

HP Service Test Management

Software Version: 11.50

Windows ®

User Guide

Document Release Date: June 2012

Software Release Date: June 2012



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 2001-2012 Hewlett-Packard Development Company, L.P.

Trademark Notices

Adobe™ is a trademark of Adobe Systems Incorporated.

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

UNIX® is a registered trademark of The Open Group.

This product includes an interface of the 'zlib' general purpose compression library, which is Copyright © 1995-2002 Jean-loup Gailly and Mark Adler.

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

| | |
|--|----|
| Service Test Management User Guide..... | 1 |
| Contents..... | 5 |
| Creating Application Components..... | 13 |
| Concepts..... | 14 |
| Creating Application Components Overview..... | 15 |
| Application Component Design..... | 16 |
| The Application Component Tree Hierarchy..... | 19 |
| Component Dependencies..... | 19 |
| Tasks..... | 21 |
| How to Define Components..... | 22 |
| Reference..... | 24 |
| ALM Direct Links..... | 25 |
| Application Components Module User Interface..... | 27 |
| Alerts Dialog Box..... | 28 |
| Details..... | 30 |
| Interaction Tab..... | 33 |
| Find Dialog Box..... | 37 |
| Modeling Tab..... | 39 |
| Dependency Details Dialog Box..... | 44 |
| New Folder Dialog Box..... | 46 |
| New Application Component Dialog Box..... | 47 |
| Application Components Module Icons..... | 51 |
| Application Components Module Window..... | 52 |
| Application Components Module Menus and Buttons..... | 54 |

| | |
|---|-----------|
| Troubleshooting and Limitations - Application Components..... | 59 |
| Application Components Module..... | 59 |
| Test Plan Module..... | 59 |
| Requirements Module..... | 59 |
| Web Services..... | 60 |
| Concepts..... | 61 |
| Defining Web Services Overview..... | 62 |
| Importing Services..... | 63 |
| Secure Services and Proxy Servers..... | 64 |
| Spot Testing Operations..... | 65 |
| Creating Tests in UFT or Service Test..... | 66 |
| Tasks..... | 67 |
| How to Define a Web Service..... | 68 |
| Reference..... | 70 |
| Web Service Components User Interfaces..... | 71 |
| Select Service from Systinet Dialog Box..... | 72 |
| Select Service from UDDI Dialog Box..... | 74 |
| Connection Settings Dialog Box..... | 76 |
| Import Service Dialog Box..... | 78 |
| Web Service Call - <Operation_Name> Dialog Box..... | 80 |
| Manual Runner Window..... | 83 |
| Troubleshooting and Limitations - Web Services..... | 88 |
| Managing Application Components..... | 89 |
| Concepts..... | 90 |
| Managing Application Components Overview..... | 91 |
| Tasks..... | 94 |
| How to Manage Components..... | 95 |
| Reference..... | 98 |

| | |
|---|------------|
| Update Linked Tests Dialog Box..... | 99 |
| Discover Components Wizard..... | 101 |
| Generate Components Wizard..... | 102 |
| Web Service Security..... | 103 |
| Concepts..... | 105 |
| Setting Security Overview..... | 106 |
| Security Levels..... | 107 |
| Security Scenarios Overview..... | 108 |
| Use the default Web Service scenario for:..... | 108 |
| Use a WCF scenario for:..... | 108 |
| Web Service Security Scenario..... | 109 |
| Transport Level Security..... | 109 |
| Message Level Security..... | 109 |
| WCF Scenario Settings..... | 112 |
| WCF Service (CustomBinding) Scenario Overview..... | 113 |
| WCF Service (Federation) Scenario Overview..... | 114 |
| WCF Services (WSHttpBinding) Scenario Overview..... | 115 |
| Advanced Security Settings..... | 117 |
| Tasks..... | 118 |
| How to Set Security for a Web Service on the Port Level..... | 119 |
| How to Set Security for a Specific Web Service Operation..... | 121 |
| How to Set Security for a Standard Web Service..... | 122 |
| How to Set Security for a WCF Service..... | 124 |
| How to Set up Common Web Services Security Scenarios..... | 125 |
| How to Customize Security for WCF Type Web Services..... | 129 |
| How to Test Web Services that use WS-Security or SSL..... | 133 |
| How to Set up Advanced Standards Testing..... | 135 |
| Reference..... | 137 |

| | |
|---|------------|
| Security User Interface..... | 138 |
| Security Settings for Port / Operation <Name> Dialog Box..... | 139 |
| Web Service Scenario..... | 141 |
| WCF Service (Custom Binding) Scenario..... | 147 |
| WCF Service (Federation) Scenario..... | 149 |
| WCF Service (WSHttpBinding) Scenario..... | 151 |
| Client Authentication Types:..... | 151 |
| Advanced Settings Dialog Box..... | 154 |
| Encoding Tab..... | 155 |
| Advanced Standards Tab..... | 156 |
| Security Tab..... | 157 |
| HTTP and Proxy Tab..... | 160 |
| HTTP (S) Transport..... | 160 |
| Select Certificate Dialog Box..... | 162 |
| Troubleshooting and Limitations - Web Service Security..... | 164 |
| Groups..... | 165 |
| Concepts..... | 166 |
| Groups Overview..... | 167 |
| Tasks..... | 169 |
| How to Manage Component Groups..... | 170 |
| Reference..... | 172 |
| New Groups User Interface..... | 173 |
| New Group Dialog Box..... | 174 |
| Generating Requirements and Tests..... | 176 |
| Concepts..... | 177 |
| Generating Requirements and Tests Overview..... | 178 |
| Tasks..... | 182 |
| How to Generate Requirements and Tests..... | 183 |

| | |
|--|------------|
| Reference..... | 184 |
| Generate Requirements and Tests Wizard..... | 185 |
| Troubleshooting and Limitations - Generate Requirements..... | 186 |
| Determining Test Coverage..... | 187 |
| Concepts..... | 188 |
| Coverage Overview..... | 189 |
| Rule Violations..... | 190 |
| Aspect Coverage..... | 191 |
| Requirements Coverage..... | 192 |
| Operations Coverage..... | 194 |
| Tests Coverage..... | 195 |
| Defects Coverage..... | 197 |
| Tasks..... | 198 |
| How to Check the Testing Status..... | 199 |
| Reference..... | 202 |
| Testing Status Tab..... | 203 |
| Testing Status Dashboard..... | 204 |
| Rules Tab..... | 206 |
| Aspects Tab..... | 208 |
| Requirements Tab..... | 210 |
| Requirements Tree Panel..... | 211 |
| Operations Tab..... | 213 |
| Tests Tab..... | 215 |
| Tests Tree Panel..... | 216 |
| Defects Tab..... | 218 |
| Linking Components and Requirements..... | 220 |
| Concepts..... | 221 |
| Linking Components Overview..... | 222 |

| | |
|---|------------|
| Tasks..... | 223 |
| How to Link Components to Requirements and Tests..... | 224 |
| Reference..... | 225 |
| Linking Component User Interface..... | 226 |
| Link <test-name> to Requirement Dialog Box..... | 227 |
| Application Components Tab - Requirements Module..... | 229 |
| Application Components Tab - Test Plan Module..... | 231 |
| Application Components Tree Panel..... | 233 |
| Customizing Service Test Management..... | 235 |
| Concepts..... | 236 |
| Customization Overview..... | 237 |
| Customizing the Application Components Module..... | 240 |
| Tasks..... | 243 |
| How to Customize a Project..... | 244 |
| Reference..... | 247 |
| Project Customization User Interface..... | 248 |
| General Tab..... | 249 |
| Select Requirement..... | 250 |
| Rules Tab..... | 253 |
| Aspects Tab..... | 255 |
| Component Types Tab..... | 257 |
| Project Entities Pane..... | 260 |
| Groups and Permissions..... | 263 |
| Module Access Pane..... | 266 |
| Change Analysis..... | 267 |
| Concepts..... | 268 |
| Change Analysis Overview..... | 269 |
| Change Notifications..... | 270 |

| | |
|--|------------|
| Reporting Changes Manually..... | 271 |
| Automatic Change Detection Through Updates..... | 272 |
| Contract Comparison..... | 273 |
| Tasks..... | 275 |
| How to Perform Change Analysis..... | 276 |
| Reference..... | 280 |
| Changes User Interface..... | 281 |
| Changes Tab..... | 282 |
| New Application Change Dialog Box..... | 284 |
| Application Change Details Dialog Box..... | 287 |
| Create Change Impact Test Set Wizard..... | 292 |
| Tests Tree Panel..... | 293 |
| Troubleshooting and Limitations - Changes..... | 295 |
| Reports and Graphs..... | 296 |
| Concepts..... | 297 |
| Service Test Management Reports and Graphs Overview..... | 298 |
| Summary Graphs..... | 300 |
| Progress Graphs..... | 301 |
| Trend Graphs..... | 302 |
| Summary Graphs..... | 303 |
| Progress Graphs..... | 304 |
| Trend Graphs..... | 305 |
| Tasks..... | 306 |
| How to Create and Generate Reports..... | 307 |
| How to Generate Graphs..... | 309 |

Creating Application Components

Concepts

- ["Creating Application Components Overview" on page 15](#)
- ["Application Component Design" on page 16](#)
- ["The Application Component Tree Hierarchy" on page 19](#)

Tasks

- ["How to Define Components " on page 22](#)

Reference

- ["ALM Direct Links " on page 25](#)
- ["Application Components Module User Interface" on page 27](#)
- ["Alerts Dialog Box" on page 28](#)
- ["Details " on page 30](#)
- ["Interaction Tab" on page 33](#)
- ["Find Dialog Box" on page 37](#)
- ["Modeling Tab" on page 39](#)
- ["Dependency Details Dialog Box" on page 44](#)
- ["New Folder Dialog Box" on page 46](#)
- ["New Application Component Dialog Box" on page 47](#)
- ["Application Components Module Icons " on page 51](#)
- ["Application Components Module Window" on page 52](#)
- ["Application Components Module Menus and Buttons" on page 54](#)
- ["Troubleshooting and Limitations - Application Components" on page 59](#)

Concepts

This section includes:

- "Creating Application Components Overview" on page 15
- "Application Component Design" on page 16
- "The Application Component Tree Hierarchy" on page 19

Creating Application Components Overview

You add entities to the Application Components tree by manually adding a new components. After you add a component, you can describe it through a contract.

Once you define application components, you can link them to requirements. This linkage allows you to connect the component to other Application Lifecycle Management entities, such as tests, test sets, and defects.

This section also includes:

- ["The Application Component Tree Hierarchy" on page 19](#)

Application Component Design

Stage 1 - Prepare to collaborate

The QA (Quality Assurance) and Dev (Software Development) managers prepare to collaborate in describing the components of the AUT (Application Under Test).

QA will use that information to track changes, execute tests, and report on the quality of the product.

Stage 2 - Set a methodology

Set a methodology to describe the application:

- What are the application areas in the AUT, to be translated into folders in the component hierarchy?
- What will constitute a component? For example, only high-level interfaces may be considered components, or its centrality or importance in the application?
- What component types to define? This could be based on technology, usage, and so forth.
- What fields (and possibly UDF) to associate with the component types?

Stage 3 - Quality policies

Determine quality policies to be implemented through rules and testing aspects. These policies are necessary to insure that components are tested based with the correct data.

Stage 4 - Create a project and customize it

Create an Service Test Management project or enable the extension on an existing ALM project. Open Customization and define component types and their fields, as well as UDFs for component and changes. Define the necessary rules and testing aspects.

Stage 5 - Create the application components

Open the **Application Components** module and create a folder structure to represent the application areas decided on above. Create component groups for cross-area functionality or deployment. If possible, define high-level components based on the planned strategy. If not possible, the developers can define them at a later time.

The project is ready for the QA and the software development teams may begin using it.

Stage 6 - Report changes to QA

Developers use the IDE add-on to report changes back to QA. Developers implement requirements and fix defects. They can create new components or application areas as required. These actions will be reported to Service Test Management and assigned to an existing or new application component. They can also be linked to the work item such as a requirement or defect, if one exists.

When making changes, developers describe their risk to QA, to assist them in their testing.

Developers may report a change while still in the process of enacting the change. This allows QA to prepare for testing the change. Optionally, the developers can wait until the change has been completed before reporting it to QA.

Stage 7 - QA processes changes

Before changes are gathered, QA can begin reviewing the application components and assign requirements and tests.

QA engineers can create new component groups to facilitate the management of the components.

After changes are generated, Service Test Management marks the changed components with alert icons and lists them in the Changes tab. QA uses the information in the Interaction tab to create tests for the changed components, using the change description to verify the coverage. During this process, QA can set the following fields:

- **Change status** to *In test*.
- **Assigned to developer** to determine the developer regarding an aspect of the component's functionality.
- **Created by** in the Change tab, to determine the developer who is responsible for updating the component's functionality.

QA can see the changes for one or more components:

- Select a component to view the changes specific to that component, and review its **Changes** tab,

- Select a group or folder to view an aggregated list of changes in the **Changes** tab. The Changes grid provides the context of the change which can assist QA in understanding the issue.

Stage 8 - Run tests

QA executes tests on the AUT in the testing environment. Based on the test results, they can set the status to **verified** or keep the **In test** status. if a change was tested it is considered **verified**, even if causes defects in the component.

Stage 9 - Reporting

QA uses the graphs and reporting capabilities to summarize the status of the components of the AUT and the changes. They send this report back to the development team. The developers can decide how to act upon QA's recommendations.

The on-going reporting should show a trend of more changes being verified in less time, and a downward trend in the number of defects on the components.

In conclusion, the **Application Components** module allows QA and the developers to verify the quality of the AUT in a comprehensive fashion.

The Application Component Tree Hierarchy

Service Test Management provides you with tools to define components and modify their details.

You organize your application components in a graphical hierarchy.

At the top level of the tree is the **Application Components** root folder, which contains sub folders. You can create custom folders that represent the structure of components in your application. For example, you might create separate folders to represent different types of users.

You can reorganize related entities in folders and subfolders, and drag and drop folders or subfolders to other locations in the tree.

The **Obsolete** folder contains components that were removed, but are still being used by tests stored in Application Lifecycle Management. For details about the Obsolete folder, see ["How to Manage Components "](#) on page 95.

You cannot rename, move, or delete the **Application Components** root or **Obsolete** folders.

Component Dependencies

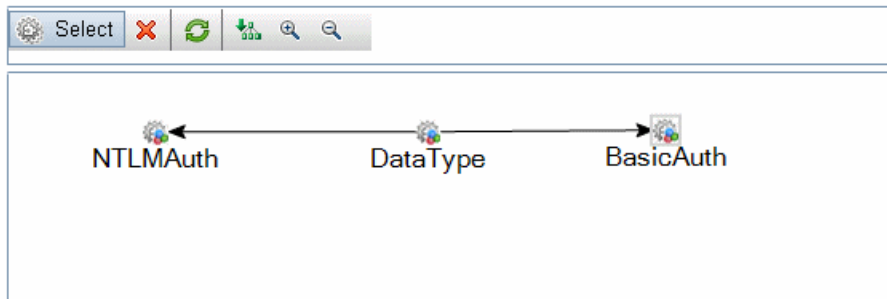
Component dependency defines a relationship between two or more components. As the operations of these linked components change, you can detect changes and track their impact on the linked components.

You can define the dependencies using Service Test Management's **Modeling** tab. The **Modeling** tab displays a diagram of the component dependencies.

An arrow indicates a dependency:

- An arrow pointing to the base component, indicates a **Dependency From** another component—the added component impacts the base component.
- An arrow pointing away from the base component indicates a **Dependency To** another component— -the added component is impacted by the base component.

You select components and add them to a modeling diagram. In the following example, the **DataType** component is dependent on both **NTLMAuth** and **BasicAuth**.



Although you add dependent components onto the canvas, the dependencies are per operation. For details, see ["How to Define Components "](#) on page 22.

Tasks

This section includes:

- ["How to Define Components "](#) on page 22


How to Define Components

This task describes how to set up your components in Service Test Managements. After you set them up, you can test them and check their functionality and compliance.

This task includes the following steps:

- Create a folder - optional
- Create a new component
- Specify a contract
- Import multiple components - optional
- Define dependencies - optional

Create a folder - optional

 Click the **New Folder** button to open the New Folder dialog box. Add new folders and subfolders to create a tree hierarchy for your components. For details, see the ["New Folder Dialog Box" on page 46](#).

You can create folders at any time and drag or copy components from folder to folder.

Create a new component

Select **Components > New > New Component** to create an new component. Select a component type and provide whatever details are available to you. For details, see the ["New Application Component Dialog Box" on page 47](#).


Specify a contract

In the New Application Component Dialog Box, click the **Interaction** link in the left pane. Click **Browse** to locate a contract file. For details, see the ["New Application Component Dialog Box" on page 47](#).

Import multiple components - optional

Select **Components > Generate Components** to open a wizard, which guides you how to import multiple contracts at once. For details, see ["Generate Components Wizard" on page 102](#).

Define dependencies - optional

1. Select the **Modeling** tab. For a description, see the "[Modeling Tab](#)" on [page 39](#).
2. Click the **Select Entities** button to open the Components list in the right pane. Expand the folders and select a component.
3. Click the arrow  to open the transfer menu.
 - a. To add a component that is dependent on the base component, click **Add Component Dependency (Dependency To)**.
 - b. To add a component upon which the base component is dependent, click **Add Component Dependency (Dependency From)**.

The Dependency Details dialog box opens. Select the operation that affects the dependency. If you do not know which operation to select, specify all of them. For details, see the "[Dependency Details Dialog Box](#)" on [page 44](#).

Reference

This section includes:

- "ALM Direct Links " on page 25
- "Application Components Module User Interface" on page 27
- "Alerts Dialog Box" on page 28
- "Details " on page 30
- "Interaction Tab" on page 33
- "Find Dialog Box" on page 37
- "Modeling Tab" on page 39
- "Dependency Details Dialog Box" on page 44
- "New Folder Dialog Box" on page 46
- "New Application Component Dialog Box" on page 47
- "Application Components Module Icons " on page 51
- "Application Components Module Window" on page 52
- "Application Components Module Menus and Buttons" on page 54

ALM Direct Links

You can use direct links to access specific views and components. You enter the direct link into your browser to access the desired view. You can save these links for future reference, or send them to others by e-mail so that they may access a specific view or component.

When working with Systinet, you can use a direct link to access the import screen, enabling you to locate services quickly.

The direct links use the following format:

```
td://<project_name>.<domain>.<server:port>/qcbn/
```

The following table describes the shortcut syntax to be added to the above string. *Italic text indicates optional parameters.*

| View | Description |
|------------------------|--|
| Systinet Import | 100?Action=ImportSystinet&SystinetID=[SYSTINET_KEY]&Environment=<ENVIRONMENT>&Uuid=<UUID> |
| Component | <p>StmModule--0000000000003130674279?EntityType=IStmService&EntityID=<component id>&View=<service tab type></p> <p><component tab type> values:</p> <ul style="list-style-type: none"> • Changes • Modeling • TestingStatus • AspectStatus • OperationStatus • StatusDashboard |
| Component Group | <p>StmModule--0000000000003130674279?EntityType=IStmGroup&EntityID=<group id>&View=<group tab type></p> <p><group tab type> values:</p> |

| View | Description |
|------|---|
| | <ul style="list-style-type: none"><li data-bbox="496 258 672 289">• Modeling<li data-bbox="496 317 743 348">• TestingStatus<li data-bbox="496 375 737 407">• AspectStatus<li data-bbox="496 434 797 466">• StatusDashboard |

Application Components Module User Interface

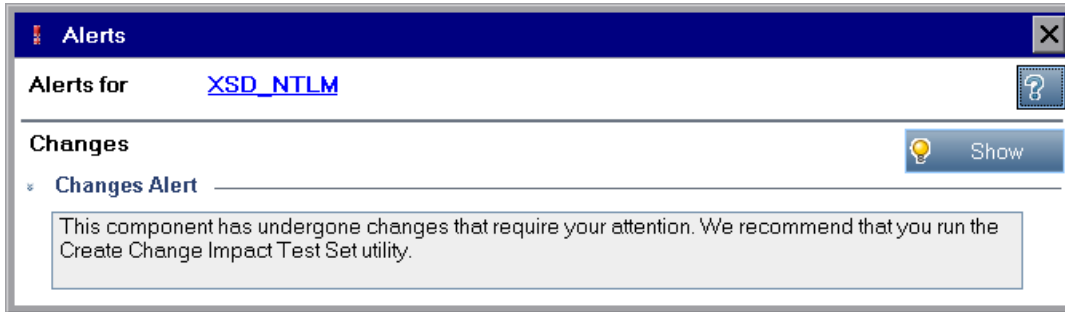
Service Test Management provides a user interface to organize your application components and customize the ALM project hosting them.

This section includes:

- "Alerts Dialog Box" on page 28
- "Details " on page 30
- "Interaction Tab" on page 33
- "Find Dialog Box" on page 37
- "Modeling Tab" on page 39
- "Dependency Details Dialog Box" on page 44
- "New Folder Dialog Box" on page 46
- "New Application Component Dialog Box" on page 47
- "Application Components Module Icons " on page 51
- "Application Components Module Window" on page 52
- "Application Components Module Menus and Buttons" on page 54

Alerts Dialog Box

This dialog box displays all alerts and rule violations associated with the selected component and enables you to resolve or disregard them.



| | |
|-----------------------|---|
| To access | Click the red exclamation point ! ,adjacent to the component name in the Application Components tree. |
| Relevant tasks | "How to Perform Change Analysis " on page 276 |
| See also | "Change Analysis " on page 267 |

The following elements are included (unlabeled UI elements are shown in angle brackets>):

| UI Elements (A-Z) | Description |
|--|--|
| | Removes the current rule violation in the Rule Violations pane: Note: When clearing a rule violation, Service Test Management disregards it— it does not delete it. |
| Alerts for <component name> | A link to the component in the Application Components tree. |
| Changes | An expandable/collapsible list of all change alerts specific to the selected component. Indicates a change to the component file since its last import or update. |

| UI Elements (A-Z) | Description |
|------------------------|---|
| Clear | Clears all rule violations in the Rule Violations pane: Note: When clearing the rule violations, Service Test disregards them — it does not delete them. |
| Resolve | For rule violations, this button opens the Create Requirements Test wizard in order to resolve the violation. For details, see the "Generate Requirements and Tests Wizard" on page 185 . |
| Rule Violations | An expandable/collapsible list of all rule violations for the selected component. A common rule violation is missing coverage for a testing aspect. |
| Show | For change alerts, this button opens the Changes tab listing all of the changes. Use the Create Test Set utility to resolve the changes. For details, see the "Create Change Impact Test Set Wizard" on page 292 . |

Details

This tab displays and enables you to edit a component's details.

| | |
|------------------------------|--|
| To access | Do the following: <ol style="list-style-type: none"> 1. Open the Application Components module, 2. Highlight a folder, component, or group 3. Click the Details tab. |
| Important information | The Details tab fields differ, depending on the entity you select: folder, component, or group, and the component type. |
| Relevant tasks | "How to Manage Components " on page 95 |
| See also | "Modeling Tab" on page 39 |

Details Tab for Components

The following elements are displayed when you select a component in the Application Components tree.

| UI Elements (A-Z) | Description |
|-----------------------------|--|
| <user-defined> | User-defined fields defined in customization. The required fields are displayed in red text. |
| Assigned to | The name of the developer to whom the component |

| UI Elements (A-Z) | Description |
|---|--|
| Dev | implementation is assigned. |
| Assigned to QA | The name of the QA engineer responsible for validating the component. |
| Authentication default credentials | Use the default user credentials: Y or N . |
| Authentication user name | The user name to use for authentication when accessing the contract file. |
| Component ID | A read-only ID assigned to the component. |
| Component name | The native component name as provided by the user. |
| Component type name | The type of component. The built-in components are General , Web Service , and JMS . If you defined a new component type, it will be displayed. |
| Created by | The name with which you logged in. You can edit this field and specify a different name. This is useful for sorting the components in reports. |
| Creation date | The date the component was created in the Component tree. |
| Development status | The development status of the component: Development , Maintenance , QA ready , or Stable . |
| Keep up to date | Keep the component up to date: Y or N . When enabled, instructs Service Test Management to update the component from its source each time you select it in the Application Component's tree. |
| Last updated | The last time the component was updated. |
| Visibility | The component's visibility level— Internal or External . |

Details Tab for Groups

The following elements are displayed when you select a group in the

Application Components tree.

| UI Elements (A-Z) | Description |
|----------------------|--|
| Assigned to | The name of the user to whom the group implementation is assigned. |
| Created by | The name with which you logged in. You can edit this field and specify a different name. This is useful for sorting the groups in reports. |
| Creation date | The date the group was created in the Application Components tree. |
| Description | A description of the group. |
| Name | The name of the group. |

Details Tab for Folders

The following elements are displayed when you select a folder in the Application Components tree:

| UI Elements (A-Z) | Description |
|--------------------|---|
| Folder ID | A read-only ID assigned to the folder. |
| Name | The folder name in the Application Components tree. |
| Description | A meaningful description of the folder. |

Interaction Tab

The **Interaction** tab displays the component's contract details and allows you to edit them. For Web services, this tab contains additional information about the WSDL and the Spot Tester to test the functionality of an operation.

The screenshot shows the 'Interaction' tab selected. It contains the following sections:


- Contract Details:**
 - Contract file: [Update From...](#)
 - Import type:
 - Service address: Override address
- Interaction Fields:**
 - WSDL nati...:
- Operations:**
 - Toolbar: Refresh, Play, Security, Ping Server
 - Table:

| Name | Description |
|-----------------------------|-------------|
| Port: mutualCert | |
| Port: mutualCertWse2Wsa0408 | |
| Port: mutualCertTripleDES | |
| Port: normal | |
| Port: mutualCertWse3 | |
| Port: wsHttpWindowsNego | |
| Port: anonymousClient | |

| | |
|------------------------------|---|
| To access | <p>Do the following:</p> <ol style="list-style-type: none"> 1. Open the Application Components module. 2. Highlight a folder, component, or group. 3. Click the Interaction tab. |
| Important information | <p>Only available when selecting a component in the tree. The information differs depending on the component type:</p> |
| Relevant tasks | <p>"How to Manage Components " on page 95</p> |
| See also | <p>"Modeling Tab" on page 39.</p> |

The Interaction tab contains the following information:

| UI Elements (A-Z) | Description |
|--|---|
| <p>Contract Details</p> | <ul style="list-style-type: none"> • Contract file. The location of the contract file defining the application component. • Browse. Opens the Open File or Import Service dialog box— visible for new components whose contracts have not yet been defined. • Update from. Opens the Open File dialog box. For Web Services, it opens the or Import Service dialog box— visible for components whose contracts have already been defined. <p>For more details, see the relevant section:</p> <ul style="list-style-type: none"> • "Import Service Dialog Box " on page 78. • "Select Service from Systinet Dialog Box" on page 72 • "Select Service from UDDI Dialog Box" on page 74 |
| <p>Contract Details - Web Services</p> | <ul style="list-style-type: none"> • Import type. Contract import method: File, URL, Systinet, or UDDI. • Toolkit. The toolkit associated with the Web service. You set this in the Customization settings, and it is read-only. • Service Address. The service deployment location to which service requests are sent. By default, it displays the endpoint address specified in the WSDL file. • Override Address. Enables you to enter an alternate endpoint for the service in the Service Address box. |
| <p>Interaction Fields - Web Service</p> | <p>WSDL Native Name. The name of the Service as it appears in the contract file.</p> <p>This section will also include any fields you designated in the Customization for the component type. For details, see "Component Types Tab" on page 257.</p> |

| UI Elements (A-Z) | Description |
|--------------------------|---|
| Interaction Fields - JMS | <ul style="list-style-type: none"> • JMS connection factory. The JNDI name of the JMS connection factory. This setting is unique per test. • JMS destination. The name of the JMS destination—either the queue or topic name. • JMS provider. The JMS service provider: Apache ActiveMQ, JBoss Messaging from JBoss, Open Message Queue, Websphere MQ from IBM, and so forth. • JNDI initial context factory. A The fully qualified class name of the factory class that will create an initial context. Provides a list of context factories and allows manual entries. • JNDI provider URL. The URL of the service provider. For example: <code>Websphere - iiop://myserver:myport</code> • Messaging Model. The type of JMS message— Point to Point, or Publish and subscribe. |
| Operations | <ul style="list-style-type: none"> •  Refresh. Reloads the list of operations. • Web Service Call. Opens the Run <operation-name> dialog box to send a test request to the server. For details, see the "Manual Runner Window" on page 83. Note: This option is only available if Service Test or the Service Test add-in is installed on the machine. • Security. Opens the Security Setting for Operation <Operation Name> dialog box, to configure the security properties for the Web service. For details, see "Web Service Security" on page 103 • Ping Server. Pings the server at the Service address URL and opens a popup indicating its status: <code>Available</code> or <code>Unavailable</code>. • <Port/Operation list>. A list of the ports for the selected application component. Expand the port to see the available operations. |

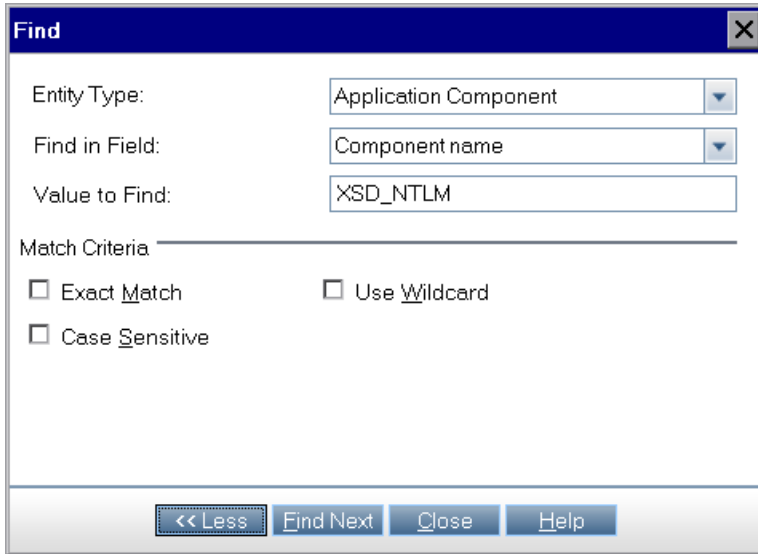
Interaction Tab for UDDI and Systinet Services

For services imported through UDDI or Systinet servers, the **Interaction** tab shows several additional read-only fields:

| UI Elements (A-Z) | Description |
|------------------------------|--|
| Systinet Server | The URL address and port of the Systinet server from which the service definition is imported. |
| UDDI Server | The URL address and version of the UDDI server from which the service definition is imported |
| UDDI/Systinet Key | A unique identifier of the service on the UDDI or Systinet server, used to locate the service definition when updating the service |

Find Dialog Box

This dialog box enables you to search for a component, group, or folder within the Application Components tree.



| | |
|-----------------------|--|
| To access | Select Edit > Find . |
| Relevant tasks | "How to Manage Components " on page 95 |
| See also | "How to Manage Components " on page 95 |

The following elements are included (unlabeled UI elements are shown in angle brackets>):

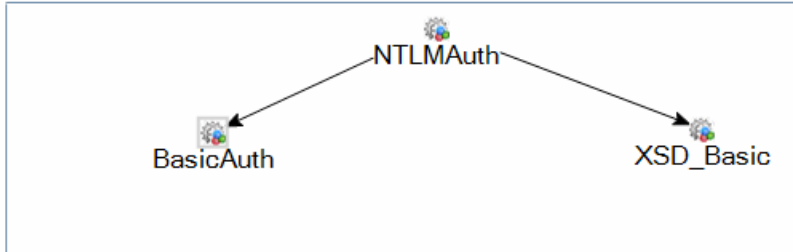
| UI Elements (A-Z) | Description |
|--------------------------|---|
| Entity Type | Type of entity in which to perform the search: Folder , ApplicationComponent , or Group . |
| Find In Field | The field within the Details tab upon which to perform the search. The available fields differ for a folder, group, or component. For information about the fields in the Details tab, see " Details " on page 30. |

| UI Elements (A-Z) | Description |
|-----------------------|---|
| Find Next | Searches for the next occurrence of the text, based on the previously used settings. |
| Less | Collapses the Find dialog box to hide the match criteria. |
| Match Criteria | Advanced match criteria:, visible when you click More . <ul style="list-style-type: none">• Exact Match. Looks for the exact string— not a partial string.• Case Sensitive. Performs a case sensitive search.• Use Wildcard. Applies wildcard rules to the search text. |
| More | Expands the Find dialog box to show the match criteria. |
| Value to Find | The text to find in the component, group, or folder. |

Modeling Tab

This tab displays component dependencies or component/group relationships in a visual representation. If you select a folder, it shows all associations and dependencies in groups and components beneath that folder.




This tab enables you to associate components with one another and assign them to component groups.



| | |
|------------------------------|--|
| To access | <p>Do the following:</p> <ol style="list-style-type: none"> 1. Open the Application Components module. 2. Highlight a folder, component, or group. 3. Click the Modeling tab. |
| Important information | To display an up to date and complete modeling view, make sure to refresh the canvas and resize it to show all of the components. |
| Relevant tasks | <ul style="list-style-type: none"> • "How to Define Components " on page 22 • "How to Manage Component Groups " on page 170 |
| See also | "Component Dependencies" on page 19 |


User interface elements are described below (unlabeled UI elements are shown in angle brackets>):






| UI Elements (A-Z) | Description |
|-------------------|---|
| | Delete. Removes the selected entity from the visual area. You cannot delete the base component or group that is selected in the left pane's Component tree. |
| | Refresh. Redraws the visual diagram in the visual area. |

| UI Elements (A-Z) | Description |
|---|---|
|  | Arrange layout. Arranges the layout of all entities in the current modeling window, in a symmetric fashion. |
|  | Zoom in. Enlarges the entities in the current drawing. |
|  | Zoom out. Reduces the size of the entities in the current drawing. |
| < modeling canvas > | A visual area showing component dependencies or component/group relationships in a graphical representation. |
| Application Component/Group Details | Details of the component or group selected in the modeling window. <ul style="list-style-type: none"> • For component details, see "Modeling Tab" on previous page. • For group details, see "Modeling Tab" on previous page. |
| Select | Opens the Application Components Tree tab in the right pane to add more component or groups to the modeling window. For information about the tree hierarchy, see the "Modeling Tab" on previous page . |

Application Components Tree

The following elements are included (unlabeled UI elements are shown in angle brackets>):

| UI Elements (A-Z) | Description |
|---|--|
|  | Add. Opens a drop down list with the following options: <ul style="list-style-type: none"> • Associate. Associates the selected component with a group. • Add Component Dependency (Dependency To) • Add Component Dependency (Dependency From) Note: The Associate option is enabled when you select a |

| UI Elements (A-Z) | Description |
|---|--|
| | <p><i>group</i> in the right pane. The Add Dependency options are enabled when you select a <i>component</i> in the right pane.</p> <p>For details about dependencies, see the "Dependency Details Dialog Box" on page 44.</p> |
|  | Refresh. Refreshes the Application Components tree. |
|  | Find. Opens the Find dialog box to search for an application component by its name. |
|  | <p>Set/Clear Filter. Opens a drop down list with the following options:</p> <ul style="list-style-type: none"> • Set Filter / Sort. Opens the Set Filter/Sort dialog box allowing you to filter and sort the components. • Clear Filter / Sort. Clears all filters and sorting settings and displays all components and groups. |
|  | Go To Application Component. Goes to an application component based on its Component ID. The ID is visible in the Details tab. |
|  | <p>Exit. Closes the Application Components Tree panel.</p> <p>To reopen this view, click the Select button in the Modeling tab.</p> |
| <Components List> | <p>An expandable list of all components by folder and group.</p> <p>Tip: If a filter is active, there could be hidden components. To see all components, click Clear Filter.</p> |

Application Component Details Pane

The following elements are displayed in the bottom pane of the **Modeling** tab, when you select a component in the modeling drawing.

| UI Elements (A-Z) | Description |
|---------------------|---|
| <user-defined> | User-defined fields defined in customization. Required fields are displayed in red. |
| Assigned to Dev | The name of the developer to whom the component implementation is assigned. |
| Assigned to QA | The name of the QA engineer responsible for validating the component. |
| Component ID | A read-only ID assigned to the component. |
| Component name | The native component name in the definition files. |
| Component type name | The type of component: General , Web Service , and JMS . If you defined a new component type in Customization, it will be displayed. |
| Created by | The name with which you logged in. You can edit this field and specify a different name. This is useful for sorting the components in reports. |
| Creation date | The date the entity was created in the Application Components tree. |
| Development status | The development status of the component: Development , Maintenance , QA ready , or Stable . |
| Has changed | Y or N . Yes indicates that the definition files (for example, WSDL), changed since the last update. |
| Keep up to date | Y or N . When enabled, updates the contract from its source each time you select the component in the Application Component's tree. |
| Last updated | The last time the WSDL was updated. |
| Update time | The time the contract was updated. |
| Visibility | The visibility of the application component: Internal or External . |




Group Details Pane

The following elements are displayed in the bottom pane of the **Modeling** tab, when you select a group in the modeling drawing.

| UI Elements (A-Z) | |
|----------------------|--|
| UI Elements | Description |
| Assigned to | The name of the user to whom the group implementation is assigned. |
| Created by | The name with which you logged in. You can edit this field and specify a different name. This is useful for sorting the groups in reports. |
| Creation date | The date the group was created in the components tree. |
| Name | The name of the group. |

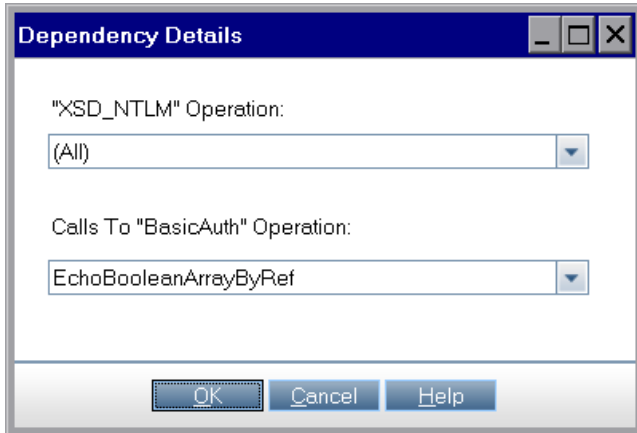
Component Dependency Details Pane



The following elements are displayed in the bottom pane of the **Modeling** tab, when you select a dependency arrow between two components, in the modeling canvas.

| UI Elements (A-Z) | |
|---|--|
| UI Elements | Description |
|  | Edit. Opens the Dependency Details dialog box for editing the entry selected in the Dependency list. |
|  | Delete. Removes the selected dependency. |
|  | Refresh. Reloads the Dependency list. |
| < Dependency list > | A list of the dependencies between the components connected by the selected arrow. |
| New | Opens the Dependency Details dialog box for defining a new dependency between the components. For details, see "Dependency Details Dialog Box" on page 44. |

Dependency Details Dialog Box

This dialog box enables you to set the dependencies between components or modify existing ones.



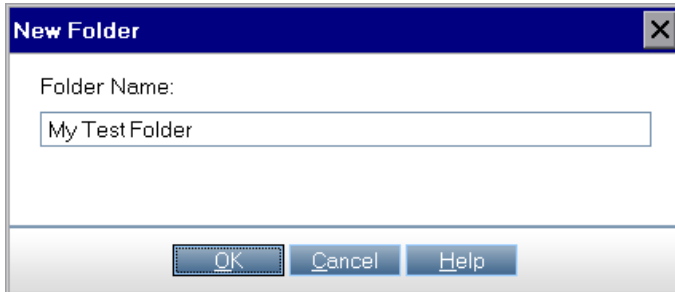
| | |
|-------------------------------------|---|
| <p>To access</p> | <p>In the Modeling tab:</p> <ul style="list-style-type: none"> • For a new dependency: Click Select to show the list of application components in the right pane. Select a component and click the Add button  to create a new dependency. The dialog box opens automatically. • For an existing dependency: Click in the lower Component Dependency Details pane, select a dependency, and click the Edit button . |
| <p>Important information</p> | <p>An arrow indicates a dependency:</p> <ul style="list-style-type: none"> • An arrow pointing to the base component indicates a Dependency From another component—the added component impacts the base component. • An arrow pointing away from the base component indicates a Dependency To another component— -the added component is impacted by the base component. |
| <p>Relevant tasks</p> | <p>"How to Define Components " on page 22</p> |


The operation selected in the upper field, must call the operation selected in the lower, **Calls to**, field. The user elements are described below:

| UI Elements (A-Z) | Description |
|--|--|
| <component_name> Operation | <p>A specific operation of the dependent component.</p> <ul style="list-style-type: none">• For Dependency To, this is the component selected in the main Application Components tree.• For Dependency From, this is the component selected in the components list in the right pane. <p>Tip: Choose All if you are unsure.</p> |
| Calls to <component_name> Operation | <p>A specific operation of the called component.</p> <ul style="list-style-type: none">• For Dependency To, this is the component selected in the components list in the right pane.• For Dependency From, this is the component selected in the main Application Components tree. <p>Tip: Choose Any if you are unsure.</p> |

New Folder Dialog Box

This dialog box page enables you to create a new folder in the Application Components tree.



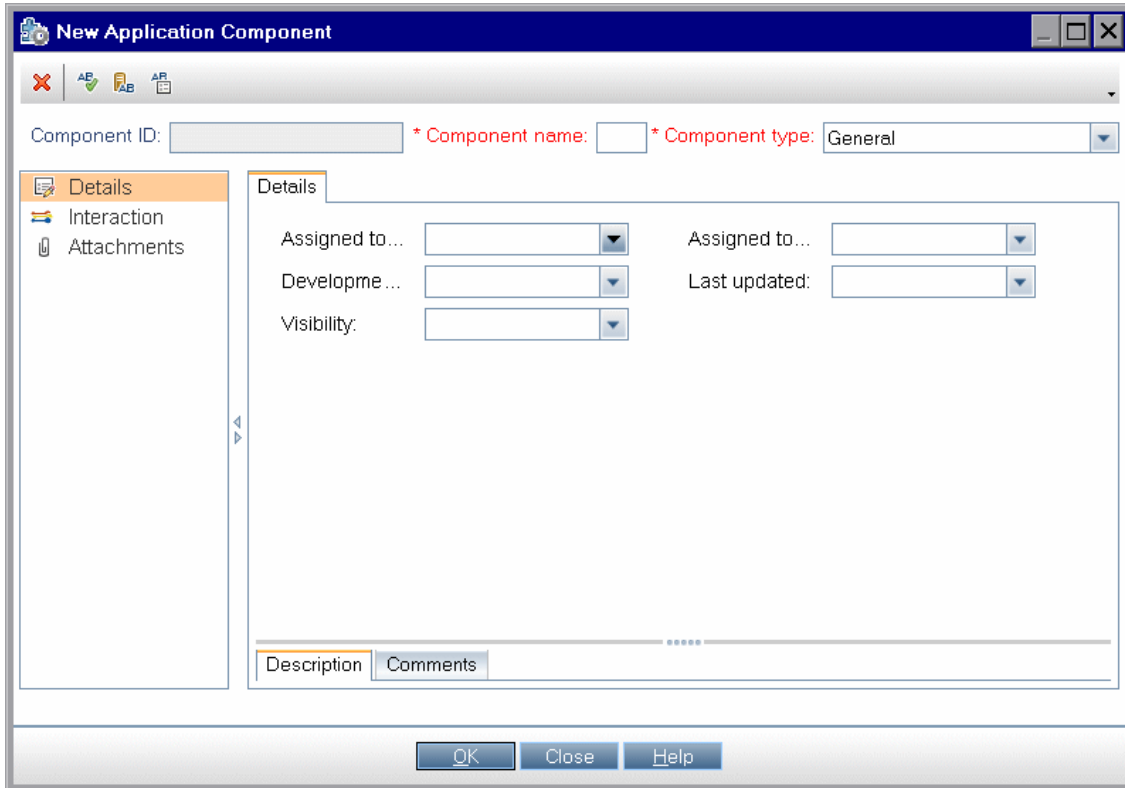
| | |
|------------------------------|---|
| To access | Do one of the following: <ul style="list-style-type: none"> • Click the New Folder  button. • Select Components > New > New Folder. |
| Important information | A folder name cannot include any of the following characters: \ / * |
| Relevant tasks | "How to Define Components " on page 22 |

User interface elements are described below:

| UI Elements (A-Z) | Description |
|--------------------|--|
| Folder Name | A name for the new folder, beneath the folder selected in the Application Components tree. You can create several levels of sub-folders. Tip: Add a description to the folder in the Details tab. |
| OK | Adds the new folder to the Application Components tree. |


New Application Component Dialog Box






This dialog box enables you to create a new component.





| | |
|-----------------------|---|
| To access | Use one of the following: <ul style="list-style-type: none"> • Select Components > New > New Component. • Select New > New Component from the shortcut menu. |
| Relevant tasks | "How to Define Components " on page 22 |

The following elements are included:

| UI Elements (A-Z) | Description |
|---|---|
|  | Clear All. Clears all the fields so that you can redefine the component. |

| UI Elements (A-Z) | Description |
|---|--|
|  | <p>Spell Check. Begins a spell check in the Component name field.</p> |
|  | <p>Thesaurus. Performs a thesaurus lookup for the term highlighted in the Component name field.</p> |
|  | <p>Spelling Options. Opens the Spelling Options dialog box for customizing the spell check in the following areas</p> <ul style="list-style-type: none"> • Ignore words in UPPERCASE • Ignore words containing numbers (enabled by default) • Ignore markup languages (HTML, XML, etc.) • Ignore Internet addresses (enabled by default) • Prompt on repeated word (enabled by default) <p>You can also select the main dictionary and a custom one.</p> |
|  | <p>Details pane. The component's properties. The displayed fields differ based on the component type and the fields enabled in the customization section. For details, see the "Component Types Tab" on page 257. Some common fields are:</p> <ul style="list-style-type: none"> • Assigned to Dev • Assigned to QA • Created by • Development Status: Development, Maintenance, QA ready, or Stable. • Last Updated • Visibly: External or Internal <ul style="list-style-type: none"> ▪ Description tab. An editable area for a description of the component. ▪ Comments tab. An editable area for adding remarks. |
|  | <p>Interaction pane. Contract information.</p> |

| UI Elements (A-Z) | Description |
|--|---|
| | <p>The displayed fields differ based on the component type and the fields enabled in the customization section. To display a field in this section, select it in the customization's Interaction Field column. For details, see the "Component Types Tab" on page 257.</p> <ul style="list-style-type: none"> • Contract File. The location of the contract. • Browse button. <ul style="list-style-type: none"> ▪ For non-Web Service components, opens the Open File dialog box. ▪ For Web Services, opens the Import Service dialog box. For details, see "Import Service Dialog Box" on page 78. • Operations. A list of the component's operations. • Parameters. The component's parameters sorted by the following columns: In/Out, Parameter Name, and Type (data type). |
|  for JMS types | <ul style="list-style-type: none"> • JMS connection factory. The JNDI name of the JMS connection factory. This setting is unique per test. • JMS Destination. The name of the JMS destination—either the queue or topic name. • JMS Provider. The JMS service provider: Apache ActiveMQ, JBoss Messaging from JBoss, Open Message Queue, Websphere MQ from IBM, and so forth. • JJNDI initial context factory. The fully qualified class name of the factory class that will create an initial context. Provides a list of context factories and allows manual entries. • JNDI provider URL. The URL of the service provider. For example: <pre>Websphere - iiop://myserver:myport</pre> • Messaging Model. The type of JMS message— Point to point, or Publish and subscribe. |








| UI Elements (A-Z) | Description |
|---|--|
|  | Attachments pane. Allows you to attach a file to the component. For details, see the <i>HP Application Lifecycle Management User Guide</i> . |
| Component ID | A read-only value assigned to each application component. |
| Component name | A name for the component. Use the Spell Check or Thesaurus buttons for the textual strings. |
| Component type | The type of component to create: General , Web Service , or JMS . If you created a custom type component in Customization, it also appears in the drop down list. |

Application Components Module Icons

This section describes the icons available in the **Application Components** module.

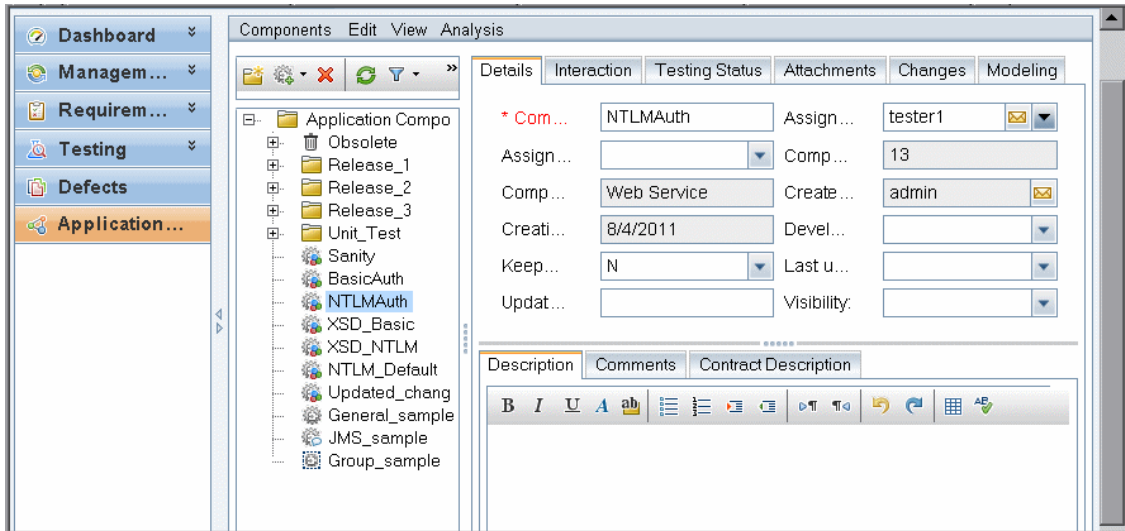
| | |
|------------------|---|
| To access | Click the Application Components button on the ALM sidebar. |
| See also | <ul style="list-style-type: none"> • "Application Components Module Window" on page 52 • "Application Components Module Menus and Buttons" on page 54 |

The icons are described below:

| UI Elements | Description |
|---|---|
|  | <p>Alert. Click to view a summary of the following alert types:</p> <ul style="list-style-type: none"> • Changes. The component definition has changed. • Rule Violations. The component violates a rule. <p>For more details, see the "Alerts Dialog Box" on page 28.</p> |
|  | Folder. A folder for organizing components. |
|  | Obsolete folder. Contains components that were deleted that may affect other components. |
|  | Web Service type. A Web Service type application component. |
|  | General type. A General type application component. |
|  | JMS type. A JMS type application component. |
|  | Application Component group. |

Application Components Module Window


This section describes the **Application Components** module window.



| | |
|-----------------------|---|
| To access | Click the Application Components button on the ALM sidebar. |
| Relevant tasks | "How to Manage Components " on page 95 |
| See also | <ul style="list-style-type: none"> • "Application Components Module Icons " on page 51 • "Application Components Module Menus and Buttons" on page 54 |

User interface elements are described below:

| UI Elements (A-Z) | Description |
|--------------------|--|
| <toolbar and menu> | Provides menus and toolbar buttons specific for the Application Components module. For details, see the "Application Components Module Menus and Buttons" on page 54. |
| <Components Tree> | Shows a tree hierarchy of all the components, component groups, and folders. For details, see "Application |







| UI Elements (A-Z) | Description |
|----------------------------------|---|
| | <p>Components Module Icons " on page 51.</p> |
| <p>Attachments tab</p> | <p>Lists the files that have been associated with the selected folder, component, or group as attachments. The tab includes an icon  if the selected component has attachments. For details, see the <i>HP HP Application Lifecycle Management User Guide</i>.</p> |
| <p>Details tab</p> | <p>Displays general details and attributes of the selected component, group, or folder. Displays data that is required to identify the component, including the creation information. For details, see the "Details " on page 30.</p> |
| <p>Changes tab</p> | <p>Lists the changes in the components. It also provides an interface to define new changes that you made to the components and create test sets to check their impact. For details, see "Automatic Change Detection Through Updates" on page 272.</p> |
| <p>Interaction tab</p> | <p>Provides the component's contract details, such as the WSDL location for Web Services. For details, see the "Interaction Tab" on page 33.</p> |
| <p>Modeling tab</p> | <p>Provides an interface to graphically associate components with groups, and to define dependencies between them. For details, see "Modeling Tab" on page 39.</p> |
| <p>Testing Status tab</p> | <p>Provides a coverage summary for aspects, requirements, operations, tests, rule violations, and defects. For details, see "Determining Test Coverage" on page 187.</p> |


Application Components Module Menus and Buttons

This section describes the menus and buttons available in the **Application Components** module.



| | |
|------------------------------|--|
| To access | Open the Application Components module on the Application Lifecycle Management sidebar. |
| Important information | All menu items are available from the right-click shortcut menu. |
| Relevant tasks | "How to Manage Components " on page 95 |

User interface elements are described below.

| UI Elements (A-Z) | Menu | Description |
|---|---------|--|
|  | Toolbar | New Folder. Opens the New Folder dialog box to create a folder under the root or an existing folder. |
|  | Toolbar | New Component. Opens the New Application Component dialog box. Note: This area on the toolbar displays the last button used: New Component or New Group . |
|  | Toolbar | New Group. Opens the New Group dialog box. |
|  | Toolbar | Delete. Deletes selected components. Deleting a component also deletes its children. You cannot delete the root folder. |
|  | Toolbar | Refresh All. Refreshes the Application Components tree so that it displays the most up-to-date versions. |
|  | Toolbar | Filter/Sort. Enables you to filter and |

| UI Elements (A-Z) | Menu | Description |
|---|------------|--|
| | | sort the requirements in the requirements tree or grid. For more details, see below. |
|  | Toolbar | Send By E-mail. Opens the Send E-mail dialog box, enabling you to send component URLs and other information to recipients from a list or to the creator of the component. Use this in conjunction with Copy URL . For more details, see the <i>HP HP Application Lifecycle Management User Guide</i> . |
| Copy URL/Paste | Edit | Copies a selected component and pastes its URL as a link. The component itself is not copied. Instead, you can paste the address into another location, such as an e-mail or a document. Clicking the link opens ALM and takes you to the requirement. If you are not already logged in, ALM first prompts for login details. |
| Copy/Paste | Edit | Copies selected components within the same project or across projects. When you copy a folder, its sub-folders are copied too. Notes: <ul style="list-style-type: none"> • Coverage and modeling data for the component are not copied. • If you paste a component that has the same name as an existing component, the suffix <code>_Copy_x</code> is added automatically to the end of the name. |
| Create Empty Test | Components | Opens the Create Empty Test dialog box, prompting you for a location and name. This is useful for testers who |

| UI Elements (A-Z) | Menu | Description |
|-----------------------------------|------------|---|
| | | know the required steps, but are unfamiliar with the technical aspects of creating tests. |
| Cut/Paste | Edit | Moves selected component to a different location in the Application Components tree. Tip: You can also move a component to a new location by dragging it. |
| Delete ✖ | Edit | Deletes the selected component. |
| Download Contract File | Components | Allows you to download a contract file from ALM to a local file system. |
| Expand All/Collapse | View | Expands all or collapses the selected branches in the Component tree. |
| Filter/Sort ⚙ | View | Enables you to filter and sort application components in the tree. For more details, click the Help button. |
| Find | Edit | Searches for a component in the Application Components module. For details, see " Find Dialog Box " on page 37. |
| Generate Requirements/Test | Components | Opens the Generate Requirements and Test wizard. For details, see " Generating Requirements and Tests " on page 176. |
| Graphs | Analysis | Generates one of the following graphs: <ul style="list-style-type: none"> • Components Summary - Group by 'Type' • Changes Progress - Group by 'Risk' • Changes Summary - Group by 'Risk' |

| UI Elements (A-Z) | Menu | Description |
|--|------------|--|
| | | <ul style="list-style-type: none"> • Changes Trend - Group by 'Status' |
| New | Components | <p>Enables you to create a new entity:</p> <ul style="list-style-type: none"> • New Folder. Enables you to create a new folder in the Application Components tree. • New Group. Opens the New Group dialog box. • New Component. Opens the New Component dialog box. |
| Refresh All  | View | Refreshes the Component tree so that it displays the most up-to-date versions. |
| Rename | Edit | <p>Renames the selected entity. You cannot rename the root folder.</p> <p>Syntax exceptions:</p> <ul style="list-style-type: none"> • Folder or group names cannot include: *^. • Component names cannot include: \, /, :, ", ?, ', <, >, , *, %, !, {, or }. |
| Restore Component | Components | Restores a deleted component from the Obsolete folder. |
| Send by  E-mail | Components | Opens the Send E-mail dialog box, enabling you to send URLs and other information to recipients from a list or to the creator of the component. Use this in conjunction with Copy URL . For more details, see the <i>HP Application Lifecycle Management User Guide</i> . |
| Update Component | Components | Enables you to update a component from : |

| UI Elements (A-Z) | Menu | Description |
|-------------------|------|--|
| | | <ul style="list-style-type: none"><li data-bbox="818 260 1395 331">• Update Component. Updates the component from its original location.<li data-bbox="818 361 1395 512">• Update Component from. Opens a dialog box for navigating to a URL or a file with an updated version of the contract. |

Troubleshooting and Limitations - Application Components

This section describes the general limitations for creating application components with Service Test Management.

Application Components Module

- When using the Find dialog box to locate an entity in the Application Components tree, certain fields are read-only and you cannot provide a value, such as **Component ID**.
- If you restore a service from the **Obsolete** folder, it may not appear in the application components tree until you perform a new login to the project.
- Certain graphs generated by the Application Lifecycle Management Dashboard, cannot be filtered by the **Component Type Name** field.
- After upgrading from a version 9.41 project, the first time you want to update an application component, you must use the **Update Component from** option—not the **Update Component** option.

Test Plan Module

- When working in the **Test Plan** module's **Application Components** tab—if you select a linkage to a component and press the keyboard's **Delete** button, Service Test Management tries to delete the entire test, instead of just the link.

Requirements Module

- When defining a link between a requirement and application component through the **Requirements** module's **Application Components** tab: if the requirement is of a type that does not support coverage (such as a folder), the link is not created and Service Test Management does not issue a warning.

Web Services

Concepts

- "Defining Web Services Overview" on page 62
- "Importing Services " on page 63
- "Secure Services and Proxy Servers " on page 64
- "Spot Testing Operations " on page 65
- "Creating Tests in UFT or Service Test " on page 66

Tasks

- "How to Define a Web Service " on page 68

Reference

- "Web Service Components User Interfaces" on page 71
 - "Select Service from Systinet Dialog Box" on page 72
 - "Select Service from UDDI Dialog Box" on page 74
 - "Connection Settings Dialog Box" on page 76
 - "Import Service Dialog Box " on page 78
 - "Web Service Call - <Operation_Name> Dialog Box " on page 80
 - "Manual Runner Window" on page 83
- "Troubleshooting and Limitations - Web Services" on page 88

Concepts

This section includes:

- "Defining Web Services Overview" on page 62
- "Importing Services " on page 63
- "Secure Services and Proxy Servers " on page 64
- "Spot Testing Operations " on page 65
- "Creating Tests in UFT or Service Test " on page 66

Defining Web Services Overview

Web services are a specific case of application components, where the contract is a WSDL file, defining the Web service, its address, and the operations. Service Test Management's functionality for application components contains several additional features specific for Web Services, such as functionality testing, change detection, and specialized reports.

When working with Web Services, you create a component hierarchy as described in "[Creating Application Components Overview](#)" on page 15.

Web Service contracts are typically WSDL files. Service Test Management allows you to import WSDL files that define the Web service and its operations.

You can import a definition from a file location, URL, UDDI or Systinet server. When you import a definition, Service Test Management extracts the service name, service address (endpoint), and data from the definition file, and stores it in the Application Lifecycle Management repository.

For defining application components that are not WSDL-based, see the "[New Application Component Dialog Box](#)" on page 47.

This section also includes:

- "[Importing Services](#)" on page 63
- "[Secure Services and Proxy Servers](#)" on page 64
- "[Spot Testing Operations](#)" on page 65

Importing Services

You can import Web service definitions based on WSDL files from a file location, URL, UDDI or Systinet server. When importing a WSDL file, Service Test Management parses the XML code and stores the service in the Application Lifecycle Management repository.

You can then link the service to other Application Lifecycle Management entities such as tests, test sets, and defects. These links help you to keep track of the service and ensure compliance with your requirements throughout the testing process. For details about requirement coverage, see "[Generating Requirements and Tests](#)" on page 176.

You can specify connection settings for WSDL files that reside in secure locations or accessible through proxy servers. These settings let you specify authentication and proxy server details. Once you enter the security or proxy information, it remains permanently associated with the service, enabling automatic updates. For details, see "[Secure Services and Proxy Servers](#)" on page 64.

For user interface details, see the "[Import Service Dialog Box](#)" on page 78.

Secure Services and Proxy Servers

When importing WSDL files from a URL or UDDI server, the WSDL may require authentication if it resides in a secure location. Certain WSDLs may only be accessed through a proxy server.

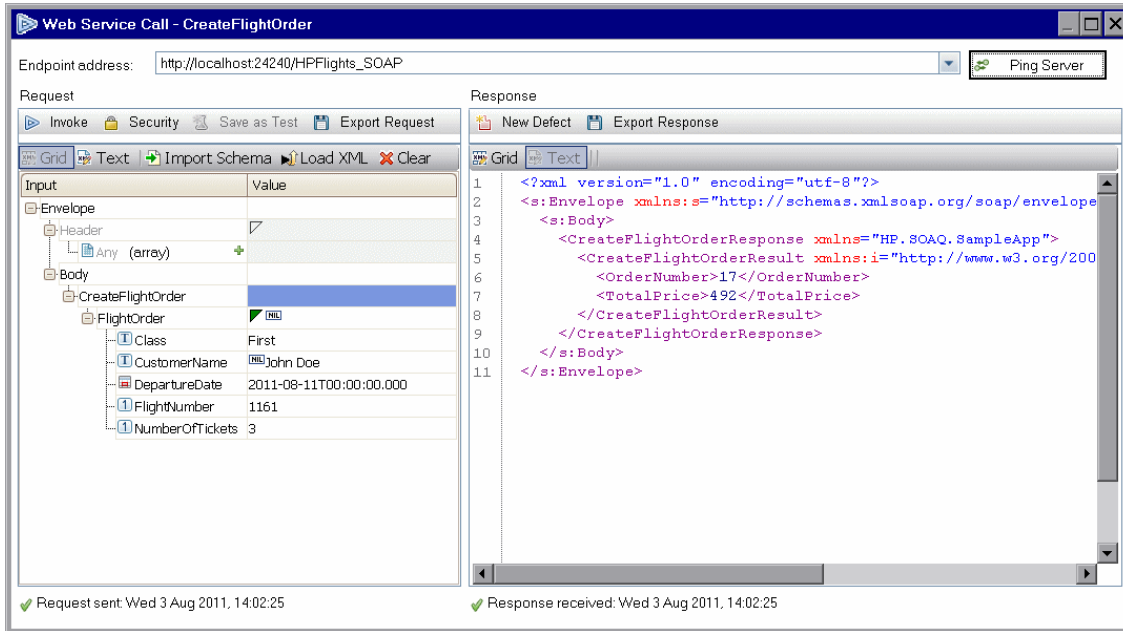
Service Test Management supports WSDLs that require authentication and WSDLs accessed through proxy servers, using basic and NTLM authentication.

The security or proxy information is always associated with the WSDL. If you enable the **Keep up to date** option to allow automatic synchronization, Service Test Management uses the authentication or proxy server settings to access the WSDL at its source.

Note: Setting default connection settings in the project customization area, instructs Service Test Management to use the same settings for all your services. For details, see ["How to Customize a Project" on page 244](#).

Spot Testing Operations

The Spot Tester lets you test your operations one by one to verify their functionality. To test the functionality, you invoke the service call and check its response.



You use the grid to enter request values and submit them to the server. You can view the SOAP response and export it to a file.

The Spot Tester also lets you save the step with its parameter values as a Service Test test, provided that Service Test is installed.

For task details, see "How to Define a Web Service " on page 68.

For user interface details, see the "Web Service Call - <Operation_Name> Dialog Box " on page 80.

Creating Tests in UFT or Service Test

If you have HP Unified Functional Testing or HP Service Test, you can create tests to run on the Web services stored in Service Test Management.

In UFT or Service Test, select **ALM > ALM Connection** to connect to the ALM project that contains your Web Service type components.

Create a test in the normal way, and save it to the appropriate location in the ALM repository.

Use the **ApplicationComponents** tab in the **Test Plan** module to link the Web service component to the test. For details, see "[Application Components Tab - Test Plan Module](#)" on page 231.

For additional details, see the HP Application Lifecycle Management User Guide.

Tasks

This section includes:

- ["How to Define a Web Service "](#) on page 68

How to Define a Web Service

This task describes how to set up your components in Service Test Management. After you set them up, you can test them and check their functionality and compliance.

This task includes the following steps:

- Create a folder
- Create a new component
- Import a service
- Set authentication or proxy information - optional
- Create dependencies - optional
- Test the service - optional
- Run the test

1. Create a folder

Open the New Folder dialog box and add new folders and subfolders based on your design requirements. The development and quality assurance teams should design a tree hierarchy representing the structure of the application. For details, see the ["New Folder Dialog Box" on page 46](#).

You can create folders at any time and drag or copy components from folder to folder.

2. Create a new component

Create a new component. Select **Components > New > New Component**. For details, see the ["New Application Component Dialog Box" on page 47](#).

3. Import a service

In the New Application Component Dialog box, open the **Interaction** view and click **Browse** to open the Import Service dialog box. Import a Web service. For details, see the ["Import Service Dialog Box " on page 78](#).


4. Set authentication or proxy information - optional

If the WSDL file requires authentication or if it accessible through a proxy

server, click the **Connection Settings** button to configure the connection settings. For details, see the "[Connection Settings Dialog Box](#)" on page 76.

5. Create dependencies - optional

Dependencies indicate the reliance of one component upon another.

- a. Select the **Modeling** tab. For a description, see the "[Modeling Tab](#)" on page 39.
- b. Click the **Select Entities** button to open the Components list in the right pane. Expand the folders and select a component.
- c. Click the arrow  to open the transfer menu.
 - o To add a component that is dependent on the base component, click **Add Component Dependency (Dependency To)**.
 - o To add a component upon which the base component is dependent, click **Add Component Dependency (Dependency From)**.
The Dependency Details dialog box opens.
- d. Select the operation that affects the dependency. If you do not know which operation to select, specify all of them. For details, see the "[Dependency Details Dialog Box](#)" on page 44.

6. Test the service - optional

To generate a Web Service call in order to test your Web Service, click the **Interaction** tab, select an operation, and click **Web Service Call**. In the grid, enter request values. Send the request to the server and check the SOAP response. For details, see the "[Web Service Call - <Operation_Name> Dialog Box](#)" on page 80.

7. Run the test

After you verify that your service is valid, run the test or test set as you would run any test in ALM, from the **Test Lab** module. For details, see the *HP Application Lifecycle Management User Guide*.

The manual running option provides an interface for working with Web Services. For details, see the "[Manual Runner Window](#)" on page 83.

Reference

This section includes:

- "Web Service Components User Interfaces" on page 71
- "Select Service from Systinet Dialog Box" on page 72
- "Select Service from UDDI Dialog Box" on page 74
- "Connection Settings Dialog Box" on page 76
- "Import Service Dialog Box " on page 78
- "Web Service Call - <Operation_Name> Dialog Box " on page 80
- "Manual Runner Window" on page 83

Web Service Components User Interfaces

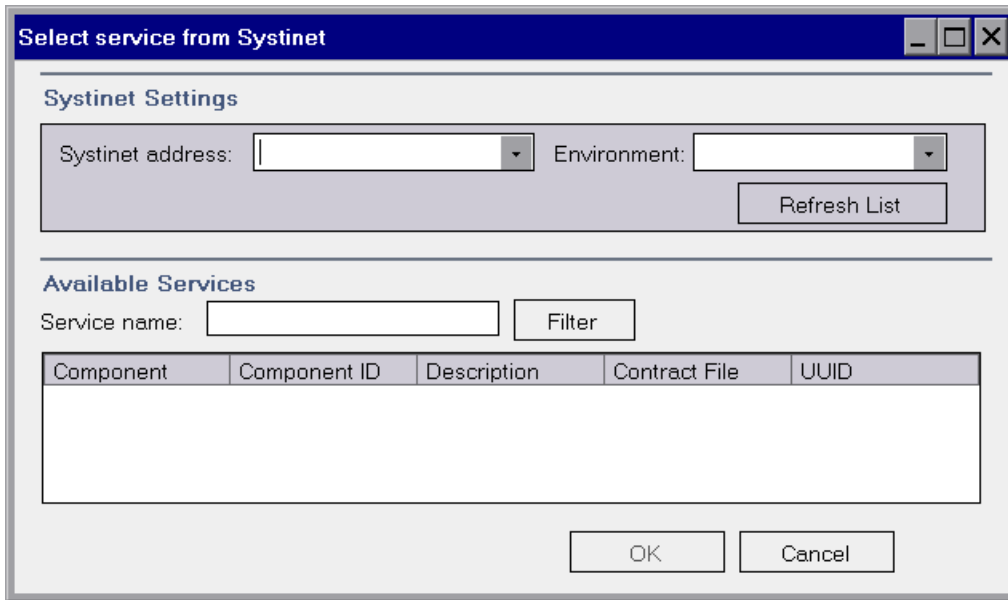
Service Test Management provides a user interface to import Web Service contracts and test the Web Service Calls.


This section includes:

- "Select Service from Systinet Dialog Box" on page 72
- "Select Service from UDDI Dialog Box" on page 74
- "Connection Settings Dialog Box" on page 76
- "Import Service Dialog Box " on page 78
- "Web Service Call - <Operation_Name> Dialog Box " on page 80
- "Manual Runner Window" on page 83

Select Service from Systinet Dialog Box

This dialog box enables you to select and import services from a Systinet server.



| | |
|------------------------------|--|
| To access | In the Import Service dialog box, choose Import WSDL from: Systinet and click the Browse  button. |
| Important information | Note: Only ALM Governance 4.0 (Systinet 4.0) is supported for Service Test Management 11.00. |
| Relevant tasks | "How to Define a Web Service " on page 68 |
| See also | <ul style="list-style-type: none"> "Import Service Dialog Box " on page 78 "Select Service from UDDI Dialog Box" on page 74 |

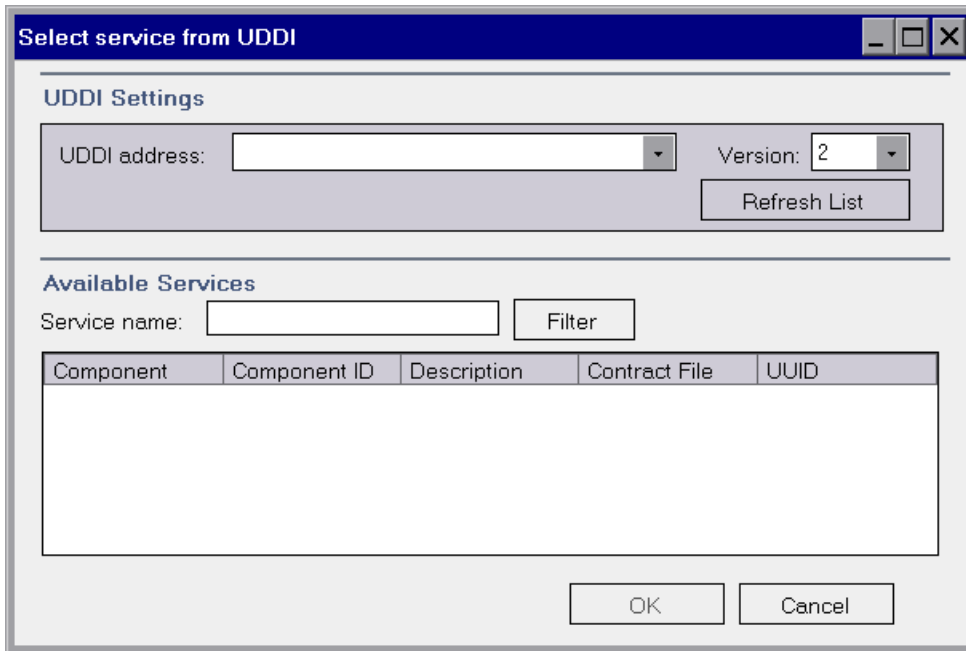
The following elements are included:


| UI Elements (A-Z) | Description |
|---------------------------|---|
| Available Services | <p>A list of the services that match the filter criteria in alphabetical order. The grid shows the following information:</p> <ul style="list-style-type: none"> Component Name |

| UI Elements (A-Z) | Description |
|-------------------------|---|
| | <ul style="list-style-type: none">• Component ID• Description• Contract File• UUID (Universally Unique Identifier) |
| Environment | The Systinet environment and version. |
| Filter | Filters the list of services to show only those services containing the string specified in the Service name field. |
| Refresh List | Reloads the list of the services on the Systinet server, in the Available Services pane. |
| Service name | A string in the service name by which to filter the list. |
| Systinet Address | The name or IP address of the Systinet server. For Example: <code>http://pumpkin:8080/soa</code> |

Select Service from UDDI Dialog Box

This dialog box enables you to select and import services from a UDDI (Universal Description, Discovery, and Integration) server.



| | |
|-----------------------|---|
| To access | In the Import Service dialog box, choose Import WSDL from : UDDI and click the Browse  button. |
| Relevant tasks | "How to Define a Web Service " on page 68 |
| See also | "Select Service from Systinet Dialog Box" on page 72 |

The following elements are included:

| UI Elements (A-Z) | Description |
|---------------------------|--|
| Available Services | <p>A list of the services that match the filter criteria. The grid shows the following information:</p> <ul style="list-style-type: none"> • Component Name • Component ID • Description |

| UI Elements (A-Z) | Description |
|----------------------|---|
| | <ul style="list-style-type: none">• Contract File• UUID (Universally Unique Identifier) |
| Environment | The UDDI version— 2 or 3. |
| Filter | Filters the list of services to show only those services containing the string specified in the Service name field. |
| Refresh List | Reloads the list of the services from the UDDI, in the Available Services pane. |
| Service name | A string in the service name by which to filter the list. |
| UDDI address | The name or IP address of the UDDI server inquiry API. For Example: http://lab1.devlab.ad:8090/juddi/inquiry . |

Connection Settings Dialog Box

This dialog box enables you to provide authentication credentials and proxy server details for the machine hosting a Web Service's WSDL file.

| | |
|-------------------------------------|---|
| <p>To access</p> | <ul style="list-style-type: none"> • For a new service: In the New Component window (Components > New > New Component), select the Interaction node and click the Browse button. In the Import Service dialog box, click the Connection Settings button. • For existing services: Select the Interactions tab, click Update from, and click Connection Settings. • In Customization: Select Service Test Management link > General Tab > Default Connection Settings. |
| <p>Important information</p> | <p>Available only for services imported through a URL and UDDI.</p> |
| <p>Relevant tasks</p> | <p>"How to Define a Web Service " on page 68</p> |

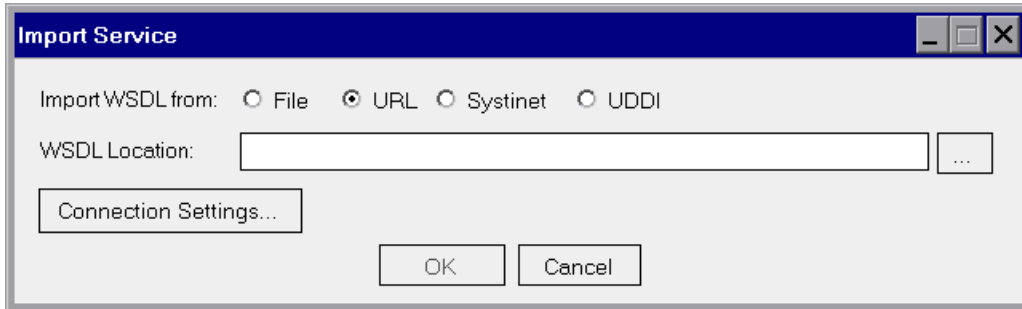
| | |
|-----------------|---|
| See also | You can set the default WSDL connection settings in the Project Customization window. For details on customizing the connection settings, see . |
|-----------------|---|

The following elements are included:

| UI Elements (A-Z) | Description |
|-------------------------|---|
| Authentication | <p>Access the WSDL located in a secure location, using the following authentication credentials. Enables you to enter credentials for authentication:</p> <ul style="list-style-type: none"> • Use default credentials. Authenticates with the Windows logon credentials of the user logged on to the machine. • User name, Password. The user name and password to be used for authentication. For users not in the default domain, type the domain name before the user name. For example, <code>domain1/qc_user1</code>. |
| Proxy | <p>Access the WSDL using the following proxy server and credentials. Enables you to enter proxy details and credentials:</p> <ul style="list-style-type: none"> • Proxy server. Name or IP address of proxy server. • Port. Port through which to access the WSDL. • Use default credentials. Connects to the proxy server with the Windows logon credentials of the user logged on to the machine. • User name, Password. The user name and password to be used for authentication. For users not in the default domain, type the domain name before the user name. For example, <code>domain1/qc_user1</code>. |
| Restore Defaults | Restores the default connection settings—user name authentication and no proxy server. |

Import Service Dialog Box

For Web Service type application components, this dialog box enables you to import WSDLs from a file system, a URL, a UDDI, or Systinet.



| | |
|-----------------------|---|
| To access | <ul style="list-style-type: none"> For new Web services: In the New Application Component window (Components > New > New Component), select the Interaction link. Click Browse. For existing Web services: Select the Interaction tab and click Update From. |
| Relevant tasks | "How to Define a Web Service " on page 68 |
| See also | <ul style="list-style-type: none"> "Connection Settings Dialog Box" on page 76 "Select Service from Systinet Dialog Box" on page 72 "Select Service from UDDI Dialog Box" on page 74 |

The following elements are included (unlabeled UI elements are shown in angle brackets>):

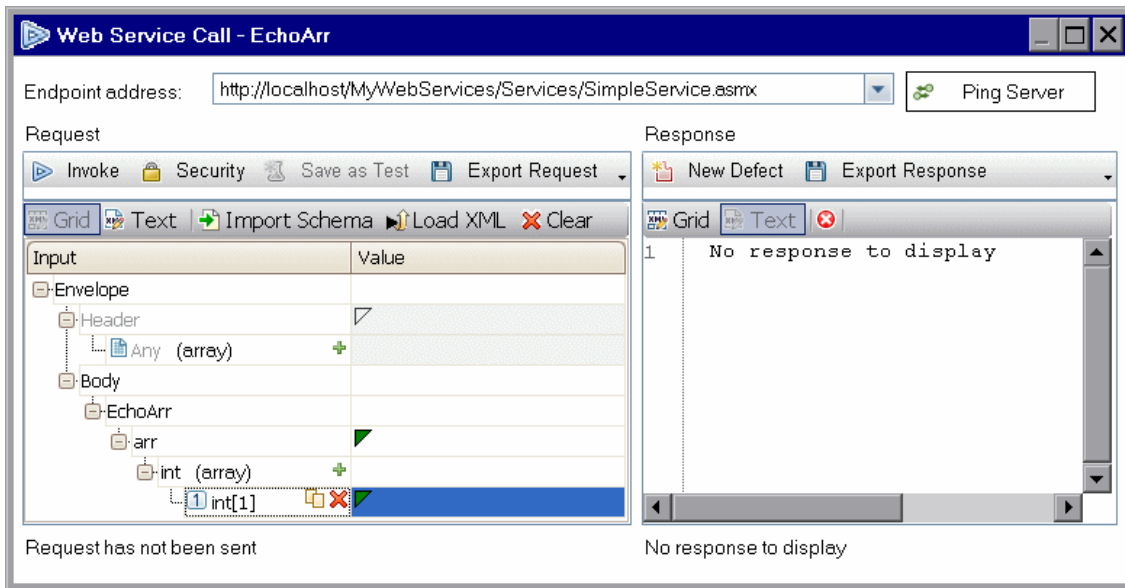
| UI Elements (A-Z) | Description |
|-------------------|---|
| [...] | <p>Lets you locate the WSDL by opening one of the following:</p> <ul style="list-style-type: none"> File. An Open File dialog box URL. A new browser window Systinet. Select Service from Systinet dialog box UDDI. Select Service from UDDI dialog box |

| UI Elements (A-Z) | Description |
|----------------------------|---|
| Connection Settings | Opens the Connections Settings dialog box for configuring the authentication and proxy settings of the server hosting the WSDL. For details, see " Connection Settings Dialog Box " on page 76. |
| Import WSDL from | The WSDL source: <ul style="list-style-type: none">• File. The file system• URL. A URL• Systinet. A Systinet registry• UDDI. A UDDI server |
| WSDL Location | Location of WSDL: <ul style="list-style-type: none">• File: full path and file name• URL: Complete URL• Systinet: Systinet ID• UDDI: Server key Use the Browse button to locate the WSDL. |

Web Service Call - <Operation_Name> Dialog Box

This Spot Tester dialog box allows you to test individual operations of a service. You set up a request and view the SOAP response to see if the operation functioned properly.

For details, see "Spot Testing Operations " on page 65.



| | |
|------------------------------|---|
| <p>To access</p> | <p>To open the Spot Tester for a specific operation:</p> <ol style="list-style-type: none"> 1. Select an operation in the Interaction tab. 2. Click the Web Service Call button. <p>To open the Spot Tester with a view of all available operations:</p> <ol style="list-style-type: none"> 1. Right-click an operation in the expanded Application Components tree. 2. Select Invoke. |
| <p>Relevant tasks</p> | <p>"How to Define a Web Service " on page 68</p> |

User interface elements are described below.

| UI Elements (A-Z) | Description |
|-------------------|---|
| <timestamp> | The date and time that the request was sent and the response received. This appears in the bottom of the window. |
| Endpoint address | The address of the server over which to submit the request. |
| Ping Server | Pings the server in the Endpoint address URL and opens a popup indicating its status: <i>Available</i> or <i>Unavailable</i> . |
| Request Pane | The request values and controls. For details, see the "Web Service Call - <Operation_Name> Dialog Box " on previous page. |
| Response Pane | The SOAP response. For details, see the "Web Service Call - <Operation_Name> Dialog Box " on previous page. |

Web Service Call Operation - Request Pane

User interface elements are described below:

| UI Elements (A-Z) | Description |
|-------------------|--|
| Clear | Clears the values you assigned to the arguments in the Grid tab. |
| Export Request | Exports the request SOAP message to an XML file. |
| Grid tab | <p>A list of all arguments, and a column for setting their values. Click on a triangle adjacent to an argument to include/exclude it from the request.</p> <ul style="list-style-type: none"> • A filled-in triangle includes the argument in the request; An empty triangle excludes the optional argument from the request. • An argument with an ABC icon can be parameterized. • A NIL icon indicates that argument can have a NIL value. • A green plus sign indicates that you can add array elements. |

| UI Elements (A-Z) | Description |
|----------------------|---|
| Import Schema | Allows you to open an XSD schema file for loading the structure of the request. |
| Invoke | Sends the request to the server address specified in the Endpoint address box. |
| Load XML | Allows you to open a structured XML file, for the purpose of loading values for the request. |
| Save as Test | Saves the call as a Service Test compatible test, to a location in ALM. The saved test includes the request's property values and security settings, and the response's values as checkpoints. Note: This option is only available when Service Test is installed on the machine. |
| Security | Opens the Security Setting for Operation <Operation Name> dialog box, allowing you to configure the security properties for the Web service. For details, see " Web Service Security " on page 103 . |
| Text tab | The request in XML format. |

Web Service Call Operation - Response Pane

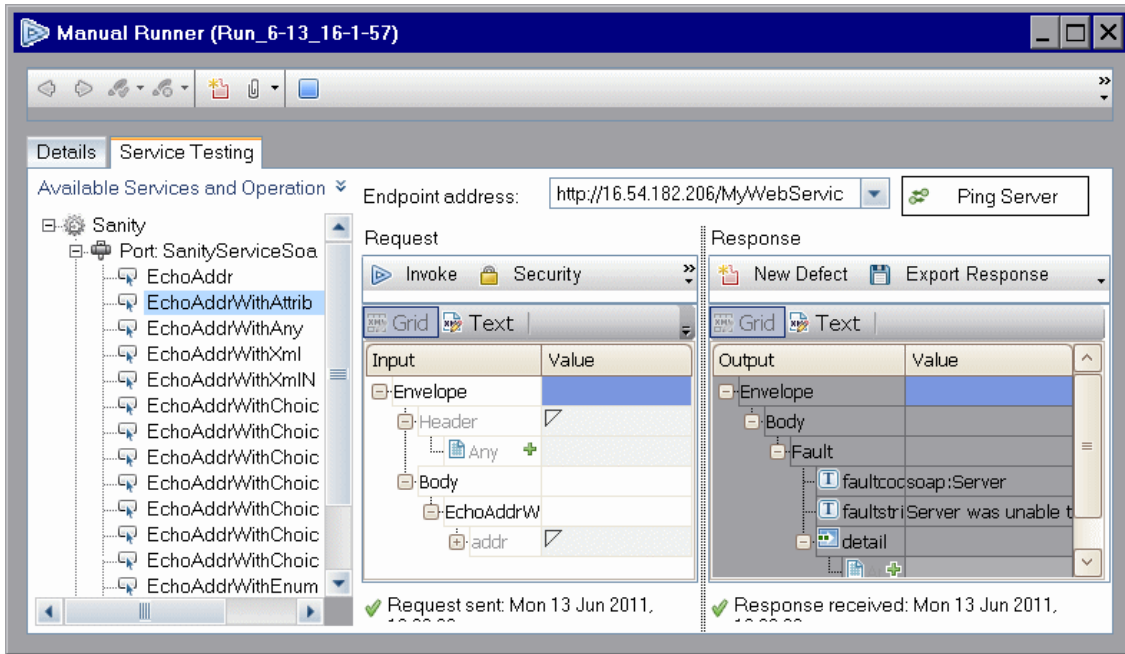
User interface elements are described below:

| UI Elements (A-Z) | Description |
|------------------------|---|
| Export Response | Opens the Save As dialog box, allowing you to export the SOAP response to an XML file. |
| Grid tab | A list of all response parameters, and their expected values. |
| New Defect | Opens the New Defect dialog box for adding a new defect relating to this operation, to the current project. |
| Text Tab | A textual (XML) representation of the SOAP response resulting from the request. |

Manual Runner Window








This window lets you manually run a Web Service test step. This applies primarily to tests containing Web Service type contract files.

To verify the functionality of a specific operation, use spot testing. For details, see "Spot Testing Operations " on page 65.



| | |
|------------------------------|---|
| <p>To access</p> | <p>Perform the following steps:</p> <ul style="list-style-type: none"> • Select a test set in the Test Lab module's Test Sets tab. • Click the Execution Grid tab. • Select a test and choose Run > Run with Application Component Manual Runner. • Click the Service Testing tab (by default this is open). <p>For tests not based on Web Service WSDL contracts, select Run with Manual Runner. For details, see the HP Application Lifecycle Management User Guide.</p> |
| <p>Relevant tasks</p> | <p>"How to Define a Web Service " on page 68</p> |

User interface elements are described below.

| UI Elements | Description |
|---|---|
|  | Previous Step. Navigates to the previous test step. |
|  | Next Step. Navigates to the next test step. |
|  | <p>Set Pass Status. Sets the test status to Pass. The drop down list provides the following options:</p> <ul style="list-style-type: none"> • Pass Selected. Sets status to Pass for selected steps. • Pass All. Sets status to Pass for all steps of the test. |
|  | <p>Set Fail Status. Sets the test status to Fail. The drop down list provides the following options:</p> <ul style="list-style-type: none"> • Fail Selected. Sets status to Fail for selected steps. • Fail All. Sets status to Fail for all steps of the test. |
|  | Opens the New Defect dialog box for creating a defect for the current test and test set. |
|  | <p>Add Attachment to Run. Adds an attachment to the test run. An attachment can be a file, URL, snapshot of your application, an item from the clipboard, or system information. The drop down list provides the following options:</p> <ul style="list-style-type: none"> • Attachment to Step. Adds an attachment to the current step. • Attachment to Run. Adds an attachment to all steps. |
|  | <p>End Run. Stops the test run.</p> <p>Note: If you stop a test run before it is complete, you can resume the test run from the point at which it was stopped. In the Execution Grid tab, select Run > Continue.</p> |
| Details tab | <p>Contains the creation and status details of the test run: Run Name, Tester, Execution Time, Status, and Execution Date.</p> <p>Tip: Set the status based on the result of the manual test run. You can set it to <code>Block</code>, <code>Failed</code>, <code>N/A</code>, <code>No Run</code>, <code>Not Completed</code>, or <code>Passed</code>.</p> |
| Service Testing | Displays the properties of the test, its request, and its response. |

| UI Elements | Description |
|-------------|-------------|
| tab | |

Manual Runner - Service Testing Tab

User interface elements are described below.

| UI Elements | Description |
|-----------------------------------|---|
| <timestamp> | The date and time that the request was sent and the response received. This appears in the bottom of the window. |
| Available Services and Operations | A tree hierarchy of all services and operations available for the test. |
| Endpoint address | The address of the server over which to submit the request. By default, the Manual Runner uses the endpoint from the contract file. To override it, specify a different endpoint in this field. |
| Ping Server | Pings the server in the Endpoint address URL and opens a popup indicating its status: <i>Available</i> or <i>Unavailable</i> . |
| Request Pane | The request values and controls. For details, see the "Manual Runner Window" on page 83. |
| Response Pane | The server response. For details, see the "Manual Runner Window" on page 83. |

Manual Runner - Service Testing Tab - Request Pane

User interface elements are described below:

| UI Elements (A-Z) | Description |
|-------------------|---|
| Clear | Clears the values you assigned to the arguments in the Grid tab. |
| Export Request | Exports the request SOAP message to an XML file. |

| UI Elements (A-Z) | Description |
|----------------------|--|
| Grid tab | <p>A list of all arguments, and a column for setting their values. Click on a triangle adjacent to an argument to include/exclude it from the request.</p> <ul style="list-style-type: none"> • A filled-in triangle includes the argument in the request; An empty triangle excludes the optional argument from the request. • An argument with an ABC icon can be parameterized. • A NIL icon indicates that the argument can have a NIL value. • A green plus sign indicates that you can add array elements. |
| Import Schema | <p>Allows you to open an XSD schema file for loading the structure of the request.</p> |
| Invoke | <p>Sends the request to the server address specified in the Endpoint address box.</p> |
| Load XML | <p>Allows you to open a structured XML file, for the purpose of loading values for the request.</p> |
| Save as Test | <p>Saves the call as a Service Test compatible test, to a location in ALM. The saved test includes the request's property values and security settings, and the response's values as checkpoints.</p> <p>Note: This option is only available when HP Service Test is installed on the machine.</p> |
| Security | <p>Opens the Security Settings for Operation <Operation Name> dialog box, allowing you to configure the security properties for the operation.</p> <ul style="list-style-type: none"> • To set the security for this operation only, clear the Use the port's security settings check box and provide the security details. • To set the security for all operations on the port, click Edit port's settings. <p>For details, see "Web Service Security " on page 103.</p> |

| UI Elements (A-Z) Description | |
|-------------------------------|----------------------------|
| Text tab | The request in XML format. |

Manual Runner - Service Testing Tab - Response Pane

User interface elements are described below:

| UI Elements (A-Z) Description | |
|-------------------------------|---|
| Export Response | Opens the Save As dialog box, allowing you to export the SOAP response to an XML file. |
| Grid tab | <p>A list of all response parameters, and their expected values.</p> <p>Click on a triangle adjacent to an argument to include/exclude it from the request.</p> <ul style="list-style-type: none"> • A filled-in triangle includes the argument in the request; An empty triangle excludes the optional argument from the request. • An argument with an ABC icon can be parameterized. • A NIL icon indicates that argument can have a NIL value. |
| New Defect | Opens the New Defect dialog box for adding a new defect relating to this operation, to the current project. |
| Text tab | A textual (XML) representation of the SOAP response resulting from the request. |

Troubleshooting and Limitations - Web Services

This section describes troubleshooting and limitations for working with Web Service type application components.

- **Issue:** An operation runs successfully from the Spot Tester, but when you try to run the test that you created using the Spot Tester's **Save as Test** option, it fails.
Reason 1: The test contains a dynamic parameter, such as Correlation ID, that changes for every replay.
Workaround 1: In Service Test/UFT, assign the checkpoint a data source that reflects the changing property value.
Reason 2: The service uses multiple ports. The Spot Tester runs the operation using the specified port, while the saved test uses the first port, as it appears in the **Interaction** tab's **Service address** field.
Workaround 2: Open the **Interaction** tab and select **Override address**. Paste the endpoint for the port (other than the first) into the **Service address** field.
Note: This will modify the endpoint address in all tests that are associated with this Web service. This may solve the issue for the specific operation, but will also affect other tests that call the operation using a different port.
- If you add attachments to a defect from the Spot Tester, they will not be available until you post the defect.
- You can import services from all versions of Systinet, including 4.00. Importing Service Test Management to Systinet, is only supported for Systinet 4.00.
- WSDL files with non-English characters in its tags, are not supported.
- You cannot import WSDL files that are located in a secure shared network.
Workaround: Copy the file to a local folder or to a non-secure location.
- Importing from UDDIs through proxy servers is not supported.
- If you override a Web service address in the Interaction tab, and invoke the service through the right-click menu, the displayed Endpoint Address in the Spot Tester window may still show the original endpoint.

Managing Application Components

Concepts

- "Managing Application Components Overview " on page 91

Tasks

- "How to Manage Components " on page 95

Reference

- "Update Linked Tests Dialog Box" on page 99
- "Discover Components Wizard" on page 101
- "Generate Components Wizard" on page 102

Concepts

This section includes:

- ["Managing Application Components Overview "](#) on page 91

Managing Application Components

Overview

After you define an application component, you can view and modify its definitions and details. You can also resolve alerts and view the changes on your component. Using the Application Components tree, you can reorganize the components under different folders.

This section includes:

- Application Component Details
- Locating Application Components
- Handling Rule Violations or Change Alerts
- Copying Components
- Adding Attachments
- Restoring and Removing Components
- E-mailing Components

Application Component Details

In the Application Components tree, you can view and modify the location of components and folders. For Web service components, you can view information such as the WSDL location, override the service address, and modify connection settings.

For details, see ["How to Manage Components "](#) on page 95.

Locating Application Components

You can search for a particular entity in the Application Components tree— folder, component, or group, by one of their **Details** tab fields. The search utility allows wildcards and searching by a specific field. For example, you can search for a component that begins with the letter "s".

Tip: If you applied a filter to the tree, the search is restricted to the folders and components currently displayed.

For details, see ["How to Manage Components "](#) on page 95.

Handling Rule Violations or Change Alerts

Service Test Management uses alert indicators to make you aware of changes

to the components or rule violations.

The changes can be those entered manually or those detected automatically by Service Test Management when you update a contract.

Rule violations are a result of non-compliance with the component's requirements. To complete your testing, you should try to resolve the alerts.

For additional information, see ["How to Manage Components " on page 95](#).

Updating Components

Service Test Management enables you to update an application component from its original location or from a different source. Once you update a component, any step that calls it from an another application, such as Service Test / UFT may be affected. Service Test / UFT, marks all steps that were affected with a warning icon, that allows you to open the Update wizard. This wizard lets you map service, port, and property names.

In addition, you can update all tests in your project's Test Plan with calls to application components stored in Service Test Management. When you run the Update Linked Tests utility, Service Test Management updates all Service Test / UFT tests that are linked to application components. If there are conflicts between the original application component and the new one, Service Test / UFT marks them with a warning icon. You can resolve the steps using the Update wizard in Service Test / UFT. All tests that call unresolved components are marked with an alert icon in the Test Plan module.

For details, see the ["How to Manage Components " on page 95](#)


Copying Components

You can copy an existing folder, component, or group and paste it to another location in the Application Components tree, or to another project. When copying a component to another project, the source and target projects must be open in separate browsers.

For more information, see ["How to Manage Components " on page 95](#).

Adding Attachments

The **Attachments** tab enables you to associate an attachment with a folder, component, or group. An attachment can be a file, URL, text from the clipboard, snapshot, or system information. It is identified by its name, associated application icon, size, and latest modification date and time. You can add, modify, and delete attachments.

The **Attachments** tab includes an icon  if the selected folder, component, or group has any attachments.

The **Attachments** tab for folders, components, and groups has the same functionality as the **Attachments** tab for other ALM entity types. For details on adding attachments, refer to the *HP Application Lifecycle Management User Guide*.

Restoring and Removing Components

You can permanently delete folders or components that are not used by a test or requirement. To see if a component is used by a test, see ["Tests Coverage" on page 195](#). If you try to delete components that are used by one or more tests, Service Test Management transfers them to the **Obsolete** folder. You cannot copy obsolete components.

You can restore components in the **Obsolete** folder to their original position in the Application Components tree if the folder still exists in the tree, or you can drag them to a different location in the tree.

For task details, see ["How to Manage Components " on page 95](#).

E-mailing Components

You can send e-mail about an entity (folder, component, or group) to other users. This enables you to routinely inform users about the status of your components. A link is included in the e-mail message that enables the recipient to go directly to the component. You can also include your own shortcuts using one of the formats described in ["ALM Direct Links " on page 25](#).

For task details, see ["How to Manage Components " on page 95](#).

Tasks

This section includes:

- "How to Manage Components " on page 95

How to Manage Components

This task describes how to work with your components by changing its details, updating a WSDL, resolving alerts, copying entities, and so forth.

This task includes the following steps:


- Manage component details - optional
- Find folders and components - optional
- Handle alerts - optional
- Update a Component - optional
- Copy entities - optional
- Remove and restore components - optional
- Mail components - optional

Manage component details - optional

Open the **Details** tab and modify the component's details as desired.

For more information, see ["Details " on page 30](#).

Find folders and components - optional

1. To search all of the application components, select **Filter**  **→ Clear Filter** to remove any filters.
2. Select **Edit > Find**. The Find dialog box opens.
3. Select an **Entity Type**: `Folder`, `Application Component`, or `Group`.
4. Select the field to search in the **Find in Field** box. For example to search for a component by its name, select **Component Name** field and specify a part or the whole name as a value.
5. Specify a value in the **Value to Find** box. The search engine is not case-sensitive and it finds parts of a string.
6. Click **Find Next** to locate the first and subsequent matches.

For details, see the ["Find Dialog Box" on page 37](#).

Handle alerts - optional

Click on an exclamation point adjacent to a component, to view its alerts **!**.

Determine which alerts to resolve. For details, see the ["Alerts Dialog Box" on](#)

[page 28](#).

Update a component - optional

To update a component contract:

1. Select the component you want to update.
2. Choose an update option:
 - To update a component using its original location, choose **Components > Update Component > Update Component**.
 - To update a component using a different location, choose **Component > Update Component > Update Component from**. Browse for a newer version of the contract in another location.

The update process begins. Service Test Management checks for changes in the contract and informs you if the component changed. For details, see ["Change Analysis " on page 267](#)

Update linked tests - optional

To update all linked tests (**Testing Status > Tests tab**) containing components:


1. Choose **Component > Update Component > Update Linked Tests**
2. In the Update Tests dialog box, review the list of tests that were detected for the update
3. Click **Update All Tests**. The update process begins. The mechanism updates the components linked to the test from their location in Service Test Management. The right column shows the update status. For details, see ["Update Linked Tests Dialog Box" on page 99](#).
4. To open a test in Service Test / UFT, select it in the grid and click the Launch Service Test button. This is useful to check a test that could not be updated or to check if a test step needs to be resolved due to the component update. Tests that have steps that need to be resolved are marked with an alert in the Test Plan module's test tree.
5. To resolve a test step in Service Test / UFT, click the step's alert icon and run the Update wizard. For details, see the Service Test / UFT user guide. Once you resolve a step, you need to manually remove the alert. In the Test Plan module's test tree, click on the alert and mark it as resolved. For details, see ["Change Analysis " on page 267](#)

Copy entities - optional

Using the right-click menu, copy existing folders, components, or groups onto the clipboard. Open the target folder in the components tree and paste the entity from the clipboard. To copy it to another project, open the target project in another browser tab and paste it in.

For details, see the "[Application Components Module Window](#)" on page 52.

Remove and restore components - optional

To remove a component, select it and click the **Delete**  button. If you remove a folder, its subfolders and all components in those folders are removed too. If a removed component is still being used by an existing test, Service Test Management stores it in the **Obsolete** folder.

Tip: To view all application components in the **Obsolete** folder, select the folder and click the **Refresh** button.

To remove a component entirely, make sure that no tests are using it. Click the **Test Script** tab in the **Test Lab** module to open the test in Service Test / UFT, and remove all steps calling the component. After you have removed all calls to the component, delete it from the **Obsolete** directory.

You can restore components in the **Obsolete** folder to their original position in the components tree, if the folder still exists in the tree. If the folder no longer exists, transfer the component to another location in the tree by dragging it from the **Obsolete** folder to the desired location.

Mail components - optional

Select a component and click the **Send by e-mail** button . A link included in the e-mail message enables the recipient to go directly to the component.

Edit the content, select recipients, and perform a spell check. For details about the Send Mail dialog box, click F1 or see the *HP ALM User Guide*.

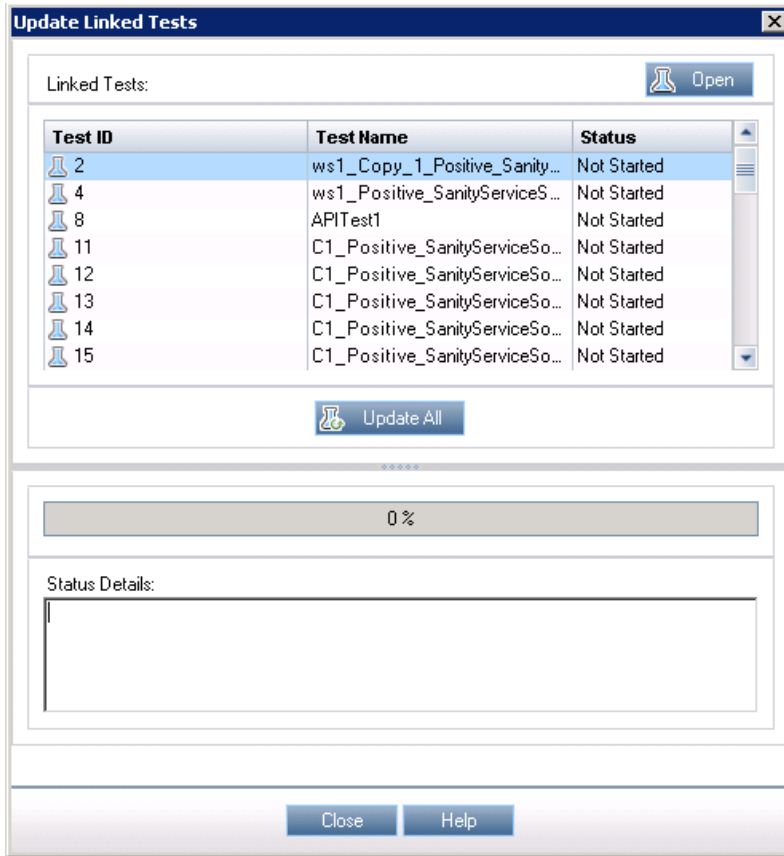
Reference

This section includes:

- "Update Linked Tests Dialog Box" on page 99
- "Discover Components Wizard" on page 101
- "Generate Components Wizard" on page 102

Update Linked Tests Dialog Box

This dialog box enables you to update all tests that are linked to application components, with the current version of its contract.



| | |
|------------------------------|---|
| To access | Select Components > Update Component > Update Linked Tests |
| Important information | Available only when Service Test / UFT is installed on the machine. |
| Relevant tasks | "How to Manage Components " on page 95 |
| See also | "Managing Application Components Overview " on page 91 |

The following elements are included:

| UI Elements | Description |
|-----------------------|---|
| Linked Tests | A list of tests calling application components stored in Service Test Management, that are linked to the components. <ul style="list-style-type: none">• Test ID. The unique ID of the test as it appears in the Test Details dialog box.• Test Name. The name the test as it appears in the Test Plan tree.• Status. The current status of the update: Not Updated, Updated, or Failed to Update. |
| Open | Launches Service Test / UFT with the test selected in the test list. |
| Update All | Begins the process of updating all tests listed in the above pane. |
| Status Details | A log indicating the results of the update. |

Discover Components Wizard

The Discover Components wizard automatically detects contract files in tests stored in ALM and creates application components based on these contracts. You can then organize the application components in Service Test Management and link them to requirements and tests.

| | |
|-----------------------|---|
| To access | Do one of the following: <ul style="list-style-type: none">• Select an entity in the Application Components tree, and choose Discover Components from the right-click menu.• Select Components > Discover Components. |
| Important | In order to enable this wizard, you need to have HP Unified Functional Testing or HP Service Test installed on the machine. |
| Relevant tasks | "How to Define Components " on page 22 |
| Wizard map | This wizard contains: Welcome Page > Select Tests Page > Options Page > Required Fields Page > Summary Page > Results Page |
| See also | "How to Define a Web Service " on page 68 |

Generate Components Wizard

This wizard enables you to generate multiple components at once.

| | |
|-----------------------|---|
| To access | Do one of the following: <ul style="list-style-type: none">• Select an entity in the Application Components tree, and choose Generate Components from the right-click menu.• Select Components > Generate Components. |
| Relevant tasks | "How to Define Components " on page 22 |
| Wizard map | This wizard contains: Select Source Page > Add Components Page > Set Required Fields Page > View Summary Page > Show Log Page |
| See also | "How to Define a Web Service " on page 68 |

Web Service Security

Concepts

- "Setting Security Overview" on page 106
- "Security Levels " on page 107
- "Security Scenarios Overview" on page 108
- "Web Service Security Scenario " on page 109
- "WCF Scenario Settings " on page 112
- "WCF Service (CustomBinding) Scenario Overview" on page 113
- "WCF Service (Federation) Scenario Overview " on page 114
- "WCF Services (WSHttpBinding) Scenario Overview " on page 115
- "Advanced Security Settings " on page 117

Tasks

- "How to Set Security for a Web Service on the Port Level" on page 119
- "How to Set Security for a Specific Web Service Operation" on page 121
- "How to Set Security for a Standard Web Service" on page 122
- "How to Set Security for a WCF Service" on page 124
- "How to Set up Common Web Services Security Scenarios " on page 125
- "How to Customize Security for WCF Type Web Services " on page 129
- "How to Test Web Services that use WS-Security or SSL" on page 133
- "How to Set up Advanced Standards Testing " on page 135

Reference

- "Security User Interface" on page 138
- "Security Settings for Port / Operation <Name> Dialog Box" on page 139
- "Web Service Scenario " on page 141
- "WCF Service (Custom Binding) Scenario" on page 147
- "WCF Service (Federation) Scenario" on page 149

- "WCF Service (WSHttpBinding) Scenario" on page 151
 - "Advanced Settings Dialog Box" on page 154
 - "Encoding Tab" on page 155
 - "Advanced Standards Tab" on page 156
 - "Security Tab" on page 157
 - "HTTP and Proxy Tab" on page 160
 - "Select Certificate Dialog Box " on page 162
- "Troubleshooting and Limitations - Web Service Security " on page 164

Concepts

This section includes:

- "Setting Security Overview" on page 106
- "Security Levels " on page 107
- "Security Scenarios Overview" on page 108
- "Web Service Security Scenario " on page 109
- "WCF Scenario Settings " on page 112
- "WCF Service (CustomBinding) Scenario Overview" on page 113
- "WCF Service (Federation) Scenario Overview " on page 114
- "WCF Services (WSHttpBinding) Scenario Overview " on page 115
- "Advanced Security Settings " on page 117

Setting Security Overview

When building Web Service applications, there is a challenge in building scalable applications that are secure. You can secure Web Services by having the message sent over a secure transport, such as Secure Sockets Layer (SSL), or by applying security at the message level, also known as **WS-Security**.

For testing a secured service, answering the following questions will help you define your security scenario.

- Is there transport security, such as SSL? What is the HTTPS URL?
- Is basic authentication required?
- Is mutual authentication required?
- What type of security is required in the SOAP header?

For task details, see ["How to Set Security for a Specific Web Service Operation"](#) on page 121.

Security Levels

Service Test lets you set the security for a service on two levels— port or operation. If you define a security for a port, all of its operations use these settings, by default. When working on an operation, you can override the default port security and customize the security for a particular operation.

For user interface details, see "[Security User Interface](#)" on page 138.

Security Scenarios Overview

Service Test provides several built-in scenarios for configuring security in Web Service calls.

A security scenario represents a typical security implementation for a Web Service. It contains information such as authentication, encoding, proxy, certificates, and so forth.

A default **Web Service** scenario can be used for most Web services. It enables you to configure both transport and message-level security. Service Test support for message-level security lets you manually configure the security elements such as tokens, message signatures, and encryption. For details, see "[Web Service Security Scenario](#)" on page 109.

WCF scenarios enables you to configure security for HTTP or custom bindings and work with advanced specifications, such as **WS-SecureConversation**.

The built-in security scenarios are:

- "[Web Service Security Scenario](#)" on page 109
- "[WCF Service \(CustomBinding\) Scenario Overview](#)" on page 113
- "[WCF Service \(Federation\) Scenario Overview](#)" on page 114
- "[WCF Services \(WSHttpBinding\) Scenario Overview](#)" on page 115

Use the default Web Service scenario for:

- Simple Web Services where no advanced standards are required.
- Web services using HTTP transport level security.
- Web services using message level security (WS-Security) for SOAP 1.1

Use a WCF scenario for:

- WCF Services that utilize advanced security and WS-Specifications.
- Web services using message level security (WS-Security) for SOAP 1.2

Such services can be written in various platforms such as WCF (Windows Communication Foundation), Metro (WSIT), and Axis2. Service Test also supports proprietary standards and transports.

Web Service Security Scenario

The default Web Service scenario is based on the WS-Security specification. This scenario lets you place security credentials in the actual SOAP message.

When a SOAP message sender sends a request, the security credentials, known as **tokens**, are placed in the SOAP message. When the Web server receives the SOAP request, it does not need to send additional requests to verify the integrity of the sender. The server verifies that the credentials are authentic before letting the Web Service execute the application. By not having to go back to the source of the credentials, the application's scalability improves significantly.

To further secure Web Services, it is common to use digital signatures or encryption for the SOAP messages. Digitally signing a SOAP message verifies that the message has not been altered during transmission. Encrypting a SOAP message helps secure a Web Service by making it difficult for anyone other than the intended recipient to read the contents of the message.

The Security Settings dialog box provides the following tabs for the Web Service scenario: **HTTP**, **WS-Security**, and **WS-Addressing**. The **HTTP** tab lets you configure the transport security, the **WS-Security** tab handles the message-level security, and the **WS-Addressing** tab sets the addressing version.

Transport Level Security

The transport level security includes the authentication and proxy server information. You can also specify Keep Alive preferences and connection timeout.

Message Level Security

The **WS-Security** tab lets you set the message level security through tokens, signatures and encryption.

To support WS-Security, Service Test enables you to create security tokens for your script. You can create multiple tokens and set their properties. After creating a token, you use it to sign or encrypt a SOAP message.

The Web Services security mechanism associates security tokens with messages. This mechanism supports several security token formats to accommodate a variety of authentication requirements. For example, a client might need to provide a proof of identity or a security certificate.

The available tokens are: **UserName**, **X509 Certificate**, **Kerberos**, **Kerberos2**, and **SAML**.

- **UserName**. The **User Name** token contains user identification information for the purpose of authentication: **User Name** and **Password**. You can also specify **Password Options**, indicating how to send the password to the server for authentication: **Text**, **None**, or **Hash**. and indicate whether to include a timestamp.
- **X509 Certificate**. This token is based on an X.509 certificate. To obtain a certificate, you can either purchase it from a certificate authority, such as VeriSign, Inc. or set up your own certificate service to issue a certificate. Most Windows servers support the public key infrastructure (PKI), which enables you to create certificates. You can then have it signed by a certificate authority or use an unsigned certificate.
- **Kerberos /Kerberos2**. (For Windows 2003 or XP SP1 and later) The Kerberos protocol is used to mutually authenticate users and services on an open and unsecured network. Using shared secret keys, it encrypts and signs user credentials. A third party, known as a KDC (Kerberos Key Distribution Center), authenticates the credentials. After authentication, the user may request a service ticket to access one or more services on the network. The ticket includes the encrypted, authenticated identity of the user. The tickets are obtained using the current user's credentials.

The primary difference between the Kerberos and Kerberos2 tokens, is that Kerberos2 uses the Security Support Provider Interface (SSPI), so it does not require elevated privileges to impersonate the client's identity. In addition, the Kerberos2 security token can be used to secure SOAP messages sent to a Web Service running in a Web farm.

- **SAML Token**. SAML is an XML standard for exchanging security-related information, called assertions, between business partners over the Internet. The assertions can include attribute statements, authentication, decision statements, and authorization decision statements.

SAML uses brokered authentication with a security token issued by STS (Security Token Service). The STS is trusted by the client and the Web Service to provide interoperable security tokens. SAML tokens are important

for Web Service security because they provide cross-platform interoperability and a means of exchanging information between clients and services that do not reside within a single security domain.

Message Signatures and Encrypted Data

When you add a security token to a SOAP message, it is added to the SOAP message in the form of an XML element in the WS-Security SOAP header.

The message, however, is exposed and therefore requires additional security. This is especially true when the credentials, including the password, are sent in plain text as it is with role-based security.

The two methods used to secure the data are message signatures and message encryption.

- **Message Signatures.** Message Signatures are used by the recipients to verify that messages were not altered since their signing. The signature is in the form of XML within the SOAP message. The recipient checks the signature to make sure it is valid.
- **Message Encryption.** Although the XML message signature offers a mechanism for verifying that the message has not been altered since it was signed, it does not encrypt the SOAP message—the message is still plain text in XML format. To secure the message in order that it should not be exposed, you encrypt it, making it difficult for an intruder to view and obtain a user's password.

For task details, see ["How to Set Security for a Specific Web Service Operation"](#) on page 121.

WCF Scenario Settings

This section describes the WCF security scenarios. It includes:

- "WCF Service (CustomBinding) Scenario Overview" on page 113
- "WCF Service (Federation) Scenario Overview " on page 114
- "WCF Services (WSHttpBinding) Scenario Overview " on page 115

WCF Service (Custom Binding) Scenario Overview

The **WCFService(Custom Binding)** scenario enables the highest degree of customization. Since it is based upon WCF customBinding standard, it enables you to test most WCF services, along with services on other platforms such as Java-based services that use the WS - *<spec_name>* specifications.

Use the **WCFService(Custom Binding)** scenario to configure a scenario that does not comply with any of the predefined security scenarios. You can customize the transport and encoding settings:

- **Transport.** HTTP, HTTPS, TCP, or NamedPipe
- **Encoding.** Text, MTOM, or WCF Binary

You can also provide additional security information:

- **Authentication mode.** The type of authentication, such as **None**, **AnonymousForCertificate**, and so forth. The options are available from the drop down list.
- **Bootstrap policy.** For the **SecureConversation** authentication mode, you can specify a bootstrap policy.
- **Net security.** The network security for **TCP** and **Named Pipe** type transports. Typical values are **None**, **Windows stream security**, or **SSL stream security** available from the field's drop down list. For services with HTTP transport, you should set the value to **None**. To enable SSL for HTTP, select **HTTPS transport**.

For task details, see ["How to Set Security for a WCF Service"](#) on page 124.

For user interface details, see ["WCF Service \(Custom Binding\) Scenario"](#) on page 147.

Note: For WSE3 security configurations, use the **WCFService(Custom Binding)** Scenario. For details, see ["How to Customize Security for WCF Type Web Services"](#) on page 129.

WCF Service (Federation) Scenario Overview

In the **WCF Service (Federation)** scenario, the client authenticates against the STS (Security Token Service) to obtain a token. The client uses the token to authenticate against the application server.

Therefore, two bindings are needed, one against the STS and another against the application server.

You define the bindings in two stages:

- Define an STS binding in the **Referenced binding** field.
- Provide security details for the application server's security scenario in the following areas:
 - **Server.** The transport and encoding methods.
 - **Security.** The authentication mode, such as `IssuedToken`, `SecureConversation`, and so forth.
 - **Identity.** Information about the server certificate and expected DNS.
 - **STS.** Settings related to the STS, such as the endpoint address and binding.

For task details, see ["How to Set Security for a WCF Service"](#) on page 124.

For user interface details, see ["WCF Service \(Federation\) Scenario"](#) on page 149.

WCF Services (WSHttpBinding) Scenario Overview

Note: The **WCFService(WSHttpBinding)** scenario only supports the testing of WCF services which utilize *wsHttpBinding* and incorporate some level of security. To test WCF services that use *wsHttpBinding* but have no security, use the **WCFService(CustomBinding)** scenario.

In the **WCF Service (WSHttpBinding)** scenario, you can select from several types of authentication: None, Windows, Certificate, or Username (message protection).

For all of the authentication types, you can apply advanced settings as described in "[Advanced Security Settings](#) " on page 117.

For user interface details, see "[WCF Service \(WSHttpBinding\) Scenario](#)" on page 151.

No Authentication (Anonymous)

In this scenario, the client uses the server's certificate to encrypt a message; there is no client authentication. Use this scenario to test Web Services where the:

- Client uses the server's X.509 certificate for encryption
- Client is not authenticated
- Communication may utilize advanced standards such as secure conversation or MTOM.

Windows Authentication

This scenario uses Windows Authentication. If you are testing a WCF service that has not been customized and uses the default configuration, use this type of scenario.

Use this scenario to test Web Services where the:

- Client and server use Windows authentication
- Security is based on Kerberos or SPNEGO negotiations

- Communication may utilize advanced standards such as secure conversation or MTOM.

Certificate Authentication

In this **WCF WSHttpBinding** scenario, the client uses the server's X.509 certificate to encrypt the message and its own certificate for a signature.

Use this scenario to test Web Services where the:

- Client uses the server's X.509 certificate for encryption.
- Client uses its own X.509 certificate for signatures.
- Communication may utilize advanced standards such as secure conversation or MTOM.

Username Authentication (Message Protection)

In the **WCF WSHttpBinding** scenario, the client uses the server's X.509 certificate to encrypt the message, and sends a user name and password to authenticate itself.

Use this scenario to test Web Services where the:

- Client uses the server's X.509 certificate for encryption.
- Client is authenticated with a username and password.
- Communication may utilize advanced standards such as secure conversation or MTOM.

Advanced Security Settings

This scenario's settings let you customize a **WCFService(CustomBinding)** scenario in the areas of **Encoding**, **Advanced Standards**, **Security**, or **HTTP and Proxy**.

Not all settings are relevant for all scenarios, so some of them might be disabled or hidden depending on the scenario.

For details, see the "[Advanced Settings Dialog Box](#)" on page 154.

Tasks

This section includes:

- ["How to Set Security for a Web Service on the Port Level" on page 119](#)
- ["How to Set Security for a Specific Web Service Operation" on page 121](#)
- ["How to Set Security for a Standard Web Service" on page 122](#)
- ["How to Set Security for a WCF Service" on page 124](#)
- ["How to Set up Common Web Services Security Scenarios " on page 125](#)
- ["How to Customize Security for WCF Type Web Services " on page 129](#)
- ["How to Test Web Services that use WS-Security or SSL" on page 133](#)
- ["How to Set up Advanced Standards Testing " on page 135](#)

How to Set Security for a Web Service on the Port Level

This task describes how to configure security settings for a Web service on the port level. You can override this by modifying the settings for a specific test step. For details, see ["How to Set Security for a Specific Web Service Operation" on page 121](#).

This task includes the following steps:

- Prerequisites
- Open the Security Setting dialog box
- Load existing settings - optional
- Create a new security scenario
- Save the settings - optional

1. Prerequisites

Import at least one Web service.

2. Open the Security Setting dialog box

Select a Web service in the Application Components tree and click the **Interaction** tab. Click the **Security** toolbar button.

3. Load existing settings - optional

To load an existing set of Service Test security settings, click **Import** and locate the **.stss** (Service Test Security Scenario) file.

4. Create a new security scenario

Use the Security Settings dialog box to create a new security scenario or modify a loaded one:

- a. In the Service Details box, select a scenario type: `Web Service`, `WCF Service`, and so forth.
- b. Configure the security settings for the selected scenario. For details, see the ["Security Settings for Port / Operation <Name> Dialog Box" on page 139](#).

- c. For Web Service type scenarios, add tokens, signatures, and encryption. For details, see "[Web Service Scenario](#) " on page 141.

5. **Save the settings - optional**

Click **Save** to save the settings to an **.stss** file.

How to Set Security for a Specific Web Service Operation

This task describes how to configure security settings for Web Services.

This task includes the following steps:

- Prerequisites
- Open the Security Setting view
- Enable the Service Settings details
- Specify a scenario type

1. Prerequisites

Import a Web Service and drag an operation onto the canvas.

2. Open the Security Setting view

Select a Web service in the Application Components tree and click the **Interaction** tab. Expand the port in the bottom pane, and select an operation. Click the **Web Service Call** button. In the Web Service Call window, click the **Security** button.

3. Enable the Service Settings details

Clear the **Use the port's security settings** option.

4. Specify a scenario type

In the **Service Details** list, select a scenario type: `Web Service`, `WCF Service`, and so forth.

5. Configure the security settings

Configure the security settings for the selected scenario. For task details, see one of the following sections

- ["How to Set Security for a Standard Web Service" on page 122](#)
- ["How to Set Security for a WCF Service" on page 124](#)

For user interface information, see the ["Security Settings for Port / Operation <Name> Dialog Box" on page 139](#).

How to Set Security for a Standard Web Service

This task describes how to configure security settings for a standard Web Service. This mode lets you define the HTTP transport information and security elements such as tokens. For UI details, see "[Web Service Scenario](#)" on page 141.

This task includes the following steps:

- Create a Web Service scenario
- Configure the HTTP settings
- Define security elements (optional)
- Set the WS-Addressing version (optional)

1. Create a Web Service scenario


Select a Web service in the Application Components tree and click the **Interaction** tab. Expand the port in the bottom pane, and select an operation. Click the **Web Service Call** button. In the Web Service Call window, click the **Security** button.

2. Configure the HTTP settings

Open the **HTTP** tab, and set the transport and proxy information. For details, see the "[Web Service Scenario](#)" on page 141.





3. Define security elements (optional)

Click the **WS-Security** tab. Add tokens, message signatures, and encryption.

- To add a token, click  and select a token type. Provide the token details in the lower pane. The fields differ based on the token type. For details, see "[Message Level Security](#)" on page 109.

Note: When adding a SAML token, if you have the complete SAML token string, you can paste it directly into the **AssertionIDReference** field. If you do not have the complete token string and you want to configure the token manually, make sure that the first row in the schema contains the **Assertion** property— not the

AssertionIDReference. You can change this using the grid's drop down menu.

- To add a message signature, (you must first add at least one token), click  and select a token in the **Signing token** box, usually an X.509 token. Provide the other required information: For details, see the "[Web Service Scenario](#)" on page 141
- To add a message encryption, (you must first add at least one token), click  and select the token that will do the encryption in the **Encryption token** box. Provide any other required information or accept the defaults.
- Organize the security elements in their order of priority. Use the **Up**  and **Down**  arrows to set the priorities. The basic order is tokens, followed by message signatures, and then encryption. In addition, your service may also require a specific order for the tokens.

4. **Set the WS-Addressing version (optional)**

Click the **WS-Addressing** tab. Select the relevant version or `None` if WS-Addressing is not used.

For details about the security elements, see the "[Web Service Security Scenario](#)" on page 109.

For user interface details see the "[Security Settings for Port / Operation <Name> Dialog Box](#)" on page 139.

How to Set Security for a WCF Service

This task describes how to configure security settings for a Web Service using WCF. For guidelines about selecting a WCF service scenario, see "[WCF Scenario Settings](#)" on page 112.

This task includes the following steps:

- Create a WCF scenario
 - Configure the security settings
 - Configure advanced settings - optional
 - Save the scenario - optional
1. **Create a WCF scenario**
 - For port level security, click on the service's port in the **Toolbox** pane and select **Security Settings** from the shortcut menu.
 - For step level security, open the **Security Settings** view in the **Properties** pane. Clear the **Use the port's security settings** option.

Select the desired **WCF Service** in the **Service Details** list.
 2. **Configure the security settings**

Configure the settings as described in the "[Security Settings for Port / Operation <Name> Dialog Box](#)" on page 139.
 3. **Configure advanced settings - optional**

Click the **Advanced** button to configure advanced security settings. For details, see the "[Advanced Settings Dialog Box](#)" on page 154.
 4. **Save the scenario - optional**

Your security scenario is automatically saved with the test. If, however, you also want to use the settings for another test, without having to redefine the scenario, you can save it to an **.stss** file.

To save the scenario, click the **Save** button. Specify a location for the scenario file. To use the file in another test, click **Import**.

How to Set up Common Web Services Security Scenarios

This section illustrates several common security scenarios. The examples apply when using the default Web Service security scenario. For WCF- based services, the **WCFService(Custom Binding)** scenario is recommended. For additional examples, see "How to Customize Security for WCF Type Web Services " on page 129.

You can set security for all operations in a Web service port or for a specific step in your test.


To set security for a port, see "How to Set Security for a Specific Web Service Operation" on page 121.

This section includes the following:

- Authenticating with a Username Token
- Signing with an X.509 Certificate
- Signing a Specific Element with an X.509 Certificate
- Encrypting with a Certificate
- Authenticating with a Username Token and Encrypting with an X.509 Certificate
- Encrypting and Signing a Message


Authenticating with a Username Token

To send a message level username/password token (a UserName token):


1. Select the **Web Service** scenario from the **Service Details** list, and click the **WS-Security** tab.
2. Click the **Add Token**  button and add a **Username** token.
3. Customize the token details, such as username and password.

Signing with an X.509 Certificate

To send a message using a X.509 certificate for a digital signature.

1. Select the **Web Service** scenario from the **Service Details** list, and click the **WS-Security** tab.
2. Select **X509 Certificate** token from the **Security Token**  drop down

list.



3. Fill in the token details to reference your private certificate. Make sure to enter a value in the **Token name** field.
4. Select a **Referencetype**. Since this token is used for a signature, the most common type is `BinarySecurityToken`.
5. Click the **Add Message Signature** button . In the **Signing Token** drop down, select the token you created in the previous steps.

Note: The certificate needs to be installed in the Windows certificate store. In the example above, you need to set the actual store name, store location, and subject name of your certificate.

Signing a Specific Element with an X.509 Certificate

It is possible to sign only a specific element in a message. The following example signs a specific element using an XPATH expression.


To send a message using an X.509 certificate for a digital signature:


1. Select the **Web Service** scenario from the **Service Details** list, and click the **WS-Security** tab.
2. Select **X509 Certificate** token from the **Security Token**  drop down list.
3. Fill the token details to reference your private certificate. Make sure to enter a value in the **Token name** field.
4. Select a **Referencetype**. Since this token is used for a signature, the most common type is `BinarySecurityToken`.
5. Click the **Add Message Signature** button . In the **Signing Token** drop down, select the token you created in the previous steps.
6. Scroll down to the **XPath** field. Enter an XPath expression, for example:

```
// *[local-name(.)='Body'].
```

Encrypting with a Certificate




To encrypt a message using the service's X.509 certificate:

1. Select the **Web Service** scenario from the **Service Details** list, and click the **WS-Security** tab.
2. Select **X509 Certificate** token from the **Security Token**  drop down list.

3. Fill in the token details to reference the server's public certificate. Make sure to enter a value in the **Token name** field.
4. Since this token is used for encryption, use `Reference` as the **Referencetype**.
5. Click the **Add Message Encryption** button . In the drop down list, select the token you created in the previous steps.
6. Scroll down to the **XPath** field. Enter an XPath expression, for example:
`// *[local-name(.)='Body']`.


Authenticating with a Username Token and Encrypting with an X.509 Certificate

The following section describes how to send a **Username** token to the service and encrypt it with the server's **X.509** certificate:




1. Select the **Web Service** scenario from the **Service Details** list, and click the **WS-Security** tab.
2. Select **Username Token** from the **Security Token**  drop down list. Provide the token details.
3. Select **X509 Certificate Token** from the **Security Token**  drop down list.
4. Fill the token details to reference the server's public certificate. Make sure to enter a value in the **Token name** field. Since this token is used for encryption, use `Reference` as the **Referencetype**.
5. Click the **Add Message Encryption** button . In the drop down list, select the X.509 token you created in Step 3.
6. To encrypt a specific message, scroll down to the **XPath** field. Enter an XPath expression, for example:
`// *[local-name(.)='Body']`.

Encrypting and Signing a Message

This example shows how to sign a message using a private key and then encrypt it using the service's public key.

1. Select the **Web Service** scenario from the **Service Details** list, and click the **WS-Security** tab.
2. Select **X509 Certificate Token** from the **Security Token**  drop down list. Fill the token details to reference your private certificate. Make sure to

enter a value in the **Token name** field. Select `BinarySecurityToken` as a **Referencetype**, since this token is used for a signature.

3. Select **X509 Certificate Token** from the **Security Token**  drop down list to add another X.509 token. Fill the token details to reference your private certificate. Make sure to enter a value in the **Token name** field. Select `Reference` as a **Referencetype**, since this token is used for a encryption.
4. Click the **Add Message Signature** button . In the drop down list, select the X.509 token you created in Step 2.
5. Click the **Add Message Encryption** button . In the drop down list, select the token you created in Step 3.

How to Customize Security for WCF Type Web Services

This section describes how to customize the security settings for Web services using WCF.

This section describes:

- How do I test a WCF service?
- How do I test a WCF service that uses WSHttpBinding?
- How do I test a WCF service that uses CustomBinding?
- How do I test a WCF service that uses netTcp or namedPipe transport?
- How do I test a Federation scenario that uses an STS (Security Token Service)?
- How do I test a scenario that uses a WSE3 security configuration with a server certificate?
- How do I test a scenario that uses mutual certificate authentication?

How do I test a WCF service?

In the **Service Details** list, select a WCF scenario.

If the Federation or WSHttpBinding scenarios are not appropriate, select the **WCFService (Custom Binding)** scenario, as it can handle all other bindings.

How do I test a WCF service that uses WSHttpBinding?

WSHttpBinding is one of the most popular bindings in WCF. In order to use this binding, select the **WCFService (Http Binding)** scenario from the **Scenario Details** list.

In the Client Authentication box, select the client credential type that you use in your binding—Windows, Certificate, or Username. This value corresponds to the **MessageClientCredentialType** property of the WCF's WSHttpBinding.

Windows authentication is the most common value for a WCF services. If you are using the WCF default settings for your service, use this option. Other options are `username`, `certificate`, or `none`.

For some scenarios you should indicate whether to use the WCF proprietary negotiation mechanism to get the service credentials.

Use the Advanced scenario properties to control the usage of a secure session.

For details, see ["WCF Services \(WSHttpBinding\) Scenario Overview"](#) on page 115.

How do I test a WCF service that uses CustomBinding?

Open the **Security View** in the **Properties** pane and select the **WCFService (Custom Binding)** scenario.

You can then customize many binding elements, such as your transport method, encoding, security, and reliable messaging.

For details, see ["How to Set Security for a WCF Service"](#) on page 124.

How do I test a WCF service that uses netTcp or namedPipe transport?

Select the **WCFService (Custom Binding)** scenario from the **Scenario Details** list.

Configure the transport to **TCP** or **NamedPipe**.

For details, see ["How to Set Security for a WCF Service"](#) on page 124.

How do I test a Federation scenario that uses an STS (Security Token Service)?

For this scenario, you must define the communication properties for both the STS and the service. Use the built-in Federation scenario.

Select the **WCFService (Federation)** scenario from the **Scenario Details** list.

For this scenario, you must to define the communication properties for both the STS and the application server.

For details, see ["WCF Service \(Federation\) Scenario"](#) on page 149.

How do I test a scenario that uses a WSE3 security configuration with a server certificate?

The following procedure describes how to set up a security scenario for WSE3.

1. Create a new test, import the WSDL for the W3ES service, and drag the operation onto the canvas.

2. Open the **Security View** in the **Properties** pane or from the port's shortcut menu. Select the **WCFService (Custom Binding)** scenario.
3. Set the Transport to **HTTP**, and the Encoding to **Text**.
4. Provide a username and password in the **Identities** section.
5. Click the **Browse** button adjacent to the Server Certificate field and specify the **Store Location**, **Store Name** and **Search text** (optional). Click **Find**, select the certificate, and click **Select**.
6. Provide the **Expected DNS**.
7. Click the **Advanced** button and configure the following settings:
 - a. **Encoding** tab— Encoding: Text, WS-Addressing: WSA 04/08 (for example).
 - b. **Security** tab:
 - **Enable secure session:** Enabled
 - **Negotiate service credentials:** Enabled
 - **Protection level:** Encrypt and Sign
 - **Message protection order:** Sign Before Encrypt
 - **Message security version:**
WSecurity11WSTrustFebruary2005WSSecureConversationFebruary2005
(first entry)
 - **Require Derived keys:** Enabled

For all other fields, use the default settings.

For details, see ["How to Set Security for a WCF Service"](#) on page 124.

How do I test a scenario that uses mutual certificate authentication?

The following procedure describes how to set up a security scenario for mutual certificates and how to comply with a WSE3 security configuration.

1. Select the **WCFService (CustomBinding)** scenario from the **Scenario Details** list.
2. Set the **Transport** to `HTTP`, and the **Encoding** to `Text`.
3. Set the authentication mode to `MutualCertificate`.

4. In the **Identities** section, select server and client certificates. For details, see ["Select Certificate Dialog Box "](#) on page 162.
5. Provide the **Expected DNS**.
6. Click the **Advanced** button and configure the following settings:
 - a. **Encoding** tab— **Encoding**: `Text`, **WS-Addressing**: `WSA 04/08` (for a WSE3 security configuration).
 - b. **Security** tab— **Require Derived keys**: `Disabled`

For all other fields, use the default settings.

For details, see ["How to Set Security for a WCF Service"](#) on page 124.

How to Test Web Services that use WS-Security or SSL

This section provides a summary of using Service Test for general security testing.

This section includes:

- How do I test a Web Service that uses SSL?
- How do I test a Web Service that requires Windows authentication at the HTTP level?
- How do I test a Web Service that uses WS-Security?
- How do I configure the low-level details of my WS-Security tokens?

How do I test a Web Service that uses SSL?

Testing a secure site does not require any special configuration. If your service URL begins with **https**, SSL is automatically used.

If in addition to SSL you are using message-level security (for example a username) then you must configure the security for the message separately.

You can use the basic Web Services security scenario and specify the message-level security such as tokens and signatures.

You can also use the **WCFService (Custom Binding)** scenario, or the **WCFService(Http Binding)** scenario with the transport credentials.

How do I test a Web Service that requires Windows authentication at the HTTP level?

1. Select the **Web Service** scenario from the **Scenario Details** list.
2. In the **HTTP** tab, specify the credentials.

For details, see ["How to Set Security for a Standard Web Service"](#) on page 122.

How do I test a Web Service that uses WS-Security?

Use the basic **Web Services** security scenario and open the **WS-Security** tab. Add the message-level security such as tokens, signatures, and encryption.

How do I configure the low-level details of my WS-Security tokens?

In most cases, you can configure the low-level details as described in ["Advanced Settings Dialog Box"](#) on page 154.

How to Set up Advanced Standards Testing

This section provides guidelines for using Service Test in advanced standards testing.

This section includes:

- How do I test a Web Service that uses MTOM?
- How do I change the WS-Addressing version of a service?
- How do I enable support for a service or activity that uses 256-bit SSL encoding?

How do I test a Web Service that uses MTOM?

1. Select the **WCFService (Custom Binding)** scenario from the **Scenario Details** list.
2. Configure the **Encoding** to **MTOM**.

If your service requires advanced settings, click the **Advanced** button. For details, see the "[Advanced Settings Dialog Box](#)" on page 154.

For more information about the scenario, see "[How to Set Security for a WCF Service](#)" on page 124.

How do I change the WS-Addressing version of a service?

1. Select the **Web Service** scenario from the **Scenario Details** list.
2. Click the **WS-Addressing** tab and select a version.

For details, see "[How to Set Security for a Web Service on the Port Level](#)" on page 119.

If your service uses WCF, use the appropriate scenario and configure the addressing version from the Advanced window's **Encoding** tab. For details, see the "[Advanced Settings Dialog Box](#)" on page 154.

How do I enable support for a service or activity that uses 256-bit SSL encoding?

Change the SSL cipher order in Windows Vista so that `AES256` precedes `AES128` in the cipher list.

Tip: Check with an IT professional before performing the following actions.

To change the cipher order:

1. Type **gpedit.msc** at a command prompt to open your group policy editor.
2. Choose **Computer Configuration > Administrative Templates > Network > SSL Configuration Settings**.
3. Open the only item—**SSL Cipher Suite Order**.
4. Select **Enabled**.
5. The first item in the list is `TLS_RSA_WITH_AES_128_CBC_SHA`
And the second item is `TLS_RSA_WITH_AES_256_CBC_SHA`
6. Change the first `128` to `256`. Then move the cursor forward and change the `256` to `128`.
7. Move the cursor through the list and change the cipher priorities as in the above step.
8. Close the group policy editor and reboot.

Reference

This section includes:

- "Security User Interface" on page 138
- "Security Settings for Port / Operation <Name> Dialog Box" on page 139
- "Web Service Scenario " on page 141
- "WCF Service (Custom Binding) Scenario" on page 147
- "WCF Service (Federation) Scenario" on page 149
- "WCF Service (WSHttpBinding) Scenario" on page 151
- "Advanced Settings Dialog Box" on page 154
- "Encoding Tab" on page 155
- "Advanced Standards Tab" on page 156
- "Security Tab" on page 157
- "HTTP and Proxy Tab" on page 160
- "Select Certificate Dialog Box " on page 162

Security User Interface

This section includes:

- "Security Settings for Port / Operation <Name> Dialog Box" on page 139
- "Advanced Settings Dialog Box" on page 154
- "Select Certificate Dialog Box " on page 162

Security Settings for Port / Operation <Name> Dialog Box

Using the Security Settings dialog box, you can configure security settings for all operations in a Web service port. To set the security for a specific step within your test, use the **Security Settings** view in the **Properties** pane.

| | |
|------------------------------|--|
| To access | <p>Do the following:</p> <ol style="list-style-type: none"> 1. Select a component in the Application Component's tree. 2. Select Invoke from the right-click menu. 3. In the Web Service Call window, select a Web Service port in the left pane. 4. In the right pane, click the Security button. 5. In the Security Settings for Operation <operation name> dialog box, click the Edit port's settings hyperlink to modify settings for the whole port, or clear the Use the port's security settings option, to change the security for this particular operation. |
| Important information | <ul style="list-style-type: none"> • For details about choosing a security scenario type, see "Security Scenarios Overview" on page 108. • For examples, see "How to Set up Common Web Services Security Scenarios " on page 125. |
| Relevant tasks | <p>"How to Set Security for a Specific Web Service Operation" on page 121</p> |

User interface elements are described below:

| UI Elements (A-Z) | Description |
|--------------------------|---|
| Advanced | <p>Opens the Advanced Settings dialog box. For details, see the "Advanced Settings Dialog Box" on page 154.</p> <p>Note: Available only for WCF type Web services.</p> |
| Import | <p>Loads security settings from a previously saved .stss file.</p> |

| UI Elements (A-Z) | Description |
|--------------------------|--|
| Save | Saves the security scenario settings to an .stss (Service Test Security Scenario) file, for use in other tests. If you are connected to ALM, it saves the file together with the test. |
| Service Details | The type of Web service. After selecting a type, Service Test provides an interface for modifying the relevant security settings. The service types are: <ul data-bbox="464 636 1383 852" style="list-style-type: none">• "Web Service Scenario " on page 141• "WCF Service (Custom Binding) Scenario" on page 147• "WCF Service (Federation) Scenario" on page 149• "WCF Service (WSHttpBinding) Scenario" on page 151 |

Web Service Scenario

The simple Web Service scenario provides the **HTTP**, **WS-Security**, and **WS-Addressing** tabs.

HTTP tab

The **HTTP** tab lets you provide the HTTP transport level settings such as user credentials for sending a message with basic authentication, proxy settings, message-level settings, encryption, and so forth.

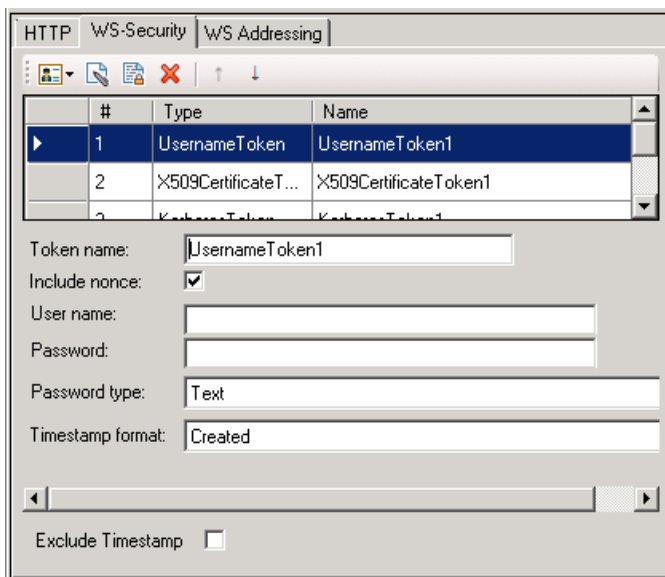
User interface elements are described below.

| UI Elements (A-Z) | Description |
|---------------------------|---|
| Client certificate | The client credentials required for client certificate authentication when using two-way SSL scenarios. The Browse button opens the "Select Certificate Dialog Box " on page 162. |
| Connection timeout | The time threshold in which to connect through the proxy server or with authentication. |
| Keep alive | Keeps the connection persistent. |
| Manage cookies | Enables the writing of cookie information. |






| UI Elements (A-Z) | Description |
|--|---|
| Proxy URL | The URL and port of the proxy server through which the message must pass. For example, <code>http://myProxy:8888/</code> . To use the default, select Use default proxy . |
| Proxy user name, Proxy password | The credentials for the proxy server through which the message must pass. |
| User name, Password | The credentials for HTTP authentication such as basic authentication, digest, or NTLM. For example, User name : <code>myDomain\myUser</code> Password : <code>myPassword</code> |

WS-Security tab

The **WS-Security** tab provides an interface to add message level security using tokens, message signatures, and encryption.



User interface elements are described below. (Unlabeled elements are shown in angle brackets>).

| UI Elements (A-Z) | Description |
|---|---|
|  | Security Tokens. Enables you to add one of the following tokens: User Name , X509 , Kerberos , Kerberos2 , or SAML . |
|  | Add Message Signature. Adds a signature to the message. This requires a token. |
|  | Add Message Encryption. Adds encryption to the message. This requires a token. |
|  | Delete. Removes the security element from the list. |
|  | <p>Up/Down. Positioning tools that allow you to set the priority of the security elements.</p> <p>Important: Make sure the security elements are positioned in order of their priority.</p> |
| <Encryption details pane> | <p>The details of the encryption token.</p> <ul style="list-style-type: none"> • Encrypting token. The token to use for encryption, usually an X.509 type. You can select from a list of all previously created tokens. • Encrypting type. Indicates whether to encrypt the whole destination <code>Element</code> or only its <code>Content</code>. • Key algorithm. The algorithm to use for the encryption of the session key: <code>RSA15</code> or <code>RSASOAEP</code>. • Session algorithm. The algorithm to use for the encryption of the SOAP message. You can select from a list of common values. • XPath (optional). An XPath that indicates the parts of the message to encrypt. If left blank, only the SOAP body is encrypted. • Token (optional). The name of the encrypted token. A drop down box provides a list of all added tokens. With most services, this field should be left empty. |

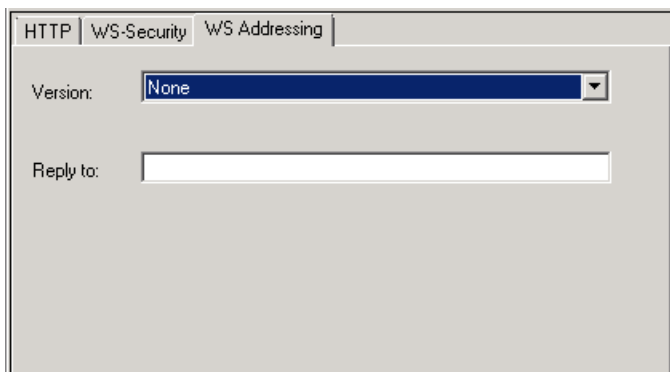
| UI Elements (A-Z) | Description |
|--|---|
| <security element list> | A list of the tokens, message signatures, and encryptions. |
| Exclude Timestamp | Removes the timestamp from the element's SOAP header before sending it to the server. |
| <Signature details pane> | <p>The details of the digital signature used to secure the token.</p> <ul style="list-style-type: none"> • Signing token. The token to use for signing, usually an X.509 type. Select from the list of all added tokens. • Canonicalization algorithm. A URL for the algorithm to use for canonicalization. A drop down list provides common algorithms. If you are unsure which value to use, keep the default. • Transform algorithm. A URL for the Transform algorithm to apply to the message signature. A drop down list provides common algorithms. If you are unsure which value to use, keep the default. • Inclusive namespaces list. A list of comma-separated prefixes to be treated as inclusive (optional). • What to sign. The SOAP elements to sign: SOAP Body, Timestamp, and WS-Addressing. • XPath (optional). An XPath that specifies which parts in the message to sign. If left blank, the elements selected in the Signature options field are signed. For example, <code>//*[local-name()='Body']</code>. • Token (optional). The target token you want to sign. Select from the drop down list of all added tokens. With most services, this field should be left empty. |
| <Token details pane> - Kerberos tokens | <p>Token details for Kerberos tokens:</p> <ul style="list-style-type: none"> • Token Name. A meaningful name for the token. |

| UI Elements (A-Z) | Description |
|---|--|
| | <ul style="list-style-type: none"> • Host. The host name of the server against which you want to authenticate. In most cases, it is the host portion of the service URL. • Domain. The Windows domain of the server against which you want to authenticate. |
| <p><Token details pane> - Username tokens</p> | <p>Token details for Username tokens.</p> <ul style="list-style-type: none"> • Token name. A meaningful name for the token (you can use the default value). • Include nonce. Includes a nonce in the token. • User name, Password • Password type: <code>Text</code>, <code>Hash</code>, or <code>None</code>. • Timestamp format: <code>Full</code>, <code>Created</code>, or <code>None</code>. |
| <p><Token details pane> - X509 Certificate tokens</p> | <p>Token details for X509 Certificate tokens:</p> <ul style="list-style-type: none"> • Token name. A meaningful name for the token. • Certificate. The path of the server certificate file. The Browse button opens the "Select Certificate Dialog Box " on page 162. • Reference type: How the token should be referenced: <code>BinarySecurity Token</code> or <code>Reference</code>. When the certificate is used for encryption, for example, a service certificate, use <code>Reference</code>. When using it for a signature (for example, a certificate with your private key) select <code>BinarySecurity Token</code>. |
| <p><Token details pane> - SAML tokens</p> | <p>Token details for SAML tokens:</p> <ul style="list-style-type: none"> • Load from file. Enables you to browse to a SAML certificate. • Certificate. The path of the certificate file. The Browse button opens the "Select Certificate Dialog Box " on page 162. |

| UI Elements (A-Z) | Description |
|----------------------|---|
| | <ul style="list-style-type: none">• Certificate reference type. How the certificate should be referenced— by X509 Data or RSA. |

WS-Addressing tab

The **WS-Addressing** tab indicates whether WS-Addressing is used by the service, and if so, its version number.



WCF Service (Custom Binding) Scenario

Use this scenario to test WCF services which require security or transport configurations. For general details, see "WCF Service (CustomBinding) Scenario Overview" on page 113.

User interface elements are described below (unlabeled elements are shown in angle brackets>):

| UI Elements | Description |
|--------------------------------|---|
| Advanced | Opens the Advanced Settings dialog box. For details, see "Advanced Settings Dialog Box" on page 154. |
| Client Windows Identity | The identity information for the client windows: <ul style="list-style-type: none"> • Current User. The identity of the user logged onto the machine. • Custom User. A user with the following credentials: Username, Password, and Domain. |
| Encoding | Encoding. Text, MTOM, or WCF Binary |
| Identities | The identity information for the bindings and certificate: <ul style="list-style-type: none"> • Username and Password • Server /Client certificate. A certificate that provides identity information for the server or client. Use the Browse button to open the "Select Certificate Dialog Box " on page 162. • Expected DNS, SPN, and UPN. The expected identity of the server in terms of its DNS, SPN, or UPN. This can be localhost, an IP address, or a server name. |
| Net Security | The type of stream security: None , Windows stream security , or SSL stream security . |
| Reliable Messaging | Enables Reliable Messaging in Ordered or Non-ordered format. |

| UI Elements | Description |
|------------------|---|
| Security | <ul style="list-style-type: none">• Authentication mode. A drop down list of possible modes of authentication, such as <code>AnonymousForCertificate</code>, <code>MutualCertificate</code>, and so forth.• Bootstrap Policy. A drop down list of possible bootstrap policies for <code>Secure Conversation</code> authentication., such as <code>SspiNegotiated</code>, <code>UserNameOverTransport</code>, and so forth. |
| Transport | Transport type. <code>HTTP</code> , <code>HTTPS</code> , <code>TCP</code> , <code>NamedPipe</code> , or <code>AutoSecuredHTTP</code> . |

WCF Service (Federation) Scenario

In the **WCF Service (Federation)** scenario, the client authenticates against the STS (Security Token Service) to obtain a token. The client uses the token to authenticate against the application server.

User interface elements are described below (unlabeled elements are shown in angle brackets>). For details, see "WCF Service (Federation) Scenario Overview " on page 114.

| UI Elements (A-Z) | Description |
|---|--|
| Advanced | Opens the Advanced Settings dialog box. For details, see "Advanced Settings Dialog Box" on page 154. |
| Identities | <p>The identity information for the bindings and certificate:</p> <ul style="list-style-type: none"> • Server certificate. A certificate that provides identity information for the server. Use the Browse button to open the "Select Certificate Dialog Box " on page 162. • Expected DNS. The expected identity of the server in terms of its DNS. This can be <code>localhost</code>, an IP address, or a server name. |
| Security | <ul style="list-style-type: none"> • Authentication mode. A drop down list of possible modes of authentication, such as <code>AnonymousForCertificate</code>, <code>MutualCertificate</code>, and so forth. • Bootstrap Policy. A drop down list of possible bootstrap policies for <code>Secure Conversation</code> authentication., such as <code>SspiNegotiated</code>, <code>UserNameOverTransport</code>, and so forth. |
| Server | <ul style="list-style-type: none"> • Transport. The transport type: <code>HTTP</code> or <code>HTTPS</code>. • Encoding. The server's encoding policy: <code>Text</code> or <code>MTOM</code>. |
| STS (Security Token Service) Details | <p>Information about the STS:</p> <ul style="list-style-type: none"> • Endpoint address. The endpoint address of the STS. This can be <code>localhost</code>, an IP address, or a server name. |

| UI Elements (A-Z) | Description |
|-------------------|--|
| | <ul style="list-style-type: none">• Binding. The scenario which references the binding that contacts the STS. |

WCF Service (WSHttpBinding) Scenario

In the **WCFService(WSHttpBinding)** scenario, you can select from several types of authentication: `None`, `Windows`, `Certificate`, or `Username(message protection)`. For details, see "WCF Services (WSHttpBinding) Scenario Overview " on page 115.

Replace the graphic below, or delete the anchored frame and print-only line after it.

User interface elements are described below (unlabeled elements are shown in angle brackets>) by the client authentication types:

| UI Elements (A-Z) | Description |
|-----------------------------------|---|
| Advanced | Opens the Advanced Settings dialog box. For details, see "Advanced Settings Dialog Box" on page 154. |
| Client authentication type | Authentication type: None , Windows , Certificate , or Username . For details, see below. |

Client Authentication Types:

- Client Authentication Type — None
- Client Authentication Type — Windows
- Client Authentication Type — Certificate
- Client Authentication Type — Username (message protection)

Client Authentication Type — None

| UI Elements (A-Z) | Description |
|----------------------------|--|
| Expected server DNS | The expected identity of the server in terms of its DNS. This can be localhost , an IP address, or a server name. It can also be the common name by which the certificate was issued. |
| Negotiate server | Negotiates the Web Service's certificate with the server. You can also provide the server's DNS information. |

| UI Elements (A-Z) | Description |
|------------------------------------|--|
| credentials | |
| Specify service certificate | <p>The location of the service's certificate. If you select this option, the Negotiate service credentials option is not relevant.</p> <p>For more information, see the "Select Certificate Dialog Box " on page 162.</p> |

Client Authentication Type — Windows

| UI Elements (A-Z) | Description |
|---------------------------------|---|
| Client Windows identity | <p>The identity information for the client windows:</p> <ul style="list-style-type: none"> • Current User. The identity of the user logged onto the machine • Custom User. A user with the following credentials: Username, Password, and Domain |
| Enable secure session | Enables a secure session using <code>Windows type</code> authentication. |
| Expected server identity | The expected server identity method: <code>SPN</code> or <code>UPN</code> . |

Client Authentication Type — Certificate

| UI Elements (A-Z) | Description |
|------------------------------|--|
| Client certificate | The location of the client certificate. The Browse button opens the "Select Certificate Dialog Box " on page 162. |
| Enable secure session | Enables a secure session using <code>Certificate type</code> authentication. |

| UI Elements (A-Z) | Description |
|-------------------------------------|---|
| Expected server DNS | The expected identity of the server in terms of its DNS. This can be <code>localhost</code> , an IP address, or a server name. It can also be the common name by which the certificate was issued. |
| Negotiate server credentials | Negotiates the Web Service's certificate with the server. You can also provide the server's DNS information. |
| Specify service certificate | The location of the service's certificate. If you select this option, the Negotiate server credentials option is disabled. For more information, see the " Select Certificate Dialog Box " on page 162. |

Client Authentication Type — Username (message protection)

| UI Elements (A-Z) | Description |
|-------------------------------------|---|
| Enable secure session | Enables a secure session using <code>Username</code> type authentication. |
| Expected server DNS | The expected identity of the server in terms of its DNS. This can be localhost , an IP address, or a server name. It can also be the common name by which the certificate was issued. |
| Negotiate server credentials | Negotiates the Web Service's certificate with the server. You can also provide the server's DNS information. |
| Specify service certificate | The location of the service's certificate. If you select this option, the Negotiate server credentials option is disabled. For more information, see the " Select Certificate Dialog Box " on page 162. |
| Username, Password | The authentication credentials of the client. |

Advanced Settings Dialog Box

This dialog box enables you to customize the security settings for your test.

| | |
|------------------------------|--|
| To access | Do the following: <ol style="list-style-type: none">1. Select a <code>WCF Service</code> scenario type from the Service Details list.2. Click Advanced |
| Important information | <ul style="list-style-type: none">• For details, see "Advanced Security Settings " on page 117. |
| Relevant tasks | <ul style="list-style-type: none">• "How to Set Security for a Web Service on the Port Level" on page 119• "How to Set Security for a Specific Web Service Operation" on page 121• "How to Set Security for a WCF Service" on page 124 |

This section includes:

- "Encoding Tab" on page 155
- "Advanced Standards Tab" on page 156
- "Security Tab" on page 157
- "HTTP and Proxy Tab" on page 160

Encoding Tab

The Encoding tab lets you indicate the type of encoding to use for the messages: **Text**, **MTOM**, or **WCF Binary**. The default is **Text** encoding.

The user interface elements are described below:

| UI Elements (A-Z) | Description |
|------------------------------|--|
| Encoding | The encoding type to use for the messages: <code>Text</code> , <code>MTOM</code> , or <code>WCF Binary</code> . |
| WS-Addressing version | The version of WS-Addressing for the selected encoding: <code>None</code> , <code>WSA 1.0</code> , or <code>WSA 04/08</code> . |

Advanced Standards Tab

This tab lets you configure advanced WS- standards, such as Reliable Messaging and the Via address option.

The user interface elements are described below:

| UI Elements (A-Z) | Description |
|-----------------------------------|---|
| Reliable messaging | Enables reliable messaging for services that implement the WS-ReliableMessaging specification. The encoding type to use for the messages: <code>Text</code> , <code>MTOM</code> , or <code>WCF Binary</code> . |
| Reliable messaging ordered | Indicates whether the reliable session should be ordered. |
| Reliable messaging version | The version to apply to the messages: <code>WSReliableMessagingFebruary2005</code> or <code>WSReliableMessaging11</code> . |
| Specify via address | <p>Sends a message to an intermediate service that submits it to the actual server. This may also apply when you send the message to a debugging proxy. This corresponds to the WCF clientVia behavior.</p> <p>This is useful to separate the physical address to which the message is actually sent, from the logical address for which the message is intended.</p> |
| Via address | <p>The logical address to which to send the message. It may be the physical of the final server or any name. It appears in the SOAP message as follows:</p> <pre><wsa:Action>http://myLogicalAddress</wsa:Action></pre> <p>The logical address is retrieved from the user interface. By default, it is the address specified in the WSDL. You can override this address using this field.</p> |

Security Tab

The Advanced security settings correspond to the **WS-Security** specifications.

WCF Service (WSHttpBinding) Scenarios

| UI Element (A-Z) | Description |
|--------------------------------------|---|
| Enable secure session | Establish a security context using the WS-SecureConversation standard. |
| Negotiate service credentials | Allow WCF proprietary negotiations to negotiate the service's security. |

WCF Service (CustomBinding) Scenarios

| UI Element (A-Z) | Description |
|--|---|
| Allow serialized signing token on reply | Enables the reply to send a serialized signing token. |
| Default algorithm suite | <p>The algorithm to use for symmetric/asymmetric encryption.</p> <p>The algorithm drop down list gets its values from the SecurityAlgorithmSuite configuration in WCF.</p> <p>Default: Basic256</p> |
| Include timestamp | Includes a timestamp in the header. |
| Key entropy mode | <p>The entropy mode for the security key. The possible values are: Client Entropy, Security Entropy, and Combined Entropy.</p> <p>Default: Combined Entropy</p> |
| Message protection | The order for signing and encrypting. Choose from : |

| UI Element (A-Z) | Description |
|--|---|
| order | <ul style="list-style-type: none"> • Sign Before Encrypt • Sign Before Encrypt-And Encrypt Signature • Encrypt Before Sign |
| Message security version | The WS-Security security version. You can also indicate whether to Require derived keys for the message. |
| Protection level | <p>Indicates whether the SOAP Body be encrypted/signed. The possible values are: None, Sign, and Encrypt And Sign (default)</p> <p>Default: Encrypt And Sign</p> |
| Require security context cancellation | Indicates whether to require the cancellation of the security context. If you disable this option, stateful security tokens will be used in the WS-SecureConversation session, if they are enabled. |
| Require signature confirmation | Instructs the server to send a signature confirmation in the response. |
| Security header layout | The layout for the message header: Strict, Lax, Lax Timestamp First, or Lax Timestamp Last. |
| X509 Inclusion Mode | <p>When to include the X.509 certificate:</p> <ul style="list-style-type: none"> • Always to Recipient • Never • Once • Always To Initiator <p>Note: This and the next three options only apply when using an X.509 certificate.</p> |
| X509 key identifier clause type | The type of clause used to identify the X.509 key. |

| UI Element (A-Z) | Description |
|----------------------------------|---|
| | <ul style="list-style-type: none">• Any• Thumbprint• Issuer Serial• Subject Key Identifier• Raw Data Key Identifier |
| X509 Reference Style | How to reference the certificate: <ul style="list-style-type: none">• Internal• External |
| X509 require derived keys | Indicates whether X.509 certificates should require derived keys. |

HTTP and Proxy Tab

This tab lets you set the HTTP and Proxy information for your test.

HTTP (S) Transport

The following table describes the HTTP(S) Transport options:

| UI Element (A-Z) | Description |
|------------------------------------|--|
| Allow cookies | Indicates whether to enable or disable cookies. |
| Authentication scheme | The HTTP authentication method: <code>None</code> , <code>Digest</code> , <code>Negotiate</code> , <code>NTLM</code> , <code>Integrated Windows Authentication</code> , <code>Basic</code> , or <code>Anonymous</code> . |
| Bypass proxy on local | Indicates whether to ignore the proxy when the service is on the local machine. |
| Keep-Alive enabled | Indicates whether to enable or disable keep-alive connections. |
| Max response size (KB) | The maximum size of the response before being concatenated. Default: 65 KB |
| Proxy address | The URL of the proxy server. |
| Proxy authentication scheme | HTTP authentication method on Proxy: <code>Digest</code> , <code>Negotiate</code> , <code>NTLM</code> , <code>Basic</code> , or <code>Anonymous</code> . |
| Realm | The realm of the authentication scheme in the form of a URL. |
| Require client certificate | Indicates whether to require a certificate for SSL transport. |
| Transfer mode | The transfer method for requests/responses. The possible values are <code>Buffered</code> , <code>Streamed</code> , <code>Streamed Request</code> , and <code>Streamed Response</code> . |
| Use default | Indicates whether to use machine's default proxy settings. |

| UI Element (A-Z) | Description |
|------------------|-------------|
| web proxy | |

Select Certificate Dialog Box


This dialog box enables you to search and locate a certificate from a file or Windows store.

| | |
|-----------------------|--|
| To access | <p>Do the following:</p> <ol style="list-style-type: none"> 1. Select a WCF Service scenario type from the Service Details list. 2. Click the Browse button adjacent to the ServerCertificate box. |
| Relevant tasks | "How to Set Security for a WCF Service" on page 124. |

Select Certificate from File

When you select **Import from: File**, the dialog box shows the relevant user elements.

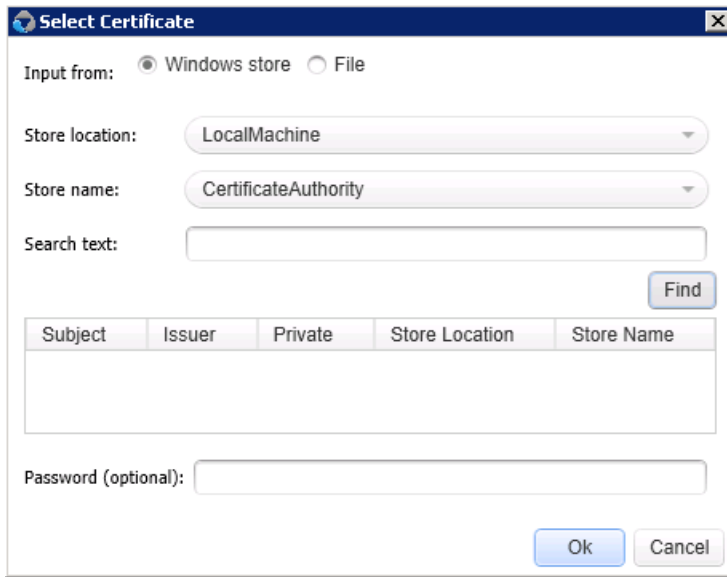
The user interface elements are described below:

| UI Elements (A-Z) | Description |
|---|--|
|  | Browse. Enables you to locate the certificate file with a .cer or .pfx extension. |
| File | The complete path of the certificate file. |
| Import from | The source of the certificate file: Windows store or File . |
| Password (optional) | The password required to access the certificate. |

Select Certificate from Windows Store

When you select **Import from a Windows Store**, the dialog box shows the Windows Store related user elements:

View Image



The user interface elements are described below:

| UI Elements (A-Z) | Description |
|----------------------------|---|
| <certificate list> | A list of the certificates in the Windows store sorted by Subject, Issuer, Private, Store Location, and Store Name . |
| Find | Begins the search for the certificate based on the Search text. |
| Import from | The source of the certificate file: Windows store or File . |
| Password (optional) | The password required to access the certificate. |
| Search text | The text to match in the certificate name. If left blank, the Find action retrieves all available certificates. |
| Store location | The store location, for example Current User . |
| Store name | The store name, for example, AuthRoot . |

Troubleshooting and Limitations - Web Service Security

This section describes troubleshooting and limitations for working with Web services security.

- Authentication and proxy security are not supported for Web Services imported from a UDDI.
- For Web Service configured with WCF settings: Configuring different security settings for operations residing on the same port is not supported.
- For Web Service configured with WCF settings, user event handlers cannot be used.
- When testing Web Services that require message-level security, the Web Service security scenario only supports SOAP version 1.1. For SOAP 1.2 use a WCF type scenario.
- When using a SAML security token for Web services security, user-provided content may contain creation and expiration timestamps. To extend the life of the test, we recommend that you hard-code an expiration date in the distant future. In this is not possible, change the timestamp by implementing the **OnBeforeApplyProtocolSettings** event.
- When using a SAML security token for Web services security, if you edit the values in Grid mode, they may not be updated in Service Test.
Workaround: To update the values, switch to **Text** mode and save the test.
- Web Service steps are not supported when using a SAML token with a certificate from the file system.
Workaround: Install the certificate to the Windows store and select the certificate from the store.
- The **AfterProcessRequestSecurity** and the **BeforeProcessResponseSecurity** events are not supported for Web Service activities configured with WCF type scenarios.
- When working with Federation type scenarios that use STS (Security Token Service), you cannot change the SOAP version.
- The **OnSendRequest** and the **OnReceiveResponse** events are not supported for Web Service activities configured with WCF type scenarios.

Groups

Concepts

- "Groups Overview" on page 167

Tasks

- "How to Manage Component Groups " on page 170

Reference

- "New Groups User Interface " on page 173
- "New Group Dialog Box" on page 174

Concepts

This section includes:

- ["Groups Overview" on page 167](#)

Groups Overview

In an organization, application components have a context. For example, the components may be located on the same server, may be exposed as part of the same application, or may take part in the same business process.

In the Application Components module, you use groups to organize your components into logical groupings. You might, for example, organize them into groups representing different physical deployments, business processes, or application types such as Enterprise Resource Planning (ERP) or Customer Relationship Management (CRM). After you create the groups, you can associate components to them, and perform tasks upon all members of the group.

You can add attachments to a group and send e-mails in a similar manner to that of individual components. For details, see the *HPHP Application Lifecycle Management User Guide*.

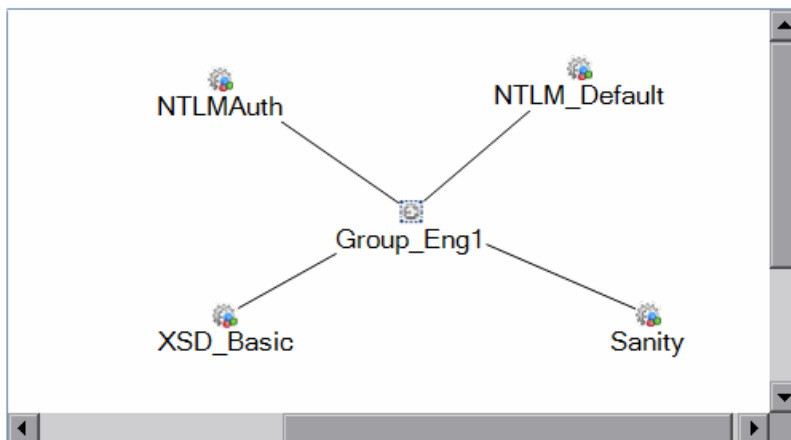
This section also includes:

- Associating Components to Groups
- Group Coverage

Associating Components to Groups

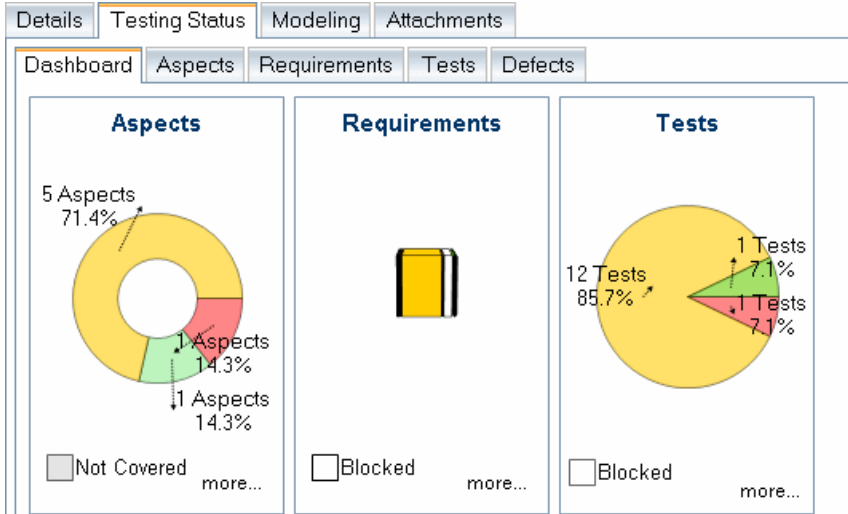
Groups help you organize your components into logical units. You might, for example, organize them into groups representing different physical deployments, business processes, or application types. After you create groups, you can associate components to the group, and perform tasks upon the entire group. The same component can be assigned to multiple groups.

The **Modeling** tab lets you create and view the associations between the groups and components. For details, see the "[Modeling Tab](#)" on page 39.



Group Coverage

You define requirements and tests for individual components— not for groups. When you associate components with groups, the group reflects the coverage of all its associated entities. You can view the complete coverage in the group's **Testing Status** tab. Click any of the sections in the **Testing Status** tab, to see the specific coverage for **Aspects**, **Requirements**, or **Tests**.



For details, see "Determining Test Coverage" on page 187

Tasks

This section includes:

- "How to Manage Component Groups " on page 170

How to Manage Component Groups

This task describes how to create and manage component groups. It also describes how to associate components to a group and how to check the testing coverage for the entire group.



This task includes the following steps:

- Create an application component group
- Add a component to a group
- View and Modify Group Details - optional
- Mail Group Information - Optional
- Results and Coverage

Create an application component group

Create a group at the root level of the component tree or under a specific folder. Service Test Management displays the groups in the component tree hierarchy. For details, see ["New Group Dialog Box" on page 174](#).

Add a component to a group

1. Select a group in the Application Components tree.
2. Click the **Modeling** tab.
3. Click **Select** in the **Modeling** tab to open the application component folders in the right pane.
4. Expand the tree and select the application component you want to add to the group.
5. Click **Add > Associate**  to add the component to the group.
6. Drag the entities in the **Modeling** tab into the desired positions, or click the **Arrange Layout** button  to do this automatically.

Tip: If you cannot see the group layout, try refreshing the window or expanding it with the splitter bar.

For more details, see the ["Modeling Tab" on page 39](#).

View and Modify Group Details - optional

Select a group in the Application Components tree and click the **Details** tab.

For further information, see the ["Details " on page 30](#).

Mail Group Information - Optional

Send group details by e-mail to other users. This enables you to routinely inform users about the status of a group and its components. The e-mail message includes a link that enables the recipient to go directly to the group.

For more information about sending e-mails directly from ALM, see the *HP Application Lifecycle Management User Guide*.

Results and Coverage

Click the **Testing Status** tab to view the complete coverage for **Aspects**, **Requirements**, or **Tests**.

For details, see "[Determining Test Coverage](#)" on page 187

Reference

This section includes:

- "New Groups User Interface " on page 173
- "New Group Dialog Box" on page 174

New Groups User Interface

This section includes:


- ["New Group Dialog Box" on page 174](#)



New Group Dialog Box

This dialog box enables you to create a new application component group.

| | |
|-----------------------|---|
| To access | Use one of the following: <ul style="list-style-type: none"> • Select Components > New > New Group. • Select New > New Group from the shortcut menu. |
| Relevant tasks | "How to Manage Component Groups " on page 170 |

The following elements are included:

| UI Elements (A-Z) | Description |
|---|--|
|  | Clear All Fields. Clears all the fields in the dialog box so that you can redefine the group. |

| UI Elements (A-Z) | Description |
|---|--|
|  | Details module. Enables access to the Details and Description tabs. |
|  | Attachments module. Allows you to attach a file to the group. For details, see the <i>HP Application Lifecycle Management User Guide</i> . |
| Close | Closes the dialog box and cancels any changes you made to the existing screen. |
| Description tab | A description of the group's objective. |
| Details tab | Lets you set the following fields: <ul style="list-style-type: none"> • Assigned To • Created By • Creation Date |
| Name | The name of the group. This field is required. |
| Submit | Creates the group whose details you specified. |

Generating Requirements and Tests

Concepts

- "Generating Requirements and Tests Overview " on page 178

Tasks

- "How to Generate Requirements and Tests" on page 183

Reference

- "Generate Requirements and Tests Wizard" on page 185
- "Troubleshooting and Limitations - Generate Requirements" on page 186

Concepts

This section includes:

- ["Generating Requirements and Tests Overview "](#) on page 178

Generating Requirements and Tests Overview

To test your application, you can manually create requirements and tests, or use the Requirement and Test Generation wizard to automatically generate requirements and test scripts.

The Requirement and Test Generation wizard guides you through the process of creating requirements and scripts to test your application components. Through the wizard, you indicate which aspects of the component you want to test. These aspects include interoperability with different toolkits, boundary testing, and standard compliance.

After you select the testing aspects, Service Test Management generates a set of requirements, or requirements and tests linked to the component or to components in the group. Only requirements are linked to components. The tests are indirectly linked to components through requirements. These requirements and tests provide pre-packaged testing methodologies and component information that specify which components of the application need to be tested, and what to test in each part.

Generated requirements provide a textual description of the testable feature and the expected behavior, based on a template of the specified testing aspects adapted to the selected components. Generated tests describe the test objective and provide an implementation of that testing objective. They include the test operations and arguments for specific testing types. After generating requirements and tests, you should customize them to your particular requirements.

You can also link components to manually created requirements, and indirectly to tests through requirements. For details on assigning components to requirements and tests, see ["Linking Components and Requirements "](#) on [page 220](#).

This section also includes:

- Testing Aspects
- Actions after Creating a Test

Testing Aspects

The Requirement and Tests Generation wizard helps you create requirements and tests that verify different aspects of your component. Service Test

Management creates a separate requirement for each aspect and child aspect, and a separate test for each child aspect.

The following table lists the built-in testing aspects:

| Aspect Name | Description |
|---------------------------------|---|
| Positive Testing | <p>Generates a full positive test that checks each operation of the component.</p> <p>The wizard creates a separate Positive Testing test for each of the service's operations.</p> |
| Standard Compliance | <p>Checks the component's compliancy with industry standards such as WS-I and SOAP.</p> |
| Service Interoperability | <p>Tests the interoperability of the component's operations with all supported technological platforms.</p> <p>Tests that the component is fully interoperable with all supported technological platforms, by calling all of its operations with default or expected values.</p> <p>For Web Services, it contains the following sub-aspects:</p> <ul style="list-style-type: none">• .NET Framework. Tests that the service is fully interoperable with .NET Framework WSE 2 toolkit.• Axis/Java Based Web Services. Tests that the service is fully interoperable with Axis 1.3 Web Services. |
| Security Testing | <p>Tests security. Contains the following sub-aspects:</p> <ul style="list-style-type: none">• SQL Injection Vulnerability. Checks if the component is vulnerable to SQL injections by injecting SQL statements and errors into relevant parameters.• Cross-site Scripting (XSS). Attempts to hack the component by injecting code with relevant parameters that will disrupt its functionality. |
| Boundary Testing | <p>Using the negative testing technique, creates tests to manipulate data, types, parameters, and the actual SOAP message to test the component to its limits.</p> <p>Contains the following sub-aspects:</p> |

| Aspect Name | Description |
|----------------------------|---|
| | <ul style="list-style-type: none">• Extreme Values. Provides invalid data types to the components and verifies they are not accepted.• Null Values. Provides NULL parameters to the components to verify they are not accepted. |
| Performance Testing | Contains the following sub-aspects: <ul style="list-style-type: none">• Stress Testing. Tests the maximum load that can be placed on the application.• Overload Sustainability Testing. Tests how well the hardware allocated for the application can support the number of anticipated users.• Volume Testing. Tests that the system can handle a massive data entry.• Longevity Test. Tests that the system can sustain a consistent number of concurrent users and executing transactions using near-peak capacity, over a minimum 24-hour period.• Scalability Testing. Repeated stress, overload, volume, and longevity tests with different server or network hardware configurations. |

You can also define custom aspects. For details, see the Customization's "Aspects Tab " on page 255.

Actions after Creating a Test

After generating requirements or requirements and tests, you can do the following:

| Action | Description |
|--|--|
| Edit the generated requirements | You can modify the details, attachments, component coverage, test coverage, and defect links for any requirement to suit your specific testing objectives. For example, you might want to define customized thresholds that are not reflected in your component definitions. For details about defining requirements, refer to the HP Application Lifecycle Management User Guide. |
| Edit the generated tests | You can edit the generated scripts to expand their scope and purpose. You can edit manual tests, scripts designed with Service Test, or those created by a third-party vendor. |
| Configure parameters for test instances | You can configure parameters for test instances in the Test Lab module. For details, refer to the HP Application Lifecycle Management User Guide. |
| Manually set requirements | You can manually set requirements for a component. For details on linking components to requirements, see " Linking Components and Requirements " on page 220. |
| Manually add test coverage | You can manually add test coverage for a component. For details, see " Linking Components and Requirements " on page 220. |
| Run the test | Run the test from ALM's Test Lab module. For details, see the HP Application Lifecycle Management User Guide. |

Tasks

This section includes:

- ["How to Generate Requirements and Tests" on page 183](#)

How to Generate Requirements and Tests

This task describes how to create requirements or tests for checking the quality of your component.

This task includes the following steps:

- Prerequisites
- Run the wizard - optional
- Review the generated tests
- Remove unwanted tests or requirements - optional

1. Prerequisites

Make sure you have at least one component in the Application Components tree. If not, use the **Components** menu to create one. Select the component for which you want to create requirements or tests.


2. Run the wizard - optional

To create aspect-based tests using a wizard, select **Components > Generate Requirements/Tests**. For details, see the "[Generate Requirements and Tests Wizard](#)" on page 185.

3. Review the generated tests

After the generation process, you can review the generated entities in the **Testing Status** tab. This tab also provides full coverage information for each requirement and test. For details, see "[Determining Test Coverage](#)" on page 187.

4. Remove unwanted tests or requirements - optional

To remove a test or requirement, select it in the **Testing Status** tab and click the **Delete** button . For details, see "[Determining Test Coverage](#)" on page 187.

Reference

This section includes:

- "Generate Requirements and Tests Wizard" on page 185

Generate Requirements and Tests Wizard

This wizard enables you to create requirements and/or tests for your component.

| | |
|-----------------------|---|
| To access | Select Components > Generate Requirements/Tests. |
| Relevant tasks | "How to Generate Requirements and Tests" on page 183 |
| Wizard map | This wizard contains: Select Entities to Generate Page > Select Testing Aspects Page > Select Locations Page > Summary Page > Generate Page |
| See also | <ul style="list-style-type: none">• "How to Check the Testing Status " on page 199• "How to Check the Testing Status " on page 199 |

Troubleshooting and Limitations - Generate Requirements

This section describes the limitations that apply to the Generate Requirements and Tests wizard.

- The **Interoperability** testing aspects are not supported when generating a test with the Requirement and Test Generation wizard and choosing the **ST11.50 Generator**. This includes the **.NET Framework** and **Axis/Java Based Web Services** testing aspects.
- When using the **Generate Requirements and Tests** wizard for a application component type other than Web Services, the **Select script generator** drop down continues to show the Service Test generator option. This option is not supported for non-Web Service components, and it may generate an error.
- When using the **Generate Requirements and Tests** wizard, you cannot select the root folder, **Subject**, as a location for the tests.

Determining Test Coverage

Concepts

- "Coverage Overview " on page 189
- "Rule Violations" on page 190
- "Aspect Coverage" on page 191
- "Requirements Coverage" on page 192
- "Operations Coverage" on page 194
- "Tests Coverage" on page 195
- "Defects Coverage" on page 197

Tasks

- "How to Check the Testing Status " on page 199

Reference

- "Testing Status Tab " on page 203
- "Testing Status Dashboard " on page 204
- "Rules Tab" on page 206
- "Aspects Tab " on page 208
- "Requirements Tab" on page 210
- "Operations Tab " on page 213
- "Tests Tab" on page 215
- "Defects Tab" on page 218

Concepts

This section includes:

- "Coverage Overview " on page 189
- "Rule Violations" on page 190
- "Aspect Coverage" on page 191
- "Requirements Coverage" on page 192
- "Operations Coverage" on page 194
- "Tests Coverage" on page 195
- "Defects Coverage" on page 197

Coverage Overview

You analyze the quality of your application components using charts that display the status of the testing elements: aspects, requirements, tests, operations, and defects. You can analyze either a specific component or all the components in a group or folder.

The **Testing Status** tab shows the coverage of the requirements and tests as they were defined for the components.

This section also includes:

- ["Rule Violations" on page 190](#)
- ["Aspect Coverage" on page 191](#)
- ["Requirements Coverage" on page 192](#)
- ["Operations Coverage" on page 194](#)
- ["Tests Coverage" on page 195](#)
- ["Defects Coverage" on page 197](#)

Rule Violations

The **Testing Status > Rules** tab shows you the rules that were violated by the component.

The following table describes the rule violations.

| Requirement Status | Description |
|-------------------------|-----------------------------|
| Rule name | The name given to the rule. |
| Rule description | A description of the rule. |

For task details, see ["How to Check the Testing Status "](#) on page 199.

Aspect Coverage

The Testing Status **Aspects** tab lets you see the coverage of the testing aspects defined for the component testing. This information is dynamic and displays the current testing status in percentages. It is useful for the business analyst or system architect who wants to see the coverage in meaningful terms.

The following table describes the aspect statuses.

| Aspect Status (A-Z) | Description |
|-----------------------------|---|
| All Failed | All the aspect's requirements are assigned the Failed status. |
| All Passed | All the aspect's requirements are assigned the Passed status. |
| No Run/Not Completed | All the aspect's requirements are assigned either the No Run, Not Completed or Not Covered status. |
| Not Covered | All the aspect's requirements are assigned the Not Covered status. |
| Some Failed | At least one of the aspect's requirements is assigned the Failed status. |
| Some Passed | At least one of the aspect's requirements is assigned the Passed status. All other requirements have either the No Run, Not Completed , or Not Covered status. |

Requirements Coverage

Requirements coverage is used to connect components to other Application Lifecycle Management entities (tests, test sets, and defects) through the requirement. This helps you keep track of the relationship between your requirements and components, and ensures compliance with your requirements throughout the testing process. After you have created tests, you associate assigned requirements with tests and defects. In this way, you can keep track of your testing needs at all stages of the testing process. If a requirement changes, you can immediately identify which tests and defects are affected, and who is responsible.

To test application components in Service Test Management, you must link them to requirements. When you define a component, you can link it to an existing requirement or create new requirements.

To use existing requirements, you drag them from the Requirement entities tree to your component.

To create new requirements, use the Requirement and Test Generation wizard as described in "[Generating Requirements and Tests](#)" on page 176.

The **Testing Status > Requirements** tab lets you see the coverage of the requirements that were linked to the component. This information is dynamic and displays the current status in percentages. It is most useful for advanced users who understand the ALM coverage model and want to integrate application components into that model.

The following table describes the requirement statuses.

| Requirement Status (A-Z) | Description |
|--------------------------|---|
| Failed | At least one test associated with the requirement has failed. |
| N/A | The terms Passed and Failed do not apply to the requirement. |
| No Run | All tests associated with the requirement have a No Run status. |
| Not Completed | All tests associated with the requirement have the Not Completed status. |

| Requirement Status (A-Z) | Description |
|--------------------------|--|
| Not Covered | There are no tests associated with the requirement. |
| Passed | All tests associated with the requirement have passed. |

To manually link requirements to components, see "[Linking Components and Requirements](#) " on page 220

Operations Coverage

The **Testing Status > Operations** tab lets you see the testing status per operation in chart form. This is useful when the component is less granular and its operations may be considered as separate use cases for quality assurance.

When you expand an operation, you can see the status of the tests associated with the operation. In this context, association indicates that:

- the test is linked to a requirement which is linked to this component
- the test uses this operation.

The following table lists the statuses:

| Aspect Status (A-Z) | Description |
|-----------------------------|---|
| All Failed | All of the component's operations had a Failed status for their tests. |
| All Passed | All of the component's operations had a Passed status for their tests. |
| No Run/Not Completed | All of the component's operations had the status of No Run, Not Completed or Not Covered for their tests. |
| Not Covered | All of the component's operations had the status of Not Covered for their tests. |
| Some Failed | At least one of the component's operations had a Failed status for their tests. |
| Some Passed | Some of the component's operations had a Passed status for their tests. All other's had either the No Run, Not Completed , or Not Covered status. |

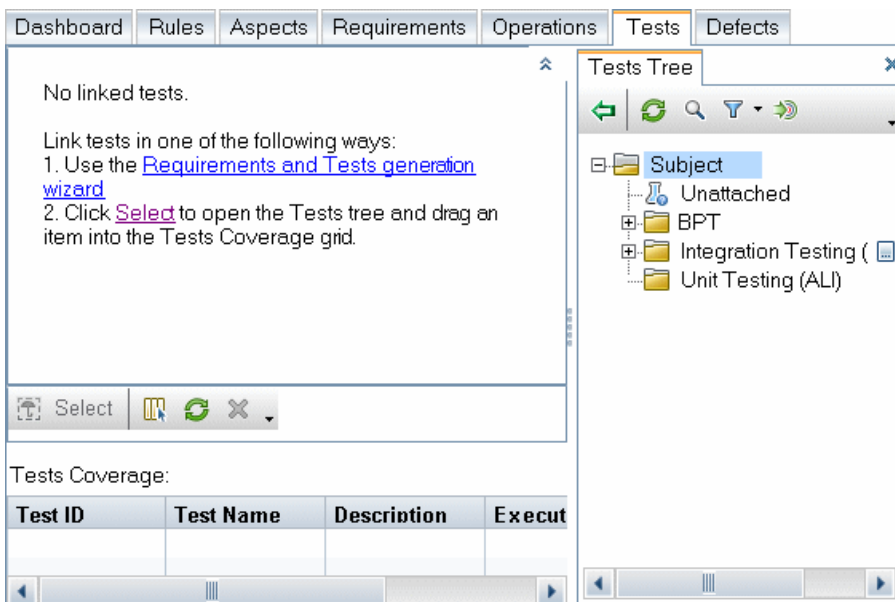
Tests Coverage

The **Testing Status > Tests** tab lets you see the coverage of the tests that were linked to the component. This information is dynamic and displays the current status in percentages.

The following table lists the possible statuses:

| Status (A-Z) | Description |
|----------------------|--|
| Blocked | The percentage of tests that were blocked from being run. |
| Failed | The percentage of tests that returned a Failed status. |
| N/A | The percentage of tests to which the terms Passed and Failed do not apply. |
| No Run | The percentage of tests that were not run. |
| Not Completed | The percentage of tests that were not completed. |
| Passed | The percentage of tests that returned a Passed status. |

If there are no tests linked to the application component, Service Test Management provides links to the **Requirements and Test Generation wizard** or to the **Tests Tree** tab.



Alternatively, you can link components to tests through the **Test Plan** module.

Note: You can create automatic coverage between test instances and components using the **ALLOW_REQ_COVERAGE_BY_TEST_INSTANCE** parameter in Site Administration. For details about this parameter, refer to the Application Lifecycle Management Administrators Guide.

Defects Coverage

The **Testing Status > Defects** tab lets you see the coverage of the defects that were linked to the component. This information is dynamic and is displayed in a chart form, based on the severity of the defects.

You do not link a defect directly to a component. Instead, you link a defect to a requirement or test. If the component was assigned a test that is linked to a defect, Service Test Management displays the defect status in this tab.

For details, see the **Test Plan** module documentation in the *HP Application Lifecycle Management User Guide*.

Tasks

This section includes:

- ["How to Check the Testing Status "](#) on page 199

How to Check the Testing Status

The following steps describe how to check the testing status of your component. You check the testing status by viewing the coverage of the aspects, requirements, operations, tests, and defects associated with your component.

This task includes the following steps:

- View the Aspects coverage - optional
- View the Requirements coverage - optional
- View the Operations coverage - optional
- View the Tests coverage - optional
- View the Rule Violations - optional
- View the Defects coverage - optional

To check the testing status, you do not need to check all of the coverage types. The coverages you need to check depends on how you designed your test plan and your area of focus. In one instance you may be interested solely in the testing aspects, while in another scenario, the test requirements may be the most relevant.

View the Aspects coverage - optional

1. Select a component, group, or folder in the tree hierarchy.
2. Select the **Testing Status** tab and click its **Aspects** button. Service Test Management shows a graphic representation of the coverage in the top pane, and a list of the required aspects in the bottom pane. The graph only shows the aspect coverage—not the requirements. For user interface details, see the "[Aspects Tab](#) " on page 208.
3. Expand an aspect to see its associated requirements. Click on a requirement to open the item in the **Requirements** module. For details, see the *HP Application Lifecycle Management User Guide*.

View the Requirements coverage - optional

1. Select a component, group, or folder and click the **Testing Status > Requirements** tab.
2. Expand a requirement to see its associated tests. Click on a test to open it

in the **Test Plan** module. In the expanded grid, you can drag a test from one parent requirement to another.

3. If there are no requirements associated with this component, use the ["Generate Requirements and Tests Wizard" on page 185](#) to create them. To add existing requirements to the coverage, click **Select** to open the **Requirements Tree** tab in the right pane, and drag an item into the Requirements Coverage grid.
4. Double-click on a requirement in the grid to open it in the **Requirements** module. To return to your last view, click on **Application Components** in the left pane.


For details about the Requirements tab, see ["Requirements Tab" on page 210](#).

View the Operations coverage - optional

1. Select a component and click the **Testing Status > Operations** tab.
2. Expand an operation to see the status of the tests associated with the operation.
3. To view the test details, click on a test in the grid to open it in the **Test Plan** module. For details about the **Requirements** and **Test Plan** modules, see the *HP Application Lifecycle Management User Guide*.
4. To return to your last view, click the **Application Components** module button in the left pane.

For user interface details, see the ["Operations Tab " on page 213](#).

View the Tests coverage - optional

1. Select a component, group, or folder and click the **Testing Status > Tests** tab.
2. To add more tests to a component's coverage, click **Select**. Expand the **Tests Tree** in the right pane and select the test you want to add to the coverage. Use the filter button to show only the desired tests.
3. Drag the test to the **Tests Coverage** grid, or click the **Add to Coverage** button .
4. Click on a test in the **Tests Coverage** grid to open it in the **Test Plan** module. To return to your last view, click **Application Components** in the left pane.

Tip: In the grid, use the right-click menu to create new tests or navigate to an existing test.

For user interface details, see the ["Tests Tab" on page 215](#).

View the Rule Violations - optional

1. Select a component in the Application Components tree.
2. Select the **Testing Status > Rules** tab. Service Test Management shows a list of the rule violations for the selected component.
3. To resolve the violations, click **Resolve All** to open the ["Generate Requirements and Tests Wizard" on page 185](#).

For user interface details, see the ["Rules Tab" on page 206](#).

View the Defects coverage - optional

1. Select a component or group in the tree hierarchy.
2. Select the **Testing Status > Defects** tab. Service Test Management shows a graphic representation of the coverage in the top pane, and a list of the defects in the bottom pane.
3. Click on a defect to view its details in the **Defects** module. To return to your last view, click the **Application Components** module in the left pane.
4. To refresh the chart, click the **Refresh** button.

For user interface details, see the ["Defects Tab" on page 218](#).

For more information about the **Requirements**, **Test Plan**, and **Defects** modules, see the *HP Application Lifecycle Management User Guide*.

Reference

This section includes:

- "Testing Status Tab " on page 203
- "Testing Status Dashboard " on page 204
- "Rules Tab" on page 206
- "Aspects Tab " on page 208
- "Requirements Tab" on page 210
- "Operations Tab " on page 213
- "Tests Tab" on page 215
- "Defects Tab" on page 218

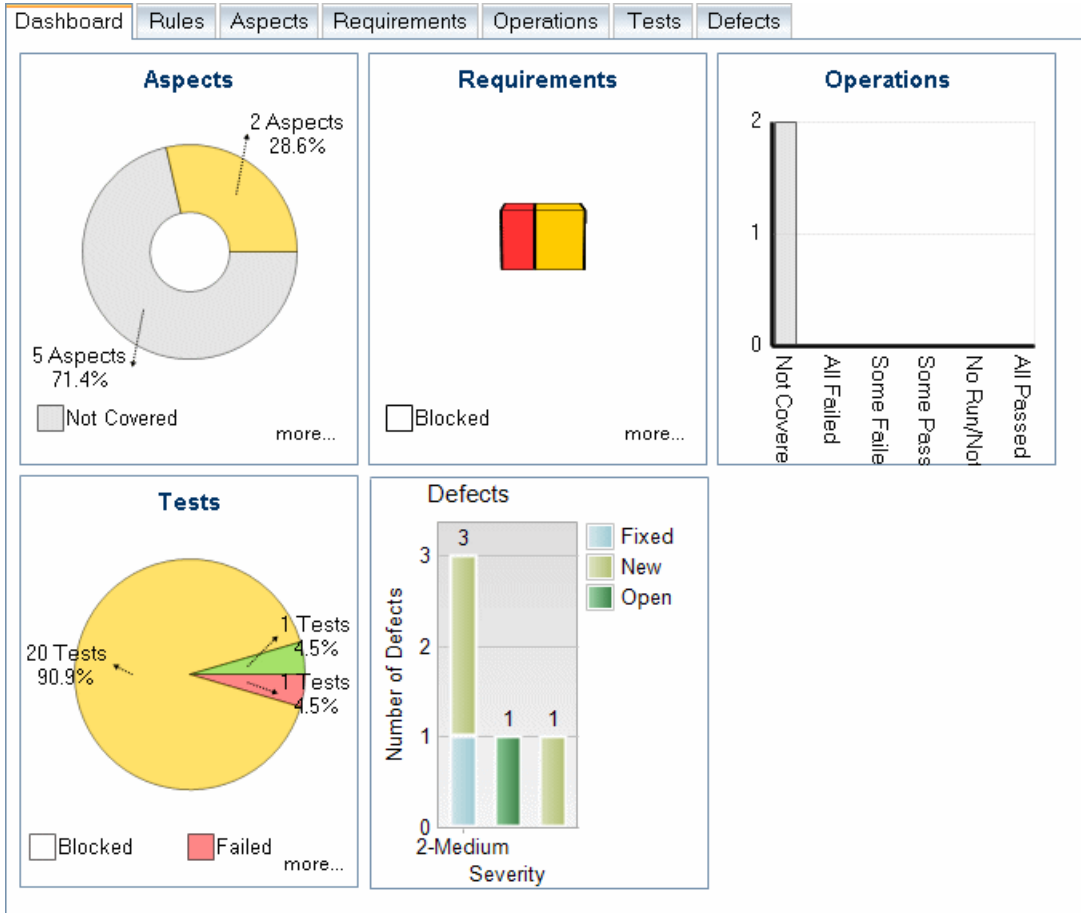
Testing Status Tab

This section describes the interface of the Testing Status sub tabs.

- "Testing Status Dashboard " on page 204
- "Rules Tab" on page 206
- "Aspects Tab " on page 208
- "Requirements Tab" on page 210
- "Operations Tab " on page 213
- "Tests Tab" on page 215
- "Defects Tab" on page 218

Testing Status Dashboard

The Testing Status dashboard displays a coverage summary of each of the following items: **Aspects**, **Requirements**, **Operations**, **Tests**, and **Defects**.



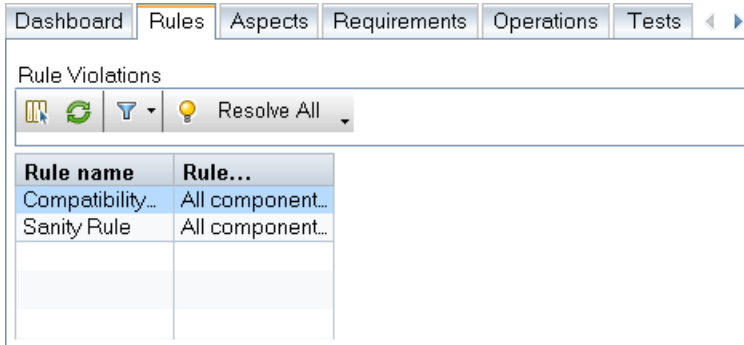
| | |
|-------------------------------------|---|
| <p>To access</p> | <p>Do the following:</p> <ul style="list-style-type: none"> • Select the Application Components module. • Select an entity in the Application Components tree. • Select the Testing Status > Dashboard tab. |
| <p>Important information</p> | <p>Click on a graph to open it in a separate tab.</p> |
| <p>Relevant tasks</p> | <p>"How to Check the Testing Status " on page 199</p> |

The graphs are described below. Click on a graph to open it in a new tab.

| UI Elements (A-Z) | Description |
|------------------------------|---|
| Aspects | A graphical representation of the coverage by the testing aspects. |
| Defects | A graphical representation of the coverage for the defects associated with the component. |
| Operations | A graphical representation of the coverage by the component's operations. |
| Requirements | A graphical representation of the status of the requirements associated with the component. |
| Tests | A graphical representation of the status of the tests associated with the component. |




Rules Tab

This tab enables you to view and manage the rules violated by the selected component.



| | |
|------------------------------|---|
| To access | <p>Do the following:</p> <ul style="list-style-type: none"> • Select the Application Components module. • Select an application component. • Select the Testing Status > Rules tab. |
| Important information | <p>See "Rule Violations" on page 190.</p> <p>Note: Available only for Application Component type entities.</p> |
| Relevant tasks | <p>"How to Check the Testing Status " on page 199</p> |

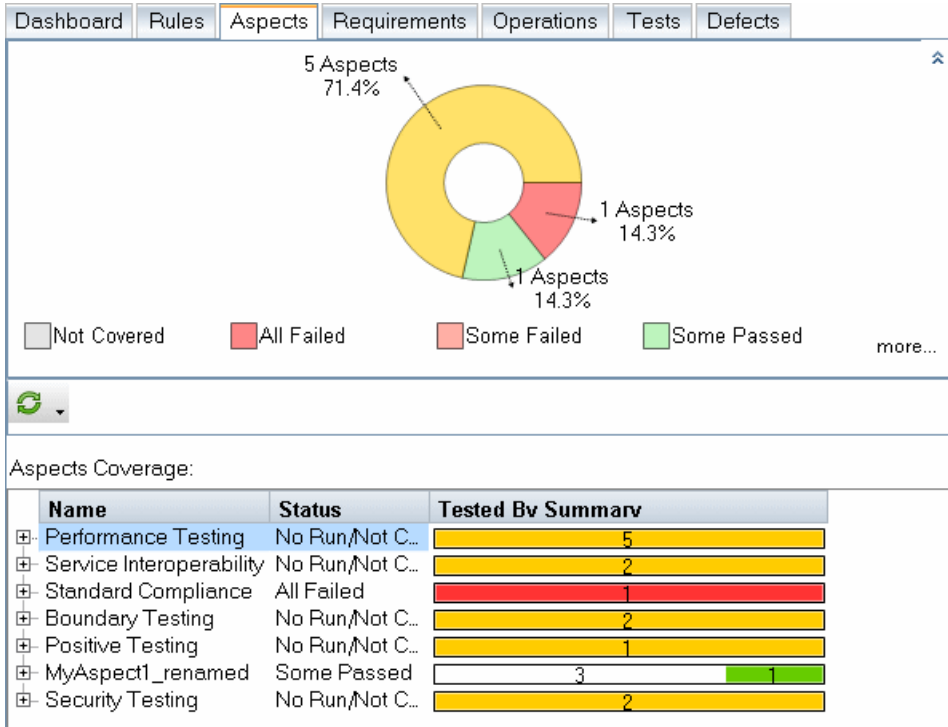
The following elements are included (unlabeled UI elements are shown in angle brackets>):

| UI Elements (A-Z) | Description |
|---|---|
|  | Select Columns. Enables you to select the columns to display in the Rule Violations grid. |
|  | Refresh. Reloads the list of rule violations. |
|  | Set Filter/Sort. Enables you to set a filter for the rule violations or sort them by a criteria. |

| UI Elements (A-Z) | Description |
|-------------------------------------|--|
| <Rule Violations grid> | A list of all the rule violations for the selected component. Default: Rule Name and Rule Description columns. Available Columns: Alert ID, Alert is active, and Component name. |
| Resolve All | Resolve All. Opens the Generate Requirements and Tests wizard to resolve the rule violations. For details, see the "Generate Requirements and Tests Wizard" on page 185. |




Aspects Tab

This tab displays the coverage per testing aspect in both a graphical representation and a grid layout.



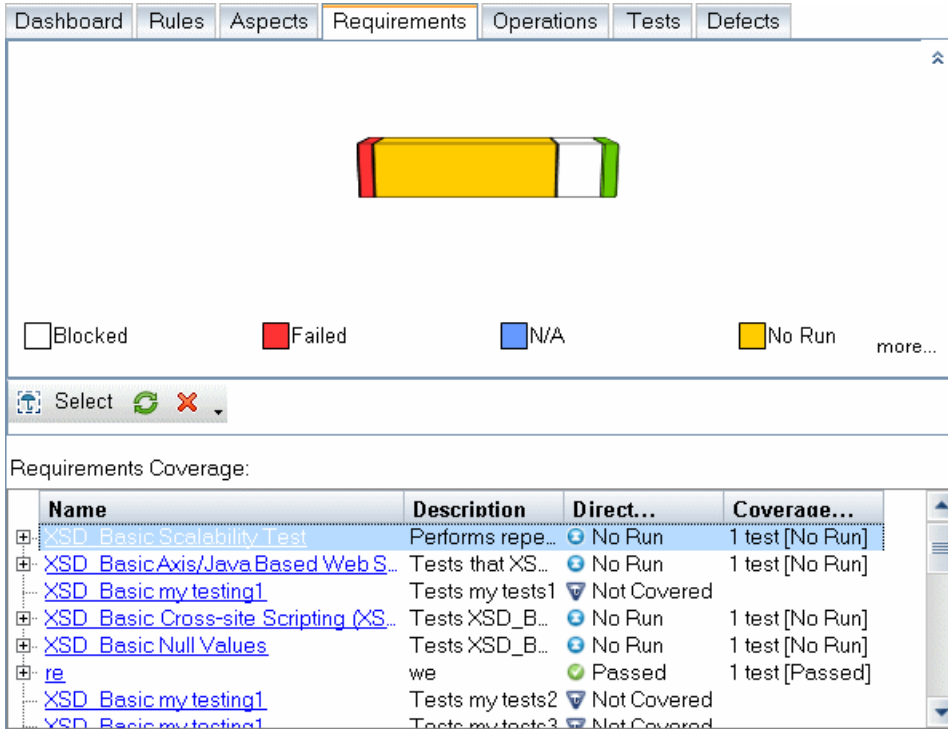
| | |
|------------------------------|---|
| To access | <p>Do the following:</p> <ul style="list-style-type: none"> • Select the Application Components module. • Select an entity in the Application Components tree. • Select the Testing Status > Aspects tab. |
| Important information | See "Aspect Coverage" on page 191. |
| Relevant tasks | "How to Check the Testing Status " on page 199 |

The following elements are included (unlabeled UI elements are shown in angle brackets>):

| UI Elements (A-Z) | Description |
|---|---|
|  | Refresh. Reloads the status of the testing aspects and refreshes the graph. |
|  | Collapse. Shows the aspect only, but not its tests or requirements. |
|  | Expand. Shows the tests associated with the aspect. |
| <graph pane> | A graphical representation of the requirement coverage |
| <grid pane> | A list of all aspects associated with the component. <ul style="list-style-type: none">• Expand the aspect to see a list of its associated tests.• Click on a test to open it in the Test Plan module.• Click Application Components in the left pane to return to the most recent view of the component. |

Requirements Tab






The Requirements tab displays the coverage per testing requirement in both a graphical representation and a grid layout.



| | |
|------------------------------|--|
| To access | <p>Do the following:</p> <ul style="list-style-type: none"> • Select an entity in the Application Components tree. • Select the Testing Status > Requirements tab. |
| Important information | See "Requirements Coverage" on page 192. |
| Relevant tasks | "How to Check the Testing Status " on page 199 |

The following elements are included (unlabeled UI elements are shown in angle brackets>):






| UI Elements | Description |
|---------------|--|
| Select | Opens the Requirements tree enabling you to add more |

| UI Elements | Description |
|---|--|
| | requirements to your test. For details see "Requirements Tree Panel" below. |
|  | Select Columns. Opens the Select Column dialog box enabling you to select the columns to display in the grid. |
|  | Refresh. Reloads the status of the requirements and refreshes the graph. |
|  | Remove. Unlinks the requirement and the component. Note that unlinking a requirement may remove the linkage to all tests and defects that are associated with the component only through this requirement. |
|  | Collapse. Collapses the graph window and show more of the grid. |
|  | Expand. Shows the graph window and displays less rows of the grid. |
| <graph pane> | A graphical representation of the requirement coverage. |
| <grid pane> | A list of all requirements associated with the component. <ul style="list-style-type: none"> • Expand a requirement to see a list of its associated tests • Click on a requirement to open it in the Requirements module. • Click on a test to open it in the Test Plan module. • Click Application Components in the left pane to return to the most recent view of the component. |

Requirements Tree Panel

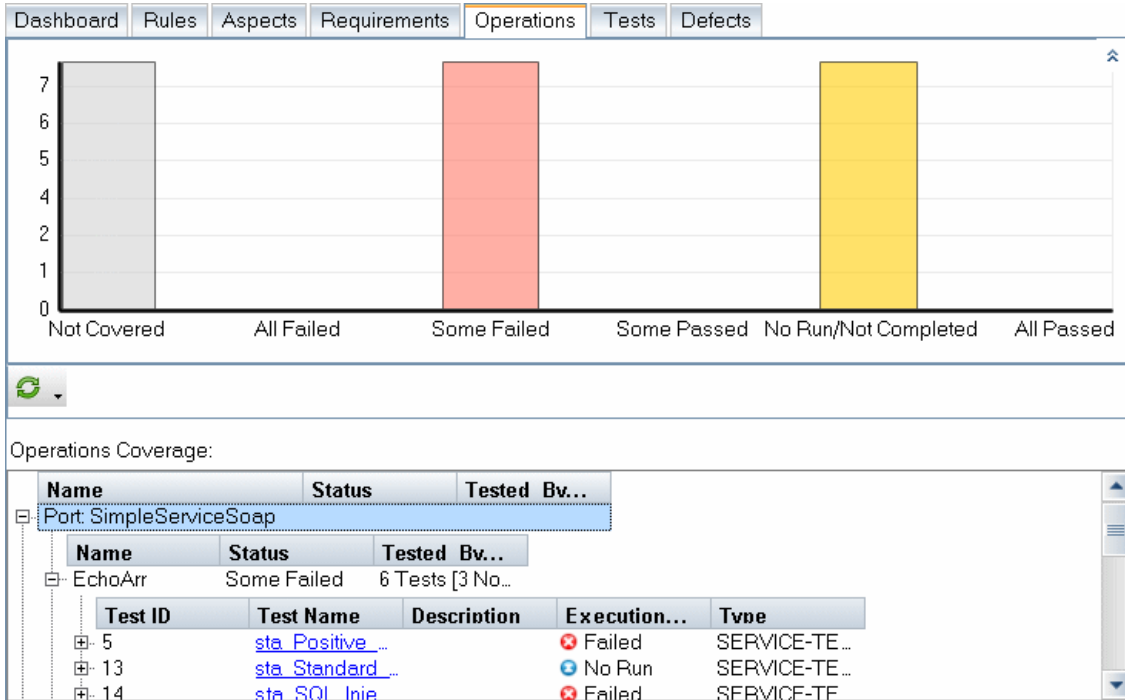
The following elements are included (unlabeled UI elements are shown in angle brackets):

| UI Elements | Description |
|---|---|
|  | Add Coverage. Adds coverage for the selected |

| UI Elements | Description |
|---|--|
| | requirement. If you select a parent requirement, it adds all sub-requirements. |
|  | Refresh. Refreshes the Requirements tree. |
|  | Find. Opens the Find dialog box for locating a requirement. |
|  | <p>Set/Clear Filter. Sets or clears the filter.</p> <ul style="list-style-type: none"> • Set Filter / Sort. Opens the Set Filter/Sort dialog box allowing you to filter out and sort the requirements by any field. • Clear Filter / Sort. Clears all filters and sorting settings and displays all requirements. |
|  | Go to Requirement by ID. Locates the requirement whose ID you specify. |
|  | <p>Close. Closes the Requirements tree pane.</p> <p>To reopen this view, click the Select button in the Requirements tab.</p> |
| <Requirements tree> | <p>A tree view of all of the requirements by folder and group. An icon indicates the requirement type.</p> <p>Tip: If a filter is active, there could be hidden requirements. To see all requirements, click Clear Filter.</p> |




Operations Tab

This dialog box displays the coverage per operation in both a graphical representation and a grid layout.



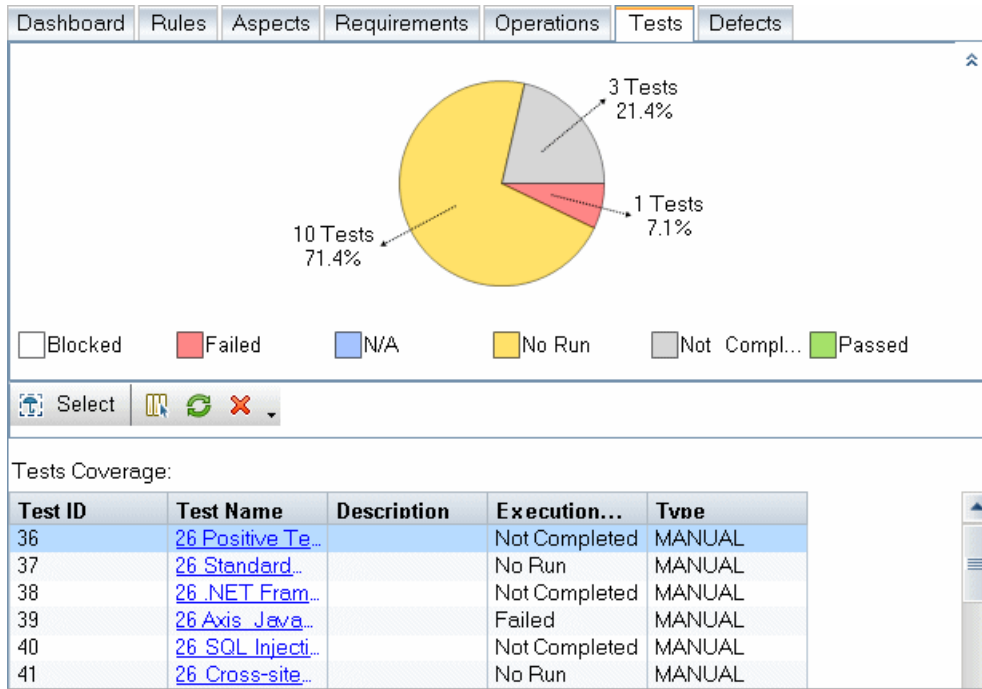
| | |
|-------------------------------------|--|
| <p>To access</p> | <p>Do the following:</p> <ul style="list-style-type: none"> • Select the Application Components module. • Select an entity in the Application Components tree. • Select the Testing Status > Operations tab. |
| <p>Important information</p> | <p>See "Operations Coverage" on page 194.</p> <p>Note: Available only for Application Component type entities.</p> |
| <p>Relevant tasks</p> | <p>"How to Check the Testing Status " on page 199</p> |

The following elements are included (unlabeled UI elements are shown in angle brackets>):

| UI Elements | Description |
|---|--|
|  | Refresh. Reloads the status of the operations and refreshes the graph. |
|  | Collapse. Collapses nodes with descendants. |
|  | Expand. Shows the operations and their testing status. |
| <graph pane> | A graphical representation of the Operations coverage. |
| <grid pane> | A list of all operations associated with the component. <ul style="list-style-type: none">• Expand the operation to see a list of its associated tests.• Click a test to open it in the Test Plan module.• Click Application Components in the left pane to return to the most recent view of the component. |






Tests Tab

This dialog box displays the status per test in both a graphical representation and a grid layout.






| | |
|------------------------------|---|
| To access | <p>Do the following:</p> <ul style="list-style-type: none"> • Select the Application Components module. • Select an entity in the Application Components tree. • Select the Testing Status > Tests tab. |
| Important information | See "Tests Coverage" on page 195. |
| Relevant tasks | "How to Check the Testing Status " on page 199 |




The following elements are included (unlabeled UI elements are shown in angle brackets>):

| UI Elements | Description |
|---|---|
| Select | Opens the Tests tree enabling you to associate more tests with your component. For details see " Tests Tree Panel " below. |
|  | Select Columns. Opens the Select Column dialog box enabling you to select which columns to show in the grid. |
|  | Refresh. Reloads the status of the test and refreshes the graph. |
|  | Remove. Unlinks the test from the requirement through which it is associated with the component. |
|  | Collapse. Collapses the graph window and shows more of the grid. |
|  | Expand. shows the graph window and displays less rows of the grid |
| <graph pane> | A graphical representation of the test coverage. |
| <grid pane> | A list of all tests associated with the component. <ul style="list-style-type: none"> • Click the test link to go to the test in the Test Plan module. • Click the Application Components to return to the most recent view of the component. |

Tests Tree Panel

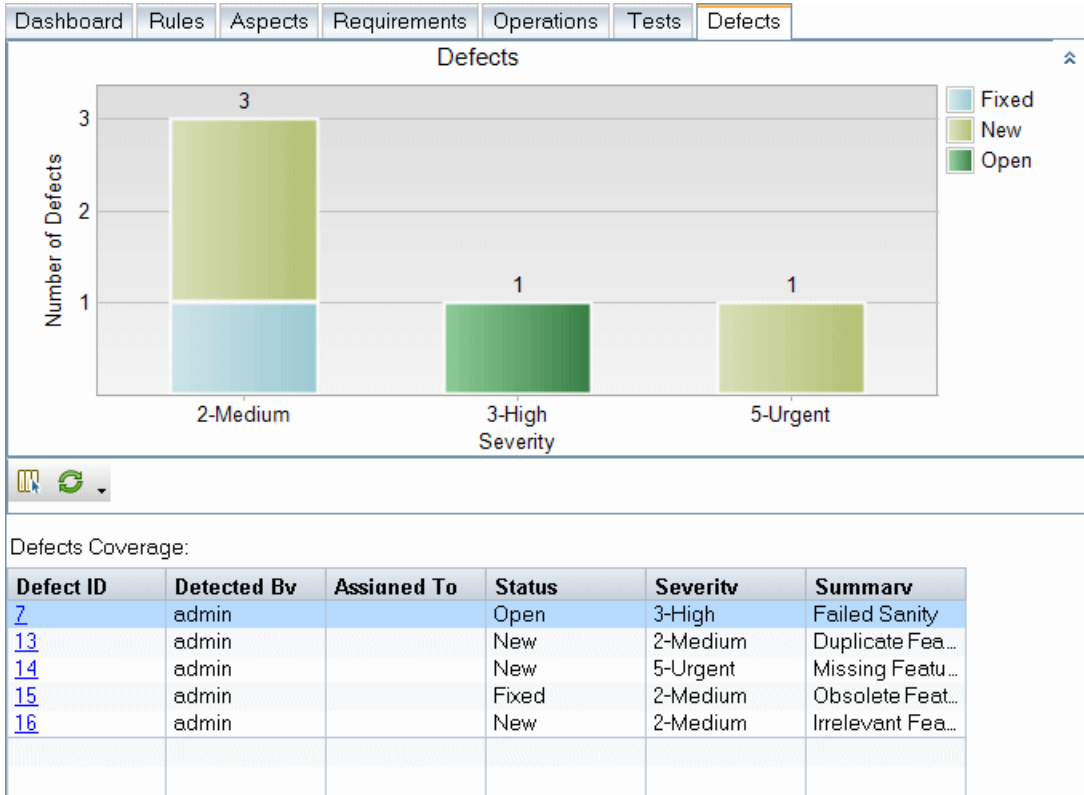
The following elements are included (unlabeled UI elements are shown in angle brackets):

| UI Elements | Description |
|---|--|
|  | Add Coverage. Adds the selected test to the coverage of the component or requirement. If you select a folder or group, it adds all sub tests. |
|  | Refresh. Refreshes the Tests tree. |
|  | Find. Opens the Find dialog box for locating a requirement. |

| UI Elements | Description |
|---|--|
|  | <p>Set/Clear Filter. Sets or clears the filter.</p> <ul style="list-style-type: none">• Set Filter / Sort. Opens the Set Filter/Sort dialog box allowing you to filter out and sort the tests.• Clear Filter / Sort. clears all filters and sorting settings and displays all tests. |
|  | <p>Go to Test by ID. Locates the test whose ID you specify.</p> |
|  | <p>Closes the Tests tree pane.</p> <p>To reopen this view, click the Select button in the Tests tab.</p> |
| <Requirements Tree> | <p>A tree view of all of the tests by folder and group</p> <p>Tip: If a filter is active, there could be hidden tests. To see all tests, click Clear Filter.</p> |



Defects Tab

This dialog box displays the status of the defects in both a graphical representation and a grid layout. It shows their status and the number of the defects, and provides direct links to those defects.



| | |
|------------------------------|---|
| To access | <p>Do the following:</p> <ul style="list-style-type: none"> • Select the Application Components module. • Select an entity in the Application Components tree. • Select the Testing Status > Defects tab. |
| Important information | <p>See "Defects Coverage" on page 197.</p> <p>Note: Available only for Application Component and Group type entities.</p> |
| Relevant tasks | <p>"How to Check the Testing Status " on page 199</p> |

The following elements are included (unlabeled UI elements are shown in angle brackets>):

| UI Elements | Description |
|---|---|
|  | Select Columns. Opens the Select Column dialog box enabling you to select which columns to show in the grid. |
|  | Refresh. Reloads the status of the defects and refresh the graph. |
| <graph pane> | A graphical representation of the defects coverage. |
| <grid pane> | A list of all defects associated with the component and their status. <ul style="list-style-type: none">• Click on a defect in the Defect ID column to open it in the Defects module.• Click Application Components in the left pane to return to the most recent view of the defects coverage. |

Linking Components and Requirements

Concepts

- "Linking Components Overview " on page 222

Tasks

- "How to Link Components to Requirements and Tests" on page 224

Reference

- "Linking Component User Interface " on page 226
- "Link <test-name> to Requirement Dialog Box" on page 227
- "Application Components Tab - Requirements Module " on page 229
- "Application Components Tab - Test Plan Module " on page 231
- "Application Components Tree Panel " on page 233

Concepts

This section includes:

- ["Linking Components Overview "](#) on page 222

Linking Components Overview

To test components in Service Test Management, you must link them to requirements or tests. Requirement coverage is used to connect components to other Application Lifecycle Management entities such as tests, test sets, and defects.

Linking components to requirements and tests helps you keep track of the relationship between your requirements and components, and ensures compliance with your requirements throughout the testing process.

For information on viewing test and requirement coverage, see "[Determining Test Coverage](#)" on page 187

When working in the **Requirements** module, you can view the application components linked to each requirements. The **Application Components** tab displays a list of those components, with other basic information.

Using the **Test Plan** module's **Application Components** tab (available only after you enable the Service Test Management extension), you can set and view the application components linked to each test.

When you link a component to a test, Service Test Management prompts you to select a requirement. To allow you to track your testing through ALM, you link tests to the default requirement or to an existing one.

Tasks

This section includes:

- ["How to Link Components to Requirements and Tests" on page 224](#)

How to Link Components to Requirements and Tests

This task describes how to link components to either requirement or tests. If you are linking to a test, you also indicate the requirement to associate with the test.

This task includes the following steps:

- Link a component to a requirement - optional
- Link a component to a test - optional
- Results

Link a component to a requirement - optional

Use the **ApplicationComponents** tab in the **Requirements** module to link components to requirements. For details, see "[Application Components Tab - Requirements Module](#) " on page 229.

Link a component to a test - optional

Use the **ApplicationComponents** tab in the **Test Plan** module to link a components to tests. For details, see "[Application Components Tab - Test Plan Module](#) " on page 231.

When you add a test to component coverage, the Link <test-name> to Requirement dialog box opens and prompts you to choose the **Default Requirement** or **Linked Requirements**. Select a requirement and click **OK**. For details, see the "[Link <test-name> to Requirement Dialog Box](#)" on page 227.

You can also link a component to a test through the **Testing Status** tab. For details, see "[Tests Coverage](#)" on page 195.

Results

You can view and modify the linkages in the **Application Components** tab within the **Test Plan** or **Requirements** modules. For details, see "[Linking Component User Interface](#) " on page 226.

Reference

This section includes:

- "Linking Component User Interface " on page 226
- "Link <test-name> to Requirement Dialog Box" on page 227
- "Application Components Tab - Requirements Module " on page 229
- "Application Components Tab - Test Plan Module " on page 231
- "Application Components Tree Panel " on page 233

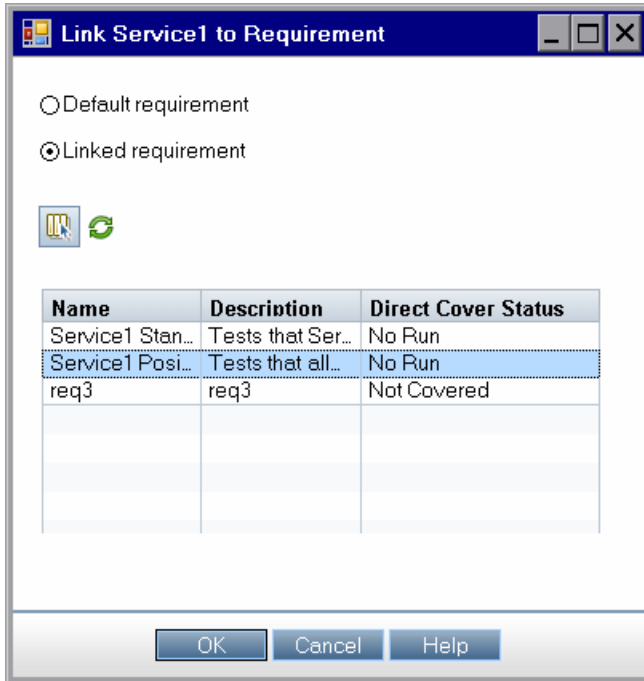
Linking Component User Interface


This section includes (in alphabetical order):

- "Link <test-name> to Requirement Dialog Box" on page 227
- "Application Components Tab - Requirements Module " on page 229
- "Application Components Tab - Test Plan Module " on page 231
- "Application Components Tree Panel " on page 233



Link <test-name> to Requirement Dialog Box

This dialog box enables you to select a requirement to associate with your test.



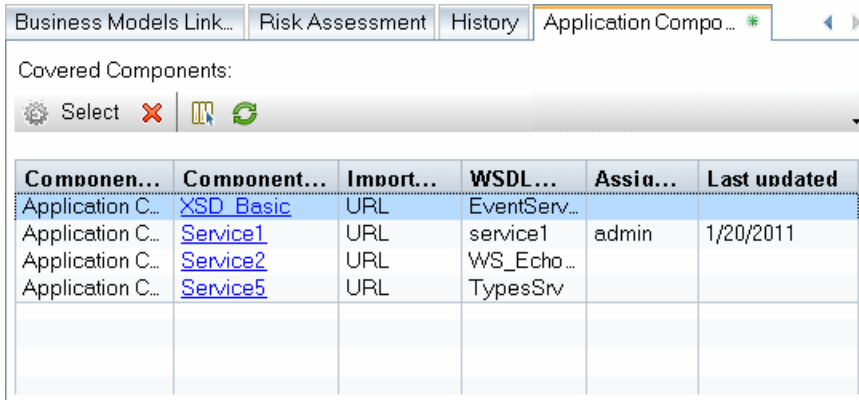
| | |
|------------------------------|--|
| To access | <p>Do the following:</p> <ol style="list-style-type: none"> 1. Open the Test Plan module, and select the ApplicationComponents tab (you may need to scroll to the right). 2. Click Select to open the selection panel, and choose a entity from the Application Components Tree tab. 3. Click the  button. |
| Important information | This dialog box only opens if the application component is linked to a requirement other than the default one. |
| Relevant tasks | "How to Link Components to Requirements and Tests" on page 224 |
| See also | "General Tab " on page 249 (Customization) |

User interface elements are described below (unlabeled elements are shown in angle brackets):

| UI Elements (A-Z) | Description |
|---|---|
|  | Select Columns. Opens the Select Column dialog box enabling you to select the columns to display in the grid. |
|  | Refresh. Reloads the components in the list. |
| <Linked Requirements list> | A list of the requirements associated with the test. To modify the columns, click the Select Columns button. |
| Default requirement | Default Requirement. Creates a link between the default requirement of the component and the test, if it does not already exist. The linked requirements list is not relevant for this option. You can set the default requirement in the Customization section, For details, see " Select Requirement " on page 250. |
| Linked requirement | Enables you to select a specific requirement to link to the test from the Linked Requirements list. This option is not available if the component or group has no linked requirements. To link a requirement with a component, see " How to Check the Testing Status " on page 199. |

Application Components Tab - Requirements Module




This dialog box enables you to link components to an existing requirement.



| | |
|-----------------------|--|
| To access | <p>Do the following:</p> <ul style="list-style-type: none"> • Open the Requirements module. • Select View > Requirement Details. • Expand the tree and select a requirement. • In the right pane, scroll right to ApplicationComponents tab. |
| Relevant tasks | "How to Link Components to Requirements and Tests" on page 224 |
| See also | "How to Check the Testing Status " on page 199 |

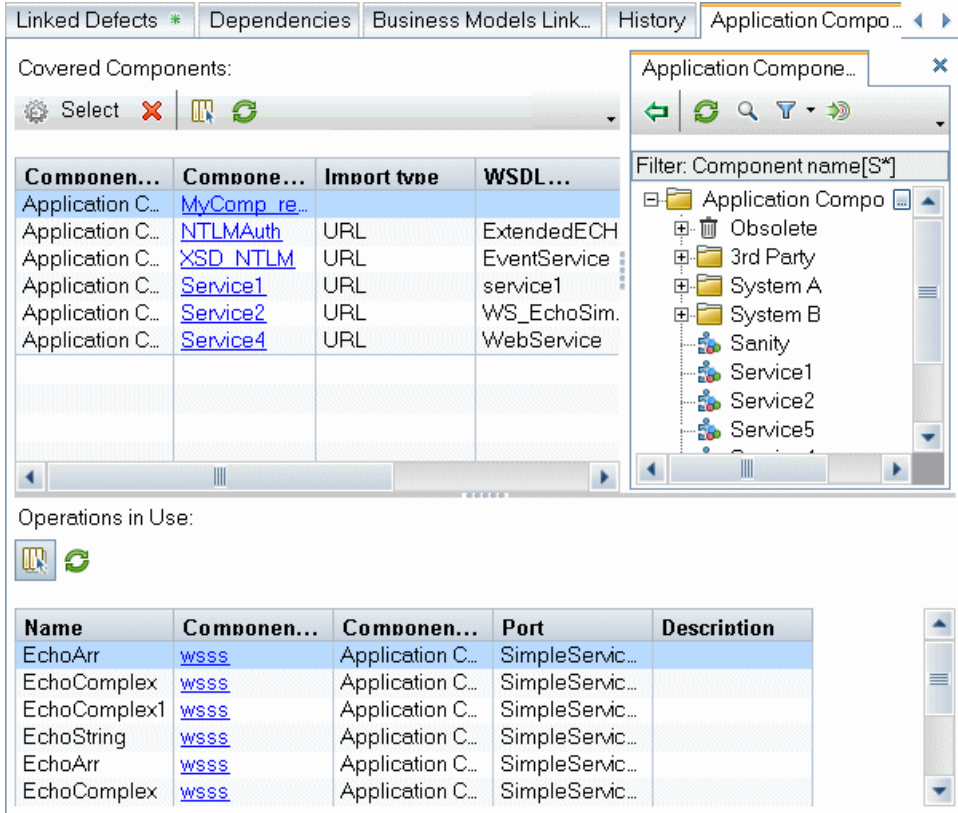
User interface elements are described below (unlabeled elements are shown in angle brackets>):

| UI Elements (A-Z) | Description |
|-------------------|---|
| Select | Opens the ApplicationComponentsTree tab in the right pane to link a component to the requirement. For details, see the "Application Components Tree Panel " on page 233. |
| | Delete. Unlinks the component and requirement. |

| UI Elements (A-Z) | Description |
|---|--|
|  | Select Columns. Opens the Select Column dialog box enabling you to select the columns to display in the grid. |
|  | Refresh. Reloads the components in the grid. |
| <Linked Components Grid> | <p>A list of the components linked to the selected requirement. The displayed columns differ per component type. For Web Services, they are:</p> <ul style="list-style-type: none"> • Component folder path. The application component path in the tree hierarchy. • Component name. The name of the application component. • Import type. For Web Services, the import type: File, URL, UDDI, Systinet • WSDL native name. The Web service name as defined in the WSDL. • Assigned to QA. QA engineer responsible for validating the requirement. • Last updated. The date the component was last updated from its original source. <p>To change the columns displayed, click the  button.</p> |





Application Components Tab - Test Plan Module

This tab enables you to link components to an existing test.



| | |
|-----------------------|--|
| To access | <p>Do the following:</p> <ul style="list-style-type: none"> • Open the Test Plan Module. • Select View > Test Plan Tree. • Expand tree and select a test in the left pane. • In the right pane, scroll right to the ApplicationComponents tab. |
| Relevant tasks | "How to Link Components to Requirements and Tests" on page 224 |
| See also | "How to Check the Testing Status " on page 199 |

User interface elements are described below (unlabeled elements are shown in angle brackets):





| UI Elements (A-Z) | Description |
|---|---|
| Select | Opens the ApplicationComponents Tree panel in the right pane to enable linking a component to the test. For more information, see the " Application Components Tree Panel " on page 233. |
|  | Delete. Unlinks the test from all requirements that associate it with the component. |
|  | Select Columns. Opens the Select Column dialog box enabling you to select the columns to display in the grid. |
|  | Refresh. Reloads the list in the grid. |
| <Linked Components grid> | <p>A list of components linked to the test. The displayed columns may differ based on the component type.</p> <ul style="list-style-type: none"> • Component folder path. The path in the tree hierarchy. • Component name. The name of the application component. • Import type. The import type: File, URL, UDDI, Systinet (Web Services only). • WSDL native name. The service name as defined in the WSDL (Web Services only). • Assigned to QA. QA engineer responsible for validating the test. • Last updated. The date the component was last updated from its original source. <p>To change the columns displayed, click the  button.</p> |


Application Components Tree Panel

This tab enables you to link components to an existing test.

| | |
|-----------------------|---|
| To access | <p>Do the following:</p> <ol style="list-style-type: none"> 1. Open the Test Plan or Requirements module. 2. Select View > Test Plan Tree or View > Requirement Details. 3. In the right pane, scroll right to the ApplicationComponents tab. 4. Click the Select button. |
| Relevant tasks | "How to Link Components to Requirements and Tests" on page 224 |
| See also | "How to Check the Testing Status " on page 199 |

The following elements are included (unlabeled UI elements are shown in angle brackets>):

| UI Elements | Description |
|---|--|
|  | Add Component. Adds the selected component to the Linked Components grid. |
|  | Refresh. Refreshes the Application Components Tree panel. |
|  | Find. Opens the Find dialog box for locating an application component by its name or another criteria. For details, click F1 in the Find dialog box. |
|  | <p>Set/Clear Filter. Opens a drop down list with the following options:</p> <ul style="list-style-type: none"> • Set Filter / Sort. Opens the Set Filter/Sort dialog box allowing you to filter and sort the components. • Clear Filter / Sort. Clears all filters and sorting settings and displays all components and groups. |

| UI Elements | Description |
|---|---|
|  | Go to Application Component by ID. Locates the application component whose ID you specify. |
| ✕ | Closes the Application Components Tree panel. To reopen this view, click the Select button. |
| < Application Components panel > | A tree view of all of the application components by folder and group (only visible when you click Select). Tip: If a filter is active, there could be hidden components. To see all components, click Clear Filter . |

Customizing Service Test Management

Concepts

- "Customization Overview" on page 237
- "Customizing the Application Components Module " on page 240

Tasks

- "How to Customize a Project" on page 244

References

- "Project Customization User Interface" on page 248
- "General Tab " on page 249
- "Rules Tab " on page 253
- "Aspects Tab " on page 255
- "Component Types Tab" on page 257
- "Project Entities Pane" on page 260
- "Groups and Permissions " on page 263
- "Module Access Pane" on page 266

Concepts

This section includes:

- "Customization Overview" on page 237
- "Customizing the Application Components Module " on page 240

Customization Overview

Service Test Management adds settings to several of the customization entities:

For task details, see ["How to Customize a Project" on page 244](#).

Alert Rules

By default, Service Test Management issues an alert when a Service Test or UFT-API Test has unresolved steps as a result of a Web Service application component update. Using this module, you can disable or enable the alert.

For details about updating an application component, see the ["Update Linked Tests Dialog Box" on page 99](#)

Application Components

This section lets you specify default values for components stored in secure locations and on a proxy server, the default Systinet and UDDI server information, the default Requirement folder, user-defined testing aspects, component types, and rules. For details, see ["Customizing the Application Components Module " on page 240](#)

Business Views

Business views are a semantic data layer that can be used for reporting. You can create data relations between different ALM entities, using only those entities that are relevant to you. Service Test Management enables application component entities, such as the application component groups and folders, and operations.

- 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 the user creates.
- For ALM 11.50, you can only use business views as a basis for graphs.

For details, see the Project Customization section in the ALM Administrator Guide.

Groups and Permissions

In this section you can customize the permission settings for user groups and

folders and the ability to link relating to their access to application components, linkage to requirements, and ability to associate application components with groups.

For task details, see ["How to Customize a Project"](#) on page 244.

For user interface details, see ["Groups and Permissions "](#) on page 263

Module Access

In the Module Access section you can grant or deny access to the Application Components module. For details, see the ["Module Access Pane"](#) on page 266

Project Entities

This section lets you customize the system fields and create new user-defined fields for Application Components, Application Component Changes, and Application Component groups. You can create new fields and indicate whether or not a field is mandatory.

If you define a field as required, the **Details** tab shows it as mandatory for all component types. The following example shows a mandatory custom field defined for an Application Component group.

The screenshot shows a user interface with two tabs: 'Details' and 'Description'. The 'Details' tab is selected and contains the following fields:

- * Custom_Info: [Text input field]
- Assigned to: [Dropdown menu]
- Created by: [Text input field containing 'admin' and a dropdown arrow]
- Creation date: [Date dropdown menu]

Below the 'Details' tab is the 'Description' tab, which is currently empty.

For details, see the ["Project Entities Pane"](#) on page 260

Project Report Templates

This section lets you upload and create report templates for application component information and changes to the components.

Note: This chapter describes the customization options that apply specifically to Service Test Management. For details on Project Customization, refer to the Application Lifecycle Management Administrators Guide.

Customizing the Application Components Module

You can customize general project data in the **Application Components** module. This includes specifying default values for components stored in secure locations and on a proxy server, the default Systinet and UDDI server information, the toolkit parsing order for importing component definitions, the default requirement folder, user-defined testing aspects, component types, and rules.

This section also includes:

- Testing Aspects
- Rules
- Component Types

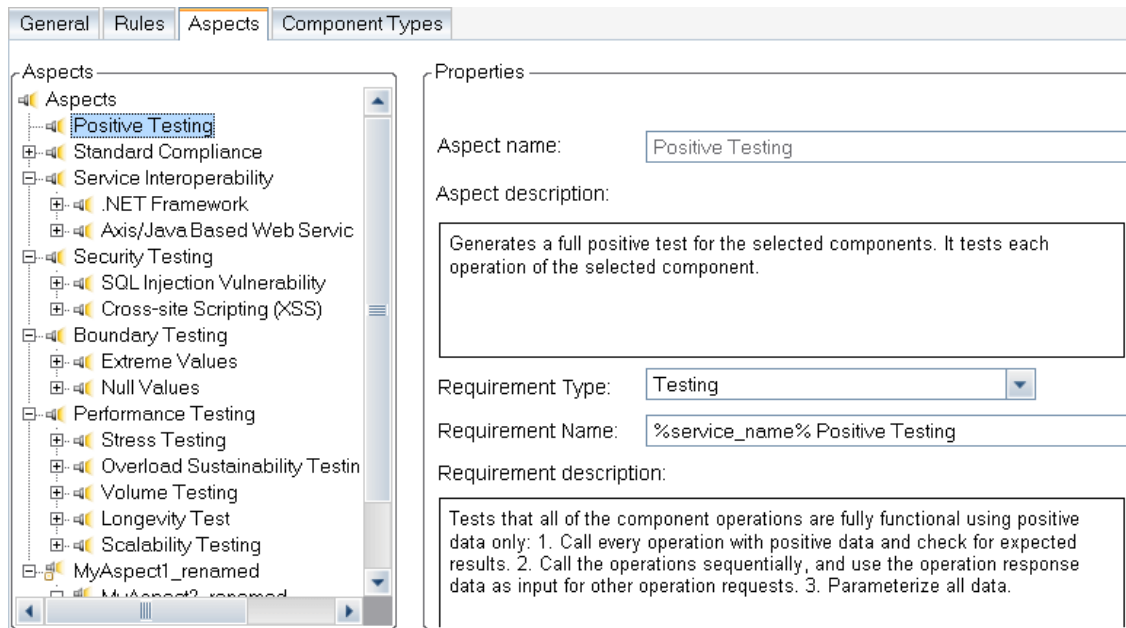
Testing Aspects



Testing aspects are criteria you choose for testing an application component. You can indicate the testing aspects to which the component must comply in order it to receive a Passed status in testing. For example, you can specify Positive Testing and Standard Compliance. For a list of the available aspects, see "[Generating Requirements and Tests Overview](#) " on page 178.

When using the **Requirement and Test Generation** wizard (**Components > Generate Requirements/Tests**), you select the aspects you want to test and the wizard creates all of the necessary tests and requirements. For details about the wizard, see the "[Generate Requirements and Tests Wizard](#)" on page 185.

Service Test Management's customization section lets you control which aspects to display in the wizard.

The customization's **Aspects** tab also lets you add new aspects, modify the properties of system and user-defined aspects, and delete user-defined aspects.



In the Aspects tree hierarchy, the  icon indicates a system-defined aspect. The  icon indicates a user-defined aspect.

In the **Rules** tab, you select the aspects for user-defined rules. Application components that were not tested for these aspects, generate a rule violation and appear with an alert in the application component tree. For details, see the "Rules Tab " on page 253.

For task details, see "How to Customize a Project" on page 244.

Rules

Rules let you define the testing aspects to which your component must comply. They help you make sure your testing standards are satisfactory and that the aspects relevant to the component are covered.

You define a rule through filters and aspects:

- **Filters.** Filters indicate the components to which the rule will be applied. For example, if you indicate a filter: **Assigned To=john_qa**, any component not assigned to **john_qa**, will not be bound by the rule. You can set a rule for a component based on a variety of filters. You can use the conditional and logical operators that are used in all ALM filters. For details, see the *HP Application Lifecycle Management User Guide*.
- **Aspects.** The testing aspects that must be covered for the application component.

If your application component was not covered by the required aspects, Service Test Management creates an **Alert**. The **Testing Status > Rules** tab lists the violations, and lets you resolve the problems. For details about defining rules, see ["How to Customize a Project" on page 244](#).

Component Types

The customization lets you define new component types. The built-in types are General, Web Service, and JMS.

You can specify the following information for your component: **Name**, **Assigned Icon**, **Supported file extensions**, **Contract parser ID**, and the fields to display in its **Details** and **Interaction** tabs. The **Contract parser ID** instructs Service Test Management that if a parser has this unique ID and is registered on the client machine, use it to parse contract files of this type.

For details, see ["How to Customize a Project" on page 244](#).

Tasks

This section includes:

- ["How to Customize a Project" on page 244](#)

How to Customize a Project

This task describes how to customize your ALM project as it relates to components. You can create rules, assign aspects, and set permissions for entities related to your application component. You can also create both mandatory or optional fields to the Component Details.

All of the customization settings are optional.

Open Project Customization

- Choose Tools > Customize to open the Customization module.
- Click the **Return** button, located in the top right corner, to return to the main ALM window.

Customize the General settings - optional

Click Application Components in the left pane. Select the **General** tab. Define the parsing order for toolkits, and the default servers that host the WSDL, UDDI registry, or Systinet registry. You can also define the authentication details for accessing WSDL files, the default requirement and whether to allow fuzzy imports.

For user interface details, see the ["General Tab "](#) on page 249.

Customize the component rules - optional

Click Application Components in the left pane. Select the Rules tab.

Click **New** to define new rules. Select the testing aspects to which the components in the current filter, must comply. If the component is not covered by the specified aspects, it issues a rule violation. For details, see ["Rule Violations"](#) on page 190.

For user interface details, see the ["Rules Tab "](#) on page 253.

Customize the aspects - optional

Click Application Components in the left pane. Select the Aspects tab. Manage the built-in testing aspects or click **New** to create new ones.

For user interface details, see the ["Aspects Tab "](#) on page 255.

Manage the component types - optional

Click Application Components in the left pane. Select the Component Types

tab.

Manage the built-in component types **General**, **WebServices**, and **JMS**. You can indicate the fields that will be available for each of the built-in types. Mark required fields as mandatory and indicate whether to display them in the **Details** or **Interaction** tab.

Click **New** to create a new component type. Specify a **Name**, **Assigned Icon**, **Supported file extensions**, and **Contract Parser ID**. Select the fields to display for the new component, for example in its **Details** tab. You can also mark fields as mandatory and indicate whether they will be visible in the **Interaction** tab.

For user interface details, see the "[Component Types Tab](#)" on page 257.

Customize group permissions - optional

Assign the permissions for the component-related tasks per user group. For a list of the component-related tasks that you can configure, see "[Groups and Permissions](#)" on page 263.

1. Click **Groups and Permissions** in the left pane.
2. Click **New Group** to open the **New Group** dialog box. Specify a name and permission level. Click **OK**.
3. Select the **Permissions** tab and click the **Application Components** subtab. The subtabs are in alphabetical order.
4. Modify the permission levels as required.
5. Click **Save**.

Customize application component module access - optional

Indicate which user groups will have access to the **Application Components** module.

1. Click **Module Access** in the left pane.
2. Expand the window or scroll to the right to view the **Application Components** column.
3. Select the check boxes as required. To enable or disable permissions for an entire column, select or clear the check box in the header row.
4. Click **Save**.

Customize project entities - optional

Use the Project Entities section to define new fields for the application component, changes, and groups.

1. Click Project Entities in the left pane.
2. Expand the node to which you want to add or modify an entity.
**Application Component, Application Component Changeor
Application Component Group.**
3. To modify a built-in field, expand the **System Fields** node and select a field. You can modify the label and indicate if the field should be required.
4. To create a new field, select the **User Fields** node and click the **New Field** button. Provide a label, type, and maximum string length for **String** type fields. Select any other relevant options as described in the "[Project Entities Pane](#)" on page 260.
5. To create a new `Memo` type field, expand the **New Field** button and select **New Memo Field**.
6. Click **Save**.

Customize the business views - optional

Use the Project Entities section to modify the built-in business views and create new ones.

1. Click Business Views in the left pane.
2. Clear or select the check boxes in the Output column to indicate which columns to show in the final view.
3. Use the interface to create new views or import existing ones. For details, refer to the HP ALM Administrators Guide.

Reference

This section includes:

- "Project Customization User Interface" on page 248
- "General Tab " on page 249
- "Rules Tab " on page 253
- "Aspects Tab " on page 255
- "Component Types Tab" on page 257
- "Project Entities Pane" on page 260
- "Groups and Permissions " on page 263
- "Module Access Pane" on page 266


Project Customization User Interface

This section includes (in alphabetical order):

- "General Tab " on page 249
- "Rules Tab " on page 253
- "Aspects Tab " on page 255
- "Component Types Tab" on page 257
- "Project Entities Pane" on page 260
- "Groups and Permissions " on page 263
- "Module Access Pane" on page 266

General Tab

This tab enables you to customize the settings for parsing contracts and importing WSDLs. You can enter a default server and credentials, so that you will not need to enter the information each time you import a service.

| | |
|------------------------------|---|
| To access | <p>Do the following:</p> <ol style="list-style-type: none"> 1. Select Tools > Customize. 2. Select Application Components in the left pane. 3. Select the General tab. |
| Important information | <p>To resume work in the ALM modules, click the  button in the top right corner.</p> |
| Relevant tasks | <p>"How to Customize a Project" on page 244</p> |





User interface elements are described below:

| General Tab Sections | Description |
|--------------------------------|---|
| Connection Settings | <ul style="list-style-type: none"> • Default Connection Settings. Opens the Connection Setting dialog box for setting the default connection settings. For details, see "Connection Settings Dialog Box" on page 76. |
| Default UDDI Server | <ul style="list-style-type: none"> • URL. The default address of the UDDI server when a user opens the "Select Service from UDDI Dialog Box" on page 74, for example <code>http://<my_server>.<my_domain>:8090/juddi/inquiry</code>. • Version. The UDDI version: 2 or 3. <p>For details, see "Select Service from UDDI Dialog Box" on page 74.</p> |
| Default Systinet Server | <p>Server Name. The default path of the Systinet server when a user opens the "Select Service from Systinet Dialog Box" on page 72.</p> |
| Default Requirements | <ul style="list-style-type: none"> • Location. The default requirement to use when linking a component to a test. The Browse button opens the "Select Requirement" below dialog box. • Requirement Type. One of the standard requirement types: Undefined, Functional, Testing, Performance, or Business Model. |
| Fuzzy Import | <ul style="list-style-type: none"> • Use fuzzy logic on import. For .NET services, use a looser set of criteria for parsing the contract. • Number of Attempts. The number of attempts to try parsing the contract before aborting the import. The default is 5. |

Select Requirement




This dialog box lets you select a default requirement.

User interface elements are described below (unlabeled elements are shown in angle brackets>):

| UI Elements (A-Z) | Description |
|---|--|
|  | Refresh All. Refreshes all of the requirements in the tree. |
|  | <p>Set/Clear Filter. Opens a drop down list with the following options:</p> <ul style="list-style-type: none"> • Set Filter / Sort. Opens the Set Filter/Sort dialog box allowing you to filter and sort the requirements. • Clear Filter / Sort. Clears all filters and sorting settings and displays all requirements. <p>For details, see "General Tab " on page 249</p> |
|  | New Requirement. Opens the New Requirement dialog box for defining a new requirement. |
|  | <p>New Requirement Folder. Creates a new requirement folder, under the selected folder.</p> <p>Note: This is only available when selecting a folder— not a requirement.</p> |
| <Requirements Tree> | A tree hierarchy of all the requirements. |

Filter Requirements Dialog Box

You open Filter Requirements dialog box from the Select Requirement dialog box. User interface elements are described below:

| UI Elements | Description |
|---|--|
|  | Clear Filter. Clears all conditions to which the rule should apply. |
|  | Copy filter. Copies the current filter settings onto the clipboard. You can paste this filter for another rule or save it to a text file for later use. |
|  | Paste filter. Uses a filter from the clipboard. |
| Filter tab | A list of field names and their conditions. The field names correspond to the component fields in the Project Entities list. |

| UI Elements | Description |
|-------------------------|---|
| | <p>Tip: Click the down arrow in the right side of the row, to open the Select Filter Condition dialog box. You can manually type the condition that must be met and use asterisks (*) as wildcards.</p> |
| Cross Filter tab | <p>Allows you to filter the following entities within the Application Components module: Groups, Requirements, Tests, Defects, Application Changes, and Alerts.</p> <p>For Alerts: Enable the Show application components with alerts option to hide all components without alerts.</p> |

Rules Tab

This tab lets you define rules for the application components in your project.

Properties

-Rule-

Rule name:

-Components-

The rule applies to components that satisfy the following condition(s):

-Aspects-

Must be covered by the following Test aspect(s):

-Description-

Rule Description (This text will be displayed whenever a component does not follow the rule):

Use automatic text

All components that meet this condition: Created by[admin] must be covered by the following testing aspects: Security Testing, Positive Testing, Standard Compliance

| | |
|------------------------------|---|
| To access | <p>Do the following:</p> <ol style="list-style-type: none"> 1. Select Tools > Customize. 2. Select Application Components in the left pane. 3. Select the Rules tab. |
| Important information | <p>To resume work in the ALM modules, click the <input type="button" value="Return"/> button in the top right corner.</p> |
| Relevant tasks | <p>"How to Customize a Project" on page 244</p> |
| See also | <p>"Alerts Dialog Box" on page 28</p> |

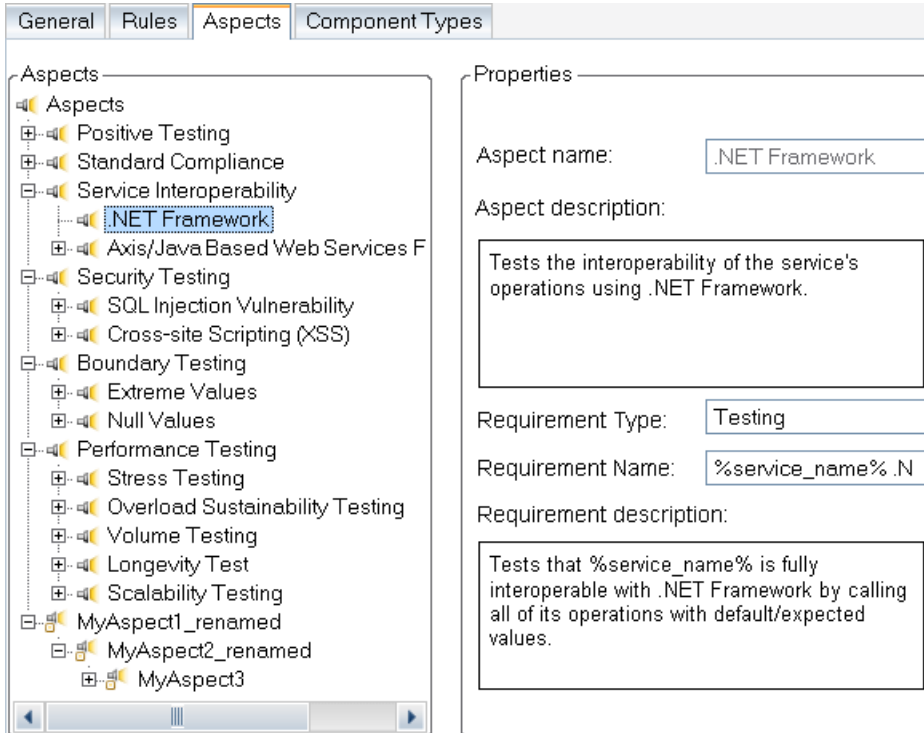
User interface elements are described below:


| UI Elements | Description |
|------------------------|---|
| Toolbar buttons | <ul style="list-style-type: none"> • Save. Saves the changes to the Rules tab. • New. Opens the Create New Rule dialog box. • Delete. Removes the selected rule. |
| Properties | <p>The properties of the rule:</p> <ul style="list-style-type: none"> • Rule. The rule name. If a component violates a rule, the Alert box lists it by this name. • Components. The rule is only applied to components that meet the selected conditions. To set a condition, click the Edit button to open the "General Tab " on page 249. To create a global rule that will apply to all components, specify a filter condition that is common to all components, for example, <code>Component ID > 0</code>. • Aspects. The testing aspects that must be covered for the component. To select the aspects, click the Edit button to open the Select Aspects dialog box described on page "Customization Overview" on page 237. Note: You can only select a direct descendent of the root aspect. • Description. Text describing the rule. To allow Service Test Management to generate a description based on the selected aspects and conditions, select Use automatic text. If the rule is violated, the Alerts dialog box shows this text. |

Aspects Tab


This tab lets you define the properties of the built-in aspects and define new ones for the project.

This dialog box enables you to run a spot test for an individual operation. You can set up a request and view the SOAP response to verify if an individual operation is functional.



| | |
|------------------------------|---|
| To access | Do the following: <ol style="list-style-type: none"> 1. Select Tools > Customize. 2. Select Application Components in the left pane. 3. Select the Aspects tab. |
| Important information | To resume work in the ALM modules, click the  button in the top right corner. |
| Relevant tasks | "How to Customize a Project" on page 244 |
| See also | "Alerts Dialog Box" on page 28 |

User interface elements are described below:

| UI Elements | Description |
|--|--|
|  Save | Saves the Rule settings. |
| New | Opens the Create New Aspects dialog box. |
| Delete | Removes the selected aspect from the list. You can only delete user-defined aspects—not built-in ones. |
| Aspects | A list of the aspects that are available for the current project. |
| Properties | <p>The properties of the aspect:</p> <ul style="list-style-type: none"> • Aspect Name. The aspect name as it will appear in the tree. For predefined system aspects, this field is read-only. • Aspect Description. Text describing the goal of the aspect. • Requirement type. The requirement type:— a drop down list of predefined requirements, such as Business, Group, or Performance, and any user-defined requirements. For details, see the <i>ALM User Guide</i>. • Requirement name. A name for the requirement. To include the component name, use the variable <code>%service_name%</code>. The Requirement and Test Generation wizard will use this name when creating a requirement for this aspect. • Requirement description. Text describing the requirement. To include the component name, use the variable <code>%service_name%</code>. |

Component Types Tab

This tab provides information about the different application component types. You can control which fields will be available in the **Details** and **Interaction** tabs for each component type and if they are mandatory. You can also define new component types.

| Field Name | Included in Type | Required in... | Interaction field |
|------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| Assigned to Dev | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Assigned to QA | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Comments | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Component name | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| Contract Description | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Description | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Development status | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Has changed | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| JMS connection factory | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| JMS destination | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| JMS provider | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| JMS request queue | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| JMS response queue | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

| | |
|-----------------------|---|
| To access | <p>Do the following:</p> <ol style="list-style-type: none"> 1. Select Tools > Customize. 2. Select Application Components in the left pane. 3. Select the Component Types tab. |
| Relevant tasks | "How to Customize a Project" on page 244 |

| UI Elements | Description |
|---------------|---|
| New | Adds a new component type to the list. To rename the component, enter text in the right pane's Name field. |
| Delete | Removes the selected non- built-in component from the list. |

| UI Elements | Description |
|-------------------------------|--|
| | <p>Note: You cannot remove a type if a component of that type exists in the repository.</p> |
| <p>Component types</p> | <p>A list of the built-in and custom component types. The built-in types are General, Web Service, and JMS.</p> |
| <p>Properties</p> | <p>The component type details:</p> <ul style="list-style-type: none"> • Name. The name of the component type as it will appear in ALM. • Assigned icon. The icon representing the component of this type in the application components tree. A drop-down includes several options. For details, see "Application Components Module Icons" on page 51. • Supported file extensions. The file extensions supported as a contract for components of this type, for example, WSDL. • Parser type. A type string for the parser to use for parsing a contract definition file, such as a WSDL. The syntax is the namespace and type followed the parser DLL, separated by a comma. For example, the standard Web Service parser string is <code>HP.STM.SOAPParser.Parsing.WSParser,WSParser.dll</code>. <p>Note:</p> <ul style="list-style-type: none"> ■ Make sure to add any custom parser DLLs to the GAC or to the folder of the ALM client. ■ Only .NET DLLs are supported. |
| <p>Fields</p> | <p>The fields that will be available in the component's Details and Interaction tabs. These settings are specific for each component type—both built-in and user-defined.</p> <ul style="list-style-type: none"> • Included in Type. Show the field in the components's Details tab, but do not make it mandatory. • Required for Type. Show the field in the components's Details tab, and make it mandatory. • Interaction Field. Show the field in the Interaction section of the Add New Component dialog box and Interaction tab. <p>For details, see the "New Application Component Dialog Box" on</p> |

| UI Elements | Description |
|-------------|-------------|
| | page 47. |

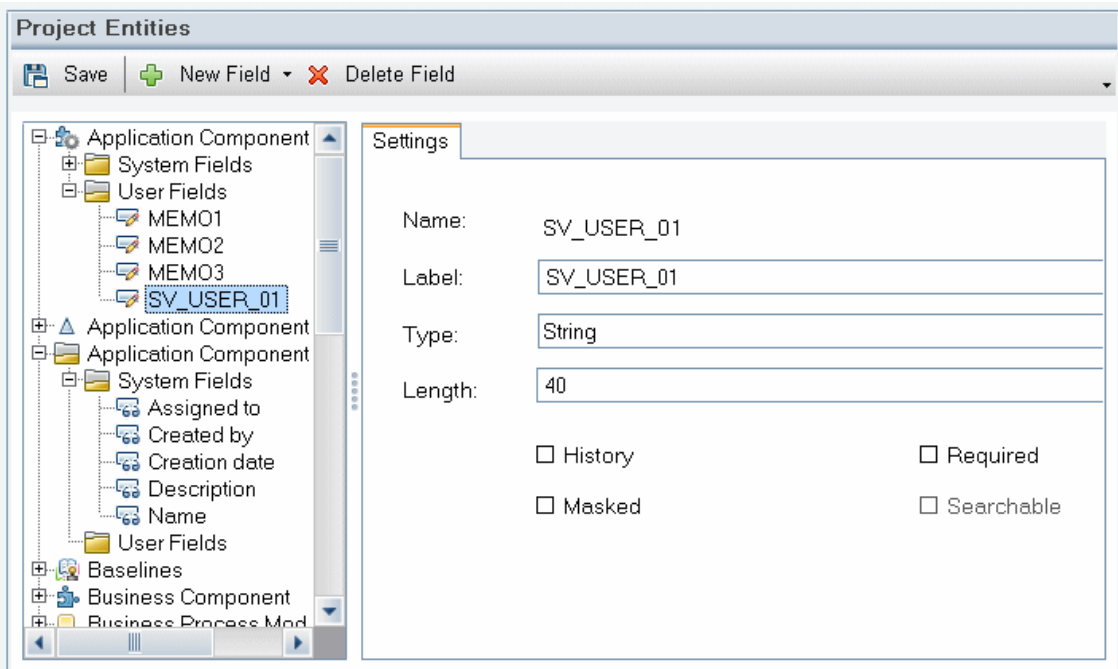
Project Entities Pane

This pane lists the 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.** Customizable user-defined fields. You can add, modify, and delete user-defined fields.



For detailed information on ALM entities and fields, refer to the *HP ALM Database Reference*.

This dialog box enables you to run a spot test for an individual operation. You can set up a request and view the SOAP response to verify if an individual operation is functional.



| | |
|------------------------------|--|
| To access | Do the following: <ol style="list-style-type: none"> 1. Select Tools > Customize. 2. Select Project Entities in the left pane. 3. Expand the Application Component, Application Component Group, or Application Component Change entities. |
| Important information | You can only create and delete User fields, but not System fields. For System fields, you can modify the label and select some additional options, depending on the field. |
| Relevant tasks | "How to Customize a Project" on page 244 |
| See also | "Modeling Tab" on page 39 |

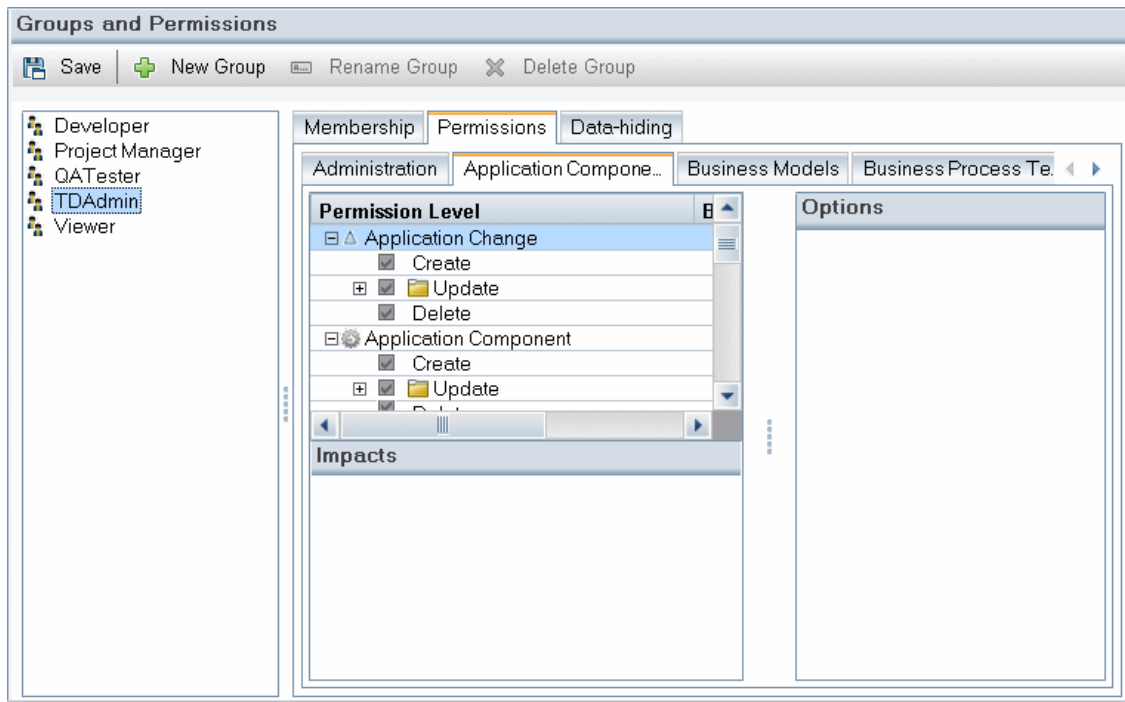
User interface elements are described below and in greater detail in the *HP ALM User Guide*.


| UI Elements | Description |
|--|---|
|  New Field ▾ | Opens a drop down list allowing you select New Field or a New Memo Field . |
|  Delete Field | Removes the selected user-defined field. |
| Name | A read-only field name used in the ALM database table. |
| Label | The name of the field to display in the Project Entities tree and the component's Details tab. This label is referenced anywhere the field is used, such as in the filter criteria, error messages, and graphs. |
| Type | The type of data valid for the field: <ul style="list-style-type: none"> • Number. Enables integer entry only. • String. Enables the entry of any character string. • Lookup List. Displays the Lookup List area and enables the selection from a drop-down list. |

| UI Elements | Description |
|------------------------------|---|
| | <ul style="list-style-type: none"> • User List. Enables the selection of a user name from your ALM users list. • Date. Enables the selection of a date. • Memo. Enables the entry of blocks of data. Note that you can add up to 5 memo fields to each ALM entity (available by selecting New Field > New Memo Field from the toolbar). |
| Length | <p>The field size (available for String type only).</p> <p>Note: The maximum field length is 255 characters.</p> |
| Goto List | <p>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 and click the Goto List button.</p> |
| History | <p>Preserves a log of values entered in the selected field.</p> |
| Masked | <p>The input data mask for the field (available only when the String type is selected).</p> |
| New List | <p>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.</p> |
| Required | <p>Indicates that a user must enter a value for the field.</p> |
| Searchable | <p>Indicates a searchable field (available only when the Text Search option is enabled in the DB Servers tab).</p> |
| Lookup List | <p>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.</p> |
| Verify value | <p>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).</p> |
| Allow multiple values | <p>Allows the user to select more than one value in any field that is associated with a predefined lookup list (available only when selecting a Lookup List type field).</p> |






Groups and Permissions

This customization section lets you define new user groups and set their permissions.



| | |
|-------------------------------------|---|
| <p>To access</p> | <p>Do the following:</p> <ol style="list-style-type: none"> 1. Select Tools > Customize. 2. Click Groups and Permissions in the left pane. 3. Select a group and click the Permissions tab. 4. Select the Application Components tab (the subtab names are in alphabetical order). |
| <p>Important information</p> | <p>To resume work in the ALM modules, click the  button in the top right corner.</p> |
| <p>Relevant tasks</p> | <p>"How to Customize a Project" on page 244</p> |

The **Application Components** tab displays the following permissions for elements available in the **Application Components** module:

| Element / Entity | Permission Level |
|--|---|
|  New Group | Opens the New Group dialog box for creating a new permission group. |
|  Rename Group | Allows you to rename the selected user-defined group. |
|  Delete Group | Removes the selected user-defined group. |
|  | <p>Alert. This icon indicates that selecting or clearing the current permission, will impact other permissions. The mouseover message provides the impact details.</p>  |
| <p>Permission Levels - Application Change</p> | <p>Lets you set the group's permissions for handling changes in an application component. You can enable or disable the following capabilities:</p> <ul style="list-style-type: none"> • Create. Create a new change. • Update. Update fields in an existing change. For details, expand the node and see "Application Change Details Dialog Box" on page 287 • Delete. Delete a change. |
| <p>Permission Levels - Application Component</p> | <p>Lets you set the group's permissions for creating and modifying application component details. You can enable or disable the following capabilities:</p> <ul style="list-style-type: none"> • Create. Create a new change. • Update. Update fields in an existing component. For details, expand the node and see "New Application Component Dialog Box" on page 47. • Delete. Delete a component. • Link to Components. Link to another application component. • Link to Groups. Link a component to an Application Component group. • Link to Requirements. Link a component to existing requirements. |

| Element / Entity | Permission Level |
|---|---|
| | <ul style="list-style-type: none"> • Update Contract. Update a component's contract from its source. |
| <p>Permission Levels - Folder</p> | <p>Lets you set the group's permissions for creating and removing folders in the application components tree. You can enable or disable the following capabilities:</p> <ul style="list-style-type: none"> • Create. Create a new folder. • Update. Update fields for an existing folder. The available fields are: Attachment, Description, Father ID, Folder ID, Logical Path, Name, and UDF. • Delete. Delete a folder. |
| <p>Permission Levels - Group</p> | <p>Lets you set the group's permissions for creating and modifying Application Component groups. You can enable or disable the following capabilities.</p> <ul style="list-style-type: none"> • Create. Create a new group. • Update. Update fields in an existing group. For details, expand the node and see "New Group Dialog Box" on page 174. • Delete. Delete a group. • Link to Requirements. Link a group to existing requirements. • Update Contracts. Update the contracts in a group's components from their source. |
| <p>Permission Levels - Customize</p> | <p>Provides the members of the group access to the Application Components section in Customization.</p> |

Module Access Pane

This pane lists the module access per group. You can control access for the various ALM modules.

For detailed information on module access, refer to the *HP ALM Database*

| | |
|-----------------------|---|
| To access | Select Tools > Customize and click Module Access in the left pane. |
| Relevant tasks | "How to Customize a Project" on page 244 |

The Module Access pane contains a table indicating each group's permission per module.

| Users Groups | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> De | <input checked="" type="checkbox"/> Te | <input checked="" type="checkbox"/> Te | <input checked="" type="checkbox"/> Re | <input checked="" type="checkbox"/> Da | <input checked="" type="checkbox"/> Co | <input checked="" type="checkbox"/> Rel | <input checked="" type="checkbox"/> M | <input checked="" type="checkbox"/> L | Application Components |
|-----------------|-------------------------------------|--|--|--|--|--|--|---|---------------------------------------|---------------------------------------|-------------------------------------|
| TDAdmin | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| QATester | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| Project Manager | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| Developer | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| Viewer | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |

Selecting the check box in the header row, checks all rows in that column.

Clearing the check box in the header row, clears all rows in that column.

For more details about Module Access permissions, see the *HP Application Lifecycle Management User Guide*.

Change Analysis

Concepts

- "Change Analysis Overview " on page 269
- "Change Notifications" on page 270
- "Reporting Changes Manually " on page 271
- "Automatic Change Detection Through Updates" on page 272
- "Contract Comparison" on page 273

Tasks

- "How to Perform Change Analysis " on page 276

Reference

- "Changes User Interface" on page 281
 - "Changes Tab" on page 282
 - "New Application Change Dialog Box " on page 284
 - "Application Change Details Dialog Box " on page 287
 - "Create Change Impact Test Set Wizard" on page 292
 - "Tests Tree Panel " on page 293
- "Troubleshooting and Limitations - Changes" on page 295

Concepts

This section includes:

- "Change Analysis Overview " on page 269
- "Change Notifications" on page 270
- "Reporting Changes Manually " on page 271
- "Automatic Change Detection Through Updates" on page 272
- "Contract Comparison" on page 273

Change Analysis Overview

Change impact analysis helps you assess the impact of changes to the application components. For each change, Service Test Management displays a description of the change, an assessment of the change-risk, and the change status.


Service Tests Management provides an interface for comparing different versions of document literal Web services and related XML schemas. You can also introduce changes manually and specify the change details. If you know of changes in the component's contract, you manually insert a new change and provide its details.

This section includes:

- ["Change Notifications" on page 270](#)
- ["Reporting Changes Manually " on page 271](#)
- ["Automatic Change Detection Through Updates" on page 272](#)
- ["Contract Comparison" on page 273](#)

Change Notifications

Service Test Management also generates a change list automatically. When you update an application component or group, it compares the latest definition with the definition stored in ALM. If an operation was added or removed, or if there were changes to the operation's data structure or at the signature level, Service Test Management lists them in the **Changes** tab.

Service Test Management flags changed application components, groups, or folders (prior to version 11.00), in the Application Components tree with an **Alert** icon .

Tip: A green asterisk on the **Changes** tab indicates a new change.

To verify the changes, you can generate a test set. These test sets can contain all tests linked to the updated component, or only tests that use the operations affected by the changes.

Each test set must be linked to a requirement. If a component does not have requirement coverage, Service Test Management lets you link it to a default requirement.

After creating or generating change-based requirements and a test set, you can run the linked or affected tests in the test set to verify whether these changes caused regression in functionality or performance. You can then determine which entities were affected and who is responsible for them.

To determine if a change violates a rule associated with a component, see the ["Rules Tab" on page 206](#).

Reporting Changes Manually

Service Test Management allows you to define changes manually. This change can indicate an update in the contract file or other changes that affected the contract file.

You can also add descriptions and comments, and assign the change to developers or QA personnel. For details, see "[New Application Change Dialog Box](#)" on page 284.

Automatic Change Detection Through Updates

You can use the **Update Component** feature to automatically check for changes in a component. If changes are found in the Web service:

- the component's definition is updated in Service Test Management.
- the Application Components tree displays an alert adjacent to the component.
- the **Changes** tab lists the changes.

You can update the definition from the original location or from a custom location.

For application components that reside on secure locations or those accessed through proxy servers, you specify the authentication credentials. For details, see the "[Secure Services and Proxy Servers](#)" on page 64.

For task details, see "[How to Perform Change Analysis](#)" on page 276.

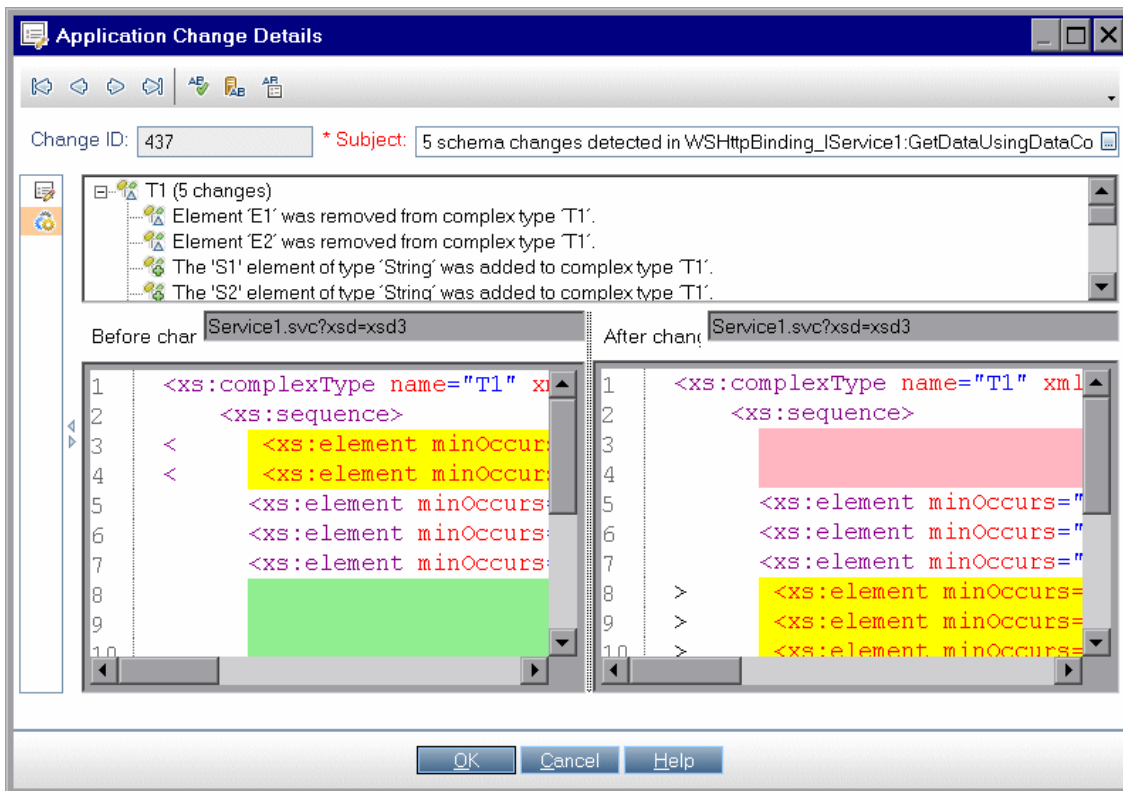
Note:

- When updating a Web service from a UDDI or Systinet server, it must have the same name as the one in the original WSDL used to import the service.
- If **Keep up to date** is set to `Yes` in the **Details** tab, the component definition is automatically updated from the original source each time it is loaded.

Contract Comparison

The **Technical Details** view of the Application Change Details dialog box, displays the contract files side by side, allowing you to visually compare them. The window highlights the differences between XSD schema versions for the selected change, using different colors for additions, deletions, and changes.

Each node indicates the nature of the change, for example, The 'xyz' element was removed from 'My operation'. The comparison window displays the changes for the selected node. If the changes were in the input or output properties, it indicates this in the **Subject** box.



For user interface details, see "Application Change Details Dialog Box - Technical Details Tab" on page 290.

By studying the changes, you can decide how to set the **Change Status** in the Application Change Details dialog box's **Details** view. You can set the status to **In test**, **Verified**, or **Obsolete**. For details, see "Application Change Details Dialog Box - Details Tab" on page 288.

To verify the functionality of the component after the changes, you can run an existing test set, or create a new one. To create a new test, click the **Create**

Test Set button in the **Changes** tab. For details, see "[Create Change Impact Test Set Wizard](#)" on page 292.

Tasks

This section includes:

- ["How to Perform Change Analysis "](#) on page 276

How to Perform Change Analysis


This task describes how to create and review the changes for your application component.

This task includes the following steps:

- Define a New Change - optional
- Update a Component - optional
- Review the changes
- View and/or modify the change details
- View the technical details
- Evaluate the tests
- Create a regression test set - optional
- Run the test sets
- Check the test status
- Update the change status

1. Define a New Change - optional

To manually define a change in the Changes list:

- a. In the Application Components tree, select a component.
- b. Select the **Changes** tab and click the New Change button  .
- c. In the New Application Change dialog box, make sure that the following mandatory fields have values:
 - **Subject.** A meaningful name for the change.
 - **Affected part.** The name of the affected application component part.
 - **Affected part type.** The type of part affected by the change:
Component, Field, Interface, or Operation.
 - **Change Status.** The change's current status: In Test, New, Obsolete, or Modified.
 - **Change type.** The nature of the change: Added, Updated, or

- **Risk.** The risk of the change: `High`, `Medium`, or `Low`.
- d. Enter values into the optional fields, such as **Description**, **Assigned to**, and so forth.


2. Update a Component - optional

To automatically generate a list of changes, update the component contract stored in Application Lifecycle Management with the latest contract file. Service Test Management flags the changed component in the Application Components tree.


- a. In the Application Components tree, select the Web service component you want to update.
- b. Choose one of the following update options:
 - To update a component using the same contract location, choose **Components > Update Component > Update Component**.
 - To update a component using a different location, choose **Component > Update Component > Update Component from**. Browse for a newer version of the contract in another location.

The update process begins. Service Test Management checks for changes in the contract and informs you if the component was updated.

3. Review the changes

Select an application component that has been updated, indicated by an alert icon . Click the **Alert** icon to open the Alerts dialog box and view a list of the alerts. Click **Show** to open the selected change in the **Changes** tab. The grid lists the changes that were made to the component.

4. View and/or modify the change details

- a. Select a change in the grid, and click the **Change Details** button .
- b. Click the **Details** link in the left pane (selected by default).
- c. View and modify the field values.

For details, see the "[Application Change Details Dialog Box - Details Tab](#)" on page 288.

5. View the technical details

- a. Click the **Technical Details** link in the left pane.

- b. Expand the changes tree in the upper pane and select a change. The bottom pane highlights the change within the schema.
- c. Click **OK** to close the Application Change Details dialog box.

For details, see the ["Application Change Details Dialog Box - Technical Details Tab"](#) on page 290.

6. Evaluate the tests

Evaluate the tests that are associated with the component in the **Testing Status** tab, and add additional tests if necessary. For more information about tests linked to components, see ["How to Check the Testing Status"](#) on page 199.

7. Create a regression test set - optional

To analyze the impact of the changes, click **CreateTest Set** in the **Changes** tab, to invoke the wizard. For details, see the ["Create Change Impact Test Set Wizard"](#) on page 292.

Tip: The **Create Change Impact Test Set** wizard does not allow you to enter values for required Test Set fields. To enter values for these fields, open the **Test Lab** module, go to the test set, and enter values for the required fields.

8. Run the test sets


Run the test sets from the **Test Lab** module. You can verify the component behavior and analyze the test results using ALM graphs, reports, and documents. For details on running test sets and analyzing test results, refer to the *HP Application Lifecycle Management User Guide*.

9. Check the test status

Review the coverage status of the parent requirement created for the changes set in the **Testing Status** tab, to verify that the coverage includes the latest changes. For details, see ["Requirements Coverage"](#) on page 192.

10. Update the change status

Verify the behavior of the changed component and manually update the change status in the **Changes** tab through the following steps:

- a. Select an updated component in the Application Components tree, and click the **Changes** tab.
- b. In the list of changes, select the change whose status you want to update.
- c. Select a change in the grid, and click the **Change Details** button .
- d. In the **Details** tab's **Change Status** box, set the change status. For example:
 - **Verified.** if the change was tested and did not cause a regression.
 - **Obsolete.** if the change does not affect the component.

For details about change status, see the "[Application Change Details Dialog Box](#)" on page 287.

Reference

This section includes:

- "Changes User Interface" on page 281
- "Changes Tab" on page 282
- "New Application Change Dialog Box " on page 284
- "Application Change Details Dialog Box " on page 287
- "Create Change Impact Test Set Wizard" on page 292
- "Tests Tree Panel " on page 293

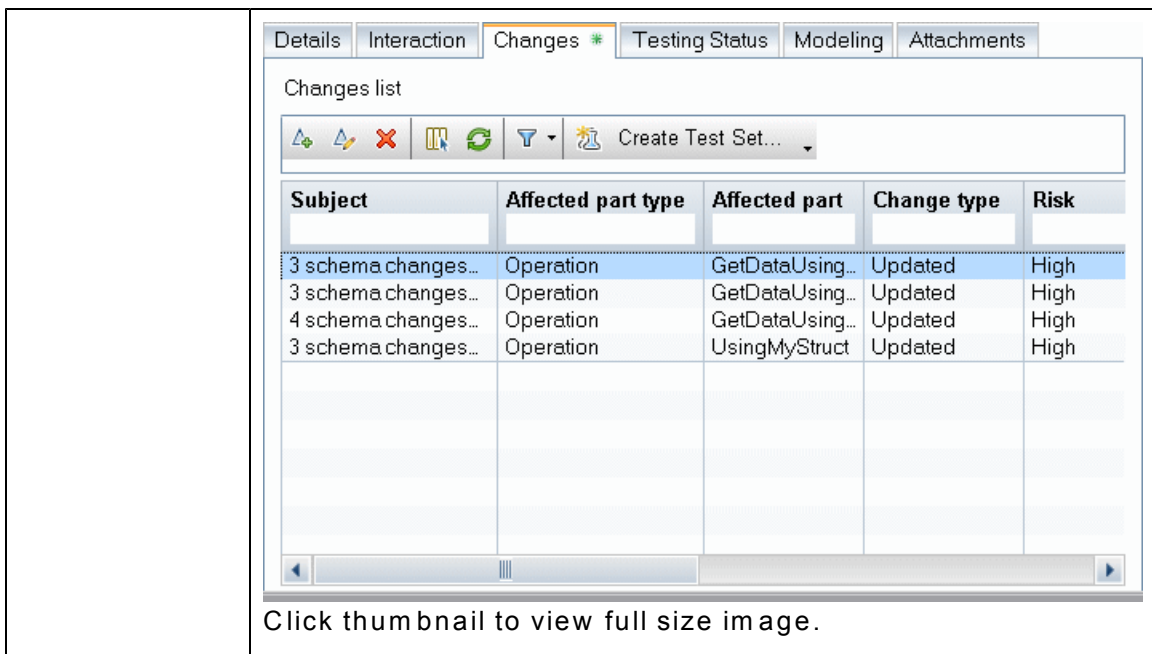
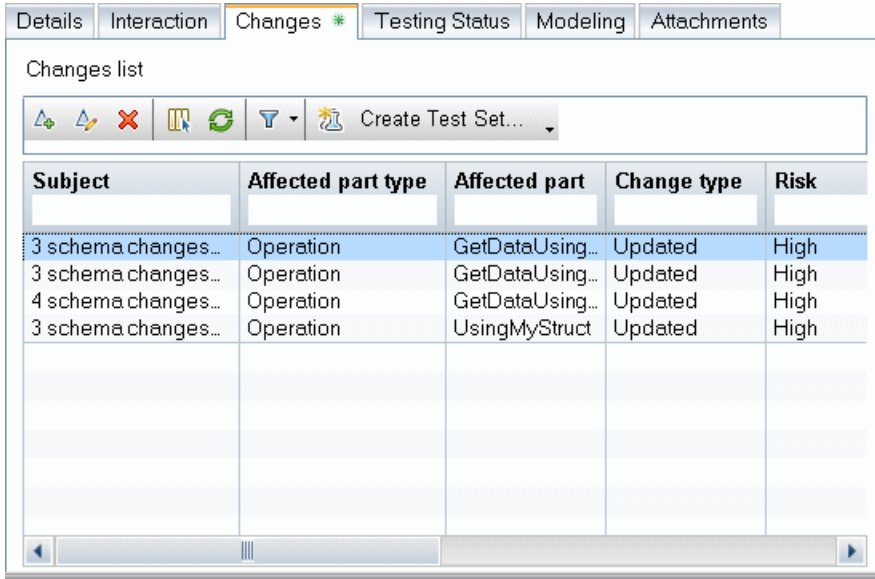
Changes User Interface

This section includes:

- "Changes Tab" on page 282
- "New Application Change Dialog Box " on page 284
- "Application Change Details Dialog Box " on page 287
- "Create Change Impact Test Set Wizard" on page 292
- "Tests Tree Panel " on page 293







Changes Tab

This tab allows you to view and manage the changes made to the application component.



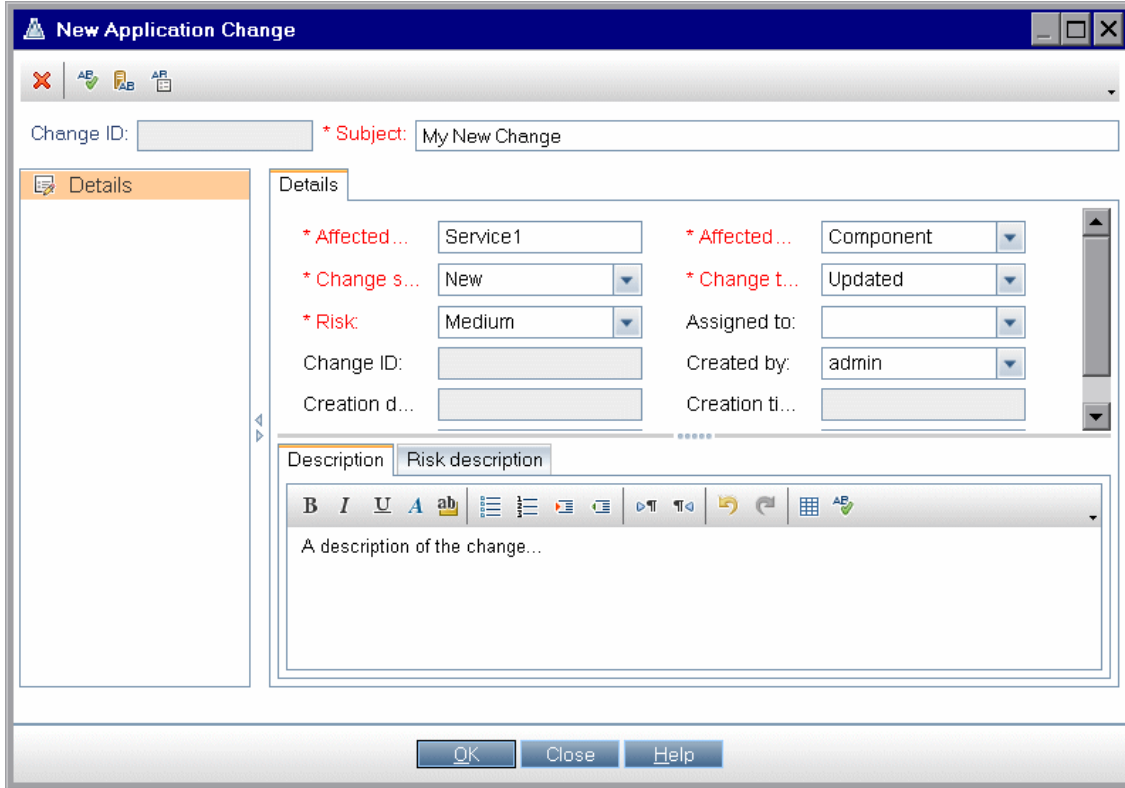
| | |
|-----------------------|---|
| To access | Do the following: <ol style="list-style-type: none"> 1. Select the Application Components module. 2. Select an entity in the Application Components tree. 3. Click the Changes tab. |
| Relevant tasks | "How to Perform Change Analysis " on page 276 |


User interface elements are described below (Unlabeled UI elements are shown in angle brackets>):

| UI Elements | Description |
|---|--|
|  | New Change. Opens the New Application Change dialog box to create a new change upon your component. See the " New Application Change Dialog Box " on page 284. |
|  | Change Details. Opens the Application Change Details for viewing and editing. See the " Application Change Details Dialog Box " on page 287. |
|  | Delete. Removes a change from the Changes grid. |
|  | Select Columns. Enables you to select the columns to display in the Changes grid. For a description of the columns, see the " Application Change Details Dialog Box " on page 287. |
|  | Refresh. Reloads the list of changes. |
|  | Set Filter/Sort. Enables you to set a filter for the changes or sort the changes by a criteria. |
| Create Test Set | Opens a wizard for creating a test set. For details, see " Create Change Impact Test Set Wizard " on page 292. |
| <Changes grid> | A list of all the changes for the selected component to which the filter applies. To clear the filter, use the Set/Filter sort button. To arrange the columns, use the Select Columns button. Default: Subject, Affected part type, Affected part, Change type, Risk, Change status, Creation date, Assigned to, Release, and Cycle. |



New Application Change Dialog Box

This dialog box lets you define a new change for your application component.



| | |
|-----------------------|---|
| To access | Select the Changes tab and click the New Change  button. |
| Relevant tasks | "How to Perform Change Analysis " on page 276 |
| See also | "Changes Tab" on page 282 |

User interface elements are described below:

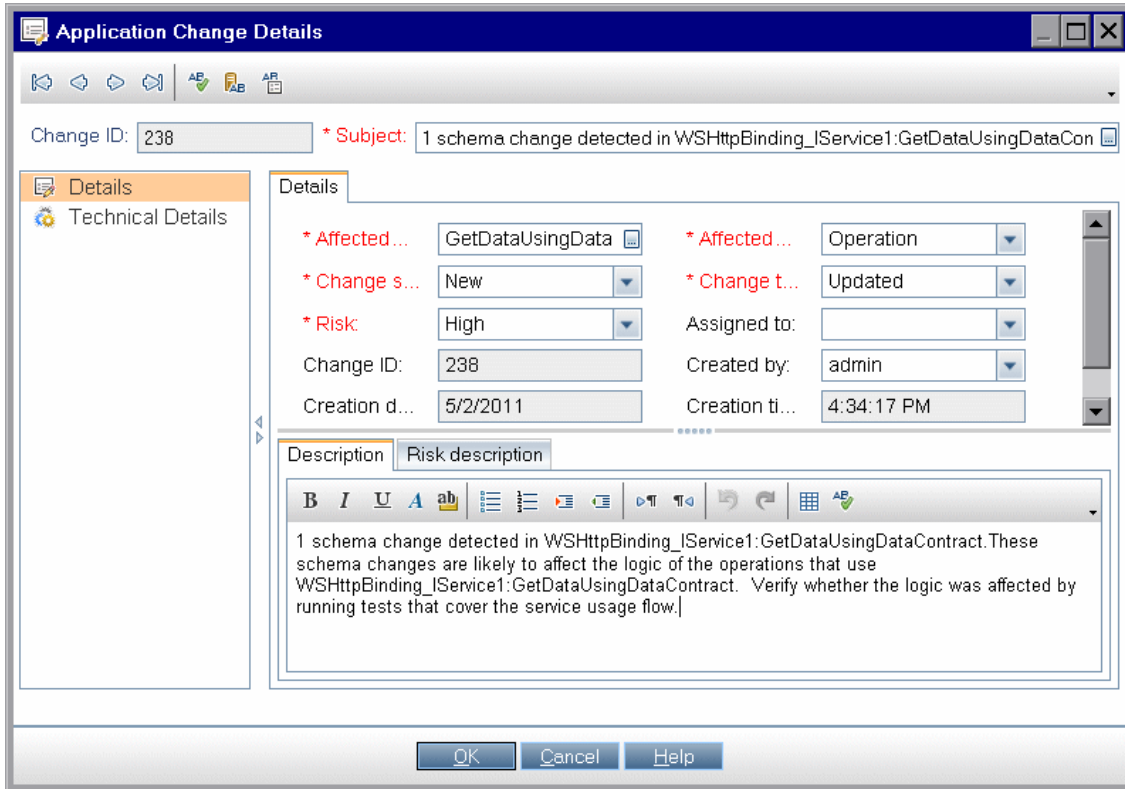
| UI Elements (A-Z) | Description |
|---|---|
|  | Clear all Fields. Clears all the fields in the dialog box. |
|  | Editing buttons: Spell Check , Thesaurus , and Spelling Options . For details, see the <i>HP Application Lifecycle Management User Guide</i> . |

| UI Elements (A-Z) | Description |
|---------------------------|---|
| Affected part | The name of the part affected by the change, such as the component name or operation name. |
| Affected part type | The part type affected by the change: Component , Field , Interface , or Operation . |
| Assigned to | The user name of the person who is responsible for the change. |
| Change ID | A unique identification string for the change. This field is read-only and is generated when you save a new change. This is visible in the Application Change Details dialog box. |
| Change status | <p>The current status of the change. By default, the change status is New. Update the status throughout the change analysis process as follows:</p> <ul style="list-style-type: none"> • In test. The change is being tested. • New. There is a change in the component. • Obsolete. The change does not affect the component. • Verified. The change was tested and did not cause a regression. |
| Change type | <p>The type of change is indicated as follows:</p> <ul style="list-style-type: none"> • Added. New functionality was added. • Removed. Functionality was removed. • Updated. Changes were made to existing functionality. Details of these changes are displayed in the Description column. |
| Created by | The user name of the person who created the changes. |
| Creation date | The date on which the change was made. By default, the creation date is set to the current database server date. This field is read-only and is generated when you save a new change. |

| UI Elements (A-Z) | Description |
|-------------------------|---|
| Creation time | <p>The time at which the change was made. This field is read-only and is generated when you save a new change.</p> <p>Default: The current database server time.</p> |
| Cycle | <p>The testing cycle in which the change was introduced or in which the change needs to be verified.</p> |
| Description | <p>A description of the change and suggested tests to verify them.</p> <p>For operation changes (where Part Type is Operation and Change Type is Updated), the change description contains details of the argument that was added or removed from the operation.</p> |
| Release | <p>The planned release version.</p> |
| Risk | <p>The business risks and costs associated with the change. A change can have one of the following risk levels:</p> <ul style="list-style-type: none"> • High. Changes that are likely to affect functionality and fail any test that is not updated. High-risk changes include operation removal and parameter changes. • Medium. Changes that may affect functionality, but may not necessarily fail existing tests. Medium-risk changes include component changes (notification that some of the component files have changed). • Low. Changes that are unlikely to fail or affect existing functionality or implemented tests. Low-risk changes include operation additions (as they do not affect existing functionality), service address changes (deployment only), and port type changes. |
| Risk Description | <p>A textual description of the risk.</p> <p>Tip: Click Add Comment to create a remark with a name, time, and date stamp.</p> |
| Subject | <p>A short title describing the change. This field is mandatory.</p> |


Application Change Details Dialog Box

This dialog box enables you to view and manage the changes applied to the component. You can view both the general and technical details of the change.







This sections contains the following information:

Application Change Details Dialog Box - General Information

| | |
|-----------------------|---|
| To access | Select the Changes tab and click the Change Details  . |
| Relevant tasks | "How to Perform Change Analysis " on page 276 |
| See also | "Changes Tab" on page 282. |

User interface elements are described below:

| UI Elements (A-Z) | Description |
|---|--|
|  | <p>Change Navigation. Enables navigation between the changes for display in the Application Change Details dialog box. The buttons perform the following actions, from left to right:</p> <ul style="list-style-type: none"> • Go to first change • Go to previous change • Go to next change • Go to last change |
|  | <p>Editing buttons: Spell Check, Thesaurus, and Spelling Options. For details, see the <i>HP Application Lifecycle Management User Guide</i>.</p> |
| Change ID | A unique ID for the change (read-only). |
| Subject | A short title describing the change. |
|  | Details tab. For details, see below. |
|  | Technical Details tab. For details, see " Application Change Details Dialog Box - Technical Details Tab " on page 290. |

Application Change Details Dialog Box - Details Tab

The **Details** tab allows you to view and modify all of the fields describing the change. User interface elements are described below:

| UI Elements(A-Z) | Description |
|---------------------------|---|
| Affected part | The component part affected by the change. |
| Affected part type | The part affected by the change: Component , Field , Interface , or Operation . |

| UI Elements(A-Z) | Description |
|------------------------|---|
| Assigned to | The user name of the person responsible for the change. |
| Change ID | A unique ID for the change (read-only). |
| Change status | <p>The current status of the change. By default, the change status is New. Update the status throughout the change analysis process as follows:</p> <ul style="list-style-type: none"> • In test. The change is being tested. • New. There is a change in the component. • Obsolete. The change does not affect the component. • Verified. The change was tested and did not cause a regression. <p>Tip: To eliminate the Alert icon next to the component, change the status of all changes to Verified or Obsolete.</p> |
| Change type | <p>The type of change is indicated as follows:</p> <ul style="list-style-type: none"> • Added. New functionality was added. • Removed. Functionality was removed. • Updated. Changes were made to existing functionality. Details of these changes are displayed in the Description column. |
| Created by | The user name of the person who ran the update that created the change. |
| Creation date | The date on which the change was made. By default, the creation date is set to the current database server date. |
| Creation time | <p>The time at which the change was made.</p> <p>Default: the database server time at the time of the change.</p> |
| Cycle | The testing cycle in which the change was introduced or in which the change needs to be verified. |
| Description tab | A description of the change and suggested tests to verify. |

| UI Elements(A-Z) | Description |
|-----------------------------|---|
| | For operation changes (where AffectedPart Type is Operation and Change Type is Updated), the change description contains details of the argument that was added or removed from the operation. |
| Release | The planned release version. |
| Risk | <p>The business risks and costs associated with the change. A change can have one of the following risk levels:</p> <ul style="list-style-type: none"> • High. Changes that are likely to affect functionality and fail any test that is not updated. High-risk changes include operation removal and parameter changes. • Medium. Changes that may affect functionality, but may not necessarily fail existing tests. Medium-risk changes include changes (notification that some of the component files have changed). • Low. Changes that are unlikely to fail or affect existing functionality or implemented tests. Low-risk changes include operation additions (as they do not affect existing functionality), service address changes (deployment only), and port type changes. |
| Risk Description tab | <p>A textual description of the risk.</p> <p>Tip: Click Add Comment to create a remark with a name, time, and date stamp.</p> |

Application Change Details Dialog Box - Technical Details Tab

The Technical Details tab lets you compare the contract files and understand the changes. For details, see "[Contract Comparison](#)" on page 273.

User interface elements are described below:

| UI Elements(A-Z) | Description |
|----------------------|---|
| <change data> | <ul style="list-style-type: none"> • Main branch. A hierarchal display of the changes in the schema. The changes are grouped by their type. For example, all removal type changes are grouped together. • Affected derived types. A sub-node showing the schemas affected indirectly by the changes. <p>When you expand the tree and select a change, the comparison window highlights the affected lines in the original and current schemas.</p> <p>The tree displays changes in the input and output properties in separate nodes.</p> |
| Before change | The name of the schema file before the change. |
| After change | The name of the schema file after the change. |
| <original schema> | A snapshot of the original XML, showing the changed type or element. |
| <changed schema> | <p>A snapshot of the changed XML, showing the modified fragment. The snapshot highlights the modified lines using the following color scheme:</p> <ul style="list-style-type: none"> • Yellow. Changed entity • Green. Added entity • Pink. Removed entity |

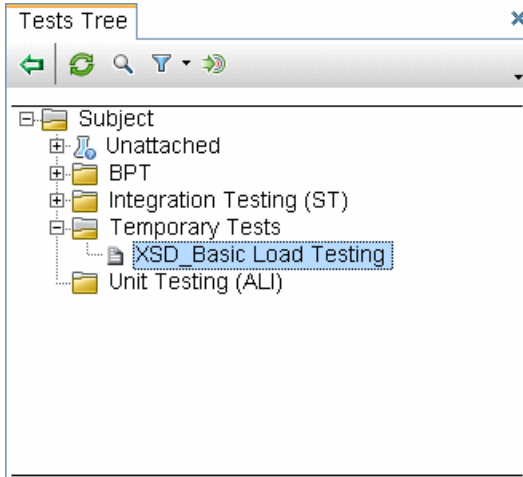
Create Change Impact Test Set Wizard

This wizard enables you to create a test set for determining the impact of changes made to an application component.

| | |
|-----------------------|---|
| To access | Select Changes tab > Create Test Set . |
| Relevant tasks | " How to Perform Change Analysis " on page 276 |
| Wizard map | The Create Change Impact Test Set wizard contains: Coverage Page > Test Set Page |



Tests Tree Panel





This tab allows you to add tests to a test set.



| | |
|-----------------------|--|
| To access | <p>Do the following:</p> <ol style="list-style-type: none"> 1. Select a component and click the Changes tab. 2. Select a change in the list and click the Create Test Set button. 3. In the Create Change Impact Test Set wizard, specify the coverage details. 4. Click Next to advance to the Test Set wizard page. 5. Click the Select button. |
| Relevant tasks | <p>"How to Perform Change Analysis " on page 276</p> |
| See also | <p>"How to Link Components to Requirements and Tests" on page 224</p> |

The following elements are included (unlabeled UI elements are shown in angle brackets>):

| UI Elements | Description |
|---|--|
|  | Add to Coverage. Adds the selected test to the test list. |
|  | Refresh. Refreshes the Tests tree. |

| UI Elements | Description |
|---|---|
|  | Find. Opens the Find dialog box for locating a test by its name or another criteria. For details, click F1 in the Find dialog box. |
|  | <p>Set/Clear Filter. Opens a drop down list with the following options:</p> <ul style="list-style-type: none"> • Set Filter / Sort. Opens the Set Filter/Sort dialog box allowing you to filter and sort the tests. • Clear Filter / Sort. Clears all filters and sorting settings and displays all tests. |
|  | Go to Test by ID. Locates the test whose ID you specify. |
|  | <p>Closes the Tests Tree panel.</p> <p>To reopen this view, click the Select button.</p> |
| <Tests tree> | <p>A tree view of all of the tests.</p> <p>Tip: If a filter is active, there could be hidden components. To see all components, click Clear Filter.</p> |

Troubleshooting and Limitations - Changes

This section describes the limitations that affect change analysis.

- The XSD schema comparison detects changes in these common element types and attributes:
 - Element
 - ComplexType
 - Sequence
 - SimpleType
 - Simple and Complex Extensions
 - Enumeration (Restriction - Simple Type)
 - Attributes: Name, Type, Ref, MaxOccurs, MinOccurs, Nillable, Id, and targetNamespace
 - Include
 - Import
- The XSD Schema comparison tool does not detect:
 - If a derived type's data type or name was changed. It only detects changes that occur in a direct reference to a type— not those derived through inheritance.
 - Changes in RPC-encoded WSDLs.
 - Changes within elements that have the same name and context, such as **Sequence**.
- When defining a new change in the **Changes** tab, and selecting `Operation` as the **Affected part type**, the **Affected part** field does not provide a drop down list of the available operations.
Workaround: Manually type in the operation name.

Reports and Graphs

Concepts

- ["Service Test Management Reports and Graphs Overview"](#) on page 298

Tasks

- ["How to Create and Generate Reports"](#) on page 307
- ["How to Generate Graphs"](#) on page 309

Concepts

This section includes:

- "Service Test Management Reports and Graphs Overview" on page 298

Service Test Management Reports and Graphs Overview

ALM's **Dashboard** module provides you with several ways to generate summary reports.

Using the standard Document Generator, you can create summary reports of your test plan and requirements. Use the **Tools** menu to open the **Document Generator**. For details, see the HP Application Lifecycle Management User Guide.

This section describes the Excel reports that you can generate through ALM, that are specific to Service Test Management.

Using the capabilities of Excel, you can create data tables and display the results in graphical form. You can also create custom reports, by designing Excel-compatible SQL queries.

You can use the data you extract from a query for external applications, such as running a Visual Basic script to process and analyze the data.

This section also includes:

- Built-in Reports
- Application Component Query Builder Entities
- Application Component Graphs
- Application Changes Graphs

Built-in Reports

Service Test Management provides the following pre-defined reports for application components:

| Report Name | Description |
|--------------------------------------|--|
| Active Changes | Exports the project's application components and their list of active changes. |
| Application Component Defects | Exports the project's application components and the list of new or open defects which are linked to them. |
| Application Component | Exports the project's application components and the list of tests which either use or cover them. |

| Report Name | Description |
|--|---|
| Tests | |
| Application Components Per Responsible User | Exports the count of application components per a responsible user and displays this data as an Excel chart |
| Coverage | Exports the requirement coverage of each application component and the total coverage. The results are displayed in a table and in a pie chart. |
| Outstanding Changes | Exports the list of changes in the application components which are in the 'New' or 'In Test' status. |
| Rule Violations | Exports the list of all rules together with components which violated them. |
| Web Service Components by Toolkits | Exports the distribution of toolkits which are used by the Web services. |
| Web Service Components by Type | Exports the distribution of Web service import types: File, URL, UDDI, or Systinet. |

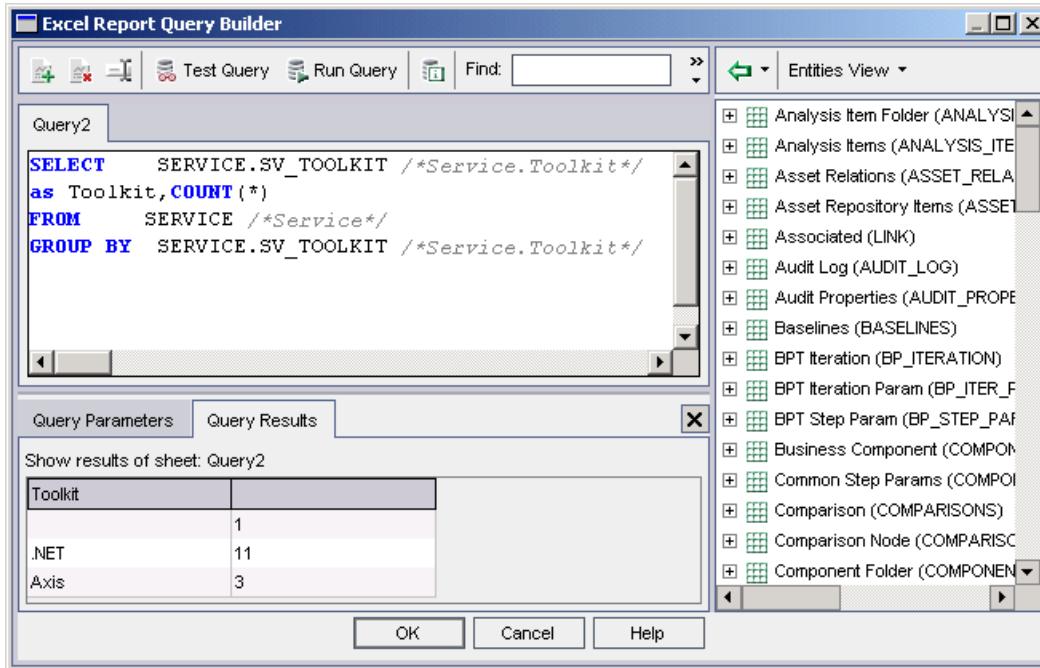
You can generate the reports as is, or modify them as necessary in the Query Builder. For details, see ["How to Create and Generate Reports"](#) on page 307.

Application Component Query Builder Entities

You can create custom queries on the project's database by choosing application component related entities from the Entities list. You can add these queries to an existing report, or create a new one.

Service Test Management adds the following entities to Query Builder's Entities View: **Component Change to Defects**, **Component Change to Requirements**, **Component Folder**, **Component Param**, **Component Step**, **Component Type**, **Component Type to Field Relation**, **Service Aspect**, **Service Dependency**, **Service Folder**, **Service Group to Requirement Relation**, **Service Operation to Component Relation**, **Service Operation to Test Relation**, **Service to Group Relation**, **Service to Requirement Relation**, **WSDL Operation**, **WSDL Operation**

Dependency, and **WSDL Operation Parameters**. You can create both private queries or public ones that you can share with other users.



For example, you could create a pivot table to summarize component-related data, such as to display in tabular format the status count for the different coverage statuses.

Once you generate an Excel report, you can use the full capabilities of Excel to display the results in graphical form, such as bar or pie charts.

For task details, see ["How to Generate Graphs" on page 309](#).

Application Component Graphs

You can create custom graphs for the Application Component module using Service Test Management's built-in graphs. You can create **Progress**, **Summary**, and **Trend** graphs for the application components and for the application component changes.

The graph's **Configuration** tab lets you customize the graphs by setting the x and y axis, filtering the data, and grouping the results.

Summary Graphs

In Application Component Summary graphs, you can configure the **x-axis** to display one of the following fields: Assigned to Dev, Assigned to QA, Authentication default credentials, Created by, Development status, Has

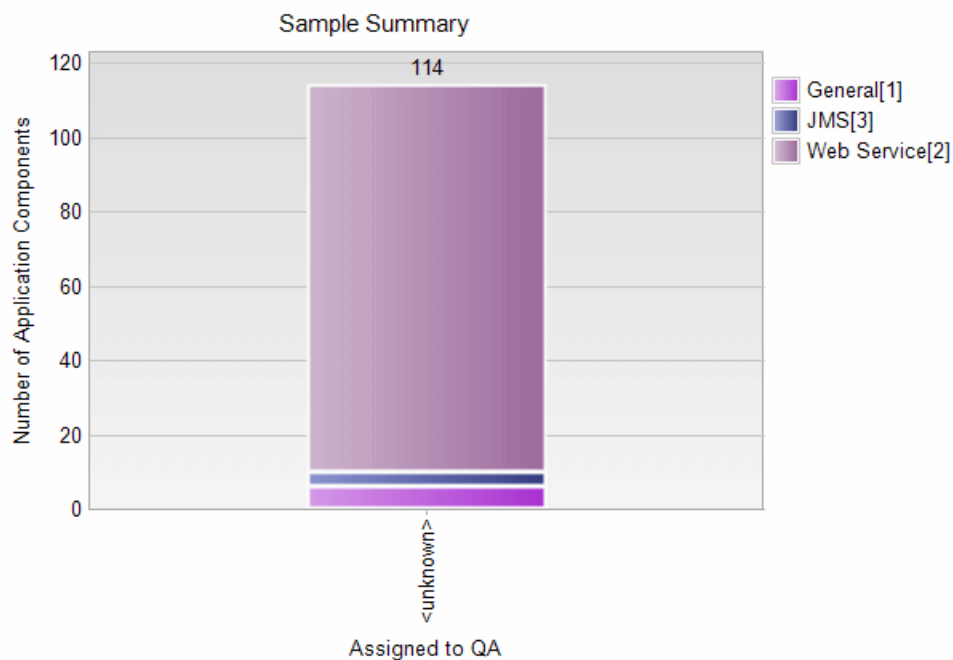
changed, Import type, JMS provider, JMS transport, JNDI initial context factory, Keep up to date, Messaging model, Override address, Proxy default credentials, Toolkit, Use authentication, Use proxy, Visibility, QC Project, or other fields that you defined in customization.

You can configure the **y-axis** to display

- A **Count** of the Application components
- A **Sum of** the custom fields, defined in the Customization module

Service Test Management also allows you to filter the graph by all system and UDF fields defined for the Application Component entity.

The following **Summary** graph shows the number of components assigned to QA as a function of the total number of components.



Progress Graphs

You can set up a **Progress** graph to plot the number of application components against a time interval.

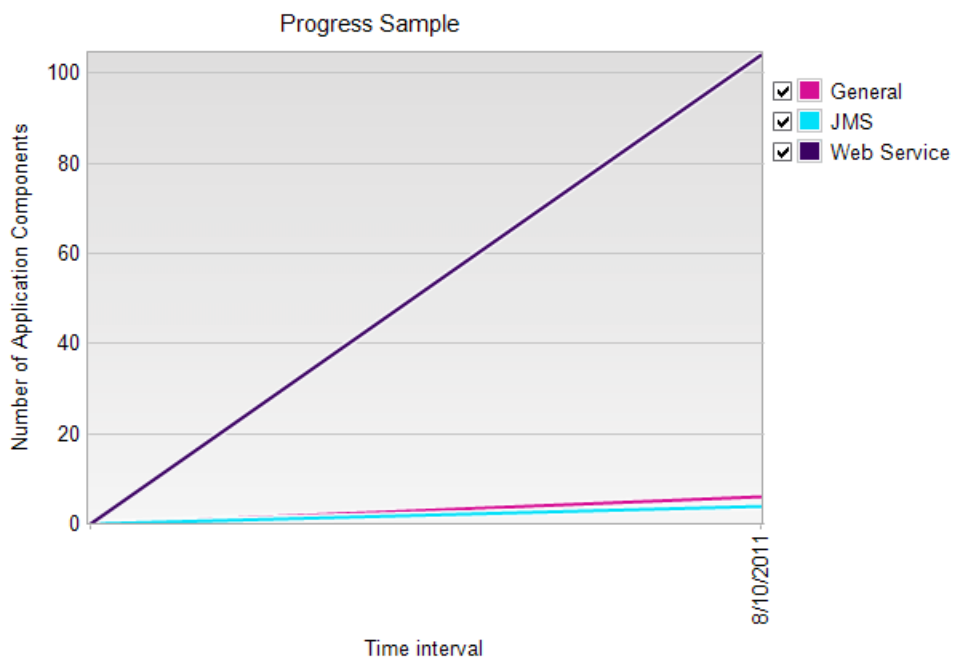
You can configure the **x-axis** to display a time interval: Last 7 days, Last 5 weeks, Last 12 months, or All days.

You can configure the **y-axis** to display

- A **Count** of the Application components
- A **Sum of** the custom fields, defined in the Customization module

You can group the results by the following criteria: Assigned to Dev, Assigned to QA, Authentication default credentials, Development status, Has changed, Import type, JMS provider, JMS transport, JNDI initial context factory, Keep up to date, Messaging model, Override address, Proxy default credentials, Toolkit, Use authentication, Use proxy, and Visibility.

The following **Progress** graph shows the time interval of the application components as a function of the total number of components. The colored lines indicate the various types of application components: **General**, **JMS**, **Web Service**, and so forth.



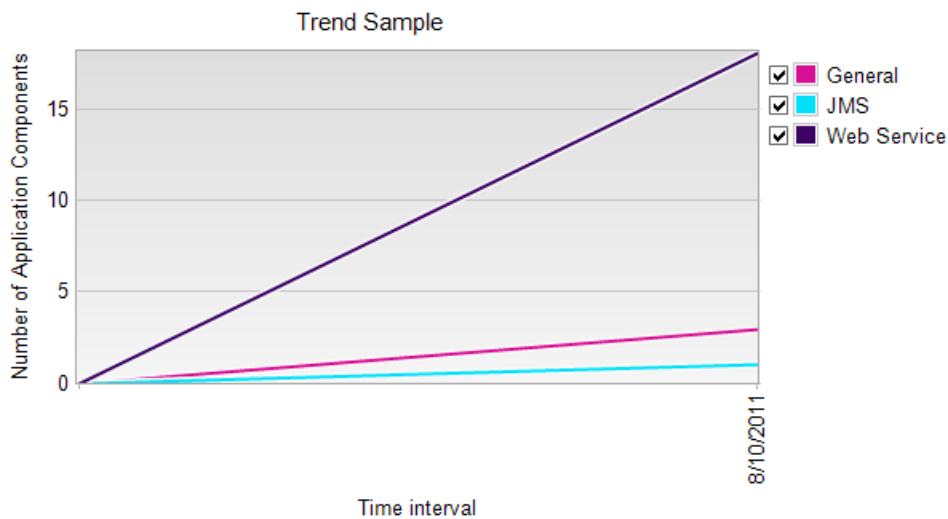
Trend Graphs

You can set up a **Trend** graph to plot the trend of application components against a time interval.

You can configure the time period to: Last 7 days, Last 5 weeks, Last 12 months, or All days.

You can configure the grouping to: All fields, Assigned to Dev, Assigned to QA, Authentication default credentials, Development status, Has changed, Import type, JMS provider, JMS transport, JNDI initial context factory, Keep up to date, Messaging model, Override address, Proxy default credentials, Toolkit, Use authentication, Use proxy, and Visibility.

The following **Trend** graph shows the different component types as a function of their creation dates.



Application Changes Graphs

You can create graphs to plot the application's changes using Service Test Management's built-in graphs. You can create **Progress**, **Summary**, and **Trend** graphs for the application components and for the application changes.

Summary Graphs

In Application Changes Summary graphs, you can configure the **x-axis** to display one of the following fields: Affected part type, Assigned to, Change status, Change type, Created by, Cycle, Release, Risk, and QC Project.

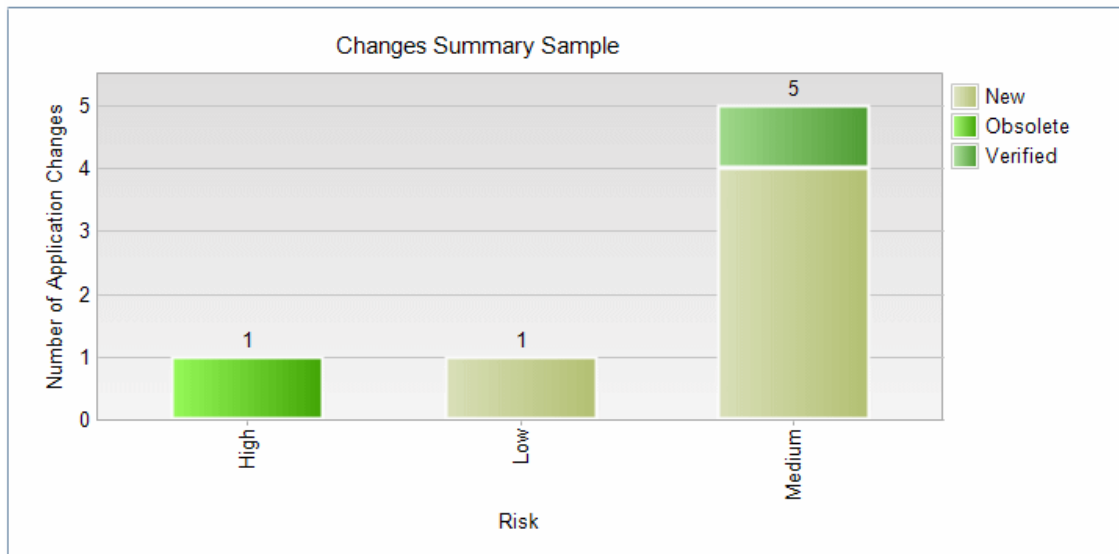
The **y-axis** displays the number (**Count**) of application changes.

You can filter the graph by all of the fields, including all system and UDF fields defined for the entity, such as Affected part type, Assigned to, Change ID, Change status, Change type, Created by, Creation

date, Creation time, Cycle, Release, Risk, Subject, or custom fields.

You can group the results by the following criteria: Affected part, Affected part type, Assigned to, Change status, Change type, Created by, Creation date, Creation time, Cycle, Release, Risk, or QC Project.

The following **ChangesSummary** graph shows the number of risks, grouped by the risk level, as a function of the total number of changes.



Progress Graphs

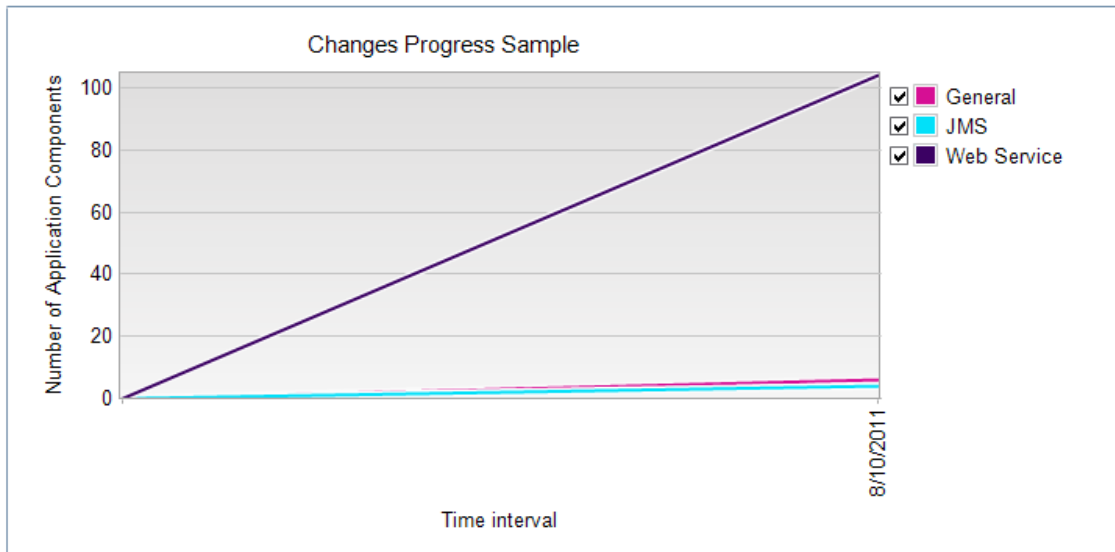
You can set up a **Progress** graph to plot the number of application changes against a time interval.

You can configure the **x-axis** to display a time interval: Last 7 days, Last 5 weeks, Last 12 months, or All days.

The **y-axis** displays the number (**Count**) of application changes.

You can group the results by the following criteria: Assigned to, Change status, Cycle, Release, or Risk.

The following **ChangesProgress** graph shows the time interval of the application components as a function of the total number of components. The colored lines indicate the various levels of risk: **High**, **Low**, and **Medium**.



Trend Graphs

You can set up a **Trend** graph to plot the trend of application changes against a time interval.

You can configure the time period to: Last 7 days, Last 5 weeks, Last 12 months, or All days.

You can configure the grouping to: All fields, Assigned to, Change status, Cycle, Release, and Risk.

Tasks

This section includes:

- "How to Create and Generate Reports" on page 307
- "How to Generate Graphs" on page 309

How to Create and Generate Reports

This task describes how to create reports using ALM's built-in reports for application components.

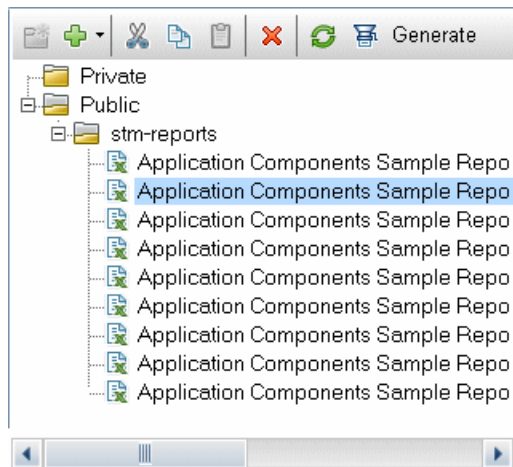
This task includes the following steps:

- Generate a pre-defined report
- Create a custom report based on a pre-defined report
- Create new Excel reports

Generate a pre-defined report

Use the **Dashboard** module to generate pre-defined Excel reports.

1. Open the **Dashboard > Analysis View** module.
2. Select a report from the **Public/stm-reports** folder.



3. Click **Generate**. Specify a location for the Excel report.

Create a custom report based on a pre-defined report

You can use an existing report as a basis for creating a custom report.

1. Open the **Dashboard > Analysis View** module.
2. Select a report from the **Public/stm-reports** folder.
3. To customize a built-in report without providing access to other users, copy it to the **Private** folder and rename it.

To customize a built-in reports and allow everyone to access your report, keep it in the Public folder.

4. Open the **Configuration** tab in the right pane, and click the **Query Builder** button.
5. Select an entity in the right pane's **Entities View** and expand it.
6. Drag the desired parameter to a location within the SQL statement. Modify the SQL as required.
7. Click **OK** to save it and close the **Query Builder**.
8. Click **Generate** and specify a location for the Excel report.

Create new Excel reports

You can design a new report without basing it on a pre-defined one.

1. Open the **Dashboard > Analysis View** module.
2. Click on the target folder and subfolder:
 - **Private**: Report is only available to current user.
 - **Public**: Report is available to all users of the project.
3. Choose **Analysis > New Excel Report**. Specify a name for the report.
4. Click **Query Builder**. ALM opens the **Entity View** in the right pane.
5. Select an entity in the right pane and expand it to view the parameters. For a list of entities, see "[Service Test Management Reports and Graphs Overview](#)" on page 298.
6. Drag a parameter to a location within the Query Builder editor. Modify the SQL as required.
7. Use the **Test Query** and **Run Query** button to test and run the query.
8. Click **OK** to save the report. You can open this report and run the query at a later time.

For details on how to customize the Excel reports, see the HP Application Lifecycle Management User Guide.

How to Generate Graphs

This task describes how to create graphs for your Application Components and the changes associated with them.

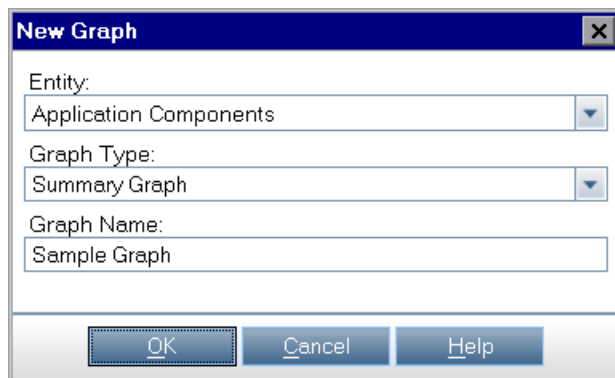
This task includes the following steps:

- Open the New Graph dialog box
- Configure the graph
- Select different projects - optional
- View the Graph

Open the New Graph dialog box

Use the **Dashboard** module to generate pre-defined graphs.

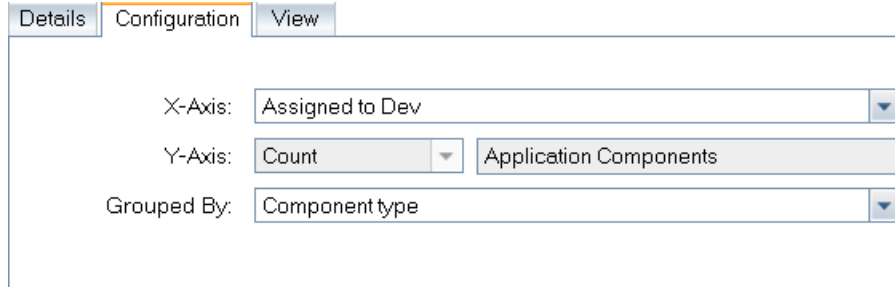
1. Open the **Dashboard > Analysis View** module.
2. Select **Analysis > New Graph (Ctrl+G)**.



3. Select an Entity - **Application Components** or **Application Changes**.
4. Select a graph type: **Summary Graph**, **Progress Graph**, or **Trend Graph**.
5. Specify a graph name and click **OK**.

Configure the graph

1. Click the **Configuration** tab in the right pane.
2. Select values for the relevant fields.



The screenshot shows a configuration panel with three tabs: 'Details', 'Configuration' (which is selected and highlighted with an orange underline), and 'View'. Below the tabs are three configuration fields:

- X-Axis: A dropdown menu with the text 'Assigned to Dev' and a downward arrow.
- Y-Axis: A dropdown menu with the text 'Count' and a downward arrow, followed by a text input field containing 'Application Components'.
- Grouped By: A dropdown menu with the text 'Component type' and a downward arrow.

For details about the configuration options, see ["Service Test Management Reports and Graphs Overview"](#) on page 298.

Select different projects - optional

To plot graphs using data from other projects, click **Select Projects** in the bottom right pane, and select the relevant project.

View the graph

Click the **View** tab to view the graph.

For more information on graphs, see the HP Application Lifecycle Management User Guide.