared when **Component > Create** or **Component Instance > Create** in the **Business Process Testing** tab is cleared.
Test > Delete	User group can delete tests from the test plan tree. To ensure that only the owner of the test can delete it, select **By Owner Only**. **Note:** Selecting this option also selects **Design Step > Delete** and **Test Configuration > Delete**, as well as **Component > Delete** and **Component Instance > Delete** in the **Business Process Testing** tab. Clearing this option also clears **Test Folder > Delete**. This option is cleared when **Component > Delete** or **Component Instance > Delete** in the **Business Process Testing** tab is cleared.
Test > Generate Script	User group can convert the test steps of a manual test, displayed in the Design Steps tab, into an automated test. To ensure that only the owner of the test can convert the manual test, select **By Owner Only**. **Note:** Selecting this option also selects **Type**, which is a field under **Test > Update**. Also, this option is cleared when you clear **Type**.

Entity > Permission Level	Description
Test > Update	User group can modify tests in the test plan tree. This permission level enables you to specify the fields that the selected user group can modify. To ensure that only the owner of the test can modify it, select **By Owner Only**. **Note:** Selecting this option also selects **Design Step > Update**, **Test Configuration > Update**, and **Test Parameter > Create, Update and Delete**, as well as **Component > Update** and **Component Instance > Update** in the **Business Process Testing** tab. Clearing this option also clears **Test > Create**. This option is cleared when **Component > Update** or **Component Instance > Update** in the **Business Process Testing** tab is cleared.
Test Configuration > Create	User group can add test configurations to the test. **Note:** Selecting this option also selects **Test Configuration > Update**. Clearing this option also clears **Test > Create**.
Test Configuration > Delete	User group can delete test configurations from the test. To ensure that only the owner of the test configuration can delete it, select **By Owner Only**. **Note:** Clearing this option also clears **Test > Delete**.
Test Configuration > Update	User group can modify test configurations. This permission level enables you to specify the fields that the selected user group can modify. To ensure that only the owner of the test configuration can modify it, select **By Owner Only**. **Note:** Clearing this option also clears **Test Configuration > Create** and **Test > Update**.
Test Folder > Copy	User group can copy folders in the test plan tree.
Test Folder > Create	User group can add folders to the test plan tree. **Note:** Selecting this option also selects **Test Folder > Update**.
Test Folder > Delete	User group can delete folders from the test plan tree. **Note:** Selecting this option also selects **Test > Delete**.

Entity > Permission Level	Description
Test Folder > Move	User group can move folders in the test plan tree. **Note:** Selecting this option also selects **Parent Folder**, which is a field under **Test Folder > Update**. Also, this option is cleared when you clear **Parent Folder**.
Test Folder > Update	User group can modify folders in the test plan tree. This permission level enables you to specify the fields that the selected user group can modify. **Note:** Clearing this option also clears **Test Folder > Create**.
Test Parameter > Create, Update and Delete	User group can add, modify and delete test parameters. **Note:** Clearing this option also clears **Test > Update**.

Timeslots Permission Levels

The Timeslots tab displays the Timeslot entity available in the Timeslots module and its corresponding permission levels.

Entity > Permission Level	Description
Timeslot > Create and Update	User group can create and update timeslots. **Note:** Clearing this option also clears **Run > Start Test Set Run** and **Run > Start Build Verification Suite Run** in the **Test Lab** tab, and **Timeslot > Abort and Free**. This option is selected when **Run > Start Test Set Run** or **Run > Start Build Verification Suite Run** in the **Test Lab** tab is selected.
Timeslot > Delete	User group can delete timeslots.

Entity > Permission Level	Description
Timeslot > Abort and free running timeslot	User group can stop timeslots that are currently running. **Note:** Selecting this option also selects **Run > Stop Run** and **Run > Stop Build Verification Suite \ Test Set Run** in the **Test Lab** tab, and **Timeslot > Create and Update**. This option is selected when **Run > Stop Run** or **Run > Stop Build Verification Suite \ Test Set Run** in the **Test Lab** tab is selected.
Timeslot > Provision and Deploy	User group can select an AUT environment and select provision and deploy options.

Customizing Module Access for User Groups

For each ALM project, you can control the modules that each user group can access. you can better utilize your ALM licenses by preventing users from accessing unnecessary modules. For example, if a user group uses ALM only to add defects to a project, you can limit the group's access to the Defects module only.

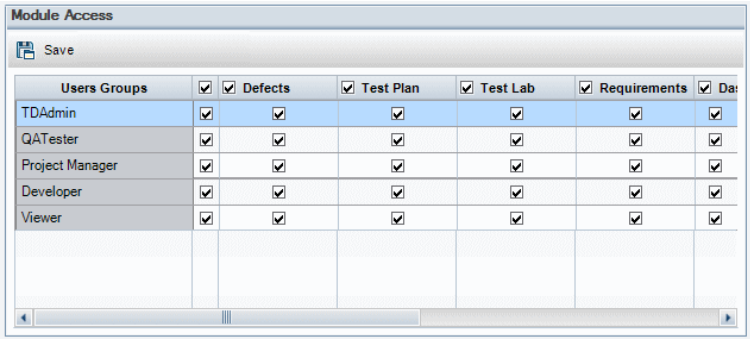
You can specify module access for the following modules: Defects, Test Plan, Test Lab, Requirements, Dashboard, Business Components, Releases, Business Process Models, and Libraries.

If access to the Business Components module is not enabled for a user group, those users can still view existing business process tests in read-only mode.

Note: Performance Center: Module access customization is not supported in Lab Management.

To customize module access for user groups:

1. In the Project Customization window, in the left pane, click **Module Access**. The Module Access page opens.



Checkmarks indicate the modules that a user group can access.

2. To select or clear a cell in the table, select the cell's checkbox.
3. To select or clear all modules for a user group, select the checkbox in the column to the right of the user group name.
4. To select or clear all user groups for a module, select the checkbox to the left of the module name (in the same cell).
5. To select or clear all user groups for all modules, select the checkbox in the column to the right of the heading "User Groups."
6. Click **Save** to save your changes.

Chapter 16: Customizing ALM Projects

As an HP Application Lifecycle Management (ALM) project administrator, you can customize a project to meet the specific needs of your organization. For example, you can add or customize fields, customize requirement types, and create categories and lists that reflect the needs of your project.

This chapter includes:

About Customizing ALM Projects 299

Customizing Project Entities 299

Customizing Project Requirement Types 306

Customizing Project Lists 310

About Customizing ALM Projects

Before you begin a project, you can customize your project to reflect your unique requirements. As a project progresses, you can further adjust the project to meet its changing needs.

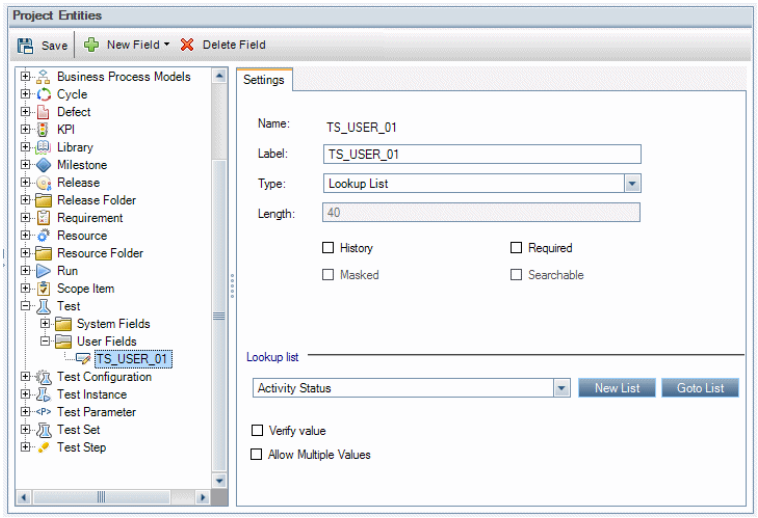
ALM contains system fields in which you enter information about ALM entities. You can modify the behavior of these fields by restricting users to selecting values only from associated lists, by making entry into certain fields mandatory, and by preserving a history of values entered in the field. In addition, you can include data unique to your project by creating user-defined fields. You can associate these fields with ALM system and user-defined lists.

For example, if you are running tests on several builds of an application, you can add a **Detected in Build** field to the Add Defect dialog box. You can then create a selection list containing the values **Build1**, **Build2**, and **Build3**, and associate the list with the **Detected in Build** field.

In the Requirements module, you can also assign each requirement to a requirement type. A requirement type defines which fields are available and which fields are required for a requirement of that type. This enables you to make available for a requirement only the fields relevant to the type to which it is assigned.

Customizing Project Entities

Using the Project Entities page, you can customize your ALM project to suit your environment.



Each ALM project is divided into project entities. **Entities** contain data entered by users for a specific application management process. The data is stored in tables.

Project Entities Tree

The project entities tree displays the available project entities.

Each entity contains system fields and user-defined fields:

- **System fields.** These are ALM default fields. You cannot add or delete system fields, you can only modify them.

- **User fields.** These are fields that you can define and include in an ALM project to customize for your specific project needs. You can add, modify, and delete user-defined fields.

For details, see:

- "Adding User-Defined Fields" on page 302
- "Modifying System and User-Defined Fields" on page 303
- "Deleting User-Defined Fields" on page 304

For detailed information on ALM entities and fields, refer to the *HP ALM Project Database Reference*.

The Settings Tab

The **Settings** tab displays the field properties. The following properties are available:

Properties	Description
Name	Indicates the field name used in the ALM database table. Read-only.
Label	Indicates the field name as it is displayed in ALM. You can type a new name or use the default name. The label cannot include the following characters: **( ) @ \ / : * ? " ` < > | + = ; , %**
Type	Specifies the type of data that the user can enter in the field. It includes the following types: • **Number.** Enables integer entry only. • **Float.** Enables the entry of a floating point/real number. • **String.** Enables the entry of any character string. • **Date.** Enables the selection of a date. • **Lookup List.** Displays the Lookup List area and enables the selection from a drop-down list. • **User List.** Enables the selection of a user name from your ALM users list. • **Memo.** Enables the entry of blocks of data. Note that by default, you can add up to five memo fields to each ALM entity. In Site Administration's Site Configuration tab, you can edit the "EXTENDED_MEMO_FIELDS" on page 179 parameter, which extends the number of memo fields you can add.
Length	Indicates the field size. (Available only when the **String** type is selected.) **Note:** The maximum field length is 255 characters.
History	Preserves a log of values entered in the selected field.

Properties	Description
Required	Indicates that a user must enter a value for the field. **Note:** If you set a field as required in a project that already contains data, users do not need to enter a value for the field when they modify an existing record if the field is already empty.
Masked	Indicates the input data mask for the field. (Available only when the **String** type is selected.) For more information, see "Defining Input Masks" on page 304.
Versioned	**Version Control:** Indicates whether to save field values for each version of entities/fields under version control. This checkbox is selected, by default, for most fields. You can turn off versioning for user-defined fields and some system fields. **Note:** In some cases, the checkbox is disabled—for example, if the field's parent entity is not version-controlled. **Tip:** Turning off versioning for relevant fields enables workflow scripts and OTA scripts to run without requiring entities to be checked in.
Searchable	Indicates a searchable field. (Available only when the **Text Search** option is enabled in the DB Servers tab in Site Administration. For more information, see "Defining Searchable Fields" on page 165.)
Lookup List	Includes a list of predefined lists. (Available only when the **Lookup List** type is selected.) To associate a field with a predefined list, select a list from the **Lookup List** box. To view or modify the selected list, click the **Goto List** button.
New List	Creates a new list. (Available only when the **Lookup List** type is selected.) To associate a field with a new list, click the **New List** button. The Project Lists dialog box opens. For more information on customizing a list, see "Customizing Project Lists" on page 310.
Goto List	Displays a predefined list. (Available only when the **Lookup List** type is selected.) To open a predefined list, select a list from the **Lookup List** box. Click the **Goto List** button. The Project Lists dialog box opens. For more information on customizing a list, see "Customizing Project Lists" on page 310.
Verify Value	Limits the user to select a value only from the items that are listed in the list box. (Available when **Lookup List** or **User List** is selected.)

Properties	Description
Allow Multiple Values	For user-defined fields, this option allows the user to select more than one value in any field that is associated with a predefined lookup list. (Available only when the **Lookup List** type is selected.) For more information, see "ALLOW_MULTIPLE_VALUES" on page 172. For example, if you create a **Language** user field in the Defect entity and enable the **Allow Multiple Values** option, a user can select `English`, `French`, and `German` language values at the same time when entering this field's value. **Note:** • This option is not available in the TEST STEP entity. • If you group a data grid or summary graph by a field containing multiple values, the information in each value is grouped as its entire value. This value is the category for grouping. For example, a value with `English` and `French` is grouped once as `English;French`, and not as part of separate `English` and `French` categories. For more information on customizing a list, see "Customizing Project Lists" on page 310.

Cross Project Customization

ALM Editions: Cross project customization is not available for Quality Center Enterprise Edition.

If you are working with cross project customization, consider the following:

- **Working with a Template Project:** If you are working with a template project, you use the **Project Entities (Shared)** link to customize system fields and create user-defined fields. System and user-defined fields in a template project are applied to the linked projects when you apply the template customization. For more information on applying template customization, see "Applying Template Customization to Linked Projects" on page 337.
- **Working with a Linked Project:** If you are working with a project that is linked to a template project, you cannot modify system or user-defined fields that are defined by the template project.

This section includes:

Adding User-Defined Fields 302
Modifying System and User-Defined Fields 303
Deleting User-Defined Fields 304
Defining Input Masks 304

Adding User-Defined Fields

You can customize an ALM project by adding up to 99 user-defined fields to each ALM entity.

Cross Project Customization: The template project and linked projects can each contain up to 99 user-defined fields for each ALM entity. **ALM Editions**: Cross project customization is not available for Quality Center Enterprise Edition.

To add a user-defined field:

1. In the Project Customization window, in the left pane, click **Project Entities**. The Project Entities page opens.
2. Under **Project Entities**, expand an entity.
3. Click the **User Fields** folder.
4. To add a user-defined field, you can:
   - Click the **New Field** button to add a number, string, date, or list type field.
   - Click the **New Field** arrow and choose **New Memo Field** to add a memo field. You can add up to 5 memo fields to each ALM entity.

     Note: In Site Administration's Site Configuration tab, you can extend the number of memo fields you can add, by editing the "EXTENDED_MEMO_FIELDS" on page 179 parameter.
5. In the **Settings** tab, set properties for the field. For more information, see "Customizing Project Entities" on page 299.
6. Click **Save** to save your changes to the Project Entities page.

Modifying System and User-Defined Fields

You can modify the properties of system and user-defined fields in your ALM project.

Note: You cannot modify the **Type** or **Length** properties for system fields. In addition, for system fields of type **Lookup List**, you cannot modify which list is associated with the field, or allow multiple values to be selected. For more information, see "Customizing Project Entities" on page 299.

Cross Project Customization: If you are working with a project that is linked to a template project, you cannot modify system or user-defined fields that are defined by the template project. Fields defined by a template project are displayed with a template icon .

ALM Editions: Cross project customization is not available for Quality Center Enterprise Edition.

To modify a system or user-defined field:

1. In the Project Customization window, in the left pane, click **Project Entities**. The Project Entities page opens.
2. Under **Project Entities**, expand an entity.
3. Expand the **System Fields** folder or the **User Fields** folder.
4. Click the field that you want to customize. The settings for that field appear in the Settings tab.

5. Modify the properties for the selected field. For more information, see "Customizing Project Entities" on page 299.
6. Click **Save** to save your changes to the Project Entities page.

Deleting User-Defined Fields

You can delete user-defined fields from your ALM project.

Cross Project Customization - Working with a Linked Project: If you are working with a project that is linked to a template project, you cannot delete user-defined fields defined by the template project.

ALM Editions: Cross project customization is not available for Quality Center Enterprise Edition.

To delete a user-defined field:

1. In the Project Customization window, in the left pane, click **Project Entities**. The Project Entities page opens.
2. Under **Project Entities**, expand an entity.
3. Expand the **User Fields** folder.
4. Click the field that you want to delete and click the **Delete Field** button.
5. Click **OK** to confirm. The field is removed from the **User Fields** folder.
6. Click **Save** to save your changes to the Project Entities page.

Defining Input Masks

The input mask option is used to prompt users for data input using a mask pattern. If the user attempts to enter a character that conflicts with the input mask, an error occurs. For example, to prompt the user to enter a phone number, you can define the following input mask:

```
!\(000\)000-0000
```

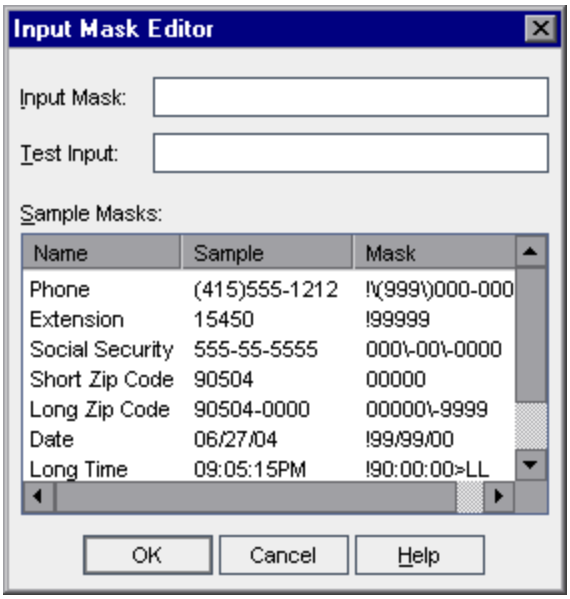
This input mask limits the user to numeric characters only. It is displayed in an edit box as follows:

```
(___) ___ - ____
```

Note: You can define input masks for string type fields only.

To define an input mask:

1. In the Settings tab, select **Masked**. For more information, see "Customizing Project Entities" on page 299.
2. Under **Masked Edit Attributes**, click the **Define** button. The Input Mask Editor dialog box opens.



3. In the **Input Mask** box, type an input mask or select a predefined mask.

   You can use the following characters when defining input masks:

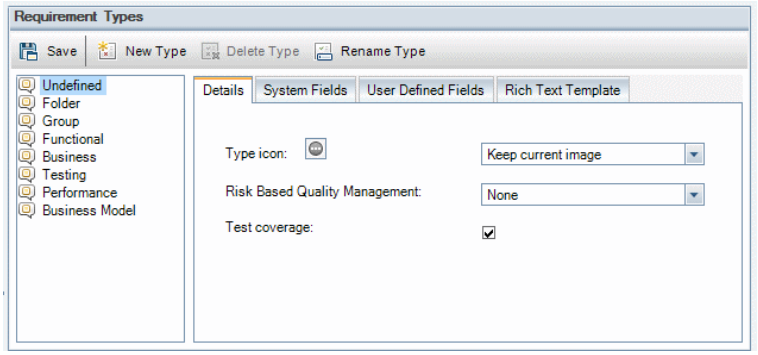
Mask Character	Description
!	A space for a leading or trailing blank.
#	A digit.
.	A decimal.
:	A time separator.
/	A date separator.
****	Treats the next character in the mask string as a literal. For example, you can include the (,), #, &, A, and ? characters in the mask.
>	Converts all the characters that follow to uppercase.
<	Converts all the characters that follow to lowercase.
A	An alphanumeric character (entry required). For example: a – z, A – Z, or 0 – 9.
a	An alphanumeric character (entry optional). For example: a – z, A – Z, or 0 – 9.
C	A character (entry required). Valid values are ANSI characters in the following ranges: 32-126 and 128-255.
c	A character (entry optional). Valid values are ANSI characters in the following ranges: 32-126 and 128-255.

Mask Character	Description
L	An alphabetic character or space (entry required). For example: a – z or A – Z.
l	An alphabetic character or space (entry optional). For example: a – z or A – Z.
0	A digit (entry required). For example: 0 – 9.
9	A digit (entry optional). For example: 0 – 9.
_	Inserts spaces. When the user types characters in the field box, the cursor skips the _ character.

4. In the **Test Input** box, you can test the input mask.
5. Click **OK** to close the Input Mask Editor dialog box.
6. Click **Save** to save your changes to the Project Entities page.

Customizing Project Requirement Types

Using the Requirement Types page, you can create requirement types for your project, and customize their properties.



You can assign each requirement in the Requirements module to a requirement type. A **requirement type** defines which fields are optional and which user-defined fields are available. This enables you to create user-defined fields that are only available for requirements of a specific type.

For example, you might create a requirement type `Security Requirement` for requirements connected to security. You could then create a user-defined field `Security Hazards` containing a list of possible security hazards a requirement might cover. This field is not relevant for requirements of types other than `Security Requirement`, so you would not make it available for any type except for `Security Requirement`.

Each requirement type has an associated icon, which is displayed next to the requirement in the Requirements module tree views, enabling you to easily identify to which type a requirement belongs. For each requirement type you can determine whether test coverage and risk-based quality management are available.

In addition, for each requirement type, you can define a rich text template to be used when adding or editing rich text within the Requirements module.

Cross Project Customization

If you are working with cross project customization, consider the following: (**ALM Editions**: Cross project customization is not available for Quality Center Enterprise Edition.)

- **Working with a Template Project:** If you are working with a template project, you use the **Requirement Types (Shared)** link to create and customize requirement types. Requirement types defined in a template project are applied to the linked projects when you apply the template customization.
- **Working with a Linked Project:** If you are working with a project that is linked to a template project, you cannot modify the default requirement types or requirement types that are defined by the template project.

This section includes:

Creating Requirement Types 307
Customizing Requirement Types 307
Renaming Requirement Types 309
Deleting Requirement Types 310

Creating Requirement Types

You can create a requirement type. ALM provides the default requirement types **Undefined, Folder**, **Group**, **Functional**, **Business**, **Testing**, and **Business Model**. For more information on these types, refer to the *HP Application Lifecycle Management User Guide*.

To create a requirement type:

1. In the Project Customization window, in the left pane, click **Requirement Types**. The Requirement Types page opens.
2. Click the **New Type** button. The New Type dialog box opens.
3. In the **Name** box, type a name for the type.
4. In the **Create As** list, assign properties of an existing requirement type.

   Tip: Choose an existing requirement type that has similar properties to the new type you want to create. This minimizes the level of customization you need to do.
5. Click **OK**. The New Type dialog box closes and the new type is added to the Types list.
6. Click **Save** to save your changes to the Requirement Types page.

Customizing Requirement Types

You can customize a requirement type by changing its icon, by setting options for test coverage and risk analysis, and by defining which fields are available and which are required for each requirement type. You can also define a rich text template for each requirement type.

Cross Project Customization - Working with a Linked Project: If you are working with a project that is linked to a template project, you cannot modify the default requirement types or requirement types defined by the template project. For requirement types defined by the template project, you can choose which user-defined fields defined in the project are available to requirements of the type.

ALM Editions: Cross project customization is not available for Quality Center Enterprise Edition.

To customize a requirement type:

1. In the Project Customization window, in the left pane, click **Requirement Types**. The Requirement Types page opens.
2. Select a requirement type.
3. In the **Details** tab, you can set the following:
   - **Type Icon.** To change the icon that is displayed in Requirements module tree views, next to the requirements of the selected type, select an icon from the **Type Icon** list. The icon is changed accordingly.

     Note: You cannot change the icon for the default requirement types **Folder** and **Group**.
   - **Risk-Based Quality Management.** To set risk-based quality management for the requirements of the selected type, select one of the following options from the **Risk-Based Quality Management** box:
     - **Perform Analysis**.
     - **Perform Assessment**.
     - **None.** Select this one if you do not want to enable risk-based quality management for requirements of the selected type.

     For more information on risk-based quality management, see the *HP Application Lifecycle Management User Guide*.
   - **Test Coverage.** To enable or disable test coverage of the requirements of the selected type, select or clear the **Test Coverage** check box.

     Note: You cannot clear the **Testing Coverage** check box for a requirement type if there are requirements of that type which already have test coverage. To clear, you must either delete the requirements of that type with test coverage, remove the test coverage from those requirements, or change their type.
4. In the **System Fields** tab, you can make a system field a required field for the type. Select the check box in the **Required** column for the field. Note that all system fields are automatically included in every type, and that some system fields cannot be set as optional.

   Tip: To set all system fields as required fields at one time, select the check box next to the Required column heading.
5. In the **User Defined Fields** tab, you can choose which user fields are available for

requirements of the type, and which of these fields are required:

- To make a user-defined field available for the type, select the check box in the **In Type** column for the field. For more information on user-defined fields, see "Customizing Project Entities" on page 299.
- To make a user-defined field that is available for the type a required field, select the check box in the **Required** column for the field.

6. In the **Rich Text Template** tab, use the HTML editor to define a page layout to be displayed as the initial view in the Rich Text tab of the Requirements module. For more information on the Rich Text tab, see the *HP Application Lifecycle Management User Guide*.

   Note:
   - You can only define one template per requirement type.
   - You cannot include graphics in the template. You can only add graphics from within the Rich Text tab of the Requirements module.
   - The template is applied automatically to all newly created requirements of the type.
   - The template can also by applied manually to existing requirements from within the Rich Text tab. Applying the template overwrites any existing content.

7. Click **Save** to save your changes to the Requirement Types page.

Renaming Requirement Types

You can rename a requirement type. You cannot rename the following default requirement type: **Folder**.

Cross Project Customization - Working with a Linked Project: If you are working with a project that is linked to a template project, you cannot rename the default requirement types or requirement types defined by the template project.

ALM Editions: Cross project customization is not available for **Quality Center Enterprise Edition**.

To rename a requirement type:

1. In the Project Customization window, in the left pane, click **Requirement Types**. The Requirement Types page opens.
2. Select a requirement type.
3. Click the **Rename Type** button. The Rename Type dialog box opens.
4. Type a new name for the requirement type.
5. Click **OK** to close the Rename Type dialog box. The requirement type name is updated.
6. Click **Save** to save your changes to the Requirement Types page.

Deleting Requirement Types

You can delete a requirement type. You cannot delete a type if there are requirements of the type in your project. To delete a type, you must first delete all requirements of the type, or change their types. You cannot delete the following default requirement types: **Folder, Group,** and **Undefined**.

Cross Project Customization - Working with a Linked Project: If you are working with a project that is linked to a template project, you cannot delete the default requirement types or requirement types defined by the template project.

ALM Editions:Cross project customization is not available for **Quality Center Enterprise Edition**.

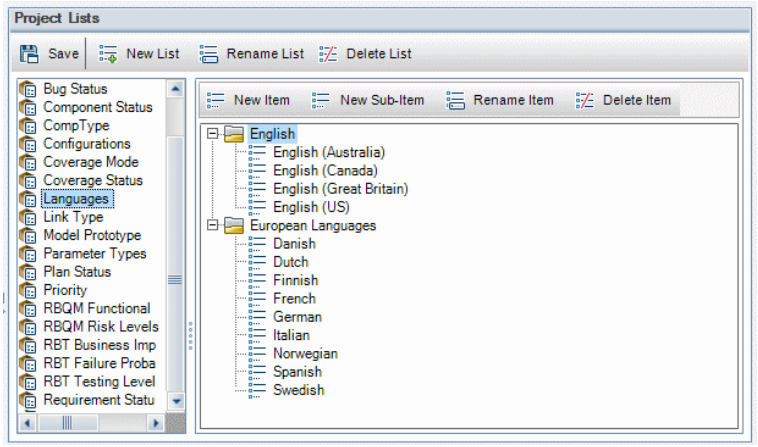
To delete a requirement type:

1. In the Project Customization window, in the left pane, click **Requirement Types**. The Requirement Types page opens.
2. Select a requirement type.
3. Click the **Delete Type** button.
4. Click **OK** to confirm. The requirement type is deleted.
5. Click **Save** to save your changes to the Requirement Types page.

Customizing Project Lists

Using the Project Lists page, you can create, rename, and delete user-defined lists. For details, see:

- "Creating Lists" on next page
- "Renaming Lists, Items, or Sub-Items" on next page
- "Deleting Lists, Items, or Sub-Items" on page 312



A list contains items, which are values that you can enter in a field. For example, the selection list for the Languages user-defined field may contain the items `English` and `European Languages`.

The list can also contain several levels of sub-items. For example, the item `English` can contain a sublist with the sub-items `English (Australia)`, `English (Canada)`, `English (Great Britain)`, and `English (US)`.

You can allow the user to select more than one value from a list by enabling the **Allow Multiple Values** option for the relevant field in the Project Entities page. For more information, see the Field Settings table in "Customizing Project Entities" on page 299.

Note: To associate a list with a field, see "Customizing Project Entities" on page 299.

Cross Project Customization

If you are working with cross project customization, consider the following: (**ALM Editions**: Cross project customization is not available for **Quality Center Enterprise Edition**.)

- **Working with a Template Project:** If you are working with a template project, you use the **Project Lists (Shared)** link to create and customize user-defined lists. Project lists defined in a template project are applied to the linked projects when you apply the template customization.
- **Working with a Linked Project:** If you are working with a project that is linked to a template project, you cannot modify, rename, or delete user-defined lists that are defined by the template project.

Creating Lists

You can create a list to be assigned to one or more fields.

To create a list:

1. In the Project Customization window, in the left pane, click **Project Lists**. The Project Lists page opens.
2. Click the **New List** button. The New List dialog box opens.
3. Type a name for the new list (maximum length 255 characters) and click **OK**.
4. To add an item to the new list or to an existing list, select the list name and click the **New Item** button. The New Item dialog box opens. Type a name for the item and click **OK**.

   Note: You should not use a semi-colon (";") as part of any list item if the list is to be used in a multiple value field. For more information on multiple value fields, see the Field Settings table in "Customizing Project Entities" on page 299.
5. To create a sub-item, select an item and click the **New Sub-Item** button. The New Sub-Item dialog box opens. Type a name for the sub-item and click **OK**.
6. Click **Save** to save your changes to the Project Lists page.

Renaming Lists, Items, or Sub-Items

You can rename user-defined lists, and system and user-defined items or sub-items.

Note: You cannot change some system list items. For example, the `Y` and `N` in the **YesNo** list. For more information on system items that cannot be changed, see HP Software Self-solve knowledge base article KM206085 (http://h20230.www2.hp.com/selfsolve/document/KM206085).

Cross Project Customization - Working with a Linked Project: If you are working with a project that is linked to a template project, you cannot rename lists, items, or sub-items defined by the template project.

ALM Editions: Cross project customization is not available for **Quality Center Enterprise Edition**.

To rename a list:

1. In the Project Customization window, in the left pane, click **Project Lists**. The Project Lists page opens.
2. Select a list.
3. Click the **Rename List** button. The Rename List dialog box opens.
4. Type a new name for the list.
5. Click **OK** to close the Rename List dialog box.
6. Click **Save** to save your changes to the Project Lists page.

To rename an item or sub-item:

1. In the Project Customization window, in the left pane, click **Project Lists**. The Project Lists page opens.
2. Select a list.
3. Select an item.
4. Click the **Rename Item** button. The Rename List Item dialog box opens.
5. Type a new name for the item. Click **OK**.
6. Click **Save** to save your changes to the Project Lists page.

Deleting Lists, Items, or Sub-Items

You can delete user-defined lists and system and user-defined items or sub-items.

Note:

- You cannot delete a user-defined list that is being used as a lookup list for a field.
- You cannot delete some system list items. For example, the `Y` and `N` in the **YesNo** list. For more information on system items that cannot be deleted, see HP Software Self-solve knowledge base article KM206085 (http://h20230.www2.hp.com/selfsolve/document/KM206085).

Cross Project Customization - Working with a Linked Project: If you are working with a project that is linked to a template project, you cannot delete lists, items, or sub-items defined by the template project.

ALM Editions: Cross project customization is not available for **Quality Center Enterprise Edition**.

To delete a list:

1. In the Project Customization window, in the left pane, click **Project Lists**. The Project Lists page opens.
2. Select a user-defined list name.
3. Click the **Delete List** button.
4. Click **Yes** to confirm.
5. Click **Save** to save your changes to the Project Lists page.

To delete an item or sub-item:

1. In the Project Customization window, in the left pane, click **Project Lists**. The Project Lists page opens.
2. In the left pane, select a list name.
3. In the right pane, select a list item.
4. Click the **Delete Item** button.
5. Click **Yes** to confirm.
6. Click **Save** to save your changes to the Project Lists page.

Chapter 17: Configuring Automail

As an HP Application Lifecycle Management (ALM) project administrator, you can routinely inform your personnel about defect repair activity. You determine the conditions for sending defect messages to each recipient by defining a mailing configuration.

This chapter includes:

About Setting Automail 315

Designating Automail Fields and Conditions 315

Customizing the Subject of Defect Mail 317

About Setting Automail

ALM enables you to automatically notify users through email each time changes are made to specified defect fields. Configuring mail for an ALM project involves the following steps:

- Click the **Automail** link in the Project Customization window to define the defect fields and specify the users and conditions. See "Designating Automail Fields and Conditions" below.
- In Site Administration's **Site Projects** tab, enable the mail configuration for a project by selecting the **Send mail automatically** check box. You must select this check box for your mail configuration to work. For more information, see "Updating Project Details" on page 73.
- In Site Administration's **Site Configuration** tab, you can edit the "MAIL_INTERVAL" on page 169 parameter, which defines the time interval for sending defect emails in all projects. You can also set parameters to define the format and character set of mail, and whether attachments or history are included in the mail. For more information, see "ATTACH_MAX_SIZE" on page 167, "AUTO_MAIL_WITH_ATTACHMENT" on page 167, and "AUTO_MAIL_WITH_HISTORY" on page 168.
- You can customize the subject line of defect email for all projects or for a specific project. For more information, see "Customizing the Subject of Defect Mail" on page 317.
- In Site Administration's **Site Users** tab, make sure you have specified the email addresses of the users who should receive defect messages. For more information, see "Updating User Details" on page 138.

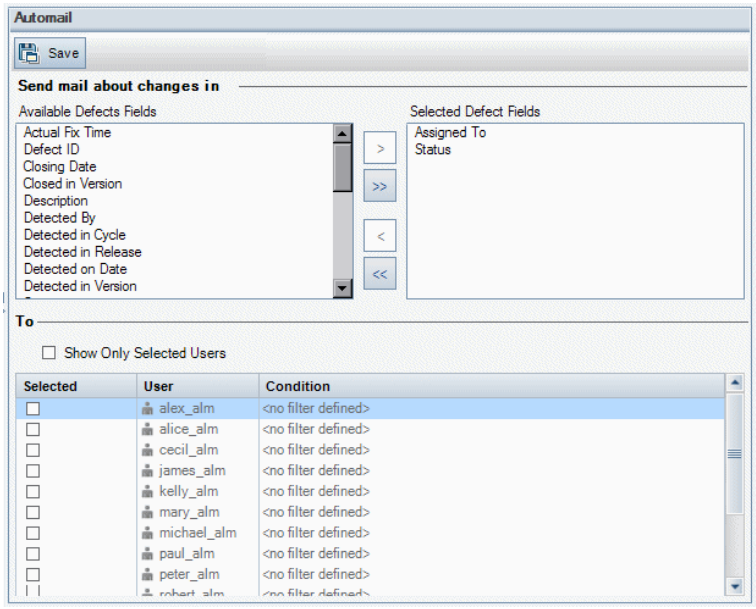
Designating Automail Fields and Conditions

When you designate a field as a mail field, any changes made to that field cause ALM to send an email message in the next time interval. For example, suppose you designate `Status` as a mail field and then update the `Status` field for a particular defect. In the next time interval, the details of the defect, including the updated status information, are sent to designated users.

Mail conditions determine when various users receive defect messages. For each user, you can define separate mail conditions. For example, you can specify that a user receives messages only for defects assigned an urgent priority.

To designate Automail fields and conditions:

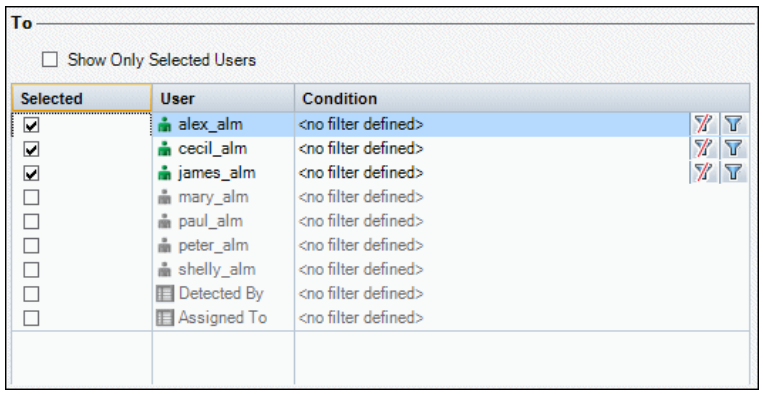
1. In the Project Customization window, in the left pane, click **Automail**. The Automail page opens.



Available Defect Fields contains the names of the fields that appear in the Defects Grid. **Selected Defect Fields** contains the names of fields currently assigned as mail fields.

2. Choose one or more fields and click the arrow buttons (> and <) to move the fields from one list to the other. Click the double arrow buttons (>> and <<) to move all the fields from one list to the other.

3. Select the users who should receive the email by selecting the checkbox next to each user name in the **To** area in the lower half of the window.



Tip: To see only the relevant selected users, select the **Show Only Selected Users** checkbox.

4. Click the **Filter** button to define a filter under which the selected user receives mail. If you define multiple filters, the selected user only receives mail if all of the conditions are met. For more information on filtering, refer to the *HP Application Lifecycle Management User Guide*.

5. Click **Save** to save your changes.

Customizing the Subject of Defect Mail

You can customize the subject line of defect email sent automatically to users, for all projects or for a specific project. For example, you can define a subject line such as the following:

```
Defect # 4321 has been created or updated - Buttons on print dialog
are not aligned
```

The line can contain the values of ALM fields. To include a field value from the defect that is being sent, prefix the field name with a question mark (?). Field names must be upper-case. For example:

```
Defect # ?BG_BUG_ID has been created or updated - ?BG_SUMMARY
```

You can customize the subject line for all your projects by adding the "AUTO_MAIL_SUBJECT_FORMAT" on page 173 parameter in the **Site Configuration** tab.

To customize the subject of defect mail for a specific project:

1. In Site Administration, click the **Site Projects** tab.
2. In the Projects list, double-click the project for which you want to customize the email subject line.
3. Select the **DATACONST** table.
4. In the SQL pane, type an SQL INSERT statement to insert a row into the table with the following values:
   - In the **DC_CONST_NAME** column, insert the parameter name **AUTO_MAIL_SUBJECT_FORMAT**.
   - In the **DC_VALUE** column, insert the strings and the names of fields to be placed in the subject line.

   For example, type the following SQL statement into the SQL pane:

   ```
   insert into dataconst values ('AUTO_MAIL_SUBJECT_FORMAT',
   'DEFAULT.TESTPROJ - Defect # ?BG_BUG_ID has been created or updated
   -
   ?BG_SUMMARY')
   ```

   The subject line you define is specific to the project, so you can include the project name in the line.

   For more information on modifying project tables, see "Querying Project Tables" on page 89.
5. Click the **Execute SQL** button. The row is added to the **DATACONST** table to set the email subject.

Chapter 18: Customizing Risk-Based Quality Management

This chapter describes how to customize the criteria and the constant values used in risk-based quality management.

This chapter includes:

About Customizing Risk-Based Quality Management 319

Customizing Risk-Based Quality Management Criteria 319

Customizing Risk Calculations 324

Customizing Risk-Based Quality Management Constants 324

About Customizing Risk-Based Quality Management

You use risk-based quality management to determine at which Testing Level to test each requirement in the Requirements module. ALM then calculates the total estimated Testing Time for an analysis requirement based on the Testing Levels of its child assessment requirements. You compare this with the resources you have available to test the analysis requirement, and if necessary, make adjustments to the Testing Levels for the requirement and its children. This enables you to plan the testing strategy for your requirements. For more information on risk-based quality management, see the *HP Application Lifecycle Management User Guide*.

The Testing Level is determined by the requirement's Risk and Functional Complexity. Risk is comprised of the Business Criticality and Failure Probability. You determine values for these factors by assigning values to a series of criteria associated with each factor. Each criterion has a number of possible values. You can customize these criteria and values and how ALM uses them to determine the Business Criticality, Failure Probability, and Functional Complexity. For more information, see "Customizing Risk-Based Quality Management Criteria" below.

You can customize how Risk is calculated from the Business Criticality and Failure Probability. For more information, see "Customizing Risk Calculations" on page 324.

You can also customize which Testing Time is associated by default with each Testing Level and Functional Complexity. In addition, you can customize how ALM determines the Testing Level with which to test a requirement based on the requirement's Risk and Functional Complexity. For more information, see "Customizing Risk-Based Quality Management Constants" on page 324.

In addition, you can customize for each requirement type whether requirements of the type enable risk assessment, risk analysis, or do not enable risk-based quality management. For more information, see "Customizing Requirement Types" on page 307.

Customizing Risk-Based Quality Management Criteria

You can customize the criteria used to determine the Risk and Functional Complexity of a requirement, the possible values for each criterion and the weight assigned to each value. You can then define how the total of these weights determines the Risk and Functional Complexity Categories.

Note: If you already calculated the Risk or Functional Complexity for a requirement, modifying these criteria does not automatically recalculate the requirement's Risk or Functional Complexity Category. To recalculate, you must reassess the requirement and change at least one of its criterion values.

This section includes:

Customizing Criteria and Values 320
Customizing Weight Boundaries 321

Customizing Criteria and Values

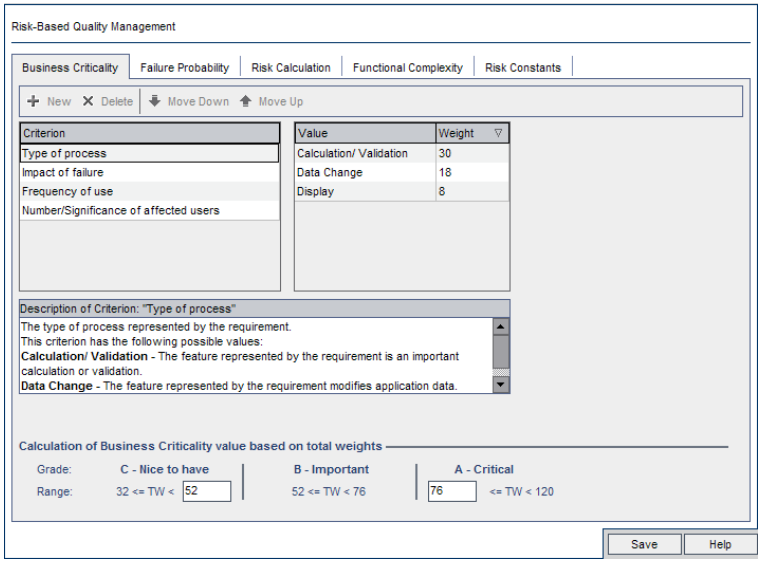
You can customize the criteria, criterion values, and weights used by ALM to determine the Risk and Functional Complexity Categories for each requirement. Risk is comprised of the Business Criticality and Failure Probability.

ALM provides a default set of criteria when you create a new project. You can delete these criteria if you do not want to use them.

To customize criteria and values:

1. In the Project Customization window, in the left pane, click **Risk-Based Quality Management**. The Risk-Based Quality Management page opens.
2. Click one of the following tabs:
   - To customize criteria for determining Business Criticality, click the **Business Criticality** tab.
   - To customize criteria for determining Failure Probability, click the **Failure Probability** tab.
   - To customize criteria for determining Functional Complexity, click the **Functional Complexity** tab.

   The selected tab displays the relevant criteria.



3. To add a new criterion, select the **Criterion** list and click the **New** button. A new row is added to the end of the **Criterion** list. Type a name for the criterion in the new row.
4. To add a description for a criterion, select the criterion from the **Criterion** list and type the description in the **Description of Criterion** box. This description is displayed in the Risk tab of the Requirements module when a user assesses Business Criticality, Failure Probability, and Functional Complexity. By providing a full description of each criterion, including an explanation of its available values, you help the user decide which value to assign to each criterion for a requirement.

5. To add a value for a criterion, select the criterion from the **Criterion** list and select the **Value** list. Click the **New** button. A new row is added to the **Value** list. Type a name for the value in the new row.

   Note: Each value for a criterion must be unique.

6. To assign a weight to a criterion value, select the criterion from the **Criterion** list and then select the value from the **Value** list. In the **Weight** column for the value, type the weight to assign to the value.

   When ALM calculates the Business Criticality, Failure Probability, or Functional Complexity of a requirement, it checks the values assigned to each criterion and calculates the sum of the corresponding weights for each value. This sum determines the Business Criticality, Failure Probability, or Functional Complexity. For more information, see "Customizing Weight Boundaries" below.

7. You can delete a criterion or a value for a criterion:

   - To delete a criterion, select the criterion from the **Criterion** list and click the **Delete** button. The criterion is deleted.
   - To delete a criterion's value, select the criterion from the **Criterion** list and the value from the **Value** list. Click the **Delete** button. The value is deleted.

   Note: Business Criticality, Failure Probability, and Functional Complexity must each have at least one associated criterion defined. In addition, each criterion must have at least one possible value.

8. To change the order in which criteria are displayed in the **Criterion** list, select a criterion and click the **Move Up** or **Move Down** buttons. Note that values for a criterion are automatically ordered by their weights.

9. Click **Save** to save your changes to the Risk-Based Quality Management page.

Customizing Weight Boundaries

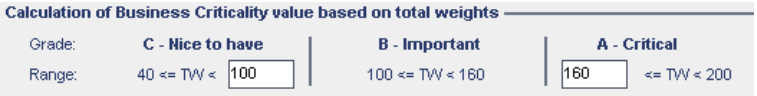
You can customize how ALM uses the values assigned to risk-based quality management criteria to determine the Business Criticality, Failure Probability, and Functional Complexity for a requirement.

Customizing Weight Boundaries for Business Criticality

For each requirement, ALM calculates the total of the weights (**TW**) of the values assigned to each of the Business Criticality criteria. ALM then uses this total to categorize the Business Criticality of the requirement as either **C- Nice to Have**, **B - Important**, or **A - Critical**. ALM automatically calculates the highest and lowest possible total weight and uses these to define the upper boundary of the **Critical** category and the lower boundary of the **Nice to Have** category. You define the boundaries between the **Nice to Have** and **Important** categories, and between the **Important** and **Critical** categories.

For example, suppose there are two criteria for Business Criticality, each with three possible values whose weights are `20`, `60`, and `100`. The minimum total weight is therefore `40` (if both criteria are assigned the value with weight `20`) and the maximum total weight is `200` (if both criteria are

assigned the value with weight `100`). ALM automatically calculates these totals and uses them to determine the lower and upper boundaries for the categories. You determine the boundaries between the categories by typing `100` in the **Nice to Have** box and `160` in the **Critical** box.



In this example, ALM determines the Business Criticality for a requirement as follows:

- If the sum of the weights of each of the criteria for a requirement is less than or equal to `100`, the requirement will have `Nice to Have` Business Criticality. This could happen, for example, if the criteria have values with weights `20` and `60`, so the total weight is `80`.
- If the sum is greater than `100` but less than `160`, the requirement will have `Important` Business Criticality. This could happen, for example, if the criteria have values with weights `60` and `60`, so the total weight is `120`.
- If the sum is greater than or equal to `160`, the requirement will have `Critical` Business Criticality. This could happen, for example, if the criteria have values with weights `100` and `60`, so the total weight is `160`.

To customize weight boundaries for Business Criticality:

1. In the Project Customization window, in the left pane, click **Risk-Based Quality Management**. The Risk-Based Quality Management page opens.
2. Click the **Business Criticality** tab. The Business Criticality tab displays criteria used to determine Business Criticality.
3. Under **Calculation of Business Criticality value based on total weights**, define the boundaries between different Business Criticality values. To define these boundaries, type the relevant values in the **Nice to Have** and **Critical** boxes.
4. Click **Save** to save your changes to the Risk-Based Quality Management page.

Customizing Weight Boundaries for Failure Probability

For each requirement, ALM calculates the total of the weights (**TW**) of the values assigned to each of the Failure Probability criteria. ALM then uses this total to categorize the Probability of the requirement as either **3 - Low**, **2 - Medium**, or **1- High**. ALM automatically calculates the highest and lowest possible total weight and uses these to define the upper boundary of the **High** category and the lower boundary of the **Low** category. You define the boundaries between the **Low** and **Medium** categories, and between the **Medium** and **High** categories.

For example, suppose there are two criteria for Failure Probability, each with three possible values whose weights are `20`, `60`, and `100`. The minimum total weight is therefore `40` (if both criteria are assigned the value with weight `20`) and the maximum total weight is `200` (if both criteria are assigned the value with weight `100`). ALM automatically calculates these totals and uses them to determine the lower and upper boundaries for the categories. You determine the boundaries between the categories by typing `100` in the **Low** box and `160` in the **High** box.

In this example, ALM determines the Failure Probability for a requirement as follows:

- If the sum of the weights of each of the criteria for a requirement is less than or equal to `100`, the requirement will have `Low` Failure Probability. This could happen, for example, if the criteria have values with weights `20` and `60` so the total weight is `80`.
- If the sum is greater than `100` but less than `160`, the requirement will have `Medium` Failure Probability. This could happen, for example, if the criteria have values with weights `60` and `60` so the total weight is `120`.
- If the sum is greater than or equal to `160`, the requirement will have `High` Failure Probability. This could happen, for example, if the criteria have values with weights `100` and `60` so the total weight is `160`.

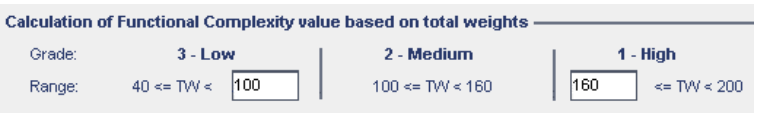
To customize weight boundaries for Failure Probability:

1. In the Project Customization window, in the left pane, click **Risk-Based Quality Management**. The Risk-Based Quality Management page opens.
2. Click the **Failure Probability** tab. The Failure Probability tab displays criteria used to determine Failure Probability.
3. Under **Calculation of Failure Probability value based on total weights**, define the boundaries between different Failure Probability values. To define these boundaries, type the relevant values in the **Low** and **High** boxes.
4. Click **Save** to save your changes to the Risk-Based Quality Management page.

Customizing Weight Boundaries for Functional Complexity

For each requirement, ALM calculates the total of the weights (**TW**) of the values assigned to each of the Functional Complexity criteria. ALM then uses this total to categorize the Functional Complexity of the requirement as either **3 - Low**, **2 - Medium**, or **1- High**. ALM automatically calculates the highest and lowest possible total weight and uses these to define the upper boundary of the **High** category and the lower boundary of the **Low** category. You define the boundaries between the **Low** and **Medium** categories, and between the **Medium** and **High** categories.

For example, suppose there are two criteria for Functional Complexity, each with three possible values whose weights are `20`, `60`, and `100`. The minimum total weight is therefore `40` (if both criteria are assigned the value with weight `20`) and the maximum total weight is `200` (if both criteria are assigned the value with weight `100`). ALM automatically calculates these totals and uses them to determine the lower and upper boundaries for the categories. You determine the boundaries between the categories by typing `100` in the **Low** box and `160` in the **High** box.



In this example, ALM determines the Functional Complexity for a requirement as follows:

- If the sum of the weights of each of the criteria for a requirement is less than or equal to `100`, the requirement will have `Low` Functional Complexity. This could happen, for example, if the criteria have values with weights `20` and `60` so the total weight is `80`.
- If the sum is greater than `100` but less than `160`, the requirement will have `Medium` Functional Complexity. This could happen, for example, if the criteria have values with weights `60` and `60` so the total weight is `120`.

- If the sum is greater than or equal to `160`, the requirement will have `High` Functional Complexity. This could happen, for example, if the criteria have values with weights `100` and `60` so the total weight is `160`.

To customize weight boundaries for Functional Complexity:

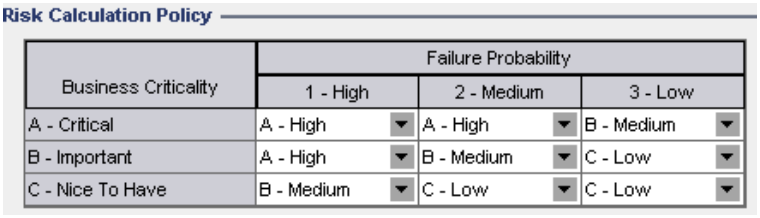
1. In the Project Customization window, in the left pane, click **Risk-Based Quality Management**. The Risk-Based Quality Management page opens.
2. Click the **Functional Complexity** tab. The Functional Complexity tab displays criteria used to determine Functional Complexity.
3. Under **Calculation of Functional Complexity value based on total weights**, define the boundaries between different Functional Complexity values. To define these boundaries, type the relevant values in the **Low** and **High** boxes.
4. Click **Save** to save your changes to the Risk-Based Quality Management page.

Customizing Risk Calculations

You can define how ALM calculates the Risk value of an assessment requirement.

To customize risk calculations:

1. In the Project Customization window, in the left pane, click **Risk-Based Quality Management**. The Risk-Based Quality Management page opens.
2. Click the **Risk Calculation** tab.



3. In the **Risk Calculation Policy** grid, you can define the Risk policy for testing a requirement.

   To define Risk calculations based on Business Criticality and Failure Probability, click the arrow next to the cell in the grid corresponding to a particular Business Criticality and Failure Probability value. Select a value. The available values are **A - High**, **B - Medium**, and **C - Low**.

Customizing Risk-Based Quality Management Constants

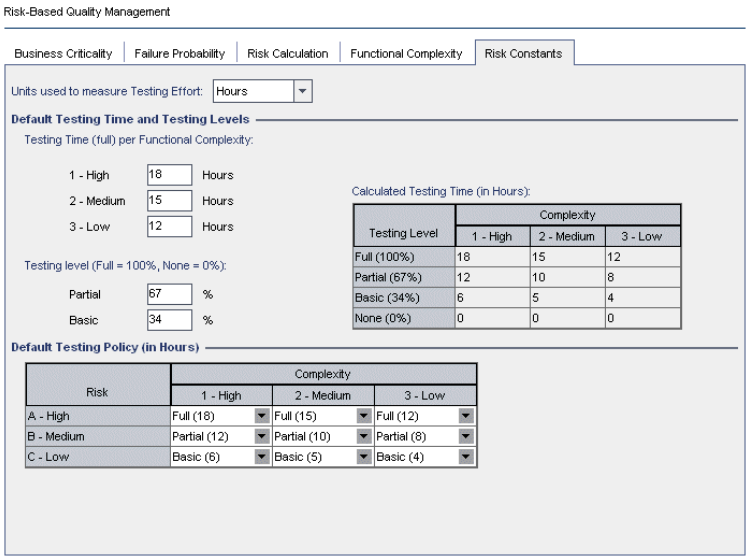
You can define the default estimated Testing Times required to test a requirement with each Functional Complexity value at each Testing Level. You can also define the default Testing Level used with each Risk and Functional Complexity Category. If the user does not enter different values for a requirement in the Requirements module, ALM uses these default values when calculating the estimated Testing Time for the requirement during the risk analysis.

Note: Modifying these criteria does not automatically affect the results of existing risk

analyses. To update the results of a risk analysis, you must perform the analysis again.

To customize risk-based quality management constants:

1. In the Project Customization window, in the left pane, click **Risk-Based Quality Management**. The Risk-Based Quality Management page opens.
2. Click the **Risk Constants** tab. The Risk tab displays constants used by default when calculating the Testing Time and Testing Level for a requirement.



3. In the **Units used to measure Testing Effort** box, select the unit of measurement that you want ALM to display when measuring Testing Time. The available units are **Hours**, **Days**, **Weeks**, and **Months**.

   Note: If you change the units of measurement for a project, the Testing Time values are not updated automatically. For example, if a requirement has Testing Time `48 hours` and you change the units of measurement from **Hours** to **Days**, the requirement has Testing Time `48 days` and not `2 days`.

4. Under **Testing Time (full) per Functional Complexity**, for each Functional Complexity value, type the estimated time required to fully test a requirement with the Functional Complexity value. The Calculated Testing Time grid is updated to reflect these changes.
5. Under **Testing level**, in the **Partial** and **Basic** boxes, type the default Testing Time required for partial testing and basic testing of a requirement. This should be expressed as a percentage of the time required for full testing. The Calculated Testing Time grid is updated to reflect these changes.
6. In the **Default Testing Policy** grid, you can define the default Testing Level for testing a requirement.

   To define the default Testing Levels, click the arrow next to the cell in the grid corresponding to a particular Risk and Functional Complexity value. Select a Testing Level from the available

Testing Levels. The available Testing Levels are **Full**, **Partial**, **Basic**, and **None**. Next to each Testing Level, you can see the estimated time needed to test a requirement at that level, based on the default Testing Times and Testing Levels you defined.

7. Click **Save** to save your changes to the Risk-Based Quality Management page.

Chapter 19: Activating Alert Rules

As an HP Application Lifecycle Management (ALM) project administrator, you can activate alert rules for your project. This instructs ALM to create alerts and send email to notify those responsible when changes occur in your project that may impact the application management process.

This chapter includes:

About Activating Alert Rules 329

Setting Alert Rules 330

About Activating Alert Rules

You can keep track of your requirements, tests, and defects as you perform your application management process. When an entity changes, you can instruct ALM to notify those responsible for any associated entities.

The alert rules you can activate are based on the following associations you can create in ALM:

- You can associate a test in the test plan tree with a requirement. This is performed by creating **requirements coverage** in the Test Plan module, or by creating **tests coverage** in the Requirements module.
- You can link a test with a defect. This is performed by adding a defect during a manual test run.
- You can create **traceability links** between requirements in the Requirements module.

After you have established associations in your project, you can then track changes using these associations. When an entity in your project changes, ALM alerts you of any associated entities that may be impacted by the change.

Version Control: ALM alerts associated entities only when a new version is checked in. The alert indicates that the version status has changed to **Checked In**. You can then compare the new version with the previous version. For more information on comparing versions, refer to the *HP Application Lifecycle Management User Guide*.

Notification involves two steps. ALM flags the associated entity, which can be seen by all users, and then sends an email to the user responsible for the entity.

There are four alert rules you can activate:

Rule	Change Made	Entities Flagged	User Notified
1	Requirement has any change, excluding changes in the **Direct Cover Status** field and the risk-based quality management fields.	Tests covering the requirement.	Test designer. Note that only the test designer can delete the alert.
2	Defect status changes to "Fixed".	Test instances associated with the defect.	Responsible tester for the test instance.
3	Test run status changes to "Passed".	Defects linked to the test instance.	User assigned to the defect.
4	Requirement is deleted or has any change, excluding changes in the **Direct Cover Status** field and the risk-based quality management fields.	The requirement's child requirements and traced to requirements.	Author of the requirement.

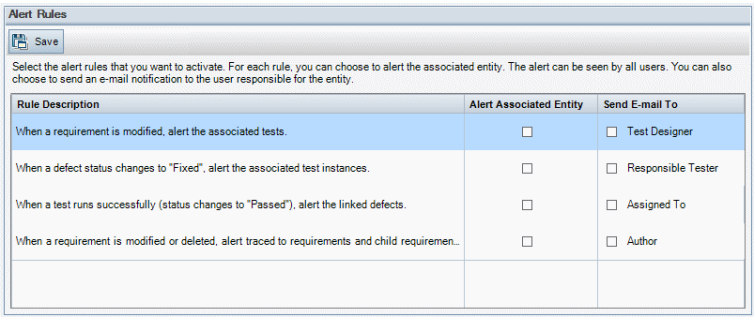
For more information on alerts, refer to the *HP Application Lifecycle Management User Guide*.

Setting Alert Rules

You can activate four alert rules. For each rule, you can choose to alert the associated entity. The alert can be seen by all users. You can also choose to send an email notification to the user responsible for the entity.

To set alert rules:

1. In the Project Customization window, in the left pane, click **Alert Rules**. The Alert Rules page opens.



2. Select **Alert Associated Entity** to activate a rule. This instructs ALM to flag the entity when the associated entity changes.
3. Select **Send E-mail To** to instruct ALM to send a notification email to the specified user when the associated entity changes.
4. Click **Save** to save your changes.

Chapter 20: Cross Project Customization

As an HP Application Lifecycle Management (ALM) template administrator, you use cross project customization to apply customization from a template project to one or more ALM projects. Cross project customization enables you to standardize policies and procedures across projects in your organization.

ALM Editions: Cross project customization is not available for Quality Center Enterprise Edition.

This section includes:

About Cross Project Customization 332

Cross Project Customization Overview 333

Updating Linked Projects 334

Updating Linked Project Details 334

Verifying Cross Project Customization 336

Applying Template Customization to Linked Projects 337

The Cross Project Customization Report 339

Updating Linked Template Details 340

About Cross Project Customization

Cross project customization enables you to use a **template project** to define and maintain a common set of project customizations for multiple projects.

Note: Cross project customization cannot be implemented between a Unicode template project and ASCII projects. Also, when a template project is converted to Unicode, any linked ASCII projects cannot receive customization changes from the template.

A **template administrator** is any user that is assigned project administrator permissions for a template project. As template administrator, you can customize a template project to meet the needs of your organization.

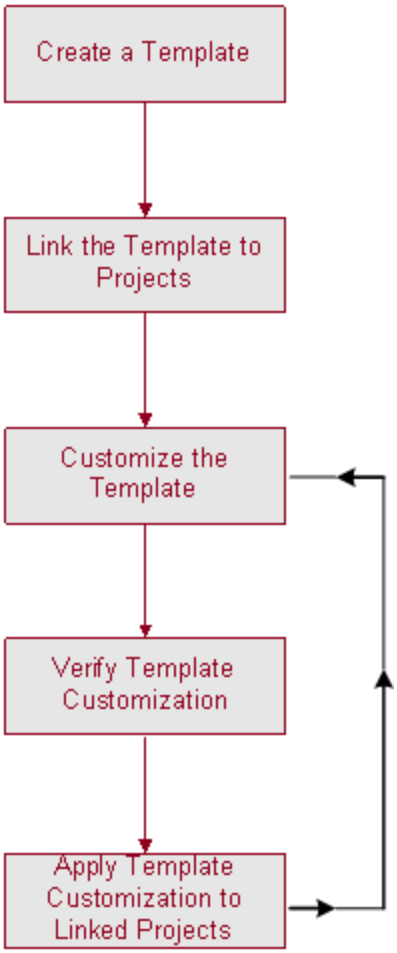
You link a template project to one or more ALM projects. This enables you to apply template customization to the **linked projects**. As the needs of your organization change over time, you can update the customization in your template project, and then reapply the template customization to the linked projects.

A template project can be used to create additional projects or templates. When the Site Administrator creates a project or template based on a template project, the template customization is copied to the newly created project or template.

Product Feature Movie: To view a movie that demonstrates how to work with cross project customization, choose **Help > Movies** in the ALM main window.

Cross Project Customization Overview

Implementing cross project customization involves the following steps:



- **Creating a Template Project.** The site administrator creates template projects and assigns template administrators in Site Administration. For more information, see "Creating Template Projects" on page 54.
- **Linking the Template to Projects.** The site administrator selects projects to link to a template in Site Administration. For more information, see "Linking a Template to Projects" on page 72.
- **Customizing the Template Project.** As template administrator, you customize a template project to meet the policy needs of your organization. The template customizations that are applied to linked projects include: user groups and permissions, module access, project entities, project requirement types, project lists, PPT, report templates, and workflow.
- **Verifying Cross Project Customization.** Before applying template customization to linked

projects, you must verify that ALM can successfully apply customization from the template to the projects. For more information, see "Verifying Cross Project Customization" on page 336.

- **Applying Customization to Linked Projects.** After defining or updating customization in the template, you apply the customization to the linked projects. For more information, see "Applying Template Customization to Linked Projects" on page 337.

Cross Project Customization Examples

The following examples demonstrate how you can use cross project customization:

- **Set a standard for working with defects**.

  The QA manager wants to limit how testers can modify defects. For example, you want to allow testers to modify the status of defects to `Fixed`, but not to `Closed`, to allow the QA manager to review the defect before closing it. You can create a customized user group for testers in your template and set transition rules for the group. After applying the template customization to the linked projects, all testers can be assigned to this group.

- **Enable consistent reporting by managers**.

  Managers in all divisions of your organization are required to report on a standard set of measures, such as defect status or priority, or coverage status of requirements. As template administrator, you can customize project lists and fields, and set required fields in a template. Applying the template customization to the linked projects provides users with a common set of fields and values for consistent reporting.

- **Create unique policies for separate sectors of an organization**.

  Your organization has acquired a new company. The new company has a standard policy for working with defects, which is different from how your organization currently operates. Both sectors want to maintain their current policies. You can customize a template for each sector of your organization, and link each to the relevant projects for its sector.

Updating Linked Projects

You manage template customization updates to linked projects in Project Customization.

This section includes:

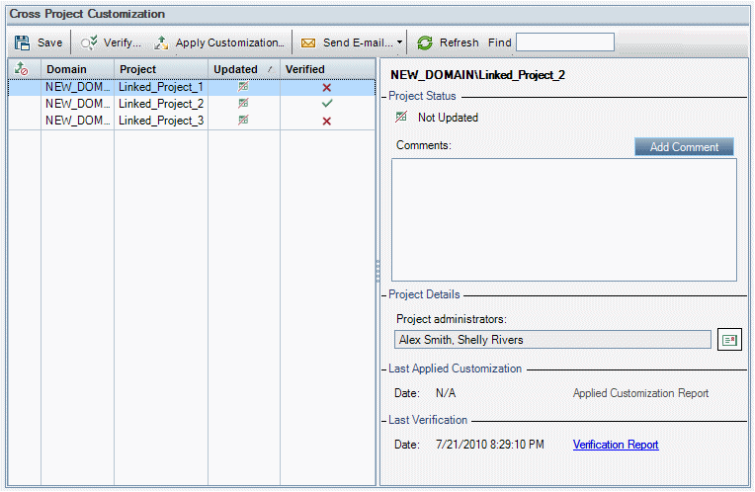
Updating Linked Project Details 334
Verifying Cross Project Customization 336
Applying Template Customization to Linked Projects 337

Updating Linked Project Details

You update linked project details in Project Customization.

To update linked project details:

1. Log into ALM using the template project.
2. In the Project Customization window, in the left pane, click **Cross Project Customization**. The Cross Project Customization - Linked Projects page opens.



3. In the Linked Projects grid, view the projects linked to the template project. The Linked Projects grid contains the following information for each project:

Column	Description
	Indicates a request by the project administrator not to apply template customization changes to the project.
Domain	The domain of the linked project.
Project	The name of the linked project.
Updated	Indicates if the linked project is updated with current template customization. The current status can be one of the following: ▪ **Not Updated** (default) ▪ **Updated**
Verified	Indicates if template customization has been verified and can be successfully applied to the linked project. By default, the status is **Not Verified**. The current status can be one of the following: ▪ **Not Verified** (default) ▪ **Verified with Warnings** ▪ **Verified**

Click on a column heading to change the sort order of the projects in the grid.

4. To refresh the data in the Linked Projects grid, click the **Refresh** button .

5. View additional details about a selected project on the right side of the Linked Projects page. For example, under **Project Status**, view the status of the project. If the project administrator has selected the **Request Suspension of Apply Customization** option in the linked project, **Requested suspension of Apply Customization** is displayed. The template administrator can choose to exclude the project from template customization updates.
6. In the **Comments** box, view comments added by the project administrator. Click **Add Comment** to add a comment to the project. The project administrator can view and add comments when viewing the project details.
7. Under **Project Details**, the names of project administrators are displayed. Click the **Send E-mail** button to send mail to project or template administrators.
8. Under **Last Applied Customization**, view the date that template customization was last applied to the linked project. Click the **Applied Customization Report** link to view details. For more information, see "The Cross Project Customization Report" on page 339.
9. Under **Last Verification**, view the date of the last verification. Click the **Verification Report** link to view details of the last verification. For more information, see "The Cross Project Customization Report" on page 339.

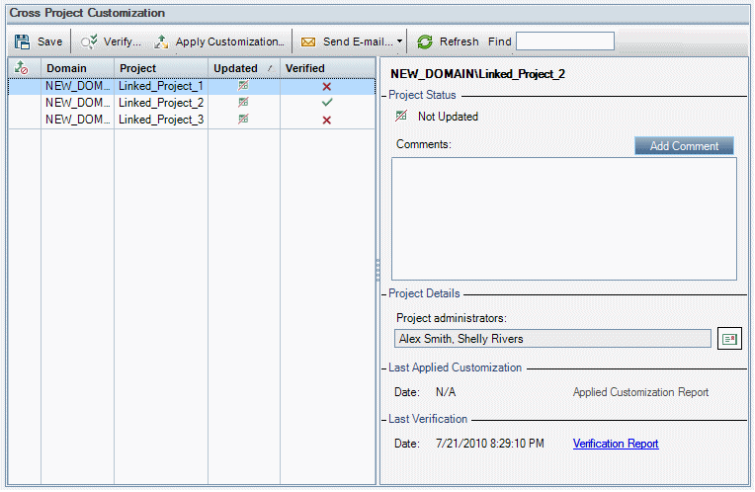
Verifying Cross Project Customization

Before you can apply template customization to linked projects, you must verify the customization. The verification process checks that ALM can successfully apply template customization to the linked projects. Verification must complete successfully before ALM can apply template customization to a linked project.

Note: For verification to complete successfully, the appropriate extensions must be enabled in linked projects. If an extension is enabled for a template project, the extension must also be enabled for the template's linked projects. Linked projects can have additional extensions enabled. For more information on enabling extensions, see "Enabling Extensions for a Project" on page 79.

To verify cross project customization:

1. In the Project Customization window, in the left pane, click **Cross Project Customization**. The Cross Project Customization - Linked Projects page opens.



2. Select a project from the grid, or press the CTRL key and select multiple projects. To display only those projects that are activated, select **Show only active projects**.
3. Click **Verify**. The Verification dialog box opens and displays progress.
4. To stop verification before it completes, click **Stop**. ALM completes the project that it is currently verifying and then stops. The remaining projects are not verified.
5. Click **Details** to view additional information during or after verification. When verification completes, click the **Report** link to view detailed results for a project.
6. When verification completes, click **Close** to close the Template Verification dialog box. The verification status for the projects is updated in the Linked Projects grid.
7. Under **Last Verification**, click the **Verification Report** link to view details of the verification. For more information, see "The Cross Project Customization Report" on page 339.

Applying Template Customization to Linked Projects

You can apply template customization to projects linked to the template. This applies the following customization: groups and permissions, module access, project entities, project requirement types, project lists, PPT, report templates, and workflow. When you apply template customization, the applied customization is set to read-only in the linked projects and cannot be edited.

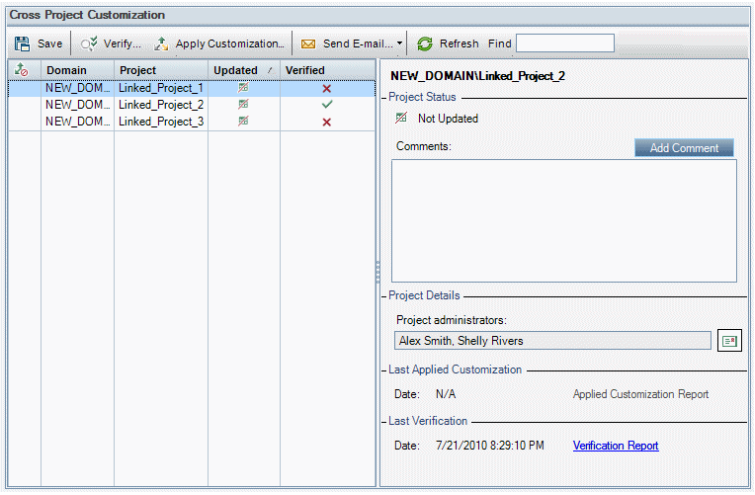
Note:

- When a new project is created based on a template and linked to the template, you must apply template customization in order to set the template customization as read-only in the linked project. For more information on creating a project based on a template, see "Creating Projects" on page 37.
- The option for setting a report template as default is not applied to linked projects and can be set by a project administrator in the linked project.

Before you can apply template customization, you must verify the customization. For more information, see "Verifying Cross Project Customization" on previous page. Verification must complete successfully before ALM can apply template customization to a linked project.

To apply template customization to linked projects:

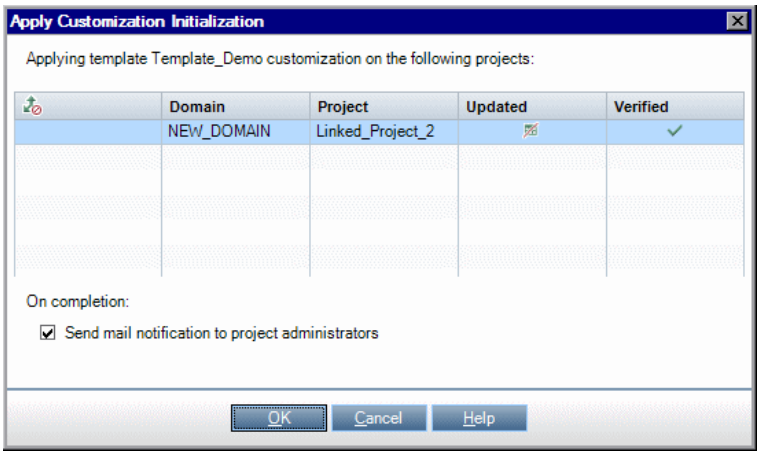
1. In the Project Customization window, in the left pane, click **Cross Project Customization**. The Cross Project Customization - Linked Projects page opens.



2. Select a project from the grid, or press the CTRL key and select multiple projects. To display only those projects that are activated, select **Show only active projects**.

3. Click **Apply Customization**. If the project administrator of any of the selected projects has requested not to apply template customization changes, a warning is displayed. Click **OK** to apply template customization to all selected projects.

   The Apply Customization Initialization dialog box opens.



4. Select **Send mail notification to project administrators** to instruct ALM to notify project administrators after the process completes.

5. Click **OK**. The Apply Customization dialog box opens and displays progress.

6. To cancel the process for projects that ALM did not yet update, click **Stop**. ALM completes the update to the current project and cancels the update to the remaining projects.

7. When the process completes, click **Close** to close the Apply Customization dialog box.

8. Under **Last Applied Customization**, click the **Applied Customization Report** link to view details of applied template customization. For more information, see "The Cross Project Customization Report" on next page.

The Cross Project Customization Report

The Cross Project Customization Report provides detailed results of the verification process, or of template customization applied to linked projects. For more information on verification, see "Verifying Cross Project Customization" on page 336. For more information on applying template customization, see "Applying Template Customization to Linked Projects" on page 337.

There are two types of Cross Project Customization Reports:

- **Verification Report.** Provides results of verification for the linked project.
- **Applied Customization Report.** Provides results of template customization applied to the linked project.

The Applied Customization Report includes the following sections:

- **Report Details.** Contains details about the type of report, the template, the linked project, the number of changes verified or applied to the linked project, and results.
- **Report by Customization Category.** A listing of all changes verified or applied to the linked project. This section lists changes by customization category including user groups, project entities, project lists, requirement types, and workflow scripts.

Report results are classified into several categories as follows:

Result category	Verification Report	Applied Customization Report
Successful	The change can be successfully applied to the linked project.	The change was successfully applied to the linked project.
Warning	The change can be applied to the linked project, but may result in data loss. For example: • reducing the length of a string type field • deleting a user-defined field • defining a field to be searchable and the **Text Search** option is not available in the linked project • disabling test coverage for a requirement type while there are tests covering requirements of the type	The change was applied to the linked project but may have caused data loss.

Result category	Verification Report	Applied Customization Report
Failure	The change cannot be applied to the linked project. For example: • Changing a field type from **Memo** type to **Number**, **String**, or **Date** type or the reverse. • Naming a new field, or renaming an existing field, with a field name that already exists in the linked project. • The report is too large for the default limit query size. For details, see "MAX_QUERY_LENGTH" on page 186.	An error occurred during the Apply Customization process. The change was not successfully applied to the linked project.

Tips:

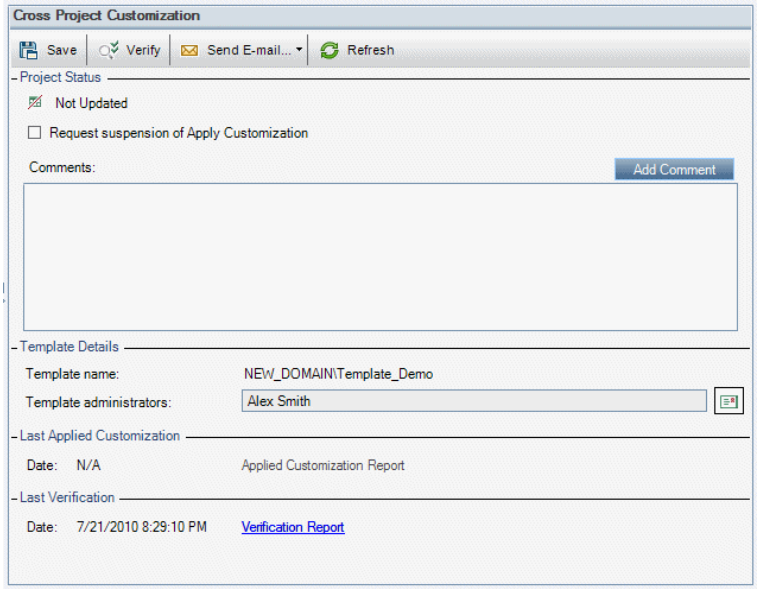
- To easily locate warnings or errors in the Cross Project Customization Report, click the **Find** button to open the browser's search tool, and search for the word `warning` or `error`.
- To preserve formatting when mailing a report to another user, save the file as an HTML archived web page by saving it with the **.mht** file extension.
- If transition rules are set for user groups in the linked or template projects, each rule is listed in the **Transition Rule** column in the format `<from state>,<to state>`. For example, `New,Open New,Rejected Open,Fixed Open,Rejected` indicates that the user group can change the field value from **New** to **Open** or **Rejected**, and from **Open** to **Fixed** or **Rejected**.

Updating Linked Template Details

If you are working with a project that is linked to a template project, you can view details about the project and about the template project from the Linked Template page. You can view details about template customization applied to the project, send email to the template administrator, check for conflicts between customization in the project and customization in the template project, or make a request to block template customization updates.

To update linked template details:

1. Log into ALM using a project that is linked to a template.
2. In the Project Customization window, in the left pane, click **Cross Project Customization**. The Cross Project Customization - Linked Template page opens.



3. Under **Project Status**, view the following status information:
   - **Updated.** Customization in the template project is applied to the project.
   - **Not Updated.** Changes made to customization in the template project have not been applied to the project.
4. Select **Request suspension of Apply Customization** to make a request to block template customization updates. The request is displayed in the template project, and the template administrator can choose to exclude the project from template customization updates.
5. Click **Add Comment** to add a comment to the project. The comment is displayed in the Comments box. The Comments box also displays comments made by the template administrator. The template administrator can add and view comments when reviewing linked project details in the template project.
6. Under **Template Details**, view the name of the template project linked to the project, and the names of the template administrators. To send email to the project or template administrators, click the **Send E-mail** button .
7. Under **Last Applied Customization**, view the date of the last time template customization was applied to the project. Click the **Applied Customization Report** link to view details. For more information, see "The Cross Project Customization Report" on page 339.
8. Under **Last Verification**, view the date of the last time customization was verified for the project. Click the **Verification Report** link to view details. For more information, see "The Cross Project Customization Report" on page 339.
9. Click the **Verify** button to verify cross project customization for the project. For example, if you make changes to project customization, you can run template verification to check for conflicts between customization in the project and customization in the template project.
10. Click **Details** to view additional information during or after verification. When verification completes, you can click the **Report** link to view detailed results for the project.
11. When verification completes, click **Close** to close the Verification dialog box.

Chapter 21: Customizing Project Planning and Tracking KPIs

This chapter describes how to customize KPIs for project planning and tracking (PPT).

ALM Editions: The Project Planning and Tracking link in Project Customization is available for ALM Edition only.

This chapter includes:

About Customizing PPT KPIs 343

Project Planning and Tracking Page 343

Project Planning and Tracking - General Tab 344

Configure Transitions Dialog Box 346

Project Planning and Tracking - KPI Analysis Tab 347

About Customizing PPT KPIs

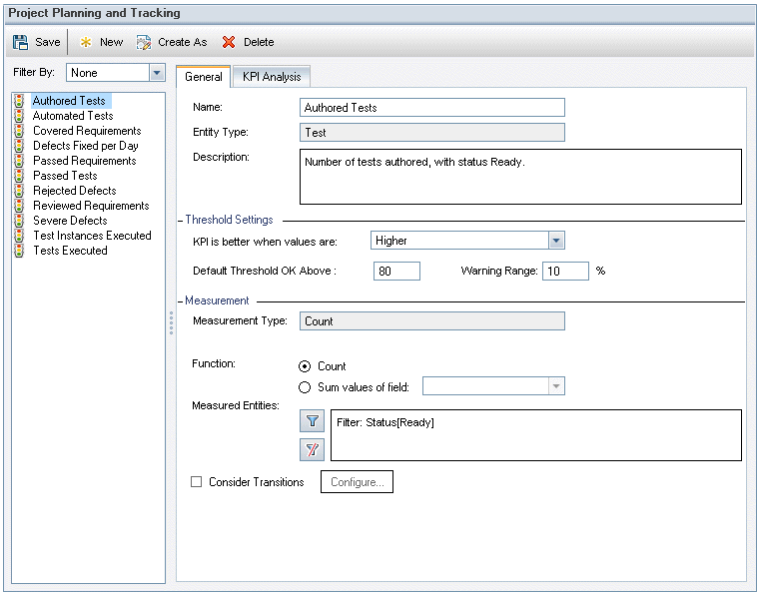
PPT collects data from your release's milestones using key performance indicators (KPIs). A **KPI** is a quantifiable measure designed to track a critical performance variable over time, and measure the essential outcome of quality assurance activities. You can customize each KPI to reflect your needs. You can customize system-defined KPIs or create user-defined KPIs.

When analyzing the overall health and deployment readiness of your release in the PPT scorecard, you can further enhance your output by customizing the KPI graphs displayed in your scorecard.

For more details on PPT, refer to the *HP Application Lifecycle Management User Guide*.

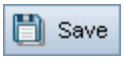
Project Planning and Tracking Page

This page enables you to customize the PPT KPIs.



To access	In Project Customization, in the left pane, click **Project Planning and Tracking**.
Important information	**ALM Editions:** The Project Planning and Tracking tab is available for ALM Edition only.
See also	"About Customizing PPT KPIs" above

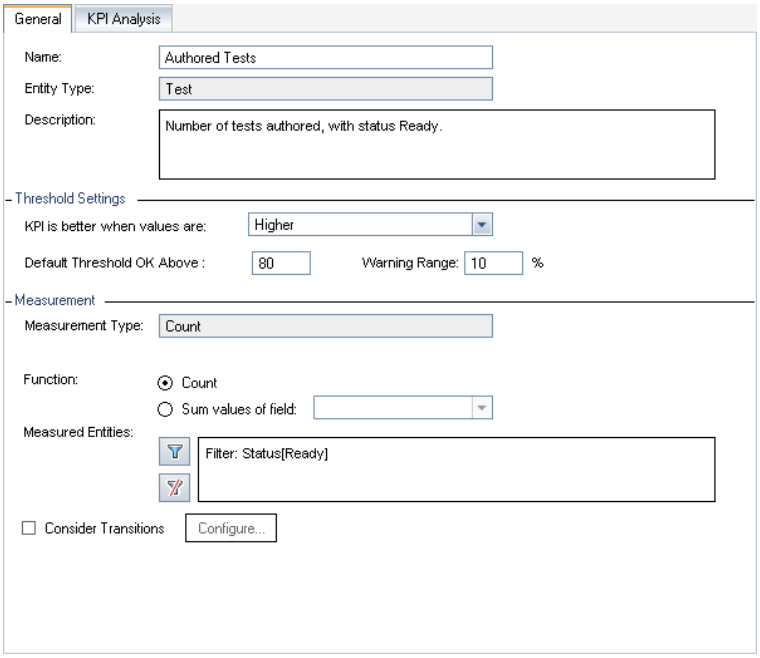
User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Elements	Description
	Saves your changes to the Project Planning and Tracking page.

UI Elements	Description
New	Opens the New KPI Type dialog box enabling you to define a new KPI by specifying a KPI name, an entity type, and measurement type.
Create As	Opens the Create As dialog box enabling you to create a KPI based on a selected KPI.
Delete	Deletes the selected KPI from the KPI Types list. **Note:** A KPI type in use cannot be deleted.
<KPI types list>	Lists available KPI types.
Filter By	KPI types associated with the selected entity type are displayed in the KPI type list. To view all KPI types, select **None**.
General tab	Displays the properties of a selected KPI type. For more details, see "Project Planning and Tracking - General Tab" below.
KPI Analysis tab	Displays the KPI drill down properties of a selected KPI type. For more details, see "Project Planning and Tracking - KPI Analysis Tab" on page 347.

Project Planning and Tracking - General Tab

This tab enables you to customize the properties of a selected KPI type.



To access	In Project Customization, in the left pane, click **Project Planning and Tracking**. Select a KPI type. The KPI properties are displayed in the General tab.

Important information	**ALM Editions:** The Project Planning and Tracking tab is available only for ALM Edition.

General Area

User interface elements are described below:

UI Elements	Description
Name	The name of the selected KPI.
Entity Type	The entity type of the selected KPI. Possible values are **Requirement**, **Test**, **Test Instance**, and **Defect**.
Description	The description of the selected KPI.

Threshold Settings Area

User interface elements are described below:

UI Elements	Description
KPI is better when values are	The expected growth direction of the values of the selected KPI. The **higher** or **lower** the value, the better it is. **Default value:** Higher
Default Threshold OK Above/Below	A value greater than the specified amount indicates a good KPI state.
Warning Range	A percentage value relevant to the **OK Above/Below** threshold. If a KPI is better when a value is higher and the **OK Above** threshold is set to **100**, and the warning range is set to **10%**, then any value between **90** and **100** will trigger a warning. Any value below **90** indicates a bad KPI state.

Measurement Area

This area enables you to define how to measure the KPI values.

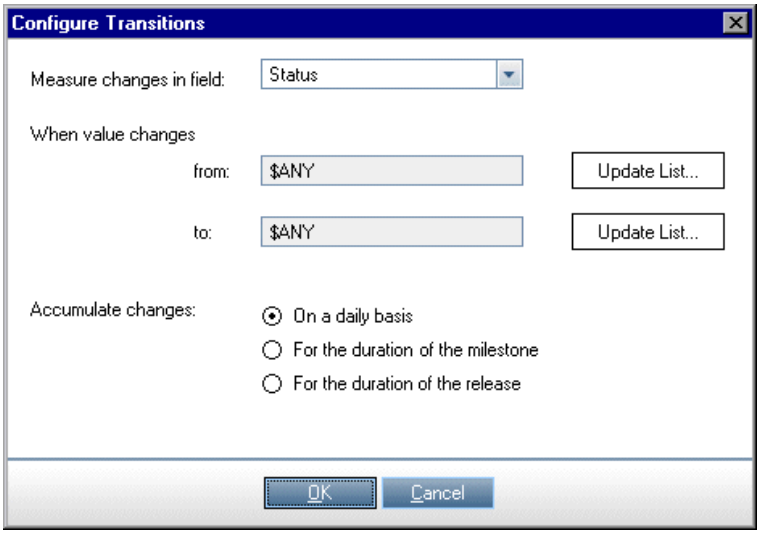
Important information	When defining the properties for the **Percentage** measurement type, the **Measure percentage of** section indicates the numerator to be used for percentage calculations. The **Out of** section indicates the denominator to be used for percentage calculations.

User interface elements are described below:

UI Elements	Description
Measurement Type	The method of measurement.
Function	Indicates one of the following: • **Count.** Counts the number of entities. • **Sum values of field.** Totals the values of a specified field for all the entities.
Measured Entities	Enables you to filter on entities of the type specified for the selected KPI: • **Set Filter/Sort.** Opens the Filter dialog box enabling you to define a filter. For more details, see the *HP Application Lifecycle Management User Guide*. • **Clear Filter.** Clears the defined filter.
Consider Transitions	Enables the **Configure** button. When transitions are configured, the KPI aggregates field changes instead of counting the changes.
Configure...	Opens the Configure Transitions dialog box, enabling you to define how field changes are counted when measuring KPI values. For more details, see "Configure Transitions Dialog Box" below.

Configure Transitions Dialog Box

This dialog box enables you to define how field changes are aggregated when measuring KPI values.



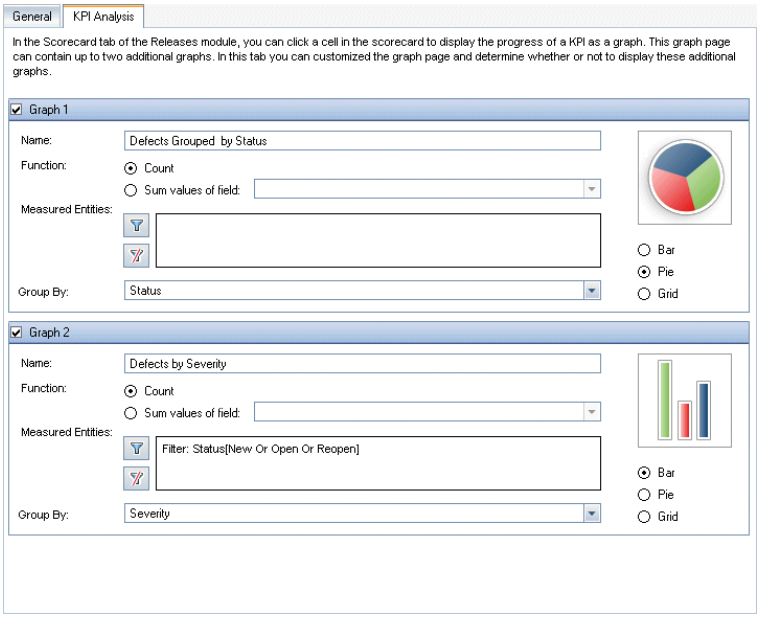
To access	In Project Customization, in the left pane, click **Project Planning and Tracking**. Select a KPI type. In the **General** tab, select **Consider Transitions** and click the **Configure** button.
Important information	**ALM Editions:** The Project Planning and Tracking tab is available only for ALM Edition.

User interface elements are described below:

UI Elements	Description
Measure changes in field	Indicates the field used for aggregating field value changes.
When value changes from	Aggregates when the value changes from the specified field value. The value **$ANY** aggregates irrespective of the currently displayed value.
When value changes to	Aggregates when the value changes to the specified field value. **$ANY** aggregates irrespective of the currently displayed value.
Update List...	Opens the Measure Values dialog box, enabling you to select the values to be used when measuring changes.
Accumulate changes	Enables you to aggregate changes on a daily basis; for the duration of a milestone; or for the duration of a release.

Project Planning and Tracking - KPI Analysis Tab

This tab enables you to define two additional breakdown graphs for inclusion in the KPI Drilldown graph. For more details, see the *HP Application Lifecycle Management User Guide*.



To access	In Project Customization, in the left pane, click **Project Planning and Tracking**. Select a KPI and click the **KPI Analysis** tab.
Important information	**ALM Editions:** The Project Planning and Tracking tab is available only for ALM Edition.

User interface elements are described below:

UI Elements	Description
Graph 1/Graph 2	Enables/disables the graph.
Name	The name of the graph.
Function	Choose one of the following: • **Count.** Counts the number of entities. • **Sum values of field.** Totals the values of a specified field for all the entities.
Measured Entities	Enables you to filter on entities of the type specified for the selected KPI: • **Set Filter/Sort.** Opens the Filter dialog box enabling you to define a filter. For more details, see the *HP Application Lifecycle Management User Guide*. • **Clear Filter.** Clears the defined filter.
Group By	Determines the fields by which ALM groups data in the graph.
Bar/Pie/Grid	Indicates the type of graph.

Chapter 22: Project Report Templates

Project report templates determine the layout and style in which data is displayed in project reports.

For more details on project reports, refer to the *HP Application Lifecycle Management User Guide*.

This chapter includes:

About Project Report Templates 351

Managing Project Report Templates 351

Designing Report Templates 354

About Project Report Templates

Project report templates are Microsoft Word files that determine the design of project reports. Users assign templates to project reports in the **Analysis View** module.

In the **Project Report Templates** page, as a project administrator, you manage report templates that are available to all project users.

Note:

- Depending on permissions, users can create and use custom report templates, in addition to the project report templates. For more information on custom templates, refer to the *HP Application Lifecycle Management User Guide*.
- Template fonts need to be installed on the client machine of the user generating the report in order for them to be properly visible in the generated document.

There are various types of templates that affect different aspects of template reports:

Template Type	Description
Document Templates	Define the outline of the report layout. For example, a document template dictates the design of the title page, whether the report includes a table of contents, page orientation, page numbering, and more.
Style Templates	Define the formatting (for example, tables, section headings, paragraphs) applied to Microsoft Word styles.
History Templates	Define the format in which history information is displayed in report sections.
Section Templates	Define the fields that are included in report sections, and the format in which they are displayed. Section templates are defined separately for each ALM entity.

Predefined templates are provided for each of the template types.

For more information on managing project report templates in Project Customization, see "Managing Project Report Templates" below.

You design report templates in Microsoft Word using the **Template Creator**. For more information, see "Designing Report Templates" on page 354.

Managing Project Report Templates

As an ALM project administrator, you manage the templates that are available to project users for creating project reports.

This section includes:

Creating a New Report Template 352

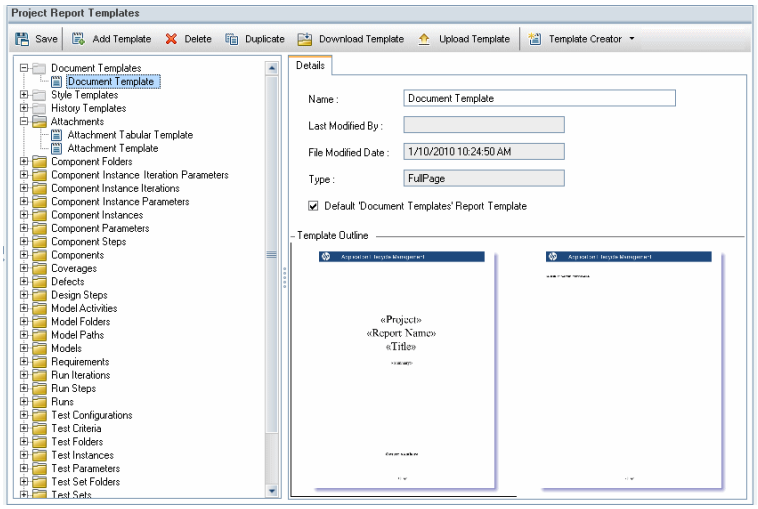
Editing a Report Template 353
Duplicating a Report Template 353
Deleting a Report Template 354

Creating a New Report Template

You can create a new report template, and make it available for users to assign to project reports.

To create a new report template:

1. In Project Customization, in the left pane, click **Project Report Templates**. The Project Report Template page opens.



2. In the templates tree, select a template type or category of the template you want to create.
3. Click the downward arrow on the **Template Creator** button, and select one of the following:
   - **Create From Default Style Template.** Creates a template file using the default style template. This is the default option if you click the button.
   - **Create From Style Template.** Creates a template file using a selected style template.
4. Design a new template file in Microsoft Word using the Template Creator tab. For details on designing template files, see "Designing Report Templates" on page 354.
5. Save and close the file.
6. In Project Customization, in the left pane, click **Project Report Templates**, and select a template category.
7. Click **Add Template,** and select the template file that you created. The following fields are displayed for the project report template:

UI Elements	Description
Name	The name of the project report template.
Last Modified By	The name of the ALM user who last made changes to the project report template.
File Modified Date	The date and time on which changes were last made to the project report template.
Type	Indicates whether the project report template is a full page or tabular template. **Full Page.** Defines the layout of ALM entity records on a page. **Tabular.** Defines the layout of ALM entity records in a table.

8. To set the new template as the default template for the category, select **Default <Category> Report Template**. When a user adds a section to a report, the default project template for the entity is initially selected.

Editing a Report Template

You can make changes to an existing report template.

To edit a report template:

1. In Project Customization, in the left pane, click **Project Report Templates**.
2. Select the template you want to edit, and click **Download Template**. A copy of the template file is saved on your computer file system, and the file opens in Microsoft Word.
3. Edit the template using the Template Creator tab. For details on designing template files, see "Designing Report Templates" on next page.
4. Save and close the template file.
5. In Project Customization, select the template, and click **Upload Template**.
6. Select the template file on your computer file system.

Duplicating a Report Template

You can create a duplicate of a report template, and modify the duplicate template.

To duplicate a template:

1. In Project Customization, in the left pane, click **Project Report Templates**.
2. Select the template you want to duplicate, and click **Duplicate**.
3. To edit the duplicated template, see "Editing a Report Template" above.
4. To set the new template as the default template for the category, select **Default <Category> Report Template**.

Deleting a Report Template

You can delete a report template.

Note: You cannot delete a template that is set as the default template in its category, or that is being used by one or more project reports.

To delete a template:

1. In Project Customization, in the left pane, click **Project Report Templates**.
2. Select the template you want to delete, and click **Delete**.

Designing Report Templates

Report templates are Microsoft Word files that contain the outline of report sections.

Note: Complete examples of report templates are provided in Project Customization. For more information, see "Managing Project Report Templates" on page 351.

This section includes:

About Designing Report Templates 354
Designing Document Templates 355
Designing Style Templates 356
Designing History Templates 356
Designing Section Templates 357
Guidelines for Creating Full-Page and Tabular Templates 359
Template Creator Tab 360

About Designing Report Templates

You design report template files in Microsoft Word. Document, history, and section templates are created using the **Template Creator** tab in Microsoft Word.

With the template creator, you select and arrange merge fields in a Microsoft Word document. **Merge fields** represent ALM field labels and values, or contain instructions for building the report. When you generate a report, the merge fields in the report templates are replaced by actual data.

For details on the template creator options, see "Template Creator Tab" on page 360.

Note: To enable the template creator:

- Microsoft Office 2007 or Microsoft Office 2010 (32 bit) must be installed on your machine.
- You must allow macros in Microsoft Word. In Word, click the **Office Button**, and then click **Word Options**. Select **Trust Center** > **Trust Center Settings** > **Macro Settings**. Select **Enable all macros**.

Designing Document Templates

In Document template files you define the outline of the report layout. For example, a Document template dictates the design of the title page, whether the report includes a table of contents, page orientation, page numbering, and more. You design document templates using the Template Creator tab in Microsoft Word.

The following elements are used in Document templates:

- **ReportName.** A merge field that is replaced in reports by the value of the report's **Name** field.
- **Custom fields.** Merge fields that represent information you want users to include in reports, for example `Author`, `Project`. You can use any string as a custom field. Users type actual values for custom fields when configuring project reports.
- **DocumentData.** A merge field that marks the point at which the report sections begin.
- **Document design.** Document formatting that you define in a Document template is used in project reports that use the Document template. This includes headers and footers, page numbers, and page layout.
- **Fixed text.** Fixed text that you type in a Document template is displayed in reports. For example, type your organization name on the cover page, or type `Created by:` before the custom field `Author`.

To design a new Document template:

1. Create a new template file using the template creator in Microsoft Word. For more information on accessing the template creator, see "Template Creator Tab" on page 360.
2. Click **Template Type** , and select **Document**.
3. To include the report name, click **Insert Field Value** , and select **ReportName**. The `ReportName` tag retrieves the report name from the **Name** field of the template based report.
4. To include customized information in areas such as the title page, headers and footers, click **Insert Custom Field** . In the Custom Field dialog box, enter a custom field name (for example, `Author`). When you create a report, you enter actual values that are displayed on the report cover page.

   Repeat the step to include additional custom fields.
5. Design the document with elements such as headers, footers, and page numbers.
6. Place the cursor at the point at which report data should begin, click **Insert Field Value** , and select **DocumentData**.

Designing Style Templates

In Style template files you define the formatting that is applied to Microsoft Word styles in all sections of a report.

For example, in a Style template you define the formatting for the `Normal` style. As a result, text in section templates that is assigned the `Normal` style, is displayed in the format you defined in the Style template.

Style formatting that is defined in the Style template overrides the formatting defined in other templates used in a project report.

Consider the following for designing Style templates:

- **Heading styles.** The `Heading 1`, `Heading 2`, `Heading n` styles that you define in a Style template are automatically applied to report sections according to the level of the section in the report. For more details on applying heading styles in section templates, see "Designing Section Templates" on next page.
- **Table styles.** To ensure a uniform style for all data tables displayed in a report, define the **Project Report Table Style**. By default, tables you create in tabular templates use this style.
- **Text.** Any text that you type in a Style template is ignored by project reports.

Designing History Templates

In History template files you define how history information is displayed in all report sections. You design History templates using the Template Creator tab in Microsoft Word.

Notes:

- To display History information in project reports, section templates must include **History** merge fields.
- History templates can be only in **tabular** format. For more details, see "Guidelines for Creating Full-Page and Tabular Templates" on page 359.

The following elements are used in History templates:

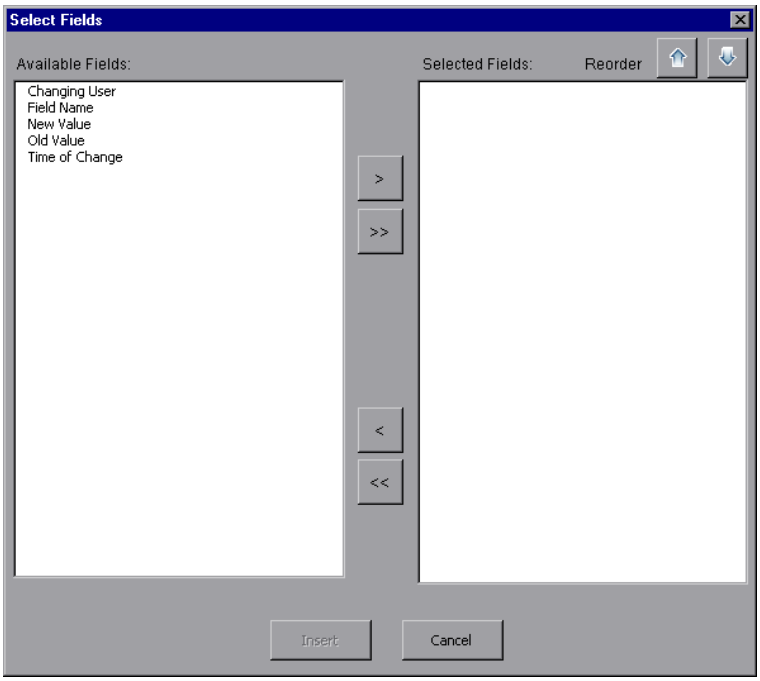
- **History fields.** Merge fields that represent history field labels and values.
- **Fixed text.** Fixed text that you type in a history template is displayed in reports. For example, type `History` in a heading row above the history merge fields.

To design a new History template:

1. Create a new template file using the template creator in Microsoft Word. For more information on accessing the template creator, see "Template Creator Tab" on page 360.
2. Click **Template Type** , and select **History**.
3. Click **Formatting**

, and select **Tabular**, to create a tabular template.

The Select Fields dialog box opens.



4. To include a field in the template, select a field in the Available Fields pane, and click the right arrow .

   Tip: Use the CTRL or SHIFT key to select multiple fields.

5. To remove a field from the template, select a field in Selected Fields pane, and click the left arrow .
6. To move all fields from one pane to the other, click the double arrows .
7. Click the **Reorder** buttons to change the order of fields in the template.
8. Click **Insert**. The fields you selected are inserted in tabular layout.

Designing Section Templates

In section templates you define how information is displayed in report sections. You define separate section templates for each ALM entity that can be included in report sections. You design section templates using the Template Creator tab in Microsoft Word.

Note: Section templates can be in either a **full-page** or **tabular** format. For more information, see "Guidelines for Creating Full-Page and Tabular Templates" on page 359.

The following elements are used in section templates:

- **Section Name.** A merge field that is replaced in reports by the value of a section's **Name** field.
- **Section Filter.** A merge field that displays the data filter applied to the report section.
- **Entity fields.** Merge fields that represent entity field labels and values.

  Note: Date and time fields are formatted using a SHORT DATE/TIME format associated with the locale defined on the ALM server. The report designer cannot change these formats.

- **History.** A merge field that inserts history information of the entity record. Use the merge field in full-page templates, within the data area.
- **Graphs.** A merge field that inserts graphs that have been added to the report. By default graphs are added to the end of a report.

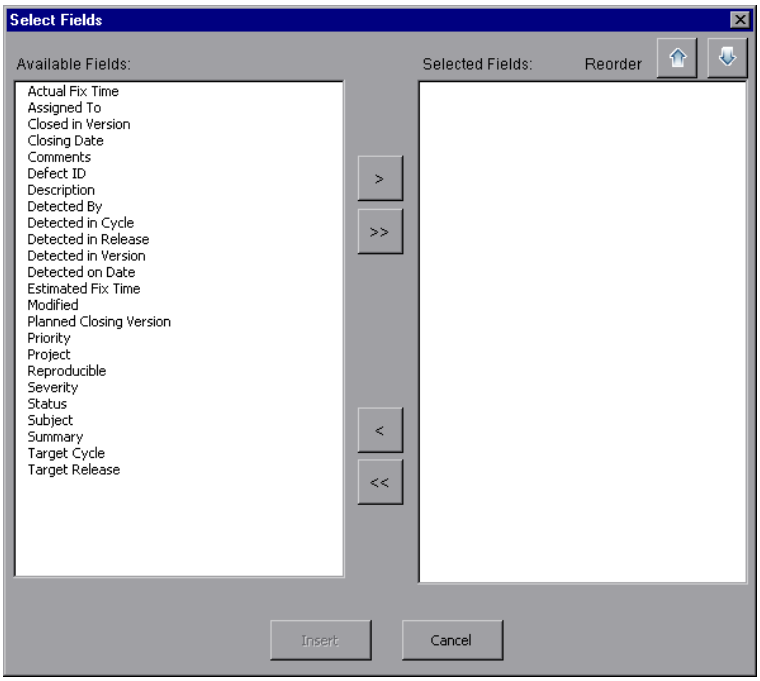
  Note: You cannot include more than one graph field in the same template.

- **Fixed text.** Fixed text that you type in a section template is displayed in reports.

To design a new section template:

1. Create a new template file using the template creator in Microsoft Word. For more information on accessing the template creator, see "Template Creator Tab" on page 360.
2. Click **Template Type** , and select a section.
3. Click **Formatting** , and select either **Full Page** or **Tabular**, to create a full-page or a tabular template.

   The Select Fields dialog box opens.



4. To include a field in the template, select a field in the Available Fields pane, and click the right arrow .

   Tip: Use the CTRL or SHIFT key to select multiple fields.

5. To remove a field from the template, select a field in Selected Fields pane, and click the left arrow .

6. To move all fields from one pane to the other, click the double arrows .

7. Click the **Reorder** buttons to change the order of fields in the template.

8. Click **Insert**. The fields you selected are inserted in full-page or tabular layout.

9. To include history information of records in a full-page template, place the cursor before the `Data End` tag, click **Insert Field Value** , and select **History**.

   Note: History information is displayed according to the History template assigned to the report.

10. To include the section name, or details of the section filter, click **Insert Field Value** , and select **Section Name** or **Section Filter**. Make sure these fields are placed outside the data area.

11. A section template can be used at any level of a report. To ensure that a section header displays in a style appropriate to its level in the report, place the cursor on the **Section Name** merge field. Make sure the **Set Auto Heading Style** button is pressed.

12. To ensure that tables use the uniform table style defined in the Style template, place the cursor in the table area. Make sure the **Set Table Style** button is pressed.

Guidelines for Creating Full-Page and Tabular Templates

Section templates can be designed in either full-page or tabular formats.

Note: Templates should consist of only one Word section. Only those text and fields in the first Word section are included in the generated report.

Full-page templates

In a **full-page** template, you arrange fields of an entity across several lines of a page. Typically, merge fields of a field's label and value are displayed on the same line, separated by a colon or tab. For example: `Detected By Label : Detected By`

In full page templates, the section of the template that is repeated for each record must be surrounded by **Data Start** and **Data End** merge fields.

Example:

Tabular templates

In a **tabular** template, you arrange fields of an entity in a two-row table. The top row of the table contains merge fields of field labels. The bottom row of the table contains merge fields of the corresponding field values.

In tabular templates, the first cell in the value row must begin with a **Table Start** merge field, and the last cell in the value row must end with a **Table End** merge field.

Example:

«Section Name»

«Defect ID Label»	«Assigned To Label»	«Detected By Label»	«Priority Label»	«Status Label»
«Table Start»«Defect ID»	«Assigned To»	«Detected By»	«Priority»	«Status»«Table End»

Template Creator Tab

The template creator enables you to design Document, History, and section templates in Microsoft Word.

To access	You access the template creator either from **Project Customization** > **Project Report Templates**, or from the project report Configuration tab. • To create a new template file, select a project report template or section, and click **Template Creator**. Microsoft Word opens, and the applicable template type is selected in the Template Creator tab. • To edit an existing template file, select a project report template, and click **Download Template** . The template opens in Microsoft Word.
Important information	To enable the template creator, you must first allow macros in Microsoft Word. In Word, click the **Office Button**, and then click **Word Options**. Select **Trust Center** > **Trust Center Settings** > **Macro Settings**. Select **Enable all macros**.
See also	"Designing Report Templates" on page 354

User interface elements are described below:

UI Elements	Description
	Choose Template Type. Lists the template types that you can create in the template creator. The selected template type is displayed in the button label, and determines the fields you can include in the template.
	Formatting. Inserts selected fields in one of the following formats: • **Full Page.** Lists selected fields vertically across multiple lines. Field values are placed alongside their labels, separated by a colon and tab. • **Tabular.** Lists selected fields horizontally in a table. Field labels are listed in the top row of the table, and field values are listed below them.
	Insert Field Label. Inserts a selected field label at the cursor position.
	Insert Field Value. Inserts a selected field value at the cursor position.
	Insert Multiple Fields. Opens the Select Fields dialog box, enabling you to insert field labels and values at the cursor location. **Note:** The selected fields are inserted on separate lines.
	Insert Custom Field. In a Document template, enables you to insert custom fields anywhere in the Document template. For example, add custom fields to the document title page, and to document headers and footers. When creating a report, users enter values that replace the custom fields in the report.
	Set Auto Heading Style. Toggles the **Template Report Auto Heading** style to the selected paragraph. In report sections based on the template, the style is automatically replaced by the **Heading** style appropriate to the section level. In full-page templates, you can apply the **Template Report Auto Heading** style both to the section heading (before the **Data Start** merge field), and to the record heading. As a result, the section heading in a report is displayed on a higher hierarchical level than the section records. In tabular templates, you can apply the **Template Report Auto Heading** style only to the section heading.
	Set Table Style. Toggles the **Template Report Table** style to the selected table.
	Connect to ALM. Enables you to connect to a different ALM project. The template creator automatically retrieves the entity fields from the selected project.
	Localize Strings. Retrieves updated field labels from Project Customization.

Chapter 23: Business Views

This section describes how to create and manage business views, which can be used as a basis for ALM's reporting tools.

This chapter includes:

Business Views Overview 363

Creating and Managing Business Views 363

Working with DQL 365

Business Views User Interface 387

Business Views Overview

Business views are a semantic data layer that can be used as a basis for the various HP Application Lifecycle Management (ALM) reporting tools.

The views are based on project entities, and ensure that only information that is relevant to a business consumer is revealed in a report. For example, you can select only those defect related entity fields that represent some business value to a report consumer. You then create a unique view which can be used and reused as a report basis.

For example, a business view based on **Baselines** could contain the **Name**, **Description**, and **Baseline ID** fields. This is because these fields convey information that could be of importance to the graph consumer who needs to understand baseline information from a business perspective only. By the same logic, the same business view does not include the **Attachment** and **Auto complete type** fields, as they are of little business significance.

Creating reports based on business views ensures standardization across the reports, as different reports that are based on the same business view reflect a common frame of reference. This provides genuine business value to the reports consumer, thereby making the reports more meaningful.

Note: Reports are generated according to the permission levels of the user generating them. Therefore, information that is included in a business view that is not available to a user will not appear in any report that user creates.

Business views can be based on a single project entity (for example, defects), or can represent more complex relationships between several entities (for example, defects, requirements, and tests). ALM provides a set of pre-defined business views that are available for selection.

Business views must be designed by a user who understands the business requirements of the organization. In addition, the process of creating business views utilizes DQL, a domain query language that is a flavor of ANSI SQL. It is important that the business view designer be able to understand DQL queries. For more information on working with DQL, see "Working with DQL" on page 365.

Creating and Managing Business Views

This task describes how to create and manage business views.

To learn more about business views, see "Business Views Overview" above.

1. **Open the Business Views page**

   a. On the common toolbar, select **Tools > Customize**. The Project Customization window opens.

   b. In the Project Customization window, click the **Business Views** link. The Business Views page opens. For user interface details, see "Business Views Page" on page 388.

2. **Create an empty business view**

   On the Business Views page toolbar, click **Add View**. The New View dialog box opens. Enter a technical name and label, and click **OK**.

3. **Add project entities to the business view**

   a. On the Business Views page, if it is not selected, select the **Query Designer** tab.

   b. On the Query Designer tab toolbar, click **Add Entity**. The Model tree opens in the right pane. The Model tree displays all project entities and fields in the current project.

   c. Select the entities you want to add to the new business view and drag them to the **Main** pane in the middle of the Query Designer tab. For each project entity you add, a <project entity> dialog box is added, displaying all available fields contained within the entity. You use the checkboxes next to the field names to select specific fields to include in the view.

   Note: By default, all fields are selected and the checkboxes are unchecked.

4. **Define relationships between selected entities**

   You create relationships between selected entities in one of the following ways:

   - Select a <project entity> dialog box, and on the Query Designer tab toolbar click **Add Related Entity**. The Add Related Entity dialog box opens. For user interface details, see "Add Related Entity Dialog Box" on page 394.
   - When more than one entity has been selected, create the relationships manually by dragging a field from one entity to another.
   - Enter the query directly in the DQL Query Builder.

5. **Edit filter criteria - optional**

   You can edit and define filter criteria for selected fields in the selected fields grid below the Main panel.

   Tip: You can change a field's label using the **Alias** column of the selected fields grid.

6. **Edit entity relationships - optional**

   To edit relationships between entities, double click on a relationship line. The Link Properties dialog box opens. For user interface details, see "Link Properties Dialog Box" on page 393.

7. **Validate the business view**

   On the Query Designer tab toolbar, click **Validate**. Any warning or error messages are shown in the Query Results pane at the bottom of the Query Designer tab.

8. **Preview the business view**

   On the Query Designer tab toolbar, click **Preview**. The Query Results pane opens at the bottom of the Query Designer tab.

9. **Update the business view status**

   On the Query Designer tab toolbar, click the **Status** arrow and select **Published**.

10. **Save the business view**

    On the Business Views page toolbar, click **Save**.

Working with DQL

The process of creating business view queries utilizes DQL, a domain query language.

For the most part, DQL is an exact match to ANSI SQL 9.2, however there are some significant differences.

Note: DQL only supports SELECT statements.

This section includes:

- Advantages of DQL
- Additions to SQL
- Supported Functions

Advantages of DQL

Building queries with DQL has the following advantages:

- DQL enforces data hiding in accordance with user's permission levels. In other words, reports are generated according to the permission levels of the user generating them. Information that is included in a business view that is not available to a user does not appear in any report that the user creates.
- The DQL query creates a database abstraction which is then used as a basis for the business view. Since the basis of the view is an abstraction of the database and not the database itself, you do not need to identify entity fields according to their actual names. Rather the query simplifies them and makes them easier to identify. For example, all fields that relate to an object's identification appear with the suffix "ID", such as Defect ID, Cycle ID, and Release ID.
- DQL queries run equally well on Oracle and SQL database servers.

Additions to SQL

DQL provides the following options, which are not available in ANSI SQL:

- **Variables.** Three variables have been added:
  - **:me.** Returns information relevant to the user creating the report. Meaning different users who create reports based on the same business view will receive different results.
  - **:current_project_name.** Returns information relevant for the project from which the reports is created.
  - **:current_domain_name.** Returns information relevant for the domain from which the report is created.

  Following is an example of the **Me** variable:

```
Select *
```

```
From defect

Where defect.detected_by = :me
```

- **Select Top.** This option enables you to limit the query results to a defined number of items.

Supported Functions

The following table lists the DQL functions that are supported in ALM. DQL functions are converted to the corresponding SQL functions for use in the database. The table lists the conversions of the DQL functions to MS SQL and Oracle. For more information, refer to the MS SQL and Oracle documentation.

Function	Signature	Return Type	Description
Aggregation			
count	count(expression)	integer	Returns the number of rows in a query.
count_big	count_big(expression)	integer (bigint)	Returns the number of rows in a query. The only difference between **count** and **count_big** is their return values. In MS SQL, **count_big** always returns a bigint data type value, and **count** always returns an int data type value. **Note:** In Oracle, the function is converted to **count**.
variance	variance(decimal)	decimal	Returns the variance of the expression . **Note:** In MS SQL, the function is converted to **var**. In Oracle, the function is coded as **round(variance (<decimal>), 14)**.

Function	Signature	Return Type	Description
var_pop	var_pop(expression)	decimal	Returns the population variance of a set of numbers after discarding the nulls in the set. **Note:** In MS SQL, the function is converted to **varp**. In Oracle, the function is coded as **round(var_pop (<expression>), 14)**.
stddev	stddev(decimal)	decimal	Returns the standard deviation of a set of numbers. **Note:** In MS SQL, the function is converted to **stdev**. In Oracle, the function is coded as **round(stddev (<decimal>), 14)**.
stddev_pop	stddev_pop(expression)	decimal	Computes the population standard deviation and returns the square root of the population variance. **Note:** In MS SQL, the function is converted to **stdevp**. In Oracle, the function is coded as **round(stddev_ pop(<expression>), 14)**.
avg	avg(expression)	decimal	Returns the average value of an expression. **Note:** In Oracle, the function is coded as **round(avg (<decimal>), 14)**.

Function	Signature	Return Type	Description
sum	sum(decimal)	decimal	Returns the summed value of an expression.
min	min(expression)	expression	Returns the minimum value of an expression.
max	max(expression)	expression	Returns the maximum value of an expression.
String Manipulation			
upper	upper(string)	string	Returns the given string with all letters converted to upper case.
lower	lower(string)	string	Returns the given string with all letters converted to lower case.
rtrim	rtrim(string)	string	Returns the given string with all trailing spaces removed from the right-hand side.
ltrim	ltrim(String)	string	Returns the given string with all leading spaces removed.
replace	replace(string, string_to_replace, replacement_string)	string	Returns the given string with all sequences of **<string_to_replace>** replaced with **<replacement_string>**.
substring	substring(string, int_start_position, int_length)	string	Returns a substring from the given string. • **int_start_position**. The position from which to start extracting the substring. • **int_length**. The number of characters to extract. **Note:** In Oracle, the function is converted to **substr**.

Function	Signature	Return Type	Description
length	length(string)	decimal	Returns the length of the specified string. **Note:** In MS SQL, the function is converted to **len**.
chr	chr(integer)	string	Converts an int ASCII code into a character. **Note:** In MS SQL, the function is converted to **char**.
soundex	soundex(string)	string	Returns a phonetic representation of a string.
ascii	ascii(string)	integer	Returns the ASCII code value of the leftmost character of a character expression.
concat	concat(string1, string2)	string	Concatenates two strings together . **Note:** In MS SQL, the function is converted to **+**, and the function is coded as **<string1> + <string2>**.
leftstr	leftstr(string, integer)	string	Returns the left part of a character string with the specified number of characters. **Note:** In MS SQL, the function is converted to **left**. In Oracle, the function is converted to **substr** and is coded as **substr(<string>, 1, <integer>)**.

Function	Signature	Return Type	Description
lpad	lpad(string1, integer, string2)	string	Returns **<string1>**, left-padded to length **<integer>** characters with the sequence of characters in **<string2>**. This function is useful for formatting the output of a query. **Note:** In MS SQL, the function is converted to the **left padding** expression, which implements the above logic.
reverse	reverse(string)	string	Returns the reverse of a string value. **Note:** In Oracle, the function is coded as **reverse(to_char (<string>))**.
rightstr	rightstr(string, integer)	string	Returns the right part of a character string with the specified number of characters. **Note:** In MS SQL, the function is converted to **right**. In Oracle, the function is converted to **substr** with the **right part** expression, which implements the above logic.

Function	Signature	Return Type	Description
rpad	rpad(string1, integer, string2)	string	Returns **<string1>**, right-padded to length **<integer>** characters with **<string2>**, replicated as many times as necessary. This function is useful for formatting the output of a query. **Note:** In MS SQL, the function is converted to **left** with the **right padding** expression, which implements the above logic.
stuff	stuff(string, integer, integer, string)	string	Inserts a string into another string. It deletes a specified length of characters in the first string at the start position, and then inserts the second string into the first string at the start position. **Note:** In Oracle, the **stuff** function is converted to an expression that implements the above logic.

Function	Signature	Return Type	Description
in_string	in_string(string1, string2, integer_start_location)	integer	Returns the location of substring **string2** in **string1**, if it exists. Otherwise, returns 0. The search starts at **integer_start_location**, which starts at 1. **Note:** In MS SQL, the function is converted to **charindex**. In Oracle, the function is converted to **instr** and is coded as **instr (<string2>, <string1>, <integer_start_location>)**.
replicate	replicate(string, integer)	string	Repeats the **<string>** value the number of times given in the **<integer>** parameter. **Note:** In Oracle, the function is converted to **rpad** and is coded as **rpad(<string>, length (<string>*<integer>), <string>)**.
Math			
sin	sin(decimal)	decimal	Returns the sine of the **decimal** parameter. **Note:** In Oracle, the function is coded as **round(sin (<decimal>), 14)**.

Function	Signature	Return Type	Description
asin	asin(decimal)	decimal	Returns the arc sine of the **decimal** parameter. The argument must be in the range of -1 to 1. **Note:** In Oracle, the function is coded as **round(asin (<decimal>), 14)**.
cos	cos(decimal)	decimal	Returns the cosine of the **decimal** parameter. **Note:** In Oracle, the function is coded as **round(cos (<decimal>), 14)**.
acos	acos(decimal)	decimal	Returns the arc cosine of the **decimal** parameter. The argument must be in the range of -1 to 1. **Note:** In Oracle, the function is coded as **round(acos (<decimal>), 14)**.
tan	tan(decimal)	decimal	Returns the tangent of the **decimal** parameter. **Note:** In Oracle, the function is coded as **round(tan (<decimal>), 14)**.

Function	Signature	Return Type	Description
atan	atan(decimal)	decimal	Returns the arc tangent of the **decimal** parameter. **Note:** In Oracle, the function is coded as **round(atan (<decimal>), 14)**.
atan2	atan2(decimal1, decimal2)	decimal	Returns the angle, in radians, between the positive x-axis and the ray from the origin to the point (y, x), where x and y are the values of **decimal1** and **decimal2**. **Note:** In MS SQL, the function is converted to **atn2**. In Oracle, the function is coded as **round(atan2 (<decimal1>), 14)**.
tanh	tanh(decimal)	decimal	Returns the hyperbolic tangent of the **decimal** parameter. **Note:** In MS SQL, the function is converted to the **tanh** expression, which implements the above logic. In Oracle, the function is coded as **round(tanh (<decimal>), 14)**.

Function	Signature	Return Type	Description
sqrt	sqrt(decimal)	decimal	Returns the square root of decimal parameters. **Note:** In Oracle, the function is coded as **round(sqrt (<decimal>), 14)**.
exp	exp(decimal)	decimal	Returns **e** raised to the **decimal** parameter power, where e = 2.71828183. **Note:** In Oracle, the function is coded as **round(exp (<decimal>), 14)**.
sign	sign(decimal)	decimal	Returns a value indicating the sign of a number. If the number < 0, then **sign** returns -1. If the number = 0, then **sign** returns 0. If the number > 0, then **sign** returns 1. **Note:** In MS SQL, the function is converted to the **sign** expression, which implements the above logic, and is coded as **CAST(sign (<decimal>) AS int)**.
floor	floor(decimal)	integer	Returns the largest integer less than or equal to the specified decimal argument.

Function	Signature	Return Type	Description
ln	ln(decimal)	decimal	Returns the natural logarithm of the decimal argument. **Note:** In MS SQL, the function is converted to **log**. In Oracle, the function is coded as **round(ln (<decimal>), 14)**.
abs	abs(decimal)	decimal	Returns the absolute value of a number.
round	round(decimal, integer)	decimal	Returns **decimal** rounded to **integer** places to the right of the decimal point. The **integer** parameter can be negative to round off digits left of the decimal point. **Note:** In MS SQL, the function is coded as **CAST(round (<decimal>, <integer>) AS int)** or as **CAST(round (<decimal>, <integer>) AS float)**, depending on whether **integer** is valued with zero..
mod	mod(decimal1, decimal2)	decimal	Returns the remainder of **decimal1** divided by **decimal2**. **Note:** In MS SQL, the function is converted to **%** and is coded as **<decimal1> % <decimal2>**.

Function	Signature	Return Type	Description
trunc_number	trunc_number(decimal, integer)	decimal	Returns **decimal** truncated to **integer** decimal places. If **integer** is omitted, then **decimal** is truncated to 0 places. **Note:** In MS SQL, the function is converted to the **truncation** expression, which implements the above logic. In Oracle, the function is converted to **trunc**.
str	str(decimal, integer1, integer2)	string	Returns character data converted from numeric data. **Note:** In Oracle, the function is converted to the **string construction** expression, which implements the above logic.
ceil	ceil(decimal)	decimal	Returns the smallest integer greater than or equal to the decimal parameter. **Note:** In MS SQL, the function is converted to **ceiling**.
Cast			

Function	Signature	Return Type	Description
to_number	to_number(string)	decimal	Converts a string to a number. **Note:** In MS SQL, the function is coded as **CAST(<string> AS float)**.
number_to_char	number_to_char(decimal)	string	Converts a number to a string. **Note:** In MS SQL, the function is coded as **CAST(<decimal> AS varchar(50))**. In Oracle, the function is converted to **to_char**.
char_to_char	char_to_char(string)	string	Converts NCHAR, NVARCHAR2, CLOB, or NCLOB data to the database character set. **Note:** In MS SQL, the function is coded as **CAST(<string> AS varchar(50))**. In Oracle, the function is converted to **to_char**.
datetime_to_char	datetime_to_char (expression)	string	Converts a date or datetime to a string. **Note:** In MS SQL, the function is coded as **convert(varchar, <date>, 121)**. In Oracle, the function is converted to **to_char** and is coded as **to_char(<date>, 'YYYY-MM-DD HH24:MI:SS:FF3')**.

Function	Signature	Return Type	Description
Miscellaneous			
nullif	nullif(expression1, expression2)	expression	Compares **expression1** and **expression2**. If **expression1** and **expression2** are equal, the function returns NULL. Otherwise, it returns **expression1**.
coalesce	coalesce()	expression	Returns the first non-null expression among its arguments. The data type of the returned value is the data type of the parameter with the highest data type precedence.
isnull	isnull(check_expression, replace_expression)	expression	Substitutes a value when a null value is encountered. If **check_expression** is null, then the function returns **replace_expression**. **Note:** In Oracle, the function is converted to **nvl**.
Date			
currentdate	currentdate()	date	Returns the current date. **Note:** In MS SQL, the function is coded as **CAST(CAST(getdate() AS date) AS datetime)**. In Oracle, the function is coded as **to_date(to_char(sysdate))**.

Function	Signature	Return Type	Description
dateadd	dateadd(integer, date)	date	Adds days to date parameters: • **integer**. The numbers of days to add. • **date**. The date to add to. **Note:** In MS SQL, the function is coded as **dateadd(DAY, <int days>, <Date date>)**. In Oracle, the function is converted to **+** and is coded as **<int days> + <Date date>**.
datediff	datediff(start_date, end_date)	integer	Returns the count (signed integer) of dates crossed between the specified **start_date** and **end_date**. **Note:** In MS SQL, the function is coded as **datediff(DAY, <date>, <date>)**. In Oracle, the function is converted to **-** and is coded as **trunc (<date> - <date>, 0)**.

Function	Signature	Return Type	Description
get_time	get_time()	datetime	Returns the current date and time. **Note:** In MS SQL, the function is converted to **getdate** and is coded as **convert (datetime, getdate(), 126)**. In Oracle, the function is converted to **LOCALTIMESTAMP**.
trunc_date	trunc_date(date, string_ format)	date	Returns a date truncated to a specific unit of measure. **string_format** is the unit of measure to apply for truncating. The following formats are supported: year, y, yy, yyyy, q, quarter, mm, month, d, dd, day, hh, mi **Note:** In MS SQL, the function is converted to **dateadd** and is coded as **dateadd (<unquoted string_ format>, datediff (<unquoted string_ format>, 0, <date>), 0)**. In Oracle, the function is converted to **trunc** and is coded as **trunc(<date>, <string_format>**.

Function	Signature	Return Type	Description
to_date	to_date(string)	date	Converts a string to a date. **Note:** In MS SQL, the function is coded as **CAST(<string> AS datetime)**. In Oracle, the function is coded as **to_date(<string>, 'YYYY-MM-DD')**.
timestamp_tz	timestamp_tz()	string	Returns the current server time zone . **Note:** In MS SQL, the function is converted to **sysdatetimeoffset**. In Oracle, the function is converted to **CURRENT_ TIMESTAMP**. This function is not supported in MS SQL Server 2005.
from_tz	from_tz(date, string)	date	Converts a timestamp value (**date**)and a time zone (**string**)to a timestamp with time zone value. **Note:** In MS SQL, the function is converted to the **time stamp** and **zone** expressions, which implement the above logic. This function is not supported in MS SQL Server 2005.

Function	Signature	Return Type	Description
sysdatetimeoffset	sysdatetimeoffset()	datetime with time zone	Returns the system date, including fractional seconds, and the time zone of the system on which the database resides. **Note:** In Oracle, the function is converted to **SYSTIMESTAMP**. This function is not supported in MS SQL Server 2005.
getyear	getyear(date)	integer	Returns an integer that represents the year of the specified date. **Note:** In MS SQL, the function is converted to **year**. In Oracle, the function is converted to **to_char** and is coded as **CAST(to_char(<date>, 'yyyy') AS number)**.
getmonth	getmonth(date)	integer	Returns an integer that represents the month of the specified date. **Note:** In MS SQL, the function is converted to **month**. In Oracle, the function is converted to **to_char** and is coded as **CAST (to_char(<date>, 'mm') AS number)**.

Function	Signature	Return Type	Description
getday	getday(date)	integer	Returns an integer representing the day of the month of the specified date. **Note:** In MS SQL, the function is converted to **day**. In Oracle, the function is converted to **to_char** and is coded as **CAST(to_char(<date>, 'dd') AS number)**.
datepart	datepart(string_datepart, date)	integer	Returns an integer that represents the specified **string_datepart** of the specified date. **Note:** In MS SQL, the function is coded as **datepart(<unquoted string_datepart>, <date>)**. In Oracle, the function is converted to the **datepart** expression, which implements the above logic.

Function	Signature	Return Type	Description
datename	datename(string_ datepart, date)	string	Returns a character string that represents the specified **string_datepart** of the specified date. The following formats are supported: yyyy, yy, year, q, quarter, m, month, d, dd, day, hh, mi **Note:** In MS SQL, the function is coded as **datename (<unquoted string_ datepart>, <date>)**. In Oracle, the function is converted to the **datename** expression, which implements the above logic.
Case			
decode	decode(expression, search, result [, search, result]… [, default])	expression	Das the functionality of an IF-THEN-ELSE statement. Compares the expression to each search value one by one. If the expression is equal to a search, then an Oracle Database returns the corresponding result. If no match is found, then Oracle returns **default**. If default is omitted, then Oracle returns **null**. **Note:** In MS SQL, the function is converted to the **case-when-else** expression, which implements the above logic.

Function	Signature	Return Type	Description
greatest	greatest()	expression	Returns the greatest of the list of parameters. **Note:** In MS SQL, the function is converted to the **greatest** expression, which implements the above logic.
least	least()	expression	Returns the least of the list of parameters. **Note:** In MS SQL, the function is converted to the **least** expression, which implements the above logic.
Special Date **Note:** For each function, the requested date is returned in the respective date format. In MS SQL, the requested date in yyyy-MM-dd format is cast as datetime. In Oracle, to_date (<requested date in dd-MMM-yy format>, DD-MON-YY)			
firstDayOfCurrentMonth	firstDayOfCurrentMonth()	date	Returns the date of the first day of the current month.
firstDayOfCurrentWeek	firstDayOfCurrentWeek()	date	Returns the date of the first day of the current week.
firstDayOfCurrentYear	firstDayOfCurrentYear()	date	Returns the date of the first day of the current year.
firstDayOfNextMonth	firstDayOfNextMonth()	date	Returns the date of the first day of the next month.
firstDayOfNextWeek	firstDayOfNextWeek()	date	Returns the date of the first day of the next week.
firstDayOfNextYear	firstDayOfNextYear()	date	Returns the date of the first day of the next year.

Function	Signature	Return Type	Description
firstDayOfPreviousMonth	firstDayOfPreviousMonth ()	date	Returns the date of the first day of the previous month.
firstDayOfPreviousWeek	firstDayOfPreviousWeek ()	date	Returns the date of the first day of the previous week.
firstDayOfPreviousYear	firstDayOfPreviousYear()	date	Returns the date of the first day of the previous year.
lastDayOfCurrentMonth	lastDayOfCurrentMonth()	date	Returns the date of the last day of the current month.
lastDayOfCurrentWeek	lastDayOfCurrentWeek()	date	Returns the date of the last day of the current week.
lastDayOfCurrentYear	lastDayOfCurrentYear()	date	Returns the date of the last day of the current year.
lastDayOfNextMonth	lastDayOfNextMonth()	date	Returns the date of the last day of the next month.
lastDayOfNextWeek	lastDayOfNextWeek()	date	Returns the date of the last day of the next week.
lastDayOfNextYear	lastDayOfNextYear()	date	Returns the date of the last day of the next year.
lastDayOfPreviousMonth	lastDayOfPreviousMonth ()	date	Returns the date of the last day of the previous month.
lastDayOfPreviousWeek	lastDayOfPreviousWeek ()	date	Returns the date of the last day of the previous week.
lastDayOfPreviousYear	lastDayOfPreviousYear()	date	Returns the date of the last day of the previous year.

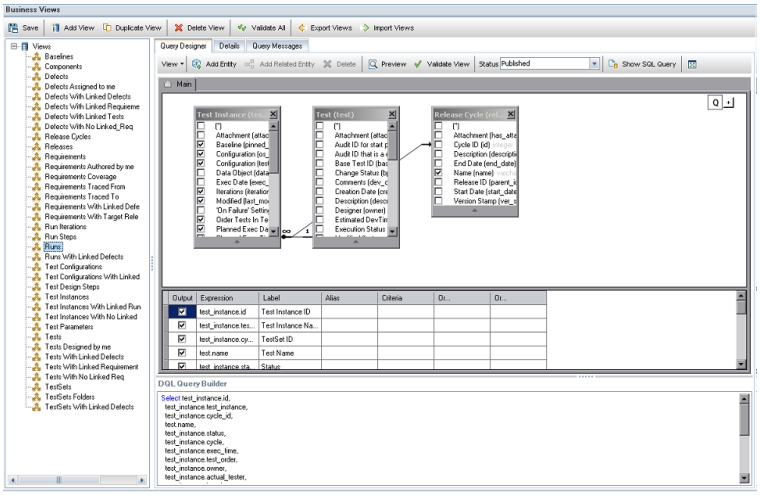
Business Views User Interface

This section includes:

Business Views Page 388
Link Properties Dialog Box 393
Add Related Entity Dialog Box 394

Business Views Page

The Business Views page enables you to create and manage business views.



To access	1. On the ALM Masthead, select **Tools > Customize**. The Project Customization page opens. 2. In the Project Customization window, in the left pane, click **Business Views**.
See also	"Creating and Managing Business Views" on page 363

Business Views Page Common Elements

User interface elements are described below:

UI Element	Description
	Saves changes made in the Business Views page.
	Opens the New View dialog box. Enter the following information: • **Label.** The name of the view which is used for identification purposes only. For example, in the Views tree, and when selecting views in the Analysis View module, views are listed according to their labels. You can change the label in the selected fields grid. For more information, see "<selected fields grid>" on page 391. • **Technical name.** The name of the view which is used as part of the DQL query. The name cannot contain spaces. Click **OK**. The view is added to the Views tree.
	Duplicates the selected view. The duplicated view is added to the Views tree.

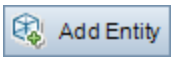
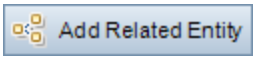
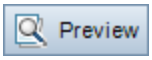
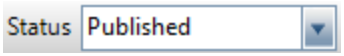
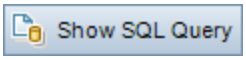
UI Element	Description
Delete View	Deletes the selected view. **Note:** You will no longer be able to view reports or graphs based on the deleted view.
Validate All	Validates all views.
Export Views	Opens the Save As dialog box, enabling you to save the selected view as an **.xml** file. **Tip:** To select multiple views, press the CTRL key and select the desired views.
Import Views	Opens the Open dialog box enabling you to import views. **Note:** By default, imported views are not valid.
Views tree	Displays pre-defined as well as user-defined views. The icon next to the view's name indicates the view's status. • The view is valid and is published. • The view is valid and is not published. • The view is not valid. **Note:** Views are listed alphabetically according to their labels. Place the cursor over the label to view a tooltip displaying the view's technical name.

Query Designer tab

This tab enables you to customize business views.

User interface details are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
View	Determines the view of the Query Designer tab. Click the arrow to select the desired view: • **QBE Only.** Hides the **DQL Query Builder**. • **DQL Only.** Displays only the **DQL Query Builder**. • **QBE and DQL.** Displays the entire Query Designer tab. This is the default view.

UI Element	Description
	Opens the Model pane on the right of the tab, enabling you to select project entities to add to a view. For more information, see Model pane.
	Opens the Add Related Entity dialog box, enabling you to define joins between selected entities. For user interface details, see "Add Related Entity Dialog Box" on page 394.
	Opens the Query Results pane at the bottom of the tab. For more information, see Query Results pane.
	Validates the selected view. The following checks are performed: • That the DQL syntax is correct. • That the query contains only fields from selected entities.
	Defines whether the selected view is available for use in reporting. • **Published.** The view is available. • **Not Published.** the view is not available. **Note:** • When changing a view from **Not Published** to **Published**, ALM first validates the view before changing the status. • When changing a view from **Published** to **Not Published**, you will no longer be able to view reports or graphs based on that view.
	Opens the SQL Query dialog box which displays the SQL query which is run against the database server. **Invalid views.** The Messages tab displays details about problems with the view. Place the cursor over the message text to view a tooltip displaying the full message.

UI Element	Description
Main pane	Displays a <Project entity> dialog box for every entity that has been added to the view, as well as defined relationships between entities. • For more information about the <Project entity> dialog box, see <Project_entity>_dialog_box. • For information about defining relationships between entities, see "Creating and Managing Business Views" on page 363
<project entity> dialog box	Appears in the Main pane after you add a project entity to the view. The dialog box displays all available fields within the entity. Use the checkboxes to select fields to add to the view. **Default:** All fields all included in the view. The checkboxes appear unchecked.
Model pane	Displays available project entities. To add an entity to the view, select it and click the **Add** button . Alternatively, you can add entities by dragging them to the Main pane. **Note:** Entities are listed alphabetically according to their Labels, with their technical names in brackets.
<selected fields grid>	Enables you to define filter criteria for entity fields. To add a field to the grid, in the <project entity> dialog box in the Main pane, use the checkboxes to select the desired fields. **Tip:** You can use the **Alias** column to change a field's label. **Note:** You cannot change a label for a field in a sub-entity.
DQL Query Builder	Displays the business view query. The query updates automatically as you add entities and define relationships. You can create and edit a view by entering the query directly in the DQL Query Builder. For more information about working with DQL, see "Working with DQL" on page 365.

UI Element	Description
Query Results pane	Appears when you click **Preview** on the Query Designer tab toolbar. Displays the following information: • **Query Results.** Valid views only. Displays a preview of the view. • **Query Messages.** Invalid views only. Displays messages detailing problems with the view. Place the cursor over the message text to view a tooltip displaying the full message.

Details Tab

This tab enables you to view or edit the selected view's details.

User interface details are described below:

UI Element	Description
Label	The name of the view which is used for identification purposes only. For example, in the Views tree, and when selecting views in the Analysis View module, views are listed according to their labels. **Tip:** You can change the label in the selected fields grid. For more information, see "<selected fields grid>" on previous page.
Technical name	The name of the view which is used as part of the DQL query. **Note:** The technical name cannot contain spaces.
Description tab	A description of the view. Click in the text box to display a toolbar for formatting and spell checking the text.

Query Messages Tab

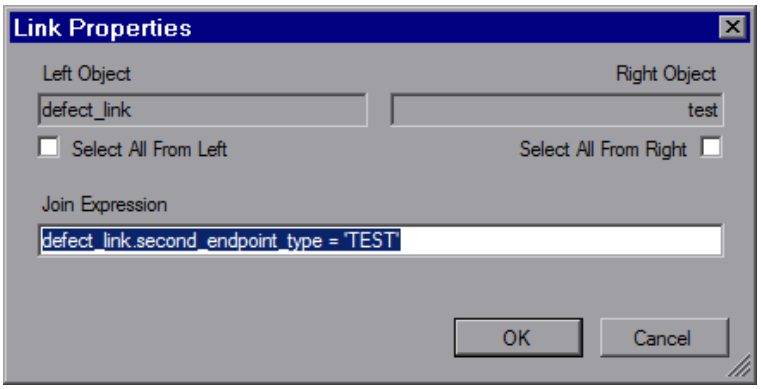
This tab enables you to view messages detailing problems with the view.

User interface details are described below:

UI Element	Description
Message Severity	The severity of the error. **Note:** Business views with warning messages can be published. Business views with errors cannot be published. Some business views with warning messages can cause the business view reports not to be generated. For example, a business view created with duplicate fields or aliases generates a warning message, but no report is generated from this business view.
Message Text	The full error message.

Link Properties Dialog Box

The Link Properties dialog box enables you to edit relationships between entities.



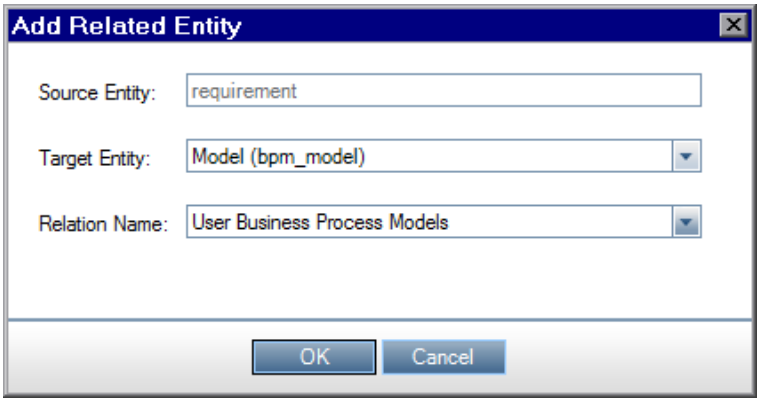
To access	Double-click a relation line between two related entities.
See also	"Creating and Managing Business Views" on page 363

User interface elements are described below.

UI Element	Description
Left/Right Object	Displays the technical names of the joined entities.
Select all from left/right	Use the checkboxes to define if the relationship contains inner, left, right, or outer joins.
Join expression	Enables you to edit the join expression.

Add Related Entity Dialog Box

The Add Related Entity dialog box enables you to define joins between project entities. The dialog box enables you to select from all entities related to the source entity.



To Access	• In the Query Designer tab Main pane, select a <project entity> dialog box and on the toolbar, click **Add Related Entity**. • Opens automatically when you add an entity to the view that already has a defined relationship with another entity in the view.
See Also	"Creating and Managing Business Views" on page 363

User interface elements are described below:

UI Element	Description
Source Entity	The currently selected entity.
Target Entity	Displays a drop-down list enabling you to select an available target entity. For each available entity, the entity label appears, with its technical name in brackets.
Relation Name	Displays the relation name given the source and target entities. If more than one relation name exists, you can choose the desired name from the drop-down list.

Chapter 24: Configuring Business Process Testing

This chapter describes how to configure Business Process Testing for creating business components in HP Application Lifecycle Management (ALM).

This chapter includes:

About Configuring Business Process Testing 397

Business Process Test Page 397

About Configuring Business Process Testing

You can set additional Business Process Testing options, such as automatic creation of manual design steps as business components are created.

For more information on working with Business Process Testing in ALM, refer to the *HP Business Process Testing User Guide*.

Business Process Test Page

The Business Process Test customization page enables the project administrator to customize Business Process Testing.

To access	Select the **Tools > Customize** menu option from the main ALM toolbar. Select the **Business Process Test** sidebar entry.
Relevant tasks	*HP Business Process Testing User Guide*

User interface elements are described below:

UI Element	Description
Automatically create new component with manual implementation	If checked, Business Process Testing assumes that new components should have a manual implementation and automatically prepares them for the creation of manual design steps when creating a business component.

Chapter 25: Configuring Sprinter

This chapter describes how to configure HP Sprinter for running tests manually in HP Application Lifecycle Management (ALM).

ALM Editions: Sprinter functionality is not available with ALM Essentials Edition or Performance Center Edition.

This chapter includes:

About Configuring Sprinter 399

Sprinter Page 399

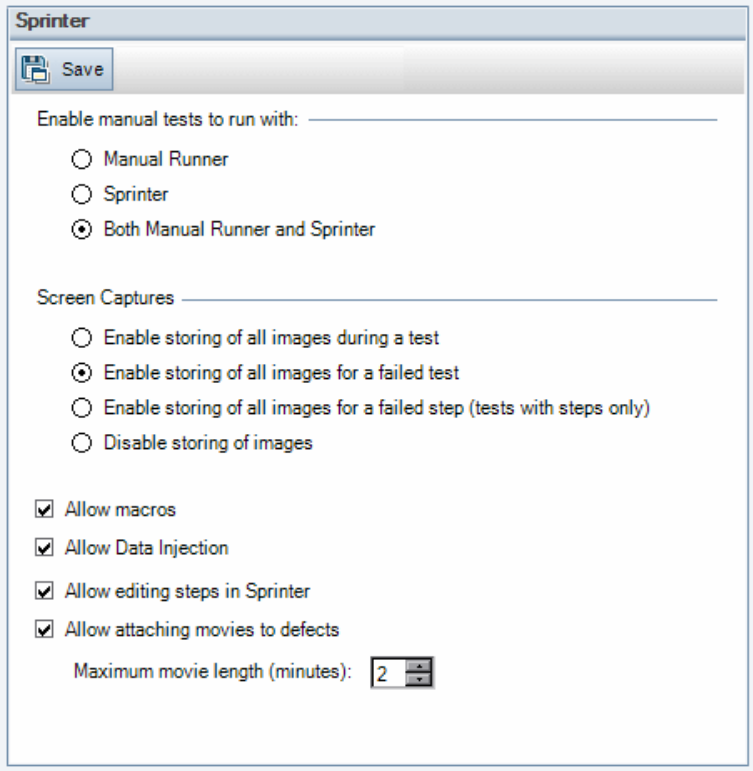
About Configuring Sprinter

As a project administrator, you can enable running tests manually in your project using Sprinter, using Manual Runner, or both. By default, running tests manually is enabled for both Sprinter and Manual Runner. You can also set additional options for working with Sprinter, such as default screen capture functionality.

For more information on running tests manually in ALM, refer to the *HP Application Lifecycle Management User Guide*. For full details on Sprinter, refer to the *HP Sprinter User Guide*, available from the HP Application Lifecycle Management Add-ins page (**Help > Add-ins**).

Sprinter Page

This page enables you to activate or deactivate Sprinter features. Deactivated features are visible in the Sprinter user interface, but are inactive.



To access	In Project Customization, in the left pane, click **Sprinter**.
Important information	The settings available in the Sprinter page control which features are enabled in Sprinter. Users still need the correct permissions within ALM that allow them to perform various functions. For example, suppose you select **Allow editing of steps in Sprinter**. The features that allow step editing will be enabled in Sprinter. However, users that do not have test editing permissions in ALM will still be unable to edit steps in a test.

See also	"About Configuring Sprinter" on previous page

User interface elements are described below:

UI Elements	Description
	Saves Sprinter customization changes.
Enable manual tests to run with	Options include: • **Manual Runner.** Enable manual tests to run with the Manual Runner only. • **Sprinter.** Enable manual tests to run with Sprinter only. • **Both Manual Runner and Sprinter. (Default)** Enable manual tests to run with Manual Runner or Sprinter.
Screen Captures	• Sprinter temporarily saves the screen captures of all the actions in your test. You indicate which images to capture during the run, in the **Run** pane in Sprinter's **Settings** dialog box. • The settings below control access to the corresponding images in the **Run** pane in Sprinter's **Settings** dialog box. They allow you to limit the amount of images saved during test runs: ▪ **Enable storing of all images during a test.** Enables the storing of all images during a run. **Note:** Storing all images during a test may cause a delay due to traffic and increase the storage needs on the ALM repository. ▪ **Enable storing of all images for a failed test. (Default)** Enables the storing of all images for a failed test during a run. ▪ **Enable storing of all images for a failed step (tests with steps only)**. Enables the storing of all images for a failed step during a run. ▪ **Disable storing of images.** Disables the storing of any images during a run. • Regardless of your selection, you can always attach screen captures to defects, both during your test, and at the end of your test from the test results.
Allow macros	Enable recording and running macros in Sprinter. Macros are available only for tests run in Sprinter using Power Mode.
Allow Data Injection	Enable Sprinter's Data Injection feature, allowing you to automatically enter data into fields in your test application. Data Injection is available only for tests run in Sprinter using Power Mode.

UI Elements	Description
Allow editing of steps in Sprinter	Enable adding, deleting, and modifying the name or description of steps in a test. If this option is cleared, you are still able to modify the actual results of a step and add screen captures to steps.
Allow attaching movies to defects	Enable attaching movies to defects when opening a defect from Sprinter's Tools sidebar, Workspace Tools sidebar, or from the test results. • **Maximum movie length (minutes).** The maximum length of the movie that you can attach to a defect. The movie length for each defect is set in Sprinter's Smart Defect Settings dialog box. You can attach a movie to a defect, of a size that is up to the length of time defined by this setting. The maximum allowable length of a movie is 10 minutes. **Note:** ▪ Increasing the length of movies that can be attached to defects may cause a delay in the time it takes to submit a defect to ALM and increase the storage needs of your ALM server. ▪ The length of the movie you can attach to a defect may be limited by the maximum size of an attachment you are allowed to attach to a defect in ALM.

Chapter 26: Generating Workflow Scripts

ALM provides script generators to enable you to perform commonly needed customizations on the Defects module dialog boxes.

For information on writing workflow scripts to customize the user interface and to control user actions in any ALM module, see "Workflow Customization at a Glance" on page 414.

Note: Workflow scripts run with the privileges of the logged-on user, not the privileges of the user who created the script. As a result, a script fails if it attempts any action or data change that the logged-on user is not privileged to perform, or if it accesses any restricted object that the user is not privileged to use. Therefore, a script developed by a privileged user should be tested with users that belong to the groups expected to trigger the script.

This chapter includes:

About Generating Workflow Scripts 403

Customizing Defects Module Field Lists 404

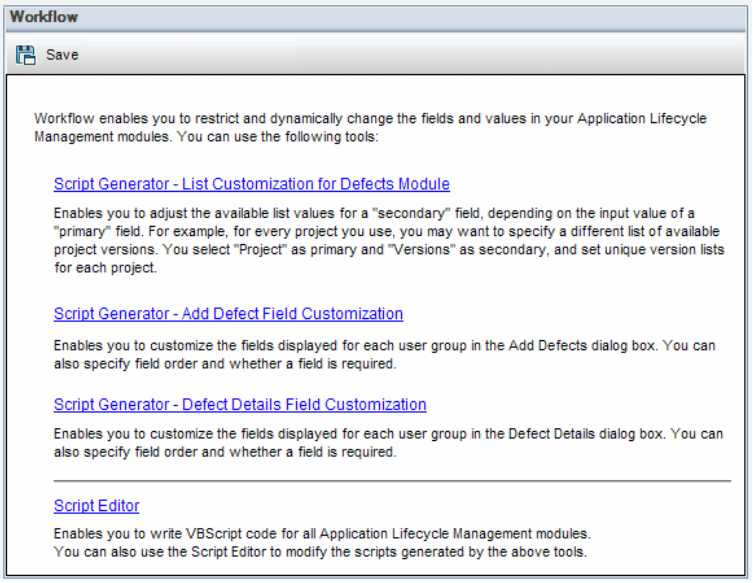
Customizing Defects Module Dialog Boxes 406

About Generating Workflow Scripts

The Workflow page provides links to script generators and a script editor. You can use the script generators to perform customizations on the input fields of the Defects module dialog boxes. You can use the script editor to create scripts to control the workflow in any ALM module.

Performance Center: The Workflow page is not supported in Lab Management.

To open the Workflow page, click the **Workflow** link in the Project Customization window.



The Workflow page contains the following links:

- **Script Generator - List Customization for Defects Module.** Enables you to customize the field lists displayed for fields on the dialog boxes and in the Defects Grid of the Defects module. For more information, see "Customizing Defects Module Field Lists" on next page.
- **Script Generator - Add Defect Field Customization.** Enables you to modify the appearance of the New Defect dialog box. For more information, see "Customizing Defects Module Dialog Boxes" on page 406.
- **Script Generator - Defect Details Field Customization.** Enables you to modify the appearance of the Defect Details dialog box. For more information, see "Customizing Defects Module Dialog Boxes" on page 406.
- **Script Editor.** Enables you to write VBScript code to customize the ALM workflow in any module. You place your code in the appropriate ALM event so that the script is triggered when the relevant user action takes place. You can also use the script editor to modify scripts created by the script generators. For more information, see "Workflow Customization at a Glance" on page 414.

Cross Project Customization

If you are working with a template project, in Project Customization, you use the **Workflow (Shared)** link to customize workflow. Workflow customization created in a template project is

applied to the linked projects when you apply the template customization. For more information on customizing workflow scripts as a part of cross process customization, see "The Script Editor" on page 417.

ALM Editions: Cross project customization is not available for for **Quality Center Enterprise Edition**.

Customizing Defects Module Field Lists

A field list is a list of values displayed in a drop-down list, from which the user can choose a value for the field.

You can specify that a different field list be used for a Defects module field, depending on the value of another field. For example, you can set the **Detected in Versions** list to change depending on the value in the **Project** field.

Note: This script generator can be used to customize field lists in the Defects module only.

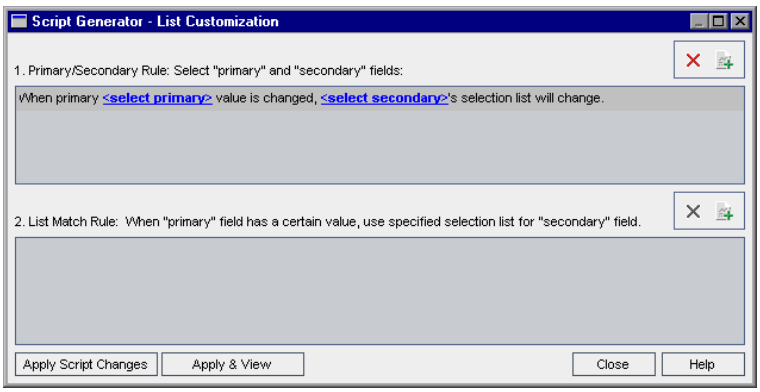
To customize a field list, you must define the following rules:

- **Primary/Secondary Rule.** Select the primary and secondary fields. When a primary field value is changed, the list of values in the secondary field changes automatically. For example, you could select **Project** as the primary field and **Detected in Versions** as the secondary field.
- **List Match Rule.** Select the list that you want to display in the secondary field for each value of the primary field.

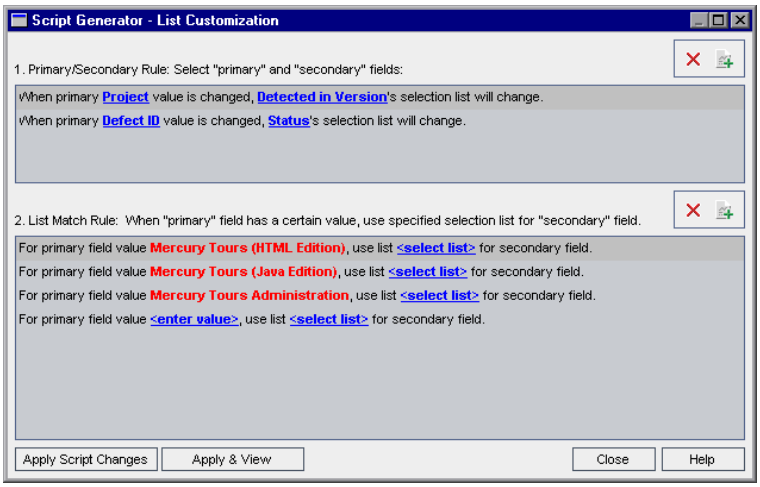
Note: When workflow customization has been used to change a list of values for a field that has transition rules defined, the field may only be modified in a way that satisfies both the workflow script and the transition rules. For more information, see "Setting Transition Rules" on page 257.

To customize a field list:

1. In the Project Customization window, in the left pane, click **Workflow**. The Workflow page opens.
2. Click the **Script Generator - List Customization for Defects Module** link. The Script Generator - List Customization dialog box opens.



3. Under **Primary/Secondary Rule**, select the primary field and the secondary field:
   - To set a rule, click **<select primary>** and select a field name. Click **<select secondary>** and select a field name.
   - To add a new rule, click the **Add Primary/Secondary Rule** button . Select field names for **<select primary>** and **<select secondary>**.
   - To delete a rule, select the rule and click the **Delete Primary/Secondary Rule** button . Click **Yes** to confirm.
4. Under **Primary/Secondary Rule**, select the primary/secondary rule for which you want to set list match rules.



5. Under **List Match Rule**, select the field list to be used in the secondary field for specific values entered into the primary field:
   - To set a rule for a defined primary field value, click **<select list>** and select a list name.
   - To set a rule for an undefined primary field value, click **<enter value>** and type a primary field value. Press **Enter**. Click **<select list>** and select a list name.
   - To add a new list match rule, click the **Add List Match Rule** button . Click **<enter value>** and type a primary field value. Click **<select list>** and select a list name.
   - To delete a list match rule, select the rule and click the **Delete List Match Rule** button . Click **Yes** to confirm.
6. To save your changes, do one of the following:
   - Click the **Apply Script Changes** button to save your changes and close the script generator.
   - Click the **Apply & View** button to save your changes and view the generated script in the Script Editor.

   If you use the Script Editor to modify a script that was created by a script generator, your modifications are overwritten the next time you run that script generator. It is recommended

that your name the generated script before you modify it. For more information on the Script Editor, see "Working with the Workflow Script Editor" on page 416.

Customizing Defects Module Dialog Boxes

You can modify the appearance of the New Defect and Defect Details dialog boxes by setting different fields to be visible for each user group. You can also sort the order in which the fields are displayed on the dialog box for each user group.

Note: This workflow customization example can be modified to display fields only to specific user groups. However, this example only covers changing the fields available in a dialog box. To set which fields are visible to specific user groups in a module grid, use the Groups and Permissions module in Customization. For task information, see "Hiding Data for a User Group" on page 260.

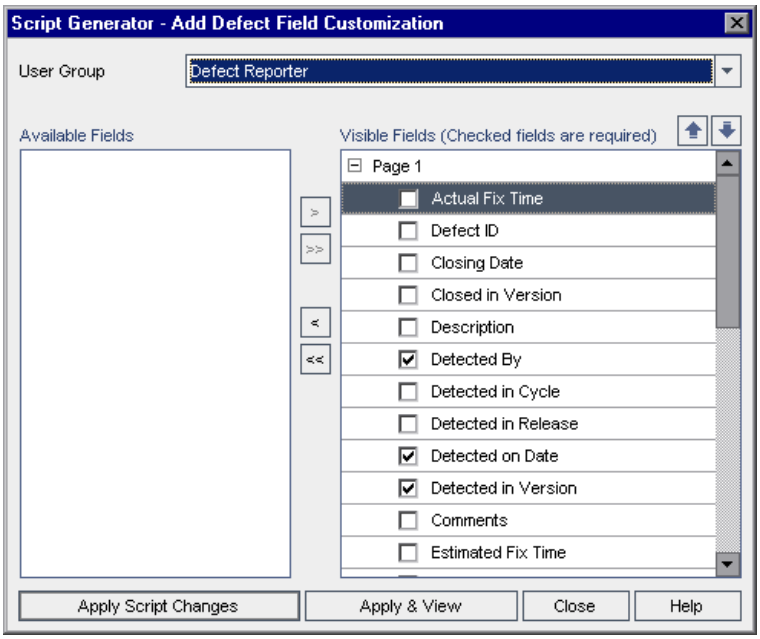
For example, you may want the **Assigned To** and **Priority** fields to appear only for a user that has developer privileges. Also, you can customize the **Assigned To** field so that it is displayed before the **Priority** field for this user group.

To perform a customization for all user groups, you can use the script editor to write a script. For more information, see "Example: Customizing a Defects Module Dialog Box" on page 476.

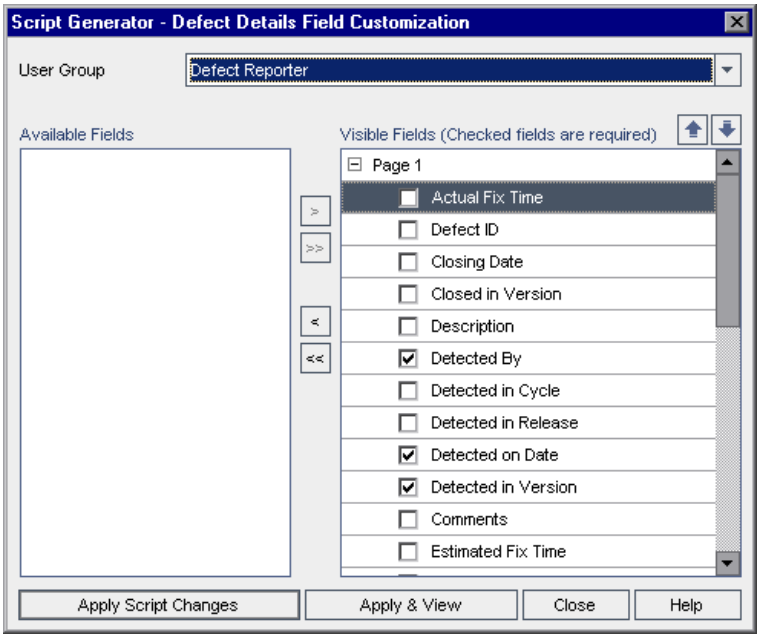
Note: These script generators can be used to customize dialog boxes in the Defects module only.

To customize Defects module dialog boxes by user group:

1. In the Project Customization window, in the left pane, click **Workflow**. The Workflow page opens.
2. To modify the appearance of the New Defect dialog box, click the **Script Generator - Add Defect Field Customization** link. The Script Generator - Add Defect Field Customization dialog box opens.



To modify the appearance of the Defect Details dialog box, click the **Script Generator - Defect Details Field Customization** link. The Script Generator - Defect Details Field Customization dialog box opens.



Available Fields contains the names of all the fields you can display. **Visible Fields** contains the names of the fields that can currently be seen by the selected user group, and their sorting priority.

3. From the **User Group** list, select the user group to which the customizations are to apply.
4. Choose field names and click the arrow buttons (> and <) to move a name between **Available Fields** and **Visible Fields**. Click the double arrow buttons (>> and <<) to move all the names from one list to the other. You can also drag the field names between lists.

5. In **Visible Fields**, to set a field as a required field, select the check box next to it. For a required field, a value is mandatory. Its title is displayed in red in the Add Defect or Defect Details dialog box.
6. You can set the order in which fields are displayed for the selected user group by using the up and down arrows . You can also drag the field names up or down.
7. You can set the Add Defect and Defect Details dialog boxes to include one or more input pages. By default, all fields are displayed on one page. Use the up and down arrows to move fields to the appropriate page.
8. To save your changes, do one of the following:
   - Click the **Apply Script Changes** button to save your changes and close the script generator.
   - Click the **Apply & View** button to save your changes and view the generated script in the Script Editor.

   If you use the Script Editor to modify a script that was created by a script generator, your modifications are overwritten the next time you run that script generator. It is recommended that you rename the generated script before you modify it. For more information on the Script Editor, see "Working with the Workflow Script Editor" on page 416.

Chapter 27: Managing Analysis Menus

The Analysis View module's Analysis Menu tab enables you to manage the behavior of graphs and project reports that are generated from within the following modules: Requirements, Test Plan, Test Lab, Defects, and Business Components.

The tab lists all graphs and project reports grouped per module. You can perform the following tasks for each individual module:

- Add or remove graphs or project reports
- Configure graphs or project reports
- View graph or project report details
- Generate graphs or project reports
- Preview project reports

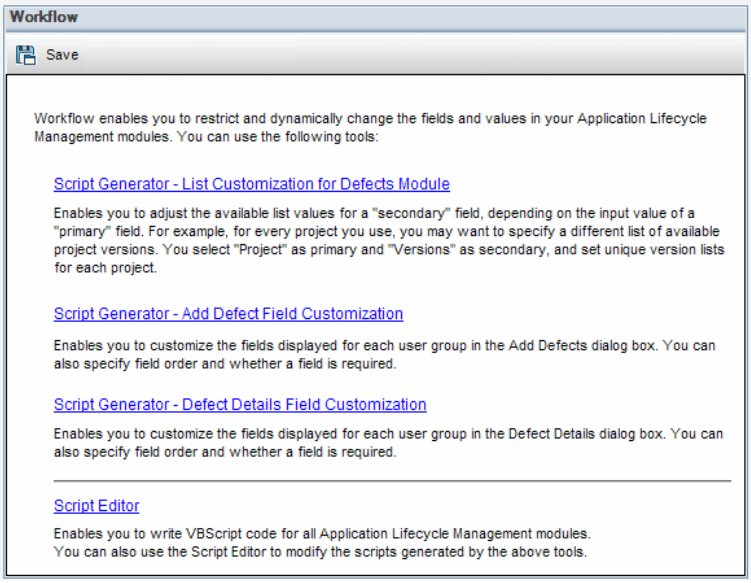
Part 3: Workflow Customization

Chapter 28: Workflow Customization at a Glance

You can write workflow scripts to customize the HP Application Lifecycle Management (ALM) user interface, and to control the actions that users can perform.

To customize workflow:

1. In the Project Customization window, in the left pane, click **Workflow**. The Workflow page opens.



2. To customize a Defects module dialog box, click the appropriate **Script Generator** link on the Workflow page. You need not be familiar with VBScript, or with ALM events and objects to use this feature. For more information, see "Generating Workflow Scripts" on page 402.

3. To write or modify scripts by entering code into the appropriate event procedures, open the Script Editor. To create workflow scripts, you must be familiar with VBScript. You can open the Script Editor either from a script generator or directly:

   - To write a script that is similar to a script created by a script generator, click the relevant **Script Generator** link and set the customization you want to perform. Click the **Apply & View** button on the script generator dialog box. The Script Editor opens to display the scripts that were generated.
   - To create your own scripts, click the **Script Editor** link. The Script Editor opens to display a Scripts Tree that lists the existing event procedures.

   For more information on the Script Editor, see "Working with the Workflow Script Editor" on page 416.

4. Decide which ALM event should trigger your script. You must place your code in the procedure of the appropriate module and event so that it is invoked for the relevant user action. For more information, see "Workflow Event Reference" on page 428.

5. Decide which ALM objects your script must access. Your script performs customizations based on information obtained from the relevant objects. You customize the workflow by using the methods and properties of the objects. For more information, see "Workflow Object and Property Reference" on page 458.
6. Examine the sample scripts to find one that can be adapted for your use. Sample scripts are provided in this guide and in the HP Self-solve knowledge base. Scripts generated by the workflow script generators can also be used as a basis for your scripts.
   - For examples of common customizations that can be performed by using workflow scripts, see "Workflow Examples and Best Practices" on page 468.
   - For an index to knowledge base articles that provide examples of workflow scripts, see HP Software Self-solve knowledge base article KM183671 (http://h20230.www2.hp.com/selfsolve/document/KM183671).

Tip: Turning off versioning for relevant fields enables workflow scripts and OTA scripts to run without requiring entities to be checked in.

Chapter 29: Working with the Workflow Script Editor

You can use the Script Editor to create workflow scripts to customize the user interface, and to control user actions.

This chapter includes:

About Working with the Workflow Script Editor 417

The Script Editor 417

Creating a Workflow Script 420

Adding a Button to a Toolbar 422

Setting the Properties of the Script Editor 424

About Working with the Workflow Script Editor

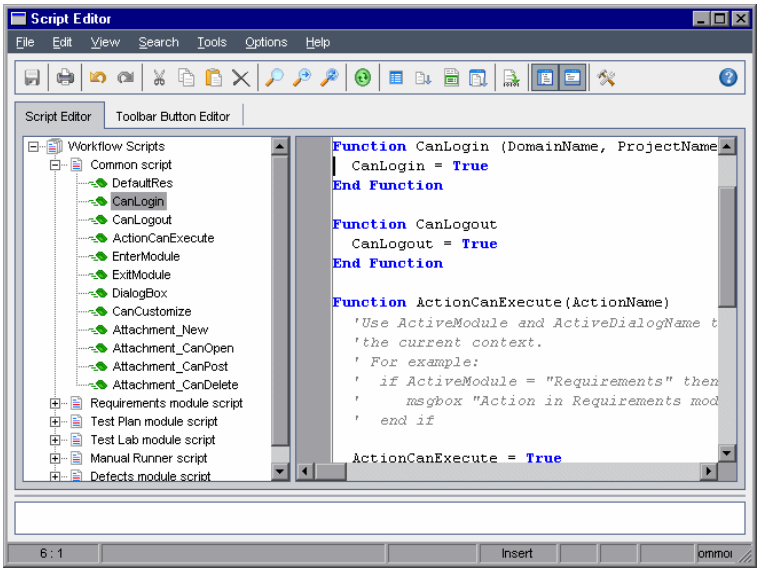
You can use the Script Editor to create workflow scripts and to add a toolbar button to the window of an HP Application Lifecycle Management (ALM) module.

The Script Editor dialog box contains two tabs:

- **Script Editor tab.** You use the Script Editor tab to create and edit workflow scripts. The Script Editor assists you in placing your code in the appropriate ALM event procedure. For more information on using the Script Editor, see "Creating a Workflow Script" on page 420.
- **Toolbar Button Editor tab.** You use the Toolbar Button Editor tab to add a toolbar button to the window of an ALM module. For more information, see "Adding a Button to a Toolbar" on page 422.

The Script Editor

You can use the Script Editor to modify scripts that have been generated by a script generator, or to create a user-defined workflow script. For information on opening the Script Editor, see "Workflow Customization at a Glance" on page 414.



The Script Editor tab contains the following elements:

- **Script Editor toolbar.** Contains buttons used when creating scripts. For more information, see "Understanding the Script Editor Commands" on page 419.
- **Scripts Tree.** Lists the event procedures to which you can add code. The event procedures are grouped by the module in which they are triggered. For more information, see "Workflow Event Reference" on page 428.
- **Scripts pane.** Displays the code of the selected event procedure. To create or modify a script, you add VBScript code to the event procedure. For more information, see "Creating a Workflow Script" on page 420.
- **Messages pane.** Displays any syntax errors encountered when you save or validate a script.

Cross Project Customization

ALM Editions: Cross project customization is not available for **Quality Center Enterprise Edition**.

If you are working in a template or linked project, the Scripts Tree displays two sections under Workflow Scripts:

- **Template Scripts (Shared).** Workflow scripts listed in this section are the scripts applied from the template to the linked project. For more information on applying template customization, see "Applying Template Customization to Linked Projects" on page 337.

  If you are working with a linked project, this section is displayed only when there are template scripts defined in the template. The template scripts cannot be edited in the linked project. If you remove a project from a template, the template scripts remain in the project and are editable.

- **Project Scripts.** Workflow scripts listed in this section apply only to the template or linked project in which you are working. Scripts in this section of a template are not applied to linked projects.

  When running workflow scripts for a linked project, ALM combines template scripts and project scripts into one script. Duplicate variables or functions in the template scripts and projects scripts can cause conflicts.

Additional considerations when working in a template project:

- Scripts generated by one of the script generators are created under **Template Scripts (Shared)**.
- ALM adds the prefix **Template_** to events in template scripts. By default, ALM triggers template event procedures. The project event procedure is triggered if the template event procedure does not exist, or if you instruct the template event procedure to call the project event procedure.

  Each template event includes a commented call to the parallel project event. For example, the Template_Bug_New event in the template script is displayed as follows:

```
Sub Template_Bug_New
    On Error Resume Next
    `call Bug_New
    On Error Go To 0
End Sub
```

  To instruct the template script to call the project event, remove the comment marker to activate the call to the project event, as follows:

```
Sub Template_Bug_New
    On Error Resume Next
    call Bug_New
    On Error Go To 0
End Sub
```

Understanding the Script Editor Commands

The Script Editor toolbar, menu bar, and right-click menu contain the following buttons and menu commands:

UI Element	Description
	Save. Saves the changes made to scripts in the selected module.
	Print. Prints the displayed script.
	Undo. Reverses the last command or deletes the last entry you typed.
	Redo. Reverses the action of your last **Undo** command.
	Cut. Removes the selected text and places it on the Clipboard.
	Copy. Copies the selected text to the Clipboard.
	Paste. Inserts the contents of the Clipboard at the insertion point.
	Delete. Deletes the selected text.
	Find. Searches for specified text in the scripts of the selected module.
	Find Next. Finds the next occurrence of the text specified in the Find Text dialog box.
	Replace. Replaces the specified text with replacement text.
	Synchronize Tree with Script. Refreshes the Scripts Tree to reflect procedures you have added, deleted or renamed.
	Field Names. Displays a list of field names in the project that you can insert into your script.
	Code Complete. Displays a list of objects, properties, methods, or field names that you can insert into your script.
	Code Template. Displays a list of templates for commonly used VBScript statements that you can insert into your script.
	List Value. Opens the Select Value From List dialog box, to enable you to choose an item from a project list.

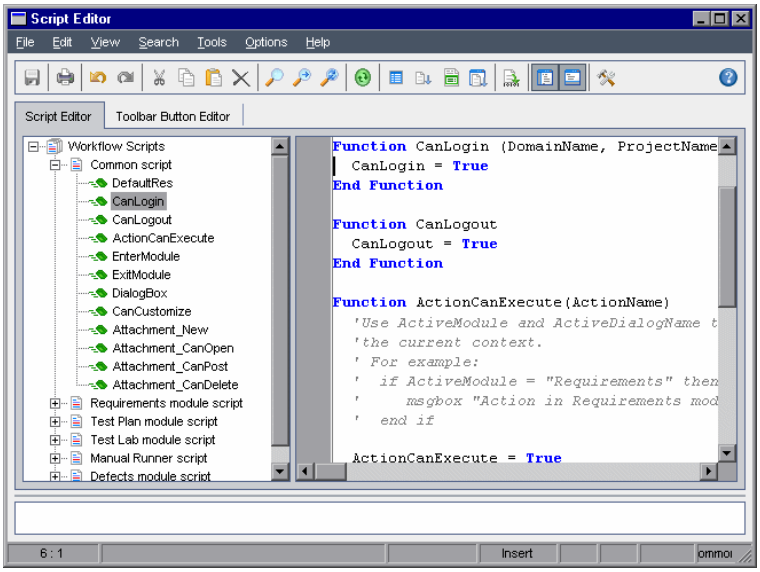
UI Element	Description
	Syntax Check. Validates the syntax of your script and displays any messages in the Messages pane.
	Show/Hide Scripts Tree. Displays or hides the Scripts Tree. If you have opened the Script Editor from a script generator, this is not available.
	Show/Hide Messages Pane. Displays or hides the Messages pane.
	Properties. Opens the Properties dialog box, enabling you to change the properties of the Script Editor. For more information, see "Setting the Properties of the Script Editor" on page 424.
Save All	To save script changes in all modules, choose **File > Save All**.
Revert to Saved	To return to a saved version of a module, select a changed module and choose **File > Revert to Saved**.
Select All	To select all text in the scripts pane, choose **Edit > Select All**.
Expand All	To expand all nodes in the Scripts Tree, choose **View > Expand All**.
Collapse All	To collapse all nodes in the Scripts Tree, choose **View > Collapse All**.
Go to Line Number	To jump to a specific line in the Script Editor, choose **Search > Go to Line Number**.
Clear Messages	To clear syntax messages displayed in the messages pane, choose **Tools > Clear Messages**.
Sort Field Names by Field Labels	When you choose the **Field Names** option, the Script Editor sorts the list by the field name used in the ALM database table (for example, **BG_BUG_ID**). To sort the fields by the field label (for example, Defect ID) right-click the script pane and choose **Sort Field Names by Field Labels**.
VBScript Home Page	To get help for the VBScript language, choose **Help > VBScript Home Page**.

Creating a Workflow Script

You use the Script Editor to add VBScript code to an ALM event procedure, or to create user-defined procedures that can be called from an ALM event procedure.

To create a workflow script:

1. In the Workflow window, click the **Script Editor** link. The Script Editor opens.



   For more information on the Script Editor window, see "The Script Editor" on page 417.

2. In the Scripts Tree, select the node of the module for which you need to customize the workflow.

   The Scripts Tree contains the **Common script** node in addition to the nodes for specific modules. When you create user-defined procedures that must be accessible from several modules, place them under the **Common script** node. To declare a global variable that can be used across all modules, declare the variable under the **Common script** node, outside of any function.

3. Expand the node and select the event procedure to which you need to add code, depending on when you want your code to be triggered. The existing script for this event procedure is displayed in the Scripts pane.

   For a description of ALM event procedures, see "Workflow Event Reference" on page 428.

4. Add your VBScript code to the script.

   Note: A red indicator ● next to a module name in the Scripts Tree indicates that there are unsaved script changes in that module.

5. To use the code complete feature instead of typing in the names of ALM objects, properties, methods, and fields, place the insertion point at the location where you want to insert an object name and click the **Code Complete** button. For information about ALM objects, see "Workflow Object and Property Reference" on page 458.

6. To use the code template feature instead of typing in commonly used VBScript statements, place the insertion point where you want to insert the code and click the **Code Template** button. Choose one of the following items from the code template list:

Template	Code Added to Script
FVal: Fields value access	`Fields.Field("").Value`
List: QualityCenter list access	`Lists.List()`
IfAct: Action "switch" If Block	`If ActionName = "" Then` `End IF`
Act: Actions access	`Actions.Action("")`

Func: Function template	`Function`
	`  On Error Resume Next`
	`  On Error GoTo 0`
	`End Function`

Sub: Sub Template	`Sub`
	`  On Error Resume Next`
	`  On Error GoTo 0`
	`End Sub`

Err: Error Handler	`On Error Resume Next`

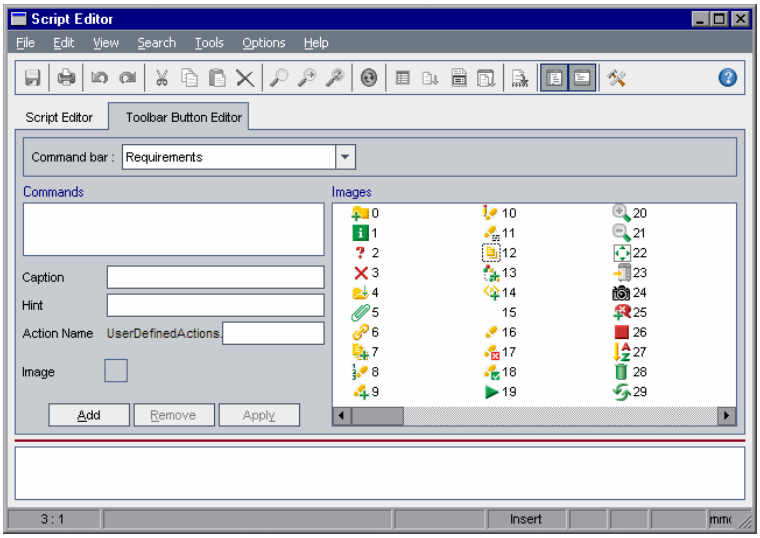
7. To insert an item from a field list defined in the project, place the insertion point at the location where you want to add the item. Click the **List Value** button. In the **Lists** box of the Select Value From List dialog box, choose the name of the list. In the **List Items** box, select the list value.
8. To insert an ALM field name, place the insertion point at the location where you want to add the field name. Click the **Field Names** button. Select a name from the list of system and user-defined fields in the ALM project.
9. To validate the syntax of the script, click **Syntax Check**. Any messages are displayed in the Messages pane.
10. Click the **Save** button to save the script.
11. Close the Script Editor.

Adding a Button to a Toolbar

You can use the Toolbar Button Editor to define a toolbar button to be displayed on the window of an ALM module, or on the Manual Runner dialog box.

To add a button to a toolbar:

1. In the Script Editor, click the **Toolbar Button Editor** tab.



2. From the **Command bar** list, select the toolbar to which you want to add a button:

Option	Toolbar Location
Requirements	Requirements module window.
TestPlan	Test Plan module window.
TestLab	Test Lab module window.
ManualRun	Manual Runner dialog box.
Component	Business Components module window. This option depends on your ALM license.
Defects	Defects module window.
Management	Management module window. This module includes Releases and Libraries.
Resources	Test Resources module window.
Dashboard	Dashboard module window.

3. Click **Add**. A default command name for the button is added to the **Commands** list.
4. In the **Caption** box, type a new command name for the button, or use the default name.
5. In the **Hint** box, type a tooltip for the button.
6. In the **Action Name** box, type a new action name for the button, or use the default name.
7. Under **Images**, select an icon for the button.
8. Click **Apply** to apply your changes.

9. To delete a button that you have created, select its command name in the Commands list, and click **Remove**.

10. Click the **Save** button to save the new button definition.

11. Click the **Script Editor** tab.

12. In the Scripts Tree of the Script Editor, select the **ActionCanExecute** event procedure located in the common script section.

13. In the procedure displayed in the scripts pane of the Script Editor, add statements to be performed if the user initiates an action with the action name you defined for the button. Set the return value to `True` or `False`.

    For example, the following code opens a message box when the user clicks the Requirements_Action1 button on the tool bar of the Requirements module:

```
Function ActionCanExecute(ActionName)
    On Error Resume Next
    ActionCanExecute = True
        If ActionName = "UserDefinedActions.Requirements_Action1"
Then
            MsgBox "You clicked the Action1 button."
        End If
    On Error GoTo 0
End Function
```

    For more information, see "Example: Adding Button Functionality" on page 484.

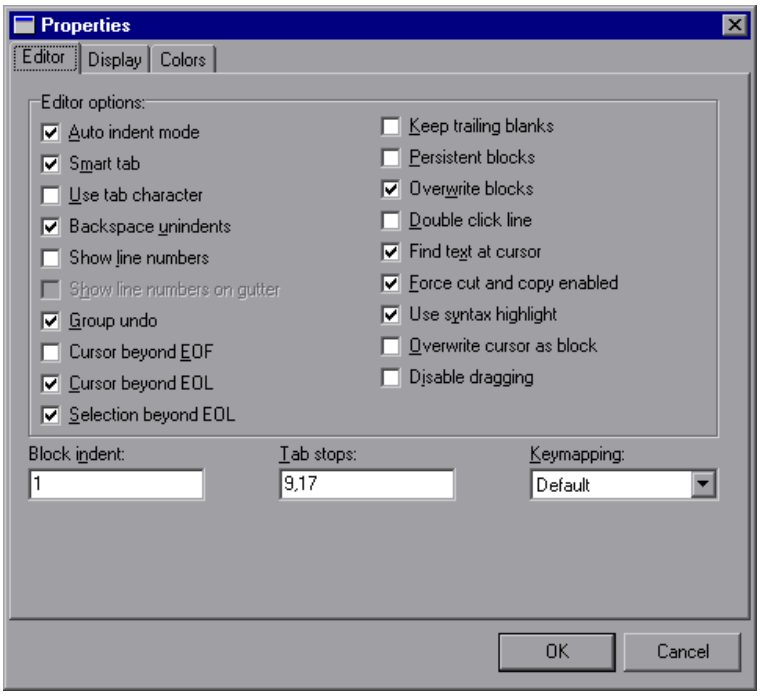
14. Click the **Save** button to save the script.

Setting the Properties of the Script Editor

You can customize the behavior of the Script Editor.

To set the properties of the Script Editor:

1. In the Script Editor, click the **Properties** button or choose **Options > Editor Properties**. The Properties dialog box opens.



2. In the **Editor** tab, you can set the following options:

Option	Description
Auto indent mode	Places the cursor under the first non-blank character of the preceding non-blank line when you press **Enter**.
Smart tab	Tabs to the first non-blank character in the preceding non-blank line. If **Use tab character** is selected, this option is cleared.
Use tab character	Inserts a tab character. If cleared, inserts space characters. If **Smart tab** is selected, this option is cleared.
Backspace unindents	Aligns the insertion point to the previous indentation level when you press **Backspace**, if the cursor is on the first non-blank character of a line.
Show line numbers	Displays line numbers. If this option is selected, **Show line numbers on gutter** is enabled.
Show line numbers on gutter	Displays line numbers in the gutter instead of in the left margin. If **Show line numbers** is selected, this option is enabled.
Group undo	Reverses your last editing command and any subsequent editing commands of the same type, if you press **Alt+Backspace** or choose **Edit > Undo**.
Cursor beyond EOF	Enables you to place the insertion point after the last line of code.

Option	Description
Cursor beyond EOL	Enables you to position the cursor after the end of the line.
Selection beyond EOL	Enables you to select characters beyond the end of the line.
Keep trailing blanks	Keeps any blank spaces you have at the end of a line.
Persistent blocks	Keeps marked blocks selected, even when the cursor is moved using the arrow keys, until a new block is selected.
Overwrite blocks	Replaces a marked block of text with new text. If **Persistent Blocks** is also selected, text you enter is appended following the currently selected block.
Double click line	Highlights the line when you double-click any character in the line. If disabled, only the selected word is highlighted.
Find text at cursor	Places the text at the cursor into the **Text To Find** list box in the Find Text dialog box when you choose **Search > Find**.
Force cut and copy enabled	Enables the **Cut** and **Copy** commands, even when there is no text selected.
Use syntax highlight	Displays script elements according to colors and attributes defined in the **Display** tab and **Colors** tab.
Overwrite cursor as block	Controls the appearance of the caret when using the Overwrite mode.
Disable dragging	Disables dragging and dropping text.
Block indent	Specifies the number of spaces to indent a marked block.
Tab stops	Specifies the locations to which the cursor moves when you press **Tab**.
Keymapping	Sets the keyboard mappings in the Script Editor. Supports the following keyboard mappings: Default, Classic, Brief, Epsilon, and Visual Studio.

3. In the **Display** tab, you can set the following options:

Option	Description
Editor gutter	Enables you to set the visibility, width, color, and style of the gutter.
Editor margin	Enables you to set the visibility, width, color, style, and position of the right margin.

Option	Description
Use mono font	Displays only monospaced screen fonts, such as Courier, in the Editor font box.
Editor font	Lists the available text fonts.
Editor color	Lists the available background colors.
Size	Lists font sizes.
Use Read-Only Color	Enables you to select a color for displaying read-only text from the Read-Only Color box.
Draw Special Symbols	Sets special characters for displaying end-of-file, end-of-line, space, and tab characters.

4. In the **Colors** tab, you can set the following options:

Option	Description
Color SpeedSetting	Enables you to configure the Script Editor display using predefined color combinations.
Element	Specifies syntax highlighting for a particular code element.
Foreground color	Sets the foreground color for the selected code element.
Background color	Sets the background color for the selected code element.
Use defaults for	Displays the code element using default system colors for the foreground, background, or both.
Text attributes	Specifies format attributes for the code element.
Open	Loads a color scheme from your computer.
Save	Saves a color scheme to your computer.

Chapter 30: Workflow Event Reference

You can write workflow scripts to customize the actions that HP Application Lifecycle Management (ALM) users can perform, and the fields that are available to users in dialog boxes. To write a workflow script, you add VBScript code to event procedures that are triggered by user actions.

This chapter includes:

About ALM Events 429

Naming Conventions for ALM Event Procedures 429

Reference for ALM Events 431

About ALM Events

During an ALM user session, as the user initiates various actions, ALM triggers event procedures. You can place code in these procedures to customize the execution of the associated user actions.

The Script Editor lists the event procedures for each ALM module, and allows you to add your code to the appropriate procedure. For more information, see "Working with the Workflow Script Editor" on page 416.

The code you add to the event procedures can access ALM objects. For more information, see "Workflow Object and Property Reference" on page 458.

Event procedures can be functions or subroutines:

- **Event functions.** These procedures are triggered by ALM to check whether the user's action should be performed. You can place code in these functions to determine whether ALM may execute the user's request. If your code returns a value of `False`, ALM does not proceed with the action.

  For example, when a user clicks the **Submit** button on the Add Defect dialog box, ALM invokes the function `Bug_CanPost` before posting the defect to the database on the server. You can add code to the `Bug_CanPost` function to control whether ALM posts the defect. For example, you can ensure that a user cannot reject a defect without adding a comment. For example, see "Example: Object Validation" on page 481.

- **Event subroutines.** These procedures are triggered to perform actions when an event takes place.

  For example, when a user opens the Add Defect dialog box, ALM invokes the subroutine `Bug_New`. You can add code to the `Bug_New` subroutine to perform actions that should be performed when a user opens the dialog box. For example, you can change the value of the **Detection Mode** field to `BTW` if the user is not in the `QA Tester` user group. For example, see "Example: Changing a Field Based on the User Group" on page 480.

Version Control: After enabling version control for a project, you should review all its workflow scripts and make adjustments for each checked in entity. This includes the following entities: **Req**, **Test**, **Resource**, and **Component**. For each checked in entity that includes a **Post** function in its script, you must modify the script. To modify, add a **Checkout** function before every **Post** function. Making this modification prevents the Check Out dialog box from opening each time a call to a **Post** function is made. For more information about the Post and Checkout functions, see the *HP ALM Open Test Architecture Reference*.

For more information on version control, refer to the *HP Application Lifecycle Management User Guide*.

Naming Conventions for ALM Event Procedures

The naming convention for an event procedure is as follows:

<*entity*>_<*event*>

See "Entity" on next page and "Event" on page 431 for more details.

Note:

- For backwards compatibility, the previous naming convention, including the module name, is still supported. However, we recommend you use the new naming conventions instead.
- Some event procedure names do not include an entity name. For example, the `GetDetailsPageName` event name does not include an entity name.

Entity

An **Entity** can be one of the following:

Entity	Description
Release	Release data
Release Folder	Release folder data
Cycle	Release cycle data
Library	Library data
Library Folder	Library folder data
Baseline	Baseline data
Req	Requirement data
Test	Test data
DesignStep	Design step data
Resource	Test resource data
Resource Folder	Test resource folder data
TestSet	Test set data
TestSetTests	Test instance data
Run	Test run data
Bug	Defect data
Step	Test run step data
AnalysisItem	Reports and graphs data
AnalysisItemFolder	Reports and graphs folder data
DashboardFolder	Dashboard folder data
DashboardPage	Dashboard page data

Entity	Description
Component	Business component data
ComponentStep	Business component step data
ComponentFolder	Business component folder data
BusinessModel	Business model data
BusinessModelActivity	Business model activity data
BusinessModelPath	Business model path data
BusinessModelFolder	Business model folder data

Event

The **Event** can be either a function name or a subroutine name. The event names are listed in "Reference for ALM Events" below.

Reference for ALM Events

This section contains an alphabetical reference of the ALM event functions and subroutines. It includes the event name, description, syntax, type (Function or Sub), the value returned by a function, and the entities for which the event procedure is available.

For information on the naming conventions for event procedures, see "Naming Conventions for ALM Event Procedures" on page 429.

The following event functions are available:

Function Name	When the Function is Triggered
"ActionCanExecute" on page 433	before performing a user action
"Attachment_CanDelete" on page 435	before deleting an attachment
"Attachment_CanOpen" on page 436	before opening an attachment
"Attachment_CanPost" on page 436	before updating an attachment
"CanAddTests" on page 437	before adding tests to a test set
"CanCustomize" on page 437	before opening Customization window
"CanDelete" on page 438	before deleting an object from the server
"CanLogin" on page 441	before a user logs in to the project
"CanLogout" on page 441	before a user logs out of the project
"CanPost" on page 441	before posting an object to the server

Function Name	When the Function is Triggered
"CanRemoveTests" on page 443	before removing tests from a test set
"CanAddComponentsToTest" on page 437	before adding business components to a test of type Flow or Business-Process
"CanAddFlowsToTest" on page 437	before adding flows to a test of type Business-Process
"CanRemoveComponentsFromTest" on page 443	before removing business components from a test of type Flow or Business-Process
"CanRemoveFlowsFromTest" on page 443	before removing flows from a test of type Business-Process
"CanDeleteGroupsFromTest" on page 441	before deleting groups from a test of type Flow or Business-Process
"CanReImportModels" on page 443	before importing business models
"DefaultRes" on page 444	before resetting project defaults
"FieldCanChange" on page 445	before changing a field value
"GetDetailsPageName" on page 449	before displaying Defect Details dialog box
"GetNewBugPageName" on page 449	before displaying Add Defect dialog box (for backward compatibility)
"GetNewReqPageName" on page 450	before displaying New Requirement dialog box (for backward compatibility)
"GetReqDetailsPageName" on page 450	before displaying Requirement Details dialog box (for backward compatibility)

The following event subroutines are available:

Subroutine Name	When the Subroutine is Triggered
"AddComponentToTest" on page 434	a component has been added to a test of type Flow or Business-Process
"AfterPost" on page 434	an object has been posted to the server
"Attachment_New" on page 436	an attachment is added
"DialogBox" on page 444	a dialog box is opened or closed
"EnterModule" on page 444	user switches modules
"ExitModule" on page 445	user exits a module
"FieldChange" on page 447	a field value changes

Subroutine Name	When the Subroutine is Triggered
"MoveTo" on page 451	user changes focus
"MoveToComponentFolder" on page 453	user moves to the specified component folder in the business component tree (for backward compatibility)
"MoveToFolder" on page 453	user clicks a folder in the test sets tree (for backward compatibility)
"MoveToSubject" on page 453	user clicks a subject in the test plan tree (for backward compatibility)
"New" on page 454	an object is added
"RemoveComponentFromTest" on page 455	user removes a component from a test of type Flow or Business-Process
"RunTests" on page 456	user clicks **Run** in the Test Lab module (provided that Sprinter is not installed and none of the tests is automated)
"RunTests_Sprinter" on page 456	user clicks **Run** in the Test Lab module (provided that Sprinter is installed and at least one test is automated)
"RunTestSet" on page 456	user clicks **RunTest Set** in the Test Lab module
"RunTestsManually" on page 457	user clicks **Run** > **Run Manually** in the Test Lab module

ActionCanExecute

This event is triggered before ALM performs an action that has been initiated by the user, to check whether the action can be executed.

You can add code to this event procedure to perform actions when the user has initiated a particular action, or to prevent the action from being executed in specific cases. For example, see "Example: Controlling User Permissions" on page 484.

Syntax	**ActionCanExecute(ActionName)** where **ActionName** is the action that the user has initiated. Actions are in the format **context.action**. **Note:** The previous format for this event is supported for purposes of backward compatibility. We recommend you use ActionCanExecute instead. User-defined actions start with the prefix `UserDefinedActions`.
Type	Function
Returns	True or False
Availability	`ActionCanExecute` (all modules)

Tip: To obtain the name of an action, see the sample code on "Action Object" on page 460.

AddComponentToTest

This event is triggered when the user adds a component to a test of type Flow or Business-Process in the Test Script tab.

Version Control: Changing components checked in or checked out by another user, using the AddComponentToTest event, is not supported.

Syntax	**AddComponentToTest**
Type	Sub
Availability	`AddComponentToTest`

AfterPost

This event is triggered after an object has been posted to the server.

Project fields should not be changed after they have been posted, because then the new value is not stored in the database.

Syntax	**<entity>_AfterPost**
Type	Sub

Availability	• AnalysisItem_AfterPost • AnalysisItemFolder_AfterPost • Baseline_AfterPost • Bug_AfterPost • BusinessModel_AfterPost • BusinessModelFolder_AfterPost • BusinessModelPath_AfterPost • Component_AfterPost • ComponentFolder_AfterPost • Cycle_AfterPost • DashboardFolder_AfterPost • DashboardPage_AfterPost • Library_AfterPost • LibraryFolder_AfterPost • Release_AfterPost • ReleaseFolder_AfterPost • Req_AfterPost • Resource_AfterPost • ResourceFolder_AfterPost • Run_AfterPost • Step_AfterPost • Test_AfterPost • TestConfiguration_AfterPost • TestFolder_AfterPost • TestSet_AfterPost • TestSetFolder_AfterPost

Attachment_CanDelete

This event is triggered before ALM deletes an attachment from the server, to check whether that attachment can be deleted.

Syntax	**Attachment_CanDelete(Attachment)** where **Attachment** is the **IAttachment** interface. For more information, refer to the *HP ALM Open Test Architecture API Reference*.

Type	Function
Returns	True or False
Availability	`Attachment_CanDelete` (all modules)

Attachment_CanOpen

This event is triggered before ALM opens an attachment from the server, to check whether the attachment can be opened.

Syntax	**Attachment_CanOpen(Attachment)** where **Attachment** is the **IAttachment** interface. For more information, refer to the *HP ALM Open Test Architecture API Reference*.
Type	Function
Returns	True or False
Availability	`Attachment_CanOpen` (all modules)

Attachment_CanPost

This event is triggered before ALM updates an existing attachment on the server, to check whether the attachment can be updated.

Syntax	**Attachment_CanPost(Attachment)** where **Attachment** is the **IAttachment** interface. For more information, refer to the *HP ALM Open Test Architecture API Reference*.
Type	Function
Returns	True or False
Availability	`Attachment_CanPost` (all modules)

Attachment_New

This event is triggered when an attachment is added to ALM.

Syntax	**Attachment_New(Attachment)** where **Attachment** is the **IAttachment** interface. For more information, refer to the *HP ALM Open Test Architecture API Reference*.
Type	Sub
Availability	`Attachment_New` (all modules)

CanAddComponentsToTest

This event is triggered before ALM adds business components to a test of type Flow or Business-Process, to check whether the specified components can be added.

Syntax	**CanAddComponentsToTest(Components)** where **Components** is an array of component IDs.
Type	Function
Returns	True or False
Availability	`CanAddComponentsToTest`

CanAddFlowsToTest

This event is triggered before ALM adds flows to a test of type Business-Process, to check whether the specified flows can be added.

Syntax	**CanAddFlowsToTest(Flows)** where **Flows** is an array of flow IDs.
Type	Function
Returns	True or False
Availability	`CanAddFlowstoTest`

CanAddTests

This event is triggered before ALM adds tests to a test set, to check whether the specified tests can be added.

Syntax	**<entity>_CanAddTests(Tests)** where **Tests** is an array of Test IDs.
Type	Function
Returns	True or False
Availability	`TestSet_CanAddTests`

CanCustomize

This event is triggered when a user attempts to open the Customization window, to check whether the specified user can customize the specified project.

Syntax	**CanCustomize(DomainName, ProjectName, UserName)** where **DomainName** is the domain name, **ProjectName** is the project name, and **UserName** is the user name.
Type	Function
Returns	True or False
Availability	`CanCustomize` (all modules)

CanDelete

This event is triggered before ALM deletes an object from the server, to check if the object can be deleted.

Syntax	**<entity>_CanDelete(Entity)**
Type	Function
Returns	True or False

Availability	• AnalysisItem_CanDelete • AnalysisItemFolder_CanDelete • Baseline_CanDelete • Bug_CanDelete • BusinessModel_CanDelete • BusinessModelFolder_CanDelete • BusinessModelPath_CanDelete • Component_CanDelete • ComponentFolder_CanDelete • Cycle_CanDelete • DashboardFolder_CanDelete • DashboardPage_CanDelete • Library_CanDelete • LibraryFolder_CanDelete • Release_CanDelete • ReleaseFolder_CanDelete • Req_CanDelete • Resource_CanDelete • ResourceFolder_CanDelete • Test_CanDelete • TestConfiguration_CanDelete • TestFolder_CanDelete • TestSet_CanDelete • TestSetFolder_CanDelete

Additional Syntax for Backward Compatibility

For purposes of backward compatibility, the following syntaxes are also available for certain objects. However, we recommend you use CanDelete instead.

- The syntax for tests or test subject folders:

Syntax	**Test_CanDelete(Entity, IsTest)** where: ▪ **Entity** is the test or subject folder. ▪ If **IsTest** is `True`, **Entity** refers to an ITest object. If **IsTest** is `False`, **Entity** refers to an ISubjectNode object. For more information on ITest and ISubjectNode, refer to the *HP ALM Open Test Architecture API Reference*.
Type	Function
Returns	True or False
Availability	`Test_CanDelete`

- The syntax for test sets or test set folders:

Syntax	**TestSet_CanDelete(Entity, IsTestSet)** where: ▪ **Entity** is the test set or test set folder. ▪ If **IsTestSet** is `True`, **Entity** refers to an ITestSet object. If **IsTestSet** is `False`, **Entity** refers to an ITestSetFolder object. For more information on ITestSet and ITestSetFolder, refer to the *HP ALM Open Test Architecture API Reference*.
Type	Function
Returns	True or False
Availability	`TestSet_CanDelete`

- The syntax for business components or business component folders:

Syntax	**Component_CanDelete(Entity, IsComponent)** where: ▪ **Entity** is the component or component folder. ▪ If **IsComponent** is `True`, **Entity** refers to an IComponent object. If **IsComponent** is `False`, **Entity** refers to an IComponentFolder object. For more information on IComponent and IComponentFolder, refer to the *HP ALM Open Test Architecture API Reference*.
Type	Function
Returns	True or False
Availability	`Component_CanDelete`

CanDeleteGroupsFromTest

This event is triggered when a user removes groups from a test of type Flow or Business-Process, to check whether the specified groups can be removed.

Syntax	**CanDeleteGroupsFromTest(Groups)** where **Groups** is an array of group IDs.
Type	Function
Returns	True or False
Availability	`CanDeleteGroupsFromTest`

CanLogin

This event is triggered to check whether the specified user can log in to the specified project.

Syntax	**CanLogin(DomainName, ProjectName, UserName)** where **DomainName** is the domain name, **ProjectName** is the project name, and **UserName** is the user name.
Type	Function
Returns	True or False
Availability	`CanLogin` (all modules)

CanLogout

This event is triggered to check whether the current user can log out of the current project.

Syntax	**CanLogout**
Type	Function
Returns	True or False
Availability	`CanLogout` (all modules)

CanPost

This event is triggered before ALM posts an object to the server, to check whether the object can be posted.

You can add code to this event procedure to prevent an object from being posted in specific cases. For example, see "Example: Object Validation" on page 481.

Syntax	**<entity>_CanPost**
Type	Function
Returns	True or False
Availability	• AnalysisItem_CanPost • AnalysisItemFolder_CanPost • Baseline_CanPost • Bug_CanPost • BusinessModel_CanPost • BusinessModelFolder_CanPost • BusinessModelPath_CanPost • Component_CanPost • ComponentFolder_CanPost • Cycle_CanPost • DashboardFolder_CanPost • DashboardPage_CanPost • Library_CanPost • LibraryFolder_CanPost • Release_CanPost • ReleaseFolder_CanPost • Req_CanPost • Resource_CanPost • ResourceFolder_CanPost • Run_CanPost • Step_CanPost • Test_CanPost • TestConfiguration_CanPost • TestFolder_CanPost • TestSet_CanPost • TestSetFolder_CanPost • TestSetTests_CanPost (does not appear in the Scripts Tree)

CanReImportModels

This event is triggered when attempting to import the specified business process models that already exist in ALM, to check if the business process models can be reimported.

Syntax	**<entity>_CanReImportModels(Models)** where **Models** is an array of Model IDs.
Type	Function
Returns	True or False
Availability	`CanReImportModels`

CanRemoveComponentsFromTest

This event is triggered when a user removes components from a test of type Flow or Business-Process, to check whether the specified components can be removed.

Syntax	**CanRemoveComponentsFromTest(Components)** where **Components** is an array of component IDs.
Type	Function
Returns	True or False
Availability	`CanRemoveComponentsFromTest`

CanRemoveFlowsFromTest

This event is triggered when a user removes flows from a test of type Business-Process, to check whether the specified flows can be removed.

Syntax	**CanRemoveFlowsFromTest(Flows)** where **Flows** is an array of flow IDs.
Type	Function
Returns	True or False
Availability	`CanRemoveFlowsFromTest`

CanRemoveTests

This event is triggered to check whether the specified tests can be removed from a test set.

Syntax	**<entity>_CanRemoveTests(Tests)** where **Tests** is an array of Test Instance IDs.
Type	Function
Returns	True or False
Availability	`TestSet_CanRemoveTests`

DefaultRes

This function is used to determine the default return value for ALM functions, such as FieldCanChange. All ALM workflow functions call this function (unless explicitly omitted by user) to determine the default return value. DefaultRes can be used to quickly replace the default return values of all ALM workflow functions.

Syntax	**DefaultRes**
Type	Function
Returns	True or False
Availability	`DefaultRes` (all modules)

DialogBox

This event is triggered when a dialog box is opened or closed.

Syntax	**DialogBox(DialogBoxName, IsOpen)** where **DialogBoxName** is the name of the dialog box, and **IsOpen** indicates whether the dialog box is open.
Type	Sub
Availability	`DialogBox` (all modules)

Note: For purposes of backward compatibility, this event is also triggered using backward compatible values for defect details (**DialogBoxName="Details"**) and test instance details (**DialogBoxName="TestInstanceDetails"**). These backward compatible values are not recommended.

EnterModule

This event is triggered when the user enters or switches to an ALM module. It is also triggered when the user logs in to ALM.

You can add code to this event procedure to perform an action whenever the user switches to the specified module.

Syntax	**EnterModule**
Type	Sub
Availability	`EnterModule` (all modules)

ExitModule

This event is triggered when the user exits the specified module.

Syntax	**ExitModule**
Type	Sub
Availability	`ExitModule` (all modules)

FieldCanChange

This event is triggered before ALM changes a field value, to determine whether the field can be changed.

You can add code to this event procedure to prevent a field from being changed in specific cases. For example, see "Example: Field Validation" on page 481.

Syntax	**<entity>_FieldCanChange(FieldName, NewValue)** where **FieldName** is the name of the field and **NewValue** is the field value.
Type	Function
Returns	True or False

Availability	• AnalysisItem_FieldCanChange • AnalysisItemFolder_FieldCanChange • Baseline_FieldCanChange • Bug_FieldCanChange • BusinessModel_FieldCanChange • BusinessModelActivity_FieldCanChange • BusinessModelFolder_FieldCanChange • BusinessModelPath_FieldCanChange • Component_FieldCanChange • ComponentFolder_FieldCanChange • ComponentStep_FieldCanChange • Cycle_FieldCanChange • DashboardFolder_FieldCanChange • DashboardPage_FieldCanChange • DesignStep_FieldCanChange • Library_FieldCanChange • LibraryFolder_FieldCanChange • Release_FieldCanChange • ReleaseFolder_FieldCanChange • Req_FieldCanChange • Resource_FieldCanChange • ResourceFolder_FieldCanChange • Run_FieldCanChange • Step_FieldCanChange • Test_FieldCanChange • TestConfiguration_FieldCanChange • TestFolder_FieldCanChange • TestSet_FieldCanChange • TestSetFolder_FieldCanChange • TestSetTests_FieldCanChange

The code for hiding a field that depends on another field should be placed in the FieldChange event procedure (not in the FieldCanChange event procedure).

FieldChange

This event is triggered when the value of the specified field changes.

Every change of value triggers the field change event when the field loses focus.

You can add code to this event procedure to perform an action when the value of a particular field is changed. For example, you can hide or display one field depending on the value the user enters into another field. For example, see "Example: Changing One Field Based on Another Field" on page 480.

Syntax	**<entity>_FieldChange(FieldName)** where **FieldName** is the name of the field.
Type	Sub

Availability	• AnalysisItem_FieldChange • AnalysisItemFolder_FieldChange • Baseline_FieldChange • Bug_FieldChange • BusinessModel_FieldChange • BusinessModelActivity_FieldChange • BusinessModelFolder_FieldChange • BusinessModelPath_FieldChange • Component_FieldChange • ComponentFolder_FieldChange • ComponentStep_FieldChange • Cycle_FieldChange • DashboardFolder_FieldChange • DashboardPage_FieldChange • DesignStep_FieldChange • Library_FieldChange • LibraryFolder_FieldChange • Release_FieldChange • ReleaseFolder_FieldChange • Req_FieldChange • Resource_FieldChange • ResourceFolder_FieldChange • Run_FieldChange • Step_FieldChange • Test_FieldChange • TestConfiguration_FieldChange • TestFolder_FieldChange • TestSet_FieldChange • TestSetFolder_FieldChange • TestSetTests_FieldChange

When a user changes a field value using the **Find/Replace** command, workflow events are not triggered. If restrictions implemented in workflow scripts are critical, consider disabling the **Replace** command for specific user groups, to ensure that your restrictions cannot be bypassed.

GetDetailsPageName

This event is triggered by ALM to retrieve the name of the page (tab) that has the index number specified in **PageNum** in the following dialog boxes:

- An entity's Details dialog box
- An entity's New <entity> dialog box

You can add code to this event procedure to customize the tab names for the Details dialog box. For example, see "Example: Changing Tab Names" on page 478.

Syntax	**GetDetailsPageName(PageName, PageNum)** where **PageName** is the default page (tab) name (for example, `Page 1`) and **PageNum** is the page (tab) number. **Note:** The page number is the absolute page number, regardless of the page's relative position in relation to the other displayed pages in the dialog box.
Type	Function
Returns	String containing the page name
Availability	`GetDetailsPageName` (all modules)

GetNewBugPageName

This event is triggered by ALM to retrieve the name of the New Defect dialog box page (tab) that has the index number specified in PageNum.

You can add code to this event procedure to customize the tab names on the New Defect dialog box. For example, see "Example: Changing Tab Names" on page 478.

Syntax	**GetNewBugPageName(PageName, PageNum)** where **PageName** is the default page (tab) name (for example, `Page 1`) and **PageNum** is the page (tab) number. **Note:** The page number is the absolute page number, regardless of the page's relative position in relation to the other displayed pages in the New Defect dialog box.
Type	Function
Returns	String containing the page (tab) name
Availability	`GetNewBugPageName`

Note: The GetNewBugPageName event is not listed in the Scripts Tree of the Script Editor. This event is triggered for backward compatibility purposes only. GetDetailsPageName should be used instead.

GetNewReqPageName

This event is triggered by ALM to retrieve the name of the New Requirement dialog box page (tab) that has the index number specified in PageNum.

You can add code to this event procedure to customize the tab names on the New Requirement dialog box. For example, see "Example: Changing Tab Names" on page 478.

Syntax	**GetNewReqPageName(PageName, PageNum)** where **PageName** is the default page (tab) name (for example, `Page 1`) and **PageNum** is the page (tab) number. **Note:** The page number is the absolute page number, regardless of the page's relative position in relation to the other displayed pages in the New Defect dialog box.
Type	Function
Returns	String containing the page name
Availability	`GetNewReqPageName`

Note: The GetNewReqPageName event is not listed in the Scripts Tree of the Script Editor. This event is triggered for backward compatibility purposes only. GetDetailsPageName should be used instead.

GetReqDetailsPageName

This event is triggered by ALM to retrieve the name of the Requirement Details dialog box page (tab) that has the index number specified in PageNum.

You can add code to this event procedure to customize the tab names on the Requirement Details dialog box. For example, see "Example: Changing Tab Names" on page 478.

Syntax	**GetReqDetailsPageName(PageName, PageNum)** where **PageName** is the default page (tab) name (for example, `Page 1`) and **PageNum** is the page (tab) number. **Note:** The page number is the absolute page number, regardless of the page's relative position in relation to the other displayed pages in the New Defect dialog box.
Type	Function
Returns	String containing the page name
Availability	`GetReqDetailsPageName`

Note: The GetReqDetailsPageName event is not listed in the Scripts Tree of the Script Editor. This event is triggered for backward compatibility purposes only. GetDetailsPageName should be used instead.

MoveTo

This event is triggered when the user changes focus from one object to another.

You can add code to this event procedure to perform actions when the user changes the focus. For example, see "Example: Presenting a Dynamic Field List" on page 482.

Tip: When moving from one object to another in a tree, the MoveTo event is not triggered. However, it is possible to trigger the event for Requirement trees. For details, see "ENABLE_ENTITY_SELECTION_TREE_REQ_MOVE_TO" on page 179.

Syntax	**<entity>_MoveTo**
Type	Sub

Availability	• AnalysisItem_MoveTo • AnalysisItemFolder_MoveTo • Baseline_MoveTo • Bug_MoveTo • BusinessModel_MoveTo • BusinessModelActivity_MoveTo • BusinessModelFolder_MoveTo • BusinessModelPath_MoveTo • Component_MoveTo • ComponentFolder_MoveTo (formerly MoveToComponentFolder) • ComponentStep_MoveTo • Cycle_MoveTo • DashboardFolder_MoveTo • DashboardPage_MoveTo • DesignStep_MoveTo • Library_MoveTo • LibraryFolder_MoveTo • Release_MoveTo • ReleaseFolder_MoveTo • Req_MoveTo • Resource_MoveTo • ResourceFolder_MoveTo • Run_MoveTo • Step_MoveTo • Test_MoveTo • TestConfiguration_MoveTo • TestFolder_MoveTo • TestSet_MoveTo • TestSetFolder_MoveTo • TestSetTests_MoveTo

MoveToComponentFolder

This event is triggered when the user moves to the specified component folder in the business component tree.

Syntax	**MoveToComponentFolder(Folder)** where **Folder** is the **IComponentFolder** interface. For more information, refer to the *HP ALM Open Test Architecture API Reference*.
Type	Sub
Availability	`MoveToComponentFolder`

Note: The MoveToComponentFolder event is not listed in the Scripts Tree of the Script Editor. This event is supported for purposes of backward compatibility. We recommend you use ComponentFolder_MoveTo event instead.

MoveToFolder

This event is triggered when the user moves to the specified test set folder in the test sets tree.

Syntax	**MoveToFolder(Folder)** where **Folder** is the **ISysTreeNode** interface. For more information, refer to the *HP ALM Open Test Architecture API Reference*.
Type	Sub
Availability	`MoveToFolder`

Note: The MoveToFolder event is not listed in the Scripts Tree of the Script Editor. This event is supported for purposes of backward compatibility. We recommend you use MoveToFolder instead.

MoveToSubject

This event is triggered when the user moves to the specified subject in the test plan tree.

Syntax	**MoveToSubject(Subject)** where **Subject** is the **ISysTreeNode** interface. For more information, refer to the *HP ALM Open Test Architecture API Reference*.
Type	Sub
Availability	`MoveToSubject`

Note: The MoveToSubject event is not listed in the Scripts Tree of the Script Editor. This event is supported for purposes of backward compatibility. We recommend you use MoveToSubject instead.

New

This event is triggered when an object is added to ALM.

You can add code to this event procedure to perform an action when a new object is added. For example, see "Example: Customizing a Defects Module Dialog Box" on page 476.

Syntax	**<entity>_New**
Type	Sub

Availability	• AnalysisItem_New • AnalysisItemFolder_New • Baseline_New • Bug_New • BusinessModelFolder_New • BusinessModelPath_New • Component_New • ComponentFolder_New • ComponentStep_New • Cycle_New • DashboardFolder_New • DashboardPage_New • DesignStep_New • Library_New • LibraryFolder_New • Release_New • ReleaseFolder_New • Req_New • Resource_New • ResourceFolder_New • Step_New • Test_New • TestConfiguration_New • TestFolder_New • TestSet_New • TestSetFolder_New

RemoveComponentFromTest

This event is triggered when the user removes a component from a test of type Flow or Business-Process in the Test Script tab.

Version Control: Changing components checked in or checked out by another user, using the RemoveComponentFromTest event, is not supported.

Syntax	**RemoveComponentFromTest**

Type	Sub
Availability	`RemoveComponentFromTest`

RunTests

This event is triggered when the user clicks the **Run** button to run tests in the Test Lab module, provided that Sprinter is not installed and none of the tests is automated.

Syntax	**RunTests(Tests)** where **Tests** is an array of Test Instance IDs.
Type	Sub
Availability	`RunTests`

RunTests_Sprinter

This event is triggered:

- When the user clicks the **Run** arrow and chooses **Run with** Sprinter to run tests in the Test Lab module.
- When the user clicks the **Run** button to run tests in the Test Lab module, if Sprinter is installed and all the tests are manual.

Syntax	**RunTests_Sprinter(Tests)** where **Tests** is an array of Test Instance IDs.
Type	Sub
Availability	`RunTests_Sprinter`

RunTestSet

This event is triggered when the user clicks the **RunTest Set** button to run a test set in the Test Lab module.

Syntax	**RunTestSet(Tests)** where **Tests** is an array of Test Instance IDs.
Type	Sub
Availability	`RunTestSet`

RunTestsManually

This event is triggered when the user clicks the **Run** arrow and chooses **Run Manually** to run tests in the Test Lab module.

Syntax	**RunTestsManually(Tests)** where **Tests** is an array of Test Instance IDs.
Type	Sub
Availability	`RunTestsManually`

Chapter 31: Workflow Object and Property Reference

Workflow scripts can reference HP Application Lifecycle Management (ALM) objects to obtain information and to change project values. They can also use properties that return information about the current module and dialog box. This chapter lists the ALM objects and properties that are available to workflow scripts.

About ALM Objects and Properties 459

Actions Object 460

Action Object 460

Fields Objects 461

Field Object 463

Lists Object 464

TDConnection Object 464

User Object 465

ALM Properties 465

About ALM Objects and Properties

Workflow scripts can obtain information, make decisions based on that information, and change values in the project based on those decisions.

You can obtain information such as the user group to which the current user belongs, and the value of a field, by accessing objects such as the **User** object or the **Field** object.

You can also obtain information about the active module and active dialog box using workflow properties. For more information on these properties, see "ALM Properties" on page 465.

Your script can change the value of a field or field list. To do so, the script modifies the **Value** property or the **List** property of the appropriate **Field** object.

For information on the event procedures in which you place VBScript code to create workflow scripts, see "Workflow Event Reference" on page 428.

The following table lists the ALM objects that are available when you write a script.

Object	Description
Actions	The list of actions that are available. See "Actions Object" on next page.
Action	The **Action** object is handled by the **Actions** object. See "Action Object" on next page.
Fields	Includes the objects that provide access to specific fields. See "Fields Objects" on page 461.
Field	The **Field** object is handled by the **Fields** objects. See "Field Object" on page 463.
Lists	Includes the lists that are available in an ALM project. See "Lists Object" on page 464.
TDConnection	Provides access to open test architecture (OTA) objects. See "TDConnection Object" on page 464.
User	Includes the properties of the current user. This object is available in all modules. See "TDConnection Object" on page 464.

Note: In some cases, a function returns the object itself instead of the ID property of the object. For example, after the following statement has been executed, `testsetf` is a reference to a **TestSetFolder** object:
`Set testsetf = TestSet_Fields("CY_FOLDER_ID").Value.`

For information on the Script Editor used to write workflow scripts, see "Working with the Workflow Script Editor" on page 416.

For each ALM object, this chapter lists the properties of the object. The list includes the property name, a description, and the data type of the property. It indicates whether the property is read-only (R) or whether your script can modify it (R/W).

Version Control: After enabling version control for a project, you should review all its workflow scripts and make adjustments for each checked in entity. This includes the following entities: **Req**, **Test**, **Resource**, and **Component**. For each checked in entity that includes a **Post** function in its script, you must modify the script. To modify, add a **Checkout** function before every **Post** function. Making this modification prevents the Check Out dialog box from opening each time a call to a **Post** function is made. For more information about the Post and Checkout functions, see the *HP ALM Open Test Architecture Reference*.

For more information on version control, refer to the *HP Application Lifecycle Management User Guide*.

Actions Object

You can use the **Actions** object to manipulate toolbar buttons, menu commands, and dialog boxes.

The **Actions** object has the following property:

Property	R/W	Type	Description
Action	R	Object	Allows access to every action in a list. The index for this property is the action name.

Action Object

You can use the **Action** object to verify whether a button or command is enabled, checked, or visible. You can also use it to execute actions.

For example, to set the Defect Details dialog box to open automatically when the user moves from one defect to another in the Defects Grid, place the following code in the `Bug_MoveTo` event procedure:

```
Set NewDefectAction=Actions.Action("Defects.DefectDetails")
NewDefectAction.Execute
```

To obtain the name of an action, add the following lines to the ActionCanExecute event procedure, perform the action, and note the action name that is printed in the message:

```
Sub ActionCanExecute(ActionName)
        On Error Resume Next
        MsgBox "You have performed an action named: " & ActionName
        On Error GoTo 0
End Sub
```

This object has the following properties:

Property	R/W	Type	Description
Checked	R/W	Boolean	Indicates whether an action is checked in ALM.
Enabled	R/W	Boolean	Indicates whether an action is enabled. A disabled action cannot be invoked by the user, but can be invoked from the workflow script.
Visible	R/W	Boolean	Indicates whether an action is visible in ALM.

The **Action** object includes the following method:

Method	Description
Execute	Executes the action.

When a workflow script invokes an action using the **Execute** method of the **Action** object, the workflow events that would be triggered if a user initiated the action from a dialog box are by default not triggered. Therefore, when using **Action.Execute**, you must ensure that you do not bypass the site policies you are enforcing with workflow events.

To enable workflow events to be triggered from within a dialog box, set the value of the **AllowReentrancy** flag to **true**. To restore the default settings, so that these events are not triggered, set the value of the **AllowReentrancy** flag to **false**. For example, to set the Add Defect dialog box to open automatically when a user enters the Defects module, place the following code in the **EnterModule** event procedure:

```
AllowReentrancy=true
Set NewDefectAction=Actions.Action("Defects.DefectDetails")
NewDefectAction.Execute
AllowReentrancy=false
```

If the value of the **AllowReentrancy** flag is set to **false**, the dialog box opens as usual, but workflow customizations will not work in the dialog because the workflow events for the dialog box are not triggered.

Caution: Consider carefully the implications of setting the value of this flag to **true**. If you set the value of the flag to **true**, you enable a function to call another function which may call the original function. This can cause an endless loop. This can also occur when functions call internal functions which call the original function.

Fields Objects

You can use the following objects in workflow scripts to access the fields of ALM modules:

Object	Description
AnalysisItem_Fields	Provides access to the fields of the reports and graphs in the Dashboard module.
AnalysisItemFolder_Fields	Provides access to the fields of the report and graph folders in the Dashboard module.
Baseline_Fields	Provides access to the fields of the baselines in the Libraries module.
Bug_Fields	Provides access to the fields of the defects in the Defects module and the Manual Runner dialog box.
Component_Fields	Provides access to the fields of components in the Business Components module.

Object	Description
ComponentStep_ Fields	Provides access to the fields of component steps in the Business Components module.
Cycle_Field	Provides access to the fields of cycles in the Releases module.
DashboardFolder_ Fields	Provides access to the fields of dashboard page folders in the Dashboard module.
DashboardPage_ Fields	Provides access to the fields of dashboard pages in the Dashboard module.
DesignStep_Fields	Provides access to the fields of the design steps in the Test Plan module.
Library_Fields	Provides access to the fields of the libraries in the Libraries module.
LibraryFolder_ Fields	Provides access to the fields of the library folders in the Libraries module.
Release_Fields	Provides access to the fields of the releases in the Releases module.
ReleaseFolder_ Fields	Provides access to the fields of the release folders in the Releases module.
Req_Fields	Provides access to the fields of the Requirements module.
Resource_Fields	Provides access to the fields of the resources in the Test Resources module.
ResourceFolder_ Fields	Provides access to the fields of the resource folders in the Test Resources module.
Run_Fields	Provides access to the fields of the test runs in the Manual Runner dialog box.
Step_Fields	Provides access to the fields of the steps in the Manual Runner dialog box.
Test_Fields	Provides access to the fields of tests in the Test Plan module.
TestSet_Fields	Provides access to the fields of the test sets in the Test Lab module.
TestSetTest_Fields	Provides access to the fields of the test instances in the Test Lab module.

For example, to set a certain property for all fields in the **Req_Fields** object, you can refer to each field by its ID number (**Req_Fields.FieldById**). To set all fields to be visible (**IsVisible**) in a dialog box, you can use the following code:

```
For i = 1 to Req_Fields.Count
        Req_Fields.FieldById(i).IsVisible = True
Next
```

These objects have the following properties:

Property	R/W	Type	Description
Count	R	Long	Returns the number of fields in the current object.
Field (FieldName)	R	Object	Accesses the fields by field name or field label.
FieldById (FieldID)	R	Object	Accesses the fields by the field ID number.

Tip: To avoid errors if your script attempts to access a non-active or a non-existing field, include **On Error Resume Next** in the script.

Field Object

You can use the **Field** object to access the properties of an entity field.

For example, to display a message box when a user does not have permission to change a value in the **Status** field, you can use the following code:

```
Msgbox "You do not have permission to change "_
& "Bug_Fields.Field("BG_STATUS").FieldLabel field."
```

The **Field** object has the following properties:

Property	R/W	Type	Description
FieldLabel	R	String	The displayed label of the field.
FieldName	R	String	The logical name of the field.
IsModified	R	Boolean	Specifies whether the value was modified.
IsMultiValue	R	Boolean	Specifies whether the field can contain multiple values from a lookup list.
IsNull	R	Boolean	Specifies whether the field value is absent.
IsReadOnly	R/W	Boolean	Specifies whether the field is read-only.
IsRequired	R/W	Boolean	Specifies whether a field value is required. This enables you to override field customization information. To modify the **IsRequired** property of a field, the **IsVisible** property must be `True`. Changes to **IsRequired** are ignored if the field is not visible. Users must always enter a value for a field that is set as required by the workflow. This applies whether they are modifying an existing record or adding a new record, and even if the field is already empty.
IsVisible	R/W	Boolean	Specifies whether the field is displayed.

Property	R/W	Type	Description
List	R/W	List	Sets or retrieves the field list attached to a field of type lookup list.
PageNo	R/W	Integer	Sets or retrieves the page (tab) on which the field is displayed in the New Defect and Defect Details dialog boxes.
Value	R/W	Variant	Sets or retrieves the value of the field.
ViewOrder	R/W	Integer	Sets or retrieves the order in which the fields are displayed in the New Defect and Defect Details dialog boxes. You must set the value for every field in the dialog box.

Lists Object

You can use the **Lists** object to limit field input to a specific list of values.

For example, to set the list in the **Planned Closing Version** field, depending on the **Project** field value, you can use the following code:

```
If Bug_Fields.Field("BG_PROJECT").Value = "Project 1" Then
        Bug_Fields.Field("BG_PLANNED_CLOSING_VER").List _
        = Lists("All Projects")
       ' ...
End If
```

For more information, see "Example: Presenting a Dynamic Field List" on page 482.

The **Lists** object can be used only with fields that are defined as the **Lookup List** type or the **String** type in Project Customization of project entities.

The **Lists** object has the following properties:

Property	R/W	Type	Description
List	R	ISysTreeNode	Accesses the ALM lists.

Note: When workflow customization has been used to change a list of values for a field that has transition rules defined, the field may only be modified in a way that satisfies both the workflow script and the transition rules. For more information, see "Setting Transition Rules" on page 257.

TDConnection Object

In workflow scripts, the only objects that are available are the objects of the module in which the code is written and a limited number of global objects. One of the global objects is the **TDConnection** object. **TDConnection** provides access to the open test architecture (OTA) objects.

You can use the **TDConnection** object to access objects from other modules, and to access general session parameters. You can access **TDConnection** properties in any procedure, from any module.

For more information about the **TDConnection** object, and a list of **TDConnection** properties, refer to the *HP ALM Open Test Architecture API Reference*.

For examples of using the **TDConnection** object in workflow scripts, see "Workflow Examples and Best Practices" on page 468.

User Object

You can access the **User** object to retrieve the user name of the current user and to check whether the user belongs to a particular user group. You can retrieve or modify the first and last name of the user.

For example, to have a message box open when the user has project administrator permissions, use the following code:

```
If User.IsInGroup("TDAdmin") Then
        MsgBox "The user " & User.FullName & _
        " has administrative permissions for this project."
End If
```

For more information, see "Example: Changing a Field Based on the User Group" on page 480, and "Example: Controlling User Permissions" on page 484.

To access user properties that cannot be accessed by the **User** object, you can use the **TDConnection** object of the ALM open test architecture (OTA).

The **User** object has the following properties:

Property	R/W	Type	Description
FullName	R/W	String	Sets or retrieves the first and last name of the current user.
IsInGroup (GroupName)	R	Boolean	Checks whether or not the current user is a member of a predefined/user-defined group.
UserName	R	String	Returns the user name used when logging in to ALM.

ALM Properties

You can use the **ActiveModule** and **ActiveDialogName** properties to obtain information about the active module and dialog box.

This section includes:

ActiveModule Property 466

ActiveDialogName Property 466

ActiveModule Property

The **ActiveModule** property returns the name of the active ALM module. The following values can be returned:

- `Releases`
- `Libraries`
- `Analysis`
- `Dashboard`
- `Requirements`
- `Business Models`
- `Test Resources`
- `Business Components`
- `Test Plan`
- `Test Lab`
- `Defects`

Example

To open a message box displaying the module name when you move to a new module, use the following code:

```
Sub EnterModule
        On Error Resume Next
        msgbox "You have just entered the " & ActiveModule & _
        " module."
        On Error GoTo 0
End Sub
```

ActiveDialogName Property

The **ActiveDialogName** property returns the name of the active dialog box.

Example

To open a message box displaying the dialog box name when you open a new dialog box, use the following code:

```
Sub DialogBox(DialogBoxName, IsOpen)
        On Error Resume Next
        msgbox "You have just opened the " & ActiveDialogName & _
        " dialog box."
        On Error GoTo 0
End Sub
```

Chapter 32: Workflow Examples and Best Practices

This chapter provides considerations and examples for workflow scripts.

About the Workflow Examples 469
Best Practices for Writing Workflow Scripts 469
Example: Customizing a Defects Module Dialog Box 476
Example: Changing Tab Names 478
Example: Adding a Template to a Memo Field 479
Example: Changing One Field Based on Another Field 480
Example: Changing a Field Based on the User Group 480
Example: Object Validation 481
Example: Field Validation 481
Example: Presenting a Dynamic Field List 482
Example: Changing Field Properties when a Field Changes 483
Example: Controlling User Permissions 484
Example: Adding Button Functionality 484
Example: Error Handling 485
Example: Obtaining Session Properties 486
Example: Sending Mail 486
Example: Storing the Last Values Entered 487
Example: Copying Field Values to Another Object 489

About the Workflow Examples

The workflow examples presented in this chapter perform several types of tasks. The following table lists the examples that illustrate each type of task.

Workflow Task	See Examples
dialog box customization	"Example: Customizing a Defects Module Dialog Box" on page 476 "Example: Changing Tab Names" on page 478
field value automation	"Example: Adding a Template to a Memo Field" on page 479 "Example: Changing One Field Based on Another Field" on page 480 "Example: Changing a Field Based on the User Group" on page 480
data validation	"Example: Object Validation" on page 481 "Example: Field Validation" on page 481
dynamic field customization	"Example: Presenting a Dynamic Field List" on page 482 "Example: Changing Field Properties when a Field Changes" on page 483
user permission control	"Example: Controlling User Permissions" on page 484
functionality	"Example: Adding Button Functionality" on page 484
error handling	"Example: Error Handling" on page 485
using OTA to obtain session parameters	"Example: Obtaining Session Properties" on page 486
sending mail	"Example: Sending Mail" on page 486
using the **Settings** object	"Example: Storing the Last Values Entered" on page 487
copying values between modules	"Example: Copying Field Values to Another Object" on page 489

Best Practices for Writing Workflow Scripts

This section describes best practices for writing workflow scripts and making sure the scripts run as expected. In addition to the best practices provided in this section, you can refer to the Microsoft Developer Network VBScript Language Reference at http://msdn.microsoft.com/en-us/library/.

The following best practices are described in this section:

General VBScript Tips and Best Practices

- "Checking Value Types Before Use" below
- "Anticipating Full Evaluation of Logical Expressions" on next page
- "Defining Default Behavior for Select Case and If-Then-Else Statements" on page 472
- "Setting Return Values in Functions" on page 473

ALM Workflow Tips and Best Practices

- "Making Sure that Entity Properties Are Set Before an Entity Comes into Focus" on page 473
- "Check if a Dialog Box is Open" on page 474
- "Avoid Defining Duplicate Subroutines" on page 475

Checking Value Types Before Use

VBScript is a "weakly-typed" programming language. This means that you can create, use, and access data values without initially declaring their types. However, certain operations can be performed only on values of a specific type. Therefore, it is important to check the type of the data before performing any operations on them.

Values of different types behave differently in different statements. Object value behavior is even more unpredictable because the behavior depends on the object's implementation. For example, the object in the call **<entity>_CanDelete(Entity)** can either be text or a subject node.

Recommendations

To avoid unpredictable results:

- Check value types before use, especially for object types. When checking an object type, also check that the object has the properties you access.

  Note: In the examples provided in this chapter, only object types are checked before use.

- Assume as little as possible—do not assume that a value is of a certain type. Write scripts that can handle all possibilities by using Else statements and Select Case statements.
- Always check parameter types before use with various VBScript functions, such as IsArray, IsDate, IsNull, IsEmpty, IsNumeric, and IsObject.
- Do not assume an object's default property is of a specific type; the type can vary from object to object.
- Use VBScript built-in conversion functions to achieve a degree of type safety.
- When working with objects, check that the value you receive is neither Null or Empty by calling the IsNull and IsEmpty functions.

Examples

For the purposes of the following examples, assume the field values are declared as in the

table below.

Field Values	Type
`Bug_Fields["BG_BUG_ID"].Value`	Integer
`Bug_Fields["BG_SUMMARY"].Value`	String
`Bug_Fields["BG_SUBJECT"].Value`	Object implementing the ISysTreeNode interface

In the following example, statement usage is correct. The integer is converted to a string.

```
If Bug_Fields["BG_BUG_ID"].Value = "10" Then...
```

In the following example, statement usage is correct. The strings are comparable.

```
If Bug_Fields["BG_SUMMARY"].Value = "some text" Then...
```

In the following example, statement usage is incorrect. This code can work only when the value of `BG_SUBJECT` field is neither Empty or Null. VBScript also assumes that this objects's default value (meaning, the default property) is either of string type or is comparable with the string type, which is not always the case.

```
If Bug_Fields["BG_SUBJECT"].Value = "My Tests" Then...
```

Anticipating Full Evaluation of Logical Expressions

The VBScript programming language does not short-circuit evaluation of Boolean conditions. VBScript evaluates all the terms in a Boolean logical expression, even if the expression can be established as True or False without evaluating all its terms. For example, in the following example, both <`statement1`> and <`statement2`> are evaluated, even if <`statement1`> resolves to False:

```
<statement 1> AND <statement 2>
```

Recommendations

To avoid errors, check that all values and objects are not Null before attempting to use them.

Examples

The following examples:

- demonstrate incorrect and correct usage of logical expressions
- take into consideration how logical expressions are evaluated

Incorrect Usage

`value.Name` is evaluated even when its value is Null. This causes an error.

```
Sub namecheck(value)
        If Not IsNull(value) And value.Name = "aName" Then
            ' ...
          End If
End Sub
```

Correct Usage

The code is correct on the condition that `value` is an object that contains the `Name` property. The code runs without errors.

```
Sub namecheck(value)
    If Not IsNull(value) And Not IsEmpty(value) Then
         If value.Name = "aName" Then
            ' ...
        End If
      End If
End Sub
```

Defining Default Behavior for Select Case and If-Then-Else Statements

Unpredictable results can occur when no default action is defined for Select Case statements or If-Then-Else statements.

Recommendations

To avoid unpredictable results, always define default behavior when using Select Case of If-Then-Else statements.

Example

The following are examples of incorrect and correct ways to define default behavior for situations not covered by the existing Select Case and If-Then-Else statements.

Incorrect Usage

The author of this subroutine intends for the `BG_USER_01` field to be visible only if the defect's status is Open, New, or Reopen. However, if the IsVisible property of a Closed or Fixed defect was set to True prior to the instance of this subroutine, that Closed or Fixed defect will also be visible. This is because there is no case statement defined specifically for Closed and Fixed defects.

```
Sub Bug_FieldChange(FieldName)
    If FieldName="BG_STATUS" Then
        Select Case Bug_Fields(FieldName).Value
            Case "Open", "New", "Reopen" _
            Bug_Fields("BG_USER_01").IsVisible = True
        End Select
```

```
    End If
End Sub
```

Correct Usage

This subroutine effectively handles all possible cases.

```
Sub Bug_FieldChange(FieldName)
    If FieldName="BG_STATUS" Then
        Select Case Bug_Fields(FieldName).Value
            Case "Open", "New", "Reopen"
                Bug_Fields("BG_USER_01").IsVisible = True
            Case Else
                Bug_Fields("BG_USER_01").IsVisible = False
        End Select
    End If
End Sub
```

Setting Return Values in Functions

If a function ends without a return value, unpredictable and inconsistent results may occur. Also, it is difficult to debug behavior if a return code is not set.

Recommendations

To avoid unpredictable results, set a default return value at the beginning of each function.

Making Sure that Entity Properties Are Set Before an Entity Comes into Focus

It is common practice to set entity properties (such as IsVisible, IsRequired, and List) when creating or modifying a new entity (New or FieldChanged). When writing ALM workflow scripts, it is also important to set entity properties when the entity comes into focus (meaning, when the user navigates to that entity in the ALM graphical user interface). When an entity comes into focus, the MoveTo event is called.

If entity values are not set in the MoveTo event, the end user experience is unpredictable—for example, incorrect values might be displayed in drop-down lists.

Recommendations

To avoid unpredictable results, such as a drop-down list not containing the most up-to-date set of values:

- Make sure that all entity properties are set in the MoveTo event—not just in the New or FieldChanged events.
- Isolate entity properties customization code into a separate routine and call that routine from all relevant events.

Example

The following table provides an example of how to make sure that a defect's properties are set appropriately when the defect is in focus—and not just when it is modified or added.

```
Sub SetupBugFields(Context1, Context2)
    ' Code for customizing the defect's properties is entered here,
    ' such as set IsVisible, IsRequired, IsReadonly, Label, List...
    If Context1="Focus" Then
          ' Code for handling the focus event is entered here
    ElseIf Context1="FieldChange" Then
            If Context2="RQ_USER_01" Then
                ' Code for handling the FieldChange event
                ' is entered here
            ElseIf Context2="RQ_REQ_STATUS" Then
                ' ... Enter your code here
            Else
                ' ... Enter your code here
            End If
  End If
End Sub
Sub Req_FieldChange(FieldName)
    If FieldName = "RQ_REQ_STATUS" Then
        SetupBugFields("FieldChange", FieldName)
    Else
        '  ...Enter your code here
    End If
End Sub
Sub Req_MoveTo
        SetupBugFields("Focus")
End Sub
```

Check if a Dialog Box is Open

It is helpful to track whether a dialog box is open before performing certain actions. For example:

- Dialog boxes do not need to be refreshed but grid displays do.
- Certain workflow events are not allowed when a dialog box is open.

The DialogBox event can be used to track the visibility of dialog boxes.

Recommendations

To avoid unpredictable results, determine if a dialog box is open before any events occur.

Example

The following example checks whether the dialog box for creating a new defect is open. This is

relevant because the `BG_USER_01` field can only be modified for a new defect. If a different dialog box is open, such as the dialog box for editing a defect, the `BG_USER_01` field cannot be modified.

```
' Declare a global variable for each dialog box of interest
Dim NewDefectDialogIsOpen
' Initialize the global variable
NewDefectDialogIsOpen = False
Sub DialogBox(DialogBoxName, IsOpen)
    If DialogBoxName="New Bug" Then
        NewDefectDialogIsOpen = True
    Else
        NewDefectDialogIsOpen = False
    End If
End Sub
Function Bug_FieldCanChange(FieldName, NewValue)
' Initialize the function's return value to avoid
' unpredictable behavior.
Bug_FieldCanChange = True
' The BG_USER_01 field can only be modified for a new defect.
If FieldName="BG_USER_01" Then
    If NewDefectDialogIsOpen Then
        Bug_FieldCanChange = True
    Else
        Bug_FieldCanChange = False
    End If
End If
End Function
```

Avoid Defining Duplicate Subroutines

If you define a subroutine in one section and then add another subroutine with the same name in another section, the subroutines will conflict. One of them will be ignored.

Example: If you define the subroutine, **MySub**, in the Test Lab module script section, and then define another subroutine, MySub, in the Manual Runner script section, one of your defined subroutines will be ignored.

Recommendations

To avoid unpredictable conflicts when defining subroutines, always check if another subroutine with the same name already exists in your project.

Example: Customizing a Defects Module Dialog Box

This example shows how you can customize the field layout and other field properties in the Add Defect dialog box. You can create similar code to arrange the layout of the Defect Details dialog box.

This example illustrates a solution that customizes field properties for all user groups. You can also use the script generators to customize the layout of the Defects module dialog boxes. If you use the script generators, you must perform customization separately for each user group. For information on these script generators, see "Customizing Defects Module Dialog Boxes" on page 406.

This example involves the following procedures:

- `SetFieldApp` is a general purpose procedure that receives a field name and its properties as parameters, and assigns the properties to the field. See "SetFieldApp" below.
- `FieldCust_AddDefect` calls `SetFieldApp` for each field in the Add Defects dialog box, to set the properties of the field. For some of the fields, `FieldCust_AddDefect` checks the user group to which the current user belongs, and customizes the field properties accordingly. A call to `FieldCust_AddDefect` is placed in the `Bug_New` event procedure. See "FieldCust_AddDefect" on next page.

Note: To implement this example, you can run the **Add Defect Field Customization** script generator and then modify the resulting scripts.

- Rename the generated function `WizardFieldCust_Add` to `FieldCust_AddDefect` and modify it as necessary. (Before you modify a generated script, you must rename it so that it is not overwritten the next time you run the script generator.)
- The script generator places a call to `WizardFieldCust_Add` in the event procedure `Bug_New`. Change this to `FieldCust_AddDefect`.
- The function SetFieldApp is generated when you run the script generator. You do not need to rename or modify this function.

SetFieldApp

The subroutine SetFieldApp receives a field name and its properties as parameters, and assigns the properties to the field.

The subroutine assigns the following field properties: field visibility, whether the field is required, the number of the page (tab) on which the field should be displayed, and the view order (from left to right and from top to bottom).

Add a call to the subroutine SetFieldApp in the user-defined function FieldCust_AddDefect. For more information on this function, see "FieldCust_AddDefect" on next page.

```
Sub SetFieldApp(FieldName, Vis, Req, PNo, VOrder)
    On Error Resume Next
    With Bug_Fields(FieldName)
        .IsVisible = Vis
```

```
        .IsRequired = Req
        .PageNo = PNo
        .ViewOrder = VOrder
    End With
    PrintError "SetFieldApp"
    On Error GoTo 0
End Sub
```

FieldCust_AddDefect

The user-defined function FieldCust_AddDefect calls the function SetFieldApp.

The function first sets all fields to be invisible, not required, and to appear on page 100 at location 0. This ensures that if you add a new field using the **Project Entities** link on the Project Customization window, the layout will not be changed.

Add a call to FieldCust_AddDefect in the Bug_New event procedure so that it will be triggered when a user adds a new defect:

```
Sub Bug_New
        FieldCust_AddDefect
End Sub
```

First, the code handles the fields that are common to all user groups. It uses conditional statements for the fields that will appear in the dialog box only for specific user groups, or that will have different properties for different users.

```
Sub FieldCust_AddDefect
        On Error Resume Next
        ' Initialize the fields of the defect
        For i= 0 To Bug_Fields.Count -1
            SetFieldApp Bug_Fields.FieldByID(i).FieldName, _
            False, False, 100, 0
        Next
        ViewNum = 0
        PageNum = 0
       ' Set fields that are in common for all user groups
        SetFieldApp "BG_BUG_ID", True, True, PageNum, ViewNum
        ViewNum = ViewNum + 1
        SetFieldApp "BG_DESCRIPTION", True, False, PageNum, ViewNum
        ViewNum = ViewNum + 1
        SetFieldApp "BG_SUMMARY", True, True, PageNum, ViewNum
        ViewNum = ViewNum + 1
        SetFieldApp "BG_DETECTED_BY", True, True, PageNum, ViewNum
        ViewNum = ViewNum + 1
        SetFieldApp "BG_DETECTION_DATE", _
        True, True, PageNum, ViewNum
        ViewNum = ViewNum + 1
        SetFieldApp "BG_DETECTION_VERSION", True, True, PageNum, _
        ViewNum
        ViewNum = ViewNum + 1
        SetFieldApp "BG_SEVERITY", True, True, PageNum, ViewNum
```

```
        ViewNum = ViewNum + 1
        SetFieldApp "BG_PRIORITY", True, True, PageNum, ViewNum
        ViewNum = ViewNum + 1
        SetFieldApp "BG_PROJECT", True, False, PageNum, ViewNum
        ViewNum = ViewNum + 1
        SetFieldApp "BG_REPRODUCIBLE", True, False, PageNum, ViewNum
        ViewNum = ViewNum + 1
        SetFieldApp "BG_STATUS", True, False, PageNum, ViewNum
        ViewNum = ViewNum + 1
        ' Set fields that are different for different user groups.
        ' Since one user can belong to multiple user groups,
        ' or none of these groups, there is no need for an Else statement.
        If User.IsInGroup("Developer") Then
            SetFieldApp "BG_PLANNED_CLOSING_VERSION", True, False, _
            PageNum, ViewNum
            ViewNum = ViewNum + 1
            SetFieldApp "BG_PLANNED_FIX_TIME", True, False, PageNum, _
            ViewNum
            ViewNum = ViewNum + 1
        End If

        If User.IsInGroup("QATester") Then
            PageNum = PageNum + 1
            SetFieldApp "BG_USER_01", True, False, PageNum, ViewNum
            ViewNum = ViewNum + 1
            SetFieldApp "BG_USER_02", True, False, PageNum, ViewNum
            ViewNum = ViewNum + 1
        End If

        SetFieldApp "BG_ACTUAL_FIX_TIME", True, False, PageNum, _
        ViewNum
        ViewNum = ViewNum + 1
        ' ...
        PrintError "FieldCust_AddDefect"
        On Error GoTo 0
End Sub
```

Example: Changing Tab Names

You can change the names of the tabs on the Add Defect dialog box. This example sets the tabs to `General`, `Environments`, and `Business Case`.

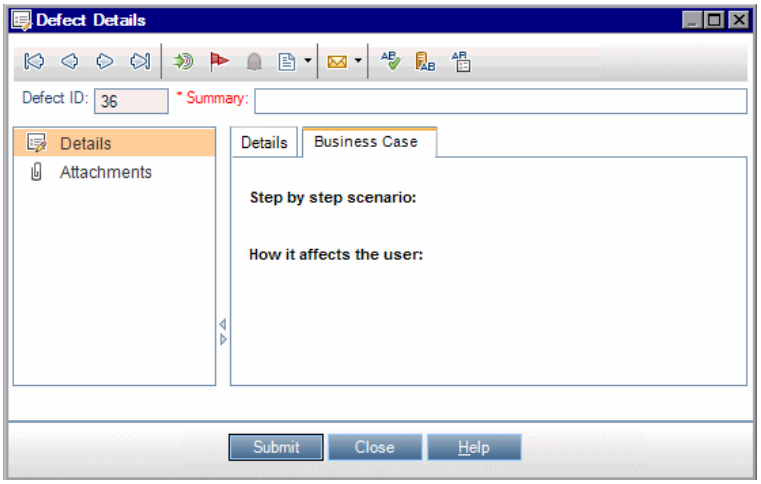
Add the following code to the `GetNewBugPageName` event procedure, which is triggered before ALM opens the Add Defect dialog box. To change the tab names on the Defect Details dialog box, add similar code to the `Defects_GetDetailsPageName` event procedure.

```
Function GetNewBugPageName(PageName, PageNum)
        ' Initialize the return value to a default value
        ' to avoid unpredictable behavior.
        GetNewBugPageName="Business Case"
        On Error Resume Next
```

```
        Select case PageNum
            case "1"
                GetNewBugPageName="General"
            case "2"
                GetNewBugPageName="Environments"
            case else
                GetNewBugPageName="Business Case"
        End Select
        PrintError "GetNewBugPageName"
        On Error GoTo 0
End Function
```

Example: Adding a Template to a Memo Field

You can use workflow scripts to add a default template to a memo field. This example adds text to a memo field called **Business Case** to display the following template:



Perform this customization by placing the HTML code for the text into the **BG_USER_25** field when a defect is added. This example assumes that the user-defined field **BG_USER_25** stores a business case string.

Add the code to the `Bug_New` event procedure, which is triggered when a user adds a new defect.

```
Sub Bug_New
        On Error Resume Next
        Bug_Fields("BG_USER_25").value = _
        "<html><body><b>Step by step scenario:</b>" & _
        "<br><br><br><b>How it affects the user:</b></body></html>"
        PrintError "Bug_New"
        On Error GoTo 0
End Sub
```

Example: Changing One Field Based on Another Field

This example demonstrates how you can change a field value based on the value entered into another field.

For example, you can cause defects to be assigned to user **alex_qc** when `UI Suggestion` is typed into the **Category** field, and to user **alice_qc** when `Security Issues` is typed.

The example assumes that the user-defined field **BG_USER_05** is used to store the category. When the **Category** field is changed in the Defects module, the **BG_RESPONSIBLE** field is assigned the appropriate value.

Add the code to the Bug_FieldChange event procedure so that it is triggered when a user changes a field value for a defect.

```
Sub Bug_FieldChange(FieldName)
        On Error Resume Next
        If FieldName = "BG_USER_05" then
            Select case Bug_Fields("BG_USER_05").Value
            case "UI Suggestion"
                Bug_Fields("BG_RESPONSIBLE").value="alex_qc"
            case "Security Issue"
                Bug_Fields("BG_RESPONSIBLE").value="alice_qc"
            Case Else
                Bug_Fields("BG_RESPONSIBLE").value="non-assigned"
            End Select
        End If
        PrintError "Bug_FieldChange"
        On Error GoTo 0
End Sub
```

Example: Changing a Field Based on the User Group

This example demonstrates how you can change a field value according to the user group of the user entering the defect.

In this example, the user-defined field **BG_USER_01** is a detection mode field in which the user who detected the defect can enter the way in which it was discovered. Possible values are `Formal testing, Informal testing,` and `BTW`.

The example sets the value of the detection mode field to `BTW` when a defect is opened by a user who is not in the `QA Tester` group. If the defect is opened by a user who is in the `QA Tester` group, the default value `Formal testing` is set.

Add the code to event procedure `Bug_New`, so that it is triggered when a defect is added.

```
Sub Bug_New
        On Error Resume Next
        If not User.IsInGroup("QATester") then
```

```
            Bug_Fields("BG_USER_01").Value = "BTW"
        Else
            Bug_Fields("BG_USER_01").Value = "Formal testing"
        End If
        PrintError "Bug_New"
        On Error GoTo 0
End Sub
```

Example: Object Validation

This example demonstrates how you can perform validations of all fields by using the `CanPost` event procedure. For example, this code segment ensures that a user cannot reject a defect without adding a comment.

In this example, a user may not post a defect where the defect status (**BG_STATUS**) has been changed to `Rejected` unless some explanatory text has been typed in the **R&D Comment** field (**BG_DEV_COMMENTS**).

Add the code to the `Bug_CanPost` event procedure so that the check is performed when the user attempts to submit the defect.

```
Function Bug_CanPost
        ' Initialize the function's return value
        ' to avoid unpredictable behavior.
        Bug_CanPost =  False
        On Error Resume Next
        If Bug_Fields("BG_STATUS").IsModified and _
        Bug_Fields("BG_STATUS").Value = "Rejected" and _
        not Bug_Fields("BG_DEV_COMMENTS").IsModified then
            Bug_CanPost = False
            msgbox "You must enter a comment when rejecting a defect."
        Else
            Bug_CanPost = True
        End If
        PrintError "Bug_CanPost"
        On Error GoTo 0
End Function
```

Example: Field Validation

This example demonstrates how to validate a single field value. For example, the following code segment shows how you can ensure that a user in a specific group cannot lower the priority of a defect.

In this example, if the user is in the `QATester` group and the **BG_PRIORITY** field is being modified, the new value of the **BG_PRIORITY** field cannot be lower than the current value.

This example assumes that in the **Priority** field list for the project, lower priorities come first when the values are sorted in ascending order. For example, the list meets this requirement if the elements are as follows: `1-Low, 2-Medium, 3-High`.

Add the code to the `Bug_FieldCanChange` event procedure so that it is triggered when the user attempts to change a defect field value.

```
Function Bug_FieldCanChange(FieldName, NewValue)
        ' Initialize the function's return value
       ' to avoid unpredictable behavior.
        Bug_FieldCanChange = True
        On Error Resume Next
        If User.IsInGroup("QATester") and FieldName ="BG_PRIORITY" _
        Then
            If NewValue < Bug_Fields("BG_PRIORITY").Value then
                Bug_FieldCanChange = False
                msgbox "You do not have permission to lower " _
                & "defect priority."
            Else
                Bug_FieldCanChange = True
            End If
        Else
            ' Enter your code here.
        End If
        PrintError "Bug_FieldCanChange"
        On Error GoTo 0
End Function
```

Example: Presenting a Dynamic Field List

This example demonstrates how you can present a different field list in a field, depending on the value of another field.

The user-defined function `SW_SetLists_Environment` checks the value of the **Environment Specification** field and assigns the appropriate field list to the **Environment Type** field.

This example assumes that the field lists have been defined in the project. For more information, see "Customizing Project Lists" on page 310.

> **Note:** To use workflow scripts to change or create lists that can be assigned to fields, you must use the Open Test Architecture (OTA) interface.

Add code to the `Bug_MoveTo` event procedure so that the user-defined function `SW_SetLists_Environment` is called when the user changes focus in the defects module.

```
Sub Bug_MoveTo()
        On Error Resume Next
        SW_SetLists_Environment
        PrintError "Bug_MoveTo"
        On Error GoTo 0
End Sub
```

Add code to the `Bug_FieldChange` event procedure so that the user-defined function `SW_SetLists_Environment` is called when a user changes the value of the **Environment Type** field in the Defects module.

```
Sub Bug_FieldChange(FieldName)
        On Error Resume Next
        If FieldName = "BG_USER_01" then
            SW_SetLists_Environment
        Else
            ' Enter your code here.
        End If
        PrintError "Bug_FieldChange"
        On Error GoTo 0
End Sub
```

The user-defined function `SW_SetLists_Environment` checks the value of the **Environment Specification** field (**BG_USER_02**) and assigns the appropriate field list to the **Environment Type** field (**BG_USER_01**).

```
Sub SW_SetLists_Environment()
        Dim listName
        On Error Resume Next
        Select Case Bug_Fields("BG_USER_01").Value
        Case "Browser"
            listName = "Browsers"
        Case "Database Type"
            listName = "Database Type"
        Case "Operating System"
            listName = "Platform"
        Case "Web Server"
            listName = "Web Server"
        Case Else
            listName = "Environment Specification"
        End Select
        Bug_Fields("BG_USER_02").List = Lists(listName)
        PrintError ("Set Environment List")
        On Error GoTo 0
End Sub
```

Example: Changing Field Properties when a Field Changes

This example demonstrates how you can change the properties of a field when a different field is changed.

In this example, if the status of the defect (**BG_STATUS**) is changed to `Closed`, the user must provide a value in the field **Closed in Build** (**BG_CLOSING_VERSION**).

Add the code to the `Bug_FieldChange` event procedure, to make the **Closed in Build** field a required field if the status is changed to `Closed`.

```
Sub Bug_FieldChange(FieldName)
        On Error Resume Next
        If FieldName= "BG_STATUS" then
            If Bug_Fields("BG_STATUS").value="Closed" then
```

```
                Bug_Fields("BG_CLOSING_VERSION").IsRequired=True
            Else
                Bug_Fields("BG_CLOSING_VERSION").IsRequired=False
            End If
        Else
            ' Enter your code here.
        End If
        PrintError "Bug_FieldChange"
        On Error GoTo 0
End Sub
```

Example: Controlling User Permissions

This example demonstrates how you can prevent members of specific user groups from performing an action.

The code allows a user to replace a defect field value only if the user belongs to the `Admin` user group.

Add the code to the ActionCanExecute event procedure so that the check is performed when a user attempts to execute an action.

```
Function ActionCanExecute(ActionName)
        ' Initialize the function's return value
       ' to avoid unpredictable behavior.
        ActionCanExecute = False
        On Error Resume Next
        If ActionName = "UserDefinedActions.BugReplaceAction1" _
            And Not User.IsInGroup("Admin") then
            ActionCanExecute = False
            msgbox "You do not have permission to perform this action"
        Else
            ActionCanExecute = True
        End If
        PrintError "ActionCanExecute"
        On Error GoTo 0
End Function
```

Example: Adding Button Functionality

This example opens a calculator when a user clicks a button defined with action name `Calculator`. For more information about adding user-defined buttons, see "Adding a Button to a Toolbar" on page 422.

Add the code to the `ActionCanExecute` event procedure, so that it is triggered when a user initiates an action.

For information about the **Wscript.Shell** object, refer to the Microsoft documentation. To access help for the VBScript language, choose **Help > VBScript Home Page** in the Script Editor.

```
Function ActionCanExecute(ActionName)
        ' Initialize the function's return value to
```

```
        ' avoid unpredictable behavior.
        ActionCanExecute = DefaultRes
        On Error Resume Next
        If ActionName = "UserDefinedActions.Calculator" Then
            Set shell = CreateObject("Wscript.Shell")
            shell.Run "Calc"
            Set shell = Nothing
        End If
        ActionCanExecute = DefaultRes
        PrintError "ActionCanExecute"
        On Error GoTo 0
End Function
```

Example: Error Handling

This example demonstrates how you can display a standard error message. Error handling should be added to each workflow script that you write, because errors that are not detected by the workflow code can cause the user's browser to crash.

The user-defined function PrintError receives the name of the calling procedure as a parameter. If an error has occurred, PrintError prints out the error number, description and severity, and the name of the procedure in which the error occurred.

You do not need to create an **Err** object, because it is intrinsic to VBScript. For more information about the **Err** object, refer to the Microsoft documentation.

```
Sub PrintError(strFunctionName)
        If Err.Number <> 0 Then
            MsgBox "Error #" & Err.Number & ": " & Err.Description, _
            vbOKOnly+vbCritical, _
            "Workflow Error in Function " & strFunctionName
        End If
End Sub
```

The following code segment illustrates how you can add error handling to your subroutines.

```
Sub <sub_name>()
        On Error Resume Next
        ...
        [Your code here]
        ...
        PrintError "<sub_name>"
End Sub
```

The following code segment illustrates how you can add error handling to your functions.

```
Function <function_name>()
        On Error Resume Next
        ...
        [Your code here]
        ...
        PrintError "<function_name>"
End Function
```

Example: Obtaining Session Properties

This example demonstrates how to use the **TDConnection** object to obtain the properties of the current session. Add the code to the procedure where these properties are needed. The properties do not depend on each other, so each of the properties can be retrieved separately.

The following are examples of session properties:

```
TDConnection.ServerName
TDConnection.ServerTime
TDConnection.DomainName
TDConnection.ProjectName
User.UserName
```

Note that there is no need to use **TDConnection** to retrieve the user name because the workflow has a predefined **User** object. For more information, see "TDConnection Object" on page 464.

The example below tests the first five characters of the server URL to determine whether the user is connected to the server using HTTP or HTTPS:

```
If Left(UCase(TDConnection.ServerName), 5) = "HTTPS" Then
        MsgBox "You are currently connected to the server using SSL."
Else
        MsgBox "You are not using SSL."
End If
```

Example: Sending Mail

These examples demonstrate how to use the **TDConnection** object to send mail when a defect is submitted, and to send mail when a field value changes in the Test Plan module.

Sending Mail when a Defect is Submitted

This example sends mail when a defect is submitted.

Add a call to the `SendDefect` procedure in the `Bug_AfterPost` event procedure.

> **Note:** If the `SendDefect` procedure is called before the defect is submitted, the values that were changed in the current modification will not be included. The database is updated with the new values only after the defect is posted.

```
Sub SendDefect (iObjectId, strTo, strCc, strSubject, strComment)
        On Error Resume Next
        Dim objBugFactory, objBug
        Set objBugFactory = TDConnection.BugFactory
        Set objBug = objBugFactory.Item(iObjectId)
        objBug.Mail strTo, strCc, 2, strSubject, strComment
        Set objBug = Nothing
        Set objBugFactory = Nothing
        PrintError "SendDefect"
```

```
        On Error GoTo 0
End Sub
```

The constant `2` in the call to `objBug.Mail` indicates that the history should be included with the mail. For a list of the constants that can be used to customize email, refer to the `tagTDMAIL_FLAGS` enumeration in the *HP ALM Open Test Architecture API Reference*. In workflow scripts, use numeric constants and not the enumeration values.

Sending Mail when a Test Plan Module Field Value Changes

The example below demonstrates mail notification when the value of the status field is changed in the Test Plan module.

The code is added to the `Test_FieldChange` event procedure. It constructs a subject and comment for the email, and calls a user-defined function, `SendTest`. SendTest sends mail from the Test Plan module. You can code `SendTest` similarly to the `SendDefect` subroutine shown in "Sending Mail when a Defect is Submitted" on previous page.

```
Sub Test_FieldChange(FieldName)
        On Error Resume Next
        Dim strSubject, strComment
        If FieldName = "TS_STATUS" Then
            strSubject = "Test Change Notification" & _
                " for project " & TDConnection.ProjectName & _
                " in domain " & TDConnection.DomainName
            strComment = "The user " & User.FullName & _
                " changed the status of the test " & _
                Test_Fields("TS_NAME").Value & _
                " to " & Test_Fields("TS_STATUS").Value
            SendTest Test_Fields("TS_TEST_ID").Value, _
                Test_Fields("TS_RESPONSIBLE").Value, "[QA Testers]", _
                strSubject, StrComment
        End If
End Sub
```

Example: Storing the Last Values Entered

This example shows how to use the **TDConnection** object to implement persistent data between actions. The lifetime of a variable in a routine is only for the routine run. Therefore, persistent data must be stored if it must be available later. It is recommended that you use the ALM API to store persistent data whenever possible instead of using external objects, files, or the registry.

In this example, a user-defined function `SW_KeepLastValue` uses the **Settings** object to save the values typed into the fields **BG_DETECTION_VERSION**, **BG_USER_01**, and **BG_USER_03** when a user posts a defect. These values are retrieved and assigned as default values when this user adds a new defect.

The user-defined function is called with the `SET` action from `Bug_CanPost`, before a new defect is posted by the user. The values in the fields are stored.

```
Function Bug_CanPost()
        ' Initialize the function's return value to
        ' avoid unpredictable behavior.
        Bug_CanPost = True
        If Bug_Fields("BG_BUG_ID").Value = "" Then
            SW_KeepLastValue ("SET")
        End If
End Function
```

The function is called with the `GET` action from the `Bug_New` event procedure. When a user adds a new defect, the values stored in the fields for this user are entered into these fields.

```
Sub Bug_New()
        SW_KeepLastValue ("GET")
End Sub
```

Depending on the action passed as a parameter, the user-defined function `SW_KeepLastValue` stores the values of the fields in the common settings table for the current user, or reads the values from the **Settings** object and assigns the values to the appropriate fields.

```
Sub SW_KeepLastValue(action)
Dim tdc, vals, flds
Dim uset, pairs, pair
Dim bld
On Error Resume Next
        bld = ""
        Set tdc = TDConnection
        Set uset = tdc.UserSettings
        If action = "SET" Then
            flds = Array("BG_DETECTION_VERSION", _
            "BG_USER_01", "BG_USER_03")
            vals = ""
            For i = 0 To UBound(flds)
                If vals <> "" Then vals = vals & ";"
                vals = vals & flds(i) & "=" & _
                Bug_Fields(flds(i)).Value
            Next
            'Open category KeepLValueSetting
            uset.Open ("KeepLValueSetting")
            'Setting KeepValueFields in category KeepLValueSetting
            uset.Value("KeepValueFields") = vals
            uset.Close
        End If 'SET
        If action = "GET" Then
            uset.Open ("KeepLValueSetting")
            vals = uset.Value("KeepValueFields")
            If vals <> "" Then
                pairs = Split(vals, ";")
                For i = 0 To UBound(pairs)
                    pair = Split(pairs(i), "=")
                    If UBound(pair) = 1 Then
                        Select Case pair(0)
```

```
                            Case "BG_USER_03"
                                bld = pair(1)
                            Case Else
                                If Bug_Fields(pair(0)).Value = "" Then
                                   Bug_Fields(pair(0)).Value = pair(1)
                                End If
                        End Select
                        If Bug_Fields("BG_DETECTION_VERSION").Value _
                        <> ""
                        And bld <> "" Then
                            SW_SetLists_VersionsBuilds _
                            "BG_DETECTION_VERSION", _
                            "BG_USER_03"
                            Bug_Fields("BG_USER_03").Value = bld
                            If Err.Number <> 0 Then Err.Clear
                        End If 'Bug_Fields
                    End If 'UBound(pair)
                Next
            End If 'vals <> ""
        End If 'GET
        uset.Close
        PrintError ("Keep Last Value (" & action & ")")
        On Error GoTo 0
End Sub
```

Example: Copying Field Values to Another Object

This example shows how to use the **TDConnection** object to copy the value from the **Build Number** field of a Run (**RN_USER_02**) to the **Last Ran On Build** field of a Test in a Test Set (**TC_USER_03**).

Add the code to the `Run_AfterPost` event procedure.

```
Sub Run_AfterPost
        On Error Resume Next
        Dim tdc
        set tdc = TDConnection
        Dim TSFact 'As TestSetFactory
        Set TSFact = tdc.TestSetFactory
        Dim TstSet 'As TestSet
        Set TstSet = TSFact.Item(Run_Fields("RN_CYCLE_ID").Value)
        MsgBox TstSet.Name
        Dim TSTestFact 'As TSTestFactory
        Set TSTestFact = TstSet.TSTestFactory
        Dim TSTst 'As TSTest
        Set TSTst = _
        TSTestFact.Item(Run_Fields("RN_TESTCYCL_ID").Value)
        MsgBox TSTst.Name

        TSTst.Field("tc_user_03").value = _
        Run_Fields("RN_USER_02").Value
```

```
        TSTst.Post

        PrintError ("Run_AfterPost")
        On Error GoTo 0
End Sub
```

Part 4: Appendix

Chapter A: Upgrade Preparation Troubleshooting

This appendix describes schema and database inconsistencies that the verification process detects. It indicates which problems the repair process can fix automatically, and which you should repair manually. Suggested solutions for repairing each issue are provided.

For information on performing the verification, repair, and upgrade processes, see "Upgrading Projects to a New Version" on page 102.

This chapter includes:

Overview 495
Quick Warning Reference 495
General Validation 501
Schema Validation 504
Data Validation 514
Changing the Database User Schema 519

Overview

During an upgrade, the verification process, described under "Verifying Domains and Projects" on page 107, detects inconsistencies and indicates which problems the repair process can fix automatically, and which you should repair manually. Suggested solutions for repairing each issue are provided in this appendix.

If an error is displayed during the verification or upgrade process, you can see error descriptions at http://support.openview.hp.com/selfsolve/document/KM1302383.

If a warning is displayed during the verification process, you can use the "Quick Warning Reference" below to locate the corresponding solution for that warning.

Some solutions necessitate the changing of the database user schema:

- **Database User Schema.** Database in SQL Server and a user schema in Oracle. This term is used for both cases because ALM can be deployed over SQL Server and Oracle. Both cases are logical sets of database objects (for example, tables, indexes, and so on) owned by the same logical owner.
- **Expected Database User Schema.** ALM Database User Schema configurations, as defined in the configuration file for a new ALM Database User Schema. As a preparation for the current version, each project database user schema should be aligned with the latest configurations, as defined in this schema.

If you need to modify the database user schema, see the additional instructions under "Changing the Database User Schema" on page 519.

Quick Warning Reference

This section lists schema and data issues found in warnings generated by the verification process.

General Issues 495
Schema Issues 496
Data Issues 500

General Issues

The following table lists general issues found in verification process warnings. Some issues are fixed automatically by the repair process. Other issues require that you repair them manually.

Type	Problem	Element	Resolution	Details
Database	Database server version not supported		manual repair	"General Validation" on page 501

Type	Problem	Element	Resolution	Details
Database	Schema name contains invalid characters		manual repair	"Valid Database User Schema Name" on page 501
Database	Table owner does not match the ALM server connection method		manual repair	"Mixed Table Ownership" on page 501
Database	Repository over database feature no longer supported		manual repair	"Repository over Database Feature" on page 502
Version control	Certain version control projects cannot be upgraded directly		manual repair	"Version Control Validation" on page 502
Database	Permissions		manual repair	"Database Permissions" on page 502
Database	Configure text search		manual repair	"Text Search Configuration" on page 503

Schema Issues

The following table lists schema issues found in verification process warnings. Some schema issues are fixed automatically by the repair process. Other schema issues require that you repair them manually.

Type	Problem	Element	Resolution	Details
Table	Extra table		manual repair	"Extra Table" on page 505
Table	Missing table		repair process	"Missing Table" on page 506
Views	Extra view		manual repair	"Extra Views" on page 516
Views	Missing view		repair process	"Views" on page 516
Column	Extra column		manual repair	"Extra Column" on page 506
Column	Missing column		repair process	"Missing Column" on page 508

Type	Problem	Element	Resolution	Details
Column	Size mismatch - column size bigger than expected		manual repair	"Column Size Mismatch" on page 507
Column	Size mismatch - column size smaller than expected		repair process	"Column Size Mismatch" on page 507

Column	Size mismatch - internal Quality Center change	`COMMON_SETTINGS.CSET_NAME`
		`REQ.RQ_REQ_TYPE`
		`REQ.RQ_REQ_AUTHOR`
		`REQ.RQ_REQ_PRODUCT`
		`REQ.RQ_REQ_REVIEWED`
		`REQ.RQ_REQ_STATUS`
			repair process	"Column Size Mismatch" on page 507

Column	Type mismatch		manual repair	"Column Type Mismatch" on page 507
Column	Precision		repair process	"Column Precision Mismatch" on page 507
Column	Nullable - column can accept NULL values		repair process	"Column Nullability Mismatch" on page 507
Index	Uniqueness		repair process	"Index Uniqueness Mismatch" on page 509
Index	Clustered		repair process	"Index Clustered" on page 510
Index	Extra		manual repair	"Internal Quality Center Changes" on page 512

Type	Problem	Element	Resolution	Details
Index	Extra - internal Quality Center changes	BUG.BG_ DETECTED_BY_ LWR_IDX BUG.BG_STATUS_ LWR_IDX BUG.BG_ RESPONSIBLE_ LWR_IDX BUG.BG_ DETECTED_BY_ LWR_IDX	repair process	"Internal Quality Center Changes" on page 512
Function based index	Extra - internal Quality Center changes	COMMON_ SETTINGS.CS_ COVER_LWR_IDX HOSTS.HOSTS_ LWR_IDX HOSTS_IN_GROUP. HG_COVER_LWR_ IDX HOST_GROUP. GH_ LWR_IDX USERS.US_USERS_ LWR_IDX	repair process	"Internal Quality Center Changes" on page 512
Index	Missing		repair process	"Missing Index" on page 510

Type	Problem	Element	Resolution	Details
Index	Missing - internal Quality Center changes	ALL_LISTS.AL_ABS_PATH_COV_IDX BUG.BG_COMPOUND_IDX CYCLE.CY_FOLDER_IDX REQ.RQ_REQ_STATUS_IDX RUN.RN_CYCLE_IDX STEP.ST_RUN_IDX TEST.TS_SUBJECT_IDX	repair process	"Internal Quality Center Changes" on page 512
Constraint	Missing		repair process	"Missing Constraint" on page 510
Constraint	Extra		manual repair	"Missing Constraint" on page 510
Index	Index changed internally	REQ_COVER.RC_ENTITY_ID_IDX RUN.RN_TEST_ID_IDX RUN.RN_TESTCYCLE_IDX	repair process	"Index Changed" on page 510
Index	Changed		repair process	"Index Changed" on page 510
Triggers	Extra		manual repair	"Extra Trigger" on page 511
Sequence	Missing		repair process	"Missing Sequence" on page 512
Sequence	Extra		manual repair	"Extra Sequence" on page 511
Sequence	Incorrect		repair process	"Incorrect Sequences" on page 512

Data Issues

The following table lists data issues found in the verification process warnings. All data issues are fixed automatically by the repair process.

Type	Problem	Element	Res-olution	Details
Duplicate data	Duplicate values		repair process	"Duplicate Values" on page 515
Duplicate data	Duplicate IDs		repair process	"Duplicate IDs" on page 515
Trees	Wrong number of children	`Tables REQ/ALL_LISTS/CYCL_FOLD`	repair process	"Tree Incon-sistencies" on page 516
Trees	Corrupted path	`Tables REQ/ALL_LISTS/CYCL_FOLD`	repair process	"Tree Incon-sistencies" on page 516
Trees	Orphan records	`Tables REQ/ALL_LISTS/CYCL_FOLD`	repair process	"Tree Incon-sistencies" on page 516
Sequence-s	Sequence mismatch	`Table SEQUENCES`	repair process	"Sequences" on page 511
Orphans	Missing parent entities		repair process	"Orphaned Entities" on page 517
Missing data	Missing entities		repair process	"Missing Entities" on page 517
Lists	Missing lists and values	`Tables SYSTEM_FIELD / LISTS`	repair process	"Missing Lists and/or List Values" on page 517
Encryption	Mismatched pass-phrases for encrypted values	`Tables LAB_HOSTS / LAB_AUT_HOSTS / LAB_DIAGNOSTICS_SERVERS`	manual repair	"Encrypted Values" on page 518

General Validation

This sections describes the general validation checks the verification process performs.

Supported Database Version 501
Valid Database User Schema Name 501
Mixed Table Ownership 501
Repository over Database Feature 502
Version Control Validation 502
Database Permissions 502
Text Search Configuration 503

Supported Database Version

The verification process checks that the project schema is stored in a supported database server. If the verification process detects that the database server version is not supported, it displays a warning. For details about the database servers versions supported by ALM, refer to http://www.hp.com/go/TDQC_SysReq.

Valid Database User Schema Name

The upgrade mechanism does not support databases that include special characters in the database name. If the verification process finds special characters, you must remove them.

To remove special characters from database names:

1. Deactivate the project.
2. Ask your database administrator to rename the database user schema to a name that does not include special characters.
3. Remove the project from Site Administration.
4. Update the **Dbid.xml** file to point to the new database user schema name.
5. Restore the project by using the updated **Dbid.xml** file.
6. Run the verification process again to make sure the problem is resolved.

Mixed Table Ownership

ALM can connect to Microsoft SQL server by using SQL authentication or Windows authentication.

For each of these methods, a different user owns the project's tables:

- **SQL Authentication.** Table owner is the user `td`.
- **Windows Authentication.** Table owner is the user `dbo` (a user mapped to the operating system

user that runs the ALM server).

If you create a project with one type of authentication (for example, SQL), and then restore it with the other type of authentication (for example, Windows), these tables cannot be accessed. In this case, new tables are created with owners that are different from those of the old tables. You will not be able to work with the project. It is likely that the upgrade will fail.

To prevent this problem, the duplicate ownership validator checks that the owner of all of the tables in the project database user schema matches the connection method that ALM is using to connect to the server.

To fix table ownership manually, do one of the following:

- **SQL Authentication:** Run the following query to make `td` the table owner:

  ```
  EXEC sp_changeobjectowner '<table name>', 'td'
  ```

- **Windows Authentication:** Run the following query to make `dbo` the table owner:

  ```
  EXEC sp_changeobjectowner 'td.<table name>', 'dbo'
  ```

Repository over Database Feature

The **Repository over Database** feature is not supported in ALM versions 11.00 and later.

If you use this feature in Quality Center 9.2, you should migrate the repository from the database to the file system (available from Quality Center 9.2 Patch 12) before upgrading the project to Quality Center 10.0, and then upgrade the project to ALM 11.00.

For more information about the tool for migrating the project repository from the database to the file system, see the ReadMe files for Quality Center 9.2 Patch 12. The verification process checks whether the project is using the **Repository over Database** feature. If the project is using the feature, the validator displays a warning.

Version Control Validation

- **Legacy version control projects.** Integration with external version control tools is not supported in ALM 11.52. Quality Center version 10.00 and ALM version 11.52 include a built-in version control functionality to support your projects. To work with projects from Quality Center 9.2 that use version control, you must first upgrade to Quality Center 10.00, migrate legacy version control data, and then upgrade to ALM 11.52.
- **Quality Center 10.00 version control enabled projects.** Version control enabled projects from Quality Center 10.00 cannot be upgraded to ALM 11.52 while there are checked out entities. The verification process checks that there are no checked out entities. If there are checked out entities, they must be checked in in Quality Center 10.00.

Database Permissions

To enable an upgrade to the current ALM version, the project schema requires a set of minimum required permissions. The verification process makes sure that both the project user and the administrator user have all the privileges needed to perform the upgrade. For more information about

the minimum permissions required for an ALM schema, see the *HP Application Lifecycle Management Installation and Upgrade Guide*.

Text Search Configuration

Quality Center versions 9.0 and later support the database text search feature. However, not all databases are configured to support this feature. If your database does support text search, ALM installs the required components when creating a new project database. ALM also activates the text search for the new database. The verification process checks whether your project has the text search feature enabled, and that it is configured correctly.

The verification process validates the following:

- "Validity of the Text Search Configuration" below
- "Only Valid Fields Configured Under "Text Search"" below
- "Text Search Validation for Oracle Database Server" on next page
- "Text Search Validation for Microsoft SQL Database Server " on next page

Validity of the Text Search Configuration

The verification process checks that text search components are installed and are valid on the database server. If a database server is text search-enabled in the DB Servers tab in Site Administration, text search must also be enabled on the Oracle or SQL database server. If the verification process detects that text search is not enabled or configured incorrectly on the Oracle or SQL database server, the upgrade process does not run until you manually repair the problem.

We recommend that you ask your database administrator to reconfigure text search on the Oracle or SQL database server. Alternatively, as a workaround, you can disable text search for the database server from Site Administration.

To disable the text search for the database server:

1. Run the following query on your Site Administration schema:

   ```
   update <SA Schema>.dbservers set db_text_search_enabled = null
   where dbserver_name = '<DB logical name>'
   ```

2. Restart the ALM server.
3. Run the repair process for your projects.
4. When the repair process completes, run the following query:

   ```
   update <SA Schema>.dbservers set db_text_search_enabled = 'Y' where
   dbserver_name = '<DB logical name>'
   ```

5. Restart the ALM server.

Only Valid Fields Configured Under "Text Search"

The verification process checks that only valid fields are defined as searchable. You can enable the text search only for specific entities, and only on fields of the type string or memo. The following entities are supported: BUG, COMPONENT, COMPONENT_STEP, DESSTEPS, REQ, TEST, BPTEST_TO_COMPONENT, and CYCLE. Any other configuration could cause functionality problems during upgrade or customization. This problem is fixed automatically by the repair

process.

Text Search Validation for Oracle Database Server

For an Oracle Database server, the verification process checks the following:

- **Validity of Text Search Indexes.** The verification process checks that database text search indexes are valid. Invalid text search indexes can cause functionality problems and even upgrade failure in ALM. If the verification process detects an invalid index, try to recreate the index by dropping it from the schema and creating it again. In Site Administration, click the **Site Projects** tab. Select the relevant project and click the **Enable/Rebuild Text Search** button. If this procedure returns an error, consult your database administrator or contact HP Support.
- **Validity of Project Database User Permissions.** The verification process checks that the project database user has the required permissions to work with text search. When text search is installed on the database, the role CTXAPP is created automatically. ALM requires that this role be granted to all projects database users that support text search. (ALM grants the CTXAPP role automatically when creating the project or enabling the text search for a project.) If this role is not granted to the project database user (configured to support text search), the verification process returns a warning. In these cases, ask your database administrator to grant the required role to the project database user.

Text Search Validation for Microsoft SQL Database Server

The verification process checks that the project database user schema enables the text search feature. To work with text search on SQL project, you need to enable the text search on the database.

To enable text search on the database:

1. Select the database from the SQL server Enterprise Manager.
2. Right-click the database name.
3. Select **Properties/Files**.
4. Select **Use Full-Text Indexing**.

Schema Validation

The verification process helps to ensure that the project database user schema is correct and configured as expected.

The verification process performs two types of schema verifications:

- **Schema Correctness.** Checks that the project database schema includes all of the required schema objects, as defined in the expected database user schema for the project. This verification ensures that all of the required entities exist and are defined as expected. It also ensures that there are no extra entities defined on top of the schema.
- **Alignment to the current version.** Notifies you about differences in the project database user schema caused by internal changes made in Quality Center 10.00. In this way, the verification process aligns the schema with the latest internal changes to the schema made in preparation for the upgrade.

The verification process displays warnings in the verification report if it finds the following:

- Extra entities defined. For example, Table, Column, Trigger, View, and Sequence for an Oracle Database.
- Differences from the expected definitions. For example, Column Size and Index Attributes.
- Missing objects.

Schema differences found by the verification process can cause upgrade failures or usage problems. As long as the verification process still finds these differences, an upgrade to the current ALM version will not start.

Note: Many of the schema changes can be fixed automatically by the repair process.

The following sections contain possible warnings, grouped by the different database objects, that the verification process can display in the verification report:

Tables 505
Columns 506
Indexes and Constraints 508
Triggers 511
Sequences 511
Internal Quality Center Changes 512

Tables

Database tables can contain the following warnings:

- "Extra Table" below
- "Missing Table" on next page

Extra Table

The ALM schema should contain only the tables that are defined in the schema configuration file. Adding extra tables on top of the schema is not supported and might cause future problems with ALM.

Problem: If the verification process finds extra tables that were added manually to the schema, it generates an **Extra Table** warning.

Note: This problem requires manual repair. The repair process cannot fix it.

Solution: Do one of the following:

- **Change the Schema.** If you use the table, copy it to a different schema. If you do not use the table, delete it. Before taking either action, back up the schema and consult your database administrator. For details, see "Changing the Database User Schema" on page 519.

- **Use the Exception File.** Not recommended: Instruct the upgrade to ignore this problem. For more information about the Exception file, see "Defining an Exception File" on page 116.

> **Note:** If the project database is case sensitive, the table name must be the same in both the database and the exception file.

Missing Table

The verification process checks that all of the tables defined for the project schema actually exist (according to the tables of each Quality Center/ALM version).

Problem: If a table is missing, the verification process generates a **Missing Table** warning.

Solution: Do one of the following:

- See "Changing the Database User Schema" on page 519.
- Run the repair process to create the missing table. Although you can use the repair process to add these objects, we recommend that you contact HP Support to make sure that the missing objects are not just symptoms of a bigger problem.

Columns

Database columns can contain the following warnings:

- "Extra Column" below
- "Column Size Mismatch" on next page
- "Column Precision Mismatch" on next page
- "Column Type Mismatch" on next page
- "Column Nullability Mismatch" on next page
- "Identity Column" on page 508
- "Missing Column" on page 508

Extra Column

The verification process checks that each table includes the required columns, as defined for the expected database user schema and version. The schema should not include extra columns. Extra columns in a table might cause upgrade failure or functionality problems.

Problem: If the verification process detects an extra column (that does not exist in the database user schema definitions) in one of tables, it generates an **Extra Column** warning.

> **Note:** This problem requires manual repair. The repair process cannot fix it.

Solution: Do one of the following:

- **Change the Schema.** If you have an internal implementation that requires extra table columns, move the extra columns to a different table in a different schema. If you do not use a particular column, delete it. Before taking either action, back up your schema and consult your database administrator. For a more detailed explanation, see "Changing the Database User Schema" on

page 519.

- **Use the Exception File.** Not recommended: Instruct the upgrade to ignore this problem. For more information about the Exception file, see "Defining an Exception File" on page 116.

Column Size Mismatch

The verification process checks that all the table columns are defined as expected. This validation ensures that the column size matches the expected size as defined for each table column. This verification excludes user-defined fields, whose size can be customized through project customization.

Some column mismatch warnings are caused by internal changes made in Quality Center 10.00 that are fixed by the repair process automatically. For details, see "Internal Quality Center Changes" on page 512.

Problem A: Size is bigger than expected. If the column size is bigger than expected, decrease the column size to the required size manually. Because this operation can cause data loss, it is not performed automatically by repair process.

> **Note:** This problem requires manual repair. The repair process cannot fix it.

Solution A: Consult your database administrator to resolve this issue. For risks involved in changing the database user schema, see "Changing the Database User Schema" on page 519.

Problem B: Size is smaller than expected. If the column size is smaller than expected, the repair process fixes the problem automatically by increasing the column size to the expected size.

Solution B: Run the repair process to increase the current size to the required size.

Column Precision Mismatch

In an Oracle Database, "precision" is the term used to define the size of fields with the INTEGER type.

Problem: The verification process generates a warning if the precision defined for a certain column is smaller than expected.

Solution: Run the repair process to increase the current precision to the required precision.

Column Type Mismatch

Changing a column type can cause the upgrade to fail and can cause major functionality problems.

Problem: The verification process generates a **Column Type** warning if the column type has changed.

> **Note:** This problem requires manual repair. The repair process cannot fix it.

Solution: Consult your database administrator to resolve this issue. For risks involved in changing the database user schema, see "Changing the Database User Schema" on page 519.

Column Nullability Mismatch

One of the attributes that is defined for a column is whether it can accept null values. A null is the absence of a value in a column of a row. Nulls indicate missing, unknown, or inapplicable data. If

you have defined a NOT NULL or PRIMARY KEY integrity constraint for a particular column, you cannot insert rows into the column without adding a value.

Problem: The verification process compares the required definitions for each column in the expected database user schema to the project database user schema. If it encounters differences in the column NULL attribute definition, it generates a **Column Nullable** warning.

Solution: Run the repair process. The repair process runs a query to modify the column attributes to the expected attributes.

If the column includes NULL values, the repair process cannot update the column attribute to NOT NULL (if this is the required attribute) for the column. Ask your database administrator how to remove the NULL values from the column. After removing the NULL values, run the repair process again. For details, see "Changing the Database User Schema" on page 519.

Identity Column

The IDENTITY property is one of the attributes defined for columns in Microsoft SQL server.

Problem: As part of the verification for the columns attributes, the verification process might find a column Identity property that is not configured as expected.

> **Note:** This problem requires manual repair. The repair process cannot fix it.

Solution: Change the IDENTITY property of the column to the expected configuration (according to the output from the verification process report) manually. Consult your database administrator to resolve this issue. For details, see "Changing the Database User Schema" on page 519.

Missing Column

If a column is missing from a table, run the repair process or contact HP Support.

Problem: If the verification process finds that a column is missing from one of the tables, it generates a **Missing Column** warning.

Solution: Do one of the following:

- Run the repair process to fix the problem.
- See "Changing the Database User Schema" on page 519.

Indexes and Constraints

A database index is a data structure that improves the speed of operations in a table. You can create indexes using one or more columns, providing the basis for both rapid random lookups and efficient ordering of access to records. Database Constraints are constraints on the database that require relations to satisfy certain properties.

Database indexes and constraints can cause the following validation warnings:

- "Extra Index" on next page
- "Extra Constraint" on next page
- "Index Uniqueness Mismatch" on next page

- "Index Clustered" on next page
- "Missing Constraint" on next page
- "Missing Index" on next page
- "Index Changed" on next page
- "Index Order Changed " on next page

Extra Index

The ALM schema should include only those indexes defined in the required schema configurations.

Problem: If the verification process finds an index that is not defined in the required schema configuration, it generates an **Extra Index** warning.

> **Note:** This problem requires manual repair. The repair process cannot fix it.

Solution: Remove the extra indexes manually. Consult with your database administrator to resolve this issue. For details, see "Changing the Database User Schema" on page 519.

Some **Extra Index** warnings are caused by internal changes made in Quality Center 10.00. These extra indexes are no longer used by ALM, and are removed by the repair process. For details, see "Internal Quality Center Changes" on page 512.

Extra Constraint

The ALM schema should include only those constraints defined in the required schema configurations.

Problem: If the verification process finds a constraint that is not defined in the required schema configuration, it generates an **Extra Constraint** warning.

> **Note:** This problem requires manual repair. The repair process cannot fix it.

Solution: Remove the extra constraint manually. Consult with your database administrator to resolve this issue. For details, see "Changing the Database User Schema" on page 519.

Index Uniqueness Mismatch

A unique index guarantees that the index key contains no duplicate values. As a result, every row in the table is unique. Specifying unique indexes on ALM data tables ensures data integrity of the defined columns. In addition, it provides helpful information that is used as a query optimizer.

Problem: If the index uniqueness attribute does not have the expected value, the verification process generates an **Index Uniqueness Mismatch** warning.

You cannot create a unique index, unique constraint, or PRIMARY KEY constraint if duplicate key values exist in the data. The verification process performs these data validations. If a table has duplicate values or IDs, based on the index definitions on that table, the verification process also displays the duplication in the verification report. In this case, the repair process automatically fixes the duplication problem before creating the unique index.

Solution: Run the repair process to fix the problem.

Index Clustered

In Microsoft SQL, index type can be classified as clustered or non-clustered. The verification process compares the required definitions for each index in the expected database user schema to the project database user schema.

Problem: If the verification process finds differences in the index clustered attribute definition, it generates an **Index Clustered** warning.

Solution: Run the repair process to fix the problem.

Missing Constraint

Constraints are rules that the database enforces to improve data integrity.

Problem: If the verification process finds a constraint that should be defined as missing, it generates a **Missing Constraint** warning.

Solution: Run the repair process to fix the problem.

Missing Index

The verification process checks that all the required indexes (as defined in the expected database user schema) exist in the projects database user schema.

Problem: If the verification process does not find all the required indexes in the projects database user schema, it generates a **Missing Index** warning.

Solution: Run the repair process to fix the problem.

Index Changed

The verification process checks that the indexes are defined according to the expected database user schema.

Problem: If the verification process finds an index that is not defined according to the expected database user schema, it generates an **Index Changed** warning.

This warning can indicate the following problems:

- Function in a function-based index is different than expected.
- Index is not defined on the expected columns.

Solution: Run the repair process to fix the problem. The repair process removes the index, and then recreates it, based on the required definitions for this index.

Index Order Changed

The verification process checks that the order of the columns in the index definition has not changed.

Problem: If the order of the columns in the index definition has changed, the verification process generates an **Index Order Changed** warning.

Solution: Run the repair process to fix the problem. The repair process removes the index, and then recreates it, based on the required definitions for this index.

Triggers

A database trigger is procedural code that is automatically executed in response to certain events on a particular table in a database.

Database triggers can contain the following warning:

- "Extra Trigger" below

Extra Trigger

Extra triggers can cause upgrade failures and functionality problems.

Problem: If the verification process finds an extra trigger, it generates an **Extra Trigger** warning.

> **Note:** This problem requires manual repair. The repair process cannot fix it.

Solution: Before upgrading, back up your database schema and remove the extra triggers manually.

Because extra triggers can cause upgrade failures, the upgrade process cannot ignore this warning by using the Exception file. For details, see "Changing the Database User Schema" on page 519.

Sequences

A sequence is an Oracle object that acts as a generator that provides a sequential series of numbers.

Database sequences can contain the following warnings:

- "Extra Sequence" below
- "Missing Sequence" on next page
- "Incorrect Sequences" on next page

Extra Sequence

ALM schemas should contain only the sequences that are defined in the schema configuration file.

Problem: If the verification process finds an extra trigger, it generates an **Extra Sequence** warning.

> **Note:** This problem requires manual repair. The repair process cannot fix it.

Solution: Do one of the following:

- **Change the Schema.** Move the sequence to a new database user schema. Before doing so, consult with your database administrator. For details, see "Changing the Database User Schema" on page 519.
- **Use the Exception File.** Not recommended: Instruct the upgrade to ignore this problem. For more information about the Exception file, see "Defining an Exception File" on page 116.

Missing Sequence

Problem: If the verification process finds that one of the sequences that should be defined on the ALM schema is missing, it generates a **Missing Sequence** warning.

Solution: Do the following:

- Run the repair process to fix the problem.
- See "Changing the Database User Schema" on page 519.

Incorrect Sequences

Problem: Sometimes the Oracle object sequence numbers become incorrect, for example, if an export of the database is done on a live activated project, in which users are still modifying tables. If the verification process finds that Oracle sequences objects are not fully synchronized with ALM schema table IDs, the verification process generates an **Incorrect Oracle sequences found** warning.

Solution: Run the repair process to fix the problem.

Internal Quality Center Changes

For upgrade from Quality Center 9.2: As a result of internal changes in Quality Center 10.00, a set of updates needs to be applied to the schema as part of the preparation for the upgrade to ALM.

To apply the updates to the schema, perform the following processes:

- "Verification Process" below
- "Repair Process" on page 514

Verification Process

If the verification process finds any internal differences, it generates warnings in the verification report. The repair process fixes them automatically.

The verification process checks for the following internal changes:

Type	Problem	Element	Comment
Column	Size mismatch	COMMON_ SETTINGS.CSET_NAME	Expected column size is 240. Actual size is 70.
Column	Size mismatch	REQ.RQ_REQ_ PRIORITY	Expected column size is 255. Actual size is 70.
Column	Size mismatch	REQ.RQ_REQ_TYPE	Expected column size is 255. Actual size is 70.
Column	Size mismatch	REQ.RQ_REQ_AUTHOR	Expected column size is 255. Actual size is 70.
Column	Size mismatch	REQ.RQ_REQ_PRODUCT	Expected column size is 255. Actual size is 70.

Type	Problem	Element	Comment
Column	Size mismatch	REQ.RQ_REQ_REVIEWED	Expected column size is 255. Actual size is 70.
Column	Size mismatch	REQ.RQ_REQ_STATUS	Expected column size is 255. Actual size is 70.
Index	Missing	ALL_LISTS.AL_ABS_PATH_COV_IDX	
Index	Missing	BUG.BG_COMPOUND_IDX	
Index	Missing	CYCLE.CY_FOLDER_IDX	
Index	Missing	REQ.RQ_REQ_STATUS_IDX	
Index	Missing	RUN.RN_CYCLE_IDX	
Index	Missing	STEP.ST_RUN_IDX	
Index	Missing	TEST.TS_SUBJECT_IDX	
Index	Extra	BUG.BG_DETECTED_BY_LWR_IDX	
Index	Extra	BUG.BG_STATUS_LWR_IDX	
Index	Extra	BUG.BG_PRIORITY_LWR_IDX	
Index	Extra	BUG.BG_RESPONSIBLE_LWR_IDX	
Index	Index changed	REQ_COVER.RC_ENTITY_ID_IDX	
Index	Index changed	RUN.RN_TEST_ID_IDX	
Index	Index changed	RUN.RN_TESTCYCLE_IDX	
Function-based indexes - relevant only for SQL server.	Extra index	COMMON_SETTINGS.CS_COVER_LWR_IDX	

Type	Problem	Element	Comment
Function-based indexes - relevant only for SQL server.	Extra index	HOSTS.HOSTS_LWR_IDX	
Function-based indexes - relevant only for SQL server.	Extra index	HOSTS_IN_GROUP.HG_COVER_LWR_IDX	
Function-based indexes - relevant only for SQL server.	Extra index	HOST_GROUP. GH_LWR_IDX	
Function-based indexes - relevant only for SQL server.	Extra index	USERS.US_USERS_LWR_IDX	

Repair Process

The repair process fixes these internal differences in the following way:

- **Column Size.** Increases the size of columns to the required size.
- **Index Definition.** Removes extra indexes. It also recreates missing indexes and indexes that were defined differently.
- **Extra Function-based Indexes.** Microsoft SQL Server only. Removes obsolete function-based indexes.

Before beginning the upgrade, run the repair process on each project.

Data Validation

One of the main functions of the verification process is to ensure that the project database contains valid data.

The verification process helps you find and fix the following problems:

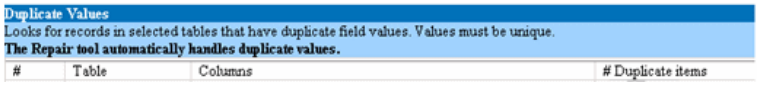
Duplicate Values 515
Duplicate IDs 515
Tree Inconsistencies 516
Views 516
Orphaned Entities 517
Missing Entities 517
Missing Lists and/or List Values 517
Encrypted Values 518

Duplicate Values

Some fields (or a combination of fields) must be unique in given tables. This constraint is enforced by the creation of a unique index on these fields. For example, the combination of fields TS_ SUBJECT and TS_NAME, which represent the ID of the test's parent folder and test name, must be unique. It is not possible to create two tests with the same name under the same folder. In rare cases, a corrupted database contains duplicate values in these fields.

Problem: The verification process checks that all unique indexes exist (and therefore enforce unique values). If the verification process finds duplicate values, it does not permit the upgrade to run on the project.

The verification report specifies the fields in which there are duplications and number of duplicate values found, as shown below.


Duplicate Values
Looks for records in selected tables that have duplicate field values. Values must be unique.
The Repair tool automatically handles duplicate values.

#	Table	Columns	# Duplicate items

Solution: Automatic Repair. Run the repair process to automatically handle the duplicate values. The repair process renames the duplicate values to resolve the problem.

Duplicate IDs

Most tables have a unique primary key, usually a unique single column. If there are duplicate values in this field, the primary key is not created.

For example, in a table called test, the column TS_TEST_ID represents the test ID, which is unique. In rare cases, a corrupted database contains duplicate IDs.

Problem: The verification process checks that all IDs in a table are unique. If it finds duplicate IDs, it does not permit the upgrade to run on the project.

The verification report specifies the fields in which there are duplicate items and values, as shown below.

Duplicate IDs
Looks for records in selected tables that have duplicate ID field values.
The Repair tool automatically deletes the duplicate records.

#	Table	Column	# Duplicate Items
1	TEST	TS_TEST_ID	2

Solution: Automatic Repair. The repair process automatically deletes one of the records with a duplicate ID.

> **Caution:**
>
> This option assumes that the entire record is duplicated, and that the duplicated record is not accessible from the ALM user interface. Because there can be exceptions, we recommend that you use this option only after verifying manually that this record deletion will not cause data loss.

Tree Inconsistencies

The verification process checks four different entity trees (hierarchical representation of entities):

- Test Plan tree
- Business Components tree
- Requirement tree
- Test Lab tree

The verification process checks that the data in the tree tables is correct.

> **Caution:** Do not manually fix any problems related to tree data. The repair process fixes them automatically.

Problem: The verification process checks for the following types of problems:

- **Corrupted Path.** This is an internal ALM field that contains a string that represents the order of each node in the tree.
- **Wrong Number of Children.** This is an internal ALM field that contains the number of children for each node in the tree.
- **Orphan Records in Trees.** By definition, orphan records do not have parent records. As a result, you cannot access them through the ALM user interface.

Solution: Automatic Repair. Run the repair process to automatically fix any problems related to tree data.

> **Caution:** Before beginning the automatic repair, review each orphan record carefully. If the verification process finds an orphan record, it deletes it (and all its descendants) from the tree automatically.

Views

Database views can contain the following warning:

- "Extra Views" below

Extra Views

ALM schemas should contain only the views that are defined in the schema configuration file.

Problem: If the verification process detects extra views that were added manually to the schema, it displays an **Extra Views** warning. Adding extra views on top of the schema is not supported and could cause problems.

> **Note:** This problem requires manual repair. The repair process cannot fix it.

Solution: Do one of the following:

- **Change the Schema.** If you use the view, copy it to a different schema. If you do not use the view, delete it. Before taking either action, back up your schema and consult your database administrator. For details, see "Changing the Database User Schema" on page 519.
- **Use the Exception File.** Not recommended: Instruct the upgrade to ignore this problem. For more information about the Exception file, see "Defining an Exception File" on page 116.

Orphaned Entities

The verification process checks for entity data that is missing corresponding parent data. For example, the following entities might be missing corresponding test configurations or test criteria:

- Test configuration coverage
- Criteria coverage
- Run criteria
- Runs
- Test instances

Caution: Do not manually fix any problems related to orphaned entities. The repair process fixes them automatically.

Problem: On version-controlled projects, deleting a test configuration or test criteria did not delete corresponding entities after checking in. This caused incorrect coverage calculation.

Solution: Automatic Repair. Run the repair process to automatically fix any problems related to orphaned entities created by this problem.

Missing Entities

The verification process checks for data that is missing. For example, the following entities might be missing:

- Test configurations
- Test criteria

Caution: Do not manually fix any problems related to missing entities. The repair process fixes them automatically.

Problem: The upgrade process can detect that certain entities are missing based on information that exists in related tables.

Solution: Automatic Repair. Run the repair process to automatically fix any problems related to missing entities created by this problem.

Missing Lists and/or List Values

The verification process checks that all of the fields of List type are associated with a list.

Problem: If a list and/or its values are missing, the verification process generates a warning about missing lists or missing list values.

Solution:

Run the repair process to create the missing list and/or its values.

Missing lists are re-created with the name: **AUTO_GENERATED_LIST_NAME_<*unique_number*>**

After running the repair process, do the following in **Customization > Project Lists**:

- Rename any lists whose names are prefixed by **AUTO_GENERATED_LIST_NAME_**.
- If necessary, add any list values that are missing.

Tip: Although you can use the repair process to add these objects, we recommend that you contact HP Support to make sure that the missing objects are not just symptoms of a bigger problem.

Encrypted Values

Some fields are saved in the database in an encrypted state. Encryption is done using confidential data passphrases.

Note: This is an issue with Performance Center and Lab Management projects.

Problem: The verification process checks that all encrypted data can be decrypted with the current confidential data passphrases. If the verification process finds encrypted values that cannot be decrypted, the project is not upgraded.

The verification report specifies the fields that cannot be decrypted.

Solution: If verifying the LAB_PROJECT fails due to a problem with the Confidential Data Passphrase, do one of the following:

- Make sure that the same Confidential Data Passphrase in defined on the original server on which the LAB_PROJECT was located, as well as on the server on which it is being restored to. For more information regarding defining the Confidential Data Passphrase, refer to the *HP Application Lifecycle Management Installation and Upgrade Guide*.
- Perform the following steps:
  a. In Site Administration: Before attempting to verify the LAB_PROJECT again, navigate to the **Lab Management** tab, and clear all encrypted field values from the project by running the following queries:
     - For a Microsoft SQL Database

```
update td.LAB_DIAGNOSTICS_SERVERS set DIAG_SVR_PASSWORD = ''
update td.LAB_AUT_HOSTS set AUTHOST_PASSWORD = ''
ALTER TABLE td.LAB_HOSTS DISABLE TRIGGER ALL
```

```
update td.LAB_HOSTS set HOST_PASSWORD = ''
ALTER TABLE td.LAB_HOSTS ENABLE TRIGGER ALL
```

- For an Oracle Database

```
update <schema name>.LAB_DIAGNOSTICS_SERVERS set DIAG_SVR_
PASSWORD = ' '
update <schema name>.LAB_AUT_HOSTS set AUTHOST_PASSWORD = ' '
update <schema name>.LAB_HOSTS set HOST_PASSWORD = ' '
```

b. Proceed with the verify, repair, and upgrade of your LAB_PROJECT.

c. Login to Lab Management and update the passwords of the AUT Hosts, Diagnostics Server and Standalone Unix Load Generators. For information on working in Lab Management, refer to the *HP ALM Lab Management Guide*.

Changing the Database User Schema

This section describes the problems that require manual repair (cannot be fixed automatically by the repair process), and recommends solutions for these problems. If you encounter any of the problems mentioned below, consult with your database administrator or contact HP Support for further guidelines to resolve these problems before upgrading.

The stability of the new database upgrade component depends on the validity of the database user schema validity. We recommend that you not change the database user schema by using the Exception file.

This section includes:

Missing Database Objects 519
Missing List Warning 520
Sequences Warning 520
Changed Database Objects 520
Extra Database Objects 521

Missing Database Objects

Missing database objects can be symptoms of a bigger problem.

Problem: Missing database objects (for example, tables and indexes) can yield unexpected and unwanted behavior.

Solution: Although you can use the repair process to add these objects, we recommend that you contact HP Support to make sure that the missing objects are not just symptoms of a bigger problem.

Missing List Warning

User-defined fields of List type must be associated with lists.

Problem: If a list is missing for a user-defined field, the verification process generates a **Missing List** warning.

Solution: Contact HP Support for instructions on changing the data type of the user-defined field from List to String in the SYSTEM_FIELD table.

Caution: Contact HP Support before attempting to fix the problem manually.

Sequences Warning

An internal mechanism manages IDs and other system numerators. The table SEQUENCES holds the name of the table or other entity whose numeration is being tracked as well as its highest current value.

Problem: If one of the records is missing in this table, or if one of the values is incorrect, the verification process generates a **Sequences Error** warning.

Solution: The repair process fixes the problem automatically.

Caution: We strongly recommend that you not attempt to fix the problem manually.

Changed Database Objects

Any of the following cases is defined as a Changed Database Object:

- Data type of a column was changed
- Length of a column was changed
- Nullability of a column was changed
- Column is defined as identity although it should not be defined as such, or vice versa

Problem: A changed column data type can result in incorrect behavior on the server side.

Solution: To avoid this behavior, make sure that you have resolved all data type and length concerns before beginning the upgrade.

For every changed database object that is found, do the following:

1. Create a new column with the required attributes as originally defined by the ALM server.
2. Move the data from the old column to the new one.

   If you cannot move the data (for example, move strings to numeric columns, or move large data to smaller fields), contact HP Support.
3. Remove the old column.
4. Rename the new column to the original column name.

Extra Database Objects

ALM has various customization options. One option is to add user-defined fields (UDFs). You can add a UDF by using either the project customization user interface or through OTA (Open Test Architecture).

Problem: Any other addition to the database user schema (for example, defining extra objects on top of ALM schema) can result in a failure, such as the following:

- **Name Conflict.** If the later version happens to include a name that you added for a proprietary database object (for example, a table, view, or column), the two names will be in conflict.
- **Copy and Synchronize Failure.** If the database user schema contains extra or missing database objects, some ALM mechanisms for copying and synchronizing might fail.
- **Extra Triggers.** If the database contains extra triggers, some update operations might fail.

Solution:

For each extra database object that is found, perform the corresponding solution:

- **Move extra columns to newly created tables.**

  To make sure a new table has a one-to-one relationship with the original table, define the primary key of the new column in the new table with the value of the primary key of the original column in the original table.

- **Move extra tables to a different database user schema.**

  These extra tables include those tables created in Step 1. You might need to amend the proprietary application data access of these tables. You can still access these tables from within the ALM database connection by specifying the full name.

  Examples:

  - Oracle

    ```
    <schema name>.<table name>
    ```

  - SQL Server

    ```
    <database name>.td.<table name>
    ```

    To be able to see these tables, you must grant the necessary permissions for the database user schema.

- **Move extra views to a different database user schema.**

  Like extra tables, these views can be moved to a different database user schema. In addition, you must grant reading permissions to the newly created database user schema on the database user schema objects.

- **Remove referential integrity between customer database objects and ALM database objects.**

  This removal includes no data loss.

- **Remove extra triggers before the upgrade, and, only if truly necessary, restore them after the upgrade.**

  No data loss is involved. The upgrade process includes data upgraders that perform some data manipulations (for example, removing duplicate values, fixing tree structures, and so on).

  Your triggers will not be invoked on these update events.

  As a result, you need to do the following:

  a. Ask HP Support for information about the data upgrader activity.

  b. Review the information about the data upgrader activity.

  c. Decide on which proprietary updates you need to perform.

- **Remove extra indexes.**

  You can log all indexes before the upgrade, and (only if truly necessary) restore them after the upgrade. No data loss is involved.

- **Oracle Database only: Move extra sequences to a newly created database user schema.**

  To access the extra sequences from the database user schema, you must grant ALM the required permissions. When moving these sequences, set them to start with the number they reached at the time of the move.

Java™
COMPATIBLE