HP Business Process Testing

Software Version: 11.50

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 2002 - 2012 Hewlett-Packard Development Company, L.P.

Trademark Notices

Adobe® is a trademark of Adobe Systems Incorporated.

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

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

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

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

UNIX® is a registered trademark of The Open Group.

Documentation Updates

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

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

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

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

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

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

Or click the **New users - please register** link on the HP Passport login page.

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

Support

Visit the HP Software Support Online web site at:

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

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

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

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

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

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

To find more information about access levels, go to:

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

Contents

Welcome to this Guide ... 12
How This Guide is Organized ... 12
Documentation Library ... 13
Documentation Library Guides ... 13
Topic Types ... 16
Additional Online Resources ... 17
ALM Extension Guides ... 18
Business Process Testing Basics ... 19
Business Process Testing Introduction ... 21
Business Process Testing Overview ... 22
Business Process Testing Benefits ... 22
Identifying Roles ... 24
Choosing Methodologies ... 25
Business Process Testing Integration with other HP Products ... 28
Business Process Testing Workflow ... 28
Business Process Testing with Unified Functional Testing Workflow ... 29
Business Process Testing Enterprise Extension Workflow ... 31
Business Process Testing Keyboard Shortcuts ... 33
Business Process Testing Keyboard Shortcut Reference ... 34
Business Component Module Shortcuts ... 34
Test Plan Module Shortcuts for Business Process Testing ... 35
Manual Runner Shortcuts for Business Process Testing ... 36
Business Components ... 37
Introducing Business Components ... 39
Business Components Overview ... 40

Components with Manual Content 41
Components with Automated Content 41
Application Area Overview 43
How to Create Business Components in ALM 46
How to Create Components in ALM - Use-Case Scenario 49
Business Components User Interface 52
Business Components Module Window 53
Business Components Module Menus and Buttons 56
Business Process Testing Icons 62
Business Components Module Fields 65
New Component/Component Details Dialog Box 69
Snapshot Tab 72
Paste Components/Component Folders to Target Project Dialog Box 74
Select Destination Folder Dialog Box 75
Components with Manual Design Steps 79
Manual Design Steps Overview 80
How to Work with Manual Design Steps 80
Manual Design Steps User Interface 82
Design Steps Tab 83
Design Steps Pane/Component Step Details Dialog Box 83
Components with Keyword GUI Automation 89
Keyword GUI Automation Overview 90
How to Work with Keyword GUI Automation 93
Keyword GUI Automation User Interface 97
Automation Tab 98
Keyword View Pane 99
Select Application Area Dialog Box 104
Keyword View Options Dialog Box 105
Select Test Object Dialog Box 108
Output Options Dialog Box 110
Value Configuration Options Dialog Box 113
Checkpoint Properties Dialog Box 115

Parameterization / Properties Dialog Box 117
Output Value Properties Dialog Box 118
Information Dialog Box 120
General Information Dialog Box 121
Components with Scripted GUI or API Automation 123
Scripted GUI and API Automation Overview 124
How to Work with Scripted GUI and API Automation 124
Scripted GUI and API Automation User Interface 125
Script Launch Pane 125
Business Process Tests and Flows 127
Introducing Business Process Tests and Flows 129
Business Process Test and Flow Overview 130
How to Plan Business Process Tests and Flows 130
Business Process Test and Flow Specification 133
Business Process Test and Flow Specification Overview 134
How to Create Business Process Tests and Flows 134
How to Create Business Process Tests - Use-case Scenario 139
How ALM Calculates Business Process Test and Flow Statuses 144
Business Process Test and Flow Specification User Interface 145
Test Plan Module Window for Business Process Testing 145
Test Plan Module Menu and Buttons for Business Process Testing 147
Test Plan Module Icons for Business Process Testing 148
Test Plan Module Fields for Business Process Testing 150
Test Script Tab for Business Process Testing 152
Test Details Dialog Box for Business Process Testing 165
Test Validation Results Dialog Box 166
Learn Flow Process 169
Learn Flow Process Overview 170
How to Learn Flows 170
Criteria for Determining if a Component Can Be Reused 175
Learn Flow Process User Interface 177
Learn Flow Wizard 177

Learn Flow Dialog Box Page ... 178
Learn Flow Toolbar Page ... 178
Learn Flow Summary Page ... 180
Select Business Components Folder Dialog Box ... 183
Available Components for Reuse Dialog Box ... 184
Requirement Coverage ... 187
Requirement Coverage Overview ... 188
How to Create Coverage by Criteria ... 189
Run Conditions ... 193
Run Conditions Overview ... 194
How to Set Run Conditions ... 194
How to Set Run Conditions - Use-Case Scenario ... 195
Run Conditions User Interface ... 196
Run Conditions Dialog Box ... 196
Business Process Test Configurations ... 199
Business Process Test Configurations Overview ... 200
How to Work with Business Process Test Configurations ... 203
How to Work with Configurations and Static Data ... 204
How to Work with Configurations and Dynamic Data ... 205
How to Generate Test Configurations from Test Instances ... 209
Business Process Test Configurations User Interface ... 211
Test Configurations Tab for Business Process Testing ... 211
Data Tab: Static View ... 212
Data Tab: Dynamic View ... 213
Map Parameters Dialog Box ... 215
Data Resource Settings Dialog Box ... 217
Test and Flow Debugging ... 221
Test and Flow Debugging Overview ... 222
How to Debug Tests and Flows Containing Automated Components ... 222
Test and Flow Debugging User Interface ... 224
Run or Debug Test Dialog Box ... 224
Debug Run Information Dialog Box ... 226

Data Handling 227
Introducing Data Handling (Parameters) 229
Data Handling Overview 230
Parameter Categories 233
How to Handle Data in Business Process Testing 235
Parameters 237
Parameters Overview 238
How to Create Parameters 238
How to Set Parameter Values 239
Methods for Creating Parameters 241
Parameters User Interface 242
Parameters Page 243
Set Values Dialog Box 248
Dynamic Date Area 251
Parameters Dialog Box 254
New Parameter / Parameter Details Dialog Box 257
Edit Table Parameter Dialog Box 260
Parameter Linkage 265
Parameter Linkage Overview 266
Linkage and Iterations 266
Data Linkage Examples 267
How to Link Data 270
Parameter Linkage User Interface 272
I/O Parameters Dialog Box 272
Select Output Parameter Dialog Box 273
Parameter Promotion 275
Parameter Promotion Overview 276
How to Promote Parameters 276
Parameter Promotion User Interface 278
Promote Parameters Dialog Box 278
Test/Flow Parameters Promotion Status Dialog Box 280
Iterations 283

Iterations Overview 284
Group Iterations Overview 286
How to Set Data for Iterations 287
How to Import and Export (Save) Parameter Values 289
Iterations User Interface 291
Iteration Pages 292
Select Iterations Dialog Box 296
Map Imported Data Dialog Box 297
Create Flow/Test Parameters Dialog Box 298
Test Runs 301
Introducing Test and Flow Execution 303
Test and Flow Execution Overview 304
How to Run Business Process Tests and Flows Manually 304
How to Run Automated Business Process Tests and Flows 305
Run Progress Pane Hierarchy 308
Test and Flow Execution User Interface 310
Manual Runner Wizard for Business Process Testing 310
Manual Runner: Run Details Page for Business Process Testing 311
Manual Runner: Step Details Page for Business Process Testing 313
Test Properties Dialog Box 317
Change Detection and Resolution 319
Change Detection and Resolution Overview 320
How to Run Tests in Change Detection Mode 321
How to Resolve Detected Changes 323
Detection and Resolution Options 325
Change Detection and Resolution User Interface 327
Select Test Set (Instance) Dialog Box 327
Change Detection Report Screen 328
Appendices 337
Business Process Testing Enterprise Extension Environment-specific Information 339
Business Process Testing Enterprise Extension Environment-specific Information Overview 340

How to Work with Business Process Testing Enterprise Extension with Support for SAP ... 340

How Business Process Testing Enterprise Extension Creates Components when Learning SAP Flows ... 341

Business Process Testing Enterprise Extension Environment-specific User Interface ... 343

An Example of a SAP-specific Change Detection Report ... 344

Business Process Testing Glossary ... 345

Welcome to this Guide

Welcome to HP Business Process Testing. Business Process Testing with HP Application Lifecycle Management (ALM) enables non-technical subject matter experts to build and work with business components in a script-free environment and to create application-quality business process tests.

The HP Business Process Testing User Guide describes how to use the ALM application to work with HP Business Process Testing. It provides descriptive and conceptual information, step-by-step guidance to help you work with the application, and explanations of reference-oriented material.

Note: HP Business Process Testing is in Technology Preview. It will be released formally after completion of the Technology Preview period.

How This Guide is Organized

The HP Business Process Testing User Guide contains the following parts:

Part	Description
"Business Process Testing Basics" on page 19	Provides an overview of Business Process Testing features and methodology.
"Business Components" on page 37	Describes how to create, define, and modify the business components that are incorporated into business process tests and flows. This part also describes how to work with manual and automated components by defining design steps for manual components, and automation scripts or keyword steps for automated components. **Relevant tasks:** • "How to Create Business Components in ALM" on page 46 • "How to Work with Manual Design Steps" on page 80 • "How to Work with Keyword GUI Automation" on page 93 • "How to Work with Scripted GUI and API Automation" on page 124
"Business Process Tests and Flows" on page 127	Describes the unique elements in the Test Plan module interface when a business process test or flow is selected. Also describes how to work with business process tests and flows, including working with requirement coverage and working with test configurations. **Relevant task:** • "How to Plan Business Process Tests and Flows" on page 130

Part	Description
"Data Handling" on page 227	Describes how to expand the flexibility and reusability of business components, flows, and business process tests by replacing fixed values with parameters. This includes linking parameters, promoting parameters, and providing different data values for iterations. **Relevant task:** • "How to Handle Data in Business Process Testing" on page 235
"Test Runs" on page 301	Describes the unique elements in the Test Lab module interface for running business process tests and flows. **Relevant tasks:** • "How to Run Business Process Tests and Flows Manually" on page 304 • "How to Run Automated Business Process Tests and Flows" on page 305
"Appendices" on page 337	Provides additional appendices and a list of terms used throughout Business Process Testing.

Documentation Library

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

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

Documentation Library Guides

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

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

HP Application Lifecycle Management (ALM) Guides	Description
HP ALM User Guide	Explains how to use ALM to organize and execute all phases of the application lifecycle management process. It describes how to specify releases, define requirements, plan tests, run tests, and track defects.
HP ALM Administrator Guide	Explains how to create and maintain projects using Site Administration, and how to customize projects using Project Customization.
HP ALM Lab Management Guide	Explains how to use Lab Management to manage lab resources used for functional and performance testing on remote hosts.
HP ALM Tutorial	A self-paced guide teaching you how to use ALM to manage the application lifecycle management process.
HP ALM Installation Guide	Describes the installation and configuration processes for setting up ALM Server.
Business Process Testing User Guide	Explains how to use Business Process Testing to create business process tests.

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

HP ALM Best Practices Guides	Description
HP ALM Agile Testing Best Practices Guide	Provides best practices for implementing agile testing principles.
HP ALM Business Process Models Best Practices Guide	Provides best practices for working with the Business Models module.
HP ALM Database Best Practices Guide	Provides best practices for deploying ALM on database servers.
HP ALM Entities Sharing Best Practices Guide	Provides best practices for sharing entities.
HP ALM Project Planning and Tracking Best Practices Guide	Provides best practices for managing and tracking releases.
HP ALM Project Topology Best Practices Guide	Provides best practices for structuring projects.
HP ALM Upgrade Best Practices Guide	Provides methodologies for preparing and planning your ALM upgrade.
HP ALM Versioning and Baselining Best Practices Guide	Provides best practices for implementing version control and for creating baselines.
HP ALM Workflow Best Practices Guide	Provides best practices for implementing workflows.

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

HP ALM API References	Description
HP ALM Project Database Reference	Provides a complete online reference for the project database tables and fields.
HP ALM Open Test Architecture API Reference	Provides a complete online reference for the ALM COM-based API. You can use the ALM open test architecture to integrate your own configuration management, defect tracking, and home-grown testing tools with an ALM project.

HP ALM API References	Description
HP ALM Site Administration API Reference	Provides a complete online reference for the Site Administration COM-based API. You can use the Site Administration API to enable your application to organize, manage, and maintain ALM users, projects, domains, connections, and site configuration parameters.
HP ALM REST API Reference	Provides an online reference for the ALM REST-based API. You can use the REST API to access and work with ALM data.
HP ALM COM Custom Test Type Developer Guide	Provides a complete online guide for creating your own testing tool and integrating it into the ALM environment using native COM development tools.
HP ALM .NET Custom Test Type Developer Guide	Provides a complete online guide for creating your own testing tool and integrating it into the ALM environment using a combination of DCOM and .NET classes.

Topic Types

The content in the above mentioned ALM guides is organized by topics. Three main topic types are in use: **Concepts**, **Tasks**, and **Reference**.

Topic Type	Description	Usage
Concepts	Background, descriptive, or conceptual information.	Learn general information about what a feature does.
Tasks	**Instructional Tasks**. Step-by-step guidance to help you work with the application and accomplish your goals. Task steps can be with or without numbering: • **Numbered steps**. Tasks that are performed by following each step in consecutive order. • **Non-numbered steps**. A list of self-contained operations that you can perform in any order.	• Learn about the overall workflow of a task. • Follow the steps listed in a numbered task to complete a task. • Perform independent operations by completing steps in a non-numbered task.
	Use-case Scenario Tasks. Examples of how to perform a task for a specific situation.	Learn how a task could be performed in a realistic scenario.

Topic Type	Description	Usage
Reference	**General Reference**. Detailed lists and explanations of reference-oriented material.	Look up a specific piece of reference information relevant to a particular context.
	User Interface Reference. Specialized reference topics that describe a particular user interface in detail. Selecting Help on this page from the Help menu in the product generally open the user interface topics.	Look up specific information about what to enter or how to use one or more specific user interface elements, such as a window, dialog box, or wizard.
Troubleshooting and Limitations	**Troubleshooting and Limitations**. Specialized reference topics that describe commonly encountered problems and their solutions, and list limitations of a feature or product area.	Increase your awareness of important issues before working with a feature, or if you encounter usability problems in the software.

Additional Online Resources

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

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

ALM Extension Guides

Extensions provide added functionality to HP ALM. If you have a license for an ALM extension, you can utilize the added functionality by enabling the extension on a per project basis. For more details on enabling extensions, refer to the *HP Application Lifecycle Management Administrator Guide*.

To view the list of extensions available with ALM 11.50, or to download documentation for ALM extensions, visit the HP ALM Add-ins Page, accessible from the main ALM **Help** menu.

Business Process Testing Basics

Chapter 1

Business Process Testing Introduction

This chapter includes:

Business Process Testing Overview 22

Business Process Testing Integration with other HP Products 28

Business Process Testing Workflow 28

Business Process Testing with Unified Functional Testing Workflow 29

Business Process Testing Enterprise Extension Workflow 31

Business Process Testing Overview

Many applications are critical for modern corporations. Effective functional testing is essential to assess the quality of your applications and ensure that they are stable.

Business Process Testing enables subject matter experts to:

- Design quality assurance tests for an application early in the development cycle and in a script-free environment using reusable business components and flows.
- Design quality assurance tests from the design of a high-level test structure down to the design of actual steps.

The *HP Business Process Testing User Guide* describes how to use Business Process Testing to create business process tests. It provides step-by-step instructions to help you create and debug business components, create and run flows, and build and run business process tests.

This chapter introduces Business Process Testing and describes workflows for working with Business Process Testing and Business Process Testing Enterprise Extension.

Note:

- For details on installing and setting up Business Process Testing Enterprise Extension, and on configuring user permissions, see the *HP Application Lifecycle Management Administrator Guide*.
- **ALM Editions:** Business Process Testing is available with ALM Edition and Quality Center Enterprise Edition. For details, see the *HP Application Lifecycle Management User Guide*.

This section also includes:

- "Business Process Testing Benefits" below
- "Identifying Roles" on page 24
- "Choosing Methodologies" on page 25

Business Process Testing Benefits

Business Process Testing, along with testing tools such as Unified Functional Testing (UFT), provides numerous benefits:

- Business process tests can be created by subject matter experts who best understand the business processes that need to be tested, without the need for a programming background.
- Business Process Testing enables structured testing of an application by combining test automation and automatically generated, easy-to-understand, plain-language test documentation.
- Business Process Testing dramatically reduces the need for repeating steps and facilitates text reusability.

- Business Process Testing supports the creation of both manual and automated steps for each component. The manual and automated steps can be synchronized.
- Business Process Testing supports a keyword GUI testing approach for creating automated business process tests.
- Business Process Testing is not dependent on the completion of detailed testing scripts. After designing the test and establishing automation guidelines, applications can be tested manually by non-technical users even before automated tests are ready. This facilitates the quick implementation of business process tests.
- Usage of automation assets in the design or implementation of test steps does not require coding skills.
- Business process test creation is accelerated by the ability to use modular, reusable flows and business component units in multiple tests.
- Version control enables you to keep track of changes made to entities in your project, including business process tests, flows, and components.
- Quality assurance experts can take advantage of the sophisticated automated testing capabilities in UFT and other testing tools. Using these tools, automation engineers can prepare object hierarchies and application-specific functions. Automation engineers can then package these hierarchies and functions building blocks for use in business process tests by subject matter experts.
- Documents containing information about the tests, flows, and components in a project can easily be generated.

Additional Benefits of Business Process Testing Enterprise Extension

Business Process Testing Enterprise Extension adds additional functionality to ALM and Business Process Testing that enables you to test enterprise applications. This functionality includes features such as learning flows and change detection.

Business Process Testing Enterprise Extension includes the following additional benefits:

- Business Process Testing Enterprise Extension can automatically learn the actions you perform on your applications, and then generate a flow containing components based on the screens and transactions within your applications. You can reuse learned components instead of creating new components in your flow.
- Business Process Testing Enterprise Extension enables you to run tests and flows on applications in Change Detection mode. This enables you to determine how the applications have changed since the test or flow was built.

For details about the supported enterprise environments, see "Business Process Testing Enterprise Extension Environment-specific Information Overview" on page 340.

For details about enabling Business Process Testing Enterprise Extension, see the *HP Application Lifecycle Management Administrator Guide*.

Identifying Roles

The Business Process Testing model is role-based, allowing non-technical subject matter experts to work on tests with automation engineers.

Roles are flexible, depending on the abilities and time resources of the personnel. There are no product-dictated rules controlling which types of users can perform which Business Process Testing tasks (provided that the users have the correct permissions).

Note: To control access to various modules and tasks, ALM enables you to configure access permissions for users. For details, see the *HP Application Lifecycle Management Administrator Guide*.

Role	Description
Subject Matter Experts	Have specific knowledge of the application logic, a high-level understanding of the entire system, and a detailed understanding of the elements and tasks fundamental to the application being tested. Subject matter experts: • Determine the business processes to test. • Identify activities common to multiple processes (such as a login procedure, used in many business process tests or flows). • Create business components and steps. • Define automated keyword GUI component steps. • Create flows and business process tests. • Define data for iterations and configurations. • Run tests to verify that they are designed appropriately and run as expected. • Review test results.

Role	Description
Automation Engineers	Are experts in automated testing using a testing tool such as UFT. **Note:** The automation engineer is only needed if the Business Process Testing framework includes automated tests. Automation engineers prepare the resources and automated functions required for testing the features associated with each component, such as: • Application areas, which can be defined within ALM and within other testing tools. • Function libraries with general scripts, which are encapsulated into general operation keyword GUI steps. • Shared object repositories, which can be populated with objects that represent the objects in the application being tested. The automation engineer can rename the object repository, create more appropriate keyword GUI steps, and filter out objects which are not relevant. The subject matter expert can use these objects to create steps in business components with keyword GUI automation. Automation engineers may also be responsible for some of the tasks listed above for the subject matter expert. Automation engineers can also create, debug, and modify business components in the testing tool.
QA Testers	Use and run the business process tests.
ALM Administrators	Set up and configure Business Process Testing.

Choosing Methodologies

Business Process Testing is flexible and does not enforce any one particular model for incorporating business processes into your testing environment. The actual workflow in an organization may differ for different projects, or at different stages of the application development life cycle.

Bottom-up Methodology

Defining low-level components first and then designing business process tests based on the defined components is called a bottom-up methodology. This methodology is particularly useful when:

- The business processes in the organization are clearly defined.
- Users are new to Business Process Testing.

The chapters in this guide are structured according to the bottom-up methodology.

Top-down Methodology

The top-down methodology is based on the perspective of the subject matter expert who has a high-level understanding of the entire system.

The top-down methodology advocates the creation of business process testing entities according to the following hierarchy:

- Business process tests, which contain flows and/or business components
- Flows, which contain business components
- Business components, which contain manual and/or automated steps

The top-down methodology is based on the following design phases:

Design Phase	Description
High-level design	Includes the high-level design, creation of a structure for business process tests, and determining the test configurations for testing different use-cases that will be needed. When designing at the high-level, facilitate automation: • By designing with modularity in mind. Design tests to use smaller, reusable components that automated tests can call multiple times. • By designing tests with reusable components, which makes maintaining tests easier. • By designing with the understanding that automation engineers can use the design as a structure for creating methods that facilitate automation, instead of for creating scripted components. This part of the design phase is often done by both the subject matter expert and the automation engineer together. Creating the structure for business process tests and determining the needed test configurations are typically performed by the subject matter expert.

Mid-level design	Includes the: • Creation of flows (sets of business components in a logical order that can be run). Flows are considered "compound components." • Creation of business components (reusable units that perform specific tasks in a business process). Only the shell of the component is created during this phase. • Specification of criteria for more granular test coverage (requirements) as necessary. • Linking to other ALM entities. • Adding business components to business process tests and flows. This part of the design phase is typically performed by the subject matter expert, but may also be done in conjunction with the automation engineer, depending on available resources and skills.
Low-level implementation	Includes the low-level implementation of business component content by: • Creating component steps (the content of the business component), including automated steps when necessary • Grouping components • Setting up iterations (for business process tests, flows, groups, and components) • Parameterizing This part of the design phase can be performed by the subject matter expert, the automation engineer, or both together.

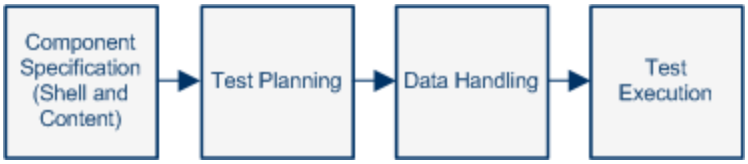
Business Process Testing Integration with other HP Products

Business Process Testing integrates with other HP products, providing a gateway between ALM, the solution for managing the core application lifecycle, and other HP products that are part of the core application lifecycle.

Product	Integration
HP Unified Functional Testing (UFT)	Automate business components using Keyword GUI, Scripted GUI, and API automation types. These components can then be included in automated tests that use the UFT testing tool. HP Unified Functional Testing (UFT) comprises the products known as HP QuickTest Professional and HP Service Test. • Functionality provided by QuickTest is known as GUI testing in UFT. • Functionality provided by Service Test is known as API testing in UFT. For an overview of how Business Process Testing works with UFT, see "Business Process Testing with Unified Functional Testing Workflow" on next page.
HP Sprinter	Run manual tests using Sprinter. Sprinter provides advanced functionality and tools to assist you in the manual testing process. For an overview of how to work with Sprinter, see the *HP Application Lifecycle Management User Guide*.
Business Process Testing Enterprise Extension	Business Process Testing Enterprise Extension adds additional functionality to ALM and Business Process Testing that enables you to test enterprise applications. This functionality includes features such as learning flows and change detection. For an overview of how Business Process Testing works with Business Process Testing Enterprise Extension, see "Business Process Testing Enterprise Extension Workflow" on page 31.

Business Process Testing Workflow

This roadmap for designing and implementing business process testing presents the bottom-up methodology and includes the following phases:



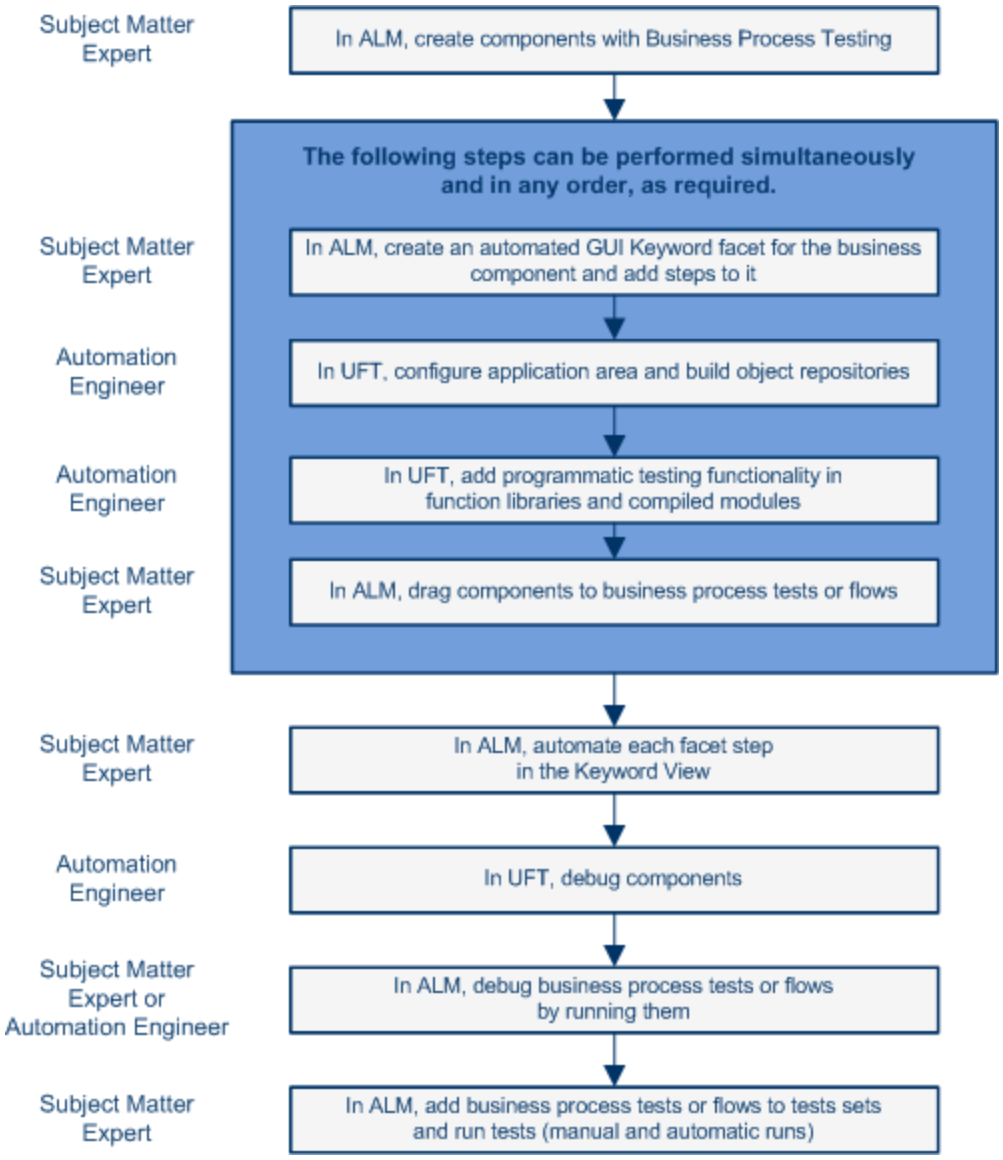
Phase	Description
Component Specification	Develop a component tree with components. Create the component shell by adding basic details. Create component content by adding facets. Facets are containers for steps or scripts that determine whether the application meets expectations. Component content can contain: • Design steps for manual components • Automation, for automated components • Both design steps and automation For task details on creating components, see "How to Create Business Components in ALM" on page 46.
Test Planning	Build test plans and design business process tests and flows. For task details, see "How to Plan Business Process Tests and Flows" on page 130.
Data Handling	Design the data that each business process test, flow, or component uses when run. For task details, see "How to Handle Data in Business Process Testing" on page 235.
Test Execution	Create a subset of the business process tests in your project and run them. For task details, see "How to Run Business Process Tests and Flows Manually" on page 304 and "How to Run Automated Business Process Tests and Flows" on page 305.

For an example of a common workflow using Business Process Testing Enterprise Extension, see "Business Process Testing Enterprise Extension Workflow" on page 31.

Business Process Testing with Unified Functional Testing Workflow

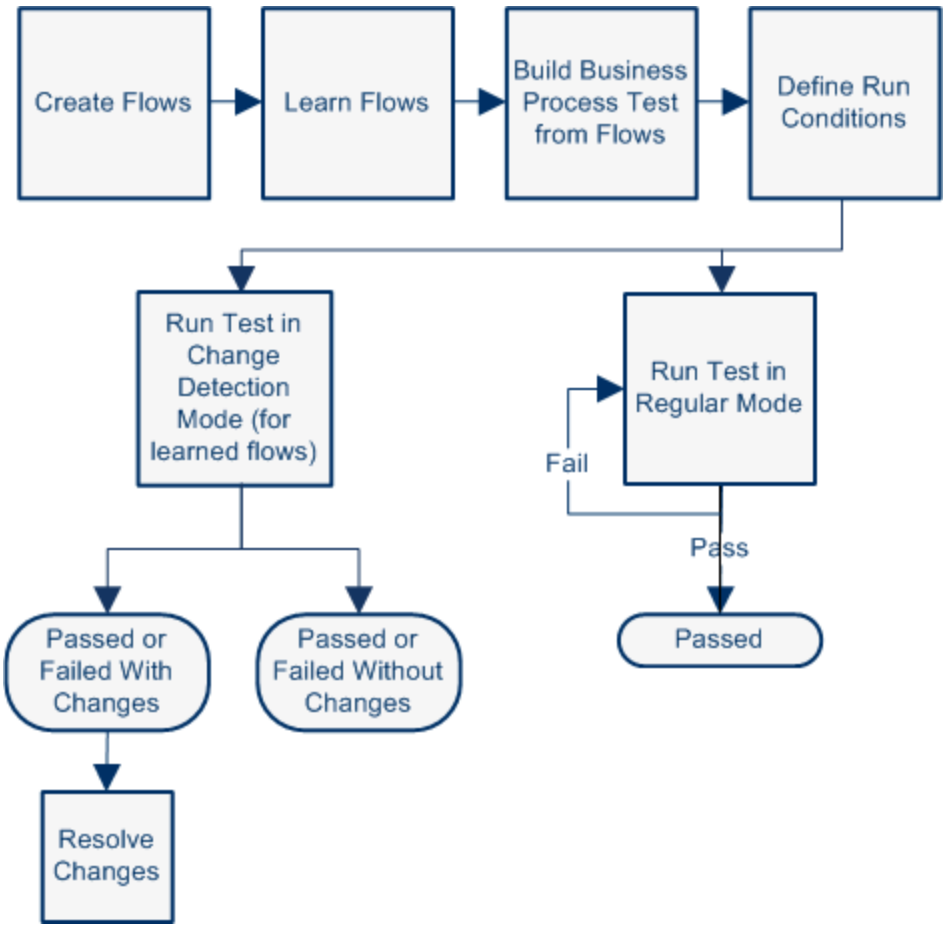
The following is an example of a common Business Process Testing workflow using Unified Functional Testing (UFT). This example demonstrates one possible method for working with components and keyword GUI steps.

The actual workflow in an organization may differ for different projects, or at different stages of the product development life cycle.


Subject Matter Expert
In ALM, create components with Business Process Testing
The following steps can be performed simultaneously and in any order, as required.
Subject Matter Expert
In ALM, create an automated GUI Keyword facet for the business component and add steps to it
Automation Engineer
In UFT, configure application area and build object repositories
Automation Engineer
In UFT, add programmatic testing functionality in function libraries and compiled modules
Subject Matter Expert
In ALM, drag components to business process tests or flows
Subject Matter Expert
In ALM, automate each facet step in the Keyword View
Automation Engineer
In UFT, debug components
Subject Matter Expert or Automation Engineer
In ALM, debug business process tests or flows by running them
Subject Matter Expert
In ALM, add business process tests or flows to tests sets and run tests (manual and automatic runs)

Business Process Testing Enterprise Extension Workflow

The roadmap with Business Process Testing Enterprise Extension includes the following additional phases:



Note: The actual workflow in an organization may differ for different projects, or at different stages of the application development life cycle. All stages in this workflow can be performed by a subject matter expert, without the involvement of an Automation Expert.

Phase	Description
Create Flows	You create flows in the Test Plan module. For task details, see "How to Create Business Process Tests and Flows" on page 134.

Phase	Description
Learn Flows	Business Process Testing Enterprise Extension enables you to learn the steps that comprise a transaction as you navigate your enterprise application. Business Process Testing Enterprise Extension creates automated business components corresponding to screens and transactions in your application, and creates automated component steps based on the operations you perform. These components are then added to the flow. During the learning process, you can insert checkpoints and output values. For task details, see "How to Learn Flows" on page 170.
Build Tests from Flows	You build a business process test from the flows that were created in the Learn Flow process. In addition, you can include other flows and business components with both manual and automated steps. For task details, see "How to Create Business Process Tests and Flows" on page 134.
Define Run Conditions	You can use run conditions to determine whether to run a component or flow, skip to the next component and continue, or end the run. This determination is based on a parameter's value and the run condition's definition. For task details, see "How to Set Run Conditions" on page 194.
Run Tests in Regular Mode	You run a business process test containing flows as you would a regular business process test. For task details, see "How to Run Business Process Tests and Flows Manually" on page 304.
Run Tests in Change Detection Mode	If your test includes flows learned automatically, you can run the test in Change Detection mode. Business Process Testing Enterprise Extension checks whether elements in your enterprise application changed since the flow was learned. This enables you to later update your flows to reflect these changes. For task details, see "How to Run Tests in Change Detection Mode" on page 321.
Resolve Changes	When a test or flow that ran in Change Detection mode is complete, you can view a report detailing the status of each flow, business component, and component step in your test, including whether objects used in your steps have changed in your application. If changes were detected during the test run, you are offered a variety of options for resolving them, such as removing an obsolete step from a component, or reporting a defect. After you resolve the changes and update your flows and tests, you can run additional testing cycles to further test your application. For task details, see "How to Resolve Detected Changes" on page 323.

Chapter 2

Business Process Testing Keyboard Shortcuts

This chapter includes:

Business Process Testing Keyboard Shortcut Reference 34

Business Process Testing Keyboard Shortcut Reference

You can navigate HP Business Process Testing, and perform many Business Process Testing commands, using shortcut keys.

Note: This chapter describes how to use several Business Process Testing features. HP Business Process Testing Enterprise Extension provides additional, automated features that enable you to test enterprise applications. Therefore, some procedures described in this chapter may not be part of the recommended best practice when using Business Process Testing to test your enterprise application.

This section includes:

- "Business Component Module Shortcuts" below
- "Test Plan Module Shortcuts for Business Process Testing" on next page
- "Manual Runner Shortcuts for Business Process Testing" on page 36

Note: You can navigate ALM, and perform common commands, by pressing the shortcut keys described in the *HP Application Lifecycle Management User Guide*.

Business Component Module Shortcuts

You can perform the following commands in the Business Components module by pressing the corresponding shortcut keys:

Shortcut Key	Function
CTRL + ALT + N	Creates a new component folder.
CTRL + N	Creates a new component.
DELETE	Deletes an entity (component, step).
F5	Refreshes. **UFT:** When accessing the Business Components module from within UFT, this shortcut key performs a run instead of a refresh.
ALT + G	Opens the Go To Component dialog box.
CTRL + U	Copies a URL.
ALT + S	**Snapshot tab:** Creates a snapshot.

Shortcut Key	Function
ALT + L	**Snapshot tab:** Loads a snapshot from file.
ALT + D	**Snapshot tab:** Deletes a snapshot.
Alt + N	**Design Steps tab > Design Steps pane:** Adds a new step. **Parameters tab**: Adds a new parameter.
CTRL + C	**Design Steps tab > Design Steps pane:** Copies steps.
CTRL + V	**Design Steps tab > Design Steps pane:** Pastes steps.
CTRL + P	**Design Steps tab > Design Steps pane:** Inserts a parameter.
CTRL + K	**Design Steps tab > Design Steps pane:** Completes a parameter name.
CTRL + DELETE	**Parameters tab:** Deletes a parameter.
CTRL + F	**Design Steps tab > Design Steps pane:** Finds a design step.
Ctrl + F11	**When positioned in the Output cell from the Keyword View of the Steps tab:** Opens the Output Options dialog box. **When positioned in the Value cell from the Keyword View of the Steps tab:** Opens the Value Configuration Options dialog box.

Test Plan Module Shortcuts for Business Process Testing

You can perform the following commands in the Test Plan module for business process tests and flows by pressing the corresponding shortcut keys:

Shortcut Key	Function
ALT + S	**Test Script tab:** Opens the Select Component pane.
CTRL + R	**Test Script tab:** Creates a new component request.
DELETE	**Test Script tab:** Removes the selected item (component, flow business process test, group).
F5	Refreshes.
F8	Validates the test.

Manual Runner Shortcuts for Business Process Testing

You can perform the following commands in the Manual Runner for business process tests and flows by pressing the corresponding shortcut keys:

Shortcut Key	Function
CTRL + R	Begins run.
CTRL + Q	Ends run.
CTRL + K	Cancels run.
CTRL + U	Attaches to run.
CTRL + B	Adds a defect.
CTRL + UP ARROW	Goes to previous step.
CTRL + DOWN ARROW	Goes to next step.
CTRL + P	Passes selected steps.
CTRL + ALT + A	Passes (including sub-steps).
CTRL + F	Fails selected steps.
CTRL + G	Fails (including sub-steps).
ALT + LEFT ARROW	Returns to previously clicked link.
ALT + S	Attaches to step.
F12	Toggles the **Always on Top** option.

Business Components

Chapter 3

Introducing Business Components

This chapter includes:

Business Components Overview 40

Components with Manual Content 41

Components with Automated Content 41

How to Create Business Components in ALM 46

Business Components User Interface 52

Business Components Overview

The Business Components module enables you to create, define, modify, and manage business components in HP Application Lifecycle Management (ALM). These components provide the basis for Business Process Testing and are incorporated into business process tests and flows.

A business component is a reusable unit that:

- Performs a specific task in a business process
- Describes the condition or state of the application before and after that task

Business Component Examples

Business Component Name	Task	Application State Before	Application State After
Login	Banker logs into banking application	<none>	The application is launched and the main home page is displayed.
SearchLoan	Banker searches for an existing loan	The banker is logged in and the main home page is displayed.	The application displays the main loan details page, or a page indicating that the loan was not found.

Business components are comprised of:

- **A Shell** (general information such as component name, status). For task details on creating components and providing the shell, see "How to Create Business Components in ALM" on page 46.
- **Content** (steps or scripts). Low-level, detailed information, such as the component's design steps and/or automation. The contents provide detailed instructions for performing business process tasks in the application. The contents are saved in containers called facets. Component content can be manual, automated, or both—depending on whether you create manual design steps and/or implement automation for the component.
  - For task details on creating manual components, see "How to Work with Manual Design Steps" on page 80.
  - For task details on creating automated components, see "How to Work with Keyword GUI Automation" on page 93 and "How to Work with Scripted GUI and API Automation" on

page 124.

Example of Content

Step	Description	Expected Result
1	Open the application.	The application is launched and the login page is displayed.
2	Enter a user name.	The cursor advances to the password field.
3	Enter a password.	The password is displayed as asterisks.
4	Click **Submit** on the Web page.	The application's main page is displayed.

You can use a component in multiple business process tests and flows. When you modify a component or its content, all business process tests or flows containing that component reflect that modification.

You can use run conditions to enable components to run selectively, based on earlier stages within the test or flow. For task details, see "How to Set Run Conditions" on page 194.

Business Process Testing Enterprise Extension: When Business Process Testing Enterprise Extension learns a flow, it creates a new business component for each screen or tab through which you navigate in your applications. Business Process Testing Enterprise Extension also enables you to identify and reuse learned components instead of creating new components in your flow. For task details on learning flows, see "How to Learn Flows" on page 170.

Components with Manual Content

Use the Design Steps tab in the Business Components module to create or modify manual business component content in the form of steps and expected results.

Design steps represent operations that should be performed on your application when you run the component in a business process test or flow. Each step comprises a step name, a textual description of the procedure to be performed on the application, and the expected result (if any), of performing that step. When a business process test or flow with a manual component runs in ALM, the tester performs the steps defined in the manual component.

For task details, see "How to Work with Manual Design Steps" on page 80.

For user interface details, see "Manual Design Steps User Interface" on page 82.

Components with Automated Content

Use the Automation tab in the Business Components module to create or modify automated business component content either:

- As scripts, implemented in a testing tool such as UFT (scripted GUI or API automation)
- In the form of keyword-driven steps and expected results (keyword GUI automation)

Steps represent operations that should be performed on your application when you run the component in a business process test or flow. Each step is made up of an item (an object in the application or an operation), and an operation. The available items and operations are defined by the automation engineer in the object repository and the function libraries. These objects and operations can be saved in ALM, making them available for insertion into steps by the subject matter expert.

Note: Most of the information, examples, and images in this guide focus specifically on working with keyword GUI automated components. However, much of the information also applies to scripted components (either scripted GUI or API).

When using the automated runner to run a business process test or flow in ALM, the steps or script defined in an automated component are performed.

This section contains the following topics:

- "Automation Phases" below
- "Automation Types" below

Automation Phases

Creating automated components consists of the following phases:

Phase	Description
Automation Type Selection	You create automation for a component when you select the automation type. After this phase, the component is accessible in the corresponding testing tool, such as UFT. Creating the automated component within ALM is typically done by the subject matter expert.
Content Automation Preparation	You prepare a component's content for automation, for example, by selecting an application area. Preparing the automated component within ALM is typically done by the subject matter expert. For conceptual details on creating application areas, see "Application Area Overview" on next page.
Content Automation	You create steps. The steps represent the operations that are performed automatically on your application when you run the component using a testing tool. After this phase, the steps are fully automated, and the automated component can run in an automatic runner without pausing. Automating steps within ALM is typically done by the subject matter expert in conjunction with the automation engineer.

Automation Types

Business Process Testing provides the following types of automation.

Automation Type	Description
Keyword GUI automation	Enables you to create and modify automated keyword GUI steps in a Keyword view from within ALM. For task details, see "How to Work with Keyword GUI Automation" on page 93. **Note:** The Keyword view in ALM provides similar functionality to the Business Component Keyword View in UFT. For a full description of the functionality provided in UFT, see the *HP Unified Functional Testing for HP Business Process Testing User Guide*.
Scripted GUI automation	Enables you to launch UFT to view and modify the automated scripted GUI and API steps. For task details, see "How to Work with Scripted GUI and API Automation" on page 124.
API automation	Enables you to launch UFT or HP Service Test to view and modify the automated API steps. For task details, see "How to Work with Scripted GUI and API Automation" on page 124. **Note:** API scripted automation is supported for HP API 11.10 or later. For details, see the *HP Service Test User Guide* or the information about API testing in the *HP Unified Functional Testing User Guide* and HP Software Self-solve knowledge base article KM1016396 (http://h20230.www2.hp.com/selfsolve/document/KM1016396).

This section also includes:

- "Application Area Overview" below

Application Area Overview

An application area provides all settings and resources required to automate content. These include:

- Representations of the objects from your application, contained in the shared object repository. The automation engineer can change the behavior of these objects in the application using the RegisterUserFunc utility statement in UFT. For details, see the *Unified Functional Testing Object Model Reference*.
- User-defined operations (keyword GUIs), contained in function libraries.
- Special operations to recover from errors and unexpected events, contained in recovery scenarios.

You can create multiple application areas, each representing a different area or aspect in your application. It is important that you select the correct application area on which to base your business component. Each application area should have a detailed description to help you to make the correct choice.

You can create basic application areas based on a default template in the ALM Test Resources module. (For details, see the *HP Application Lifecycle Management User Guide*.) However, most application area functionality is provided by the subject matter expert in UFT.

When you create a business component and choose an associated application area, the settings and resources that are defined in the current application area are embedded in the component. Therefore, any changes made to the application area in UFT at a later time are applied directly to the component.

The shared object repository, function libraries, and scenario files that comprise an application area are links to the actual repository and files stored in ALM. Changes to the stored repository and files affect all the components that reference them, and it is not necessary to update the application area for such changes.

You cannot delete an application area that is associated with a business component.

This section contains the following topics:

- "Who Creates Application Areas?" below
- "BPT Resources Folder" below
- "Other Resources" on next page

Who Creates Application Areas?

The following users can create application areas:

- The automation engineer, in UFT. For details on working with and creating application areas in UFT, see the *HP Unified Functional Testing User Guide*. For details on working with and creating application areas in API, see the *HP Service Test User Guide* or the information about API testing in the *HP Unified Functional Testing User Guide*.
- Either the subject matter expert or the automation engineer in the Test Resources module in ALM. The application area entity that is created in ALM provides the structure for automation, using default resources.

You can start to create an automated component, even if its application area has not yet been created or fully-defined. Create the automated component by adding comments describing the steps you need to create when the application area is available. However, if the application area has not yet been created, you cannot add keyword GUI steps to your component yet.

BPT Resources Folder

The business component resources for the project are created automatically in the Test Resources module by the Unified Functional Testing Add-in for Business Process Testing. These resources are stored in the **BPT Resources** folder. The automatic creation occurs:

- The first time you click **Business Components** module **> Automation** tab in a new project
- When you create a UFT automated component for the first time
- When UFT connects to ALM for the first time

For business process tests or flows to perform properly, the **BPT Resources** folder and its subfolders should not be renamed or deleted.

Note: In previous versions of Quality Center, the **BPT Resources** folder was created in the Test Plan module. When upgrading from Quality Center version 9.2, a **BPT Resources** folder that exists in the Test Plan module must be moved to the Test Resources module using the QuickTest Asset Upgrade Tool for Quality Center.

The **BPT Resources** folder contains all the UFT resources available for business components in the project, and includes the following subfolders.

Folder	Resource Description
Application Areas	Provides all of the settings and resources required to create the automated content for a particular application or part of an application. The application area typically contains all the items contained in the folders listed below.
Function Libraries	Contains all the function libraries available for business components in the UFT project. Function libraries provide customized operations (keyword GUIs) for components (VBScript functions, subroutines, and so on). The **Libraries** folder contains the default function library containing operations that can be used when creating steps. The automation engineer can store additional function libraries in this folder.
Object Repositories	Contains all the shared object repository files available for components in the UFT project. Object repository files define the test objects that can be used in the component steps.
Recovery Scenarios	Contains all the recovery scenario files available for components in the UFT project. Recovery scenarios define special operations to recover from errors and unexpected events during the component run. By default, the **Recovery Scenarios** folder contains the **DefaultWeb.qrs** file, which is a recovery scenario file that can be used in the Web environment.

Other Resources

This section lists additional resources that impact the objects available to you when automating using the UFT keyword GUI method:

Associated Add-ins

The specified set of Unified Functional Testing Add-ins that is associated with the business component determines the types of objects that UFT recognizes and that can be tested using that business component. Each add-in is associated with a development environment. For example, UFT includes built-in add-ins for testing in Web, ActiveX, and Visual Basic environments. Additional Unified Functional Testing Add-ins are available for testing environments such as SAP Solutions, Java, Oracle, Siebel, terminal emulators, and more.

ALM assumes that the add-ins associated with the first automated components in a business process test are required for all the automated components in the same test. Therefore, it is important to make sure that all required Unified Functional Testing Add-ins are associated with the first business component in the test. If they are not, you should ask the automation engineer to add them to the application area used by the first component in the test. Then update your copy of the application area.

Applications List

UFT runs components only on the set of Windows-based applications that are specified for the component. It can also run on applications in any other environment for which the appropriate Unified Functional Testing Add-in is loaded.

How to Create Business Components in ALM

This task describes how to create business components, their shells, and their content (design steps or automation) in ALM.

Note:

- This task is part of a higher-level task. For details, see "Business Process Testing Workflow" on page 28.
- **Product Feature Movie:** To view a movie that demonstrates basic information about Business Process Testing, select **Help > Product Feature Movies** in the ALM main window.

This task includes the following steps:

- "Prerequisites" below
- "Create the component tree" on next page
- "Create the component shell" on next page
- "Update the component shell" on next page
- "Create component content" on page 48
- "Handle component requests - optional" on page 48
- "Convert manual tests - optional" on page 48
- "Analyze components" on page 48

1. ## Prerequisites

   - Determine the component scope by gathering information such as business process specifications, marketing and business requirements documents, and stakeholder goals.

     Examples

     Some questions you may want to ask are:

     - What is the main purpose and direction of the application from a business perspective?
     - What are your business or testing priorities? How can you represent these priorities as business components?

- How much reuse is necessary? This can help determine the division of application testing into tests, flows, groups, and components.
- Is testing going to be automated, manual or both?

- Decide if Business Process Testing, by default, should create a facet (container) for manual design steps automatically when you create a component. For details, see the *HP Application Lifecycle Management Administrator Guide*.

2. ## Create the component tree

Define a hierarchical framework for your business components by creating a component tree, as you would any other tree in ALM. For user interface details, see "Business Components Module Window" on page 53.

3. ## Create the component shell

Define the shell for your business components. For user interface details, see "New Component/Component Details Dialog Box" on page 69.

Tip: You can also create components:

- Using other testing tools, such as UFT. See the *HP Unified Functional Testing User Guide* or the relevant guide for the testing tool.
- By converting existing manual ALM tests (not business process tests) to components. For task details, see "Convert manual tests - optional" on next page.
- **Business Process Testing Enterprise Extension:** When learning flows. For user interface details, see "Learn Flow Process User Interface" on page 177.
- Copying components from other projects. For user interface details, see "Paste Components/Component Folders to Target Project Dialog Box" on page 74
- Handling component requests. For task details, see "Handle component requests - optional" on next page.

Components are added to the component tree with the manual component icon. The components are assigned the status **Under Development**.

Tip: By default, components are created with one manual facet associated with it for adding design steps. You can change this default if you prefer that the component be created without a manual facet. In this case, an "empty" component is created. For details, see the *HP Application Lifecycle Management Administrator Guide*.

4. ## Update the component shell

For each component, you can update its content and implementation requirements (details, design steps, automation, attachments, parameters, dependencies, and history).

Right-click a component and select **Component Details**. The Component Details dialog box opens. For user interface details, see "New Component/Component Details Dialog Box" on page 69.

5. ## Create component content

Determine if the content should be manual, automated, or both.

For automated components, determine the automation testing tool (such as UFT) and method (scripted GUI, keyword GUI, API).

> **Tip:** Review the guidelines under "Choosing Methodologies" on page 25.

For task details on adding manual design steps to the component content, see "How to Work with Manual Design Steps" on page 80.

For task details on adding keyword GUI automation to the component content, see "How to Work with Keyword GUI Automation" on page 93.

For task details on adding scripted automation (either scripted GUI or API) to the component content, see "How to Work with Scripted GUI and API Automation" on page 124.

6. ## Handle component requests - optional

Another way to create components is to handle component requests submitted in the Test Plan module.

After confirming that a requested component and its parameters are necessary for business process testing, in the component tree, drag the requested component from the **Component Requests** folder to the appropriate folder. The **Not Implemented** status changes to **Under Development** as the component request becomes a component.

> **Note:**
> - You can rename and edit the requested component as you would any other ALM entity.
> - You can delete a component request in the Business Components module only.

For task details on requesting components, see the step "How to Create Business Process Tests and Flows" on page 134..

7. ## Convert manual tests - optional

You can use existing ALM manual tests as a basis for defining your manual business components.

Convert existing ALM manual tests using the Select Destination Folder dialog box in the Test Plan module. For user interface details, see "Select Destination Folder Dialog Box" on page 75.

8. ## Analyze components

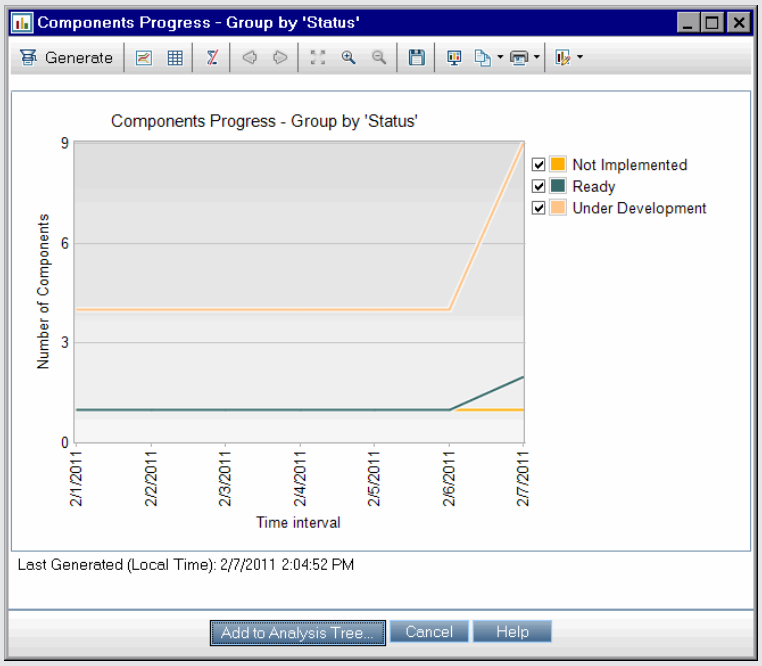
- Review your business components to ensure that they meet the defined scope. After a component is approved, you change the status of a component from **Under Development**

to **Ready**.

- To help review the components, you can generate reports and graphs. For task details, see the information about analyzing data in the *HP Application Lifecycle Management User Guide*.

Example

To see the development status of your components, you could view the Components Progress graph:



How to Create Components in ALM - Use-Case Scenario

This use-case scenario describes the workflow for creating components.

Note: For a task related to this scenario, see "How to Create Business Components in ALM" on page 46.

This scenario includes the following steps:

- "Background" on next page
- "Create a component tree" on next page
- "Create a component shell" on next page
- "Create component content (manual or automated)" on next page
- "Analyze components" on page 51

1. ## Background

   Suppose you are developing a banking application that processes loans.

   As with any application, the banker or banking manager needs to log in and log out of the application. The log in and log out activities are candidates to become business components because logging in and out are contained sets of operations that are reused.

   Similarly, other sets of banking activities could include:

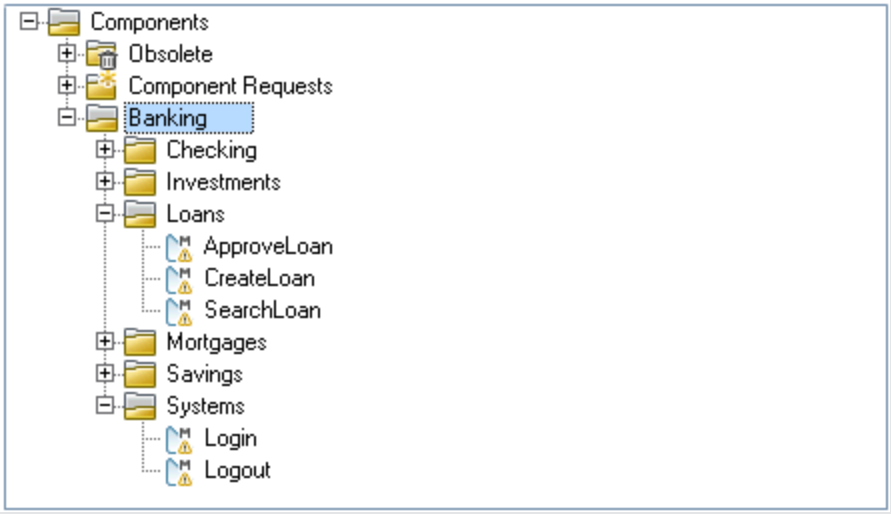
   - Creating a loan
   - Searching for a loan
   - Approving a loan

2. ## Create a component tree

   In the Business Components module, you create a component tree that represents various aspects of the banking application.

   Example

   Your component tree may be defined as follows:

3. ## Create a component shell

   In the Business Components module, create a component for creating a new loan for a customer.

   For user interface details, see "New Component/Component Details Dialog Box" on page 69.

4. ## Create component content (manual or automated)

   For task details on creating manual content, see "How to Work with Manual Design Steps" on page 80.

For task details on creating automated content, see "How to Work with Keyword GUI Automation" on page 93 or "How to Work with Scripted GUI and API Automation" on page 124.

Example

The following illustration shows a set of manual steps for a **CreateLoan** component.

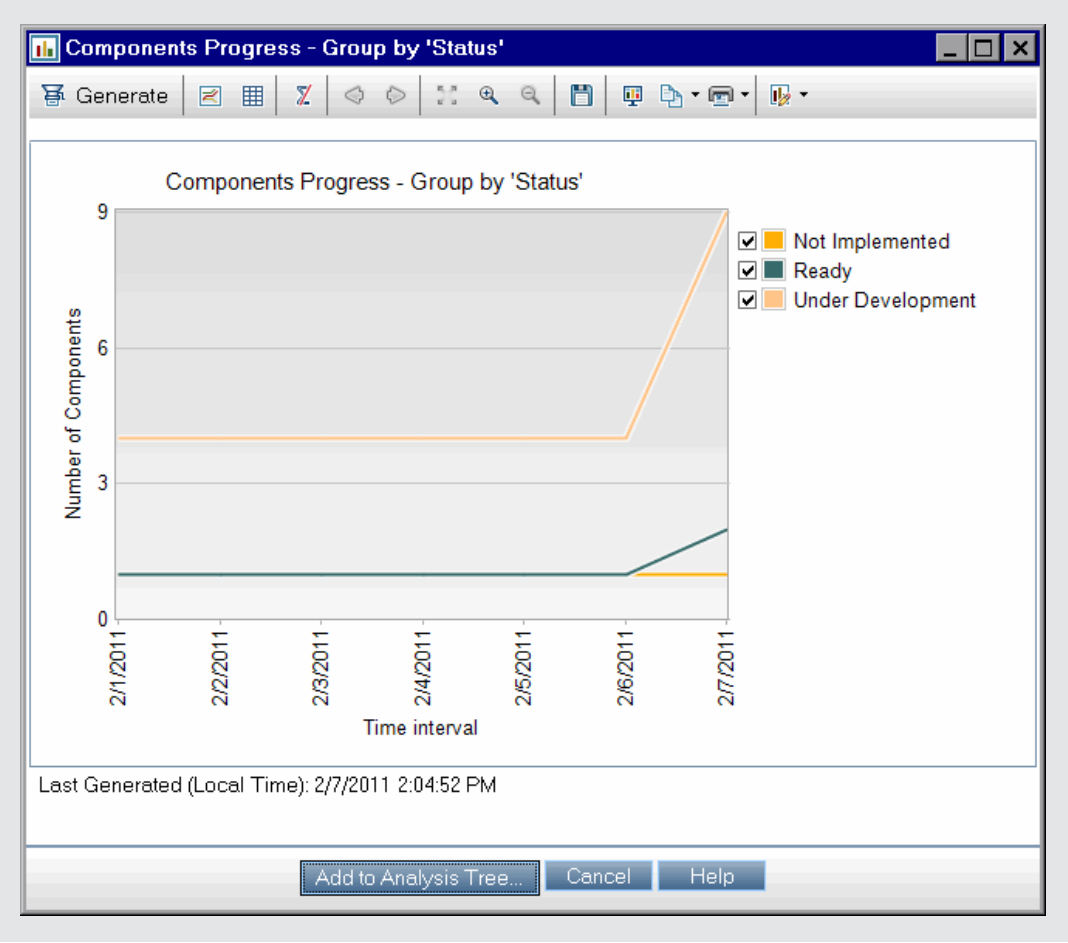
Step Name	Description	Expected Result
Step 1 Customer enters name	Customer enters the name: <<<CustomerName>>> and presses either <tab> or <Enter>.	The active field is now the CustomerPhone field.
Step 2 Customer enters phone number	Customer enters the phone number: <<<CustomerPhone>>> and presses either <tab> or <Enter>.	The active field is now the CustomerAddress field.
Step 3 Customer enters address	Customer enters the address: <<<CustomerAddress>>> and presses either <tab> or <Enter>.	The active field is now the Amount field.
Step 4 Customer enters amount	Customer enters the amount: <<<Amount>>> and presses either <tab> or <Enter>.	All fields are entered. Submit button is highlighted.
Step 5 Customer clicks Submit button	Customer clicks Submit.	The loan ID: <<<LoanID>>> is generated and displayed at the bottom right of the screen.

5. ## Analyze components

You can run various reports and graphs to analyze your components.

Example

To see the development status of your components, you could view the Components Progress graph:



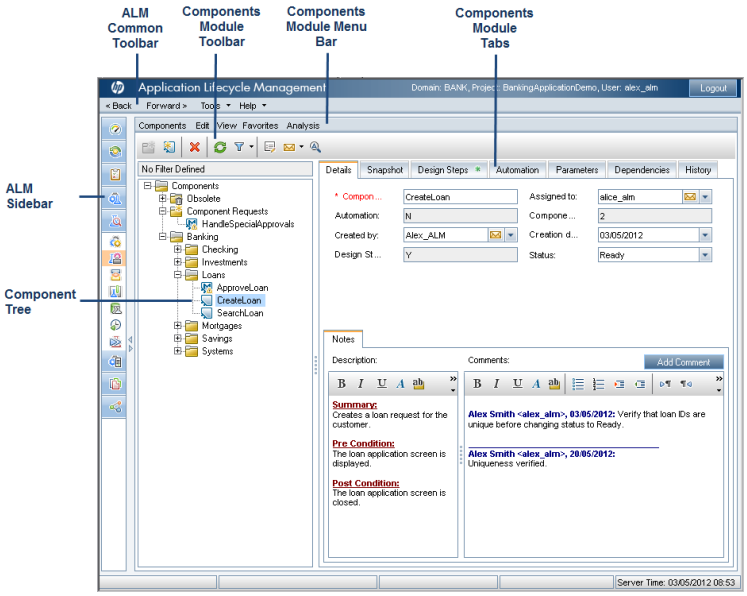
Business Components User Interface

This section includes:

Business Components Module Window 53
Business Components Module Menus and Buttons 56
Business Process Testing Icons 62
Business Components Module Fields 65
New Component/Component Details Dialog Box 69
Snapshot Tab 72
Paste Components/Component Folders to Target Project Dialog Box 74
Select Destination Folder Dialog Box 75

Business Components Module Window

This window enables you to define and maintain business components.



To access	On the ALM sidebar, under **Testing**, select **Business Components**.
Relevant tasks	• "How to Create Business Components in ALM" on page 46 • "How to Work with Manual Design Steps" on page 80 • "How to Work with Keyword GUI Automation" on page 93 • "How to Work with Scripted GUI and API Automation" on page 124
See also	"Business Components Overview " on page 40

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

UI Element (A-Z)	Description
<Business Components module common UI elements>	• **Business Component fields.** For field definitions, see "Business Components Module Fields" on page 65. • **Business Component menus and buttons.** For command and button descriptions, see "Business Components Module Menus and Buttons" on page 56. • **Business Component icons.** For icon descriptions, see "Business Process Testing Icons" on page 62. • ALM **main menu and sidebar.** For details on the Tools menu, Help menu and sidebar, see the information about the ALM Common User Interface in the *HP Application Lifecycle Management User Guide*.
<filter status bar>	Describes the filter currently applied to the tree. Located directly above the tree.
<component tree>	Located on the left side of the window, organizes and displays your business components hierarchically. The tree contains the following special folders: • **Components**. For user interface details, see "Components root folder" below. • **Obsolete**. For user interface details, see "Obsolete folder" on next page. • **Component Requests**. For user interface details, see "Component Requests folder" below.
Components root folder	Represents the top level of the component tree. Components cannot be created directly in the root folder. This folder cannot be copied within the same project, renamed, or deleted.
Component Requests folder	Contains requests for new components that were submitted from the Test Plan module. This folder cannot be renamed or deleted. In this folder, component requests can be edited but cannot be copied. Component requests can also be edited in the Test Plan module. When importing libraries, components in the **Component Requests** folder are not included in the process. **Version Control:** Version control is supported. For details on working with version control in ALM, see the *HP Application Lifecycle Management User Guide*. For task details on requesting components, see "How to Create Business Process Tests and Flows" on page 134.

UI Element (A-Z)	Description
Obsolete folder	Contains any components or component requests that have been removed but are still being used by one or more business process tests or flows. This folder cannot be renamed or deleted. Components in this folder: • Are read-only. • Cannot be copied. • Cannot be modified. • Cannot be added to other tests and flows. • Can only be deleted from the **Obsolete** folder if they are no longer being used by any business process test or flow. • When importing libraries, are not included in the process. • In a version control enabled project, cannot be checked in or out. For details on working with version control in ALM, see the *HP Application Lifecycle Management User Guide*. **Tip:** • You can simultaneously delete from the **Obsolete** folder all the business components that are no longer being used by any business process test or flow. Right-click the **Obsolete** folder and select **Clean unused components**. • You can restore business components from the **Obsolete** folder. For details, see "Restore" on page 61.
Details	Lists the component fields. For details on component fields, see "Business Components Module Fields" on page 65.
Snapshot	Captures and attaches a snapshot image from the application. For user interface details, see "Snapshot Tab" on page 72. Snapshots can be viewed and modified in the Business Component Settings dialog box in UFT. For details on the Business Component Settings dialog box, see the HP Unified Functional Testing User Guide.
Design Steps	Enables you to create or view the manual design steps of your business component. For user interface details, see "Design Steps Tab" on page 83.
Automation	Enables you to create or view the automation of your business component. For user interface details, see "Automation Tab" on page 98.

UI Element (A-Z)	Description
Parameters	Defines input and output component parameters and default parameter values for the business component. For user interface details, see "Parameters Page" on page 243.
Dependencies	Displays the dependency relationships between components, tests, flows, and resources (including application areas). For user interface details, see the *HP Application Lifecycle Management User Guide*.
History	Displays a log of changes made to the component. For user interface details, see the *HP Application Lifecycle Management User Guide*.
Live Analysis	When a folder is selected, this tab is available for creating a graphical representation of data related to business components. For user interface details, see the *HP Application Lifecycle Management User Guide*.

Business Components Module Menus and Buttons

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

To access	On the ALM sidebar, under **Testing**, select **Business Components**.
Important information	**Tip:** You can access the Business Components menu bar from the Business Components module by pressing the shortcut key ALT. Additionally, you can perform many other ALM operations using shortcut keys. For a list of ALM shortcut keys, see the *HP Application Lifecycle Management User Guide*. For a list of Business Process Testing shortcut keys, see "Business Process Testing Keyboard Shortcut Reference" on page 34. **Version Control:** Additional menu commands and buttons are available in a version control enabled project. For user interface details, see the *HP Application Lifecycle Management User Guide*.
Relevant tasks	• "How to Create Business Components in ALM" on page 46 • "How to Work with Manual Design Steps" on page 80 • "How to Work with Keyword GUI Automation" on page 93 • "How to Work with Scripted GUI and API Automation" on page 124 "How to Create Business Components in ALM" on page 46
See also	• "Business Process Testing Icons" on page 62 • "Business Components Overview " on page 40

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

UI Elements (A-Z)	Available from	Description
<list of recently used favorite views>	Favorites	Displays the four most recently used favorite views in the Business Components module. You can define the number of views displayed on the menu by setting the **FAVORITES_DEPTH** parameter in the Site Configuration tab in Site Administration. For details, see the *HP Application Lifecycle Management Administrator Guide*.
Add to Favorites	Favorites	Opens the Add Favorite dialog box, enabling you to add a favorite view to your private or the public folder. For details on working with favorites in ALM, see the *HP Application Lifecycle Management User Guide*.
Change Status	<right-click menu>	Changes the status of a component without switching to the Details tab. For details on component statuses, see "Business Components Module Fields" on page 65.
Clean unused components	Components	Simultaneously removes all components in the **Obsolete** folder that are not being used by any test or flow. **Available from: Obsolete** folder, when the folder is selected in the tree
Collapse	View	Collapses all branches in the component tree.
Component Details	<right-click menu>	Displays the Component Details dialog box, enabling you to display and modify the details of the selected component. For user interface details, see "New Component/Component Details Dialog Box" on page 69.
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 email or a document. Clicking the link opens ALM and takes you to the component. If you are not already logged in, ALM first prompts for login details.

UI Elements (A-Z)	Available from	Description
Copy/Paste	Edit	Copies selected components and/or folders (including subfolders) within the same project or across projects. **Note:** • Defect linkage data are not copied. • The root folder cannot be copied within the same project. • If you paste a component that has the same name as an existing component, the suffix **_Copy** and a sequential number are added to the end of the name. • When you copy a component between projects, associated entities can be copied using one of three methods. For user interface details, see "Paste Components/Component Folders to Target Project Dialog Box" on page 74. • **Version Control:** Before copying checked out entities, follow the guidelines described in the *HP Application Lifecycle Management User Guide*.
Cut/Paste	Edit	Moves selected components or folders to a different location in the component tree. The root folder cannot be moved. **Tip:** You can also move a component to a new location in the component tree by dragging it.

UI Elements (A-Z)	Available from	Description
Delete	Edit	Removes the selected folders or business components from the component tree. The contents of a removed folder are also removed. You cannot delete the root folder. Any removed component that is unused is permanently deleted. Any removed component that is used by one or more business process tests or flows is moved to the **Obsolete** folder in the tree. **Tip:** Business components can be restored from the **Obsolete** folder. For details, see "Restore" on page 61. **Note:** To check whether any business process tests or flows use a component, select the component and click the **Dependencies** tab. **Version Control:** Deleting a business component deletes all previous versions of the component.
Expand All	View	Expands all branches in the component tree.
Favorite bar	View	Displays the Favorite bar.
Filter bar	View	Displays the current filter.
Filter/Sort	View	Enables you to filter and sort the business components in the component tree. For user interface details, see the *HP Application Lifecycle Management User Guide*. For a list of fields available for filtering and sorting, see "Business Components Module Fields" on page 65.
Find	Edit	Searches for a component in the Components module. For user interface details, see the *HP Application Lifecycle Management User Guide*.
Find Next	Edit	Searches for the next item that meets the previously defined search criteria.

UI Elements (A-Z)	Available from	Description
Go to Component	Components	Opens the Go To Component dialog box, enabling you to find a specific component by the component ID. You can open the details dialog box for the component or you can go to the component's location in the tree. **Note:** You can only go to components that are in the current filter.
Graphs	Analysis	Lists the graphs you can generate for component data. Select a predefined graph, or launch the graph wizard. For details, see the *HP Application Lifecycle Management User Guide*.
Live Analysis	Analysis	Provides a graphical representation of data related to business components related to the currently selected component folder. For user interface details, see the *HP Application Lifecycle Management User Guide*. **Available from**: Components folder
Most Recently Used	Analysis	Displays the most recently viewed reports and graphs in the Components module.
New Component	Components	Opens the New Component dialog box, enabling you to add a component. For more details, see "New Component/Component Details Dialog Box" on page 69.
New Component Folder	Components	Opens the Create New Component Folder dialog box, enabling you to add a folder under a selected folder. **Syntax exceptions:** The folder name cannot contain two consecutive semicolons (**;**) or include any of the following characters: **\ * ^**
Organize Favorites	Favorites	Opens the Organize Favorites dialog box, enabling you to organize the list of favorite views by changing properties or deleting views. For details on working with favorites in ALM, see the *HP Application Lifecycle Management User Guide*.
Private	Favorites	Lists the favorite views that are accessible only to the user who created them.
Public	Favorites	Lists the favorite views that are accessible to all users.
Refresh All	View	Refreshes the component tree so that it displays the most up-to-date business components.

UI Elements (A-Z)	Available from	Description
Rename	Edit	Renames folders and business components in the component tree. You cannot rename the root, **Obsolete**, or **Component Requests** folders. You also cannot rename components in the **Obsolete** folder. **Syntax exceptions:** A component name cannot include the following characters: **\ / : " ? ' < > | * % ! { }**
Replace	Edit	Replaces a field value in the tree. For user interface details, see the *HP Application Lifecycle Management User Guide*.
Report Selected	<right-click menu>	Displays a report of the selected components.
Project Reports	Analysis	Lists the predefined reports you can produce on component data. For details, see the *HP Application Lifecycle Management User Guide*.
Restore	Components	Restores the selected components from the **Obsolete** folder to their original folder, if the folder still exists in the tree. You can view the folder in which the business component was located when it was last deleted in the **Original Location** field. The date that the component was deleted is displayed in the **Deleted on** field. **Note:** To restore components whose original folder no longer exists, select the component in the **Obsolete** folder and drag it to the appropriate folder in the component tree.
Send by E-mail	Components	Opens the Send E-mail dialog box, enabling you to send components by email to recipients selected from a list or to the author of the component. For user interface details, see the *HP Application Lifecycle Management User Guide*.
Text Search	Edit	Opens the text search pane in the lower part of the Components module window, enabling you to search for records in predefined fields. For user interface details, see the *HP Application Lifecycle Management User Guide*.
Update Selected	Edit	Opens the Update Selected dialog box, enabling you to update a field value for multiple selected components. For user interface details, see the *HP Application Lifecycle Management User Guide*.

Business Process Testing Icons

This section describes the icons available for Business Process Testing in the Business Components module, Test Plan module, and Test Lab module.

To access	On the ALM sidebar, under **Testing**, select **Business Components**, **Test Plan**, or **Test Lab**.
Important information	• The status of a business component affects the status of any business process tests or flows of which it is a part. The component with the most severe status determines the status of the test or flow. For example, a business component with an **Error** status causes any business process test or flow of which it is a part to have an **Error** status. • Additional statuses can be defined by updating the project list for business component statuses. For details, see the *HP Application Lifecycle Management Administrator Guide*.
See also	• "Business Components Module Menus and Buttons" on page 56 • "Business Components Module Window" on page 53

The icons are described below:

UI Element	Description
	<green asterisk>. The applicable tab is populated.
	Automated. A business component that has been automated using a testing tool such as UFT.
	Error. The red symbol indicates that the business component contains errors that need to be fixed, for example, due to a change in the application. When a business process test or flow contains a component with this status, the status of the entire business process test or flow is also **Error**.
	Learned. **Business Process Testing Enterprise Extension**: The business component was created using the learn flow process.
	Maintenance. The yellow triangle indicates that this business component was previously implemented and is now being modified.
	Manual. A manual business component.
	Not Implemented. This icon indicates that a manual business component has been requested in the Test Plan module. The component request status changes from **Not Implemented** to **Under Development** when you move the request from the **Component Requests** folder in the component tree. **Note:** This status is available from the Test Plan module.
	Obsolete. The component is obsolete, cannot be modified, and should not be used in business process tests or flows.
	Ready. The green indicates that this business component is fully implemented and ready to be run.
	Under Development. The yellow triangle indicates that this business component is currently under development. This status is assigned to: • New components • Component requests dragged or moved out of the **Component Requests** folder into a component folder
	Comment. The information on the current row in the automated component's Keyword View is a comment. **Available from**: UFT Keyword View in the Automation tab

UI Element	Description
	Missing Object. The automated component contains an object that is missing from the shared object repository. **Available from**: UFTKeyword View in the Automation tab
	Constant. The automated component contains an output value configured as a constant. **Available from**: UFT Keyword View in the Automation tab
	Component Parameter. The automated component contains an output value configured as a component parameter. **Available from**: UFT Keyword View in the Automation tab
	Local Parameter. The automated component contains an output value configured as a local parameter. **Available from**: UFT Keyword View in the Automation tab
	Version Control. The component is checked out. • **Green lock.** The component is checked out by the current user. • **Red lock.** The component is checked out by another user. For details on working with version control in ALM, see the *HP Application Lifecycle Management User Guide*.

Business Components Module Fields

This section describes the Business Components module fields.

To access	On the ALM sidebar, under **Testing**, select **Business Components**.
Important information	• You can add user-defined fields and change the label of the fields in the Business Components module. You can also customize project lists. For details, see the *HP Application Lifecycle Management Administrator Guide*. • You can use the Script Editor to restrict and dynamically change the fields and values in the Business Components module. For details, see the *HP Application Lifecycle Management Administrator Guide*. • **Version Control:** Additional fields are available in a version control enabled project. For user interface details, see the *HP Application Lifecycle Management User Guide*.
Relevant tasks	• "How to Create Business Components in ALM" on page 46 • "How to Work with Manual Design Steps" on page 80 • "How to Work with Keyword GUI Automation" on page 93 • "How to Work with Scripted GUI and API Automation" on page 124
See also	"Business Components Module Window" on page 53

User interface elements are described below:

UI Element (A-Z)	Description
Assigned To	The user to whom the business component implementation is assigned. **Converted ALM manual tests**: This field remains empty.
Automation	Indication that the component has automation.
Comments	Comments about the business component. A toolbar for formatting and spell checking the text. **Converted ALM manual tests**: The description is copied from the Description tab of the ALM manual test.
Component ID	A unique numeric ID for the component, assigned automatically by ALM. This is a read-only field.
Component Name	The name of the selected component. **Syntax exceptions**: Business component names cannot exceed 255 characters, contain two consecutive semicolons (;), or include any of the following characters: **! % * { } \ | ` : " / < > ?** Spaces at the beginning or end of business component names are ignored. **Converted ALM manual tests**: The name you enter in the **New Component Name** box in the Destination Folder dialog box. The default is the name of the test. If a component with the same name already exists in the destination folder, a suffix, for example **_Copy_1**, is automatically added to the name to create a unique name within the folder.
Created by	The user that created the business component. **Converted ALM manual tests**: The user that converted the test.
Creation date	The date that the business component was created. **Converted ALM manual tests**: The date the test was converted.
Deleted on	The date that the business component was deleted. This is a read-only field. **Available for:** Components in the **Obsolete** folder

UI Element (A-Z)	Description
Description	A description of the business component. You can include information such as: • A summary of the business component's purpose or contents. • The intended pre-condition of the application at the start of the component run. • The intended post-condition of the application at the end of the component run. Examples of pre-conditions: • Which applications should be open or closed • Whether the component should launch an application • The state of the application prior to the first step in the component Examples of post-conditions: • Whether the component steps should close any applications. • The state of the application after the last step in the component. If you want the component to allow iterations, the post-conditions should specify that the application's end state should match its state at the beginning of the component. This field displays a toolbar for formatting and spell checking the text. **UFT:** • For keyword GUI components, you can also use this area to specify additional functionality that needs to be provided by the automation engineer in the form of new, registered operations. • The description can be viewed in the Business Component Settings dialog box in UFT. **Converted ALM manual tests**: The description is copied from the Description tab of the ALM manual test. The name of the source test and the test ID are also displayed.
Design Steps	Indication that the component has manual design steps.
Notes	The area containing the **Description** and **Comments** fields.
Original Location	The folder in which the business component was located when the component was deleted. This data is automatically entered by ALM. **Available for:** Components in the **Obsolete** folder

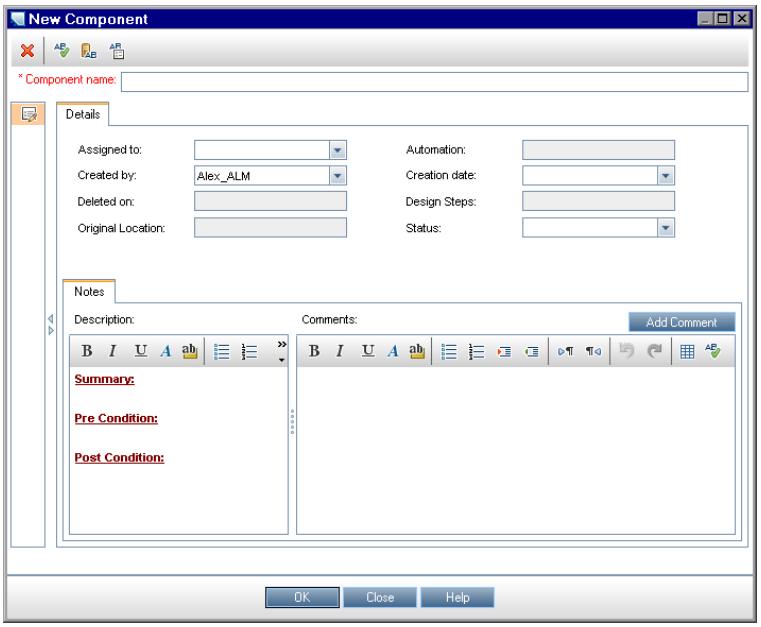
UI Element (A-Z)	Description
Status	The status of the business component: • **Ready.** The component is complete and ready for use in a business process test or flow. • **Maintenance.** The component is being modified, for example, to adapt it for changes made in the application. • **Not Implemented.** A component request has been initiated in the Test Plan module. • **Under Development.** One of the following applies: ▪ The component has been created in the Business Components module. ▪ The component has been converted from an ALM manual test. ▪ A component request has been dragged into a component folder in the component tree. • **Error.** The component does not run successfully and needs to be debugged. For icons representing business component statuses, see "Business Process Testing Icons" on page 62. Additional statuses can be defined by updating the project list for business component statuses. For details, see the *HP Application Lifecycle Management Administrator Guide*. The status can also be modified by the automation engineer in the Business Component Settings dialog box in UFT.
Subtype ID	The facet of the component. A facet is a container that holds the manual design steps and/or automation of the business component. The **Subtype ID** is read-only, and is displayed in the Design Steps and Automation tabs. Valid subtype IDs are: • **MANUAL**. Manual design steps. This value is available in the Design Steps tab. • **QT-KW**. Keyword GUI automation. This value is available in the Automation tab. • **QT-SCRIPTED**. Scripted GUI automation. This value is available in the Automation tab. • **SERVICE-TEST-AUTOMATED**. Scripted API automation. This value is available in the Automation tab.

New Component/Component Details Dialog Box

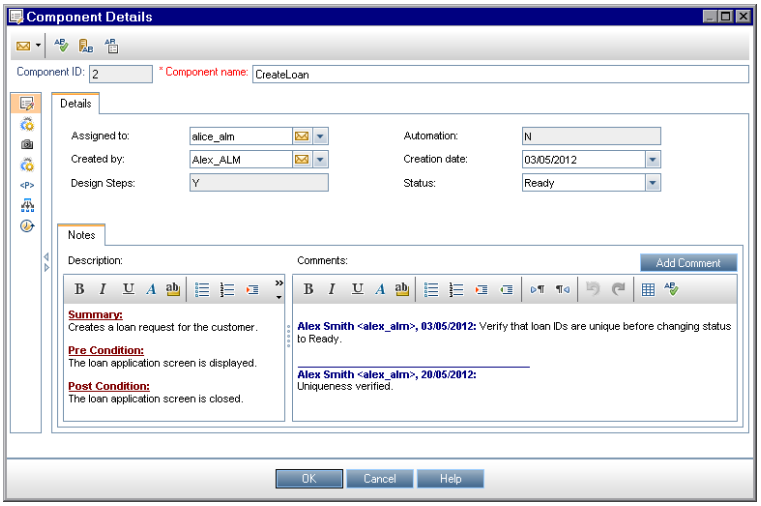
The New Component dialog box enables you to define a new component.

The Component Details dialog box enables you to view and update the details of the selected component.

New Component Dialog Box



Component Details Dialog Box



To access	• **New Component dialog box:** In the Business Components module, right-click a component folder and select **New Component**. • **Component Details dialog box:** In the Business Components module, right-click a component and select **Component Details**.
Relevant tasks	• "How to Create Business Components in ALM" on page 46 • "How to Work with Manual Design Steps" on page 80 • "How to Work with Keyword GUI Automation" on page 93 • "How to Work with Scripted GUI and API Automation" on page 124
See also	• "Components with Manual Content" on page 41 • "Manual Design Steps Overview" on page 80 • "Components with Automated Content" on page 41

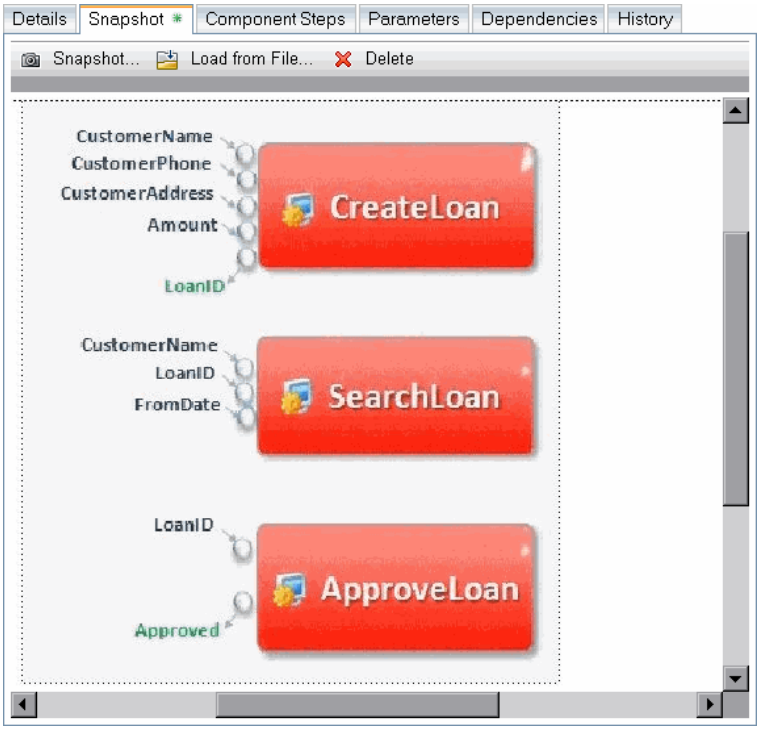
User interface elements are described below:

UI Element	Description
	Clear All Fields. Clears the data in the dialog box.
	Send by E-mail. Sends an email with the component details. For user interface details, see the *HP Application Lifecycle Management User Guide*.
	Spell Check. Checks the spelling for the selected word or text box.
	Thesaurus. Displays a synonym, antonym, or related word for the selected word.
	Spelling Options. Enables you to configure the way ALM checks spelling.
Component ID	ID assigned to the component when the component was created. This data is automatically entered by ALM, and cannot be modified. **Available from:** Component Details dialog box
Component name	Name for the new business component.
Details	Required fields are displayed in red. For details on the available component fields, see "Business Components Module Fields" on page 65.
Snapshot	Captures and attaches a snapshot image from the application. For user interface details, see "Snapshot Tab" on next page. Snapshots can be viewed and modified in the Business Component Settings dialog box in UFT. For details on the Business Component Settings dialog box, see the *HP Unified Functional Testing User Guide*. **Available from:** Component Details dialog box
Design Steps	Enables you to create or view the manual design steps of your business component. For user interface details, see "Design Steps Tab" on page 83. **Available from:** Component Details dialog box
Automation	Enables you to create or view the automation for your business component. For user interface details, see "Automation Tab" on page 98. **Available from:** Component Details dialog box
Parameters	Defines input and output component parameters and default parameter values for the business component. For user interface details, see "Parameters Page" on page 243. **Available from:** Component Details dialog box

UI Element	Description
Dependencies	Displays the dependency relationships between components, tests, flows, and resources (including application areas). For user interface details, see the *HP Application Lifecycle Management User Guide*. **Available from:** Component Details dialog box
History	Displays a log of changes made to the component. For user interface details, see the *HP Application Lifecycle Management User Guide*. **Available from:** Component Details dialog box

Snapshot Tab

This tab enables you to attach an image associated with the business component.



To access	From the Business Components module, click the **Snapshot** tab.

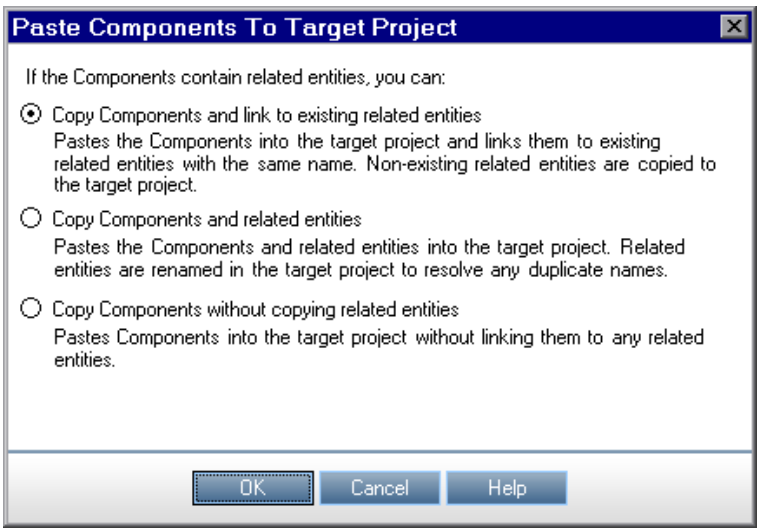
Important information	• You can view the image that you attach or load in the Snapshot tab in the Test Script tab of the Test Plan module by clicking the relevant thumbnail image. Viewing the sequence of these images can help you to better understand the business process test or flow. • The image can also be viewed or replaced in the Business Component Settings dialog box in UFT. • You can capture and attach an image associated with the business component using the Snapshot dialog box, or you can load a previously saved **.png**,**.jpg**, **.gif**, or **.bmp** image from your local drive. • Only one image can be attached to a business component.
Relevant tasks	• "How to Work with Manual Design Steps" on page 80 • *HP Application Lifecycle Management User Guide*
See also	*HP Application Lifecycle Management User Guide*

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

UI Element	Description
	Snapshot. Opens the Snapshot dialog box. For user interface details, see the *HP Application Lifecycle Management User Guide*.
	Load from File. Enables you to load a previously saved **.png**, **.jpg**, **.gif**, or **.bmp** file containing the snapshot using the Select Snapshot dialog box.
	Delete. Deletes the currently attached snapshot.
<Snapshot area>	Area containing the snapshot.

Paste Components/Component Folders to Target Project Dialog Box

This dialog box enables you to select the way in which components and component folders are copied from one project to another project.



To access	1. In the source project, right-click a component or component folder and select **Copy**. 2. Open the target project as the same user in a separate browser window. 3. In the target project, right-click a folder below which you want to insert the copied component or component folder, and select **Paste**.
Important information	• Run information is not copied. • The source and target projects must use the same ALM version and patch level. • If the component path in the target project already exists, and a business component in a test or flow matches the name and signature of an existing component (meaning the number of parameters, and so on), the existing component is linked to the business process tests or flows that include it, and the original component is not copied. • When you copy a business component, its associated application area is also copied. However, if an application area with the same name already exists in the target, the application area is not copied, and the new component is linked to the existing application area. Check to ensure that the application area in the new location has the correct settings and resource files.
See also	"Business Components Module Window" on page 53

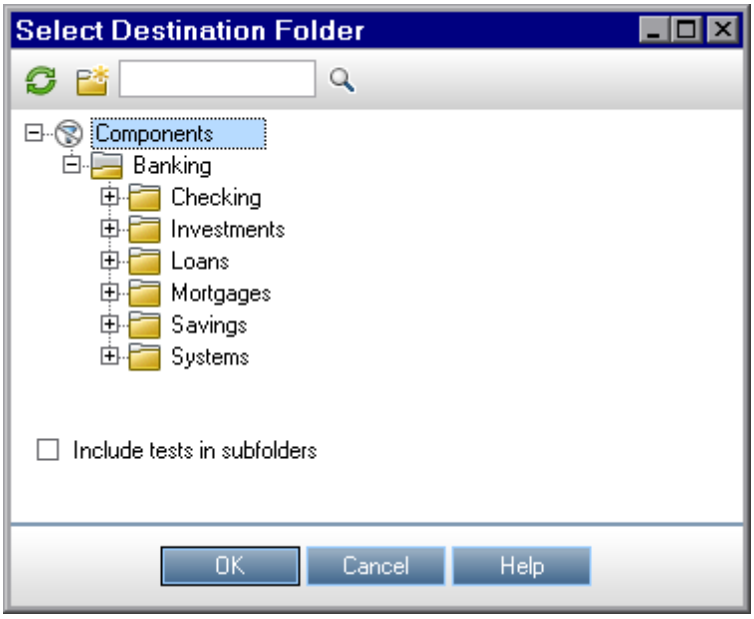
User interface elements are described below:

UI Element	Description
Copy components/component folders and link to existing related entities	ALM copies the components or component folders and pastes them into the target project. The copied components or component folders are linked to related entities with the same name and path. If an entity does not exist in the target project, ALM copies it to the target project and links it to the component.
Copy components/component folders and related entities	ALM copies the components or component folders along with the related entities and pastes them into the target project. If a related entity already exists in the target project, the copied entity is renamed to resolve the duplicate name.
Copy components/component folders without copying related entities	ALM copies the components/component folders without copying related entities, and pastes them into the target project. The copied items are not associated with any entities.

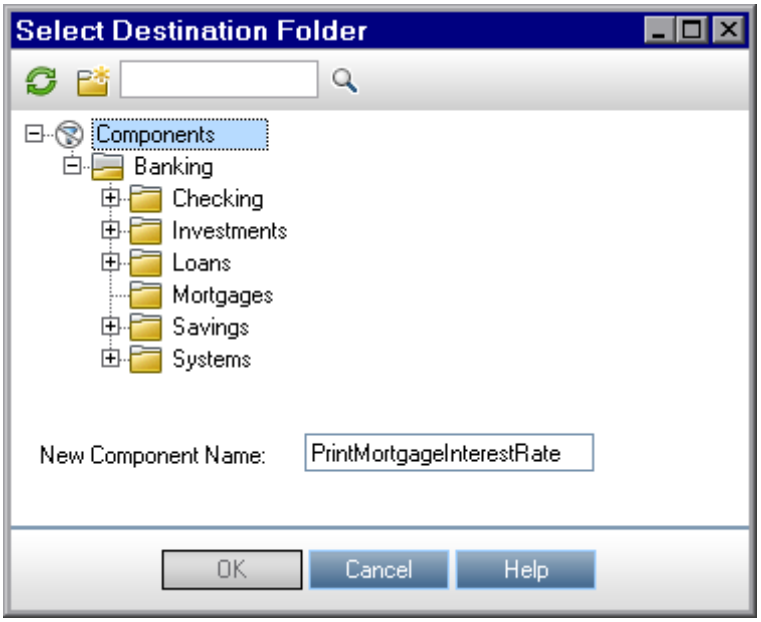
Select Destination Folder Dialog Box

This dialog box enables you to convert existing manual ALM tests (not business process tests) to manual components in a destination folder that you select.

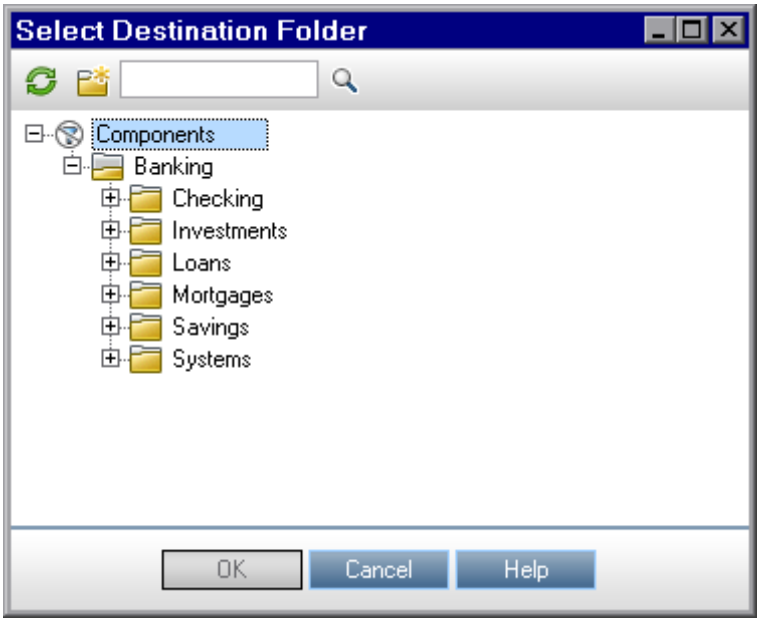
Select Destination Folder Dialog Box with a Folder Selected



Select Destination Folder Dialog Box with a Single Test Selected



Select Destination Folder Dialog Box with Multiple Tests Selected



To access	From the Test Plan module, select the ALM manual test that you want to convert. Click the **Convert to Component** button.
Relevant tasks	• "How to Work with Manual Design Steps" on page 80 • *HP Application Lifecycle Management User Guide*

Considerations when Converting ALM Manual Tests to Components

Entity	Considerations
Parameters	• Test parameter names may contain special characters that are not permitted in a component parameter name. Such characters are replaced with an underscore (_) when the test is converted. • If a parameter name does not start with an English letter, the letter **p** is added as a prefix to the name when the test is converted. • The parameter default value is copied from the test parameter. • The parameter description is copied from the test parameter.
Steps	In the Design Steps tab of the new component: • Step names and expected results are copied from the steps in the test. • Parameter names may be changed during the conversion process. Parameter names in the step descriptions are changed accordingly. • Certain test steps may call another test. If a component step is of type "call to", it is copied as a regular step with the step name "**Call**". The description is displayed as "**Call <linked test name>**".
Other	Attachments, linked defects, and the test history are not copied over to the converted component.

User interface elements are described below:

UI Element	Description
	Refresh. Updates the data in the tree.
	New Folder. Opens the New Component Folder dialog box, enabling you to add a folder under a selected folder in the component tree. **Syntax exceptions**: The folder name cannot contain two consecutive semicolons (;) or include any of the following characters: **\ * ^**
	Find. Searches the component tree by name. Type the name of the folder or test in the **Find** box and click **Find**. ALM highlights the first entity in the tree that contains the search string.
<component tree>	Displays the components in a hierarchical format that can be collapsed and expanded.

UI Element	Description
New Component Name	Enables you to enter a new name for a converted component. The default is the name of the test. This field is mandatory. **Note:** If a component with the same name already exists in the destination folder, a suffix is automatically added to the name to create a unique name within the folder (for example **_Copy_1**). **Available when**: Either a single test is selected in the test plan tree
Include tests in subfolders	Enables you to convert all the tests in the selected folder and all of its subfolders simultaneously. If the number of tests is large, this action may take some time. **Available when**: A folder is selected in the test plan tree

Chapter 4

Components with Manual Design Steps

This chapter includes:

Manual Design Steps Overview. 80

How to Work with Manual Design Steps. 80

Manual Design Steps User Interface. 82

Manual Design Steps Overview

In the Business Components module, you can create or modify manual business component content by adding manual design steps.

Steps represent operations that should be performed on your application when you run the component in a business process test or flow. Each step comprises a step name, a textual description of the procedure to be performed on the application, and the expected result (if any), of performing that step. When a business process test or flow runs in ALM, the design steps defined in the component are performed.

Tip:

- You can use your design steps as guidelines for creating automated steps at a later time, using an applicable testing tool.
- You can use your design steps even before the application is ready to be tested automatically, or before automated components are complete.

The following example demonstrates the application steps of a **Login** component for a Web application:

Example:

Step	Description	Expected Result
1	Open the application.	The application is launched and the login page is displayed.
2	Enter a user name.	The cursor advances to the password field.
3	Enter a password.	The password is displayed as asterisks.
4	Click **Submit** on the Web page.	The application's main page is displayed.

When designing a manual test (or, before automated testing resources are available for an automated test), you add design steps in the Design Steps tab of each manual component. You run manual tests using the Manual Runner or Sprinter.

For task details, see "How to Work with Manual Design Steps" below.

How to Work with Manual Design Steps

This task describes how to create and update a component's manual design steps in the Business Components module.

To learn more about creating manual components, see "Manual Design Steps Overview" above.

Note:

- **Higher-level task:** This task is part of a higher-level task. For task details, see "How to Create Business Components in ALM" on page 46.

This task includes the following steps:

- "Select a component" below
- "Add steps" below
- "Parameterize steps - optional" below
- "Export steps - optional" on next page
- "Results" on next page

1. ## Select a component

   In the Business Components module window, select a component in the component tree.

2. ## Add steps

   A step is a detailed operation to be performed on an application as part of testing the business process. Each step includes a description of an action and the expected results.

   You add steps In the Design Steps tab. If there are existing steps, select the step below which you want to add a new step. For user interface details, see "Design Steps Tab" on page 83.

   Click **New Step** . The Component Step Details dialog box is displayed. For user interface details, see "Design Steps Pane/Component Step Details Dialog Box" on page 83.

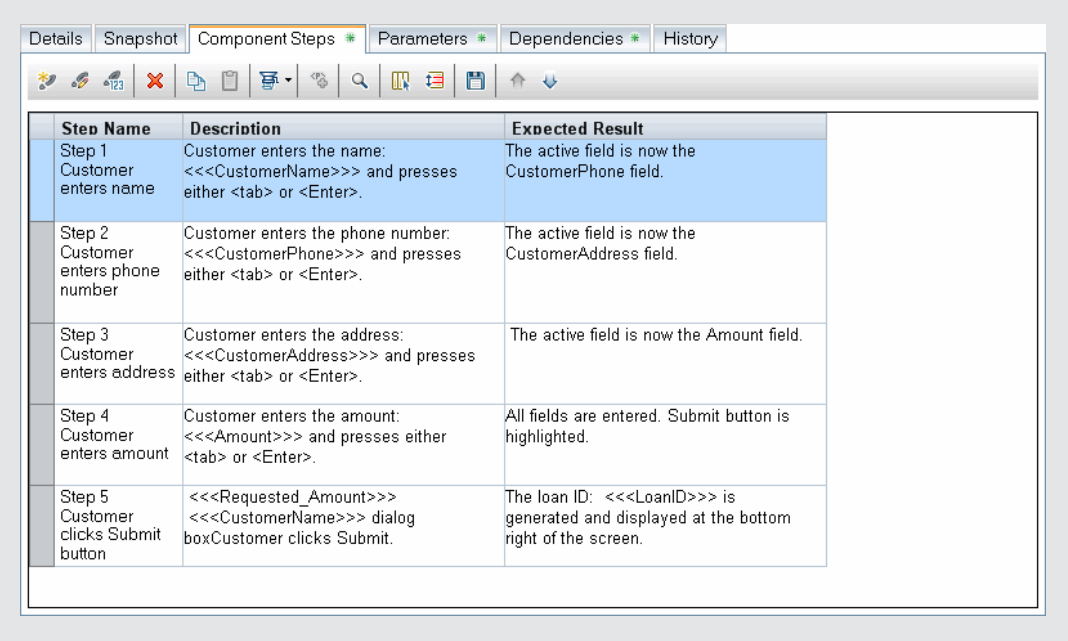
3. ## Parameterize steps - optional

   You can insert parameters into a step's description and expected results.

   To add a parameter to a step, in the Component Step Details dialog box, click **Insert Parameter** . For task details on working with parameters, see "How to Create Parameters" on page 238.

   Example:

   To test the loan request page in the banking application, you might create steps as follows:



Step Name	Description	Expected Result
Step 1 Customer enters name	Customer enters the name: <<<CustomerName>>> and presses either <tab> or <Enter>.	The active field is now the CustomerPhone field.
Step 2 Customer enters phone number	Customer enters the phone number: <<<CustomerPhone>>> and presses either <tab> or <Enter>.	The active field is now the CustomerAddress field.
Step 3 Customer enters address	Customer enters the address: <<<CustomerAddress>>> and presses either <tab> or <Enter>.	The active field is now the Amount field.
Step 4 Customer enters amount	Customer enters the amount: <<<Amount>>> and presses either <tab> or <Enter>.	All fields are entered. Submit button is highlighted.
Step 5 Customer clicks Submit button	<<<Requested_Amount>>> <<<CustomerName>>> dialog boxCustomer clicks Submit.	The loan ID: <<<LoanID>>> is generated and displayed at the bottom right of the screen.

4. **Export steps - optional**

   From the Design Steps tab, you can export the step information to several formats, including text file, Microsoft Word document, Microsoft Excel spreadsheet, or HTML document.

   For user interface details on exporting steps using the **Export** command in the Design Steps tab/Component Step Details dialog box, see the **Export** UI element in the "Design Steps Pane/Component Step Details Dialog Box" on next page.

5. **Results**

   The steps are listed in the Design Steps tab.

Manual Design Steps User Interface

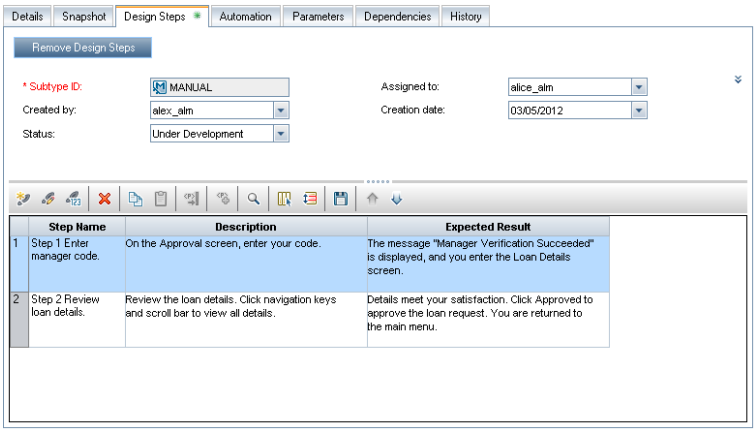
This section includes:

Design Steps Tab 83

Design Steps Pane/Component Step Details Dialog Box 83

Design Steps Tab

This tab enables you to create and edit manual design steps for components.



To access	From the Business Components module, click the **Design Steps** tab. From the Component Details dialog box, click the **Design Steps** sidebar entry.
Relevant tasks	• "How to Work with Manual Design Steps" on page 80
See also	• "Business Components Overview " on page 40 • "Components with Manual Content" on page 41 • "Manual Design Steps Overview" on page 80

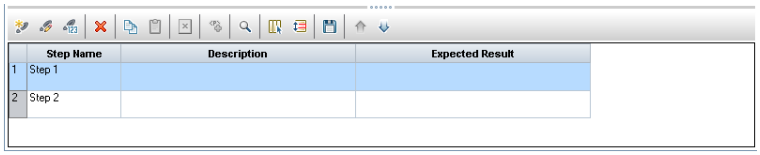
User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Add Design Steps Remove Design Steps	Enables you to add or delete manual design steps to/from the business component.
<Business Component fields>	For field definitions, see "Business Components Module Fields" on page 65.
<Design steps pane>	Enables you to create and edit manual steps. For user interface details, see "Design Steps Pane/Component Step Details Dialog Box" below.

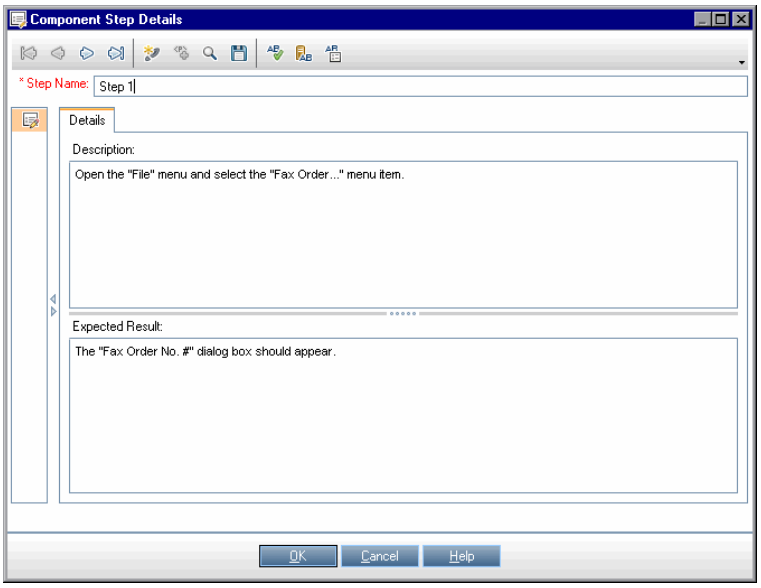
Design Steps Pane/Component Step Details Dialog Box

The Design Steps pane and the Component Step Details dialog box enable you to create and edit individual steps.

Design Steps Pane



Component Step Details Dialog Box

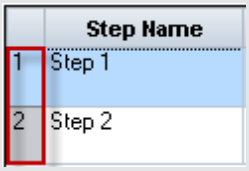


To access	**Design Steps pane:** In the Business Components module, select a component in the component tree. Click the **Design Steps** tab. **Component Step Details dialog box:** In the Business Components module, select a component in the component tree. In the Design Steps tab, do one of the following in the Design Steps pane: • Right-click the step grid, or a step, and select **New Step**. • Right-click a step and select **Edit Step**.
Important information	• This pane or dialog box enables you to create manual design steps for your component in an editable table. You can describe each step to be performed, create and insert parameters where required, and describe the expected result of the step. • If you apply formatting to a parameter name in a step, you must apply the same formatting to the entire parameter name, including the **<<<** and **>>>** characters. For example, if you want to italicize the parameter `password`, you must italicize the entire string `<<<password>>>` and not just the word `password`. • Specifying less than 2,500 characters in user-defined fields in this tab or dialog box can improve performance.
Relevant tasks	"How to Work with Manual Design Steps" on page 80
See also	"Manual Design Steps Overview" on page 80

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

UI Element	Description
<Business Component fields>	For field definitions, see "Business Components Module Fields" on page 65. **Available from:** Design Steps pane
	Allows you to browse through manual steps. **Available from:** Component Step Details dialog box
	New Step. Creates a new step. • **From the Design Steps pane.** Opens the Component Step Details dialog box, enabling you to add a step below the selected step. • **From the Component Step Details dialog box.** Clears most of the Component Step Details dialog box fields, enabling you to enter details for a new step. Increments the value of the **Step Name** field by one (for example, **Step 1** becomes **Step 2**).
	Edit Step. Opens the selected step in the Component Step Details dialog box. **Tip:** Double-click a step to edit. **Available from:** Design Steps pane
	Renumber Steps. Renumbers the steps sequentially. Use this command after you rearrange steps. **Available from:** Design Steps pane
	Delete Selected. Deletes the selected step(s).
	Copy/Paste Steps. Copies the selected steps and pastes them. The copied steps are inserted above the selected step. When you paste steps to another project, the Paste steps to Target Project dialog box opens. For user interface details, see "Paste Components/Component Folders to Target Project Dialog Box" on page 74. **Note:** Parameters included in the step are copied to the target. **Available from:** Design Steps pane
	Complete Parameter Name. After you enter the first characters of a parameter name, clicking this button fills in the rest of the parameter name and encloses the parameter name between **<<< >>>** brackets.
	Insert Parameter. Inserts a parameter into the step. This command is enabled only when your focus is in the **Description** or **Expected Result** cells. For details on inserting parameters, see "How to Create Parameters" on page 238.

UI Element	Description
	Find Text. Opens the Find dialog box, enabling you to search in the tab for steps that contain a specified text value. Select the column in which you want to search, and enter the value you want to find. For user interface details, see the *HP Application Lifecycle Management User Guide*.
	Select Columns. Opens the Select Columns dialog box, enabling you to specify which columns to display their order. You can also set the number of columns that remain stationary when you scroll horizontally. **Available from:** Design Steps pane
	Adjust row heights. Changes the row height of the step so that more or less of the text in the step is visible. **Available from:** Design Steps pane
	Save steps. Saves the steps. Steps are not saved until either you click this button or you leave the pane or dialog box.
	Move Up. Moves the selected step up in the list. **Available from:** Design Steps pane
	Move Down. Moves the selected step down in the list. **Available from:** Design Steps pane
	Thesaurus. Displays a synonym, antonym, or related word for the selected word. **Available from:** Component Step Details dialog box
	Check Spelling. Checks the spelling for the selected word or text box. **Available from:** Component Step Details dialog box
	Spelling Options. Enables you to configure the way ALM checks spelling. **Available from:** Component Step Details dialog box
Export	This right-click menu command opens the Export All Grid Data dialog box for exporting the step information to several formats (text, Microsoft Word, Microsoft Excel, HTML). Specify a location, file name, and file type for the exported data. **Available from:** Design Steps pane

UI Element	Description
<Step grid>	Displays steps that are included in the selected component. Each row in the step grid is numbered sequentially. **Example**  **Available from:** Design Steps pane
Step Name	Displays the name of each step. **Default value:** The sequential number of the step. For example, `Step 1` if you are adding a step to a component for the first time.
Description	Displays the description of each step. Parameters included in the Description column are surrounded by **<<< >>>** brackets. **Tip:** • Clicking in this field displays a toolbar for formatting and spell checking the text instructions for the tester to perform. • Specifying less than 2,500 characters in user-defined fields in this tab or dialog box can improve performance.
Expected Result	Displays the expected state after the step instructions are performed. **Note:** Parameters included in the **Expected Result** column are surrounded by **<<< >>>** brackets. **Tip:** • Clicking in this field displays a toolbar for formatting and spell checking the text. • Specifying less than 2,500 characters in user-defined fields in this tab or dialog box can improve performance.

Chapter 5

Components with Keyword GUI Automation

This chapter includes:

Keyword GUI Automation Overview 90

How to Work with Keyword GUI Automation 93

Keyword GUI Automation User Interface 97

Keyword GUI Automation Overview

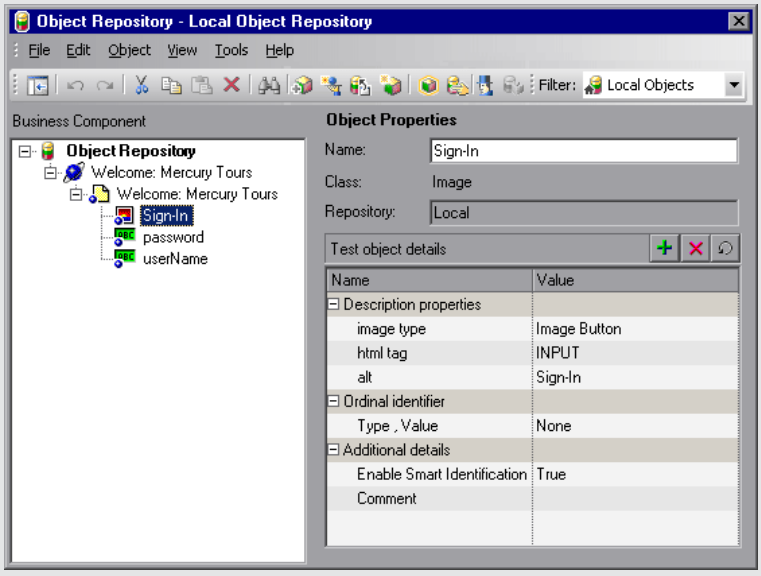
Keyword GUI automated components are automated, reusable modules that perform a specific task when testing your application. The tasks are defined as keyword steps. These steps can be viewed and edited both in ALM and in UFT.

ALM provides an Automation tab > Keyword View pane, which enables the subject matter expert to create and modify automated keyword steps. For user interface details on the Keyword View, see "Keyword View Pane" on page 99.

Each step is made up of an item (an object in the application or an operation), and an operation. The available items and operations are defined by the automation engineer in the object repository and the function libraries. These objects and operations can be saved in ALM, making them available for insertion into steps by the subject matter expert.

Example:

The objects in a Sign-in login window may be captured in the object repository in Unified Functional Testing as follows:



For a sample workflow of using Business Process Testing and Unified Functional Testing keyword GUI automation together, see "Business Process Testing with Unified Functional Testing Workflow" on page 29.

Note: This chapter describes how to use several Business Process Testing features. HP Business Process Testing Enterprise Extension provides additional, automated features that enable you to test enterprise applications. Therefore, some procedures described in this chapter may not be part of the recommended best practice when using Business Process Testing Enterprise Extension to test your enterprise application.

This section includes the following topics:

- "Object Overview" below
- "Operation Overview" below
- "Operation Values and Their Arguments" below
- "Keyword View Parameterization" on next page
- "Output Value Overview" on next page
- "Comment Overview" on page 93
- "Checkpoint Overview" on page 93

Object Overview

Tests perform operations on objects. Objects are contained in the shared object repository that is associated with the application area on which your business component is based.

Objects can be sibling and child objects of the previous step's object.

Note: If an object included in a step is later removed from the shared object repository by the automation engineer, the run of the automated component fails.

Operation Overview

Operations, or **keywords**, are customized functions that may group together multiple steps, as well as programming logic that can perform a specific task in your application. For example, you can use an operation to open an application at the start of a business component, or check the value of a specific property of an object in your application.

Business Process Testing provides a set of basic keywords for you to use. The automation engineer can add to or enhance the keywords in UFT. These keywords are derived from built-in methods and properties, as well as user-defined functions associated with the application area. The automation engineer manages the keywords and selects which of them should be available in the **Select an Item** list when creating automated components.

Note: If an operation included in a step is later removed from the library by the automation engineer, the run of the automated component fails.

For details on managing keywords, see the information about GUI testing and the Keyword View in the *HP Unified Functional Testing User Guide*.

Operation Values and Their Arguments

An operation **argument** provides the specific information that an operation uses to accomplish a specific purpose.

A particular operation may have required arguments, optional arguments, or no arguments at all.

Example:

- The argument of an edit box Set operation indicates the specific value to enter in the edit box.
- The argument numbers for a Click operation can optionally indicate the specific coordinates of the click.

In the keyword view, **Value** cells are partitioned according to the number of possible arguments of the selected operation. Each partition contains different options, depending on the type of argument that can be entered in the partition.

Keyword View Parameterization

You can define input parameters that pass values into your business component, and output parameters that pass values from your component to external sources or from one step to another step. You can also define local parameters for use within the component only. You can then use these parameters to parameterize input and output values in automated steps. You can define the following parameters from within a keyword step:

- **Local parameter.** A local parameter is defined for a specific business component. It is not accessible by other business components. You define local parameters in the "Keyword View Pane" on page 99 using the "Value Configuration Options Dialog Box" on page 113 for input parameters and the "Output Options Dialog Box" on page 110 for output parameters. You cannot delete local parameters, but you can cancel their input or output.
- **Component parameter.** A component parameter whose value is set can be returned to or retrieved from a location outside the component. The values of these parameters can be passed from one component in a business process test or flow to a later one during a test run. You define component parameters in the "Parameters Page" on page 243 of the Business Components module. For task details on working with component parameters, see "How to Handle Data in Business Process Testing" on page 235.

After you define a parameter you can use it to parameterize a value. You can also change the parameterized value back to a constant value by entering it directly in the **Value** cell.

Output Value Overview

For steps that return an output value, you can define settings that determine where the output value is stored and how it is used during the component run session. When the output value step is reached, each value set for output is retrieved and stored in the specified location for use later in the run session.

Output values are stored only for the duration of the run session. When the run session is repeated, the output values are recaptured.

After the run session, you can view the values retrieved during the session as part of the session results. For details, see "View run results" on page 307.

You can select a number of properties to output for the same object and define the output settings for each property value. When the output value step is reached during the run session, UFT retrieves all of the specified property values.

If, after you specify an output value, you choose not to save the output value, you can cancel it.

Business Process Testing Enterprise Extension: Business Process Testing Enterprise Extension users can also add output values to a component during the Learn Flow process.

Comment Overview

A comment is a free text entry that you can enter on a new line below the currently selected step.

Comments are not processed when business components run.

You can use comments for several purposes.

- You may want to add comments to a business component to improve readability and make it easier to update. For example, you may want to add a comment before each section of a component's automated steps to specify what that section includes.
- You may want to use comments to plan steps to be included in a business component before your application is ready to be tested. Then, when the application is ready, you can use your plan (comments) to verify that every item that needs to be tested is included in the steps.

Note: After you insert a comment, you cannot change it to a step.

Checkpoint Overview

When creating a component, automation engineers working in UFT can add standard checkpoints. A checkpoint compares the current value of a specified property with the expected value for that property, and can help identify whether your application is functioning correctly.

When a component containing one or more checkpoints runs, UFT compares the expected value of the checkpoint to the actual value. If the value does not match, the checkpoint fails. You can view the results of the checkpoint in the Run Results Viewer. For details, see "View run results" on page 307.

Business Process Testing Enterprise Extension: Business Process Testing Enterprise Extension users can also add object property checkpoints to a component during the Learn Flow process.

How to Work with Keyword GUI Automation

This task describes how to automate a component by adding keyword GUI steps to the component.

Note:

- This task is part of a higher-level task. For details see "Business Process Testing Workflow" on page 28.
- For a sample workflow of using Business Process Testing and keyword GUI automation together, see "Business Process Testing with Unified Functional Testing Workflow" on page 29.

This task includes the following steps:

- "Prerequisites" on next page
- "Create an application area" on next page

- "Select a component" on next page
- "Add keyword GUI automation" on next page
- "Select the application area" on next page
- "Add content (steps)" on page 96
- "Parameterize steps" on page 96
- "Enhance steps - optional" on page 96
- "Results" on page 97

1. ## Prerequisites

   - Make sure a testing tool, such as UFT, is installed.

     Tip: If a testing tool is not installed, the

button in the Automation tab is disabled.

   - Install the Unified Functional Testing Add-in for Business Process Testing on the client computer. The Unified Functional Testing Add-in for Business Process Testing is installed from the HP Application Lifecycle Management Add-ins page.
   - To be able to debug or run automated keyword GUI components, or view test results, make sure that one of the following is installed:
     - QuickTest Professional 10.00 or 11.00
     - UFT 11.50 or later

2. ## Create an application area

   Automating components involves the creation of application areas, which provide access to the settings and resources required for a particular application.

   a. Make sure that your application is ready to be tested, and all of the required automation resources have been prepared.

   b. In UFT, if not yet connected, connect to the ALM project (for which you want to create the application area).

      For details, see the information about connecting to ALM projects in the *HP Unified Functional Testing User Guide*.

   c. Do one of the following:

      - Create a basic application area based on a default template in the ALM Test Resources module. The default application area template is stored in the **System Application Areas** subfolder and cannot be moved or modified. An automation engineer can then enhance the application area in UFT.

        The default application area contains very limited operation options from which to

- Work with an automation engineer to create the application area in UFT.

d. In ALM, create and upload the application area resource in the Test Resources module. You can also create other resources with which the application area is associated, such as libraries, object repositories, and recovery scenarios in ALM (and associate them to the application area in UFT). For details, see the *HP Application Lifecycle Management User Guide*.

e. In UFT, define the settings and resource files that you want business components associated with the application area to use. By associating a component with an application area, the component is automatically linked to these settings and resource files.

f. In UFT, define dependencies between entities. For details, refer to the *HP Unified Functional Testing User Guide*.

Tip: You can view the associations between the application area and the other resources in the Dependencies tab in the Test Resources module in ALM.

3. ## Select a component

In the Business Components module window, select a component in the component tree.

4. ## Add keyword GUI automation

In the Automation tab, click

and select **Keyword GUI**.

The Keyword View pane is displayed. For user interface details, see "Keyword View Pane" on page 99.

5. ## Select the application area

Note:

A keyword GUI component must be associated with an application area to create steps for it, but you can also change the application area when required.

Changing an application area may affect the automated component and prevent it from running correctly. For example, if a step uses an object that is not contained in the object repository of the new application area, the step fails.

In the Keyword View pane, select an application area if the following message appears next to the toolbar buttons:

```
Choose an application area by clicking the 'Select Application
Area' button
```

Select an application area by clicking . For user interface details on the Select Application Area dialog box, see "Select Application Area Dialog Box" on page 104.

All application objects in the associated object repository, and any operations defined in associated function libraries, are now available when automating steps.

6. ## Add content (steps)

Create the steps required to test your application.

Each step is an operation to be performed, and is defined as a row in the Keyword View pane.

For each step you add or modify, you specify:

- The item, such as an object from the application area's shared object repository, on which the step is performed.
- The operation (keywords specified in the application area) to be performed on the item.
- Any required values.
- Output values.
- Documentation, such as comments.

Save the steps.

Tip:

- Since changes you make in the Keyword View pane are not saved automatically, it is recommended that you save them periodically.
- A component is not fully automated until all of its steps are automated. For example, if you specify an **Operation** item with an **Operation** value of **ManualStep**, the automation is not yet complete. When running the component, there is a pause for user input.

For conceptual details on the entities used to create steps, such as items and operations, see "Keyword GUI Automation Overview" on page 90.

For user interface details on the Keyword View pane, see "Keyword View Pane" on page 99.

7. ## Parameterize steps

To expand the scope of your tests, flows, and business components, use variable input and output parameter values in your automated components.

To parameterize...	Use the...
Input values for a step using local or component parameters	"Value Configuration Options Dialog Box" on page 113
Output values for a step	"Output Value Properties Dialog Box" on page 118
Expected property values from the Checkpoint Properties dialog box or the Output Value Properties dialog box	"Output Value Properties Dialog Box" on page 118

Save the steps.

8. ## Enhance steps - optional

You can enhance your automated components by adding the following enhancements to your steps.

Enhancement	Description
Comments	Comments provide additional information in the form of free-text. You can insert comments between, and as placeholders for, step operations. For conceptual details about comments, see "Comment Overview" on page 93. For user interface details, see "<Comment Row>" on page 104.
Checkpoints	Checkpoints compare the current value of a specified property with the expected value for that property, and can help identify whether your application is functioning correctly. For conceptual details, see "Checkpoint Overview" on page 93. The checkpoint is visible and editable in the Checkpoint Properties dialog box in ALM, enabling you to view which properties are set to be checked during the run. For user interface details, see "Checkpoint Properties Dialog Box" on page 115.
Output Values	For steps that return a value, output values store settings that determine where the output value is stored and how it is used during the component run session. When the output value step is reached, each value set for output is retrieved and stored in the specified location for use later in the run session. For conceptual details about output values, see "Output Value Overview" on page 92. For user interface details, see "Output Value Properties Dialog Box" on page 118.

Save the steps.

9. **Results**
   - The automation of the component is complete.
   - The icon for the component changes from the manual icon to the automated icon. For icon details, see "Business Process Testing Icons" on page 62.
   - The automated component can be accessed from UFT.
   - Tests and flows that contain fully automated components can run in an automatic runner without pausing for user input.

Keyword GUI Automation User Interface

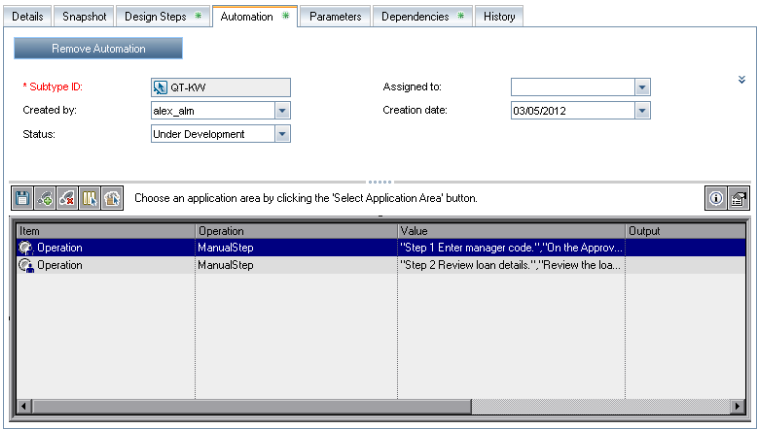
This section includes:

Automation Tab 98
Keyword View Pane 99
Select Application Area Dialog Box 104

Keyword View Options Dialog Box ... 105
Select Test Object Dialog Box ... 108
Output Options Dialog Box ... 110
Value Configuration Options Dialog Box ... 113
Checkpoint Properties Dialog Box ... 115
Parameterization / Properties Dialog Box ... 117
Output Value Properties Dialog Box ... 118
Information Dialog Box ... 120
General Information Dialog Box ... 121

Automation Tab

This tab enables you to implement automation for a component.



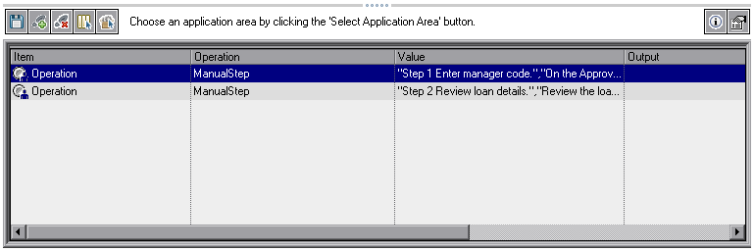
To access	From the Business Components module, click the **Automation** tab. From the Component Details dialog box, click the **Automation**sidebar entry.
Important information	The contents of the Automation tab changes depending on the automation implemented for the component.
Relevant tasks	• "How to Work with Keyword GUI Automation" on page 93 • "How to Work with Scripted GUI and API Automation" on page 124
See also	• "Business Components Overview " on page 40 • "Components with Automated Content" on page 41 • "Keyword GUI Automation Overview" on page 90 • "Scripted GUI and API Automation Overview" on page 124

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

UI Element	Description
Add Automation Remove Automation	Enables you to add or remove automation to/from the business component. When adding automation, click the down arrow next to the button and choose the type of automation: • Keyword GUI • Scripted GUI • API For details on the automation types, see "Automation Types" on page 42.
<Business Component fields>	For field definitions, see "Business Components Module Fields" on page 65.
<Keyword View pane>	Enables you to create and edit keyword steps. For user interface details, see "Keyword View Pane" below.

Keyword View Pane

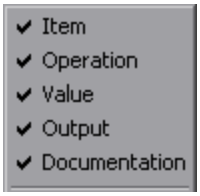
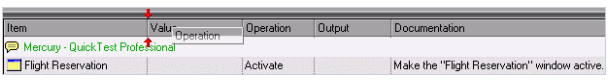
This pane enables you to create and modify keyword GUI steps in a keyword view.

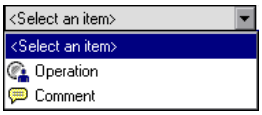


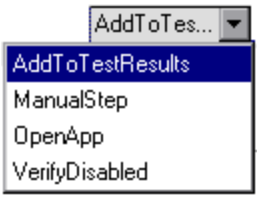
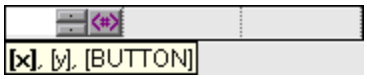
To access	1. In the Business Components module, select a component with Keyword GUI automation. 2. Click the **Automation** tab. The Keyword View pane is displayed in the lower half of the window.
Important information	• The Keyword View in ALM provides similar functionality to the Business Component Keyword GUI view in UFT. For a full description of the functionality provided in UFT, see the *HP Unified Functional Testing for HP Business Process Testing User Guide*. • You can choose how the Keyword View appears, such as which columns to display. For user interface details, see "Keyword View Options Dialog Box" on page 105. • Business components that are currently open in another session of ALM or in UFT are locked and read-only.

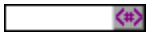
Relevant tasks	"How to Work with Keyword GUI Automation" on page 93
See also	• "Keyword GUI Automation Overview" on page 90 • *HP Unified Functional Testing User Guide*

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

UI Element	Description
	Save. Saves the component steps.
	Add Step. Adds a step after the currently-selected row and opens the **Select an Item** list. **Tip:** This button can be used to add a comment.
	Delete Step. Deletes the current step. Before you delete a step, make sure that removing it does not prevent the component from running correctly. You cannot delete a step if one of its cells is in edit mode. **Note:** This button can be used to delete a comment. When deleting a comment, a warning message is not displayed.
	View Options. Enables you to set how ALM displays the keyword view. For user interface details, see "Keyword View Options Dialog Box" on page 105
	Select Application Area. Enables you to select an application area for the automation from the application area resources uploaded in the Test Resources module.
	**<column name context menu options>**.Hides or displays the column with the same name as the context menu option.
Column header row	Provides the name for each column. You can rearrange columns by dragging a column header and dropping it at a new location. Red arrows are displayed when the column header is dragged to an available location. 

Item column	Lists items, such as: • **Objects**, from the shared object repository, on which you perform an operation. For conceptual details, see "Object Overview" on page 91. • **Operation indicator**, signifying that the **Operation** column will contain keyword GUIs (user-defined functions) from a function library or libraries associated with the component's application area. For conceptual details, see "Operation Overview" on page 91. • **Comments**, which can be used to: ▪ Add a manual step ▪ Add other text information between steps ▪ Add notes about the component ▪ Provide information about adjacent steps
	**Select an Item** list. Enables you to select the following: • **Objects**. • **Operation** option. Selects the **Operation** item, and enables you to select, in the **Operation** column, keyword GUIs (user-defined functions) from a function library or libraries associated with the component's application area. • **Comment** option. Adds a comment. • **Select another object** option. Used if the object for which you want to add a step is not displayed in the **Select an Item** list, this option opens the Select Test Object dialog box. This dialog box enables you to select an item other than a sibling or child object of the previous step's object from the shared object repository. For user interface details, see "Select Test Object Dialog Box" on page 108. You must select an option from the **Item** list before you can add additional content to a step. If you select an object, you then specify an operation for it. **Note:** If no application area is associated with the component, or no objects are available, only the **Operation** and **Comment** items are included in the **Select an Item** list. Some operation functions, such as **OpenApp**, are available without an associated application area.
Operation column	Lists the operation to be performed on the object selected in the **Item** column. **Tip:** When you position the cursor over an operation in the list, a tooltip describes the operation. For conceptual details, see "Operation Overview" on page 91.

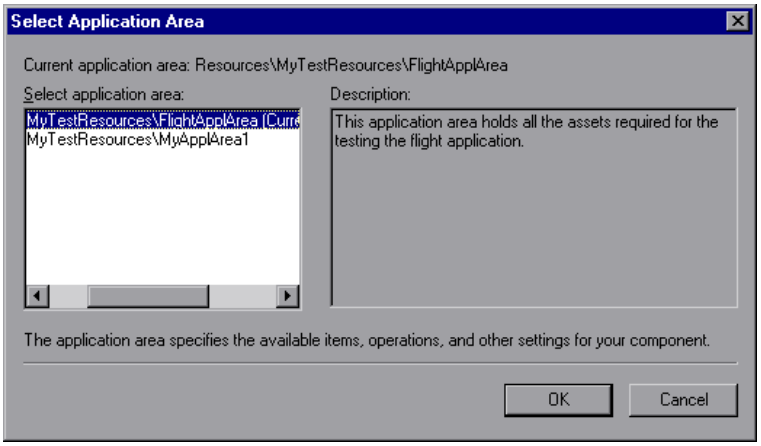
	**Select an Operation** list. Enables you to select the operation to be performed on the object selected in the **Item** column. The available operations vary according to the item selected: • If you select an application object in the **Select an Item** list, the **Operation** list for that object includes built-in operations and any operations that were created for that specific object type in the application area's function libraries. For example, if you selected a button object, the list contains the most commonly used methods available for the button object, such as **Click**. • If you select **Operation** in the **Item** column, the list contains the operations defined in the function library or libraries associated with the component's application area. For details on function libraries, see the *HP Unified Functional Testing User Guide*. The default operation (most commonly-used operation) for the item is automatically displayed in the **Operation** column. Operations for the object are available only if functions were added by the automation engineer to a function library that is associated with the application area on which your business component is based.
Value column	Lists argument values for the selected operation (if relevant). The value is partitioned according to the number of possible arguments of the selected operation. Each partition contains different options, depending on the type of argument that can be entered in the partition: **String**, **Integer**, **Boolean**, or **Predefined Constant**. For conceptual details, see "Operation Values and Their Arguments" on page 91. When you click in a **Value** cell, a tooltip displays information for each argument. In the tooltip, the argument for the partition that is currently highlighted is displayed in bold, and any optional arguments are enclosed in square brackets. In the example below, the **x** argument is displayed in bold because that argument is currently highlighted.  The **Value** column cell also contains a parameterization button for parameterizing the value. **Available when**: An operation has been selected that needs additional arguments specified.

	**String** argument value. Enables you to specify a string as an argument value for an operation. Enter any alphanumeric string enclosed by quotes. If you do not enter the quotes, Business Process Testing adds them automatically. If you modify a cell that contains a string enclosed by quotes by removing the quotes, Business Process Testing does not restore the quotes, and the value is treated as a variable name. **Available from:** An argument partition in a **Value** cell
	Integer argument value. Enables you to specify an integer as an argument value for an operation. Enter any number, or use the up and down arrows to select a number. **Available from:** An argument partition in a **Value** cell
	Boolean argument value. Enables you to specify a true/false value as an argument value for an operation. Enter any **True** or **False**, or use the up and down arrows to toggle between these values. **Available from:** An argument partition in a **Value** cell
	Predefined constant argument value. Enables you to select a value from the list. **Available from:** An argument partition in a **Value** cell
	Parameterization button. Enables you to parameterize the value for an argument using a local or component parameter. For user interface details, see "Output Value Properties Dialog Box" on page 118. **Available from:** An argument partition in a **Value** cell
	Cancel button. Cancels stored output values. For user interface details on storing output values, see "Output Value Properties Dialog Box" on page 118. **Available from:** An **Output** cell for a step with an output parameter
	Checkpoint button. Opens the Checkpoint Properties dialog box, for accepting or modifying property values of the checkpoint. For user interface details, see "Checkpoint Properties Dialog Box" on page 115. **Available from:** An argument partition in a **Value** cell for a checkpoint step
	Specify parameter for output button. Opens the Output Options dialog box, which enables you to configure output parameters. For user interface details, see "Output Options Dialog Box" on page 110.
Output column	Lists the parameter in which output values for the step are stored (if relevant).

Documentation column	Provides a read-only, plain-language textual description of what the step does. **Examples**: • `Click the "Sign-in" image` • `Select "San Francisco" in the "toPort" list.` The documentation is automatically displayed after you have entered the item, operation, and any required values for the selected operation. You can use these Documentation descriptions to run your automated components manually. For details, see "How to Run Business Process Tests and Flows Manually" on page 304.
<Comment Row>	Comments span an entire row in the Keyword View. After you add a comment, it is always visible in your component, as long as one or more columns are displayed. In addition, as you scroll from side to side across the tab, the comment can always be seen.
Documentation Only context menu option	Toggles the display of only the Documentation column or all columns. Select this option, for example, if you want to use the steps as instructions for manual testing. The Documentation column and any comments defined in the business component are displayed.
Insert Comment context menu option	Adds a comment after the currently-selected row.
Copy / Cut / Paste / Delete context menu options	Performs these standard operations on the currently-selected rows in the Keyword View.

Select Application Area Dialog Box

This dialog box enables you to select an application area for the purposes of automating UFT keyword GUI components.



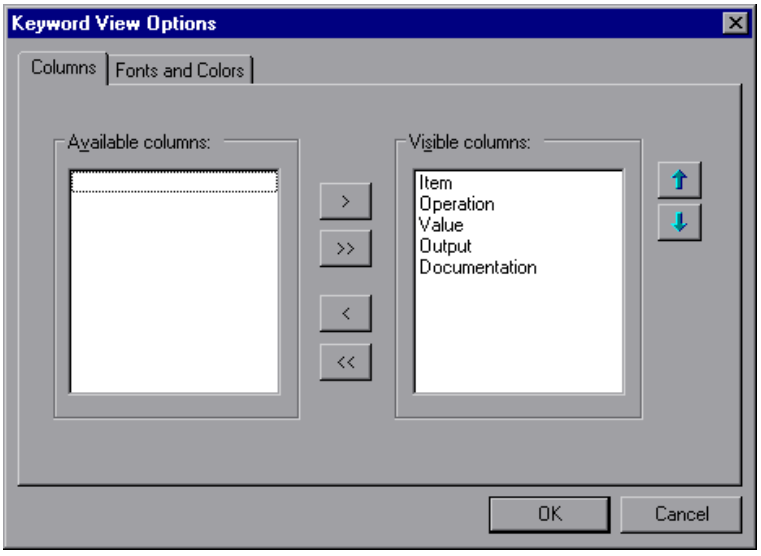
To access	1. In the Business Components module, select a component with Keyword GUI automation. 2. Click the **Automation** tab. 3. Click **Select Application Area** .
Important information	• You may be prompted to connect to the project from within UFT the first time you try to select an application area. For instructions, see the *HP Unified Functional Testing User Guide*. • Changing an application area may affect the business component and prevent it from running correctly. For example, if a component step uses an object that is not contained in the object repository of the new application area, the step fails.
Relevant tasks	"How to Work with Keyword GUI Automation" on page 93
See also	"Application Area Overview" on page 43

User interface elements are described below:

UI Element	Description
Current application area	Displays the name of the selected application area.
Select application area	Lists the application areas available for selection.
Description	Describes the selected application area.

Keyword View Options Dialog Box

This dialog box enables you to set how ALM displays the keyword view.



To access	1. In the Business Components module, select a component with Keyword GUI automation. 2. Click the **Automation** tab. 3. Click **View Options** .
Relevant tasks	"How to Work with Keyword GUI Automation" on page 93
See also	• "Keyword View Pane" on page 99 • *HP Unified Functional Testing User Guide*

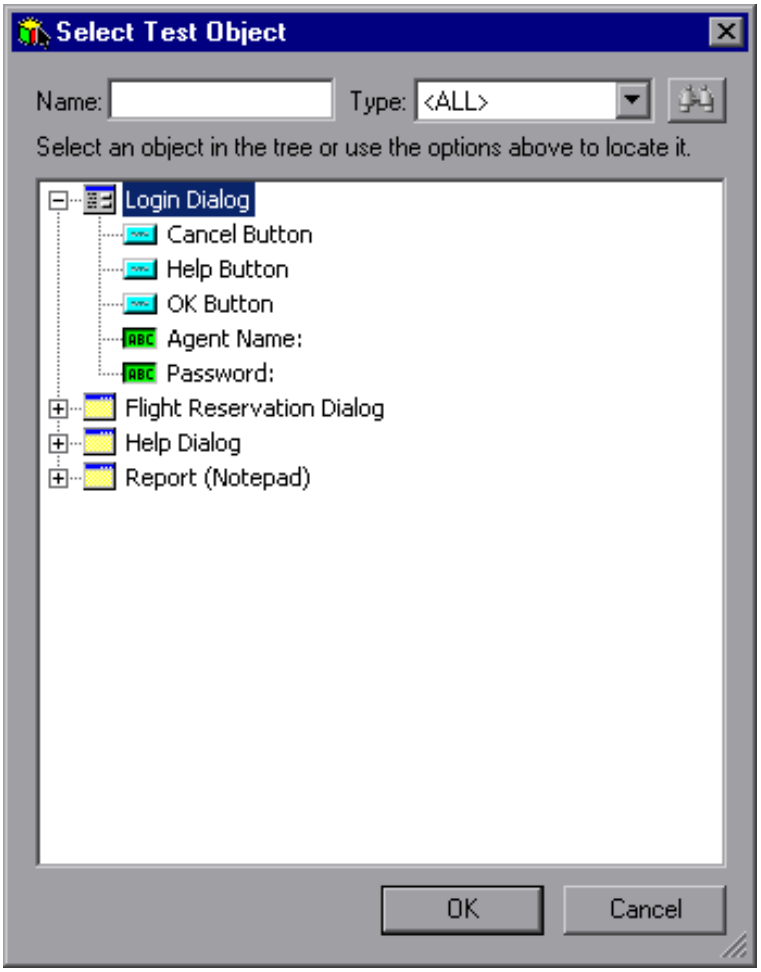
User interface elements are described below:

UI Element	Description
Columns	Provides options for setting the columns to display, and their order, in the Keyword View.
Fonts and Colors	Sets color and font options for the Keyword View.
Available columns	Lists columns not currently displayed in the Keyword View. **Available from**: The Columns tab
	Moves column names between the **Available columns** and **Visible columns** boxes. **Available from**: The Columns tab
Visible columns	Lists columns currently displayed in the Keyword View. **Available from**: The Columns tab
	Sets the order in which columns appear. Select a column in the Visible columns box and click the up or down button to place the column in the desired order. **Note:** The order of the columns in the Keyword View does not affect the order in which the cells need to be completed for each step. For example, if you choose to display the **Operation** column to the left of the **Item** column, you select the item first, and the **Operation** column list is then refreshed to match the selection you made in the **Item** column. **Available from**: The Columns tab

Element	Selects the rows for which your font and color selections apply. • **Default.** Applies your selections to all rows. • **Selected Row.** Applies your selection of text and background color to the currently selected row only. • **Alternate Rows.** Applies your selection of a background color to alternate rows. • **Comment.** Applies your selections to comment rows. **Available from**: The Fonts and Colors tab
Font name	Selects the font in which the text displays. **Available from**: The Fonts and Colors tab
Size	Selects the font size in which the text displays. **Available from**: The Fonts and Colors tab
Style	Selects the font style; **Regular**, **Bold**, **Italic**, or **Underline**. **Available from**: The Fonts and Colors tab
Foreground	Selects the color in which the text displays. **Available from**: The Fonts and Colors tab
Background	Selects the background color of the rows. **Available from**: The Fonts and Colors tab
Foreground for read-only	Selects the color in which read-only text displays. **Available from**: The Fonts and Colors tab
Reset all	Changes all your selections back to the default. **Available from**: The Fonts and Colors tab

Select Test Object Dialog Box

This dialog box enables you to select objects from the shared object repository that are not displayed in the **Select an Item** list when creating automated component steps. For example, you can select an item other than a sibling or child object of the previous step's object.



To access	1. In the Business Components module, select a component with Keyword GUI automation. 2. Click the **Automation** tab. 3. Click in an **Item** cell, and then click the arrow to display the **Select an Item** list. **Note:** If you have just created a new step, the list opens automatically as soon as you create the new step. 4. Select **Select another object**.

Important information	After selecting an object from the shared object repository tree, the selected object is displayed in the **Item** cell and is also added to the **Select an Item** list.
Relevant tasks	"How to Work with Keyword GUI Automation" on page 93
See also	*HP Unified Functional Testing User Guide*

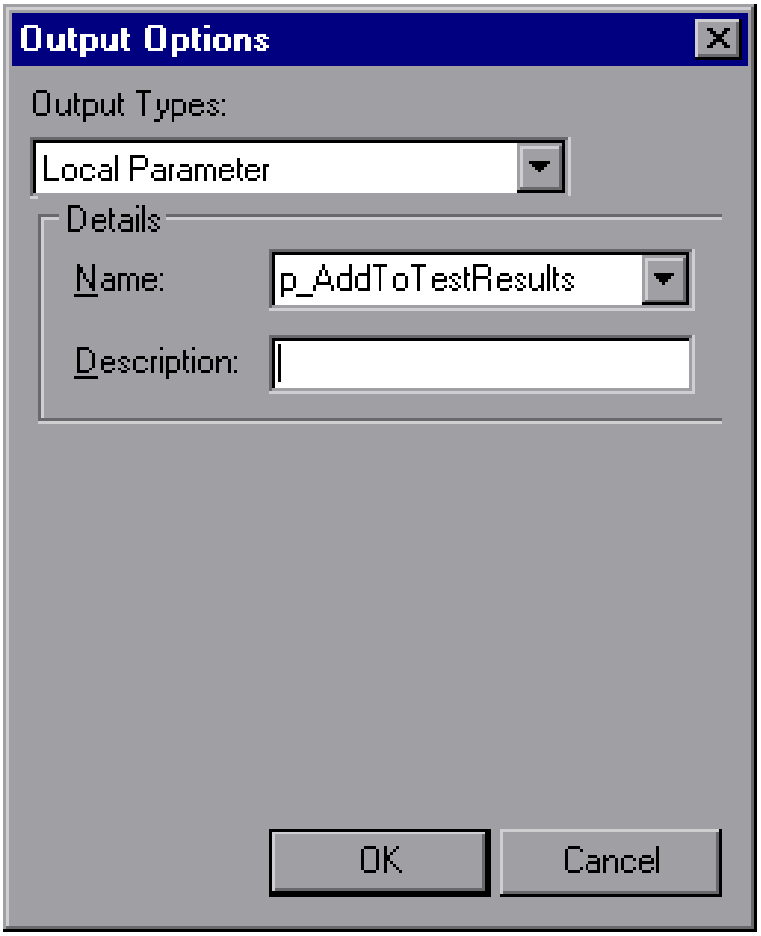
User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Name	Enables you to find an object in the shared object repository tree by entering the object name (or the beginning letters of the name) and then clicking **Find Next**.
Type	Enables you to limit the number of objects displayed in the shared object repository tree by selecting the type of object you require, for example, **Check Box**, or **Button**.
	Find Next. Enables you to find an object in the shared object repository tree according to the object name, specified in the **Name** box.
<shared object repository tree>	Lists the objects in the shared object repository.

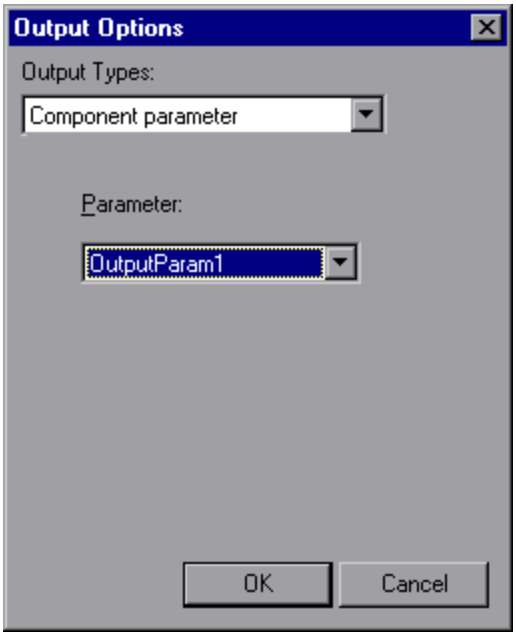
Output Options Dialog Box

This dialog box enables you to parameterize output values for a step using local or component parameters in the Output cell of a step. You can then use the output parameter value as an input value in a later step in the component, or in a later component or flow in the test.

Output Options Dialog Box for Local Parameters



Output Options Dialog Box for Component Parameters



To access	1. In the Business Components module, select a component with Keyword GUI automation. 2. Click the **Automation** tab. 3. Click in an **Output** cell of a step that returns an output value. 4. Click **Specify parameter for output** .
Important information	• The default output type is **Component parameter** if at least one output parameter is defined in the business component Parameter tab. The default output name is the first output parameter from the Parameters tab of the Business Components module. • If no component parameter is defined, you must define one before you can use it to parameterize an output value. For details, see "How to Handle Data in Business Process Testing" on page 235.
Relevant tasks	"How to Work with Keyword GUI Automation" on page 93
See also	*HP Unified Functional Testing User Guide*

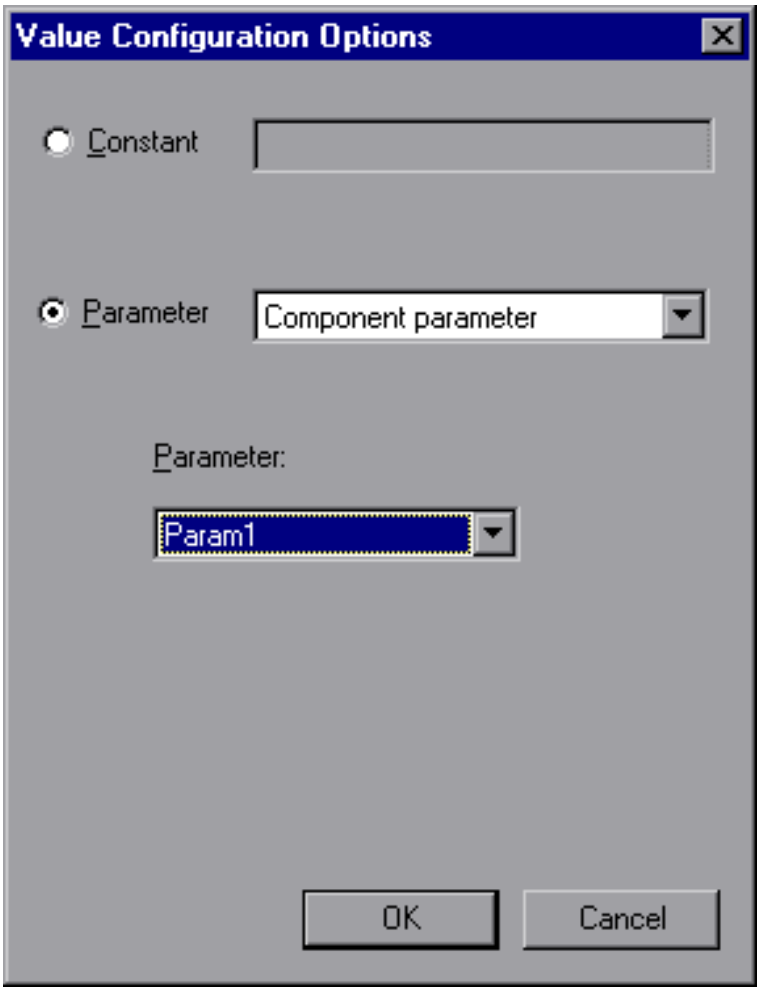
User interface elements are described below:

UI Element	Description
Output Types	Enables you to specify the type of the output parameter: • **Component parameter**. Configures the output parameter as a component parameter. The dialog box changes to the Output Options dialog box for a component parameter. The **Parameter** box displays the available component parameters. The information displayed is read-only. The icon is displayed in the **Output** cell. • **Local parameter**. Configures the output parameter as a local parameter. The dialog box changes to the Output Options dialog box for a local parameter. The **Details** area displays the name and description of the available local parameters. You can select a local parameter and specify the details for it, or create a new local parameter if needed. The icon is displayed in the **Output** cell.
Details	Displays additional fields for specifying local parameters. **Available when**: **Local Parameter** is selected in the **Output Types** field.
Name	Enables you to enter a name for a new local parameter or select an existing local parameter from the list. **Available when: Local Parameter** is selected in the **Output Types** field.
Description	Enables you to enter a brief description for the local parameter. **Available when**: **Local Parameter** is selected in the **Output Types** field.
Parameter	Enables you to choose from a list of component parameters and their descriptions. The details for the component parameter are displayed in read-only format. **Available when**: **Component Parameter** is selected in the **Output Types** field.

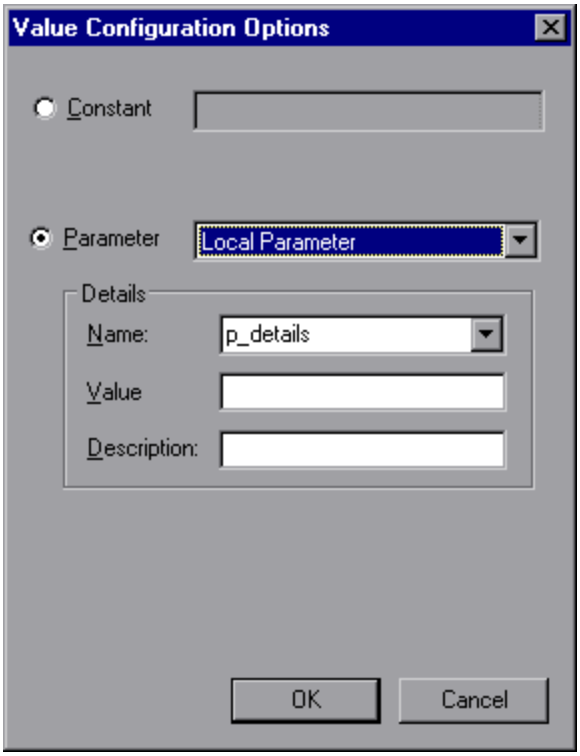
Value Configuration Options Dialog Box

This dialog box enables you to enter input values for a step using constants, local parameters, or component parameters. When the business component runs, it uses this value for the step.

Value Configuration Options Dialog Box for Component Parameters



Value Configuration Options Dialog Box for Local Parameters



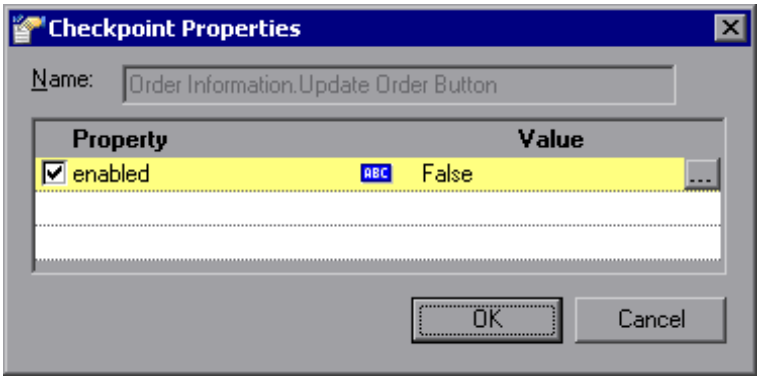
To access	1. In the Business Components module, select a component with Keyword GUI automation. 2. Click the **Automation** tab. 3. Click in a **Value** cell. 4. Click **Configure the value** .
Important information	If at least one input component parameter is defined in the component, the default input type is **Component parameter** and the default input name is the first input parameter from the Parameters tab of the Business Components module. **Note:** If no component parameter is defined, you must define one before you can use it to parameterize an input value. For details, see "Parameters" on page 237. **Tip:** To cancel the parameterization of a value, select the **Constant** radio button in the Value Configuration Options dialog box and enter a constant value.
Relevant tasks	"How to Work with Keyword GUI Automation" on page 93
See also	*HP Unified Functional Testing User Guide*

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

UI Element	Description
Constant	Enables you to enter a constant for the input value or cancel the parameterization of a value.
<constant>	Enables you to enter a constant. **Available when**: The **Constant** radio button is selected.
Parameter	Enables you to choose a parameter type.
<parameter type>	Enables you to choose a local or a component parameter for the input value. **Available when**: The **Constant** radio button is selected.
Parameter	Provides a list of defined component parameters and their descriptions. **Available when**: The **Parameter** radio button is selected.
Details	Displays additional fields for specifying local parameters. **Available when**: **Local Parameter** is selected.
Name	Enables you to enter a name for a new local parameter or select an existing local parameter from the list. **Available when:** The **Local Parameter** parameter type is selected.
Value	Enables you to enter an input value for the local parameter. **Available when:** The **Local Parameter** parameter type is selected.
Description	Enables you to enter a brief description for the local parameter. **Available when**: The **Local Parameter** parameter type is selected.

Checkpoint Properties Dialog Box

This dialog box enables you to view and edit checkpoints added to your keyword GUI automated component, and accept or modify the property values of the checkpoint.



To access	1. In the Business Components module, select a component with Keyword GUI automation. Make sure the Keyword GUI automation contains a checkpoint step (created in UFT). 2. Click the **Automation** tab. 3. Click **Checkpoint Properties** in the **Value** cell of a checkpoint step.
Relevant tasks	"How to Work with Keyword GUI Automation" on page 93
See also	• "Checkpoint Overview" on page 93 • *HP Unified Functional Testing User Guide*

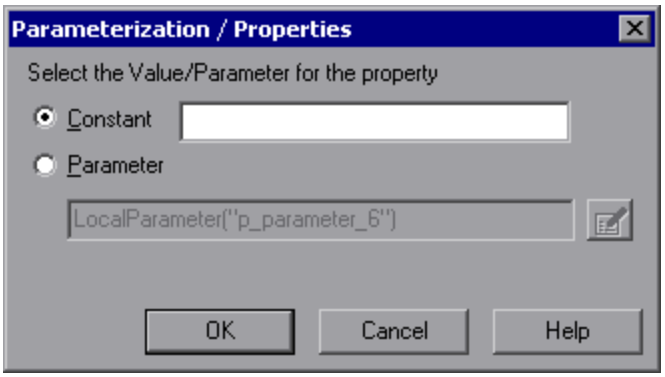
User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Name	The name assigned to the checkpoint.
Property	The property name, and an icon indicating the expected value type of the property. • The icon indicates that the value of the property is currently a constant. • The icon indicates that the value of the property is currently a local parameter. • The icon indicates that the value of the property is currently a component parameter.
<checkbox>	You can accept the selected checks or modify them accordingly. • To check a property, select the corresponding check box. • To exclude a property check, clear the corresponding check box.
...	**Browse**. Opens the Parameterization / Properties dialog box, which enables you to set the expected property value as a Constant or a Parameter. For user interface details, see "Parameterization / Properties Dialog Box" on next page.

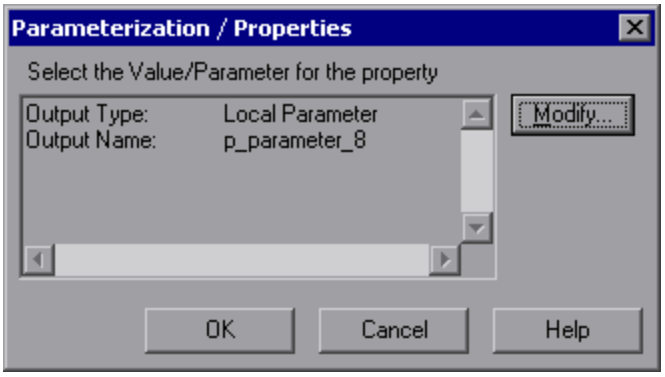
Parameterization / Properties Dialog Box

This dialog box enables you to set the expected property value as a constant or a parameter.

Parameterization Dialog Box



Properties Dialog Box



To access	From the Checkpoint Properties dialog box or the Output Value Properties dialog box, click **Browse** ...
Relevant tasks	"How to Work with Keyword GUI Automation" on page 93
See also	*HP Unified Functional Testing User Guide*

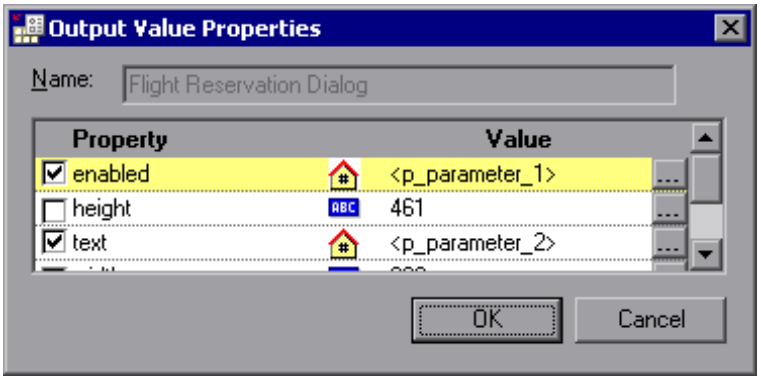
User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Constant	A value that is defined directly in the step and remains unchanged when the component runs. If you select **Constant**, you can edit the value directly in the Constant box. **Available from:** Parameterization dialog box

UI Element	Description
Parameter	A value that is defined or generated separately from the step and is retrieved when the specific step runs. If you select **Parameter** for a value that is already parameterized, the Parameter box displays the current parameter definition for the value. If you select **Parameter** for a value that is not yet parameterized, you can click the **Parameter Options** button to open the Parameter Options dialog box. Specify the property details for the parameter. For details on using parameters in your components, see "How to Handle Data in Business Process Testing" on page 235 **Available from:** Parameterization dialog box
	Parameter Options. Opens the Parameter Options dialog box, which enables you to parameterize the value. **Available from:** Parameterization dialog box
<output definition>	Pane containing the output definition (type and name) for the selected property. You can change the output definition by clicking the **Modify** button. **Available from:** Properties dialog box
Modify	Opens the Output Options dialog box, which displays the current output type and settings for the value, and enables you to modify these settings. For user interface details, see "Output Options Dialog Box" on page 110 **Available from:** Properties dialog box

Output Value Properties Dialog Box

This dialog box enables you to choose which property values to output for output value steps and to define the settings for each value that you select.



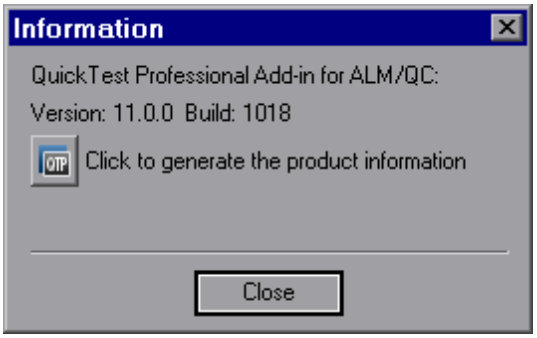
To access	1. In the Business Components module, select a component with Keyword GUI automation. Make sure the Keyword GUI automation contains a checkpoint step (created in UFT). 2. Click the **Automation** tab. 3. Click **Output Value Properties** in the Value column for an output value operation.
Relevant tasks	"How to Work with Keyword GUI Automation" on page 93
See also	• "Keyword View Pane" on page 99 • "Checkpoint Properties Dialog Box" on page 115 • *HP Unified Functional Testing User Guide*

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

UI Element	Description
Property	The name assigned to the output value.
<value type>	The expected value type of the property. • The icon indicates that the value of the property is currently a constant. • The icon indicates that the value of the property is currently a local parameter. • The icon indicates that the value of the property is currently a component parameter.
Value	Property value that is currently stored with the object in the object repository,
<checkbox>	To specify a property to output, select the corresponding check box. You can select more than one property for the object and specify the output options for each property value you select.
...	**Browse**. Opens the Parameterization / Properties dialog box, which enables you to modify the property value. For user interface details, see "Parameterization / Properties Dialog Box" on page 117.

Information Dialog Box

This dialog box displays the version and build information for the UFT Professional Add-in for ALM installed on your client machine.



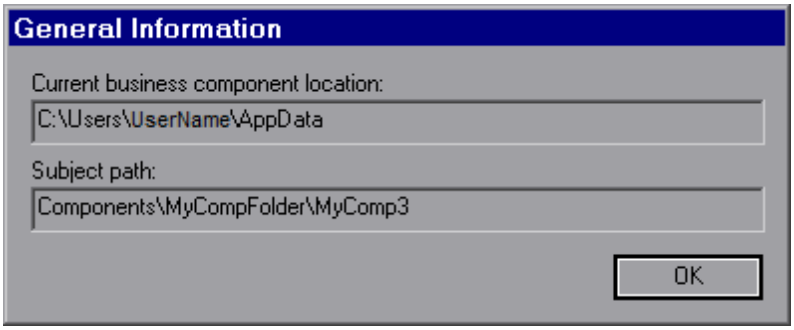
To access	In the Automation tab, click .
Relevant tasks	"How to Work with Keyword GUI Automation" on page 93
See also	*HP Unified Functional Testing User Guide*

User interface elements are described below:

UI Element	Description
Version	Version number for the UFT Professional Add-in for ALM.
Build	Build number for the UFT Professional Add-in for ALM.
	Displays additional product information in a browser window. **Example** Product Information Product name: QuickTest Professional Add-in for ALM/QC Product version: 11.0.0 Product ID: TDPLUGPID11.0/00 Product build: 1018 Operating system: Service Pack 1 (Build 7601) Patch Information: Name / Readme © Copyright 1992-2010 Hewlett-Packard Development Company, L.P.

General Information Dialog Box

This dialog box displays the location of the business component.



To access	In the Automation tab, click .
Relevant tasks	"How to Work with Keyword GUI Automation" on page 93
See also	*HP Unified Functional Testing User Guide*

User interface elements are described below:

UI Element	Description
Current business component location	The path to the business component as stored locally on your client machine.
Subject path	The path to the business component in the components tree in the Business Components module.

Chapter 6

Components with Scripted GUI or API Automation

This chapter includes:

Scripted GUI and API Automation Overview 124

How to Work with Scripted GUI and API Automation 124

Scripted GUI and API Automation User Interface 125

Scripted GUI and API Automation Overview

Components automated with either the scripted GUI or API automation types are automated, reusable modules that perform a specific task when testing your application. The tasks are defined using scripted programming logic.

You can edit scripted automation programming logic only in the application in which the automation was created, such as Unified Functional Testing (UFT) or HP Service Test. You cannot modify scripted automation in ALM, but you can include scripted components in any business process test or flow.

From within ALM, you can launch a scripted component in the testing tool in which the component was automated.

Note:

- For details on scripted GUI automation in UFT, see the *HP Unified Functional Testing User Guide*.
- For details on API automation in UFT, see the *HP Service Test User Guide* or the information about API testing in the *HP Unified Functional Testing User Guide*.
- Most of the information, examples, and images in this guide focus specifically on working with keyword GUI automation. However, much of the information also applies to API automation or scripted GUI automation.

How to Work with Scripted GUI and API Automation

This task describes how to automate a component by associating the component with scripted GUI and API automation.

Note: This task is part of a higher-level task. For details, see "Business Process Testing Workflow" on page 28.

This task includes the following steps:

- "Prerequisites" below
- "Create an application area" on next page
- "Select a component" on next page
- "Implement scripted automation" on next page
- "Results" on next page

1. **Prerequisites**
   - Make sure a testing tool, such as UFT, is installed.

Tip: If a testing tool is not installed, the

button in the Automation tab is disabled.

2. **Create an application area**

   Automation involves the creation of application areas, which provide access to the settings and resources required for a particular application. Work with an automation engineer to create the application area in UFT. For details, see the Create an application area step under the "How to Work with Keyword GUI Automation" on page 93 task.

3. **Select a component**

   In the Business Components module window, select a component in the component tree.

4. **Implement scripted automation**

   a. In the Automation tab, click

and select either **Scripted GUI** or **API**. For user interface details, see "Script Launch Pane" below.

   b. Click **Launch** to launch UFT or HP Service Test. Work with an automation engineer to create the script (content) in the testing tool.

   For details on scripted GUI automation, see the information about GUI testing in the *HP Unified Functional Testing User Guide*.

   For details on API automation, see the *HP Service Test User Guide* or the information about API testing in the *HP Unified Functional Testing User Guide*.

5. **Results**

   - The component is fully automated.
   - The icon for the component changes from the manual icon to the automated icon. For icon details, see "Business Process Testing Icons" on page 62.
   - The automated component can be accessed from the testing tool.
   - Tests and flows that contain fully automated components can run in an automatic runner without pausing for user input.

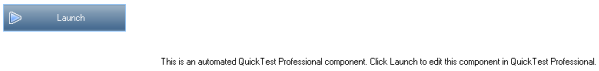
Scripted GUI and API Automation User Interface

This section includes:

Script Launch Pane 125

Script Launch Pane

This pane enables you to launch the UFT testing tool from within ALM.You can add or modify component content in the form of automated scripts in the testing tool.



To access	1. In the Business Components module, select a component with Scripted GUI or API automation. 2. Click the **Automated** tab. The Scripted Launch pane is displayed in the lower half of the window.
Relevant tasks	"How to Work with Scripted GUI and API Automation" on page 124
See also	• "Scripted GUI and API Automation Overview" on page 124 • *HP Unified Functional Testing User Guide*

User interface elements are described below:

UI Element	Description
	Opens the applicable testing tool so you can view and edit the automation script.

Business Process Tests and Flows

Chapter 7

Introducing Business Process Tests and Flows

This chapter includes:

Business Process Test and Flow Overview 130

How to Plan Business Process Tests and Flows 130

Business Process Test and Flow Overview

This topic provides an overview of working with business process tests and flows in the Test Plan module in ALM.

- A business process test is a scenario comprising a sequence of business components or flows, designed to test a specific scenario in an application.
- A flow is a type of test that comprises a logical set of business components, in a fixed sequence, that performs a specific task. Flows share the same functionality as business process tests (for example, iterations, parameters, and results) in addition to functionality unique to flows (validation, debug mode). When designing flows, we recommend that you consider them "compound components."

  Flows cannot contain other flows.

  You can use a flow in multiple business process tests. When you modify a flow or any of its components, all business process tests containing that flow reflect that modification.

  Business Process Testing Enterprise Extension: You can create a flow by navigating your enterprise application. This process is called "learning a flow." When your flow is ready, you can add it to business process tests, together with other flows and business components. For details, see "Learn Flow Process" on page 169.

Working with business process tests and flows is very similar to working with other ALM test types. This overview highlights the differences. For conceptual details on the Test Plan module for other ALM test types, see the *HP Application Lifecycle Management User Guide*.

How to Plan Business Process Tests and Flows

This task describes how to work with business process tests and flows using Business Process Testing.

Note:

- This task is similar to the same task for other ALM test types, but includes modified steps relevant for Business Process Testing. For task details on working in the Test Plan module for other ALM test types, see the *HP Application Lifecycle Management User Guide*.
- This task is part of a higher-level task. For details, see "Business Process Testing Workflow" on page 28.

This task includes the following steps:

- "Create a test plan tree" on next page
- "Define parameters" on next page
- "Define iterations" on next page
- "Define test configurations" on next page
- "Create requirement coverage" on next page

- "Link a test to a defect" on next page
- "Debug a test or flow" on next page
- "Debug a test or flow" on next page
- "Establish a baseline" on next page

1. ## Create a test plan tree

   Create a test plan tree of test subject folders, tests, and flows.

   You define the details and description of a test, and then add content to the test structure by adding a sequence of business components and/or flows for a specific business process. Similarly, you can define the details and description of a flow, and then add content to the flow structure by adding a sequence of business components.

   For task details, see "How to Create Business Process Tests and Flows" on page 134.

2. ## Define parameters

   To make tests and flows more flexible, you can include parameters in business component steps, and make them available to your tests and flows. The business components and flows can pass variable values between each other.

   You can define input and output parameters for flows.

   You can define only input parameters for business process tests.

   For task details, see "How to Create Parameters" on page 238.

3. ## Define iterations

   You can run the same test or flow iteratively (repeatedly). Each run is called an iteration, and can access different parameter values.

   For task details, see "How to Set Data for Iterations" on page 287.

4. ## Define test configurations

   To run tests for different use-cases, you can define test configurations. This enables you to run the same test under different scenarios, using different sets of data.

   In addition to accessing static data (data supplied directly within ALM), test configurations for Business Process Testing can also access dynamic data (data supplied from an external data table).

   For task details, see "How to Work with Business Process Test Configurations" on page 203.

5. ## Create requirement coverage

   Link business process testing criteria (tests, flows, and components) with requirements in the requirements tree. By defining requirements coverage for business process testing criteria, you keep track of the relationship between the tests and flows in your test plan and your original requirements.

   For task details, see "How to Create Coverage by Criteria" on page 189.

6. **Link a test to a defect**

   Link a business process test or flow to specific defects. This is useful, for example, when a new test is created specifically for a known defect. By creating a link you can determine if the test should run based on the status of the defect. For user interface details, see the *HP Application Lifecycle Management User Guide*.

7. **Analyze test plan data**

   Analyze the test plan by generating reports and graphs.

   For details on analysis tools in ALM, see the *HP Application Lifecycle Management User Guide*.

8. **Debug a test or flow**

   Debug a test or flow by running in debug mode.

   For user interface details, see "Test and Flow Debugging User Interface " on page 224.

9. **Establish a baseline**

   After your test plan has been reviewed and approved, you can create a baseline. A baseline provides you with a snapshot of your test plan at a specific point in time. Use a baseline to mark any significant milestone in the application lifecycle. The baseline then serves as a point of reference against which changes can be compared. For task details, see the *HP Application Lifecycle Management User Guide*.

Chapter 8

Business Process Test and Flow Specification

This chapter includes:

Business Process Test and Flow Specification Overview 134

How to Create Business Process Tests and Flows 134

How ALM Calculates Business Process Test and Flow Statuses 144

Business Process Test and Flow Specification User Interface 145

Business Process Test and Flow Specification Overview

The chapter describes the unique elements in the Test Plan module interface when a business process test or flow is selected.

You use the Test Plan module to build business process tests and flows by combining business components into a testing structure.

- Business process tests are testing scenarios comprised of business components and flows.
- Flows are a type of test that comprise a sequence of business components.

You create business process tests and flows in the Test Plan module test tree. At this stage, you define basic information about the business process test or flow. Afterwards, you select components and flows to add to the business process tests (and, similarly, the components to add to the flows).

Note:

- The Test Plan module can be used to design many types of tests. The information provided in this chapter is relevant only to Business Process Testing. For details on using the Test Plan module with other test types, see the *HP Application Lifecycle Management User Guide*.
- This chapter describes how to use several Business Process Testing features. HP Business Process Testing Enterprise Extension provides additional, automated features that enable you to test enterprise applications. Therefore, some procedures described in this chapter may not be part of the recommended best practice when using Business Process Testing to test your enterprise application.
- All standard functionality in the Test Plan module applies to Business Process Testing as well. For details on using the Test Plan module, see the *HP Application Lifecycle Management User Guide*.

For task details on creating business process tests and flows, see "How to Create Business Process Tests and Flows" below.

How to Create Business Process Tests and Flows

This task describes how to create, modify, and work with business process tests and flows.

Note:

- **Higher-level task:** This task is part of a higher-level task. For details, see "How to Plan Business Process Tests and Flows" on page 130.
- For a use-case scenario related to this task, see "How to Create Business Process Tests - Use-case Scenario" on page 139.

This task includes the following steps:

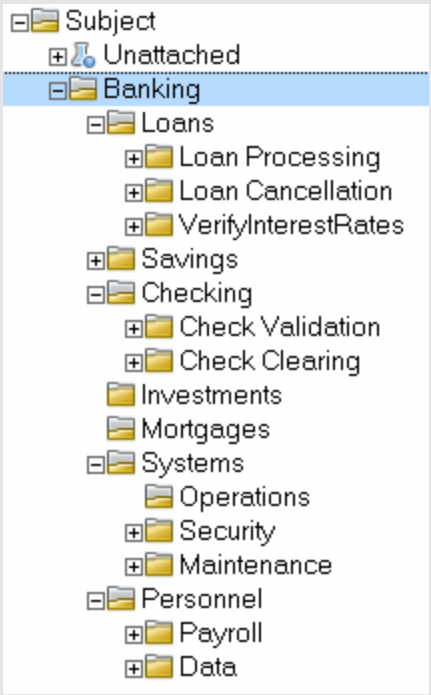
- "Create test subjects" below
- "Create business process tests and flows in the test plan tree" on next page
- "Update business process tests and flows" on next page
- "Add content to business process tests and flows" on next page
- "Learn flows - optional" on page 137
- "Request components - optional" on page 137
- "Set run conditions for components in flows" on page 138
- "Set failure conditions for components and flows" on page 138
- "Group components and flows" on page 138
- "Validate tests" on page 138
- "Debug tests and flows from the Test Plan module" on page 139

1. ## Create test subjects

   Create folders in the test plan tree as you would for any ALM test type.

   Example:

   To test a banking application that lets you manage loan requests, mortgages, checking accounts, savings accounts, bank personnel, and system operations, you might define the following subjects for the test plan tree:



Notice how the **Loans** subject contains additional subject folders. These folders further categorize the tests in the test plan tree by creating additional levels of subjects.

2. ## Create business process tests and flows in the test plan tree

   Create business process tests and flows under test subject folders.

   a. In the Test Plan tree, right-click a folder, and select **New Test**.

   b. Fill in the **Type** field in the New Test dialog box, making sure to select one of the following test types:

      - **BUSINESS-PROCESS** to create a business process test
      - **FLOW** to create a flow

   c. Fill in the remaining fields in the New Test dialog box.

   d. Save the test or flow. For user interface details, see the information about the New Test Dialog Box in the *HP Application Lifecycle Management User Guide*.

3. ## Update business process tests and flows

   For each business process test or flow, you can update its details, attachments, parameters, and so on. Right-click a business process test or flow and select **Test Details**. The Test Details dialog box opens. For user interface details, see "Test Details Dialog Box for Business Process Testing" on page 165

4. ## Add content to business process tests and flows

   Build the content of a business process test by:

   - Dragging business components and/or flows from the Select Components and Flows pane, and dropping them into the test in the Test Script tab.
   - Dragging business components and from the Select Components and Flows pane, and dropping them into the flow in the Test Script tab.

   Building the content can be done from both the Test Script Tab Grid and the Test Script Tab Canvas.

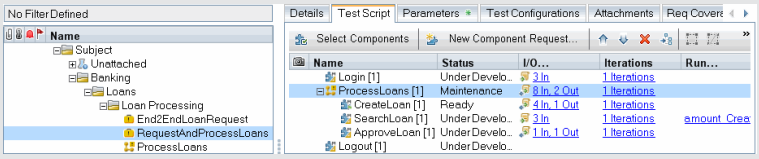
   For user interface details, see "Test Script Tab for Business Process Testing" on page 152.

   Tip: From both the Test Script Tab Grid and the Test Script Tab Canvas, you can arrange the business components and flows into a logical testing order by:

   - Selecting business components and flows using the up and down arrow buttons in the toolbar.
   - Dragging and dropping the business components and flows.

Example:

The test for the loan request business process, **RequestAndProcessLoans**, may include components for logging in and logging out, and a flow, **ProcessLoans**, that contains the components for processing the loan request (**CreateLoan**, **SearchLoan**, and **ApproveLoan**):



5. Learn flows - optional

Business Process Testing Enterprise Extension users can create flows automatically by navigating through environment-specific applications.

For task details, see "How to Learn Flows" on page 170.

6. Request components - optional

While designing a test or flow, you can request components be created and you can specify the details for creating the components. The component request can be included in your test or flow the same way as a regular component.

a. In the Test Plan tree, select the business process test or flow for which you are requesting the component and click the **Test Script** tab.

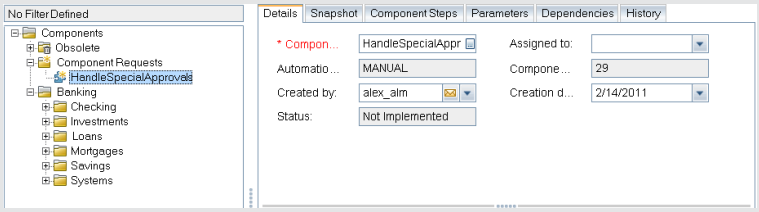
b. Click to open the Component Details dialog box.

c. Rename the component request (from **New Component Request**), and fill in the relevant fields, such as **Assigned to**.

d. Click **OK**. The new component request is displayed in the Test Script tab for the selected business process test or flow, and in the Component Tree pane under the **Component Requests** folder with a status of **Not Implemented**. For task details on creating a component from a component request, see the step "How to Create Business Components in ALM" on page 46.

Example:

From the Test Plan module, you request the creation of a component named **HandleSpecialApprovals**. The following shows the new component in the **Component Requests** folder:



Tip: Unlike other components, requested components can be modified directly from the Test Plan module, unless the component request is contained within a flow in a business process test. (In this case, you can only view the component request from the business process test.)

7. ## Set run conditions for components in flows

   You can define whether a business component that does not meet a condition stops flow execution or advances to the next component in the flow.

   In the Test Script tab, click for a business component in a flow. The Run Condition dialog box opens. For user interface details, see "Run Conditions Dialog Box" on page 196.

8. ## Set failure conditions for components and flows

   You can define whether a business process test run continues or ends if a specific business component or flow in the test fails.

   In the Test Script tab, click in the **On Failure** column for the business component. Select either **Exit** or **Continue**.

   For user interface details, see "Test Script Tab for Business Process Testing" on page 152.

9. ## Group components and flows

   In some cases, it may be helpful to iterate several business components or flows together as a group.

   In the Test Plan tree, click the **Test Script** tab. Select the business process tests or flows you want to group, right-click, and select **Grouping > Group**.

   When iterating groups:

   - All items to be included in the group must have the same number of iterations and the same iteration range, or a warning message is displayed.
   - The number of iterations indicates the total number of value sets defined for the group.
   - The iteration range indicates the subset of iterations currently set for use when the test runs.

   For user interface details about creating groups, see "Test Script Tab for Business Process Testing" on page 152.

   For conceptual details about group iterations, see "Iterations Overview" on page 284.

   For user interface details about group iterations, see "Iteration Pages" on page 292.

   For task details on iterating groups, see "How to Set Data for Iterations" on page 287.

10. ## Validate tests

    You can check your business process tests and flows to locate any component errors that could halt the successful running of the test or flow. Validating a business process test or flow in the Test Plan module checks all instances of the test or flow within all test sets.

In the Test Plan tree, click the **Test Script** tab. Select the business process test or flow you want to validate. Click .

If errors are found, modify the test as required. Repeat the validation check. If the errors have been resolved, a message is displayed indicating that the validation completed successfully.

For user interface details, see "Test Validation Results Dialog Box" on page 166.

11. **Debug tests and flows from the Test Plan module**

You can check whether automated components run together successfully in the test or flow by debugging the test or flow in the Test Plan module. You can check, for example, whether the component and flow order in the test is logical, or that components start where previous components in the test end.

For task details, see "How to Debug Tests and Flows Containing Automated Components" on page 222.

How to Create Business Process Tests - Use-case Scenario

This use-case scenario describes how to create a business process test that includes a flow.

Note: For a task related to this scenario, see "How to Create Business Process Tests and Flows" on page 134.

This scenario includes the following steps:

- "Background" below
- "Create a flow in the test plan tree" on next page
- "Add content to the flow" on next page
- "Create a business process test in the test plan tree" on page 141
- "Add content to the test" on page 141
- "Request components " on page 141
- "Set run conditions for components" on page 142
- "Set failure conditions for components and flows" on page 143
- "Group components and flows " on page 143
- "Validate tests " on page 144

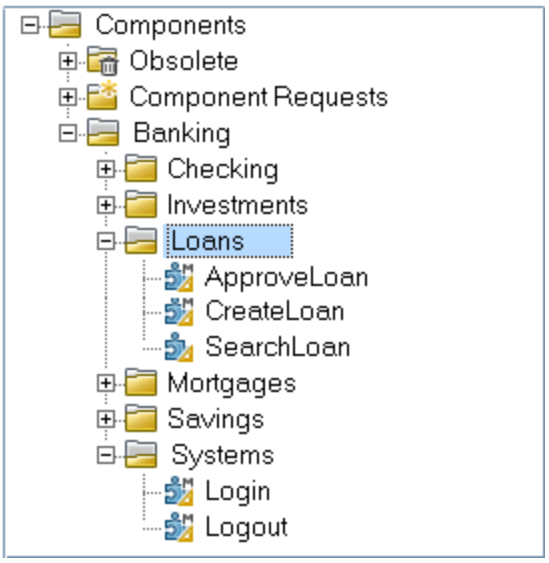
1. **Background**

The banking application for loan processing incorporates several business processes. Each of these processes is defined using business components. To test a business process for processing a loan, we could create a business process test that includes the following flows and components:

- The **CreateLoan**, **SearchLoan**, and **ApproveLoan** components represent the processes a loan officer would have to perform repeatedly throughout the day. These components are good candidates for inclusion in a flow because they are a set of operations that are often performed together under various situations and may need to run iteratively. In this use-case, we call the flow **ProcessLoans**.

  If, at any time in the future, the transactions for processing loans change in our application, we would update the **ProcessLoans** flow. All business process tests containing this flow would reflect the changes in our application.

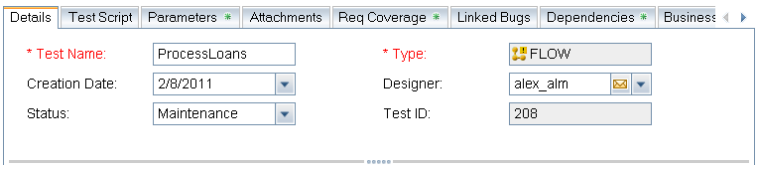
- The **Login** and **Logout** components are done once at the beginning of the day, and once at the end of the day, respectively. They are standalone, yet reusable, entities. Most business processes in the banking application will start and end with logging in and logging out.



Notice how the components related to the day-to-day loan application processes are located in the **Loans** subject, while the components related to once-a-day, technical operations are located in the **Systems** subject.

2. ## Create a flow in the test plan tree

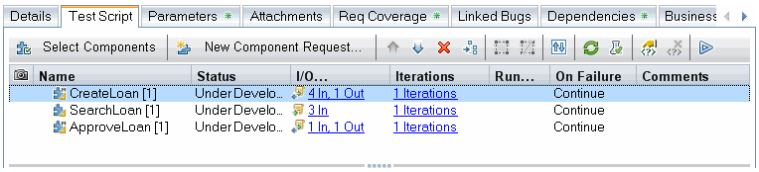
In the Test Plan module, the Details tab for the **ProcessLoans** flow could look as follows:



Notice that the value for the **Type** field is **FLOW**.

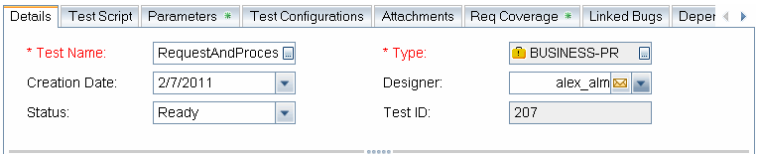
3. ## Add content to the flow

In the Test Script tab, we add the content to the flow by selecting components. Our flow could contain components as follows:



4. ## Create a business process test in the test plan tree

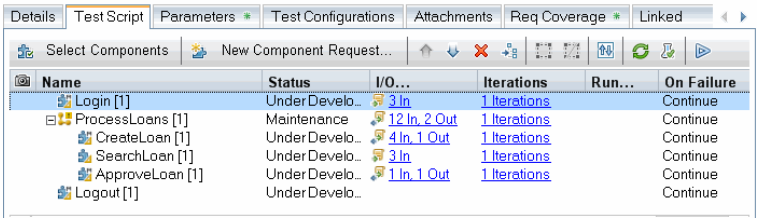
In the Test Plan module, the Details tab for the business process test, **RequestAndProcessLoans**, could look as follows:



Notice that the value for the **Type** field is **BUSINESS-PROCESS**.

5. ## Add content to the test

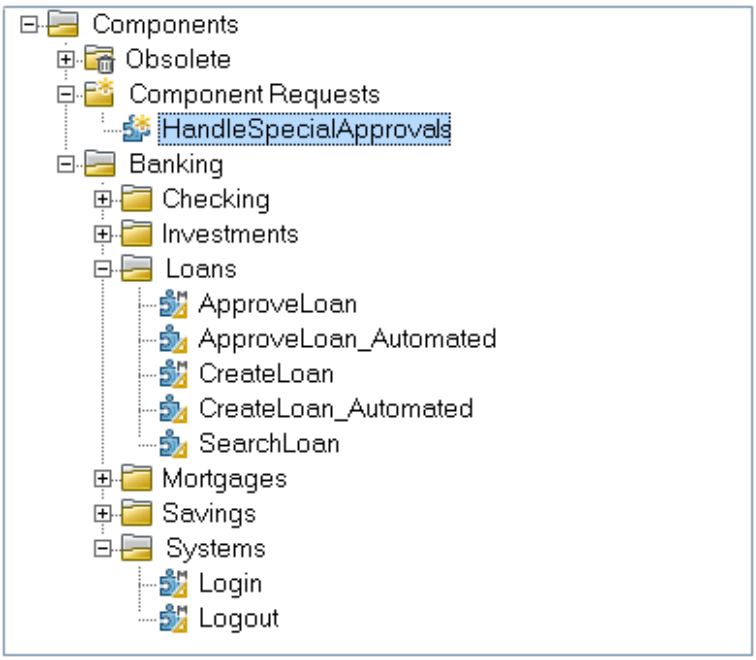
In the Test Script tab, we add the content to the test by selecting components and flows. Our test could contain the **Login** and **Logout** components and the **ProcessLoans** flow (expanded) as follows:



6. ## Request components

Suppose, while adding content to the test, we realize that a business process for authorizing loans requesting exceptionally high amounts is not represented in the loan processing flow. Most loan officers can process and approve loans up to a certain amount, but above a certain amount, special managerial approval is needed.

We can request the creation of a business component called **HandleSpecialApprovals**.

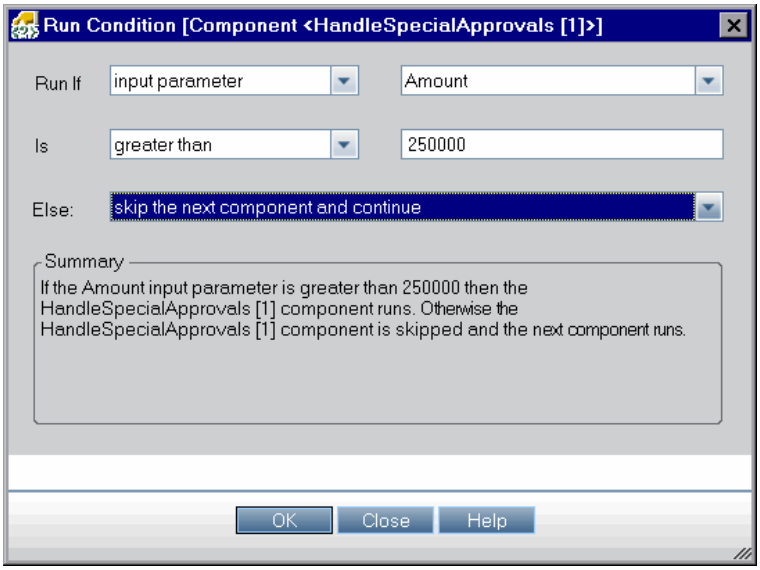


Even though the component is located in the **Component Requests** folder, it can be added to flows and tests like any other component.

7. Set run conditions for components

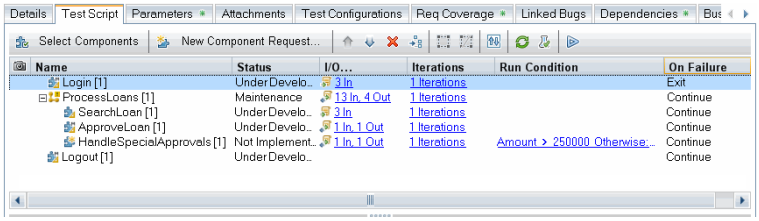
The **HandleSpecialApprovals** business component is only needed if the requested loan amount is higher than a certain amount. We set the run condition for the **HandleSpecialApprovals** component so that the component runs only when the amount is greater than $250,000.

Note: For one component to be able to access a value from another component earlier in the test script, the parameters must be linked. For task details, see "How to Link Data" on page 270.



8. Set failure conditions for components and flows

It would be reasonable to assume that if the **Login** component fails, the remaining flows and components in the business process test should not run. Therefore, we set the **On Failure** condition for the **Login** component to **Exit**.

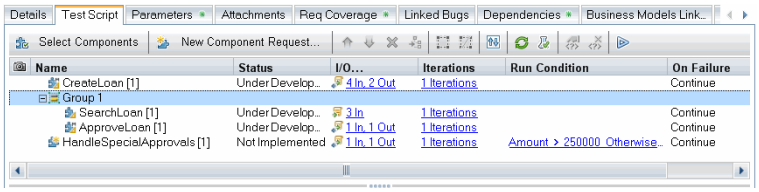


9. Group components and flows

Assume the roles and work flows in our bank are such that:

- A loan processor processes the customer's loan request using the **CreateLoan** process in real-time when a customer arrives at the bank.
- A mid-level manager searches (**SearchLoan**) for loan requests and approves (**ApproveLoan**) the loans daily at a set time.

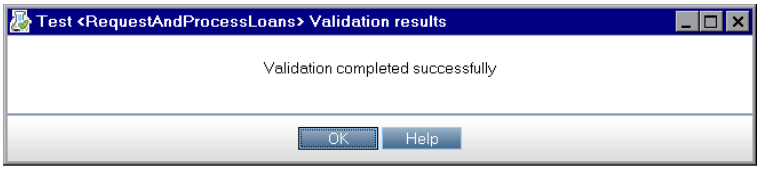
It could be beneficial to group the mid-level manager's process together, as it is likely that the manager will systematically repeat these processes iteratively. Returning to the Test Script tab for the flow, we select the components we intend to group.



Notice the expandable/collapsible **Group 1** node. We can give the group a more meaningful name, such as **ApprovalGroup**.

10. **Validate tests**

    We validate our tests to see if we have component errors that could halt the successful running of the test. We hope to see the following validation results.



How ALM Calculates Business Process Test and Flow Statuses

This section provides examples to help you understand how ALM calculates the statuses for business process tests and flows.

The status of the test or flow is determined by the status of its business components. The component with the most severe status determines the test or flow status.

The Error Status

Suppose you have a business process test that contains:

- 2 **Ready** components
- 1 **Maintenance** component
- 1 **Under Development** component
- 1 **Error** component
- 1 **Not Implemented** (requested) component

The test status is **Error**, because **Error** is the most severe status of a business component in the test.

The Outdated Status

Suppose you have a business process test that contains:

- 2 **Ready** components, one of which is outdated and moved to the **Obsolete** folder in the component tree.
- 1 **Maintenance** component
- 1 **Under Development** component
- 1 **Error** component
- 1 **Not Implemented** (requested) component

The test status is **Outdated**, because an obsolete component is more severe than the **Error** status.

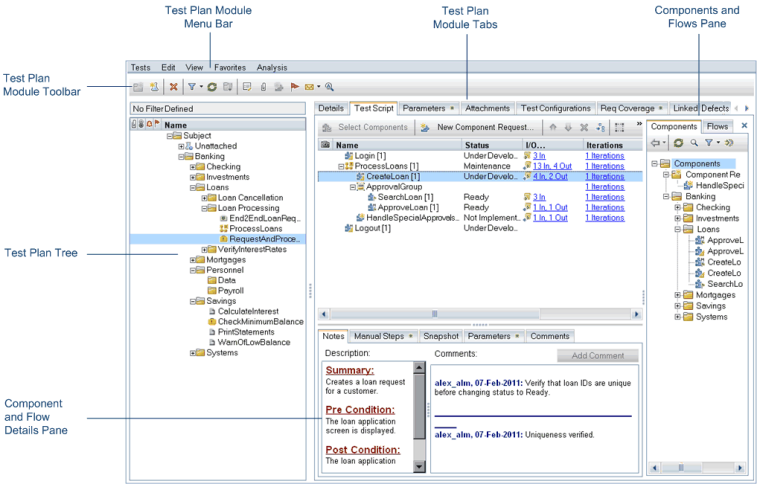
Business Process Test and Flow Specification User Interface

This section includes:

Test Plan Module Window for Business Process Testing 145
Test Plan Module Menu and Buttons for Business Process Testing 147
Test Plan Module Icons for Business Process Testing 148
Test Plan Module Fields for Business Process Testing 150
Test Script Tab for Business Process Testing 152
Test Details Dialog Box for Business Process Testing 165
Test Validation Results Dialog Box 166

Test Plan Module Window for Business Process Testing

This version of the Test Plan module window enables you to define and maintain business process tests and flows.



To access	On the ALM sidebar, under **Testing**, select **Test Plan**. Select a business process test or flow in the test plan tree.
Important information	The Test Plan module for Business Process Testing differs from the Test Plan module for other ALM test types. This section describes the elements that are significant specifically when designing business process tests and flows. For user interface details on the Test Plan module for other test types, see the *HP Application Lifecycle Management User Guide*.
Relevant tasks	• "How to Plan Business Process Tests and Flows" on page 130 • "How to Create Business Process Tests and Flows" on page 134

See also	• "Business Process Test and Flow Specification Overview" on page 134 • "Test Plan Module Fields for Business Process Testing" on page 150 • *HP Application Lifecycle Management User Guide*

User interface elements unique to Business Process Testing are described below (unlabeled elements are shown in angle brackets):

UI Element (A - Z)	Description
<Test Plan for Business Process Testing module common UI elements>	• **Test Plan for Business Process Testing fields.** For field definitions, see "Test Plan Module Fields for Business Process Testing" on page 150. • **Test Plan for Business Process Testing menus and buttons.** For command and button descriptions, see "Test Plan Module Menu and Buttons for Business Process Testing" on next page. • **Test Plan for Business Process Testing icons.** For icon descriptions, see "Test Plan Module Icons for Business Process Testing" on page 148.
<Test plan tree>	Organizes and displays your tests hierarchically. **Note:** In previous versions of ALM and Quality Center, the **BPT Resources** folder was created in the Test Plan module. For business process tests or flows to perform properly in ALM, this folder and its subfolders should not be renamed or deleted. In ALM, the **BPT Resources** folder no longer exists in the Test Plan module. This folder, which contains the business component resources in the project, is created automatically in the Test Resources module the first time you click the **Automation tab** in a new project, when you create a UFT automated component for the first time, or when UFT connects to ALM for the first time.
Criteria	Enables you to view the entities, such as flows and business components, that comprise a test. These entities are defined as criteria, and are used for the purposes of analyzing requirement coverage at a more granular level. For task details, see "How to Create Coverage by Criteria" on page 189.
Details	Lists the business process test or flow fields. For details on fields specifically for Business Process Testing, see "Test Plan Module Fields for Business Process Testing" on page 150. For details on other test fields, see the *HP Application Lifecycle Management User Guide*.
Parameters	Lists parameters associated with the currently selected test or flow. Enables you to define parameters at the test and flow level. Parameters can be incorporated in the test's or flow's components' steps. For task details, see "How to Handle Data in Business Process Testing" on page 235.
Test Configurations	Enables you to define and update test configurations. For task details, see "How to Work with Business Process Test Configurations" on page 203.

UI Element (A - Z)	Description
Test Script	Enables you to build the structure of the business process test or flow by adding business components (and, in the case of tests, flows). For more details, see "Test Script Tab for Business Process Testing" on page 152.

Test Plan Module Menu and Buttons for Business Process Testing

This section describes the additional menus and buttons available in the Test Plan module for Business Process Testing.

To access	On the ALM sidebar, under **Testing**, select **Test Plan**. Select a business process test or flow in the test plan tree.
Important information	**Tip:** You can access the menu bar from the Test Plan module for Business Process Testing by pressing the shortcut key ALT. Additionally, you can perform many other ALM operations using shortcut keys. For a list of Business Process Testing shortcut keys, see "Business Process Testing Keyboard Shortcuts" on page 33.
Relevant tasks	"How to Create Business Process Tests and Flows" on page 134
See also	• *HP Application Lifecycle Management User Guide* • "Test Plan Module Window for Business Process Testing" on page 145

User interface elements unique to Business Process Testing are described below:

UI Elements (A-Z)	Available from	Description
Convert to Component	Tests	Opens the Select Destination Folder dialog box, which enables you to convert existing ALM manual tests to manual components. For user interface details, see "Select Destination Folder Dialog Box" on page 75. **Available for**: ALM manual tests

UI Elements (A-Z)	Available from	Description
Delete	Edit	Deletes the selected business process tests, flows, or folders. **Caution:** Additional cautions for Business Process Testing include: • You cannot delete any component or flow that is used by a business process test. • Any removed flow, test, or test folder is permanently deleted. The test or flow is removed from all test sets and the run history is erased. • It is highly recommended that you do not run a business process test or flow from the **Unattached** folder. Move the test or flow to a valid folder in the Test Plan tree before running it.
Test Details	<right-click menu>	Opens the Test Details dialog box, displaying the details of the selected test or flow. For user interface details specific for Business Process Testing, see "Test Details Dialog Box for Business Process Testing" on page 165.

Test Plan Module Icons for Business Process Testing

This section describes the icons available in the Test Plan module for Business Process Testing.

To access	On the ALM sidebar, under **Testing**, select **Test Plan**. Select a business process test or flow in the test plan tree.
Important information	• Tests created in other testing tools can also be included in the Test Plan module tree, and are identified by other icons. • Status icons for business process tests and flows are displayed in the Test Plan module tree and in the **Status** box in the test or flow's Details tab.
See also	• "Test Plan Module Menu and Buttons for Business Process Testing" on previous page • "Test Plan Module Window for Business Process Testing" on page 145

The icons in the Test Plan module that are unique to Business Process Testing are described below:

UI Element	Description
	<green asterisk>. The applicable tab is populated.

UI Element	Description
	Ready status for a flow. All the business components included in the flow have **Ready** status. This is the least severe status for a flow. For details about this status, see "Status" on page 151.
	Ready status for a test. All the business components included in the business process test have **Ready** status. This is the least severe status for a test. For details about this status, see "Status" on page 151.
	Maintenance status for a flow. For details about this status, see "Status" on page 151.
	Maintenance status for a test. For details about this status, see "Status" on page 151.
	Error status for a flow. This is the second-most severe status for a flow. For details about this status, see "Status" on page 151.
	Error status for a test. This is the second-most severe status for a test. For details about this status, see "Status" on page 151.
	Outdated status for a flow. The flow contains obsolete components. A more up-to-date component may be required in the flow. This is the most severe status for a flow. For details about this status, see "Status" on page 151.
	Outdated status for a test. The test contains obsolete components. A more up-to-date component may be required in the test. This is the most severe status for a test. For details about this status, see "Status" on page 151.
	Group. Identification of a group node. The components and flows displayed hierarchically under the group node are part of a group. For task details on creating groups, see "Group components and flows" on page 138.
	Learned. Components created by the Business Process Testing Enterprise Extension learn flow are identified by this icon. For task details about learning flow, see "How to Learn Flows" on page 170.
	Similar. The learned component has identical or similar components in the project. The component can be substituted with a reused component. **Available from**: The Learned Flow Summary dialog box > Advanced Details view. For user interface details, see "Learn Flow Summary Page" on page 180.
	No Similar. The learned component has no identical or similar components in the project. The component cannot be substituted with a reused component. **Available from**: The Learned Flow Summary dialog box > Advanced Details view. For user interface details, see "Learn Flow Summary Page" on page 180.
	Reused. Component reuse has been applied to this learned component. **Available from**: The Learned Flow Summary dialog box > Advanced Details view. For user interface details, see "Learn Flow Summary Page" on page 180.

UI Element	Description
	Unresolved Changes in Flow. The flow was run in change detection mode and its changes are unresolved. A flow indicates unresolved changes if it contains components with unresolved changes.
	Unresolved Changes in Test. The test was run in change detection mode and its changes are unresolved. A test indicates unresolved changes if it contains any flows or components with unresolved changes.
	Version Control. The test or flow is checked out. • **Green lock.** The test or flow is checked out by the current user. • **Red lock.** The test or flow is checked out by another user. For details on working with version control in ALM, see the *HP Application Lifecycle Management User Guide*.

Test Plan Module Fields for Business Process Testing

This section describes the Test Plan module fields for Business Process Testing.

To access	On the ALM sidebar, under **Testing**, select **Test Plan**. Select a business process test or flow in the test plan tree.
See also	• *HP Application Lifecycle Management User Guide* • "Test Plan Module Window for Business Process Testing" on page 145

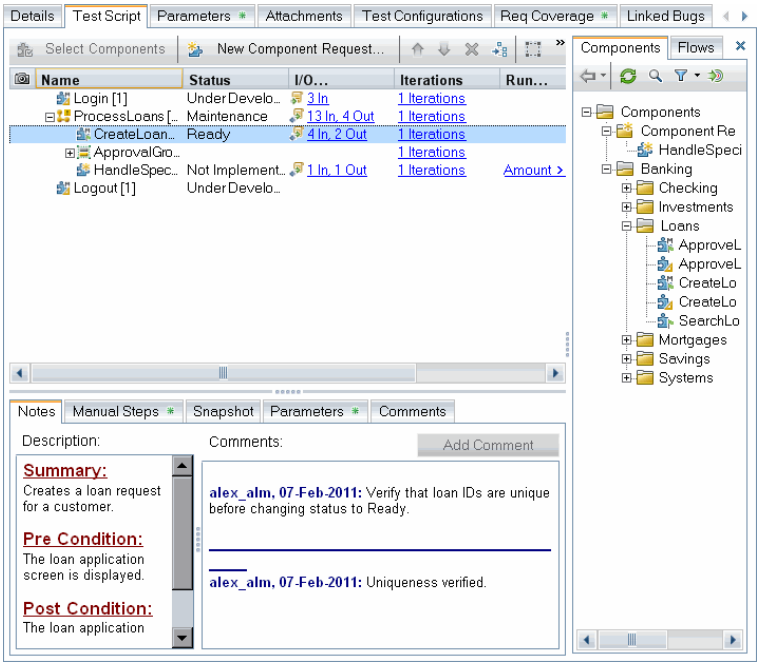
Business Process Testing fields that are unique to, or operate differently from, other test types are described below:

UI Element (A-Z)	Description
Status	The status of the business process test or flow. For Business Process Testing, this field is read-only. The status of the test or flow is determined by the status of its business components. The component with the most severe status determines the test or flow status. For examples, see "How ALM Calculates Business Process Test and Flow Statuses" on page 144. • **Design.** The business process test or flow is first created. • **Ready.** All the business components included in the business process test or flow have **Ready** status. • **Maintenance.** One or more business components included in the test or flow are being modified or are not yet complete, and have **Under Development** or **Maintenance** status (and no components in the test or flow have a more severe status). • **Error.** One or more business components included in the business process test or flow have **Error** status. • **Outdated.** One or more business components included in the business process test or flow are **Obsolete**. For icons representing business component statuses, see "Test Plan Module Icons for Business Process Testing" on page 148. For component statuses, see "Business Components Module Fields" on page 65. For details on test or flow statuses, see "How ALM Calculates Business Process Test and Flow Statuses" on page 144.

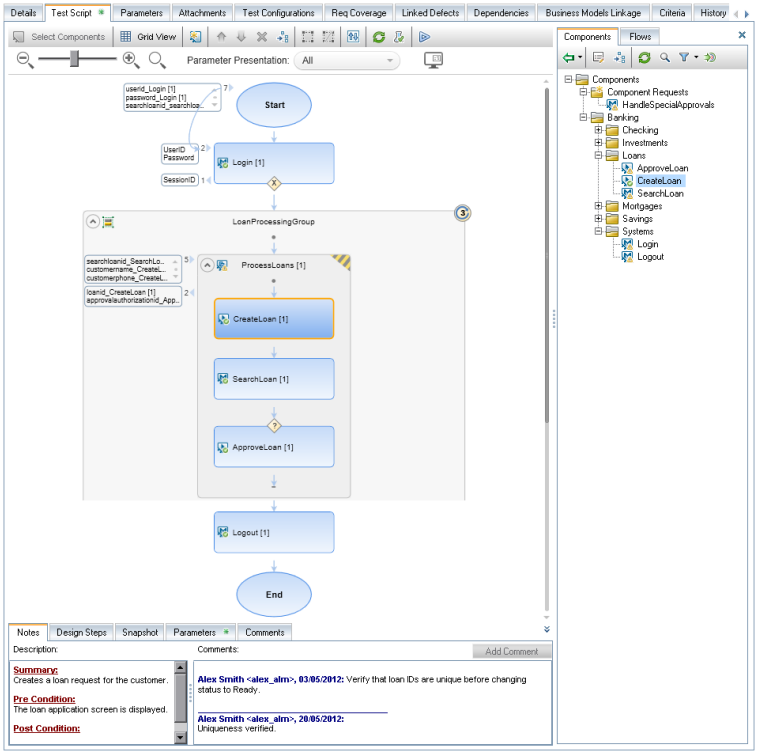
Test Script Tab for Business Process Testing

This tab enables you to create and organize business components into business process tests and flows, and flows into business process tests. The Test Script tab has a grid view and a canvas view.

Grid View

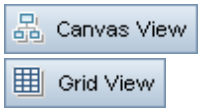


Canvas View



To access	In the Test Plan tree, select a business process test or flow and click the **Test Script** tab.
Important information	• For user interface details on the Test Script tab when the selected test is not a business process test or flow, see the *HP Application Lifecycle Management User Guide*. • **For Business Process Testing Enterprise Extension:** ▪ In addition to manually adding existing business components to a flow, you can create a flow comprising components created automatically by recording your operations as you navigate an enterprise application. For details, see "How to Learn Flows" on page 170. ▪ You can group components in a flow as you would in a business process test. However, it is recommended that you do not group components in a flow for compatibility with future versions of Business Process Testing Enterprise Extension.
Relevant tasks	"How to Create Business Process Tests and Flows" on page 134

User interface elements are described below:

UI Element	Description
	Select Components. Opens the Select Components and Flows Pane, enabling you to add content to your flow or business process test. For user interface details, see "Select Components and Flows Pane" on page 163.
	**Canvas/Grid View.** Toggles between the grid view and the canvas view. For user interface details on the grid view, see "Grid View" on page 157. For user interface details on the canvas view, see "Canvas View" on page 159.
	New Component Request. Enables you to request a new business component. For task details on generating a request for a new component, see the step "Request components - optional" on page 137.
	Learn Flow. Enables you to learn a flow automatically by navigating your enterprise application. For task details, see "How to Learn Flows" on page 170. **Available for:** Business Process Testing Enterprise Extension users with a flow selected. • Business Process Testing Enterprise Extensionmust be enabled for the current project. Access to Business Process Testing Enterprise Extensionmust be provided by your HP Application Lifecycle Management (ALM) Project Administrator using Project Customization. For details, see the *HP Application Lifecycle Management Administrator Guide*. • You must have the correct permissions to learn a flow, For details on the permissions needed to learn a flow, see "Prerequisites" on page 171, or see the *HP Application Lifecycle Management Administrator Guide*.
	Move up/down. Enables you to change the order of entities in the business process test by moving a selected component, group or flow up or down in the order. **Tip:** You can also change the testing order by dragging and dropping selected items.

UI Element	Description
	Remove from Test. Removes the selected business component, group, or flow from the business process test (or flow). **Not available when:** Attempting to remove a business component from a flow whose business process test is selected in the test plan tree. First, select the flow in the test plan tree, and then delete the business component. **Note:** If criteria are specified for a component that is about to be removed, a warning message is displayed. The component is still available for future use, if required, from the component tree. **Tip:** You can also change the testing order by dragging and dropping selected items. **Caution:** If you are deleting the last component in a group, the entire group is deleted.
	Go To Component/Flow. Goes to, and opens, the selected business component in the Business Components module or the flow in the Test Plan module.
	Group. Creates a group that includes the selected business components and/or flows. The components and flows must be contiguous. A component or flow can belong to one group only. A group node is created above the grouped items and is identified by the group icon . The group is named, by default, **Group** followed by a unique number. **Tip:** • You can rename the group by right-clicking its name and selecting **Rename**. • You can add other business components or flows to an existing group by dragging and dropping a component or flow from the component tree or the test script pane to the relevant position in the group. • You can change the order of the members of the group by dragging and dropping.

UI Element	Description
	Ungroup. Ungroups components and/or flows. To completely remove a group, including its members, select the group and click the **Remove from Test** button in the toolbar. To remove a business component or flow from a group, select the component or flow. Drag the item up or down out of the group, and drop it at the required location.
	Show Test Parameters Promotion Status. Enables you to view a list of test and flow parameters that were promoted and where they were originally defined. For user interface details, see "Test/Flow Parameters Promotion Status Dialog Box" on page 280.
	Refresh. Updates the data, such as the component parameter data and snapshot, for each business component in the business process test. The test itself is not updated.
	Validate. Checks the business process test or flow and all the test instances within the test set for errors. For user interface details, see "Test Validation Results Dialog Box" on page 166.
	Run or Debug Test. Enables you to run or debug a business process test or flow. For user interface details, see"Run or Debug Test Dialog Box" on page 224.
	Add/Edit Run Condition. Enables you to add run conditions to your flow. For user interface details, see "Run Conditions Dialog Box" on page 196. **Available when**: A flow is selected.
	Delete Run Condition. Enables you to remove an existing run condition from your flow. **Available when:** A flow is selected.
	Stop Run. Enables you to stop the current run or debug session. **Available when:** Running or debugging a test from the Test Plan module.
Iterations option	Opens the iterations dialog box for the selected entity. For user interface details, see "Iteration Pages" on page 292. **Available from:** <right-click menu>
	Hides or shows the tabs at the bottom of the Test Script tab.

UI Element	Description
Grouping option	Opens a submenu with the **Group** and **Ungroup** options. **Available from:** <right-click menu>. If more than one component is selected, the **Group** option is available. If a group is selected, the **Ungroup** option is available.
Expand all below this node/Collapse option	Expands/collapses the node for the selected entity. **Available from:** <right-click menu>, for flows and groups
Details option	Opens the Details dialog box for the component, component request, or flow. **Available from:** <right-click menu>
Notes	Displays the description and comments of the business component, flow, or group in read-only format.
Snapshot	Enables you to add an image of your application to an ALM record.
Manual Steps tab	Displays the manual steps defined for the component.
Parameters	Lists the parameters defined for the component or flow.
Comments	Displays additional comments for the component instance.

Grid View

The grid view enables you to modify your test scripts in a table-like format.

User interface elements are described below:

UI Elements	Description
Name	Displays the name of the business component or flow. **Note:** If an entity is selected for the current test or flow multiple times, an instance notation is automatically appended to the name in this column. Example **Syntax Exceptions**: • **Flows and business process tests**: Do not include two consecutive semicolons (`;;`) or any of the following characters: **\ / : " ' ? ` < > | * %**

UI Elements	Description
Status	Displays the status of the business component or flow.
I/O Parameters	Displays the number of input and output parameters defined for the business component or flow as a dynamic link, which opens the I/O Parameters dialog box. For user interface details, see "I/O Parameters Dialog Box" on page 272.
Iterations	Displays the number of iterations defined for the business component, group, or flow, followed by the range of selected iterations. This information is displayed as a dynamic link, which opens the Iterations dialog box for the selected business component, flow, or group. For task details, see "How to Set Data for Iterations" on page 287.
Run Conditions	Displays a link to the run conditions defined for the business component. **Tip:** If a run condition is not valid, the run condition link is displayed in red. This can happen, for example, if a reference parameter was deleted, a parameter value was encrypted, and so on. Delete the run condition and define a new one.
On Failure	Enables you to define whether a run continues or ends if a specific business component or flow in the test fails. • **Exit.** The business process test run ends if the selected business component fails. • **Continue.** The business process test will run the next business component or flow if the selected component fails. By default, this failure condition is defined for each component when it is added to a test.
Comments	Displays the comments for the business component, group, or flow.

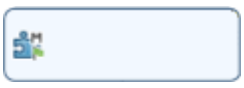
Canvas View

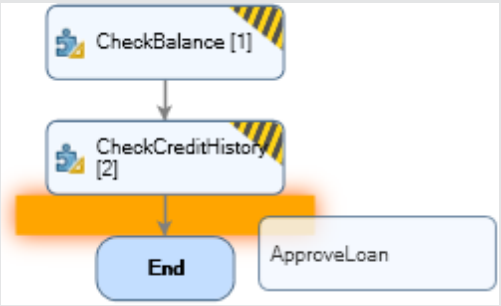
The canvas view enables you to modify your test scripts using a graphical display.

Important information	The canvas view includes all the functionality of the grid view. The canvas enables you to visualize the data flow in the test or flow, including parameters and links. The following describes how to work in the canvas view. • Drag entities onto the canvas from the Select Components and Flows pane. For details on the Select Components and Flows pane, see "Select Components and Flows Pane" on page 163. • Drag entities within the canvas to reorder them. • Drag components into or out of groups. • Double-click a component, component request, or flow node to view the details in a separate window. **Note:** You cannot edit component request details by double-clicking its node from a flow within a business process test. The component request opens in read-only mode. • Double-click a group to expand or collapse it. • Hold down the CTRL key and use the mouse wheel to zoom in and out.

User interface elements are described below:

UI Elements	Description
	Zoom Out. Decreases the zoom level of the canvas. **Tip:** You can press and hold this button for quicker zooming.
	Zoom Level. Increases/decreases the zoom level of the canvas using a slider.
	Zoom In. Increases the zoom level of the canvas. **Tip:** You can press and hold this button for quicker zooming.
	Reset Zoom. Resets the zoom level to the default zoom settings.

UI Elements	Description
	Show Minimap. Toggles the display of the overview in the canvas view. Located in the lower-left side of the canvas view, the overview is a reduced canvas of the currently-selected business process test or flow. The framed area of the overview is magnified in the canvas. **Example** **Tip:** • Select an area in the overview to focus on the selected area in the canvas. • Drag or resize the frame to change the focus of the model in the canvas.
Parameter Presentation	Enables you to select the detail level for parameters and links in the canvas. The following levels are available: • **None** • **Links only** • **Links and names** • **All** • **User defined**
	Start node. Represents the start of the business process test or flow. For display purposes only.
	**Component node.** Represents a component entity.
	Component Request node. Represents a component request entity.

UI Elements	Description
	Group node. Represents a group entity. **Tip:** • Click the expand and collapse buttons to show/hide components in the group. • Click the group name to edit the name. • You can drag and drop components into and out of groups.
	Flow node. Represents a flow entity. **Tip:** Click the expand and collapse buttons to show/hide components in the flow.
	Drop indicator. When selecting an entity from the Select Components and Flows pane into the canvas, or moving an entity within the canvas, this indicator helps you position the entity. The **ApproveLoan** component is being pasted after the second instance of the **CheckCreditHistory** component. 
	Status. Located at the upper right of an entity's node, indicates the status of the entity. • A red triangle indicates that the entity (or one of its sub-entities) has an **error**. • A striped triangle indicates that the entity (or one of its sub-entities) has a status of **Maintenance**, **Under Development**, or **Not Implemented**. • Lack of a triangle indicates that the entity (and all of its sub-entities) is **ready**.
	Collapse. Hides nodes for the selected entity.

UI Elements	Description
	Expand. Shows nodes for the selected entity.
	Entity Order. Indicates the order of the entities in the business process test or flow.
	Input parameters. Indicates the number of input parameters for the entity. **Tip:** Click to display a scrollable list of the entity's input parameters.
	Output parameters. Indicates the number of output parameters for the entity. **Tip:** Click to display a scrollable list of the entity's output parameters.
	Promotion/Linkage Arrow. Indicates from where parameters were promoted or to where parameters are linked. **Note:** When many links exist between parameters, click the triangle next to a parameter name to display that parameter's link in yellow. **Tip:** Hover over the arrow to see the name of the target node of the link.
[<instance>]	**Instance.** Indicates the instance of the entity in the test or flow, if the entity was selected multiple times. **Examples:** [1] [2]
	Iterations. Indicates the number of iterations selected for the entity in the test or flow. For example, if an entity has 5 iterations, but only iterations 3 and 4 are selected to run, this **Iterations** UI element will display **2**. Click to open the Iterations dialog box for the entity. Hover over to see a tooltip that displays which iterations are selected. **Tip:** If the number of iterations is greater than 999, a K notation is used (for example, 3K is displayed if there are 3145 iterations. If the number of iterations is greater than 999,999, an M notation is used (for example, 1M is displayed if there are 1,334,452 iterations). For user interface details, see "Iteration Pages" on page 292.

UI Elements	Description
	Exit. Indicates the component has an On Failure condition that is set to exit.
	Run condition. Indicates the component has a run condition. Click to open the Run Condition dialog box. Hover over to see a tooltip that displays the run condition. For user interface details, see "Run Conditions Dialog Box" on page 196.
	Notification. Click to see the notification, such as **(iterations mismatch)**.
	End node. Node representing the end of the business process test or flow. For display purposes only.

Select Components and Flows Pane

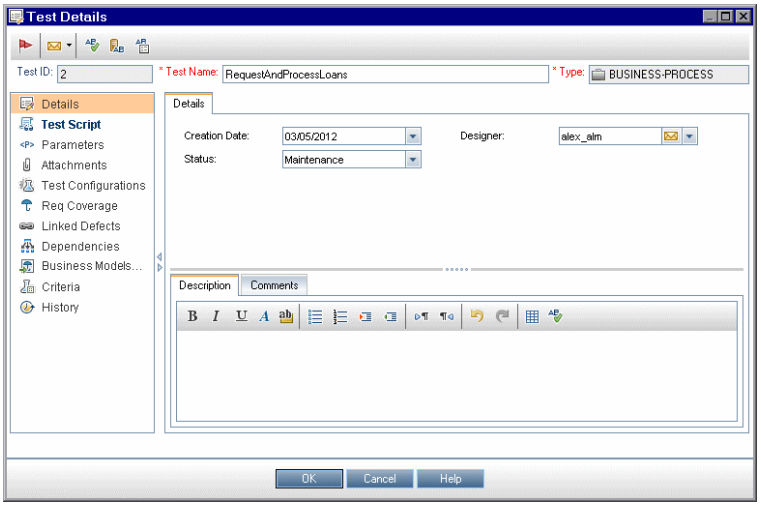
This pane enables you to select business components and flows for inclusion in a business process test (or flow). ALM adds instances of the selected components or flows to the test (or flow).

User interface elements are described below:

UI Element	Description
Components	Displays the component tree, from which components can be added to flows and tests in the Test Script tab.
Flows	Displays the flows tree, from which flows can be added to tests in the Test Script tab.
	Quick Add. Adds the selected business component or flow after the component, group or flow selected in the Test Script tab (or at the end of the test if no component or flow is selected). As the component or flow is added, parameters, if any exist, are promoted according to the last promotion method used. • **Add and Automatically Promote All Parameters**. All parameters are promoted to the next level as the component or flow is added. • **Add without Promoting Parameters**. Parameters are not promoted to the next level as the component or flow is added. • **Add while Setting Promote Options**. The Promote Parameters dialog box opens, enabling you to individually set which parameters should be promoted. For user interface details, see "Promote Parameters Dialog Box" on page 278. For task details on promoting parameters, see "How to Promote Parameters" on page 276.
	Show Entity Details. Opens the Details dialog box for the selected entity in read-only mode.
	Go to Entity in Module Tree. Navigates to the selected entity's location in that entity's module, and highlights the entity.
	Refresh. Refreshes the tree.
	Find. Searches for a specific test in the tree. Type the name (or part of the name) of the test in the Find box and click **Find**. If the search is successful, the test is highlighted in the tree. For more details, see *HP Application Lifecycle Management User Guide*.
	Filter/Sort. Filters and sorts the test in the tree. For more details, see *HP Application Lifecycle Management User Guide*.
	Go to Component/Test by ID. Opens the Go To Component/Test dialog box, enabling you to find a specific component by component ID of a flow by test ID. **Note**: You can only go to components or flows that are in the current filter.
	Close. Hides the Select Components and Flows pane.

Test Details Dialog Box for Business Process Testing

The Test Details dialog box for Business Process Testing enables you to view and update a single business process test or flow.



To access	On the ALM sidebar, under **Testing**, select **Test Plan**. Right-click a business process test or flow, and select **Test Details**.
Important information	The Test Details dialog box for business process tests and flows is similar to the Test Details dialog box for other test types. **Note:** The Test Details dialog box for business process tests and flows does not have a Design Steps tab.
Relevant tasks	"How to Create Business Process Tests and Flows" on page 134
See also	*HP Application Lifecycle Management User Guide*

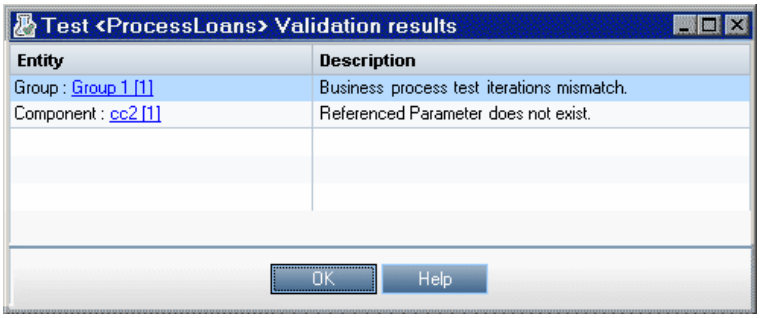
Business Process Testing UI elements that are unique to, or operate differently from, other test types are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
<ALM Test Detail dialog box UI elements>	For UI element descriptions for the ALM Test Details dialog box for all test types, see the *HP Application Lifecycle Management User Guide*.
Test Script	The components and flows that are run for the currently selected business process test (or the components that are run for the currently selected flow). For user interface details, see "Test Script Tab for Business Process Testing" on page 152.

UI Element	Description
Test Configurations	Displays configurations of a selected test. When the selected test is a business process test, additional functionality is available that enables you to run test configurations using dynamic data, in addition to static data. **Not available when:** A flow is selected. For user interface details, see the *HP Application Lifecycle Management User Guide*. For task details, see "How to Work with Business Process Test Configurations" on page 203.
Criteria	Displays criteria of the selected business process test. **Not available when**: A flow is selected. For user interface details, see the *HP Application Lifecycle Management User Guide*. For task details, see "How to Create Coverage by Criteria" on page 189.

Test Validation Results Dialog Box

This dialog box enables you to validate all instances of the business process test or flow.



To access	In the Test Plan tree, select the flow or business process test to validate and click the **Test Script** tab. Click **Validate** .
Important information	If no validations errors are found, a message is displayed indicating that the validation was successfully completed. In the case of an iteration range mismatch, the error is indicated in the second component.
Relevant tasks	"How to Create Business Process Tests and Flows" on page 134

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

UI Element	Description
Entity	Displays the entities that failed validation.
Description	Describes validation errors.
<Entity link>	Enables you to jump directly to the business component or group causing the error.

Chapter 9

Learn Flow Process

This chapter includes:

Learn Flow Process Overview 170

How to Learn Flows 170

Criteria for Determining if a Component Can Be Reused 175

Learn Flow Process User Interface 177

Learn Flow Process Overview

This chapter describes how Business Process Testing Enterprise Extension users can learn flows automatically by navigating through environment-specific applications.

During the learn flow process, Business Process Testing Enterprise Extension:

- Learns the actions performed when navigating enterprise applications.
- Breaks down the learned actions into a sequence of automated UFT keyword GUI business components, each representing a screen, tab, or transaction in your application.
- Creates steps in each component based on the operations you perform.
- Populates an existing flow with the learned automated components.
- Automatically creates input component parameters for controls in the user interface that require user input, such as text fields. The default value for these parameters is the value inserted in these fields during the learn flow process.
- Automatically creates output parameters.

  Tip: By default, output parameters that are added during the Learn Flow process are promoted to flow parameters. You may want to remove an output parameter from the flow level and define it as an output component parameter if it is used only within its flow.

- Enables you to insert checkpoints and output values into the flow.
- Analyzes each of the components in the flow to see if one or more components exists that are similar to, or identical to, the learned components. If such a component exists, you can reuse the existing component instead of creating a new component in your flow.

Note: During the Learn Flow process, resources are created in the **BPT Resources** folder in the Test Resources module. For business process tests or flows to perform properly, this folder and its subfolders should not be renamed, moved, or deleted.

For enterprise-specific details on how Business Process Testing Enterprise Extension breaks up a learned flow into components and creates parameters, see "Business Process Testing Enterprise Extension Environment-specific Information" on page 339

For task details, see "How to Learn Flows" below.

How to Learn Flows

This task describes how to learn a flow automatically while navigating your application.

Note:

- If you do not have a license for Business Process Testing Enterprise Extension, the features described in this chapter are not relevant to you.

- This task is part of a higher-level task. For details, see "How to Create Business Process Tests and Flows" on page 134.

This task includes the following steps:

- "Prerequisites" below
- "Open and log into the enterprise application to learn" on next page
- "Create or select a flow in ALM" on next page
- "Start the Learn Flow wizard in ALM" on next page
- "Learn the enterprise application" on page 173
- "Add checkpoints while learning the enterprise application" on page 173
- "Add output values while learning the enterprise application" on page 174
- "Stop the Learn Flow wizard" on page 174
- "Review the Learn Flow summary" on page 174
- "Change the default location for learned components - optional" on page 174
- "Reuse components - optional" on page 174
- "Save the learned components into the flow" on page 175
- "Results" on page 175

1. ## Prerequisites

   Consider the following:

   - Make sure Business Process Testing Enterprise Extension is enabled. Access to Business Process Testing Enterprise Extension must be provided by your HP Application Lifecycle Management (ALM) Project Administrator using Project Customization. An administrator can enable Business Process Testing Enterprise Extension in ALM by choosing **Tools > Customize > Business Process Testing** and selecting the **Enable Business Process Testing Enterprise Extension** checkbox. For details, see the *HP Application Lifecycle Management Administrator Guide*.
   - We recommend you close UFT before learning a flow.
   - Make sure that you belong to a user group that has the following task permissions: **Modify Folder (Test Plan), Modify Test, Add Component Folder, Add Component, Add Step, Add Parameter, Modify Component, Modify Step, Modify Parameter**.
   - In UFT, select **Tools > Options** and click the **Run** node. Ensure that the **Allow other HP products to run tests and components** check box is selected in the Run pane.
   - Set Business Process Testing Enterprise Extension reuse options. For details, see the *HP Application Lifecycle Management Administrator Guide*.

For additional details on configuring UFT to work with enterprise applications, setting up Business Process Testing Enterprise Extension, and on configuring user permissions, see the *HP Application Lifecycle Management Administrator Guide*.

2. **Open and log into the enterprise application to learn**

   Log into the application for which you want to learn the flow, and navigate to the point at which you want to start learning the flow.

   Business Process Testing Enterprise Extension is not able to learn the login procedure.

3. **Create or select a flow in ALM**

   a. In the Test Plan module, before learning a flow, do one of the following:

      - Create a new flow in which to insert learned components. For task details on creating flows, see "How to Create Business Process Tests and Flows" on page 134.
      - Select an existing flow in which to insert learned components.

   b. If there are already components in the flow, and you want to insert the new components between existing components, in the Test Script tab, select the component after which you want to insert the new components.

   > **Note:** To insert the new components at the beginning of the flow (before existing components), you must first insert the new components to a different location, and then change the order of the components in the flow manually after the learn process is finished.

   Example:

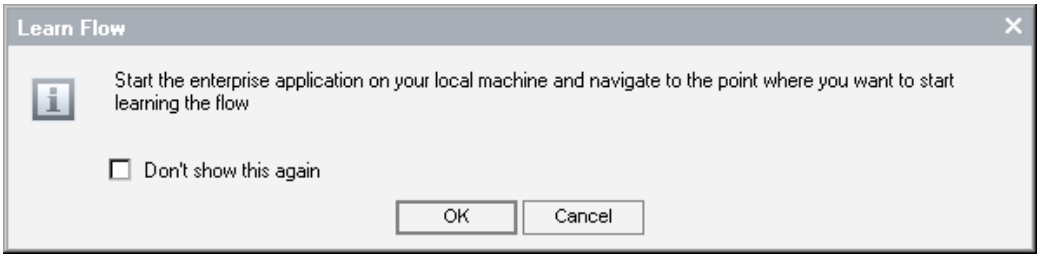
   Suppose your flow contains the components `Comp_1`, `Comp_2`, and `Comp_3`. If you want to insert the new components between `Comp_2` and `Comp_3`, select `Comp_2`.

4. **Start the Learn Flow wizard in ALM**

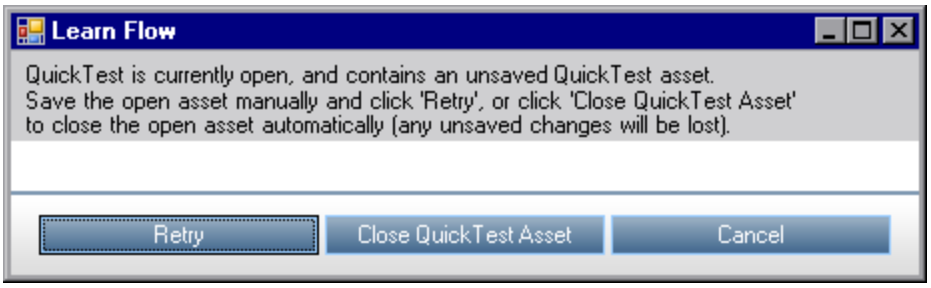
   a. In the Test Script tab toolbar, click the **Learn Flow** button to start the Learn Flow wizard.

   b. If there are already business components in the flow, the Learn Flow dialog box opens. For user interface details on the Learn Flow dialog box, see "Learn Flow Dialog Box Page " on page 178

   c. A message box reminds you that you must open your application. Click **OK** to continue. If you do not want to receive this reminder in the future, select the **Don't show this again** check box.



d. If other messages appear related to granting access to other assets, scripts, or applications, resolve the issues and click **Retry**.



e. The Learn Flow toolbar opens. Click the **Learn Flow** button. The button changes to **Stop Learn**, and the status in the toolbar changes from **Ready** to **Learning**. For user interface details on the Learn Flow toolbar, see "Learn Flow Toolbar Page" on page 178

5. Learn the enterprise application

Perform the operations that you want to learn. In general, the operations you perform for a single flow should represent those of a single screen or transaction in your application.

While Business Process Testing Enterprise Extension is learning the flow, you can control the Learn Flow process using the Learn Flow toolbar. For user interface details on the Learn Flow toolbar, see "Learn Flow Toolbar Page" on page 178.

Tip:

- To enable more efficient component reuse, it is recommended that you perform the same operation in the same manner each time you learn a flow. For example, clicking the **Enter** button or pressing the ENTER key on the keyboard results in different learned steps. Therefore, if you are not consistent when performing such operations, two otherwise identical components might be considered only similar rather than identical. For reference details on how components are selected for reuse, see "Criteria for Determining if a Component Can Be Reused" on page 175.
- Using keyboard input to navigate your application instead of clicking buttons outside your screen or tab results in fewer learned components in your flow. For example, clicking the **Enter** button during the Learn Flow process results in the creation of a new component for the **Enter** button, whereas pressing ENTER results in an additional step in the existing component. For enterprise-specific details, see "Business Process Testing Enterprise Extension Environment-specific Information" on page 339.

6. Add checkpoints while learning the enterprise application

You can add object property checkpoints to a component during the learn flow process. Object property checkpoints enable you to check object property values during a test run to determine if they match expected values. The test results provide information about which object property checkpoints failed during a test run, and why.

a. Perform the operations you want to learn up to the point that you want to insert a checkpoint.

b. Click the **Add Checkpoint** button on the Learn Flow toolbar. The cursor changes to a pointing hand and the Learn Flow toolbar is hidden.

c. Point to the object in your application for which you want to create an object property checkpoint. The Checkpoint Properties dialog box opens. For user interface details, see "Checkpoint Properties Dialog Box" on page 115.

d. By default, all the properties in the dialog box are selected for verification. Clear the check box for any values you that do not want to check and click **OK**. The object property checkpoint is added to the component.

e. Continue performing operations in your application to continue learning the flow.

7. Add output values while learning the enterprise application

You can add an object property output value step to a component during the Learn Flow process. Object property output value steps allow you to capture and store values during your test. These values can later be used as input at a different point in the run session.

a. Perform the operations you want to learn up to the point that you want to insert an output value.

b. Click the **Add Output Value** button on the Learn Flow toolbar. The cursor changes to a pointing hand and the Learn Flow toolbar is hidden.

c. Point to the object on the screen for which you want to create an output value. The Output Value Properties dialog box opens. For user interface details on the Output Value Properties dialog box, see "Output Value Properties Dialog Box" on page 118.

d. By default, none of the properties are selected for the Output. Select those properties whose values you want to output and click **OK**. The output value is created and the object property output value step is added to the component.

e. Continue performing operations in your application to continue learning the flow.

8. Stop the Learn Flow wizard

When you finish performing the operations that you want to learn, click the **Stop Learn** button in the Learn Flow toolbar.

9. Review the Learn Flow summary

After stopping the learn flow process, the Learn Flow Summary dialog box opens. For user interface details on the Learn Flow Summary Page, see "Learn Flow Summary Page" on page 180.

You can use the Learn Flow Summary dialog box to view details of the learned steps.

10. Change the default location for learned components - optional

In the Learn Flow Summary dialog box, you can change the default location in which learned business components are stored. For user interface details on the Select Business Components Folder dialog box, see "Select Business Components Folder Dialog Box" on page 183.

11. Reuse components - optional

In the Learn Flow Summary dialog box, you can reuse an existing component in your learned

flow in place of creating a new component. Select a component that can be substituted with a reused component from the flow tree in the Learn Flow summary. For user interface details on the Available Components for Reuse dialog box, see "Available Components for Reuse Dialog Box " on page 184.

12. **Save the learned components into the flow**

    Click **Save** to close the Learn Flow Summary dialog box, create the new components to the Business Component module, and add the components to the flow.

    Creation of business components for a learned flow may take some time.

13. **Results**

    Business Process Testing Enterprise Extension performs the following:

    - Creates automated business components in the Business Components module that correspond to screens, transactions, and so on, in your application.
    - Creates component steps based on the operations you perform.
    - Adds the created components to the flow. Components that are created through the Learn Flow process are indicated with a special learned component icon .
    - Inserts the checkpoints and output values you defined.
    - Automatically creates input component parameters for controls in the application's user interface that require user input, such as text fields. The default value for these parameters is the value you insert in these fields during the Learn Flow process.

Criteria for Determining if a Component Can Be Reused

When Business Process Testing Enterprise Extension learns a flow, it analyzes the learned components to see if there exists within the project one or more components that are similar to, or identical to, the learned components. If such a component exists, Business Process Testing Enterprise Extension enables you to reuse the existing component instead of creating a new component in your flow.

Only components in the project that were created through the Learn Flow process are analyzed for similarity with the learned component. If components are similar to the learned component, it can be reused, and a new learned component is not created.

Note: A component that is substituted with a reused component is not listed in the flow folder of the Business Components module. A flow folder is not created in the Business Components module for flows that use only reused components. To find the location of a reused component, right-click the component in the **Test Script** tab of the **Test Plan** module and select **Go to Component**. The Business Component module opens and displays the selected component.

This section contains the following topics:

- "Criteria for Comparison " on next page
- "Percentage Similarity" on next page

- "Other Factors" below

Criteria for Comparison

Existing learned components in the project are compared to the learned component using the following criteria:

1. **Both components represent the same screen.**
2. **Both components represent the same screen with exactly the same objects.** Checkpoints and Output objects must also be identical for this criterion to match.
3. **Both components contain the same steps.** Steps are considered identical when they perform the same action, and refer to the same objects.
4. **Both components contain the same steps in the same order.**

Percentage Similarity

Percentage similarity is defined as follows, based on the criteria specified in "Criteria for Comparison " above:

Percentage	Definition
100	All four criteria are met.
75	Criteria **1**, **2**, and **3** are met.
50	Criteria **1** and **2**, or **1** and **3**, are met.
25	Criterion **1** is met.

Other Factors

The following entities also determine if the currently-learned component can be substituted with an existing learned component:

Entity	Similarity
Checkpoints and Output objects	Are considered identical when they refer to the same properties.
Steps that reference table parameters	Are considered identical when they reference the same table object and use the Input operation. The structure of the table parameter (columns, rows) and the contents of individual cells within the tables are not compared.
Steps that reference methods	Are considered identical when they refer to the same objects and methods, and contain the same number of arguments. The values of the arguments are not compared.

Learn Flow Process User Interface

This section includes:

Learn Flow Wizard 177
Select Business Components Folder Dialog Box 183
Available Components for Reuse Dialog Box 184

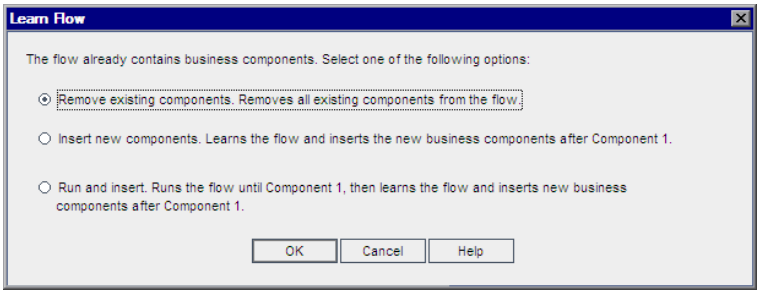
Learn Flow Wizard

This wizard enables you to learn a flow in an enterprise application.

To access	In the Test Plan module, in the Test Script tab, select the flow in which you want to insert learned components. Click **Learn Flow** .
Wizard map	This wizard contains: ("Learn Flow Dialog Box Page " on next page) > "Learn Flow Toolbar Page" on next page > "Learn Flow Summary Page" on page 180
Important information	This wizard is available to Business Process Testing Enterprise Extension users with a flow selected. This wizard is not available when: • Business Process Testing Enterprise Extension is not enabled for the current project. Access to Business Process Testing Enterprise Extension must be provided by your HP Application Lifecycle Management (ALM) Project Administrator using Project Customization. An administrator can enable Business Process Testing Enterprise Extension in ALM by choosing **Tools > Customize > Business Process Test** and selecting the **Enable** Business Process Testing Enterprise Extension checkbox. • You do not have the correct permissions to learn a flow, For details on the permissions needed to learn a flow, see "How to Learn Flows" on page 170, or see the *HP Application Lifecycle Management Administrator Guide*.
Relevant tasks	"How to Learn Flows" on page 170
See also	"Learn Flow Process Overview " on page 170

Learn Flow Dialog Box Page

This wizard page enables you to determine how ALM should handle existing components in the flow before beginning the learn flow process.



Wizard map	The "Learn Flow Wizard" on previous page contains: ("Learn Flow Dialog Box Page " above) > "Learn Flow Toolbar Page" below > "Learn Flow Summary Page" on page 180
Important information	General information about this wizard is available here: "Learn Flow Wizard" on previous page.
See also	"How to Learn Flows" on page 170

User interface elements are described below:

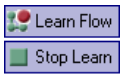
UI Element	Description
Remove existing components	Removes all the existing business components from the flow and learns the components in the emptied flow.
Insert new components	Learns the flow and inserts the resulting components after the selected existing component.
Run and insert	Runs the existing components in the flow until the selected component, then learns the flow, and inserts the resulting components after the selected existing component.

Learn Flow Toolbar Page

This wizard page enables you to start and control the learn flow process.

Wizard map	The "Learn Flow Wizard" on page 177 contains: ("Learn Flow Dialog Box Page " on previous page) > "Learn Flow Toolbar Page" on previous page > "Learn Flow Summary Page" on next page
Important information	General information about this wizard is available here: "Learn Flow Wizard" on page 177.
See also	"How to Learn Flows" on page 170

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

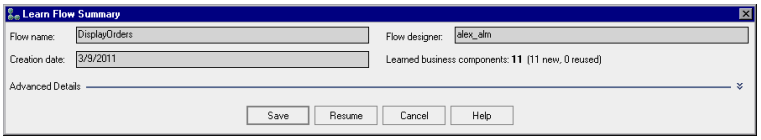
UI Element	Description
<Status>	Displays the status of the learn flow process. • **Ready**. Business Process Testing Enterprise Extension is not yet learning a flow. • **Learning**. Blinking red and white, Business Process Testing Enterprise Extension is learning a flow.
<Screens Learned>	Indicates the number of screens learned during the Learn process. **Available when**: Business Process Testing Enterprise Extension is learning a flow.
	**Learn Flow/Stop Learn.** Starts or stops learning the actions in the enterprise application.
	Pause/Resume. Pauses or resumes learning a flow. When the Learn Flow process is paused, any actions you perform on your application are not learned. To resume learning actions, click the **Resume** button.
	Add Checkpoint. Inserts a checkpoint into the automated component. A checkpoint is a verification point that compares a current value for a specified property with the expected value for that property. This enables you to identify whether your application is functioning correctly. The test results provide information about which object property checkpoints failed during a test run, and why. For user interface details, see the "Checkpoint Properties Dialog Box" on page 115
	Add Output Value. Adds an output value to the automated component. An output value is a step in which one or more values are captured at a specific point in your test and stored for later use during a test. These values can later be used as input at a different point in the run session. For user interface details, see "Output Value Properties Dialog Box" on page 118.
	Help. Opens help for the Learn Flow toolbar.
	Cancel. Cancels the current Learn Flow process. If you cancel the Learn Flow process, all actions learned during that session are not saved.

Learn Flow Summary Page

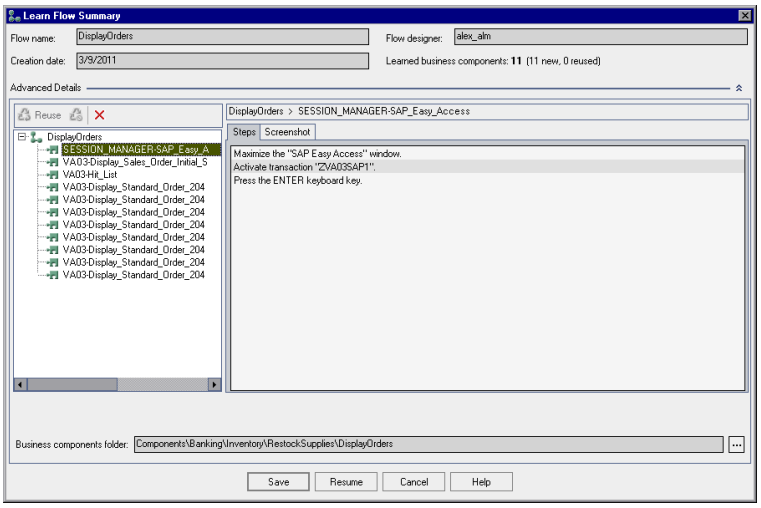
This wizard page enables you to:

- View summary information on the learn flow process.
- View the details of the steps performed during the learn flow process.
- View information about, and save, the learned components.

Learn Flow Summary: Basic View



Learn Flow Summary: Advanced Details View



Wizard map	The "Learn Flow Wizard" on page 177 contains: ("Learn Flow Dialog Box Page " on page 178) > "Learn Flow Toolbar Page" on page 178 > "Learn Flow Summary Page" above
Important information	• General information about this wizard is available here: "Learn Flow Wizard" on page 177. • The Learn Flow Summary dialog box is displayed with the Advanced Details section collapsed by default.
See also	"How to Learn Flows" on page 170

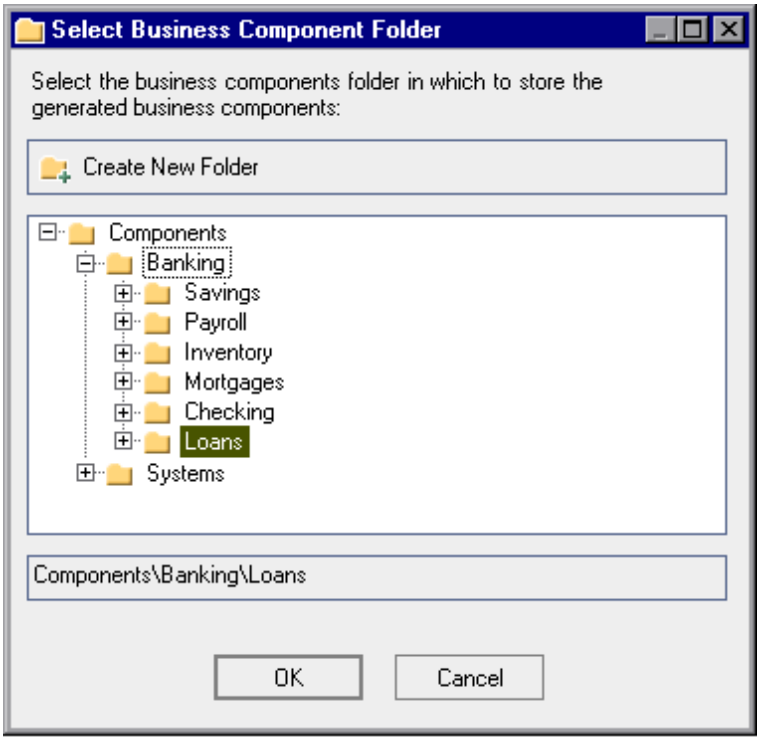
User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Flow name	Displays the name of the flow.
Flow designer	Displays the ALM user name of the user who designed the flow.
Creation date	Displays the date on which the flow was created.
Learned business components	Displays the number of learned components. Indicates how many components will be created as new components and how many will reuse existing components in the project.
Advanced Details	Performs the following: • Displays additional information about the learned flow. • Enables you to reuse existing components in your project in place of creating new components for your flow. For user interface details, see "Available Components for Reuse Dialog Box " on page 184. • Enables you to change the default location for newly created components. For user interface details, see "Select Business Components Folder Dialog Box" on page 183. The Advanced Details are collapsed by default. Click **Expand** at the far right to expand and collapse this area.
	Expand/Collapse. Opens and closes the Advanced Details area.
Save	Saves the new components to the Business Component module, and adds them to the flow.
Resume	Resumes learning the flow without saving the learned components.
Cancel	Cancels learning the flow. All the learned components are deleted.
	Reuse. Opens the Available Components for Reuse dialog box, which enables you to select components for reuse instead of creating new components. For user interface details, see "Available Components for Reuse Dialog Box " on page 184. **Tip:** To select an alternate component for reuse, click the **Reuse** button again. **Available if:** Components exist in the project that are identical to, or similar to the component selected in the flow tree.

UI Element	Description
	Cancel Reuse. Cancels reuse for the selected component. A new component is created for the selected component. For user interface details, see "Available Components for Reuse Dialog Box " on page 184. **Available for**: Reused components.
	Remove Component from Flow. Deletes the selected component from the learned flow.
<flow tree>	Displays the business components learned as part of the flow. When you select a component you can view its steps in the **Steps** tab, and a screenshot of the application as it appeared at the beginning of the component's execution in the **Screenshot** tab. The component icon in the flow tree indicates the component's reuse status. For user interface details on reuse icons, see "Test Plan Module Icons for Business Process Testing" on page 148.
<component name>	Displays the full name of the component including its flow.
Steps tab	• For a learned component, the Steps tab displays a description of the steps in the currently selected business component. • For a reused component, the Steps tab displays a description of the steps in the reused component. • For a reused component that is identical to the learned component, the **Steps** tab displays the steps and values of the component that will be saved with the flow. For details on which values will be saved with the flow, see HP Application Lifecycle Management Administrator Guide.
Screenshot tab	Displays a screenshot of your application as it appeared at the beginning of the selected component in the flow. For a reused component, the **Screenshot** tab displays the screenshot of the reused component.
<reuse summary>	Displays the reuse status of the selected component in the flow tree.
Business components folder	Displays the default location where the learned components are stored.
	Changes the location where learned components are stored by browsing to the new location. For user interface details, see "Select Business Components Folder Dialog Box" on next page.

Select Business Components Folder Dialog Box

This dialog box enables you change the location where learned business components are stored.



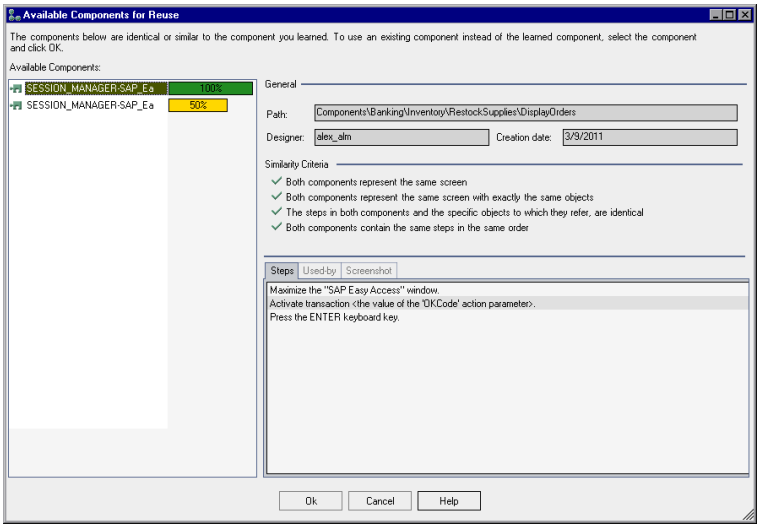
To access	After learning a flow, in the Advanced Details area of the Learn Flow Summary dialog box, click **Browse** ... next to **Business components folder**.
Important information	By default, learned components are added to the Business Component tree in a folder with a path parallel to the path of the flow in the Test Plan Tree. For example, if the flow `MyFlow` is located in the folder `Subject\MyFlows` in the Test Plan Tree, the learned business components are, by default, created in the folder `Components\MyFlows\MyFlow`. You can change the default location for learned components using this dialog box.
Relevant tasks	"How to Learn Flows" on page 170
See also	"Learn Flow Process Overview " on page 170

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

UI Element	Description
Create New Folder	**Create new Folder.** Enables you to create a new folder in the component folder tree.
<Component Folder Tree>	Lists the component folders, from the Business Components module, from which you can select a new default location for learned business components.
<Component Path>	Displays the currently selected path.

Available Components for Reuse Dialog Box

This dialog box enables you to reuse an existing component in the project, instead of using the newly-learned component.



To access	In the Learn Flow Summary dialog box, select a component and click the **Reuse** button.
Important information	Component reuse is customizable via the **Business Process Testing Enterprise Extension**page in the **Project Customization** screen (**Tools > Customize**). For details, see *HP Application Lifecycle Management Administrator Guide*.
Relevant tasks	"How to Learn Flows" on page 170
See also	"Criteria for Determining if a Component Can Be Reused" on page 175

User interface elements are described below:

UI Element	Description
Available Components area	Lists components in the project that are identical to, or similar to, the learned component. Percentages indicate the degree to which the existing component is similar to the learned component. For reference details, see "Criteria for Determining if a Component Can Be Reused" on page 175.
General area	Displays general information about the selected component. • **Path.** The location of the existing component. • **Designer.** The ALM user name of the user who designed the flow. • **Creation date.** The date and time on which the flow was created.
Similarity Criteria area	The criteria by which the existing component is found to be similar to the learned component. A check mark next to the criterion indicates that the criterion was met. For reference details, see "Criteria for Determining if a Component Can Be Reused" on page 175.
Steps tab	Displays the steps in the component selected in the **Available Components** area.
Used-by tab	Displays details for all the flows and tests that use the selected component. The details include the flow name, path, status, owner, and creation date.
Screenshot tab	Displays a screenshot of the selected component, at the beginning of its execution.

Chapter 10

Requirement Coverage

This chapter includes:

Requirement Coverage Overview 188
How to Create Coverage by Criteria 189

Requirement Coverage Overview

Like in ALM, you can create coverage between requirements and business process tests. The difference is that when working with business process testing, instead of covering each requirement only at the level of a test and its configuration(s), you can define coverage by criteria, such as flows or business components. This enables you to determine if a test passed or failed on a more granular level. For example, a test can be considered passed even if just one of its business components passed, such as the most critical one. Less critical business components have no effect on the overall status of the test.

The logic used for calculating coverage by criteria is similar to the logic used for calculating coverage by tests and test configurations. For conceptual details on coverage analysis for tests and test configurations, see the *HP Application Lifecycle Management User Guide*.

For task details, see "How to Create Coverage by Criteria" on next page.

Coverage Calculations

When defining coverage for criteria, consider the following:

- Coverage is calculated only for each instance of the selected criteria.

  Example

  If a business process test contains three components, **Component1**, **Component2**, and **Component3**, and **Component2** is the only criterion selected for coverage, each instance of **Component1** and **Component3** has no affect on coverage, regardless of whether the instances pass or fail. When all the instances of **Component2** run, the requirement for the entire configuration is considered covered.

- Coverage for flows is calculated as follows:
  - Coverage for flows that are run from within a business process test is calculated for the flow in its entirety, and not according to the individual criteria within the flow.
  - Coverage for flows that are run independently of a business process test is calculated according to the individual criteria within the flow.
- When working with more than one configuration and multiple iterations, coverage for a criterion is calculated for each iteration of each configuration. If one of the criterion for an iteration of a configuration fails, the requirement coverage for the associated test fails.

Example

Business process test **OrderFlights** has two business components, **CheckFlights** and **ReserveFlights**. Only **ReserveFlights** is set as a criterion for covering the test requirements, perhaps because it is assumed that if the user was able to reserve a flight, the user must have been able to check the flight. The test has two different configurations, **Domestic** and **International**, representing different types of flights. Each configuration iterates three times, testing the reservation of three domestic flights and three international flights.

Using this example, only the following component instances are examined when calculating coverage:

- **Domestic configuration, Iteration 1, ReserveFlights**
- **Domestic configuration, Iteration 2, ReserveFlights**
- **Domestic configuration, Iteration 3, ReserveFlights**
- **International configuration, Iteration 1, ReserveFlights**
- **International configuration, Iteration 2, ReserveFlights**
- **International configuration, Iteration 3, ReserveFlights**

If just one of these six component instances fails, the associated tests fails when requirement coverage is calculated.

How to Create Coverage by Criteria

This task describes how to create requirement coverage for flows and business components in addition to test configurations.

Note:

- This task is part of a higher-level task. For details, see "How to Plan Business Process Tests and Flows" on page 130.
- To learn more about Requirement Coverage, see "Requirement Coverage Overview " on previous page.
- When defining the criteria that ALM should use to calculate coverage for a test configuration, keep in mind that the same criteria are used for all configurations of the test. It is not possible to define different criteria for different configurations for the same test.

This task includes the following steps:

- "Prerequisites" below
- "Create Coverage" below
- "Modify the Criteria for Requirement Coverage - optional" on next page
- "Analyze Coverage" on next page

1. ## Prerequisites
   - Make sure that requirements are defined in the project.
   - Make sure that the test has defined parameters.
2. ## Create Coverage

   You can create coverage from either the Test Plan module, or from the Requirements module.
   - **From the Test Plan module.** In the Test Plan module, select the **Test Plan Tree** view. Select the business process test or flow, and click the **Req Coverage** tab. Click the **Select Req** button to display the requirements tree in the right pane. Select the requirements to add

and click the **Add to Coverage** button. For user interface details, see the *HP Application Lifecycle Management User Guide*.

You can view the criteria you defined in the Criteria tab. For user interface details, see *HP Application Lifecycle Management User Guide*.

- **From the Requirements module.** In the Requirements module, select the **Requirement Details** view. Select a requirement and click the **Test Coverage** tab. Click the **Select** button to display the test plan tree in the right pane. Select the flows or tests to add and click the **Add to Coverage** button.

  ALM does not create criteria for a flow's business components when selecting a flow from the test plan tree. To add business components as criteria, add coverage by selecting the corresponding business process test from the test plan tree.

For user interface details, see the *HP Application Lifecycle Management User Guide*.

3. Modify the Criteria for Requirement Coverage - optional

You can modify requirement coverage by criteria in the Criterion Settings tab of the Req Coverage tab. For user interface details, see the *HP Application Lifecycle Management User Guide*.

4. Analyze Coverage

Setting criteria for coverage enables you to analyze requirement coverage on a detailed level, such as by business component and flow, and not only on a general level, such as by test. The following analysis methods are available:

Analysis method	Description	Access
Coverage Analysis View	When working with Business Process Testing, this view enables you to examine the status of requirements according to criteria coverage.	For details, see the *HP Application Lifecycle Management User Guide*.
Criterion Results Tab	Displays results of the last test criteria run for the selected business process test.	Perform one of the following: ▪ In the Test Lab module > Test Sets tab, select a test set and click the **Execution Grid** tab. Then select a business process test. The tab is displayed in the lower pane. ▪ In the Test Lab module > Test Runs tab, select a business process test run. The tab is displayed in the lower pane. ▪ In the Test Lab module > Run Details dialog box sidebar select **Criterion Results**.

Analysis method	Description	Access
Criteria Status dialog box	Displays the status of each criterion for the last test criteria run used for the coverage of the selected requirement.	In the Test Configuration Status tab, in the Status column, click the hypertext link for the business process test. For user interface details, see the *HP Application Lifecycle Management User Guide* **Note:** If there is no hypertext link to click, it means no components or flows have been added to the test, and therefore there are no criteria to check. Add components to the test in the Test Plan module and refresh the display in the Requirements module.
Create Test Criteria Reports	You can create a template-based report for tests that includes test criteria coverage.	For task details, see the *HP Application Lifecycle Management User Guide*.

Chapter 11

Run Conditions

This chapter includes:

Run Conditions Overview 194

How to Set Run Conditions 194

Run Conditions User Interface 196

Run Conditions Overview

You can use run conditions to insert condition statements into your flows. A **run condition** checks the current value of a component parameter before running the component in a flow. Based on the parameter value and the run condition definition, HP Application Lifecycle Management (ALM) determines whether to:

- Run the component
- Skip to the next component
- End the component run and set the component status to fail

When you run business process tests containing flows with run conditions, the test run results display the results of run conditions in the test, and lists the components that did not run because a run condition was not met. If a run condition is not met, the test results also provide details about the condition that was not met to help you understand why the component run failed or did not run. As with regular business process tests, you can view test results in the Actual section of the Last Run Report tab in the Test Lab module.

Note: If you set run conditions, and later add or remove a component or change the component order within a flow, the parameters may no longer be relevant and the run condition may not work. For example, if `Component B` uses an output parameter value from `Component A`, and you change the order of the components so that `Component B` precedes `Component A`, then `Component B` cannot receive the output parameter value from `Component A` and the invalid run condition is ignored.

How to Set Run Conditions

The following steps describe how to set run conditions.

Note:

- This task is part of a higher-level task. For details, see "How to Plan Business Process Tests and Flows" on page 130.
- For a use-case scenario related to this task, see "How to Set Run Conditions - Use-Case Scenario" on next page.

This task includes the following steps:

- "Prerequisites" below
- "Add run conditions" below
- "Test run conditions" on next page

1. **Prerequisites**

   Verify that a component step uses at least one flow parameter or component parameter.

2. **Add run conditions**

In the Run Conditions dialog box, specify the criteria for the component to run and click **OK**.

The Run Condition dialog box closes and the run condition is added to the flow. The condition is displayed in the **Run Condition** column of the Script pane in the Test Script tab (for both grid and canvas views).

For user interface details, see "Run Conditions Dialog Box" on next page.

3. ### Test run conditions

Run the test set/flow and verify that the components ran according to the defined run conditions.

For task details on running tests and flows manually and automatically, see "How to Run Business Process Tests and Flows Manually" on page 304 and "How to Run Automated Business Process Tests and Flows" on page 305 respectively.

Note: If a run condition is not valid, the run condition link is displayed in red. This can happen, for example, if a reference parameter was deleted, a parameter value was encrypted, and so on. Delete the run condition and define a new one.

How to Set Run Conditions - Use-Case Scenario

This use-case scenario describes how you can create run conditions.

Note: For a task related to this scenario, see "How to Set Run Conditions" on previous page.

This scenario includes the following steps:

- "Background" below
- "Add run condition" below
- "Test run condition" on next page

1. ### Background

Suppose you work for a bank. Create a flow called **ProcessLoans**, which contains all the transactions necessary for the business process of creating and approving loans: **CreateLoan**, **SearchLoan**, and **ApproveLoan**.

In the **ProcessLoans** flow, depending on the loan being processed, some screens may or may not open. For example, if the requested amount of the loan is above a certain amount, such as $500,000, an additional screen opens for approval from a higher-level bank manager. When requesting a loan for less than $500,000, the extra approval screen does not open.

2. ### Add run condition

a. In the **CreateLoan** component that enters the loan amount, promote the parameter containing the amount, **Amount**, to test parameter.

b. Create a run condition for the **CreateLoan** component that checks the status of the parameter. If the **Amount** value is greater than $500,000, the flow runs the component for the extra approval screen. Otherwise, it skips to the next component and continues.

3. **Test run condition**

When running the test, assign the parameter, **Amount**, values above and below $500,000, and check that the relevant screens open.

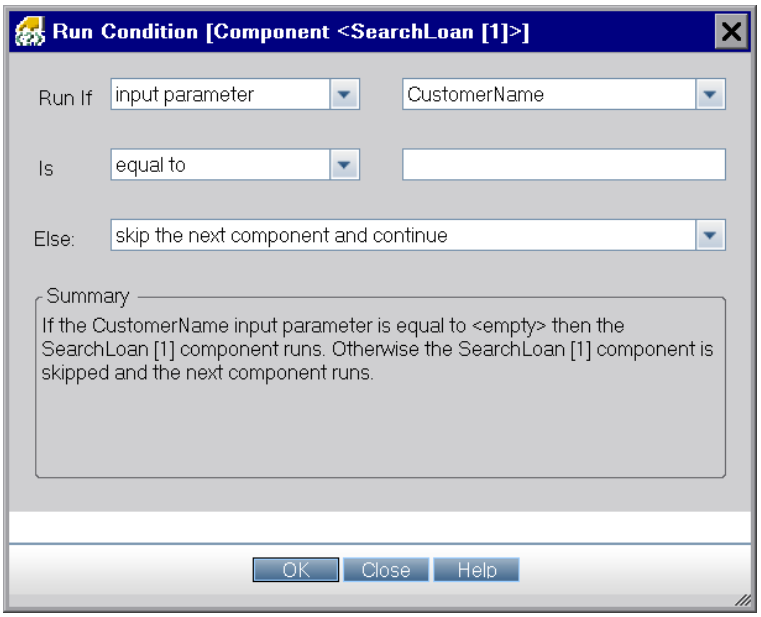
Run Conditions User Interface

This section includes:

Run Conditions Dialog Box 196

Run Conditions Dialog Box

This dialog box enables you to define which attributes must match for the component to run.



To access	In the Test Plan module, select a flow and select the **Test Script** tab. Select the component for which you want to add the run condition and click **Add/Edit Run Condition** .
Important information	• You can only define one run condition per business component. • If defining a run condition for a component, with an output parameter, that has more than one iteration, ALM uses the value from the corresponding iteration. **Tip:** In some cases, it may be preferable to define different flows, rather than use a large number of run conditions within a particular flow.

Relevant tasks	"How to Learn Flows" on page 170
See also	"Criteria for Determining if a Component Can Be Reused" on page 175

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

UI Elements	Description
Run If	Lists the parameter types defined in that component. The following types are available: • **Input Parameter.** Available only if one or more input parameters are defined for this component. • **Output Parameter.** Available only if one or more output parameters are defined for one or more of the previous components in the flow. • **Flow Parameter.** Available only if one or more flow input parameters are defined. **Note:** When creating a run condition on a parameter that contains a dynamic date value, define the run condition as a static date (for example, 12/10/2011), which is compared to the actual date used in the run.
<parameter name>	Lists the available parameter. Encrypted parameters are not listed. Parameters with encrypted values cannot be chosen from the list when defining run conditions. It is recommended that you do not use parameters whose default values are encrypted when defining run conditions.
Is	Lists the operators that need to be met for the component to run. The following conditions are available: • **equal to** • **not equal to** • **less than** • **less than or equal** • **greater than** • **greater than or equal**
<value>	Enables you to enter the valid value for the condition.

UI Elements	Description
Else	Specifies what to do if the condition is not met. The following options are available: • **Skip to next component and continue.** If the condition is not met, the component for which the run condition is set does not run, and the test run continues with the next component in the flow. ▪ When running with the Manual Runner, the test results list the run status for the component with the run condition as **Ignored**. ▪ When running with the Automatic Runner, the test results list the run status for the component with the run condition as **Not Run**. The component is not displayed in the run results. • **End component run and fail.** If the condition is not met, the component for which the run condition is set does not run, but instead sets the status of the component run as **Failed**. The flow either continues to the next component or ends, depending on the failure condition set for the component. **Note:** If a run condition is not valid, the run condition link is displayed in red. This can happen, for example, if a reference parameter was deleted, a parameter value was encrypted, and so on. Delete the run condition and define a new one. **Tip:** Your selection in the **Else** box applies only if the run condition is not met. To specify whether to continue or end the entire run if a component run fails, set the failure condition for the component. You do this in the Test Script tab of the Test Plan module. For user interface details, see "Test Plan Module Menu and Buttons for Business Process Testing" on page 147.
Summary	Displays a text description of the run condition you have defined for the component.

Chapter 12

Business Process Test Configurations

This chapter includes:

Business Process Test Configurations Overview 200

How to Work with Business Process Test Configurations 203

How to Work with Configurations and Static Data 204

How to Work with Configurations and Dynamic Data 205

How to Generate Test Configurations from Test Instances 209

Business Process Test Configurations User Interface 211

Business Process Test Configurations Overview

This chapter describes how to expand the flexibility and reusability of business components, flows, and business process tests by creating test configurations that represent different use-case scenarios, and having each configuration access different sets of data. Associating test configurations with requirements provide finer granularity for requirement coverage by enabling coverage by different use-cases of a test.

For example, you may want to check the same test in a Windows environment and again in a UNIX environment. In this case, you can define one business process test with two test configurations. The parameters values for the Windows configuration would contain Windows-specific values (for example, backslashes in a path name) while the UNIX configuration would contain UNIX-specific values (for example, forward slashes in a path name).

A test configuration can be associated with multiple iterations, based on the number of sets of data supplied for that test configuration.

When using test configurations with business process tests, you can run a test many times in succession with different values specified in a Microsoft Excel file.

When creating a business process test, by default, a test configuration is created by ALM after a test parameter is created. This test configuration is created with the same name as the test. This test configuration can be renamed.

Note: You cannot define configurations for flows.

This chapter includes the following topics:

- "Test Configurations Associated with Static Data" below
- "Test Configurations Associated with Dynamic Data" on next page
- "Configuration Example with Iterations" on page 202

Test Configurations Associated with Static Data

The basic ALM test configuration accesses static data. Static data are sets of values provided for each parameter directly from within ALM. For task details on creating configurations for ALM manual tests, see the *HP Application Lifecycle Management User Guide*.

When creating test configurations for manual and automated business process tests, you also can create a set of static data for each use-case. For each test configuration, you can also add iterations and define different parameter values for each iteration. For task details on creating test configurations that access static data for business process tests, see "How to Work with Configurations and Static Data" on page 204

Test Configurations Associated with Dynamic Data

Business process test configurations can access dynamic data. Dynamic data are sets of values that are provided for each iteration's parameters from outside of ALM, in an external data table uploaded in the Test Resources module. The external data table is a Microsoft Excel file.

Note: To work with external data tables for test configurations, install the HP Unified Functional Testing Add-in for Business Process Testing on the client computer. The HP Unified Functional Testing Add-in for Business Process Testing is installed from the HP Application Lifecycle Management Add-ins page.

When the business process test instance runs using a configuration that accesses dynamic data, the test instance fetches the parameter values for each iteration of the configuration from the list of parameter values in the Microsoft Excel file.

You can supply ALM with dynamic data resources:

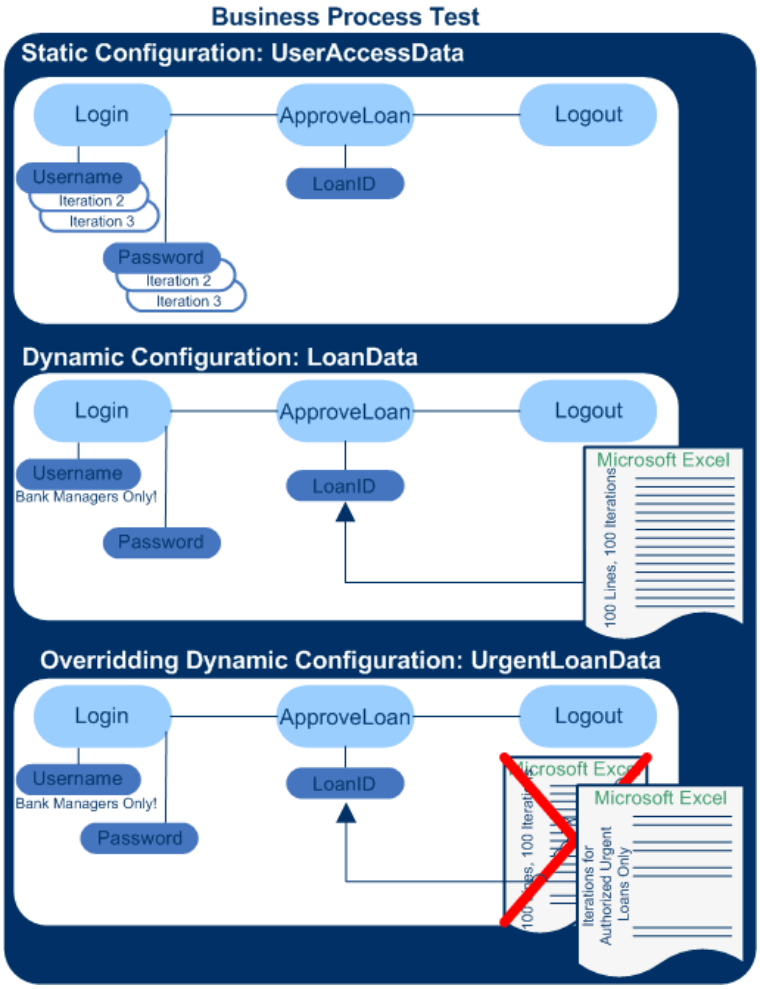
- **At the test level.** You can supply a Microsoft Excel file that contains all parameter values for each iteration of the configuration is specified at the test level. The Microsoft Excel file is uploaded as a data table in the Test Resources module, and associated with the business process test configuration in the Parameters tab of the Test Plan module.
- **At the configuration level (overriding any test-level data resources).** You can supply an alternate Microsoft Excel file, overriding the test-level data resources file that is associated with the business process test as a whole. The Microsoft Excel file is uploaded as a data table in the Test Resources module, and associated with a specific configuration of the business process test in the Test Configurations tab of the Test Plan module.

Note: Configurations that access dynamic data are not available for use with ALM manual test configurations.

For task details, see "How to Work with Configurations and Dynamic Data" on page 205

Configuration Example with Iterations

Consider the following business process test and its three configurations for a banking application comprising three business components: Login, ApproveLoan, and Logout:



- The entire business process test tests the following aspects of the application:
  - Access rights are correct, based on the identity of the user.
  - Load is handled well, such as when a bank manager approves many loans using an external list of data created in Microsoft Excel.
  - A bank manager can approve urgent, high priority loans, based on a different external list created in Microsoft Excel. This external list is to be further narrowed down to include only those loans that have manager authorization.
- To test access rights, a configuration named UserAccessData accesses static data, which provides different values for the test parameters Username and Password for three test iterations: One iteration provides the login information for an average loan processor, another for a bank manager, and the third for a system administrator for the application. Each of these users has different permissions and access to the application. The parameter values are specified within ALM.

- A configuration named LoanData accesses dynamic data for one hundred different values for the LoanID input parameters used by the ApproveLoan component. These values are provided in a Microsoft Excel test resource of the data table type. A test iteration is run for each value in the Microsoft Excel file, so the test instance that uses this configuration runs one hundred times.

  Note: Configurations can supply data values for test iterations, but not for component iterations.

- To test urgent loans, a configuration-level data resource can be defined to override the test-level data resource for the business process test. This data resource, UrgentLoanData, only contains high priority loans that are to be approved. One test iteration is run for each value that matches the filter criteria in the alternative configuration-level Microsoft Excel file (in this case, urgent loans that have been authorized for approval).

How to Work with Business Process Test Configurations

This task describes how to work with business process test configurations and other test types whose configurations can access external, dynamic data (such as UFT GUI tests).

Note:

- This task is part of a higher-level task. For details, see "How to Plan Business Process Tests and Flows" on page 130.

This task includes the following steps:

- "Prerequisites" below
- "Design a test configuration that accesses static data" below
- "Design a test configuration that accesses dynamic data" on next page
- "Create coverage" on next page
- "Define and run test configurations" on next page
- "View test results" on next page
- "Fine-tune the results and generate a new configuration - optional" on next page

1. ### Prerequisites

   In the Parameters tab for the business process test, create test-level parameters that represent the values for the first iteration of the configuration.

   For task details on creating test-level parameters, see "How to Create Parameters" on page 238.

2. ### Design a test configuration that accesses static data

   Defining static data enables you to specify sets of parameter values for the test directly from

within ALM, including the number of time the test configuration is run (iterated).

For task details, see "How to Work with Configurations and Static Data" below.

For task details on iterating test configurations, see "How to Set Data for Iterations" on page 287.

3. **Design a test configuration that accesses dynamic data**

   Defining dynamic data enables you to specify sets of parameter values for the test from an external Microsoft Excel file. The test configuration is run (iterated) for each selected row in the external file.

   Tip: The same Microsoft Excel file can be used as a data resource for different tests.

   For task details, see "How to Work with Configurations and Dynamic Data" on next page.

4. **Create coverage**

   Link each test configuration in the test plan tree with one or more requirements in the requirements tree. By defining requirements coverage for a test configuration, you keep track of the relationship between the test configurations in your test plan and your original requirements. For task details, see "How to Create Coverage by Criteria" on page 189.

5. **Define and run test configurations**

   Run the test to confirm the test runs according to the expected configurations, iterations, and parameter values.

   For task details on creating test sets, see the *HP Application Lifecycle Management User Guide*.

   For task details on running manual test sets, see "How to Run Business Process Tests and Flows Manually" on page 304.

   For task details on running automated test sets, see "How to Run Automated Business Process Tests and Flows" on page 305.

6. **View test results**

   For task details on viewing test results, see the *HP Application Lifecycle Management User Guide*.

7. **Fine-tune the results and generate a new configuration - optional**

   For task details on generating test configurations, see "How to Generate Test Configurations from Test Instances" on page 209.

How to Work with Configurations and Static Data

This task describes how to work with static test configurations in Business Process Testing.

Defining static data enables you to specify sets of parameter values for the test directly from within ALM.

You can create configurations that access static data from the Test Plan module, and modify the data from the Test Plan and Test Lab modules.

Note:

- This task is part of a higher-level task. For details, see "How to Work with Business Process Test Configurations" on page 203.

This task includes the following steps:

- "View the test configurations for the business process test" below
- "Add test configurations " below
- "Select a static configuration" below
- "Create iterations and modify parameter values" below

1. ### View the test configurations for the business process test

   In the Test Plan tree, select the business process test for which you want to create a test configuration that accesses static data.

   Click the **Test Configurations** tab. A test configuration is listed in the configuration grid with the same name as the test.

   For user interface details on the Test Configurations tab, see "Test Configurations Tab for Business Process Testing" on page 211.

2. ### Add test configurations

   For user interface details, see *HP Application Lifecycle Management User Guide*.

3. ### Select a static configuration

   In the Test Plan module > Test Configurations tab, click the **Data** tab and then select **Static**.

   For user interface details on the Data tab for static configurations, see "Data Tab: Static View" on page 212.

4. ### Create iterations and modify parameter values

   Existing iterations and the test parameter default values are displayed in a grid. Add /remove iterations, and modify parameter values, as necessary.

   For user interface details on iterations, see "Iteration Pages" on page 292.

How to Work with Configurations and Dynamic Data

This task describes how to work with test configurations in Business Process Testing that access dynamic data.

Defining dynamic data enables you to specify sets of parameter values for the test from an external Microsoft Excel file.

Tip: The same Microsoft Excel file can be used as a data resource for different tests.

Note:

- This task is part of a higher-level task. For details, see "How to Work with Business Process Test Configurations" on page 203.

This task includes the following steps:

- "Prerequisites" below
- "Create a data table in Microsoft Excel" below
- "Create ALM test resources" on next page
- "Associate the data table test resource with the business process test" on page 208
- "Map test parameters to resource parameters " on page 208
- "Filter the data by selecting iteration ranges - optional" on page 209
- "Associate an alternate data table with the test configuration - optional" on page 209

1. ## Prerequisites

   To work with dynamic data for automated tests, make sure that the Unified Functional Testing Add-in for Business Process Testing is installed on the client computer. The Unified Functional Testing Add-in for Business Process Testing is installed from the HP Application Lifecycle Management Add-ins page.

2. ## Create a data table in Microsoft Excel

   This data table contains rows of values which become parameter values for each iteration of your test configuration.

   a. In Microsoft Excel, create a spreadsheet.

      Note: Microsoft Excel version 2003 or lower is supported.

   b. Select the **Sheet1** sheet. Do not define data in the other sheets. ALM ignores data entered in the other sheets.

      UFT: Select the **Global** sheet instead of Sheet1 when working with UFT.

   c. Specify parameter names in the first row. The parameter names do not have to match the parameter names defined in the test.

   d. Specify values for each parameter under the parameter names, each on a different row. Each row represents an iteration.

   e. Save the file.

Example

	A	B	C
1	Name	Phone	Amount
2	Orville	Bramson	10000
3	Bernadette	Washington	4500
4	Kim	Wang	5000
5	Tom	Smith	43000
6	Barbara	Wilson	15000

3. Create ALM test resources

In ALM, upload the Microsoft Excel data table as a test resource. For task details, see the *HP Application Lifecycle Management User Guide*.

a. Select the **Test Resources** module.

b. Either create or select a test resource folder.

c. Click **New Resource** to create a new test resource. The New Resource dialog box opens.

d. Enter **Data Table** as the resource type and click **OK**.

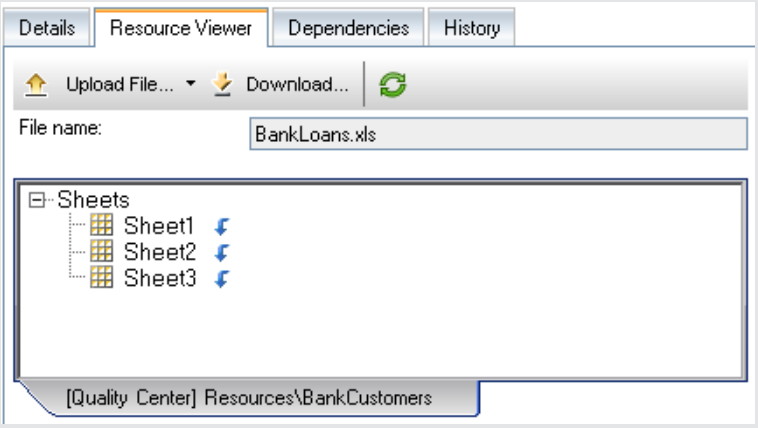
e. Click the **Resource Viewer** tab.

f. Click the **Upload File** button.

g. Navigate and select the Microsoft Excel file you created and click **Open**.

h. Click **OK** when prompted that the upload completed successfully. The Resource Viewer lists the default Microsoft Excel sheets. The data is now available in ALM.

Example



You can double-click **Sheet1** to view the data you created.

Note: The data in the Microsoft Excel file starts with row number 2. It is assumed that the first row in the Microsoft Excel file contains the names for the data columns, which are mapped to parameter names in ALM.

4. **Associate the data table test resource with the business process test**

   a. In the Test Plan module Parameters tab for a business process test, click the down arrow next to the **Data Resource** field in the toolbar.

   b. Navigate to, and select, the data resource and click **OK**. The Microsoft Excel file is now the default external data file for the test.

5. **Map test parameters to resource parameters**

   a. In the Parameters tab for the test, click **Map Parameters** . The Map Parameters dialog box is displayed. For user interface details on the Map Parameters dialog box, see "Map Parameters Dialog Box" on page 215.

   b. Do one of the following:

   - For each test parameter, click in the cell for its corresponding resource parameter. Select the corresponding resource parameter and click **OK**.
   - Click **Automap** . ALM maps each test parameter to the corresponding resource parameter based on the column headings in the first row of the data table.

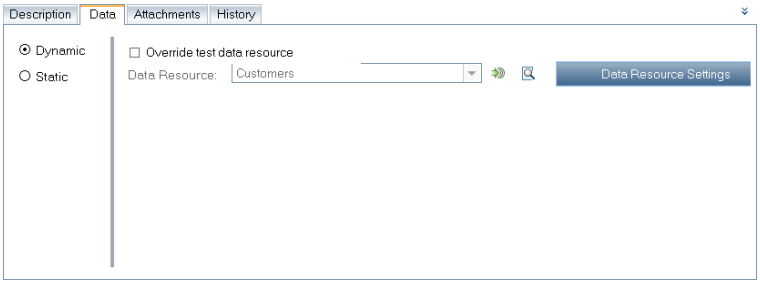
Tip: The mapping of resource parameters to test parameters facilitates reuse of the same resource file by different tests.

6. **Create the test configuration**

   a. In the Test Plan tree, select the business process test for which you want to create a test configuration that accesses dynamic data. Click the **Test Configurations** tab.

   b. Click **New Configuration** . The New Test Configuration dialog box opens. For user interface details on the New Test Configuration dialog box, see *HP Application Lifecycle Management User Guide*.

   c. Click the **Data** tab near the bottom of the Test Configurations tab, and select **Dynamic**. This instructs ALM to use the Microsoft Excel test resource file defined for the test in the Parameters tab.



For user interface details on the Data tab for dynamic configurations, see "Data Tab: Dynamic View" on page 213.

7. **Filter the data by selecting iteration ranges - optional**

   To filter the data so that the test runs only on a subset of the parameter values, either by row or by value, click **Data Resource Settings**. The Data Resource Settings dialog box opens. For user interface details on the Data Resource Settings dialog box, see "Data Resource Settings Dialog Box" on page 217.

8. **Associate an alternate data table with the test configuration - optional**

   You can create configurations that process alternate dynamic data resource files instead of the dynamic data resource file associated with the business process test.

   Tip: The same file can be used as a data resource for different tests.

   a. In the **Data** tab, select **Dynamic**.

   b. Select **Override test data resource** and specify an alternate data resource.

   c. Click **Data Resource Settings** to map the test parameter names to the resource parameter names in the Data Resource Settings dialog box. For user interface details on the Data Resource Settings dialog box, see "Data Resource Settings Dialog Box" on page 217.

How to Generate Test Configurations from Test Instances

This task describes how to generate a new test configuration from an existing static test configuration instance whose configuration properties have been modified to no longer match its original static configuration.

- This task is part of a higher-level task. For details, see "How to Work with Business Process Test Configurations" on page 203.
- For user interface details on the Generate Test Configuration dialog box, see the *HP Application Lifecycle Management User Guide*.

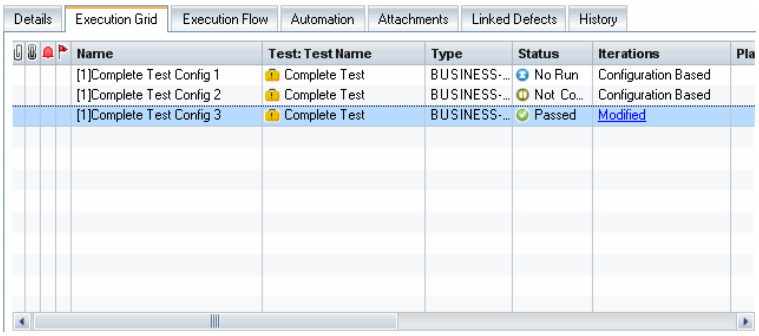
This task includes the following steps:

- "Locate the test instance" below
- "Fine-tune the iterations for the test configuration" on next page
- "Generate the new test configuration" on next page
- "Results" on page 211

1. **Locate the test instance**

   a. In the Test Lab module > Test Sets tab, right-click the static test configuration instance

that was modified in the Execution Grid tab, and select **Test Instance Details**.



Note: If a test instance has not been modified, **Configuration Based** appears in the Iterations column for that instance.

b. Select the **Execution Settings** sidebar entry.

c. Click the **Test Iterations** tab.

2. ## Fine-tune the iterations for the test configuration

Add and delete iterations, defining values for each test parameter in each iteration. For user interface details on iterations, see "Iteration Pages" on page 292.

3. ## Generate the new test configuration

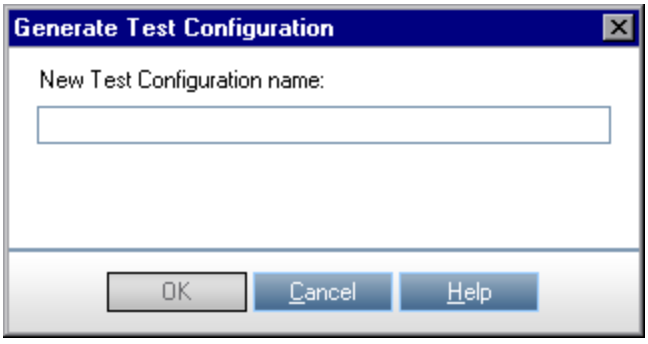
a. Close the Iteration page, and right-click the test instance.

b. Right-click and select **Generate Test Configuration**.

The Generate Test Configuration option is disabled if:

- The test set is pinned to a baseline. You cannot generate test configurations for pinned test sets.
- **Version Control**: If the test is checked out by another user, wait until the test is checked in before generating the test configuration.

The Generate Test Configuration dialog box opens.



c. Enter a name for the new test configuration and click **OK**.

Tip: Instead of generating a new test configuration, you can also restore test instance

values so the test instance values match the static data values defined for the test configuration on which the test instance is based. For user interface details, see the **Restore Instances** button in the "Iteration Pages" on page 292.

4. **Results**

   The test instance name in the Execution Grid is renamed to reflect the name of the new configuration with which this instance is now associated.

   In the Test Plan module, a new static test configuration is added for the corresponding text in the Test Configurations tab.

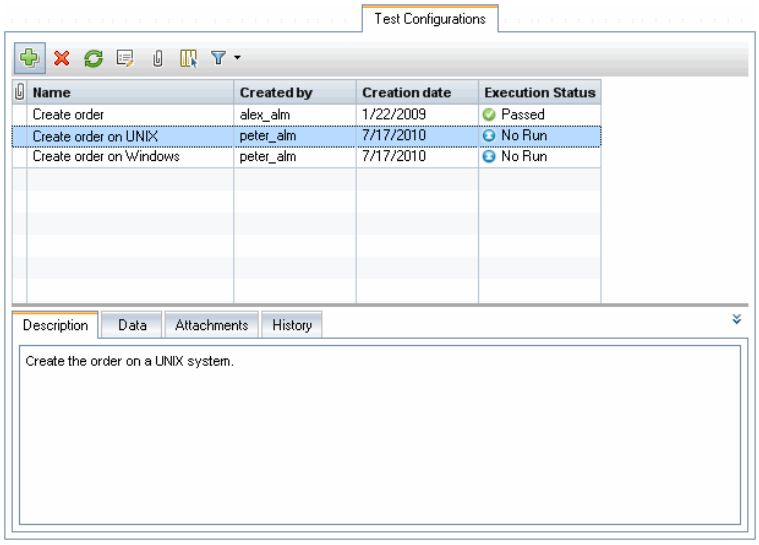
Business Process Test Configurations User Interface

This section includes:

Test Configurations Tab for Business Process Testing 211
Data Tab: Static View 212
Data Tab: Dynamic View 213
Map Parameters Dialog Box 215
Data Resource Settings Dialog Box 217

Test Configurations Tab for Business Process Testing

This version of the Test Configurations tab enables you to define and update test configurations for business process tests.



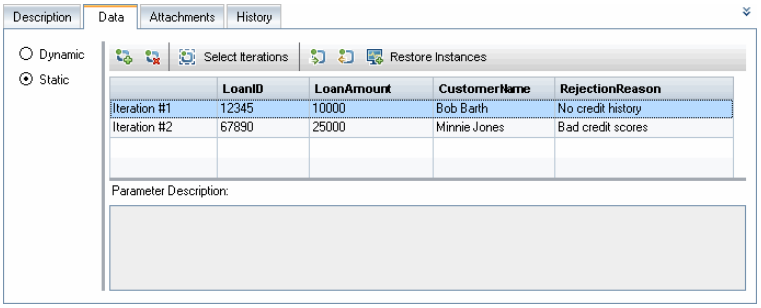
To access	From the Test Plan module, select a business process test and click the **Test Configurations** tab.
Important information	The Test Configurations tab for Business Process Testing differs from the Test Configurations tab for other ALM test types. This section describes the elements that are significant specifically when designing business process test configurations. For user interface details on the Test Configuration tab for other test types, see the *HP Application Lifecycle Management User Guide*.
Relevant tasks	• "How to Work with Business Process Test Configurations" on page 203 • *HP Application Lifecycle Management User Guide*.
See also	"How to Create Parameters" on page 238

User interface elements unique to Business Process Testing are described below:

UI Element	Description
Data tab	Displays the data associated with this test configuration. For user interface details on static data, see "Data Tab: Static View" below. For user interface details on dynamic data, see "Data Tab: Dynamic View" on next page.

Data Tab: Static View

This tab enables you to create test configurations for business process tests that access static data (meaning, data defined within ALM).



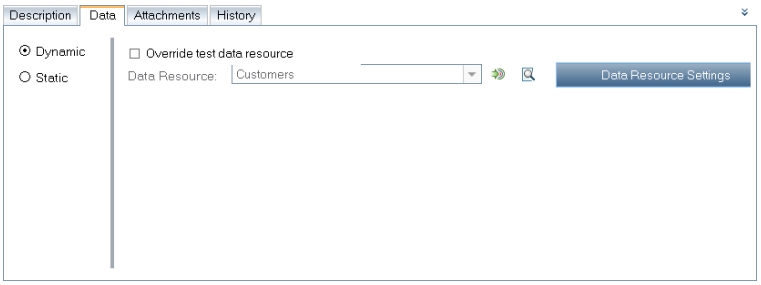
To access	In the Test Plan module > Test Configurations tab, click the **Data** tab and then click **Static**.
Important information	To change the values of a parameter, click in its cell. Type a value directly in the grid, or click the down arrow to open the Set Values page. For task details on setting values, see "How to Set Parameter Values" on page 239.
Relevant tasks	• "How to Work with Business Process Test Configurations" on page 203 • "How to Work with Configurations and Static Data" on page 204
See also	"Business Process Test Configurations Overview" on page 200

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

UI Element	Description
Static radio button	Changes the test configuration so it accesses static data defined within ALM.
Dynamic radio button	Changes the test configuration so that it access an external Microsoft Excel data resource file. For user interface details, see "Data Tab: Dynamic View" below.
<Iteration page> common elements	Enables you to add iterations, delete iterations, modify each iteration's parameter values, and select iteration ranges to run in a business process test. Also enables you to import and export parameter values. For user interface details, see "Iteration Pages" on page 292.

Data Tab: Dynamic View

This tab enables you to modify test configurations for business process tests so that the test configurations access dynamic data (data that resides outside of ALM in an external data resource).



To access	Do one of the following: • In the Test Plan module > Test Configurations tab, click the **Data** tab and select **Dynamic**. • In the Test Lab module > Test Instance Details dialog box > Execution Settings view for business process test configurations and UFT test configurations that access dynamic data, click the **Data** tab.
Important information	• By default, the data resource used is the one specified in the Parameters tab for the business process test. This can be overridden by specifying a different data resource in this Data tab. • When accessed from the Test Instance Details dialog box Execution Settings View, the information in this tab is read-only.
Relevant tasks	"How to Work with Business Process Test Configurations" on page 203 "How to Work with Configurations and Dynamic Data" on page 205
See also	"Business Process Test Configurations Overview" on page 200 *HP Application Lifecycle Management User Guide*

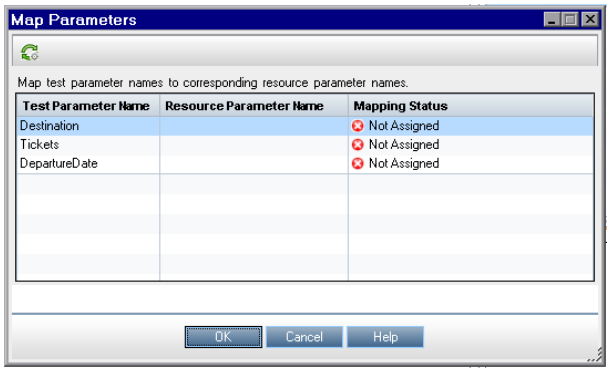
User interface elements are described below:

UI Element	Description
Static radio button	Changes the test configuration so it accesses static data defined within ALM.
Dynamic radio button	Changes the test configuration so that it access an external Microsoft Excel data resource file. For user interface details, see "Data Tab: Dynamic View" on previous page.
Override test data resource checkbox	Indicates that you want to override the dynamic data resource specified at the test level with a data resource at the configuration level. **Available from:** Test Configurations tab
Data Resource	Enables you to enter the name of a data resource that will override the data resource specified at the test level. **Available from:** Test Configurations tab when **Override test data resource** is selected
	Go to data resource. Opens the data resource in the Test Resources module. **Available when:** An alternate data resource is specified in the **Data Resource** field.
	Resource Preview. Opens the asset viewer's Test Configuration Resource Preview for the test resource. This enables you to see the data defined in the external data resource (meaning, all data available to the test configuration). **Available when:** An alternate data resource is specified in the **Data Resource** field.

UI Element	Description
	Filtered Resource Preview. Opens the Data Viewer for the test resource. This enables you to see the dynamic data in the external data resource that match filter criteria (meaning, the data that will be used when the test configuration runs). **Filter.** Name of the filter. **Row Filter:** The rows as defined in Data Resource Settings dialog box (all rows or selected rows). For user interface details, see "Data Resource Settings Dialog Box" on page 217. **Show mapped columns only**. If selected, only data that is mapped to test parameters are displayed. **<headings>**. Column headings for the grid that also display the parameter mappings. **<grid>**. Rows of data that match the filter criteria. **Available from:** Test Configurations tab when an alternate data resource is specified in the **Data Resource** field.
Data Resource Settings	Opens the data resource settings dialog box, which enables you to set iteration ranges when running the test configuration. For user interface details, see "Data Resource Settings Dialog Box" on page 217. **Available when:** An alternate data resource is specified in the **Data Resource** field.

Map Parameters Dialog Box

This dialog box enables you to associate data from an external data resource to the parameters for the test configuration in the business process test.



To access	In the Test Plan module > Parameters tab for the business process test, click **Map Parameters** .

Important information	You can map test parameters to external data manually and automatically.
Relevant tasks	"How to Work with Configurations and Dynamic Data" on page 205
See also	"Business Process Test Configurations Overview" on page 200

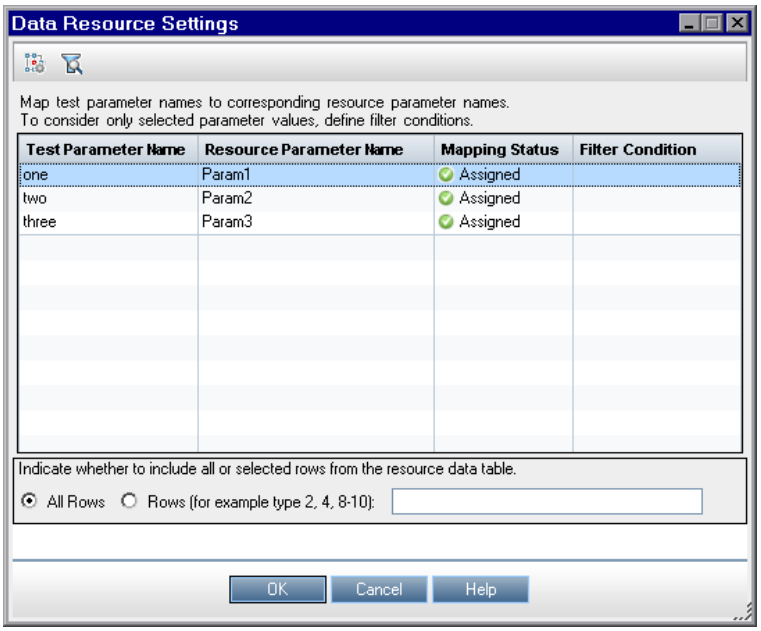
User interface elements are described below:

UI Element (A-Z)	Description
	Automatically maps each non-mapped test parameter to a parameter (column heading with the same name) in the external data table resource. For automatic mapping, the column heading in the external data table resource must match the test parameter name. The mapping is not case-sensitive. The order of the parameters in the external data table resource does not have to match the order of the parameters in the test. **Note:** If a parameter has already been mapped, the automap operation does not reset its mapping.
Test Parameter Name	Name of the test parameter as defined in the test.
Resource Parameter Name	Name of the column heading as defined in the external data table resource. **Tip:** Select the column headings from a drop-down list. To clear the resource parameter name, select the blank row in the drop-down list.
Mapping Status	Status indicating if the test parameter is mapped to a parameter (column) in the external data table resource. • **Not Assigned.** The test parameter is not associated with an external data table resource parameter. • **Assigned.** The test parameter is associated with an external data table resource parameter.

Data Resource Settings Dialog Box

This dialog box enables you to:

- Define a subset of the parameter values (an iteration range) to use for running the test when working with configurations that access dynamic data.
- Map test parameter names to alternate data resource parameter names when overriding the default data resource associated with the automated business process test.



To access	From the Test Plan module's Test Configurations tab, select the dynamic configuration whose data you want to filter. Click the **Data** tab near the bottom of the window, and then click **Data Resource Settings.**
Important information	• You can filter the parameter values by specifying filter conditions and specific rows. Only values that match both the filter condition and the specified rows are processed. • When mapping test parameter names to alternate data resource parameter names, for each parameter listed in the **Test Parameter Name** column, click the cell in the **Resource Parameter Name** column and type the name of the corresponding resource parameter from the Microsoft Excel file.
Relevant tasks	"How to Work with Business Process Test Configurations" on page 203 "How to Work with Configurations and Dynamic Data" on page 205
See also	"Business Process Test Configurations Overview" on page 200

User interface elements are described below:

UI Element (A-Z)	Description
	Automatically maps each non-mapped test parameter to a parameter (column heading with the same name) in the external data table resource. For automatic mapping, the column heading in the external data table resource must match the test parameter name. The mapping is not case-sensitive. The order of the parameters in the external data table resource does not have to match the order of the parameters in the test. **Note:** If a parameter has already been mapped, the automap operation does not reset its mapping.
	Filtered Resource Preview. Opens the Data Viewer for the test resource. This enables you to see the dynamic data in the external data resource that match filter criteria (meaning, the data that will be used when the test configuration runs). **Filter.** Name of the filter. **Row Filter.** All rows or selected rows. **Show mapped columns only.** If selected, only data that is mapped to test parameters are displayed. **<headings>.** Column headings for the grid that also display the parameter mappings. **<grid>.** Rows of data that match the filter criteria.
Test Parameter Name	Name of the test parameter as defined in the test.
Resource Parameter Name	Name of the test parameter as defined in the data resource.
Filter Condition	Condition for filtering out rows of data in the data resource. The test will not run on the filtered out rows of data. Specify a filter according to the following guidelines: • Only strings can be specified. • Do not specify quotation marks or operators. • Only one value can be specified. • The filter is case-sensitive.

UI Element (A-Z)	Description
All Rows	To process all rows that meet the specified filter condition, if one exists, in the data resource file, select **All Rows**.
Rows	To indicate a subset of rows, select **Rows** and enter the row numbers, separated by commas. A range of rows can be specified with a hyphen (for example, 3-5 means rows 3, 4, and 5 are processed).

Chapter 13

Test and Flow Debugging

This chapter includes:

Test and Flow Debugging Overview....222

How to Debug Tests and Flows Containing Automated Components....222

Test and Flow Debugging User Interface....224

Test and Flow Debugging Overview

You can debug a business process test or flow comprised of one or more automated business components by running the components in Debug mode in the Test Plan module.

Debugging is usually performed after the automation engineer has ensured that the individual components can run successfully in the relevant testing tool (such as UFT), and all the components have a **Ready** status.

Debugging ensures that the test runs properly, and locates errors that may have occurred during the building of the test. For example, you can check the logical order of business components in the test, the pre-conditions and post-conditions required by each component, and the suitability of component parameter values (especially where output values are used as input values in other components).

You can choose whether to run each of the business components in the test or flow in **Debug** mode or in **Normal** mode.

- When you run an automated component in Debug mode, the component opens and immediately pauses, enabling you (or an automation engineer) to use the debug tools available in the testing tool to debug the component. When one component ends, the next one immediately opens.
- When you run in Normal mode, the testing tool runs all the steps in the component consecutively, and then opens the next component. You cannot access debug options while a component runs in Normal mode.

When you debug a test or flow, ALM opens the testing tool. The requested test is then run on your local computer.

Note: With the first keyword GUI or scripted component in the test, ALM opens the testing tool and loads the relevant add-ins from the application area that is associated with that component. It assumes that these are the required add-ins for all the components in the test.

When a debug run is complete, ALM opens the Debug Run Information dialog box that shows which of the business components passed or failed the test. These results are provided for debugging purposes only and are not stored in ALM. For user interface details, see "Debug Run Information Dialog Box" on page 226.

For task details, see "How to Debug Tests and Flows Containing Automated Components" below.

Tip: You can also access the testing tool, such as UFT, to run and debug individual business components. For details on running and debugging components in UFT, see the *HP Unified Functional Testing User Guide*.

How to Debug Tests and Flows Containing Automated Components

This task describes how to debug tests and flows to make sure that they run properly.

This task is part of a higher-level task. For details, see "How to Plan Business Process Tests and

Flows" on page 130.

This task includes the following steps:

- "Prerequisites" below
- "Debug automated tests and flows" below
- "Monitor the debug run" below
- "Results" on next page

1. ## Prerequisites

   - Make sure that the flow or business process test that you intend to debug contains only automated components.
   - Ensure that default values are set for all parameters. For task details on setting values, see "How to Set Parameter Values" on page 239.
   - **UFT:** To be able to debug or run flows and business process tests that contain automated keyword GUI business components, or view test results, make sure that one of the following is installed:
     - QuickTest Professional version 10.00 or later
     - UFT version 12.00 or later
   - **UFT:** Close all browsers before running a test on a Web browser. UFT must load the Web Add-in (as defined in the application area in UFT) before a step in the test opens the browser.
   - **UFT:** If the test contains UFT components, the test should include a UFT component at the beginning of the test before the browser opens.

2. ## Debug automated tests and flows

   In the Test Plan module, select the relevant test or flow in the Test Plan tree. Start debugging by clicking in the **Test Script** tab. For user interface details on the Run or Debug test dialog box, see "Run or Debug Test Dialog Box" on next page.

3. ## Monitor the debug run

   While running a test or flow in debug mode, you can monitor the debug run.

   - From your task bar, you can toggle between the Test Plan module, the testing tool (such as UFT), and the application you are testing.
   - In the Test Plan module, the status of the test and the name of the component currently being run are shown below the toolbar in the Test Script tab, for example, `Running(OrderStart)`. You can click the **Stop Run** button to stop the run at any time.
   - In the relevant testing tool, the status of the test being run, for example **Running**, or **Ready**, is shown in the status bar at the bottom of the window.

- **UFT Keyword GUI or GUI Scripted**: All the debugging functions from the **Debug** menu, for example, **Step Over** or **Step Into**, are enabled for use. For details on applicable debug options, see the *HP Unified Functional Testing User Guide*.
- **UFT Keyword GUI or GUI Scripted:** If you selected a business component to run in **Debug** mode, the test pauses after opening the component in UFT. Use the UFT debug options to control the continuation of the run through the displayed component. When you are ready to proceed with the run, continue the run in UFT.
- When one component finishes running, it closes and the next component in the test opens in the testing tool.

4. **Results**

   When the debug run is complete, ALM imports the results from the testing tool and displays summary information in the Debug Run Information dialog box.

   For user interface details, see "Debug Run Information Dialog Box" on page 226

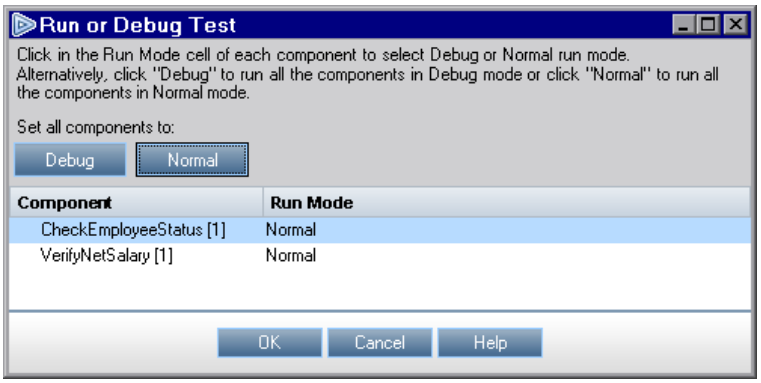
Test and Flow Debugging User Interface

This section includes:

Run or Debug Test Dialog Box 224
Debug Run Information Dialog Box 226

Run or Debug Test Dialog Box

This dialog box displays the automated components that make up the selected business process test or flow, and enables you to choose whether to run each component in Debug mode or in Normal mode.



To access	In Test Plan module, select the relevant test or flow in the Test Plan tree, and click the **Test Script** tab. Click **Run or Debug Test** .

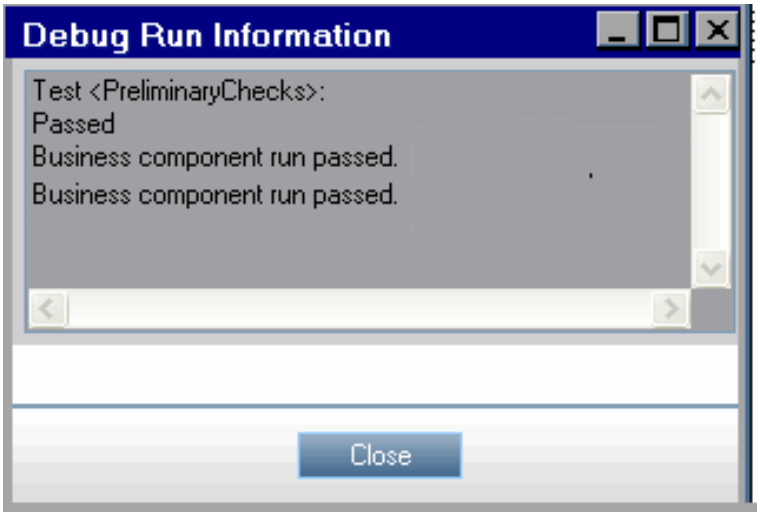
Important information	Tests can contain both automated and manual components. Initiating a test to debug a manual component or test containing one or more manual components displays a warning message. To run tests that contain manual components, use the Manual Runner. For details, see "Manual Runner Wizard for Business Process Testing" on page 310.
Relevant tasks	• "How to Create Business Process Tests and Flows" on page 134 • "How to Run Automated Business Process Tests and Flows" on page 305
See also	"Test and Flow Debugging Overview " on page 222

User interface elements are described below:

UI Element	Description
Debug	Sets all the components to run in Debug mode.
Normal	Sets all the components to run in Normal mode.
Component column	Lists the available business components.
Run Mode column	Indicates the run mode to be used with running a test flow. Clicking the relevant cell in the **Run Mode** column enables you to select one of the following: • **Debug.** Causes the test to pause before running the first step in the specified business component. The position of the breakpoint in the test is indicated by a red spot icon in the left margin of the Keyword View (or, for UFT components, the Expert View in UFT). This enables you to use all the debug options of the testing tool to check the performance of one or more specific steps in the component. You can then continue the run. For details on UFT debugging options, see the *HP Unified Functional Testing User Guide*. • **Normal.** Runs the selected business component from start to finish without pausing.

Debug Run Information Dialog Box

This dialog box enables you to view the results of the debug run for the test or flow overall, and each individual automated component.



To access	In the Test Plan module, select the relevant test or flow in the Test Plan tree, and click the **Test Script** tab. Click **Run or Debug Test** . When the debug run completes, this dialog box opens.
Important information	The results displayed in this dialog box are provided for debugging purposes only and are not stored in ALM.
Relevant tasks	"How to Debug Tests and Flows Containing Automated Components" on page 222
See also	"Run or Debug Test Dialog Box" on page 224

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

UI Element	Description
<Information area>	Displays the following: • The pass/fail status for the test or flow overall, and the pass/fail status for each individual automated component. • All relevant messages from the testing tool.

Data Handling

Chapter 14

Introducing Data Handling (Parameters)

This chapter includes:

Data Handling Overview. 230
How to Handle Data in Business Process Testing. 235

Data Handling Overview

You can affect the behavior and results of a business process test by using parameters to define the values that components and flows receive and return. This process is known as **parameterization.**

Parameterization enables you to perform operations on the application you are testing with multiple sets of data. Each time you run a business process test, you can supply different values for the parameters in the test (or its components and flows).

This chapter provides an introduction to working with data and parameters in Business Process Testing.

For task details, see "How to Handle Data in Business Process Testing" on page 235.

Product Feature Movie: To view a movie that demonstrates how to handle Business Process Testing parameters, select **Help > Product Feature Movies** in the ALM main window.

This section includes:

- "Parameter Categories" below
- "Parameter Linkage" on next page
- "Parameter Promotion" on next page
- "Iterations and Test Configurations" on page 232
- "Integration with Parameters in Other Testing Tools" on page 233

Parameter Categories

Business Process Testing provides several categories of parameters, such as:

- **Component / Test / Flow parameters.** Component parameters provide data for components. Similarly, flow parameters provide data at the flow level and test parameters provide data at the test level.
- **Input / Output parameters.** Input parameters are data that are used by an entity (component, flow, or test) in order for the entity to perform its function. For example, in order for a component to simulate a login operation, it must receive the login name and password as input parameters. Output parameters are data that are outputted or generated by an entity (component or flow), such as an invoice number.

For reference details, see "Parameter Categories" on page 233.

Parameter Category Example

To test the business process of a banker logging in to an online banking application, you might structure a business process test from components that:

- Log in to the application (Login)
- Select a customer loan (SelectLoan)

- View transactions for the loan (ViewLoan)
- Log out (Logout)

The steps in each of these business components can be set up to receive data from the business process test that runs the components (for example, the loans that a customer has). Any element of data, which may have different values each time the business component is run, can be parameterized. For example, the banker may choose a different customer and customer loan to view each time he or she logs in.

Here are parameters you might create for this scenario, listed by category:

Category	Parameters
Input Component Parameters	• LoginName, entered as input by the banker while logging in • AccountNo, entered by the banker, perhaps from a written inquiry
Output Component Parameters	• SessionNo, a number for the login session, outputted by the business component when the banker logs in successfully • SelectedAccountNo, outputted by the business component after the banker selects a loan from a list
Test Parameters	• CustomerLoans, a comma-delimited list of all loans for a particular customer, accessed from the test level

Parameter Linkage

For component parameters within a business process test or flow to be accessible to other components, the parameters can be linked. You link output parameters from one component to input parameters in another component.

Tip: Flow parameters can also be linked.

For task details, see "How to Link Data" on page 270.

Parameter Linkage Example

Assume that a banking application contains business processes for:

- Selecting a customer loan (SelectLoan)
- Viewing transactions for the loan (ViewLoan)

You can structure your business process test so that it contains a component for selecting the loan, SelectLoan, and a component for viewing the loan's transactions, ViewLoan. For ViewLoan to know which loan to view, it receives an input parameter, for example, ViewLoanID, from the SelectedLoanID output parameter in the SelectLoan component.

Parameter Promotion

Promoting parameters enables the components in other flows (and components and flows in other

tests) to access a parameter value. Promotion widens the scope of a parameter so that more entities can use its value when a business process test runs.

You can promote component parameters to the flow or test level at the same time as you add a component to a flow or test. Similarly, flow parameters can be promoted to the test level at the same time as you add a flow to a test.

For task details, see "How to Promote Parameters" on page 276.

Promotion Example

Continuing the previous example, you might decide that once a loan ID is selected by the banker, all components within a specific flow or test should have access to that loan ID.

To make the SelectedLoanID output parameter available to all components in a flow, promote in the flow level and/or test level. Once promoted, the following sample business components can use SelectedLoanID as an input parameter. Any component in the flow or test would then have access to the SelectedLoanID without needing to link input and output parameters:

- Searching for a loan (SearchLoan)
- Approving a loan (ApproveLoan)
- Printing a loan (PrintLoan)
- Canceling a loan (CancelLoan)
- Closing a paid-off loan (CloseLoan)

Iterations and Test Configurations

Parameterization enables you to iterate specific components in a business process test, specific components in a flow, entire business process tests, or test configurations for a business process test—thereby creating data-driven tests.

- Defining iterations enables you to automatically run business components, flows, and tests multiple times, each time using different parameter values. For conceptual details, see "Iterations Overview" on page 284.
- You can run business process tests for different use-cases of the application you are testing. Examples of use-cases include: Running the same test on different operating systems, running the same test for different browser versions, or running the same test with different languages' character sets.

  Each use-case of a test is represented by a test configuration. You can associate different sets of data for each test configuration and its iterations. For conceptual details, see "Business Process Test Configurations Overview" on page 200.

Iteration Example

You, the tester, must test that the business process in the banking application for approving loans works as expected for different scenarios:

- Loans pre-approved during the last marketing campaign are automatically approved
- Loans under a certain amount follow the standard business process for approving loans
- Loans over a certain amount must be flagged for additional approvals

To test the business process, you can iterate the ApproveLoan component, providing different LoanAmount and PromotionCode parameter value combinations for each iteration.

Test Configuration Example

You, the tester, must verify that a manual business process test, CreateLoan, in the banking application runs as expected in different countries, since the bank has branches all over the world. You need to make sure that data can be supplied to the test in different languages.

You create different test configurations for each supported language, and supply different data tables for each test configuration using characters unique to that language. For example, you can create a test configuration called CreateLoanFrench for branches in French-speaking countries. The data for this test configuration might contain customer names with grave accents (à). Also, the Franc (₣) monetary symbol may appear in the data.

Integration with Parameters in Other Testing Tools

You can also define input and output parameters for a business component in *UFT*. For details, see the *HP Unified Functional Testing User Guide*.

Parameter Categories

Below is an explanation of the parameter categories for Business Process Testing.

Parameter Categories	Parameter and Description
Input / Output Input and output parameters can be linked to make data available between components or flows within the same business process test. See "Data Linkage Examples" on page 267.	**Input parameters** enable you to define data used by a component or flow that is provided from an external source. When creating components, tests, and flows, you define how the values are supplied for input parameters. An input parameter can receive: • A predefined default value, if no other value is supplied by the test or flow. • An output parameter value returned by a component or flow earlier in the flow or test. • A parameter value that is supplied at the test or flow level, when the test or flow is run. **Output parameters** allow data values retrieved from a component step or flow (the **source**) to be passed as input parameters to a subsequent component or flow (the **target**) in the test run. **Note:** You cannot set a default value for an output parameter.
Component, Flow, and Test You can use parameters at the component level, the flow level, and at the test level. Parameters can be promoted from one level to the next. See "Example: Parameter Promotion" on page 276.	**Component parameters** are parameters defined within a component. Component parameters can be input or output parameters. These parameters are available to: • All subsequent steps in the same component. • Subsequent components in a flow or a test, provided that: ▪ The component parameter is defined as an output parameter in the current test or flow, and as an input parameter in the subsequent component in the current test or flow. ▪ The output parameter in the current test or flow is linked to the input parameter of the subsequent component in the current test or flow. **Flow parameters** are parameters defined within a flow. These parameters are available to all components in the flow. Like component parameters, flow parameters can be input or output parameters. **Test parameters** are parameters defined within a business process test. These parameters are available to all components and flows in the test. Test parameters can only be input parameters.

Parameter Categories	Parameter and Description
Local Local parameter values are not available outside of the component in which they are defined.	**Local parameter** values are defined within a business component and can be accessed by that component only. It is intended for use in a single step or between component steps, for example, as an output parameter for one step and an input parameter for a later step. This type of parameter is generally used when working with automated UFT components in keyword GUI view. For details, see "How to Work with Keyword GUI Automation" on page 93.

How to Handle Data in Business Process Testing

This task provides general information for how to work with parameters, iterations, and configurations in Business Process Testing.

This task is part of a higher-level task. For details, see "Business Process Testing Workflow" on page 28.

This task includes the following steps:

- "Design data" below
- "Create parameters and set default values" below
- "Link parameters" on next page
- "Promote parameters" on next page
- "Set data values for each iteration" on next page
- "Set data values for each test configuration" on next page
- "Set data values for test instances" on next page
- "Import and save parameters - optional" on next page

1. ## Design data

   Consider the following before working with parameters:

   - Determine which parameters are dependent on other parameters so that you can link them. For conceptual details, see "Parameter Linkage Overview " on page 266.
   - Determine which parameters should be available at the component, flow, and test levels. For conceptual details, see "Parameter Promotion Overview" on page 276.
   - Business process tests can be iterated for different use-case scenarios by setting up iterations and test configurations with different values. Design how many times each component, flow, and business process test configuration should run, and with what values. For conceptual details, see "Iterations Overview" on page 284 and "Business Process Test Configurations Overview" on page 200.

2. ## Create parameters and set default values

Define input and output parameters. For task details, see "How to Create Parameters" on page 238.

Set default values for input parameters. For task details, see "How to Set Parameter Values" on page 239.

You can use the default value for providing a default value in case a value is not supplied for the run, or you can use the default value as an example for the type of value that can be provided (for example, a phone number example could be **###-###-####**).

3. ## Link parameters

   Link output and input parameters between components. For task details, see "How to Link Data" on page 270.

4. ## Promote parameters

   Parameter promotion enables you to create new parameters based on parameters defined on the component or flow level. The new parameters are then accessible to other components and flows in the test.

   As you promote parameters, new parameters with the same name are created at the higher level.

   You promote parameters to the next level when adding components or flows to a test, or when adding components to a flow.

   For task details, see "How to Promote Parameters" on page 276.

5. ## Set data values for each iteration

   Define how many times each business component or flow should be iterated when the business process test runs. For task details on iterating components, see "How to Set Data for Iterations" on page 287.

6. ## Set data values for each test configuration

   Define configurations which determine how many times each business component or flow, or test should be iterated for each use-case. Set values for each test configuration's parameters. For task details on iterating business process test configurations, see "How to Work with Business Process Test Configurations" on page 203.

7. ## Set data values for test instances

   Define how many times each business component or flow should be iterated for a test instance, and set values for parameters. For task details, see "How to Set Data for Iterations" on page 287.

8. ## Import and save parameters - optional

   The task of defining lists of component or flow parameter values for iterations can be facilitated by importing the values from an external file. You can also save a set of defined parameter values as a file, for example, to use with other business components or flows. For task details, see "How to Import and Export (Save) Parameter Values" on page 289.

Chapter 15

Parameters

This chapter includes:

Parameters Overview 238

How to Create Parameters 238

How to Set Parameter Values 239

Methods for Creating Parameters 241

Parameters User Interface 242

Parameters Overview

This chapter describes how to create and work with Business Process Testing parameters at the component, flow, and test levels.

You can define parameters using various methods and at different points in working in Business Process Testing. For reference details, see "Methods for Creating Parameters" on page 241.

For conceptual details on parameters, see "Data Handling Overview " on page 230.

For task details on creating parameters, see "How to Create Parameters" below.

For task details on setting parameter values, see "How to Set Parameter Values" on next page.

You assign the following types of values to parameters:

- **Default values.** If no other value is specified, Business Process Testing uses default values when running business process tests and flows. You set default values when creating and editing variables in the Business Components and Test Plan modules. Only input parameters can have default values.
- **Actual values.** These values are used when the test or flow runs. You can specify actual values in the Test Plan module when working with iterations and test configurations. You can also specify actual values when running a business process test or flow in the Test Lab module.

How to Create Parameters

This task describes some ways to create parameters.

For a comprehensive list of all the ways to create parameters, see "Methods for Creating Parameters" on page 241.

For task details on setting values for parameters, see "How to Set Parameter Values" on next page.

For a use-case scenario related to this task, see How to Create Parameters - Use-Case Scenario.

This task includes the following steps:

- "Create parameters" below
- "Create parameters while defining design steps" below
- "Generate many parameters at one time while defining iterations - optional" on next page
- "Create parameters by promotion" on next page
- "Results" on next page

Create parameters

You can define component, flow, and test parameters in the Parameters tab. This is the basic method for defining parameters. For user interface details, see "New Parameter / Parameter Details Dialog Box" on page 257.

Create parameters while defining design steps

ALM enables you to create input and output component parameters directly in the Design Steps tab

as you define design steps.

1. In the Business Components module > Design Steps tab, create a new step or modify an existing step, and position your cursor in the **Description** or **Expected Result** field.
2. Create the new parameter in one of the following ways:
   - Click the **Insert Parameter** button to open the Parameters dialog box. Enter a name, default value (for input parameters), and a description. For user interface details, see "New Parameter / Parameter Details Dialog Box" on page 257 and "Parameters Page" on page 243. The parameter is added after the currently-selected parameter.
   - Enter a parameter name enclosed between three sets of angle brackets (**<<<*parameter*>>>**) to quickly create a parameter without defining a default value or a description.

Generate many parameters at one time while defining iterations - optional

While defining iterations in your flow or test, you can create many flow or test parameters at one time. For user interface details, see "Create Flow/Test Parameters Dialog Box" on page 298.

Create parameters by promotion

When you promote parameters, new parameters with the same name are created at a higher level. The Promote Parameters dialog box enables you to choose whether to promote component or flow parameters to the next level by selecting the parameters you want to promote. For task details, see "How to Promote Parameters" on page 276. For user interface details, see "Promote Parameters Dialog Box" on page 278.

Results

Open the **I/O Parameters** dialog box to see a list of all parameters. For details, see "I/O Parameters Dialog Box" on page 272.

How to Set Parameter Values

This task describes how to set different types of values for Business Process Testing parameters.

This task includes the following steps:

- "Specify string, numeric, and boolean values" below
- "Specify a table value" on next page
- "Specify a dynamic date value" on next page
- "Encrypt the value" on next page

Specify string, numeric, and boolean values

In any tab, dialog box, or grid, where you have the option of entering free text, you can enter string, numeric and boolean values. In a grid, you also have the option of clicking the down arrow to open the Set Values page, where you can enter your string.

Regardless of the type of value specified (string, numeric, boolean), Business Process Testing saves the value as a string.

For user interface details, see "Set Values Dialog Box" on page 248.

Business Process Testing Enterprise Extension: The only value that is not saved as a string is the table value, which is available in Business Process Testing Enterprise Extension only.

Specify a table value

This parameter type, which represents an enterprise application (such as SAP) table, is only available when a flow is learned using Business Process Testing Enterprise Extension. This is the only value that is not stored internally as a string.

For user interface details, see "Edit Table Parameter Dialog Box" on page 260.

Specify a date value

In any tab, dialog box, or grid, where you have the option of entering dates, click the down arrow to open the Set Values page.

In the Set Values page, click **Set Date**. A calendar appears enabling you to set a date value.

For user interface details, see "Set Values Dialog Box" on page 248.

Note: The dates use the date on the server (not the client).

Specify a dynamic date value

A dynamic date is a date relative to the current date (today, yesterday, a year from today). You can add or subtract an offset time period to the relative date that you specify (such as 3 months before one year ago). For example, to specify a date 10 days before the current date, the dynamic date could be: **WeekAgo - 3 days**

In any tab, dialog box, or grid, where you have the option of entering dynamic dates, click the down arrow to open the Set Values page.

In the Set Values page, click **Set Dynamic Date**. Fields appear enabling you to set dynamic dates.

For user interface details, see "Set Values Dialog Box" on page 248.

Encrypt the value

Encrypted values are displayed as asterisks so that their actual values cannot be seen.

In any tab, dialog box, or grid, where you have the option of encryption, click the down arrow to open the Set Values page.

In the Set Values page, select the **Encrypted** checkbox.

Caution: Once encrypted, the value cannot return to its original value. If you clear the **Encrypted** checkbox, the value is reset to empty.

For user interface details, see "Set Values Dialog Box" on page 248.

Methods for Creating Parameters

You can define parameters using various methods. When defining input parameters, you can also set default values.

Method	Purpose / Use	Available from	See Also
Basic	Lets you define parameters, their descriptions, and, for input parameters, default values.	Business Component / Test Plan modules > Parameters tabs	For user interface details, see "Parameters Page" on page 243. For task details, see "How to Create Parameters" on page 238
Component Step Definition	Provides the ability to define parameters "on the fly" directly when defining component steps.	Business Component module > Design Steps tab / dialog box	For task details, see "How to Create Parameters" on page 238.
Creation of Parameters while Setting Values to Other Parameters	Enables you to define flow and test parameters "on the fly" while setting values for component or flow parameters, such as when the value for a flow parameter should match the value of a new, "as-of-yet-undefined" test parameter. Useful for multiple iterations. **Tip:** This method for creating parameters is convenient when you need to set different parameter values for each iteration of the test or flow in which the component (or flow) exists.	Test Plan module > Iterations dialog boxes > Set Values page	For user interface details, "Set Values Dialog Box" on page 248.

Generation	Allows you to generate input flow and test parameters from business component parameters. Useful for multiple iterations.	Test Plan module > Iterations dialog box > **Create Test Parameters** button	For user interface details, see "Create Flow/Test Parameters Dialog Box" on page 298.
Promotion	Enables you to widen the scope of a parameter so that more entities can use its value when a business process test runs. For example, when a business component parameter is promoted to the flow level and then to the test level, a parameter with the same name is created for the flow and the test. All other entities, whose parameters have also been promoted, access the highest-level value for the parameter.	Test Plan module > Test Script tab > Select Components and Flows pane > button	For user interface details, see "Promote Parameters Dialog Box" on page 278. For task details, see "How to Promote Parameters" on page 276.
Learn Flow	**Business Process Testing Enterprise Extension:** Generates flow parameters automatically during the Learn Flow process.	Test Plan module > Test Script tab > **Learn Flow** button	For task details, see "How to Learn Flows" on page 170

Parameters User Interface

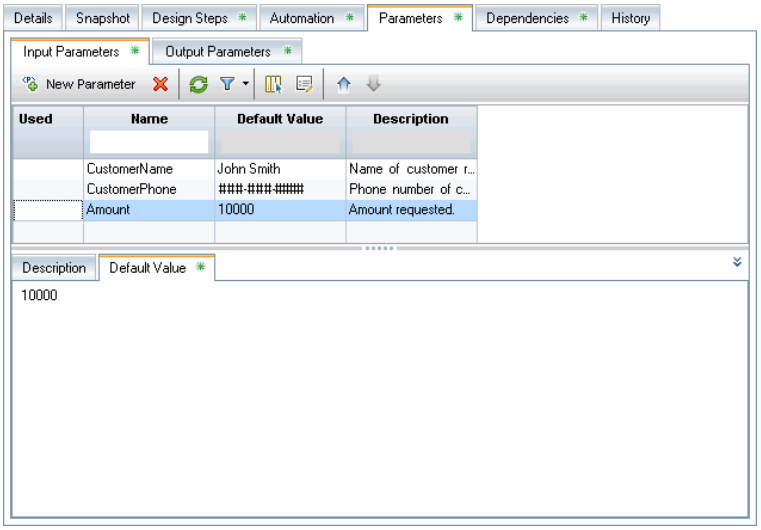
This section includes:

Parameters Page 243
Set Values Dialog Box 248
Dynamic Date Area 251
Parameters Dialog Box 254
New Parameter / Parameter Details Dialog Box 257
Edit Table Parameter Dialog Box 260

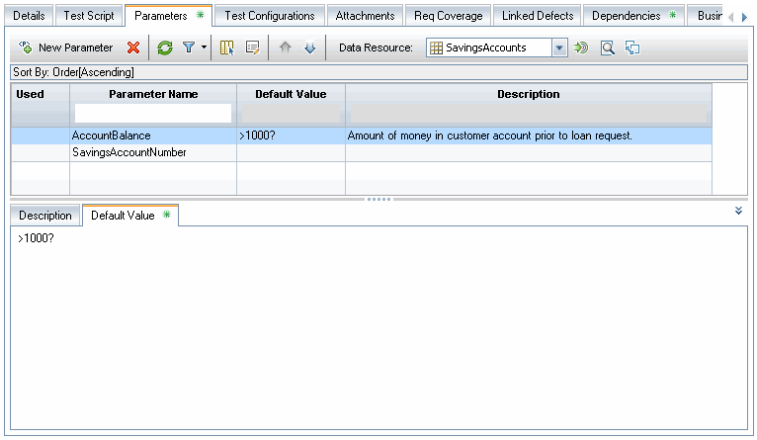
Parameters Page

This dialog box or tab enables you to view, select, and insert parameters for business components, flows, and business process tests.

Parameters Tab: Component Level



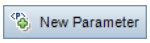
Parameters Tab: Test Level



To access	Use one of the following: • In the Business Components module component tree, select a component and click the **Parameters** tab. • In the Test Plan module test tree, select a business process test or a flow and click the **Parameters** tab. • In the Test Plan module test tree, select a business process test or a flow and click the **Test Script** tab. Click the **Parameters** tab at the bottom of the window.

Important information	• You can insert component parameters into your steps in the Design Steps tab in the Business Components module. After you select a parameter, the parameter is inserted at the cursor location in the component step, using the syntax `<<<parameter name>>>`. For task details, see "Create parameters while defining design steps" on page 238. • Parameter values are stored, compared, and processed as characters strings (even if values are specified as numeric or date values). • When creating input flow parameters, they must be assigned the value of a component in the flow. Otherwise, the parameters have no relevance in the business process test. • Test parameters can only be input parameters. • Component parameters can be viewed and modified in the Business Component Settings dialog box, see the *HP UFT for HP Business Process Testing User Guide*. • **Notes for users of UFT:** Parameters for keyword GUI components can be viewed and modified in the Business Component Settings dialog box. In addition, you can insert Data Table parameters and output values to business components, but you can store the values only in the first row of the Data Table. Component iterations are defined for the business process test in ALM and are not affected by the Data Table.
Relevant tasks	• "How to Create Parameters" on page 238 • "How to Set Parameter Values" on page 239
See also	"Parameters Overview" on page 238

User interface elements are described below:

UI Element	Description
	**New Parameter.** Opens the New <entity> Parameter dialog box, enabling you to define a new parameter. For details, see "New Parameter / Parameter Details Dialog Box" on page 257. **Note:** This button is disabled when selecting the **Parameters** tab from the Test Script tab in the Test Plan module. **Tip:** When creating input flow parameters must be assigned the value of a component input parameter in the flow. Otherwise, they have no relevance in the business process test.
	Delete. Deletes the currently selected parameter. If you delete a parameter that is included in a component step, the parameter in the component step is replaced by regular text, using the syntax: `<parameter name>` If a business process contains the component parameter that you are deleting, the test may not run successfully. **Not available when:** • When selecting the **Parameters** tab from the Test Script tab in the Test Plan module. • From the Parameters dialog box.
	Refresh. Refreshes the component parameters so that they display the most up-to-date information. **Not available when:** • When selecting the **Parameters** tab from the Test Script tab in the Test Plan module. • From the Parameters dialog box.
	Set Filter/Sort. Enables you to filter and sort the component parameters in the grid. Any currently applied filters or sorting orders are displayed below the toolbar. For details, see the *HP Application Lifecycle Management User Guide*. **Not available when:** Selecting the **Parameters** tab from the Test Script tab in the Test Plan module.

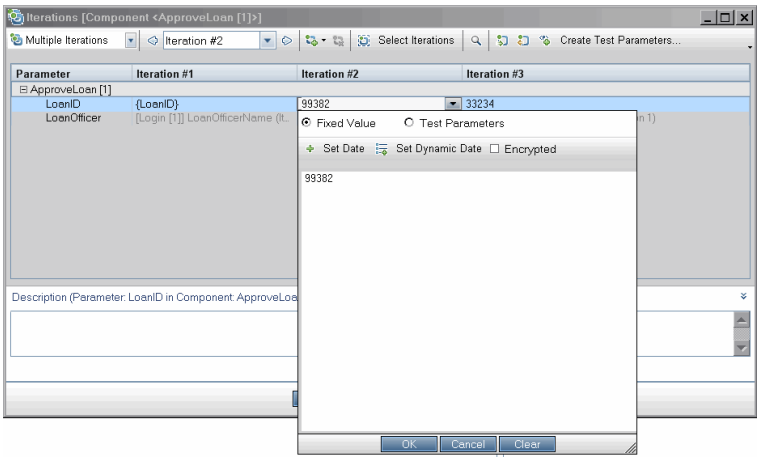
UI Element	Description
	Select Columns. Opens the Select Columns dialog box, enabling you to determine which fields are displayed and their order. For more details, see the *HP Application Lifecycle Management User Guide*. **Note**: This option is not available when selecting the **Parameters** tab from the Test Script tab in the Test Plan module.
	Parameter Details. Opens the Parameter Details dialog box, displaying the details of the selected parameter. For more details, see "New Parameter / Parameter Details Dialog Box" on page 257.
	Move Up. Moves the selected parameter up in the list. **Tip:** In other tabs and dialog boxes, such as the in Iteration pages, parameters are displayed according to the order defined in the Parameters page.
	Move Down. Moves the selected parameter down in the list. **Tip:** In other tabs and dialog boxes, such as the in Iteration pages, parameters are displayed according to the order defined in the Parameters page.
	Go To Resource. For use with configurations that access dynamic data. Opens the resource in the Test Resources module. **Available from**: Parameters tab from Test Plan module. This field is not available when selecting the **Parameters** tab from the Test Script tab in the Test Plan module.
	Resource Preview. For use with configurations that access dynamic data. Enables you to preview the actual data that will be used with the configuration when the test runs. The external data table resource is specified in **Data Resource** in the Resource Preview dialog box. The Resource Preview dialog box is similar to the Resource Viewer tab in the Test Resources module. For user interface details, see the *HP Application Lifecycle Management User Guide*. **Available from**: Parameters tab from Test Plan module. This field is not available when selecting the **Parameters** tab from the Test Script tab in the Test Plan module.

UI Element	Description
	Map Parameters. For use with configurations that access dynamic data. Enables you to specify sets of parameter values from an external Microsoft Excel file by mapping the parameter names in the Excel file to the parameter names in ALM. For task details, see "How to Work with Configurations and Dynamic Data" on page 205. **Available from**: Parameters tab from Test Plan module. This button is not available when selecting the **Parameters** tab from the Test Script tab in the Test Plan module.
Sort By	Specifies the order in which the parameters are listed, as defined in the Set Filter dialog box. For user interface details, see *HP Application Lifecycle Management User Guide*.
Data Resource	Specifies the name of an external Microsoft Excel file that contains sets of parameter values for each configuration of the test. Used only for configurations defined to access dynamic data. For task information, see "How to Work with Configurations and Dynamic Data" on page 205. **Available from**: Parameters tab from Test Plan module. This field is not available when selecting the **Parameters** tab from the Test Script tab in the Test Plan module.
<grid filters> 	Displays a grid filter box under the name of each column that can be filtered, enabling you to define a filter condition for that column. Type directly into the box, or click the box to display the browse button, which opens the Select Filter Condition dialog box. For details, see the *HP Application Lifecycle Management User Guide*.
Filters	Describes the filters currently applied to the list of parameters. (For the columns that can be filtered, the filters are located directly above the grid filter boxes.)
Used	When checked, indicates that this user interface element means the component parameter is included in a step.

UI Element	Description
Name	Displays a descriptive name for the parameter. • Parameter names must be in English. • Component parameter names must begin with a letter and cannot include spaces or any of the following characters: **! @ # $ % ^ & * ( ) + = \ { } | ; ' : " , . / < > ? ~ ` [ ] – ~** Any invalid character is automatically replaced by an underscore (_). • Test and flow parameter names must begin with a letter and cannot include spaces or any of the following characters: **< > ~ ? %** Any invalid character is automatically replaced by an underscore (_). • Parameter names must be unique within a component, flow, or test. • The parameter name must begin with a letter. If the first character of the component name is not a letter, a prefix, such as **comp_,** is added.
Input Parameters	Lists all defined input parameters. **Available from**: Parameters tabs
Output Parameters	Lists all defined output parameters. **Available from**: Parameters tabs for components and flows
Description	Tab providing a description of the component parameter's purpose, or other descriptive text.
Default Value	Tab containing the default value of the input parameter. This value is used when the component runs if no other value is supplied by the business process test or flow. **Available from:** Input Parameters tabs

Set Values Dialog Box

This dialog box enables you to set default and actual values for parameters at the component, flow, and test levels.



To access	1. Click in a field or cell in which you want to set a value. 2. Click the drop-down arrow on the right side of the cell or field, if an arrow exists.

Important information	• The structure and fields of the dialog box change depending on:
	  ▪ The location from which you opened the dialog box
	  ▪ The radio buttons or buttons you selected in this dialog box
	• All parameter values are internally stored as strings, regardless of how the parameter values are specified. For example, you can set a value as a date, using a calendar. But when the value is stored and compared to other values, it is compared as a string.
	• Table parameters, which represent enterprise application tables, are only available when a flow is learned using Business Process Testing Enterprise Extension. For details, see "Edit Table Parameter Dialog Box" on page 260.
	• You can provide a default value in case a value is not supplied for the run.

Relevant tasks	• "How to Create Parameters" on page 238 • "How to Set Parameter Values" on page 239
See also	"Parameters Overview" on page 238

Set Value dialog box views are described below.

Fixed Value

The Fixed Value view displays areas, fields, and buttons that enable you to enter a fixed value for the parameter.

To access	1. If accessing from Test Plan > Flow and Component Iteration dialog boxes or the Test Lab > Test Iterations tab, when a component or a flow is included in a test: Select **Fixed Value**. 2. Click the field or cell for the value. 3. Click the drop-down arrow on the right side of the cell or field, if an arrow exists.
Important information	• This is the default view for the Set Values dialog box. • You can enter XML code values, but use an external tool to validate the XML. • Values can also be entered by typing a string value directly in the relevant cell in the grid. **Tip:** You can also use the default value to show the tester a sample value. A default value for phone number, for example, could be **###-###-####**.

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

UI Element	Description
	Displays a calendar area, enabling you to select a fixed date. Select **Encrypted** to encrypt the date—optional.
	Displays the dynamic date area, enabling you to select a date that is relative to a fixed date (such as **Today**, **WeekAgo**). For user interface details, see "Dynamic Date Area" on next page.
Encrypted	Encrypts the fixed value (including dates). The characters you enter are encrypted and displayed as asterisks. You can modify an encrypted value. It remains encrypted. When cleared, erases the fixed value. Once a value is encrypted, it can only be modified in the Set Value tab, and not directly in the grid. **Available when:** Entering fixed values, and if it appears, **Fixed Value** is selected.
<text area>	Displays the current value for the parameter, and enables you to enter and modify the value. When you enter a parameter value as a string, numeric, or boolean value, the value is treated in the same way as any other character. For example, "`1`" and "`1.0`" are treated as different strings.

Test Parameters

The Test Parameters view displays the Parameters dialog box. Use the Parameters dialog box to select (or create) a flow or test parameter value to use for the selected parameter.

To access	1. From Test Plan > Flow and Component Iteration dialog boxes or the Test Lab > Test Iterations tab: Select **Test Parameters**. 2. Click the field or cell for the value. 3. Click the drop-down arrow on the right side of the cell or field, if an arrow exists.

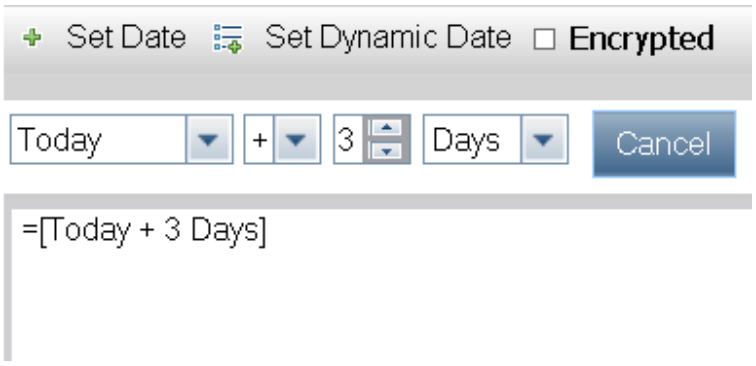
Important information	• You can set the value of a component parameter to the value of a flow or test parameter. Similarly, you can set the value of a flow parameter to the value of a test parameter. • After selecting or creating a test or flow parameter, the name of the parameter is entered as a value in the grid, surrounded by curly brackets **{ }** . The curly brackets instruct ALM to treat the parameter as a parameter, and not as a fixed value.
See also	"New Parameter / Parameter Details Dialog Box" on page 257

User interface elements are described below:

UI Element	Description
Promoted to:	Displays promotion information for the selected parameter. Located at the bottom of the Test Parameters view. For conceptual details about parameter promotion, see "Parameter Promotion Overview" on page 276. **Example**

Dynamic Date Area

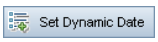
This area enables you to set dynamic dates.



To access	From any Set Values page, select **Set Dynamic Date**.
Important information	A dynamic date is a time relative to the current date (such as **Today**, **WeekAgo**).

Relevant tasks	"How to Set Parameter Values" on page 239
See also	"Set Values Dialog Box" on page 248

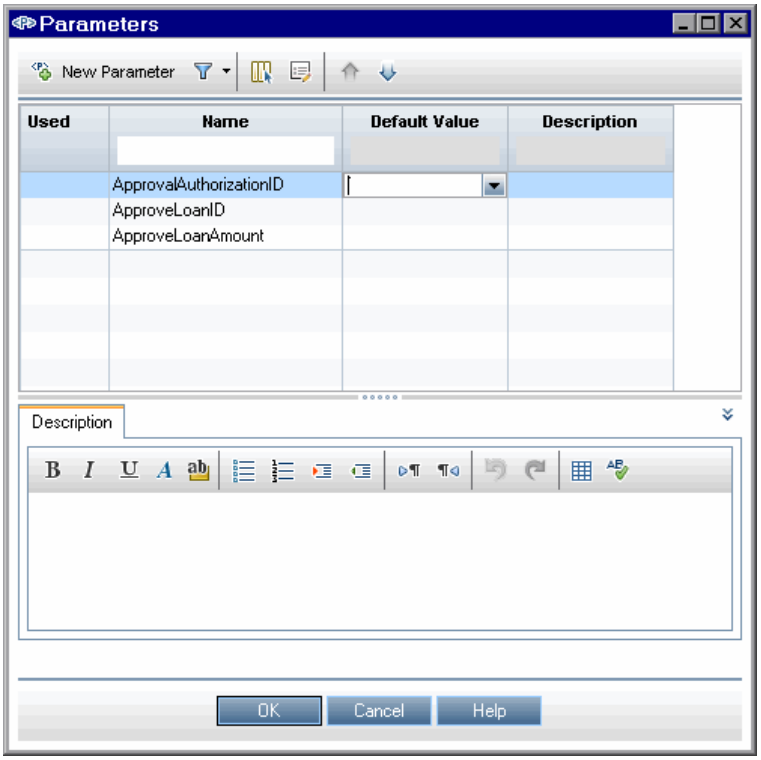
User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
	Displays **<calendar area>**, enabling you to select a fixed date value using a calendar.
	Displays **<dynamic date area>**, enabling you to select a date that is relative to a fixed date.
<relative time frame>	Contains a list of relative time frames: • Today • Yesterday • Tomorrow • WeekAgo • WeekFromToday • MonthAgo • MonthFromToday • YearAgo • YearFromToday Date strings are not case-sensitive. For example, both **[WeekAgo]** and **[Weekago]** are valid values.
<+/->	Determines whether the **<offset>** and <**time period>** values are added to (**+**) or subtracted from (**-**) the **<relative time frame>** value.
<offset>	Contains a list of positive integers (including zero), enabling you to determine the number of days/weeks/months/years by which you want to adjust the relative time frame. The maximum offset values that apply to time periods and used to modify date strings are: • 5000 for years • 60000 for months • 240000 for weeks • 1680000 for days **Example** **[today + 5000 years]** is valid. **[today + 5001 years]** is not valid.

UI Element	Description
<time period>	Contains a list of time periods, enabling you to adjust the relative time frame. The options are: • Day (or Days or D). Time period in days. • Week (or Weeks or W). Time period in weeks. • Month (or Months or M). Time period in months. • Year (or Years or Y). Time period in years. Time periods used to modify date strings are not case-sensitive. For example, both **[YearFromToday + 5 Months]** and **[YearFromToday + 5 months]** are valid. Spaces are not required in dynamic date values. For example, both **[MonthFromToday + 3 w]** and **[MonthFromToday+3w]** are valid. **Examples** • [Tomorrow + 1 day] • [Today + 3 days] • [Yesterday - 5 d] • [Tomorrow - 1 week] • [Today + 2 weeks] • [MonthFromToday + 3 w] • [YearAgo - 1 month] • [YearFromToday + 5 months] • [Today + 3 m] • [Today - 1 year] • [Tomorrow + 2 years] • [WeekFromToday + 2 y]
<expression area>	Displays the expression created from selecting the **<relative time frame>**, **<offset>**, and **<time period>**. You can also manually customize a date string by typing the value of the relative time frame and adding or subtracting a time period to or from it.

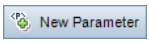
Parameters Dialog Box

This dialog box enables you to insert existing parameters in the steps described in the cells in the Description and Expected Result columns of a step.



To access	1. In the Business Components module, select a component with manual design steps from the Components tree. 2. In the Design Steps tab > Design Steps pane, do one of the following: ▪ Double-click a step to insert the parameter in the Component Step Details dialog box. ▪ Click **New Step** to add a step and insert the parameter in the Component Step Details dialog box. ▪ Edit a step directly in the Design Steps pane. 3. Place the curser in a cell in the **Descriptions** column or the **Expected Result** column. 4. Click **Insert Parameter** .
Relevant tasks	"How to Create Parameters" on page 238

User interface elements are described below:

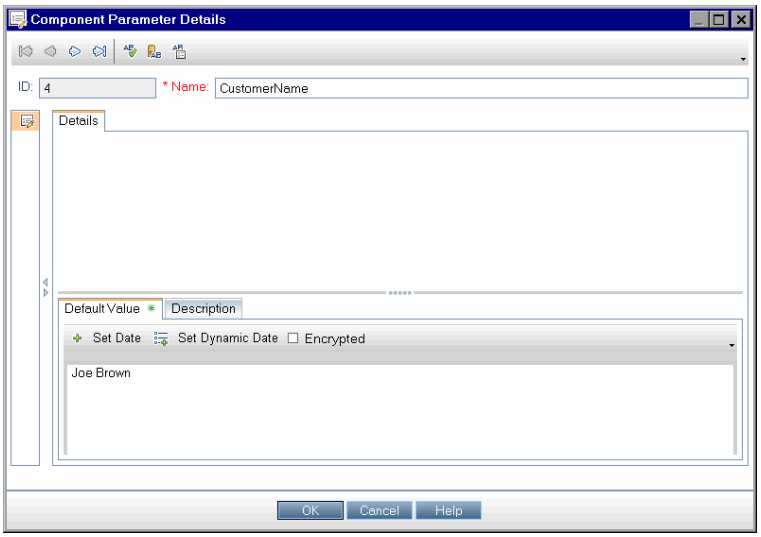
UI Element	
	Opens the New Component Parameter dialog box, enabling you to define a new component parameter. For more details, see "New Parameter / Parameter Details Dialog Box" on next page.
	Set Filter/Sort. Enables you to filter and sort the component parameters in the grid. Any currently applied filters or sorting orders are displayed below the toolbar. For details, see the *HP Application Lifecycle Management User Guide*.
	Select Columns. Opens the Select Columns dialog box, enabling you to determine which fields are displayed and their order. For more details, see the *HP Application Lifecycle Management User Guide*.
	Parameter Details. Opens the Component Parameter Details dialog box, displaying the details of the selected parameter. For more details, see "New Parameter / Parameter Details Dialog Box" on next page.
	Move Up. Moves the selected parameter up in the list. **Tip:** In other tabs and dialog boxes, such as the in Iteration pages, parameters are displayed according to the order defined in this dialog box.
	Move Down. Moves the selected parameter down in the list. **Tip:** In other tabs and dialog boxes, such as the in Iteration pages, parameters are displayed according to the order defined in this dialog box.
Used	When checked, this user interface element means the component parameter is included in a step.
Name	A descriptive name for the parameter. • Parameter names must begin with a letter and cannot include spaces or any of the following characters: **! @ # $ % ^ & * ( ) + = \ { } | ; ' : " , . / < > ? ~ ` [ ] – ~** Any invalid character is automatically replaced by an underscore (_). • Parameter names must be unique within a component, flow, or test. • The parameter name must begin with a letter. If the first character of the component name is not a letter, a prefix, such as **comp_,** is added.
Default Value	The default value of the parameter. This value is used when the component runs if no other value is supplied by the business process test or flow. Default values are saved and compared as strings, but can be specified according to different value types.
Description	Provides a description of the component parameter's purpose, or other descriptive text.

New Parameter / Parameter Details Dialog Box

The New Parameter dialog box enables you to create:

- Input and output component parameters
- Input and output flow parameters
- Input test parameters

The Parameter Details dialog box enables you to view the details for parameters.



To access	• **Parameter Details dialog box:** From a Parameters tab or dialog box, double-click the parameter. • **New Parameter dialog box:** In the Business Components or Test Plan module, select the **Parameters** tab and click **New Parameter**. **Tip:** You can also access the New Parameter dialog box from: • **Set Values page.** Select the **Test Parameters** radio button, when available, in the Set Values page and click the **New Parameter** button. • **Business Components Module.** In the Design Steps tab's Steps pane, click the **Insert Parameter** button and then click the **New parameter** button.

Important information	Parameters are inserted after the currently-selected parameter. Parameter values are stored, compared, and processed as characters strings (even if values are specified as numeric or date values). **Notes for users of** UFT**:** Parameters for keyword GUI components can be viewed and modified in the Business Component Settings dialog box. In addition, you can insert Data Table parameters and output values to business components, but you can store the values only in the first row of the Data Table. Component iterations are defined for the business process test in ALM and are not affected by the Data Table.
Relevant tasks	• "How to Create Parameters" on page 238 • "How to Set Parameter Values" on page 239
See also	• "Parameters Overview" on page 238 • "Parameters Page" on page 243

User interface are described below:

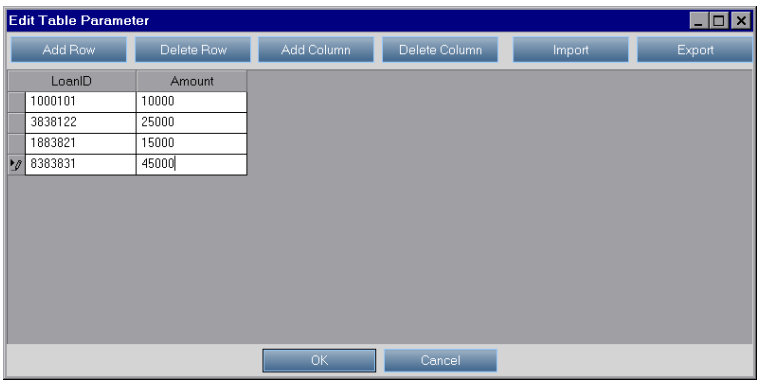
UI Element	Description
	First Entity. Jumps to the first component in the business components module. **Available from**: Parameter Details tabs/dialog boxes
	Previous Entity. Jumps to the previous component in the business components module. **Available from**: Parameter Details tabs/dialog boxes
	Next Entity. Jumps to the next component in the business components module. **Available from**: Parameter Details tabs/dialog boxes
	Last Entity. Jumps to the last component in the business components module. **Available from**: Parameter Details tabs/dialog boxes
	This button does one of the following: • **Delete.** From a parameter list, deletes the currently selected parameter. If you delete a parameter that is included in a design step, the parameter in the design step is replaced by regular text, using the syntax: `<parameter name>`. • **Reset all fields.** From the New Parameter dialog box, clears the values in all fields.
	Spell Check. Checks the spelling of the selected word or all the words in the parameter.
	Thesaurus. Displays a synonym, antonym, or related word for the selected word.
	Spelling Options. Configures the way ALM checks spelling.
ID	The ID of the parameter.
Parameter Name	A descriptive name for the parameter. • Parameter names must be in English. • Parameter names must begin with a letter and cannot include spaces or any of the following characters: **! @ # $ % ^ & * ( ) + = \ { } | ; ' : " , . / < > ? ~ ` [ ] – ~** If the parameter is in a manual test that was converted to a business component, any invalid character is automatically replaced by an underscore (_). • Parameter names must be unique within a component, flow, or test. • The parameter name must begin with a letter. If the parameter is in a manual test that was converted to a business component, and the first character of the component name is not a letter, a prefix, such as **comp_,** is added.

UI Element	Description
Details tab	Enables you to modify details for the parameter.
Mapping Status	The name of the corresponding external data resource parameter (column) to which the parameter is mapped. This UI element is blank if no parameter is mapped. **Available from:** Test Parameter Details dialog box
Order	The position of the parameter in the set of parameters. **Available for:** Test and flow parameters
Default Value tab	The default value of the input parameter. This value is used when the component runs if no other value is supplied by the business process test or flow. The default value can be entered in the grid directly, in the Set Values page, or in the tab at the bottom of the window. For user interface details, see "Set Values Dialog Box" on page 248. **Available from:** Input Parameters sub-tabs
Description tab	A description of the parameter's purpose, or other descriptive text. This tab enables you to modify details for the parameter.

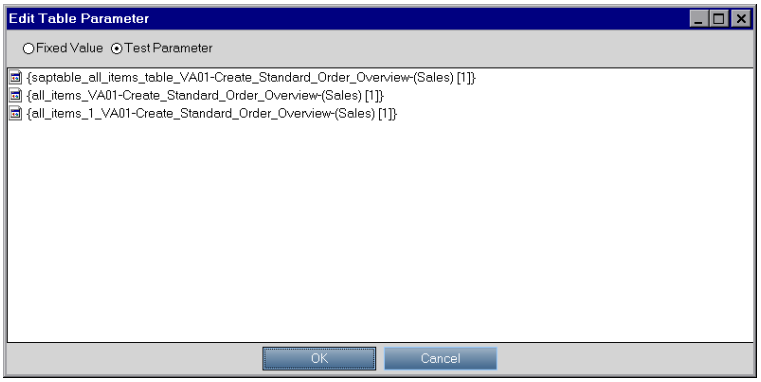
Edit Table Parameter Dialog Box

This dialog box enables you to modify table values for parameters created during the learn flow process. The table parameter definition includes the table structure and default values.

Fixed Values View



Flow/Test Parameter View



To access	You can access the Edit Table Parameter dialog box on the component, flow, test, or test set level. • **Component, flow, test levels**: When setting default values for parameters in a Parameters tab in the Test Plan module, the table parameter default value is displayed as **[Table Data]**. Click this link to open the Edit Table Parameter dialog box in Fixed Value view. • **Iteration levels**: When setting default values for parameters in an Iterations dialog box in the Test Plan module, the table parameter default value is displayed as **[<table_name>]**. Click this link to open the Edit Table Parameter dialog box in Flow/Test Parameter view. • **Test set level**: When setting values for iterations in the Test Lab module's Execution Grid, click the **Iterations** link for the relevant test set. In the Iterations dialog box, the table parameter iteration value is displayed as **[Table Data]**. Click this link to open the Edit Table Parameter dialog box in Fixed Value view.
Important information	• This dialog box is only available for Business Process Testing Enterprise Extensionusers, and only available for learned components that contain enterprise-specific table parameters. • Table parameters are available for use in: ▪ Components ▪ Flows ▪ Tests ▪ Test sets ▪ Iterations ▪ Run conditions
Relevant tasks	"How to Set Parameter Values" on page 239
See also	"How to Learn Flows" on page 170

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

UI Element (A-Z)	Description
Fixed Values view	Enables you to enter fixed values into the cells to the table parameter in a grid format.
Flow/Test Parameter view	Enables you to select a different table parameter instead of the current one (provided the table parameter is at the flow or test level).
Fixed value	Displays the Fixed Values view. **Available when**: Accessing the Edit Table Parameter dialog box via an iterations dialog box in the Test Plan module.
Flow parameter	Displays the Flow/Test Parameter view, enabling you to select a different flow table parameter. **Available when**: Accessing the Edit Table Parameter dialog box via an iterations dialog box in the Test Plan module.
Test parameter	Displays the Flow/Test Parameter view, enabling you to select a different test table parameter. **Available when**: Accessing the Edit Table Parameter dialog box via an iterations dialog box in the Test Plan module.
Add Row	Adds a row to the table parameter grid. The row is added below the row in which the cursor is currently located. If the cursor is not in any row, the row is added at the end. **Available in:** Fixed Value view
Delete Row	Deletes the selected row from the table parameter grid. **Available in:** Fixed Value view
Add Column	Adds a column to the table parameter grid. The column is added to the right of the column in which the cursor is currently located. If the cursor is not in any column, the column is added at the end. Each column must have a unique header name. **Available in:** Fixed Value view
Delete Column	Deletes the selected column from the table parameter grid. **Available in:** Fixed Value view
Import	Imports a table parameter definition from a .csv file. Ensure that the file is in the proper format. Use the export option to generate the .csv file and then edit or add data as necessary. This ensures the proper format of the .csv file. **Available in:** Fixed Value view

UI Element (A-Z)	Description
Export	Exports the table parameter definition to a .csv (comma separated values) file. **Available in:** Fixed Value view
<column header row>	Enables renaming of columns. Double-click on the header to rename the column. Column names must match the column names in the supported business application environment application. Column headers cannot be blank or a duplicate of an existing column name. You can also use the index of a column with the **#** sign to name a column. For example, **#3**. **Available in:** Fixed Value view
<table cell>	Enables editing of the default value for a table cell. Select the cell and enter the default value. **Available in:** Fixed Value view

Chapter 16

Parameter Linkage

This chapter includes:

Parameter Linkage Overview 266

How to Link Data 270

Parameter Linkage User Interface 272

Parameter Linkage Overview

Parameter linkage enables you to make data available between business components and flows.

Product Feature Movie: To view a movie that demonstrates how to handle Business Process Testing parameters, select **Help > Product Feature Movies** in the ALM main window.

To link parameters, you pass the values of output parameters to input parameters of subsequent business components and flows in a test. The process of setting the value of an input parameter to the value of an output parameter is called linking output and input parameters.

Example

The CreateLoan business component has an output parameter that contains a generated loan ID. A subsequent business component, SearchLoan, can verify the loan if it has access to CreateLoan's loan ID value. This access is provided by linking the CreateLoan's output parameter to SearchLoan's input parameter.

The component or flow in which the output parameter is defined is the source. The component or flow that links to that output parameter is the target. In the example above, CreateLoan is the source component and SearchLoan is the target component.

For task details, see "How to Link Data" on page 270.

This section also includes the following topics:

- "Linkage and Iterations" below
- "Data Linkage Examples" on next page

Linkage and Iterations

You can configure a component or flow to run a certain number of times during a single run, using different data values for the component's or flow's parameters. Each time the component or flow runs is called an iteration. For details on iterations, see "How to Set Data for Iterations" on page 287.

Iterations of a source can result in multiple parameter output values. In these cases, the value provided by each iteration is passed as input to the corresponding iteration of the target.

Linking can occur successfully only when ALM can determine the target iteration for each source iteration. One of the following conditions must exist:

- **Condition 1.** The source has one iteration and the target has one or more iterations (a "1–to–n" relationship). For an example, see "Iterations Example ("1–to–n" Relationship)—Graphical Representation" on page 268.
- **Condition 2.** The source and the target have the same number of iterations (an "n–to–n" relationship). For an example, see "Iterations Example ("n–to–n" Relationship)—Graphical Representation" on page 269.

Note: When a source or target is a member of a group, the number of iterations is that of the

group.

If the component iterations are not represented by a “1–to–n” or “n–to–n” relationship, a warning message is displayed.

Considerations

Take the following into account when working with parameter linkage and iterations:

- When you use the output of a previous component as the value for an input component parameter, the option applies to all component iterations for that input parameter.
- When iterations of a source component in a business process test result in multiple parameter output values, the value that is provided by a given iteration run is passed as input to the corresponding iteration of the target component.
- Moving a business component, group, or flow can cause a parameter reference conflict, for example, where a target component is moved to a position preceding the source component. If the resulting warning message is ignored, the conflicting link to the source parameter is deleted. This causes the iteration to fail, and the link must be reinstated in the target Iterations dialog box if necessary.
- In cases where there is an incompatible number of iterations between input and output component parameters, the parameter name in the column header is displayed in red text in the Iterations dialog box.
- Iteration errors cause a business process test or flow that contains the relevant components to fail. These errors are indicated in the Test Script tab in the Test Plan module, and reported when a flow or test is validated. For user interface details, see "Test Validation Results Dialog Box" on page 166.

Data Linkage Examples

This topic provides examples that demonstrate how to link data.

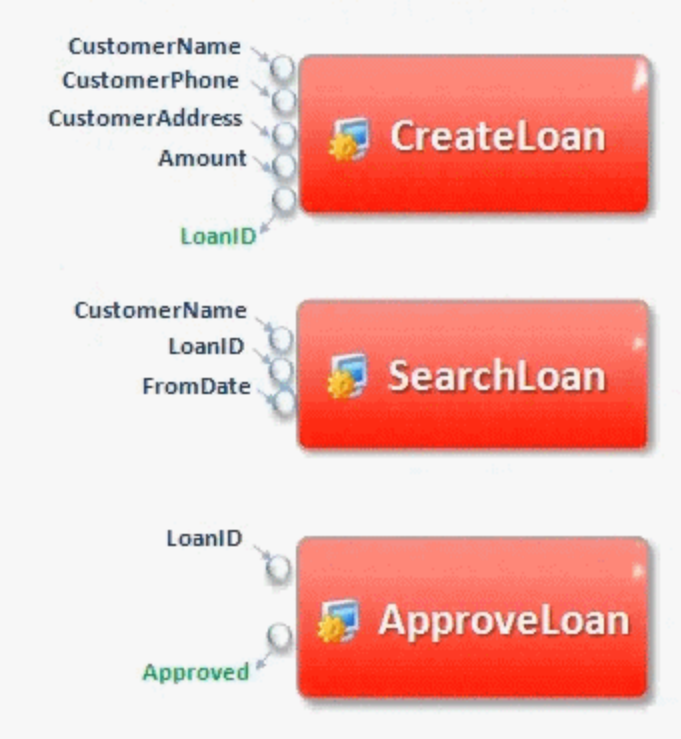
The following examples are provided:

- "Example: Linking Input and Output Component Parameters—Graphical Representation" below
- "Iterations Example (“1–to–n” Relationship)—Graphical Representation" on next page
- "Iterations Example (“n–to–n” Relationship)—Graphical Representation" on page 269
- "Iterations Examples—Representation in Iterations Dialog Box" on page 270

Example: Linking Input and Output Component Parameters—Graphical Representation

Suppose you create three components corresponding to different stages in processing a customer loan request:

- CreateLoan. Receives a loan request and generates a unique loan ID for the request.
- SearchLoan. Searches the existing loans to verify if the loan exists.
- ApproveLoan. Approves the loan request.

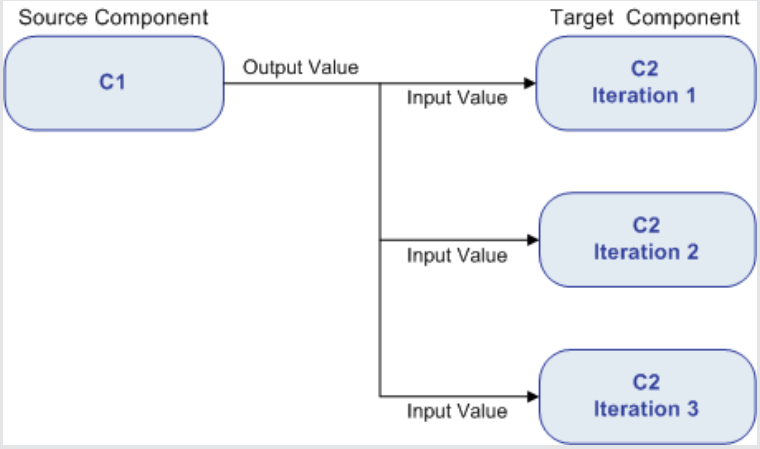


In this example, a business process test tests the entire loan processing workflow from receiving the request through approving the request. An output parameter called LoanID is created by the component CreateLoan. This output parameter stores the value of the loan ID generated when the loan is requested. You could then use this value as an input parameter for subsequent components or flows in the business process test, such as SearchLoan and ApproveLoan.

Iterations Example ("1–to–n" Relationship)—Graphical Representation

The source component has one iteration and the target component has one or more iterations (a "1–to–n" relationship).

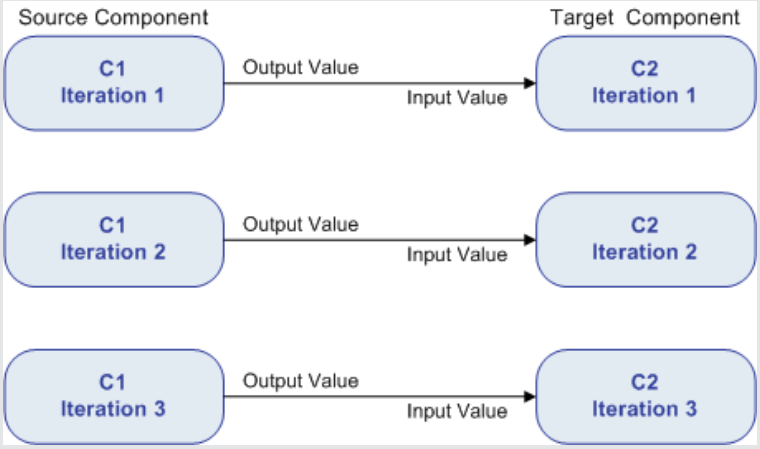
In this case, the same output value of the source component is used in each iteration of the target component



Iterations Example ("n–to–n" Relationship)—Graphical Representation

The source component and the target component have the same number of iterations (an "n–to–n" relationship).

In this case, the different output values of each of the source component's iterations are used in the respective iterations of the target component



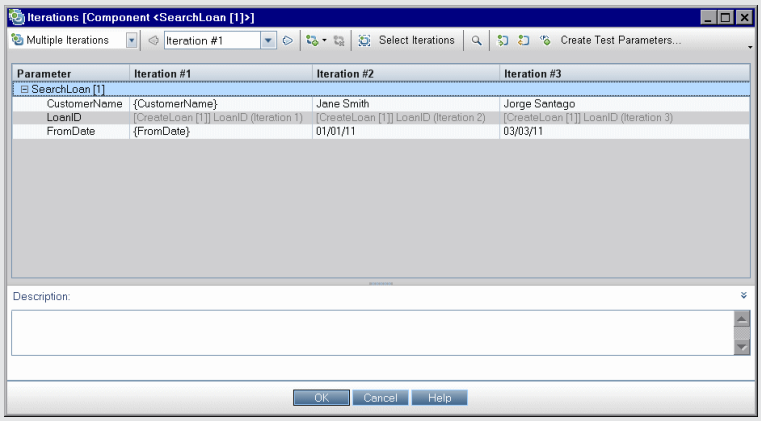
The number of iterations in the relationship between the components can be the number of actual iterations in each component or group, or the number of iterations selected in a partial range.

In this example therefore, the three output iteration values of C1 could also have been used as input values in the three iteration runs of C2 (iterations 3, 4 and 5), if that range was selected in C2.

For user interface details on selecting partial iteration ranges, see "Select Iterations Dialog Box" on page 296.

Iterations Examples—Representation in Iterations Dialog Box

In the example below, the source component CreateLoan has an “n–to–n” (“3 to 3”) relationship with the target component SearchLoan. For each iteration of component input parameter LoanID in SearchLoan, there is a corresponding iteration of component output parameter LoanID from CreateLoan.



Conversely, the source component Login has a “1–to–n” (“1–to–3”) relationship with the target component ApproveLoan. Each iteration of component input parameter LoanOfficer in the ApproveLoan component uses the same component output parameter (Username) from the Login component.

For user interface details on the Iterations dialog box, see "Iteration Pages" on page 292.

How to Link Data

This section describes how to link output and input parameters.

Note:

- This task is part of a higher-level task. For details, see "How to Handle Data in Business Process Testing" on page 235.

- For user interface details, see "I/O Parameters Dialog Box" on next page and "Select Output Parameter Dialog Box" on page 273.

This task includes the following steps:

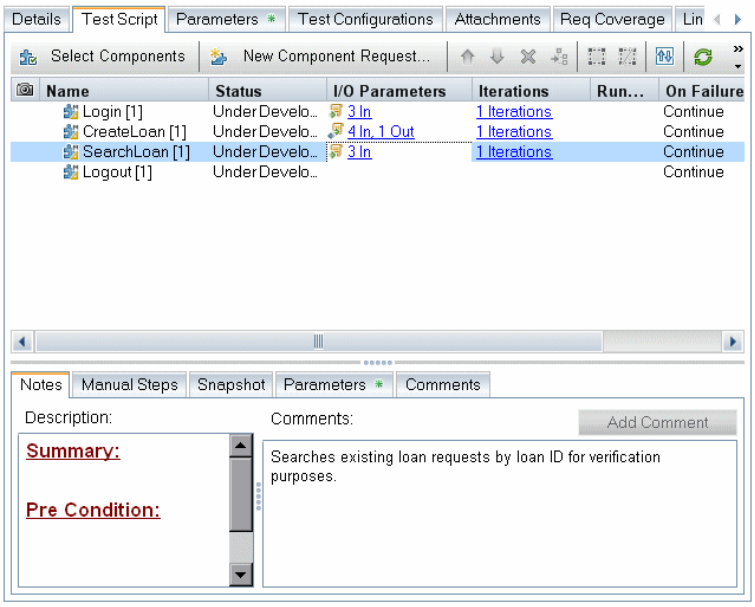
- "Design linkage" below
- "Display the parameters for the relevant business process test or flow" below
- "Link the parameters" on next page

1. ## Design linkage

   - Determine which parameters are generated by the application being tested. Create output parameters for these parameters either at the component level or the flow level.
   - Determine which input parameters are dependant on other parameters in other business components or flows. You will link these input parameters to their corresponding output parameters.

2. ## Display the parameters for the relevant business process test or flow

   From the Test Plan module, click the **Test Script** tab for the relevant business process test or flow.



   Note the I/O Parameters column. This column displays the total number of input and output parameters defined for each component or flow. For example, the SearchLoan component has three input parameters. The total **(3 in)** is a link.

   Click the link in the I/O Parameters column for a component or flow whose input parameter you want to link to an output parameter (that is, the link of the target). The I/O Parameters dialog box opens. For user interface details, see "I/O Parameters Dialog Box" on next page.

In the lower pane of this dialog box, you can see which input parameters are already linked to the output parameters.

3. **Link the parameters**

For each component or flow input parameter to link:

- Select the checkbox in the Link I/O column. The Select Output Parameter dialog box opens. The Select Output Parameter dialog box lists output parameters from each source business component or flow in the business process test prior to the current target component or flow. For user interface details, see "Select Output Parameter Dialog Box" on next page.
- Select the output parameter to link to the input parameter and click **OK**.

> **Note:** If the number of iterations for the output parameter and the input parameter do not match, the parameter name is displayed in red. The number of iterations are considered a match if either the number of output parameter iterations is equal to the number of input parameter iterations, or if the output parameter iterations is equal to one (regardless of the number of input parameter iterations).

Click **OK** to close the I/O Parameters dialog box.

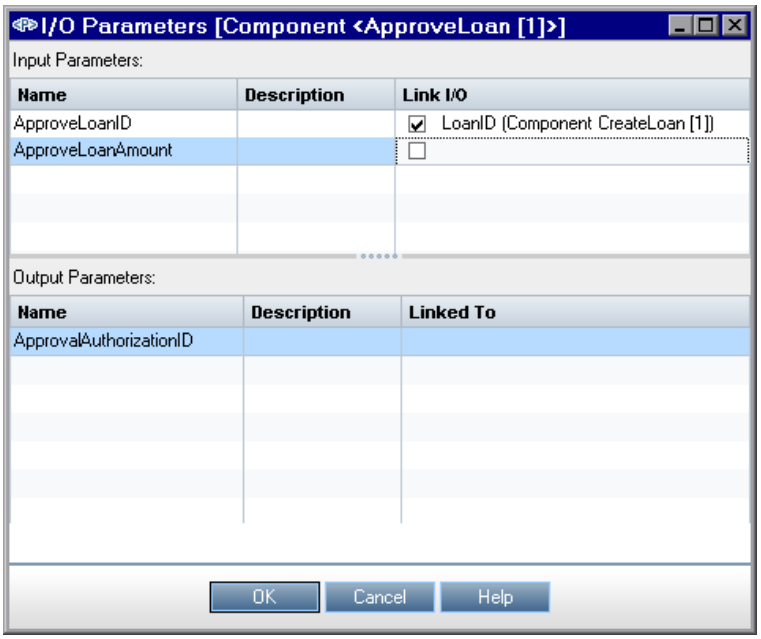
Parameter Linkage User Interface

This section includes:

I/O Parameters Dialog Box 272
Select Output Parameter Dialog Box 273

I/O Parameters Dialog Box

This dialog box enables you to view the parameters defined at the component, flow, and test levels and link target input parameters to source output parameters.



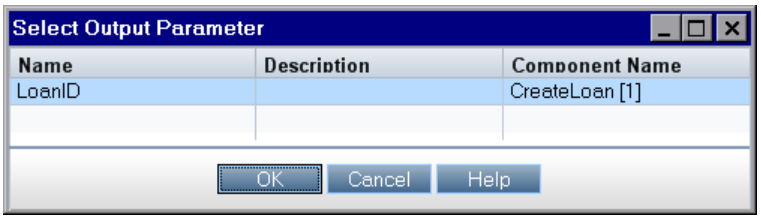
To access	From the Test Script tab containing a component or a flow, click the link in the I/O Parameters column.
Important information	The title of the dialog box differs according to the entity for which you opened the dialog box (component or flow).
Relevant tasks	"How to Link Data" on page 270
See also	"Parameter Linkage Overview " on page 266

User interface elements are described below:

UI Element (A - Z)	Description
Description	Description of the parameter.
Input Parameters	Target input parameters defined for the component or flow.
Link I/O checkbox	Checkboxes for linking the target input parameters to source output parameters.
Linked To	Target input parameters linked to the source output parameter.
Name	Name of the parameter.
Output Parameters	Source output parameters defined for the component or flow.

Select Output Parameter Dialog Box

This dialog box enables you to select the source output parameter to link to a target input parameter.



To access	From the I/O Parameters dialog box, click a checkbox in the Link I/O column.
Important information	• Click the row for the output parameter you want to select, and then click **OK**. • Output parameters from components or flows prior to the current component or flow in the test hierarchy are displayed.
Relevant tasks	"How to Link Data" on page 270
See also	"Parameter Linkage Overview " on page 266

User interface elements are described below:

UI Element (A - Z)	Description
Component Name	Name of the component or flow in which this source output parameter is defined.
Description	Description of the parameter.
Name	Name of the parameter.

Chapter 17

Parameter Promotion

This chapter includes:

Parameter Promotion Overview 276

How to Promote Parameters 276

Parameter Promotion User Interface 278

Parameter Promotion Overview

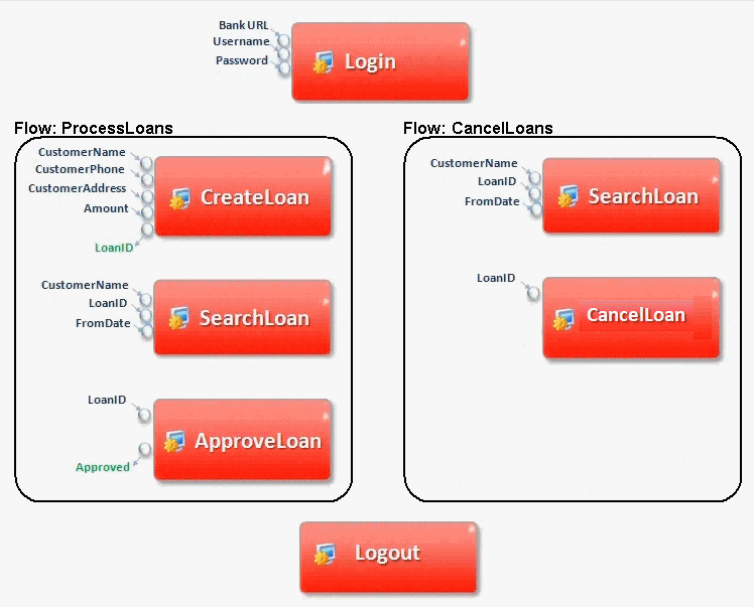
You can promote component parameters to the flow or test level at the same time as you add a component to a flow or test. Similarly, you can promote flow parameters to the test level at the same time as you add a flow to a test.

You can also view the status of promoted parameters.

Example: Parameter Promotion

The three components, CreateLoan, VerifyLoan, and ApproveLoan, process a customer loan request using a parameter called LoanID. The loan ID is stored as a flow parameter and is available to the three components in the flow.

However, the business process needs other flows in the business process test, such as CancelLoan, to access the loan ID as well.



For LoanID to be accessible to the other flows, promote the LoanID parameter to the flow level for each flow, and then promote the LoanID flow parameter to the test level. A new test level parameter called LoanID is created that is accessible to every component and flow in the business process test.

For task information, see "How to Promote Parameters" below.

How to Promote Parameters

This task describes how to create flow-level or test-level parameters by promotion.

Note:

- This task is part of a higher-level task. For details, see "Introducing Data Handling (Parameters)" on page 229.

This task includes the following steps:

- "Prerequisites" below
- "Select a flow or test" below
- "Select components or flows and promote parameters to the next level" below
- "Results" on next page

1. ## Prerequisites

   Make sure that the components or flows that you are adding to the flows and business process tests have parameters defined.

2. ## Select a flow or test

   In the Test Plan module > Test Script tab, select the flow or business process test to which you want to add components or flows. This is the flow or test in which new components are created, based on the parameters promoted from the components or flows.

3. ## Select components or flows and promote parameters to the next level

   a. In the Test Script tab, click the **Select Components** button. The Select Components and Flows pane opens. For user interface details, see "Test Script Tab for Business Process Testing" on page 152.

   b. Select the component or flow that you want to add to the flow or test and contains the parameters you want to promote.

   c. Click the **Quick Add** button in the pane's toolbar and select one of the promotion options.

   d. Depending on the promotion option you select, the Promote Parameters dialog may open. For user interface details, see "Promote Parameters Dialog Box" on next page.

   To add the component or flow using the last-used method for parameter promotion, do one of the following:

   - Double-click the component or the flow.
   - Drag the component or flow to the Script pane.
   - Click the **Quick Add** button in the toolbar without selecting an option.
   - Click the **Quick Add** button's down arrow, and select **Quick Add**.

   Tip: When you promote a flow output parameter from a specific component output parameter, the flow output parameter is automatically named to match the component and flow parameter naming convention. If required, you can change the flow output

parameter name to suit your needs.

4. **Results**

   You can view a list of test and flow parameters, and how they were originally defined, using the Parameters Promotion Status dialog box. For user interface details, see "Test/Flow Parameters Promotion Status Dialog Box" on page 280.

Parameter Promotion User Interface

This section includes:

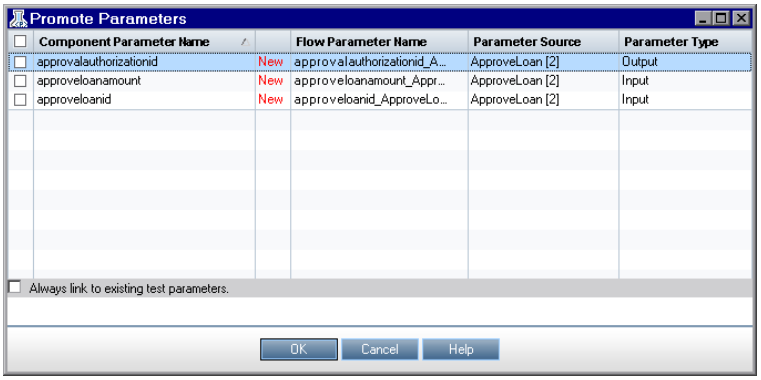
Promote Parameters Dialog Box 278

Test/Flow Parameters Promotion Status Dialog Box 280

Promote Parameters Dialog Box

This dialog box enables you to promote parameters to the next level.

The contents of the dialog box differs for components and flows.



To Access	In the Test Plan module, display the Components and/or Flows panes to the right of the Test Script tab by clicking the Select Components button. Then, select a component or flow that contains parameters in the tree and click **Quick Add** and select **Add while Setting Promote Options**.

Important information	• This dialog box enables you to promote component parameters to the flow or test level, and flow parameters to the test level. • The Promote Parameters dialog box opens if the **Add while Setting Promote Options** menu option was the last menu option chosen when adding components or flows to the Script pane. • The content and text of the Promote Parameters dialog box depends on whether you are adding components to a flow or test, or flows to a test. • Clicking **Cancel** or pressing the Esc key closes the dialog box but still adds the component or flow, without promoting any parameters.
Relevant tasks	"How to Promote Parameters" on page 276
See also	"Parameter Promotion Overview" on page 276

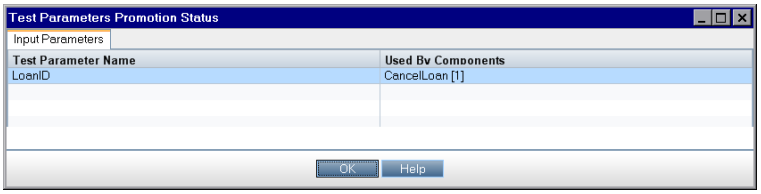
User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
<New indicator column>	If the parameter is first being created as it is being promoted, **New** is displayed in this column.
<Select All checkbox>	Selects, or clears, the listed parameters for promotion when adding a component to a test or flow, or a flow to a test.
<Select checkboxes>	Selects, or clears, the parameter for promotion when adding a component to a test or flow, or a flow to a test.
Cancel	Closes the dialog box but still adds the component or flow, without promoting any parameters.
Component Parameter Name	Displays the parameters of the component or flow.
Parameter Source	Displays the source for the parameter (that is, in which component or flow the parameter was originally defined).
Parameter Type	Displays the type of parameter, **Input** or **Output**.
Test/Flow Parameter Name	Displays the names assigned to the promoted parameters in the current business process test or flow. If the parameter was promoted, a suffix is added to the parameter name indicating the source component.
Always link to existing test parameters	When a parameter with the same name already exists in the test or flow, this UI element determines if ALM uses the existing test parameter during promotion, or creates an additional test parameter. If checked, the **Test/Flow Parameter Name** suffix is removed, because the parameter may have been promoted from multiple components. **Default:** Unchecked.

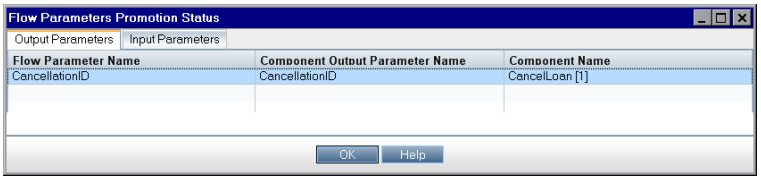
Test/Flow Parameters Promotion Status Dialog Box

This dialog box enables you to view parameters and how the parameters became available to the business process test or flow (definition or promotion).

Test Parameters Promotion Status



Flow Parameters Promotion Status



To access	In the Test Plan module > Test Plan tree, select a business process test or flow, and click the **Test Script** tab. Click **Show Test Parameters Promotion Status** .
Important information	The title bar, structure and fields of the dialog box change depending on the object that was selected in the Test Plan Module: flow or business process test.
Relevant tasks	"How to Promote Parameters" on page 276
See also	"Parameter Promotion Overview" on page 276

User interface elements are described below:

UI Element	Description
Output Parameters tab	Displays the output parameters defined for the flow. **Available from:** The Flow Parameters Promotion Status dialog box
Input Parameters tab	Displays the input parameters defined for either the flow or the business process test.
Test / Flow Parameter Name	Name of the parameter defined for the business process test or flow.

UI Element	Description
Component Output Parameter Name	The name of the corresponding output parameter as defined in the business component. If the flow parameter was created manually (that is, not promoted from a business component), this value appears as N/A (not applicable). **Available from:** The Output Parameters tab of the Flow Parameters Promotion Status dialog box.
Component Name	Name of the component from which the output parameter was promoted. If the flow parameter was created manually (that is, not promoted from a business component), this value appears as N/A (not applicable). **Available from:** The Output Parameters tab of the Flow Parameters Promotion Status dialog box.
Used By Components	Name of the component or flow from which the input parameter was promoted. If the parameter was created manually (that is, not promoted), this value appears as N/A (not applicable). **Available from:** The Input Parameters tab of the Flow Parameters Promotion Status dialog box, and the Test Parameters Promotion Status dialog box.

Chapter 18

Iterations

This chapter includes:

Iterations Overview 284

How to Set Data for Iterations 287

How to Import and Export (Save) Parameter Values 289

Iterations User Interface 291

Iterations Overview

This chapter describes how configure a group, component, flow, or test to run a specified number of iterations during a single run. Each iteration can use different values for parameters.

Note:

- For task details on defining iterations for components, groups, and flows, see "How to Set Data for Iterations" on page 287.
- You can also run business process tests for different use-cases of the application you are testing. Each use-case of a test is represented by a test configuration. You can associate different sets of data for each test configuration and its iterations. For task details, see "How to Work with Business Process Test Configurations" on page 203.
- **Product Feature Movie:** To view a movie that demonstrates how to handle Business Process Testing parameters, select **Help > Product Feature Movies** in the ALM main window.

You can configure how many times, and with which data:

- A business component runs in a test (using the Component Iterations dialog box).
- A flow runs in a test (using the Flow Iterations dialog box or the Iteration Dialog dialog box).
- A group runs in a test (using the Group Iterations dialog box).
- A test instance runs (using the Test Iterations tab).
- A test instance runs using certain test configurations (using the Test Configuration tab > Data tab)

Examples of iteration usage include:

- You can create iterations for a flow that searches for different loans in a test for a banking application by supplying different loan IDs.
- You can create iterations for a test that prepares loans with different interest rates to see which rate is most affordable for the customer.

The number of iterations is set according to the number of data values supplied. For example, a test or flow with ten sets of data runs ten times, and a component with six sets of component parameter values runs six times before proceeding to the next component in the run.

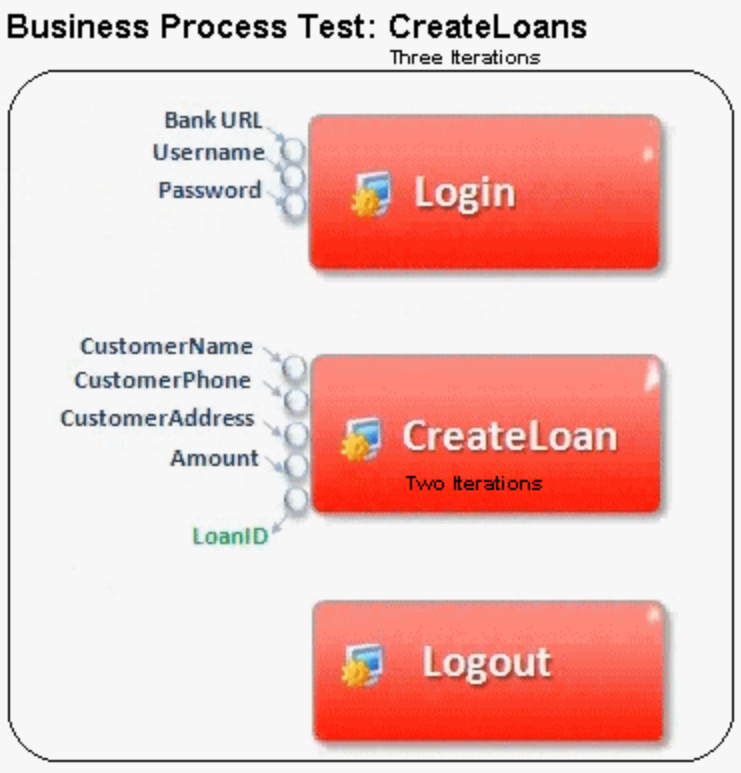
You can also specify a single iteration or a range of iterations to run in a business process test or flow. This enables you to run and check one or more specific iterations without having to run all the defined iterations. Iteration ranges for components, flows, and test configurations that use static data are specified using the Select Iterations dialog box. For user interface details, see "Select Iterations Dialog Box" on page 296.

In addition, you could use the same business process test or flow definition for different scenarios. Different combinations of iterations can be run for different test runs. Iteration ranges for test configurations that use dynamic data are specified using the Data Resource Settings dialog box. For user interface details, see "Data Resource Settings Dialog Box" on page 217.

Note: When running tests and flows with iterations in Change Detection mode, only the first iteration of the selected range of iterations runs. For details about change detection, see "Change Detection and Resolution" on page 319.

Example

Consider the following business process test for a banking application comprising three business components: Login, CreateLoan, and Logout:



As illustrated above:

- The entire business process test is iterated three times.
- Different values for the test parameters BankURL, Username, and Password can be used for each test iteration.
- Within each of the three test iterations, the CreateLoan component is iterated twice. This means that the CreateLoan component iterates a total of six times.
- Different values for the CustomerName, CustomerPhone, CustomerAddress, and the Amount input parameters are used for each iteration of the CreateLoan component. Six different input parameters can be supplied in total.
- The CreateLoan component provides an output value for the LoanID parameter for each iteration (six output values provided in total).

This topic also includes "Group Iterations Overview" on next page.

Group Iterations Overview

In some cases, it may be helpful to iterate several business components together as a group.

Component groups in the Test Script tab are identified by a group node listed above its member components. The group node contains the group icon and displays the number of iterations for the group. The group node can be expanded and collapsed to show or hide its member components.

- The number of iterations indicates the total number of value sets defined for the group.
- The iteration range indicates the subset of iterations currently set for use when the test runs.

Consider a business process test that contains four business components; C1, C2, C3, and C4. The test requires that the components be iterated as follows:

- Component C1 - Two iterations
- Component C2 - Three iterations
- Component C3 - Three iterations
- Component C4 - One iteration

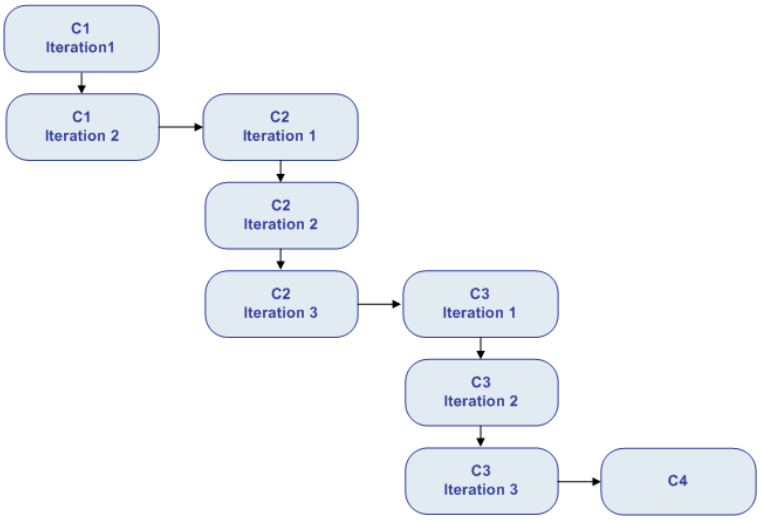
This topic demonstrates iterations for this scenario:

- "Without Grouping" below
- "With Grouping" on next page

Without Grouping

Without grouping, the business process test would run each business component in sequence, C1 and its iterations, then C2 and its iterations, then C3 and its iterations, and then C4.

This is illustrated as follows:

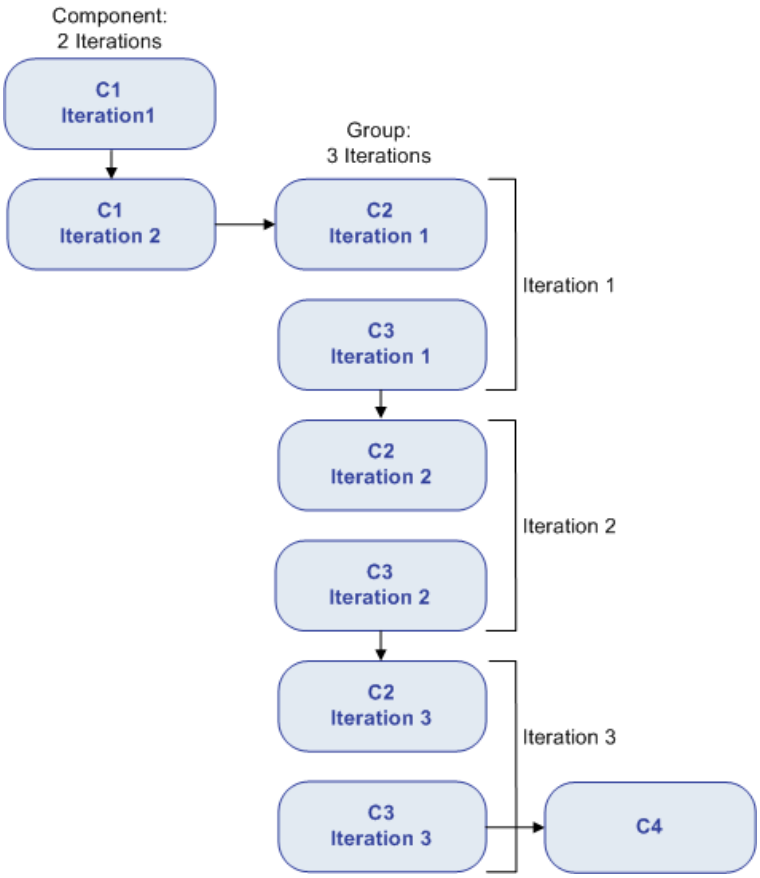


With Grouping

Instead of running all the iterations of component C2 followed by all the iterations of component C3, you could group these two components together, enabling the business process test to run in the following order:

- The first iteration of C1
- The second iteration of C1
- The first iteration of C2
- The first iteration of C3
- The second iteration of C2
- The second iteration of C3
- The third iteration of C2
- The third iteration of C3
- C4

This is illustrated as follows



How to Set Data for Iterations

This task describes how to create iterations and define their corresponding data values.

Note: This task is part of a higher-level task. For details, see "How to Handle Data in Business Process Testing" on page 235.

This task includes the following steps:

- "Prerequisites and considerations" below
- "Iterate components, flows, and groups" below
- "Iterate flow run instances" on next page
- "Iterate test configurations" on next page
- "Iterate according to Imported data - optional" on next page
- "Results" on next page

- ## Prerequisites and considerations

  Take the following guidelines into account to successfully work with iterations:

  - Match an application's post-condition with the next iteration's pre-condition. For a business component to run iterations successfully, it is essential that the post-condition (the state of the application after the last step in the component runs) match the pre-condition (the state of the application before the first step in the component runs).
  - Link output parameters to input parameters. Iterations in a business process test or flow can result in multiple output parameter values. In these cases, each iteration passes its output value as input to the corresponding target component or flow. This is called parameter linkage. For conceptual details, see "Parameter Linkage Overview " on page 266.
  - Components or flows in a group with input parameters must have the same number of iterations. When grouping, ALM requests confirmation that the iteration range should be set to that of the first component's iteration range. For example, if the first component in the group is set to run iterations 2 to 3, the entire group will be set to run iterations 2 to 3.
  - For iterations of a group to be successful, the state of the application at the end of the last item in the group must match the state of the application before the first item in the group. For example, if the first component in the group assumes that the Login dialog box in an application is open, then at the point where the last component of the group ends, the Login dialog box but be in an open state before the next iteration begins.
  - Moving a group or a member within a group could cause a parameter reference conflict, for example, where a group is moved to a position preceding the component that provides an input component parameter needed by a parameter in the group. If the resulting warning message is ignored, the conflicting link to the source parameter is deleted. The value for the source parameter will be empty. You can then either supply a value for the parameter or reinstate the link in the Group Iterations dialog box.

- ## Iterate components, flows, and groups

  You use the Iteration pages to set the parameter values that a component, group, or flow uses for each iteration. For user interface details, see "Iteration Pages" on page 292.

  Set the value of each input parameter for each iteration. For user interface details, see "Set

Values Dialog Box" on page 248.

If you want the group, component, or flow to run for a specific range of iterations, click the **Select Iterations** button and select the relevant iterations in the Select Iterations dialog box. The headings of the iterations that are not in the selected range are dimmed. For user interface details, see "Select Iterations Dialog Box" on page 296.

Note: If a component, group, or flow has input parameters that reference the output parameters of a different business component or flow, differences between the number of iterations can result in an error. The parameter name in the Iterations dialog box is displayed in red, indicating an iteration range mismatch.

- **Iterate flow run instances**

  You use the Test Iterations tab in the Test Lab module to set the parameter values that a flow instance uses for each iteration. For user interface details, see "Iteration Pages" on page 292 and the information about the Test Iterations tab in the *HP Application Lifecycle Management User Guide*.

- **Iterate test configurations**

  You use the Data tab in the Test Configurations tab in the Test Plan module to set the parameter values that a test configuration uses for each iteration. For user interface details, see "Iteration Pages" on page 292 and "Data Tab: Static View" on page 212.

  Note: Iterations for test configurations that access dynamic data are set according to the number of rows of data in external data resource tables, and are not defined using the Iteration pages.

- **Iterate according to Imported data - optional**

  The task of defining lists of component or flow parameter values for iterations can be facilitated by importing the values from an external file. For task details, see "How to Import and Export (Save) Parameter Values" below.

- **Results**

  When the component or flow runs, only the defined iterations run.

  The iterations that are not selected for inclusion in the run are disabled.

How to Import and Export (Save) Parameter Values

This task describes how to create iterations based on lists of component or flow parameter values, and how to save a set of existing parameter values to an external file, for example, to use with other business components or flows.

Note: This task is part of a higher-level task. For details, see "How to Set Data for Iterations" on page 287.

This task includes the following steps:

- "Import data from an external file" below
- "Export (save) data to an external file" on next page

- ## Import data from an external file

  You can import parameter values for component or flow iterations from a **.csv** file.

  When you import a file, all the records in the file become iterations in the business process test or flow.

  Note: The parameter values that you import from a **.csv** file are the values that were defined when the file was last saved before the import. The values in the **.csv** file do not change dynamically according to value changes that may be implemented in ALM. Conversely, the parameter values in ALM do not change dynamically when you make changes to the **.csv** file.

  a. Prepare external file for import

  Parameter values are imported and saved in **.csv** (comma separated value) files. This is a file format in which each line is a record, and each data value field in the record is separated by a comma.

  You can create a **.csv** file manually in a text editor, save a spreadsheet file as a **.csv** file from a program such as Microsoft Excel, or save a set of already defined parameters in a business process test or flow as a **.csv** file.

  Tip: You can enter a few rows in the Component Iterations or Flow Iterations dialog box and then use the **Export** option to create an initial **.csv** file in the proper format. Edit and save the **.csv** file in a text editor or spreadsheet program to enter actual data.

  Before importing a **.csv** file, ensure that it is set up in the proper format, as shown in the following example.

  Example

  The format of a typical **.csv** file for a customer list in a bank application is shown below:

  - The first row, **Version 1.0**, is reserved as a title row. It can contain any text.
  - The second row (**"CustomerName","CustomerPhone","CustomerAddress"**) contains the column headers.
  - The subsequent rows contain the parameter values that are used in iterations of the business process test or flow.

  b. Start the import

  In the Test Plan tree, select the business process test that contains the relevant business component or flow, and in the **Test Script** tab, open its Iterations dialog box. For user interface details, see "Iteration Pages" on page 292.

> **Tip:** Make sure the business component has at least one input parameter.

Click **Import** . The Open dialog box opens.

Select the **.csv** file that contains the required data and click **Open**. The Map Imported Data dialog box opens.

c. Map the data

In the Map Imported data dialog box, expand the components to locate the relevant parameters for mapping. For user interface details, see "Map Imported Data Dialog Box" on page 297 .

After mapping, the values from the **.csv** file are imported and displayed in the Iterations dialog box.

> **Tip:** If the Iterations dialog box already has values defined for one or more iterations, these values are overwritten by the imported values. If the Iterations dialog box has more iterations currently defined than the number of records in the imported file, the surplus iterations in the dialog box are removed.

- **Export (save) data to an external file**

You can save (export) parameter values that you have defined for iterations to a **.csv** file. You can then apply the same values to other component or flow iterations by importing the saved data.

> **Tip:** Saving parameters is also a useful way to create an initial **.csv** file that you can edit in a spreadsheet program, and then re-import to the component or flow.

a. In the Test Plan tree, select the business process test that contains the relevant business component or flow, and in the **Test Script** tab, open its Iterations dialog box. For user interface details, see "Iteration Pages" on next page.

b. To save iteration values, click the **Export** button. The Save CSV (Comma Delimited) File dialog box opens.

Enter a descriptive name in the **File Name** box and click **Save**. The component or flow parameter values are saved in a **.csv** file. The **.csv** file is saved in a comma delimited table format.

Each column in the table represents the list of values for a single parameter. The first row of the table is a title row. The second row contains the parameter names (column headers). Each subsequent row in a column represents a value for the displayed parameter during a single iteration of the business process test or flow.

Iterations User Interface

This section includes:

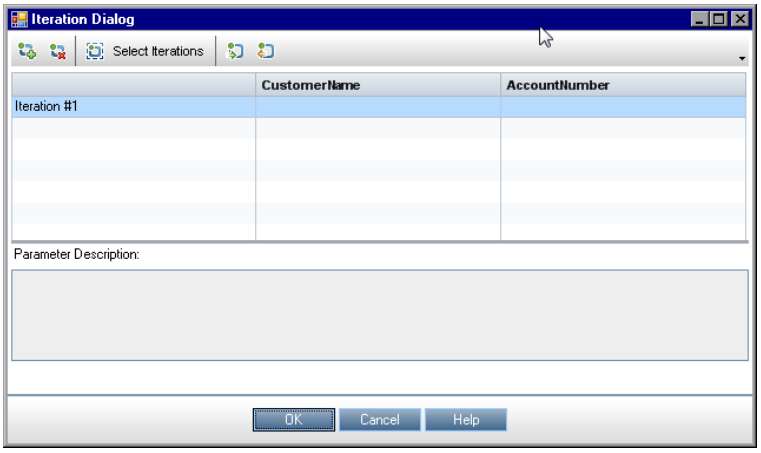
Iteration Pages 292

Select Iterations Dialog Box 296

Map Imported Data Dialog Box 297
Create Flow/Test Parameters Dialog Box 298

Iteration Pages

These pages enable you to set parameter values for iterations of components, groups, flows, test configurations, and test run instances.



To access	Do one of the following. • **To iterate components, groups, and flows:** From the Test Plan module > Test Script tab, click **Iterations** in the Iterations column. • **To iterate flows:** From the Test Lab module > Execution Grid tab, click **Iterations** or **Undefined** in the Iterations column. **Tip:** By default, the Iterations column is not displayed in the Execution Grid tab. For details on how to display the column, see the *HP Application Lifecycle Management User Guide*. • **To iterate test configurations that access static data:** From the Test Plan module > Test Configurations tab, click the **Data** tab. • **To iterate test instances:** From the Test Lab module > Test Instance Details—Execution Settings View, click the **Test Iterations** tab.
Important information	• Depending on context, iteration pages are displayed in different ways with different title bars, according to the entity for which you opened the page. • You can expand and collapse entities in Iterations dialog boxes for groups, flows, and components from the Test Plan module. • Iterations for test configurations that access dynamic data are set according to the number of rows of data in external data resource tables, and are not defined using the Iteration pages.

Relevant tasks	• "How to Set Data for Iterations" on page 287 • "How to Work with Business Process Test Configurations" on page 203
See also	• "Iterations Overview" on page 284 • "Group Iterations Overview" on page 286 • *HP Application Lifecycle Management User Guide*

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

UI Element (A-Z)	Description
	Multiple Iterations View / Single Iteration View. Enables you to toggle between: • A view that shows all iterations in one dialog box. • A view that shows only one iteration at a time in the dialog box. **Available from:** Iterations dialog boxes in the Test Plan module
	Previous / Next Iteration. Displays the next or previous iteration in the dialog box. **Available from:** Iterations dialog boxes in the Test Plan module
	Iteration. Displays the iteration currently in focus in the dialog box, and enables you to select a different iteration on which to focus. **Not available from:** Iterations dialog boxes in the Test Plan module
	Add Iteration. Adds an iteration for the entity (component, group, test, or flow). The following options may be available, depending on the context in which you opened the Iteration page. If these options are not available, iterations are added by default without setting parameter values. • **Add New Iteration.** Adds a new iteration without setting parameter values. • **Copy Iteration.** Adds a new iteration by copying the values of a selected iteration's parameters. • **Create Iteration with Default Values.** Adds a new iteration and sets the parameter values to the default values. **Disabled when:** The component or flow for which you are adding an iteration is in a group.
	Remove/Delete Iteration. Deletes the selected iteration from the entity (component, group, test, or flow). **Disabled when:** The component or flow for which you are adding an iteration is in a group.

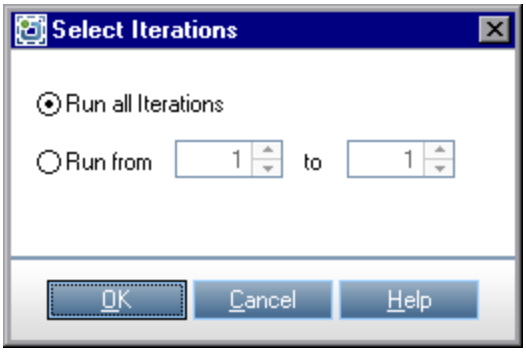
UI Element (A-Z)	Description
	Select Iterations. Opens the Select Iterations dialog box to select the iterations you want to run during the test run. You can specify one, all, or a range of iterations. **Disabled when:** • The component or flow for which you are selecting iterations is in a group • There is only one iteration.
	Find. Enables you to find a parameter value, a parameter, or a component. For details on Find and Replace functionality, see the *HP Application Lifecycle Management User Guide*. **Available from:** Iterations dialog boxes
	Import. Enables you to import data values from a .csv (comma separated value) file into the Iteration page. An iteration is created for each row in the .csv file, and the test or flow runs using the data specified for each iteration. **Tip:** Once the data values are imported, there is no longer any connection between the .csv file and the iteration values (meaning,changes you make to the data in the .csv file are not reflected in the Iteration page). To maintain data dynamically in an external file, and have the data be updated automatically in the Iteration page, create test configurations. For task details, see "How to Work with Configurations and Dynamic Data" on page 205.
	Export. Enables you to export data values to a .csv (comma separated value) file from the iteration data values specified in the Iteration page. **Examples** • Export, format, and print the data for use while testing manually. • Export the data, modify in a preferred editor , and then import the data back into the Iteration page.

UI Element (A-Z)	Description
	Restore Instances. Resets iteration parameter values for test instances based on the selected test configuration for static data. Test instance parameter values are reset to the values entered in the Test Configurations tab > Data tab. **Available from:** Test Plan module > Test Configurations tab > Data tab for static configurations **Tip:** Instead of restoring the original test configuration iteration values, you can also generate a new test configuration based on test instance iteration values. For task details, see "How to Generate Test Configurations from Test Instances" on page 209.
	Create Flow/Test Parameters. Opens the Create Flow/Test Parameters dialog box, which enables you to generate flow or test parameters from business component input parameters or flow input parameters. For user interface details, see "Create Flow/Test Parameters Dialog Box" on page 298.
Expand All	Expands all nodes. **Available from:** <right-click menu>, for Iteration dialog boxes
Collapse	Collapses the nodes for the selected entity. **Available from:** <right-click menu>, for Iteration dialog boxes
<value columns>	Displays the value for each parameter in the displayed iteration. **Available from**: Single Iterations view, the Test Iterations tab, Test Configurations Data tab,and the Iteration Dialog dialog box
<values>	The value of the input parameter when the instance runs. The value can be entered in the grid directly or in the Set Value page (accessed by clicking the arrow in the relevant cell). For task details on specifying default values for parameters, see "Set Values Dialog Box" on page 248. If no value is specified, the default value for that parameter is used. If no default value is specified, no value is displayed and the entity may not run correctly. Values are saved and compared as strings, but can be specified according to different value types. Test or flow parameters (parameters whose values are to be taken from the business process test or flow), are shown in curly brackets **{ }**. This instructs ALM to treat the parameter as a parameter and not as a fixed value.

UI Element (A-Z)	Description
Description area	The description for the parameter and its original source entity (business component, test, or flow).
Parameter Description area	Parameter descriptions are initially entered in the module in which they were created (the Business Components module or the Test Plan module), or in UFT for automated components.
Iteration # columns/rows	Displays the current values for each parameter in each iteration. **Available from:** Multiple Iterations view and the Test Iterations tab
Parameter column	Names of the parameters, grouped by entity (business component, group, test, or flow). **Available from:** Iterations dialog boxes

Select Iterations Dialog Box

This dialog box enables you to specify a range of iterations to run for a business process test or flow.



To access	Perform one of the following: • For business components and flows: In any Iterations dialog box toolbar, click **Select Iterations** . • For business process tests that use configurations that access static data: a. In the Test Plan module, select the business process test. b. Click the **Test** tab. c. Click the **Data** tab near the bottom of the window. d. In the Data tab toolbar, click **Select Iterations** .
Important information	You can also set iteration ranges using the Data Resource Settings dialog box. For user interface details, see "Data Resource Settings Dialog Box" on page 217.

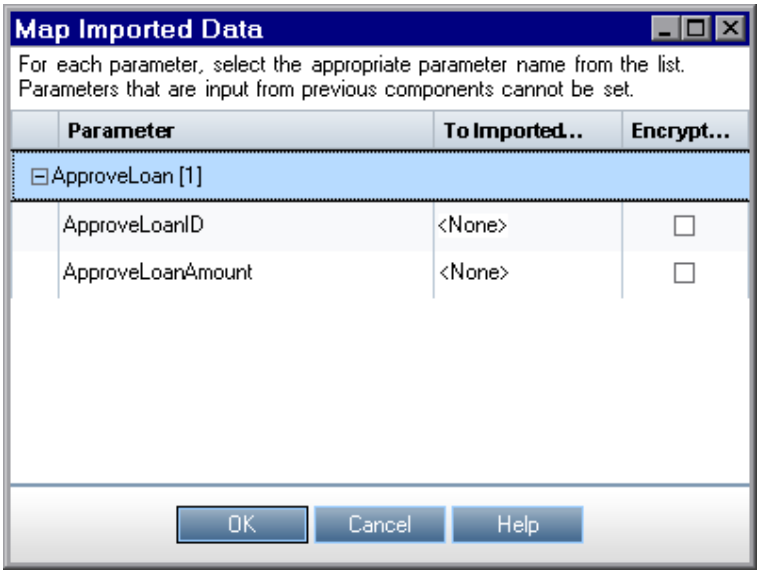
Relevant tasks	"How to Set Data for Iterations" on page 287
See also	"Iterations Overview" on page 284

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

UI Element (A- Z)	Description
Run all iterations	Runs all iterations.
Run from <start> to <end>	Runs a range of iterations, from the iteration specified in **<start>** through the iteration specified in **<end>**. **Tip:** To run one specific iteration only, enter the same iteration number in both boxes.

Map Imported Data Dialog Box

This dialog box enables you to associate, or map, each component or flow parameter to a column header of an imported **.csv** file.



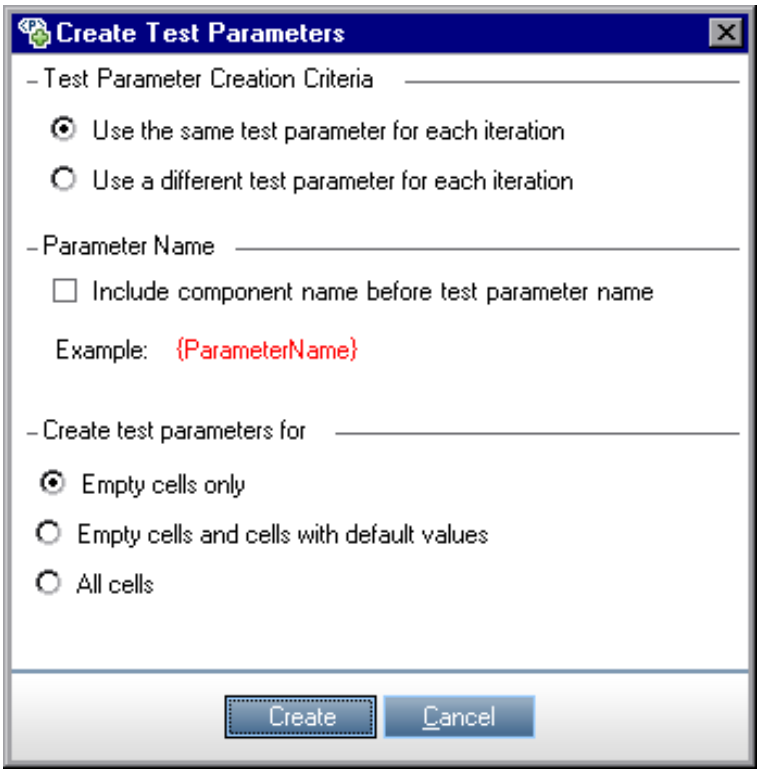
To access	From any iteration page, click **Import** to open the **Open** dialog box. Open the **.csv** file that contains the required data and click **Open**.
Important information	The values in the column corresponding to the selected header become the iteration values used for that component or flow parameter.
Relevant tasks	"How to Import and Export (Save) Parameter Values" on page 289
See also	"How to Set Data for Iterations" on page 287

User interface elements are described below:

UI Element	Description
Parameter	Displays the name of the parameter as defined in Business Process Testing.
To Imported Parameter	Displays the name of the parameter in the external file to which you want to map. **Note:** When the Map Imported Data dialog box opens, it automatically maps any column headers in the **.csv** file to the corresponding set of parameters if the column headers are identical. If no identical column header exists, the value **<None>** is displayed for that parameter.
Encrypt Values	Encrypts the values.

Create Flow/Test Parameters Dialog Box

This dialog box enables you to define criteria for creating, naming, and generating flow and test parameters.



To access	From the **Test Plan** module > **Test Script** tab, select the business process test that contains the relevant business component or flow, or select the flow that contains the relevant business component. Click the **Iterations** link for the component or flow whose input parameters you want to use as a basis for generating test input parameters. In the Iteration page, click the **Create Flow/Test Parameters** button.
Important information	• While defining iterations for business components or flows, you can generate test parameters to use as values for component or flow parameters. • While defining iterations for business components, you can also generate flow parameters to use as values for component parameters. • You can also generate flow/test parameters automatically from the Group Iterations dialog box, based on input parameters for components within component groups. You generate flow input parameters from the Group Iterations dialog box as you would from the Component Iterations dialog box. • The business component or flow must already have at least one parameter defined. • You can generate flow/test input parameters based on existing input parameters for a component or flow. When the component or flow runs, its parameters take the values assigned to the corresponding test or flow parameters.
Relevant tasks	"How to Set Data for Iterations" on page 287
See also	"Iterations Overview" on page 284

User interface elements are described below:

UI Element	Description
Flow/Test Parameter Creation Criteria	• **Use the same flow/test parameter for each iteration.** Clears the flow or test parameter values between iterations so that the same parameter names are used for every iteration, but the values are not passed from one iteration to the next. • **Use a different flow/test parameter for each iteration.** Automatically creates a new flow or test parameter for each iteration. The parameter values are saved with each iteration run.
Parameter Name	**Include component name before flow/test parameter name.** If checked, adds the component or flow name as a prefix preceding the generated flow/test parameter name. **Example**. Provides an example of the generated parameter name based on the selected criteria.
Create flow/test parameters for	• **Empty cells only.** Creates flow/test parameters only for cells that do not contain any value (default). • **Empty cells and cells with default values.** Creates flow/test parameters for all cells that either do not contain a value or contain the default value defined in the component or flow parameter. • **All cells.** Creates flow/test parameters for all cells in the Component Iterations, Group Iterations, or Flow Iterations dialog boxes.
Create	Creates flow or test parameters based on the specified criteria, and adds iterations as necessary.

Test Runs

Chapter 19

Introducing Test and Flow Execution

This chapter includes:

Test and Flow Execution Overview 304

How to Run Business Process Tests and Flows Manually 304

How to Run Automated Business Process Tests and Flows 305

Run Progress Pane Hierarchy 308

Test and Flow Execution User Interface 310

Test and Flow Execution Overview

You run business process tests and flows, and view run results, in the:

- Test Plan module, typically when designing your tests.
- Test Lab module, when you are ready to run a complete business process test or flow, or to run it as part of a larger test set. From the Test Lab module, you can view the results of the test run. These results include the steps in each business component, the actual value of each component parameter, and the results of individual steps.

Business process tests and flows can contain manual and automated components:

- **Manual runs**. When manual tests are ready (or, before automated parts of the test are ready), you can perform a manual run of the business process test or flow. For task details, see "How to Run Business Process Tests and Flows Manually" below.
- **Automated runs**. Automated components (such as those automated by UFT) can be used in the same business process test. The appropriate application is launched to run the components. For task details, see "How to Run Automated Business Process Tests and Flows" on next page.

Tip:

If automating business components, you can check for problems arising from the combination and order of components in a business process test or flow by running the test in Debug mode. You can also check for syntax or logic errors in specific business components by running them individually in UFT or another testing tool. For task details, see "How to Debug Tests and Flows Containing Automated Components" on page 222.

How to Run Business Process Tests and Flows Manually

This task describes how to run business process tests and flows with one or more manual components.

This task is part of a higher-level task. For details, see "Business Process Testing Workflow" on page 28.

This task includes the following steps:

- "Add iterations to business process tests and flows - optional" below
- "Use the Manual Runner or Sprinter to run the manual tests" on next page
- "View and analyze test results" on next page

1. ### Add iterations to business process tests and flows - optional

   In the Test Lab module, you can determine how many times a business process test runs

(iterates) and with what data. For task details on iterating tests, see "How to Set Data for Iterations" on page 287.

> **Tip:** Parameter values can also be provided using the Test Configurations tab in the Test Plan module.

2. **Use the Manual Runner or Sprinter to run the manual tests**

   Run your tests and flows using Sprinter or the Manual Runner.

   For general task details on running tests manually, see the *HP Application Lifecycle Management User Guide*.

   For user interface details on the Manual Runner for Business Process Testing, see "Manual Runner Wizard for Business Process Testing" on page 310.

   > **Note:** Business Process Testing Enterprise Extension users can run tests and flows in change detection mode, to see if changes were made to the enterprise application since the test or flow was created/last modified. Tests and flows that are run in Change Detection mode run only the first iteration of the selected range of iterations of any component, flow, or test, even if multiple iterations are defined. For details about detecting changes, see "How to Run Tests in Change Detection Mode" on page 321.

3. **View and analyze test results**

   You can view the results of the latest test run in the Test Lab module in the Last Run Report tab at the bottom of the Execution Grid. This pane lists each step of all the business components in the test.

   For user interface details on the Last Run Report tab, see the *HP Application Lifecycle Management User Guide*.

   > **Note:** When a test or flow run ends, the parameters displayed in Last Run Report show the date used in the run and not the dynamic date value. Dynamic dates are converted to the actual dates they represent.

You can view details about the automatic test execution after execution is complete. In the Automatic Runner dialog box, select **Run > View Execution Log**. For user interface details, see the *HP Application Lifecycle Management User Guide*.

How to Run Automated Business Process Tests and Flows

This task describes how to run automated business process tests and flows.

This task is part of a higher-level task. For details, see "Business Process Testing Workflow" on page 28.

This task includes the following steps:

- "Prerequisites" below
- "Debug tests and flows - optional" below
- "Add iterations to business process tests and flows - optional" below
- "Determine if tests and flows should be run in change detection mode" on next page
- "Use the Automatic Runner to run the tests" on next page
- "Resolve detected changes" on next page
- "View run results" on next page

1. ## Prerequisites

   - Before running a test within a test set, you can specify the values for any test parameters defined for the test. If you do not specify values, and valid default values have not been specified, the test run may fail.
   - Before you run an automated test on a host computer, ensure that the relevant testing tool on the host allows ALM to run tests.

   a. To enable ALM access to a specific UFT client, open UFT on the host computer, select **Tools > Options**, click the **Run** node, and ensure that **Allow other HP products to run tests and components** is selected in the Run pane.

   b. To run a UFT component from ALM, you must have the necessary ALM permissions. For details, see the *HP Application Lifecycle Management Administrator Guide*.

   c. When you run a UFT business process test from ALM, the test run may be influenced by UFT Remote Agent settings on the UFT computer. For details on the UFT Remote Agent, see the *HP Unified Functional Testing User Guide*.

   d. For improved performance when running tests or flows that contain only UFT automated components, ALM creates and runs a Business Process Testing wrapper test in UFT. The Business Process Testing wrapper test is recreated each time the test or flow runs. By default, the wrapper test is not saved with the run.

      For auditing purposes, you can instruct ALM to save the Business Process Testing wrapper test as an attachment to the test or flow run. In Site Administration, add the **BPT_WRAPPER_TEST_AUDIT** parameter to the list of parameters and set the value to **Y**. For details on setting parameters, see the *HP Application Lifecycle Management Administrator Guide*.

      To view the contents of a Business Process Testing wrapper test, from the Test Lab module > Test Runs tab, double-click the attachment icon next to the business process test or flow run in the grid.

2. ## Debug tests and flows - optional

   Debug automated tests and flows to make sure that they run properly. For task details on debugging, see "How to Debug Tests and Flows Containing Automated Components" on page 222.

3. ## Add iterations to business process tests and flows - optional

In the Test Lab module, you can determine how many times a business process test runs (iterates) and with what data. For task details on iterating tests, see "How to Set Data for Iterations" on page 287.

Tip: Parameter values can also be provided using the Test Configurations tab in the Test Plan module.

4. ## Determine if tests and flows should be run in change detection mode

If you have Business Process Testing Enterprise Extension enabled, you can run automated tests containing learned flows in change detection mode. This enables you to determine if the enterprise application being tested has been modified. For task details on detecting changes, see "How to Run Tests in Change Detection Mode" on page 321.

Tests and flows that are run in Change Detection mode run only the first iteration of the selected range of iterations of any component, flow, or test, even if multiple iterations are defined.

5. ## Use the Automatic Runner to run the tests

You can run manual and automated tests automatically on your machine or on remote hosts using the Automatic Runner.

In the Test Lab module > Test Sets tab, use one of the following:

- Select a test set that includes one or more automated tests and click the **Run Test Set** button.
- To run selected tests, select a test set and click the **Execution Grid** tab or **Execution Flow** tab. Select one or more tests and click the **Run** button.
- To run only manual tests, select the tests and click the **Run** button. In the Manual Test Run dialog box, select **Automatic Runner** and click **OK**.

Note: Dynamic dates are converted to the actual dates they represent.

For user interface details on the Automatic Runner, see the *HP Application Lifecycle Management User Guide*.

6. ## Resolve detected changes

If you have Business Process Testing Enterprise Extension enabled, and you ran the business process test or flow in change detection mode, you can resolve changes detected in the enterprise application during the run. For task details on resolving changes, see "How to Resolve Detected Changes" on page 323.

7. ## View run results

After you run the test or flow from the Test Lab module, you can view the results in various formats:

- **View results in the Test Lab module.** You can view the results of the latest test run in the Test Lab module > Execution Grid > Last Run Report tab. This pane lists each step of all the business components in the test, and, where applicable, the summarized results of main events reported by the testing tool.

  For user interface details on the Last Run Report tab, see the *HP Application Lifecycle Management User Guide*.

  When a test or flow run ends, the parameters displayed in Last Run Report show the date used in the run and not the dynamic date value.

- **View the execution log.** You can view details about the automatic test execution after execution is complete in the Automatic Runner dialog box > Run > View Execution Log. For user interface details, see the *HP Application Lifecycle Management User Guide*.

- **View the contents of a BPT wrapper test.** From the Test Lab module > Test Runs tab, double-click the attachment icon next to the business process test or flow run in the grid.

- **View UFT reports.** If you have Unified Functional Testing or the Unified Functional Testing Add-in for ALM installed on your computer, you can view a complete UFT report. The hierarchical report contains details on all the steps for all the different iterations and business components within the automated test run.
  From the Test Lab module > Execution Grid tab > Last Run Report tab, click the link to the report about each iteration in the **Status** column of the pane. The Run Results Viewer opens.

  For details on viewing and analyzing the results of tests in the Run Results Viewer, see the *HP Unified Functional Testing User Guide*.

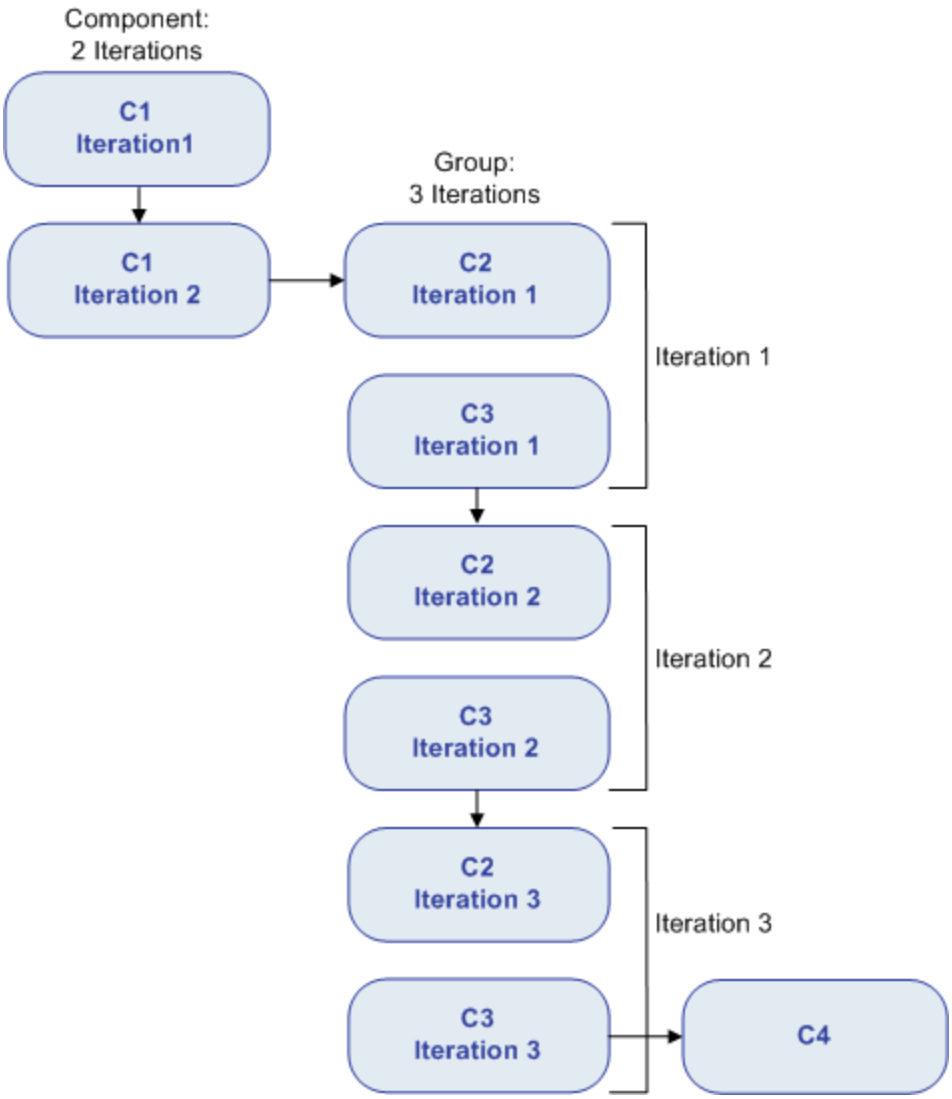
Run Progress Pane Hierarchy

The Run Progress pane hierarchy enables you to manually perform each of the individual component steps in the correct testing sequence.

For example, consider a business process test that contains four business components; Component1 (C1), Component2 (C2), Component3 (C3), and Component4 (C4). The test requires that components C2 and C3 be grouped and that the components and group are then run and iterated as follows:

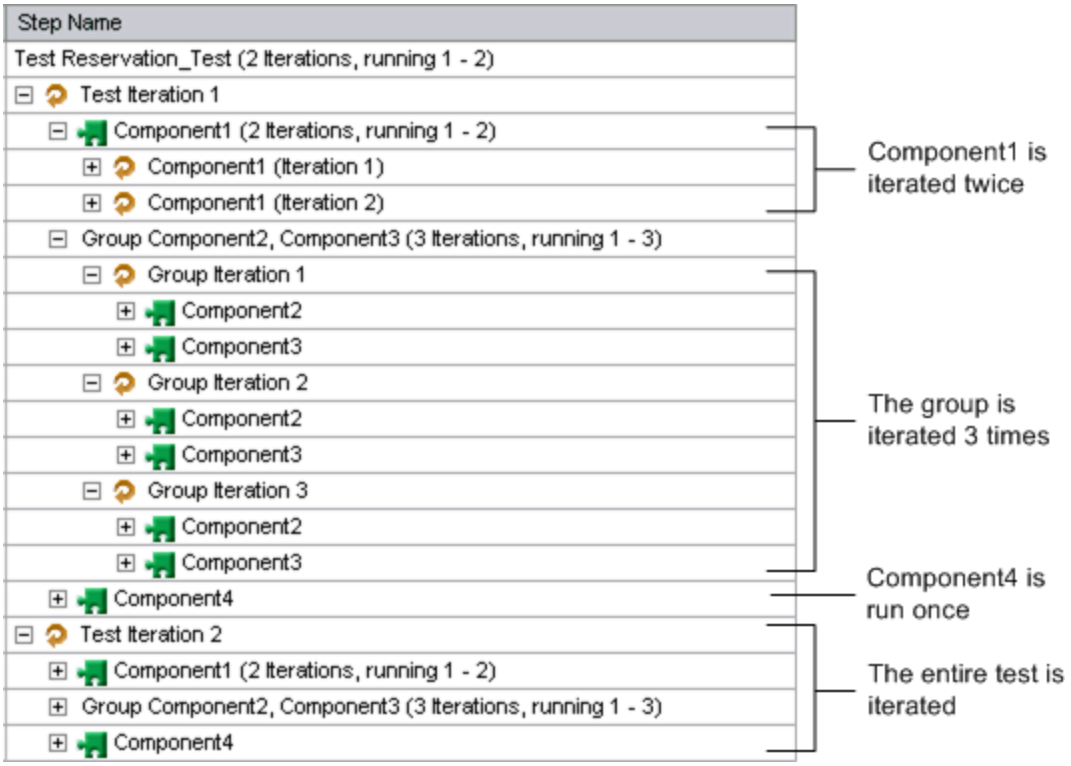
- the first iteration of C1, then the second iteration of C1, followed by
- the first iteration of the group comprising C2 and C3, followed by
- the second iteration of the group, followed by
- the third iteration of the group, followed by the component C4.

This is illustrated as follows:



In addition, the entire test is iterated twice.

The test is displayed in the Manual Runner as follows:



For clarity, the components in the tree have not been expanded to display their individual steps.

Test and Flow Execution User Interface

This section includes:

Manual Runner Wizard for Business Process Testing 310

Test Properties Dialog Box 317

Manual Runner Wizard for Business Process Testing

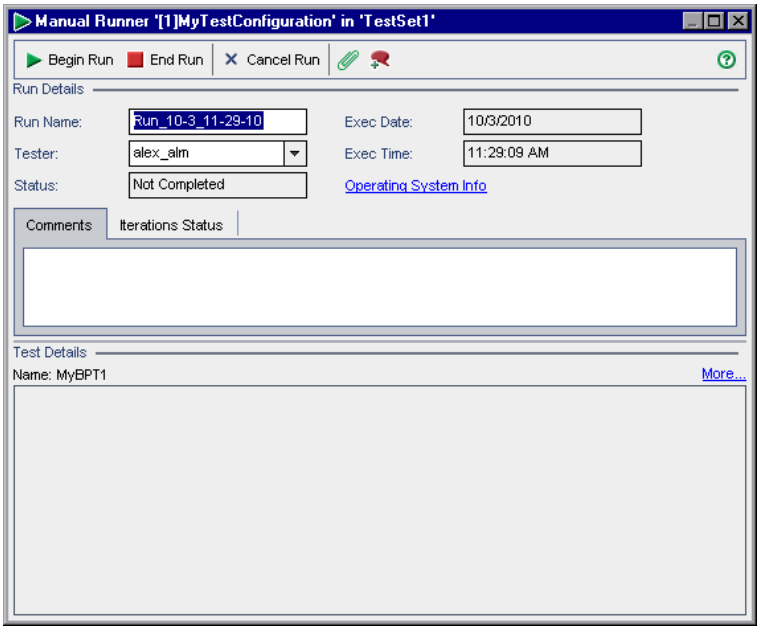
This wizard enables you to run business process tests and flows manually. You can run both manual and automated business process tests and flows manually.

To access	In the Test Lab module > Test Sets tab, select a test set that contains business process tests, and click the **Execution Grid** tab or **Execution Flow** tab. Select one or more business process tests and select **Tests > Run with Manual Runner**.
Relevant tasks	"How to Run Business Process Tests and Flows Manually" on page 304
Wizard map	This wizard contains: "Manual Runner: Run Details Page for Business Process Testing" on next page> "Manual Runner: Step Details Page for Business Process Testing" on page 313

See also	"Test and Flow Execution Overview" on page 304 *HP Application Lifecycle Management User Guide*

Manual Runner: Run Details Page for Business Process Testing

This wizard page enables you to review run details and run business process tests and flows manually.



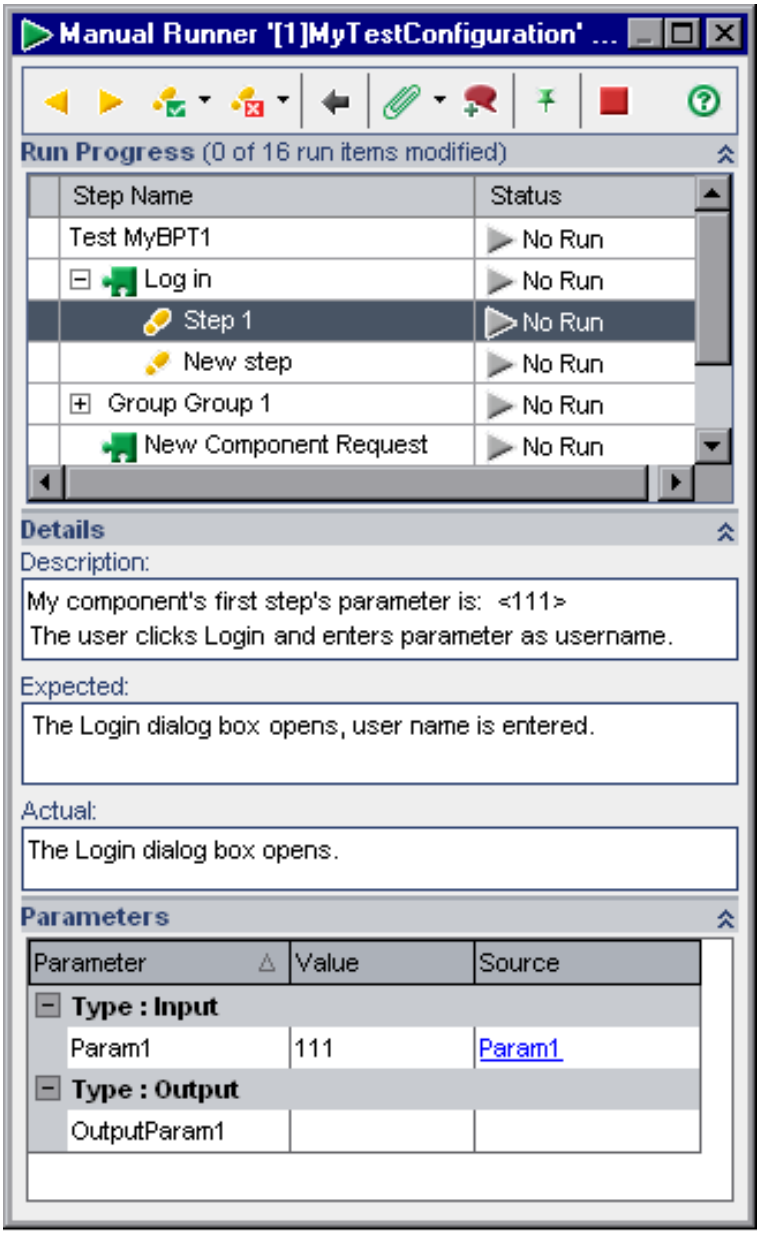
Important information	• General information about this wizard is available here: "Manual Runner Wizard for Business Process Testing" on previous page. • To advance to the next page of this wizard, click Begin Run. • The Run Details page for Business Process Testing differs from the Run Details page for other ALM test types. This section describes the elements that are significant specifically when running business process tests and flows. For user interface details on the Run Details page for other test types, see the *HP Application Lifecycle Management User Guide*.
Wizard map	The "Manual Runner Wizard for Business Process Testing" on previous page contains: "Manual Runner: Run Details Page for Business Process Testing" above > "Manual Runner: Step Details Page for Business Process Testing" on page 313
See also	"Test and Flow Execution Overview" on page 304

User interface elements unique to Business Process Testing are described below:

UI Element	Description
Operating System Info	Opens the Operating System Information dialog box. You can edit the following: • **Operating System.** The operating system on the machine where you perform your test steps. By default, the operating system on your ALM client machine is displayed. • **OS Service Pack.** The operating system service pack. • **OS Build Number.** The operating system build number.
Iteration Status tab	Contains comments about the iterations. Click in the text box to display a toolbar for formatting and spell checking the text.
Test Details pane	Provides the description of the business process test or flow as defined in the Test Plan module.
More	Opens the Test Properties dialog box, enabling you to view details about the test. For more details, see "Test Properties Dialog Box" on page 317.

Manual Runner: Step Details Page for Business Process Testing

This wizard page enables you to set the pass/fail status for each component step, and add, delete, or modify component steps during the run. After you finish executing the business process test or flow, you can save your component steps with the changes you make.



To access	From the Manual Runner: Run Details page, click Begin Run.

Important information	• General information about this wizard is available here: "Manual Runner Wizard for Business Process Testing" on page 310. • The Step Details page for Business Process Testing differs from the Step Details page for other ALM test types. This section describes the elements that are significant specifically when running business process tests and flows. For user interface details on the Run Details page for other test types, see the *HP Application Lifecycle Management User Guide*.
Wizard map	The "Manual Runner Wizard for Business Process Testing" on page 310 contains: "Manual Runner: Run Details Page for Business Process Testing" on page 311 > "Manual Runner: Step Details Page for Business Process Testing" on previous page
See also	"Test and Flow Execution Overview" on page 304

Step Details Page Buttons

User interface elements unique to Business Process Testing are described below:

UI Element (A-Z)	Description
	Previous Step. Jumps to the previous step.
	Next Step. Jumps to the next step.
	Return to previously clicked link. Jumps back to the step containing the input parameter link you clicked previously

Run Progress Pane

This pane displays a hierarchical tree of each test or flow iteration, business component, and component iteration contained in the business process test or flow.

See also	"Run Progress Pane Hierarchy " on page 308

User interface elements unique to Business Process Testing are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
	Show/Hide. Shows/hides the pane.
Step Name	The name of the step. Select a tree item to display its description, parameters and values, and expected result in the other panes.

UI Element	Description
Status	Indicates the run status of each item in the tree. In general, if a run item fails, all of that item's parent items fail. Similarly, if all the items of a particular type pass, their immediate parent also passes. For example: • If one or more steps receive a **Failed** or **Not Completed** status, all of its parent nodes (component iteration, component, flow, and test) also receive a **Failed** or **Not Completed** status. • If all the steps pass in a component iteration, then their immediate parent (the component iteration) receives a **Passed** status. • If all the iterations for a component receive a **Passed** status, the component receives a **Passed** status. • If all the components within a test receive a **Passed** status, the test receives a **Passed** status.
<number of run items/total number of run items>	The number of run items modified during the current manual run, and the total number of run items in the test, are shown next to the Run Progress pane title.

Details Pane

User interface elements unique to Business Process Testing are described below:

UI Element (A-Z)	Description
	Show/Hide. Shows/hides a pane.
Description	Provides the textual description of the selected test, flow, component, or step. This is the text originally entered in the Business Components or Test Plan modules.
Expected	Indicates the expected result of running the selected component. This is the text originally entered in the Steps tab of the component in the Business Components module. Input and output parameters within the text in the Description and Expected boxes are indicated by sets of angle brackets: Input parameters. Input parameters are enclosed in one set of angle brackets. If a value has been assigned to the parameter, the value is enclosed in the angle brackets, for example, **<Bob Smith>**, for the AgentName parameter. If no value has been assigned to the parameter, an empty set of angle brackets is shown, for example, <>. Output parameters. Output parameter names are enclosed in three sets of angle brackets, for example, <<<Order_Number>>>.
Actual	Indicates the actual result of running the step during the test or flow.

Parameters Pane

This pane displays the input parameter values for you to use when manually running the component selected in the run tree. The Parameters pane also enables you to enter values output by the application that you can use later in the test or flow run.

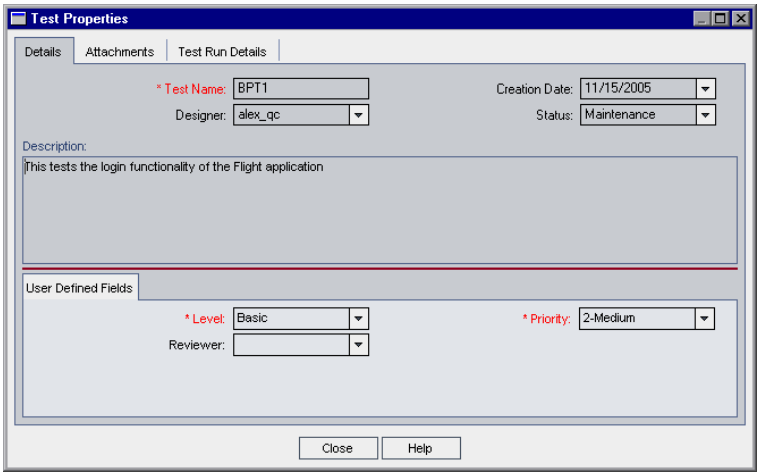
Important information	• Default values can be defined for input parameters in the Parameters tab. Default values are used if no other value is provided during the run. For details, see "How to Create Parameters" on page 238. • The output parameter must already have been created in the component Parameters tab in the Business Components module. In addition, the output parameter must have been linked to the input parameter in the Component Iterations dialog box. For details, see "How to Create Parameters" on page 238.

User interface elements are described below:

UI Element	Description
⇈ ⇊	**Show/Hide.** Shows/hides the pane.
Type: Input	**Parameter.** The names of the input parameters defined for the component selected in the Run Progress Pane tree. For details, see How to Define Iterations to Run with Different Values. **Value**. Displays the input parameter values defined for the selected component. **Source.** Enables you to jump to the component or iteration that provides the parameter value. You can click the **Return to previously clicked link** button to jump back to the step containing the link. **Note:** Input parameters that are defined as the default values do not have any associated link.
Type: Output	**Parameter.** The names of the output parameters defined for the component selected in the Run Progress Pane tree. For details, see How to Define Iterations to Run with Different Values. **Value.** Enables you to enter a value provided by a step in the application you are testing. You can then use this value as an input value in a subsequent step. Input parameters derived from the output parameter of a previous step are shown as links in the **Source** cells in the **Type: Input** section. The name of the link indicates the name of the source component and the name of the relevant output parameter, for example, `[Component1] Order_Number`. To retrieve the output parameter value, you can click the link to jump to the step that provides the value. You can click the **Return to previously clicked link** button to jump back to the step containing the link.

Test Properties Dialog Box

This dialog box enables you to view details about the test.



To access	From the Run Details page of the Manual Runner, click **More**.
Important information	• Most of the fields on this page are read-only. • You can send the test properties by email to the designer of the business process test or flow.
Relevant tasks	"How to Run Business Process Tests and Flows Manually" on page 304
See also	• "Test and Flow Execution Overview" on page 304 • "Manual Runner Wizard for Business Process Testing" on page 310

User interface elements are described below:

UI Element	Description
Details tab	Lists the business process test or flow fields. • For details on Business Process Testing fields, see "Test Plan Module Fields for Business Process Testing" on page 150. • For other fields common to most test types, see *HP Application Lifecycle Management User Guide*.
Attachments tab	Enables you to open and view attachments for the business process test or flow.
Test Run Details tab	Displays details about the business process test or flow run. **Available when**: Run details exist.
User Defined Fields tab	Displays user-defined fields. **Available when**: User-defined fields exist.
Description tab	Displays a description of the business process test's or flow's purpose, or other descriptive text.
Comments tab	Displays additional comments for the business process test or flow instance.

Chapter 20

Change Detection and Resolution

This chapter includes:

Change Detection and Resolution Overview 320

How to Run Tests in Change Detection Mode 321

How to Resolve Detected Changes 323

Detection and Resolution Options 325

Change Detection and Resolution User Interface 327

Change Detection and Resolution Overview

This chapter describes how you can detect changes in your enterprise application that necessitate modifications to your flows, and how you can instruct Business Process Testing Enterprise Extension to make these modifications automatically.

Change detection and resolution simplify component maintenance, thereby reducing the effort needed to invest in automated testing.

Note:

- If you do not have a license for Business Process Testing Enterprise Extension, the features described in this chapter are not relevant to you.
- If Business Process Testing Enterprise Extension is not enabled for your project, this feature is not available. Access to Business Process Testing Enterprise Extension must be provided by your HP Application Lifecycle Management (ALM) Project Administrator using Project Customization. An administrator can enable Business Process Testing Enterprise Extension in ALM by choosing **Tools > Customize > Business Process Test** and selecting the **Enable Business Process Testing Enterprise Extension** checkbox. For details, see the *HP Application Lifecycle Management Administrator Guide*.

Change Detection Overview

Business Process Testing Enterprise Extension enables you to run tests (and flows) on an enterprise application in Change Detection mode. This enables you to determine how the application has changed since the test or flow was built, such as whether controls in the user interface were added, deleted, or changed.

Note: Only business components created using the Learn Flow process can run in Change Detection mode.

Change Resolution Overview

When the test or flow run is finished, you can view the details of these changes in the Change Detection Report. This report provides an indication of the modifications you should make to your test or flow to make sure that it is up-to-date.

For each change detected, Business Process Testing Enterprise Extension offers you a choice of ways to resolve the change, such as:

- Opening a defect if the change detected in the application is a defect.
- Making updates to your test or flow automatically, or removing a step from one of the components in the test if the change detected in the application is correct.

For a list of supported changes and resolutions, see "Detection and Resolution Options" on page 325.

When changes to a component are detected while running a test or flow in Change Detection mode, Business Process Testing also checks if there are components similar to those containing the changes in other tests or flows in the project. Components are considered similar if they at least

represent the same learned screen. If Business Process Testing finds similar components, it sends alerts to the tests and flows that are using the components. The alerts recommend that change detection be run for tests and flows containing those components.

Similar components can be reused. If the component in which a change was detected is a reused component, or the step in which the change was detected is within a reused component, you need to decide where to apply the change resolution. You can choose to apply the resolution to the original component, in which case all tests and flows using that component will be affected. You can alternatively choose to apply the resolution to a copy of the component, and use that copy in your flow instead of the reused component.

Example

Suppose you are testing a screen for inputting contact information for new customers. The screen contains the fields **Name**, **Address**, and **Phone Number**. You create a test that verifies that information entered in these fields is correctly added to your customer database. Suppose you now add an **E-mail Address** field to the screen. If you run your test in regular mode, the test may pass and you may not notice that there is an additional field that should be tested. However, if you run the test in Change Detection mode, Business Process Testing Enterprise Extension notices that the field was added to the screen and suggests adding a step to the component corresponding to the new field. You can then run an updated version of the test that includes verification of the additional field.

Similarly, if a field was removed from the screen, Business Process Testing Enterprise Extension notices that the field was removed, even if no step in the component corresponds to the field. The Change Detection Report suggests updating the component to the changed screen.

How to Run Tests in Change Detection Mode

This task describes how to detect changes made to your enterprise application by running tests and flows for the application in change detection mode.

You can detect changes to an individual flow, test, or test set.

Note: If you do not have a license for Business Process Testing Enterprise Extension, the features described in this chapter are not relevant to you.

This task is part of a higher-level task. For details, see "How to Run Automated Business Process Tests and Flows" on page 305.

This task includes the following steps:

- "Prerequisites" on next page
- "Locate the flow, test, or test set" on next page
- "Start the run " on next page
- "Run the flow, test, or test set" on next page
- "Results" on page 323

1. ## Prerequisites

   - Make sure Business Process Testing Enterprise Extension is enabled. Access to Business Process Testing Enterprise Extension must be provided by your ALM Project Administrator using Project Customization. An administrator can enable Business Process Testing Enterprise Extension in ALM by choosing **Tools > Customize > Business Process Test** and selecting the **Enable Business Process Testing Enterprise Extension** checkbox. For details, see *HP Application Lifecycle Management Administrator Guide*.
   - To detect changes in Change Detection mode, a user must belong to a user group that has permissions for the Run task, and permissions to modify tests and business components. For more information on configuring user group permission settings, see the *HP Application Lifecycle Management Administrator Guide*: .
   - In UFT, select **Tools > Options** and click the **Run** node. Ensure that the **Allow other HP products to run tests and components** check box is selected in the Run pane.
   - For additional details on configuring UFT to work with enterprise applications, setting up Business Process Testing Enterprise Extension, and on configuring user permissions, see the *HP Application Lifecycle Management Administrator Guide*.
   - If detecting changes for a flow, make sure that the flow for which you want to detect changes is associated with a business process test and all its components have a **Ready** status.

2. ## Locate the flow, test, or test set

   - To detect changes for an individual flow or test , select the flow or test from the tree in either the Test Plan or Test Lab module.
   - To detect changes for a test set, select the test set from the tree in the Test Lab module.

3. ## Start the run

   In the Test Lab Module:

   a. In the Execution Grid or Execution Flow tabs, click **Run**.

   In the Test Plan Module:

   a. Right-click the flow or test for which you want to detect changes.

   b. Select **Detect Changes**.

   c. If the Select Test Instance dialog box opens, select from the lists to define which instance of the flow you want to run.

      For user interface details, including the situations under which this dialog box opens, see "Select Test Set (Instance) Dialog Box" on page 327.

   The Automatic Runner dialog box is displayed.

4. ## Run the flow, test, or test set

   In the Automatic Runner dialog box, select the **Detect Changes** checkbox and the flow or test you want to run, and then click **Run**.

Components in the flow that were not created using the Learn Flow process cannot be run in Change Detection mode.

For task details on learning flows, see "How to Learn Flows" on page 170.

For user interface details on the Automatic Runner dialog box and task details on running tests automatically, see the *HP Application Lifecycle Management User Guide*.

5. **Results**
   - If Business Process Testing finds similar components, it sends alerts to the tests and flows that are using the components.
   - A Change Detection Report is created. You can open the Change Detection Report screen from the Test Plan module or the Test Lab module.

     For task details on how to work with this report, see "How to Resolve Detected Changes" below.

     For user interface details, see "Change Detection Report Screen" on page 328.

How to Resolve Detected Changes

This task describes view and resolve changes detected in enterprise applications using the Change Detection Report screen.

If you do not have a license for Business Process Testing Enterprise Extension, the features described in this chapter are not relevant to you.

This task is part of a higher-level task. For details, see "How to Run Automated Business Process Tests and Flows" on page 305.

This task includes the following steps:

- "Prerequisites" below
- "Open the Change Detection Report screen" on next page
- "Select the entity that changed" on next page
- "Resolve the change" on next page
- "Handle similar, changed components during change resolution" on next page
- "Execute the resolution" on next page
- "Advance to the next entity that changed" on next page
- "Save all changes" on page 325
- "Results" on page 325
- "Handle similar, changed components after resolving changes - Optional" on page 325

1. **Prerequisites**

   To resolve changes, a user must belong to a user group that has permissions for the **Modify**

Test and the **Modify Component** tasks. For more information on configuring user group permission settings, see the *HP Application Lifecycle Management Administrator Guide*.

2. ### Open the Change Detection Report screen

   For access details, see "Change Detection Report Screen" on page 328.

3. ### Select the entity that changed

   Select the element in the changes tree for which you want to resolve changes.

4. ### Resolve the change

   In the **Changes** tab, under **Resolve**, select the change resolution you want to use. The following resolutions are available for all changes:

   - **Ignore and mark as resolved**. Marks the change as resolved without taking any action to resolve it.
   - **Open a defect**. Opens the New Defect dialog box, enabling you to open a defect for the change. Note that if you open a defect for a change, you can also select another resolution for the change. For more information on opening defects, see the *HP Application Lifecycle Management User Guide*.

   Note: If the change detected is that there are new objects in the screen or tab, the new objects are inserted in the shared object repository for that screen or tab when you select **Ignore** and mark as **resolved** or open a defect.

   In addition, there may be additional possible resolutions, depending on the type of change. For a list of available change resolutions for each type of change, see "Detection and Resolution Options" on next page.

5. ### Handle similar, changed components during change resolution

   If a change was detected in a reused component, a warning message is displayed in the **Resolve** section of the **Changes** tab, indicating how the resolution will be applied.

   To change the resolution options for a reused component:

   a. Select the component from the Changes tree.

   b. Click the **Used By** tab.

   c. Select a radio button from the **Reused Component Options**.

6. ### Execute the resolution

   For each change, click the **Execute** button. The change is marked as resolved. Note that for change resolutions other than **Open a defect**, the resolution is not implemented until you click **Save**.

   Version Control: When you click **Execute**, assuming the component was not checked out by another user, the component is checked out by you. The component is checked in after you click **Save**.

7. ### Advance to the next entity that changed

Click the **Next Change** button to select the next node in the Changes tree with a detected change that was not yet resolved.

Repeat steps "Select the entity that changed" on previous page to "Execute the resolution" on previous page until all changes in the test are resolved. You can view the number of changes resolved and the total number of changes in the **Status** pane, next to the **Start Resolution** button.

8. **Save all changes**

   Click the **Save** button. The Change Detection Results screen closes.

9. **Results**

   Any change detection alerts associated with tests and flows are removed.

   Your tests, flows, business components and object repositories are updated according to the resolutions you selected.

10. **Handle similar, changed components after resolving changes - Optional**

    When changes to a component are detected while running a test or flow in Change Detection mode, Business Process Testing also checks if there are components similar to those containing the changes in other tests or flows in the project. Components are considered similar if they at least represent the same learned screen. If Business Process Testing finds similar components, it sends alerts to the tests and flows that are using the components.

    a. In the Test Plan module, refresh the Test Plan tree.

    b. Look for red exclamation marks (alerts) to the left of test or flow names.

    c. Click the red exclamation marks to open the alerts.

    d. If the alerts recommend change detection, run those tests and flows in change detection mode.

Detection and Resolution Options

This section lists the possible changes detected for flows, business components, and component steps. It also lists the resolutions offered by Business Process Testing Enterprise Extension for each change type.

In addition to the options described in this section, the resolutions **Ignore and mark as resolved** and **Open a defect** are always available. For details on these resolutions, see "How to Resolve Detected Changes" on page 323.

This section includes the following topics:

- "Changes and Resolutions for Flows" on next page
- "Changes and Resolutions for Business Components " on next page
- "Changes and Resolutions for Steps" on page 327

Changes and Resolutions for Flows

Change	Resolutions Offered
A screen corresponding to a component was removed.	• **Remove the business component from the flow.** Removes the component representing the removed screen from the current test or flow only. • **Remove the component from all flows.** Removes the component representing the removed screen from all tests or flows in which it is used.

Changes and Resolutions for Business Components

Change	Resolutions Offered
New controls were added to the screen corresponding to the component.	**Select the steps you want to add.** Adds new steps to the component for the new controls, and the component is updated to reflect the changed screen. Each new control has a suggested step that can be added to your component. Select a check box for each step you want to add to your component. Controls are added before the last step of the component. Steps are added with the default operation for the control.
A control was added, removed, or changed in the screen corresponding to the component. (The control was not included in any step.)	**Update the component to the changed screen.** The component is updated to reflect the changed screen. You can also select the step or steps you want to add to reflect the addition or change.
One or more controls could not be found in the shared object repository.	**Update the component to the changed screen.** The shared object repository is updated.
No controls matching the steps in the component were found. All steps in the component failed.	**Change the status of the business component to Maintenance.** The user can update the component at a later time.
The component did not open because all steps in the previous component failed.	**Set the previous component to Maintenance status.** The user can update the component at a later time.
The screen number was recorded on a previous version of the application.	Relearn the flow to update the screen number and run Change Detection again.

Changes and Resolutions for Steps

Change	Resolutions Offered
The control corresponding to the step was removed from the screen.	**Remove the step from the business component.** Removes the step from the component.
The type of the control corresponding to the step changed.	**Update the step.** Updates the step to reflect the new control type.

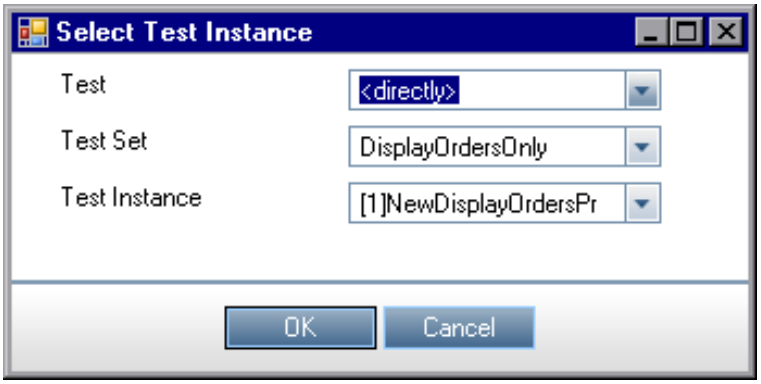
Change Detection and Resolution User Interface

This section includes:

Select Test Set (Instance) Dialog Box 327
Change Detection Report Screen 328

Select Test Set (Instance) Dialog Box

This dialog box enables you to specify an instance of a business process test or flow for which you want to detect changes or view the Change Detection report.



To access	Run an automated flow, business process test, or test set that contains a learned component. Select **Detect Changes** and click **OK**. **Note:** Under certain circumstances, this dialog box is not displayed. See "Important information" below.
Important information	• If the flow is included only once in one test set and it is included directly (not from within a test), the Select Test Instance dialog box does not open (the Automatic Runner dialog box opens instead). • If a flow is not contained in any test set, it is added directly to the **default** test set in the **Unattached** folder. The Select Test Set Dialog box does not open and the Automatic Runner dialog box opens instead.

Relevant tasks	"How to Run Tests in Change Detection Mode" on page 321
See also	"Change Detection and Resolution Overview" on page 320

User interface elements are described below:

UI Element	Description
Test	The list of tests containing the flow. If the flow is included in a test set directly, and not just from within a test, **<directly>** is displayed as a choice in the Test list. Select **<directly>** to detect changes on the flow itself, instead of on the test in which it is included. **Available when:** The flow is included in more than one test set in the Test Lab module.
Test Set	**For flows**. The list of test sets containing the test selected in the Test list. If **<directly>** is selected in the Test list, the Test Set list includes only those instances that include the flow directly and not from within a test. **For business process tests**. The list of test sets that include the test you selected.
Test Instance	The list of instances of the business process test or flow in the test instance selected in the Test Set list. **Available when:** The flow is included in more than one test instance in the Test Lab module.

Change Detection Report Screen

This screen enables you view results of a test or flow run in Change Detection mode and resolve detected changes .

For an example of an application-specific Change Detection Report screen, see "An Example of a SAP-specific Change Detection Report" on page 344.

To access	**From the Test Plan module:** 1. In the Test Plan Tree, right-click the test or flow and select **Update Changes**. 2. Select from the lists to define which report you want to open from the Select Test Instance dialog box, if displayed. For user interface details, see "Select Test Set (Instance) Dialog Box" on page 327. 3. Click **OK**. **From the Test Lab module:** 1. In the Test Sets Tree, select the test set containing the test for which you want to open the Change Detection Report screen. 2. Click the **Execution Grid** tab. 3. In the Execution Grid tab, select the test for which you want to open the Change Detection Report screen. **Note:** Tests or flows whose last run was in Change Detection mode are indicated with a **Y** in the Change Detection column. Tests and flows that were run in Change Detection mode whose changes are unresolved, are indicated with a delta symbol in the lower right corner of the test icon or flow icon . A test or flow indicates unresolved changes if it contains any flows or components with unresolved changes. Tests and flows whose changes were resolved will have the delta symbol removed from their icon in the Test Plan and Test Lab modules. A test or flow will change its status only when all its flows or components have had their changes resolved. 4. In the Last Run Results pane, in the Status column, click the link for the test. You can open the regular Test Results window by right-clicking the link in the Status column and selecting **Open Regular Report**.

Important information	• If a test or flow was run in Change Detection mode, but no changes were detected, **Update Changes** is disabled. You can view the Change Detection Report screen from the Test Lab module only. • If a test or flow has changes detected for only one test set, the Change Detection Report screen opens directly. If the test or flow appears in more than one test set, but was already resolved and rerun from one test set, the last results open. • It is not necessary to resolve detected changes in every component in the Change Detection Report at one time. You can open a saved report and update it multiple times. In a reopened report, components in which one or more changes were previously resolved are shown as read-only. However, the components in which changes were not resolved are active, and resolution options are displayed for selection. • Assuming no other user has checked out the component, a component in a version controlled project is checked out when a resolution option for a detected change in the component is executed in the Change Detection Report screen. The component is checked in again when the Change Detection Report screen is saved. • Components in the Change Detection Report may be displayed as read-only for several reasons. For example: ▪ Changes to the component were already resolved. ▪ The component is duplicated in the flow and changes in the duplicate were resolved. ▪ The component is checked out. ▪ The component was removed from the flow. ▪ The component was changed outside of the Change Detection Report (for example, in UFT). ▪ A new report was run and the component is not up to date.
Relevant tasks	• "How to Run Tests in Change Detection Mode" on page 321 • "How to Resolve Detected Changes" on page 323
See also	• "Change Detection and Resolution Overview" on page 320

This screen contains the following areas:

- "Details area" on next page
- "Status Area" on next page
- "Entities Area" on page 332
- "Filter Area" on page 332
- "Changes Tree" on page 332
- "Entities Details Area" on page 333

- "Changes Tab" on page 333
- "Used By Tab" on page 334
- "Screenshot Tab" on page 334

Details area

This area is located at the top of the Change Detection Report screen. The Details area displays basic details about the test.

User interface elements are described below:

UI Element (A-Z)	Description
Test Name	The name of the test.
Creation Date	The date on which the test was created.
Owner	The user who is the current owner of the test.
Test Status	The status of the test run. The following statuses are possible: • **Passed**. The test run passed; No changes detected. • **Failed**. The test run failed; No changes detected. • **Passed with changes**. The test run passed; Changes detected. • **Failed with changes**. The test run failed; Changes detected.
Run started	The date and time on which the test run began.
Run Ended	The date and time on which the test run ended.

Status Area

This area is located at the top of the Change Detection Report screen, under the **Details** area.

User interface elements are described below:

UI Element (A-Z)	Description
Business Components	Graphically displays the number of business components in the test, and the breakdown of these components by their status. You can click a component status to show only components with that status in the changes tree.
Resolved	Displays the number of components that have been resolved from the total components in the test.

Entities Area

This area is located under the Status area.

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

UI Element	Description
Filter area	Displays the currently-defined filter. For details, see "Filter Area" below
<Changes tree>	Displays a hierarchical representation of your test, flows, business components, and component steps. For details, see "Changes Tree" below.
<Entity details area>	Displays entity details. For details, see "Entities Details Area" on next page.
Changes tab	Display changes found in the enterprise application. For details, see "Changes Tab" on next page.
Used by tab	Displays other entities that use the selected entity. For details, see "Used By Tab" on page 334.
Screenshot tab	Displays screenshots for the entity. For details, see "Screenshot Tab" on page 334.

Filter Area

Displays the currently-defined filter for the Changes tree, if any.

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

UI Element	Description
<filter>	Displays the currently defined filter.
	Clear Filter. Clears any filters set from the **Changes tree**.

Changes Tree

The Changes Tree is located at the left of the Entities area. This tree displays a hierarchical representation of your test, flows, business components, and component steps.

Tip: You can filter the **Status** or the **Resolved** column by clicking **Set Filter** for the column and selecting the values you want to display. To clear the filters, click **Clear Filter** in the Filter area.

User interface elements are described below:

UI Element	Description
Entity Name	The name of the test, flow, business component, or step. A reuse indicator in the component's icon indicates that the component is a reused component.
Status	The status of the entity: • The entity passed, and no changes were detected. • The entity failed, but no changes were detected. • The entity passed, but changes were detected. • The entity failed, and changes were detected.
Resolved	For an entity in the changes tree with associated changes, displays **Y** if the changes were resolved and **N** if the changes were not resolved. If there are no changes associated with the node, --- is displayed. If a component was changed outside of the Change Detection Report (for example, in UFT), it is displayed as read-only. A message is displayed in the Changes tab that the component was changed by another testing tool after the run of the report.

Entities Details Area

This area is located at the top right of the Entities area, and displays details of the entity selected in the changes tree.

Changes Tab

Located at the bottom right of the Entities area, the Changes tab displays a description of any changes detected in selected component, flow, or step, and offers possible resolutions.

User interface elements are described below:

UI Element	Description
Changes	Displays the description of the change that was found..
Resolve	Displays options for resolving the change, including: • Ignore and mark as resolved. • Open a defect. • Remove the step from the business component. The component is updated to the changed screen. • Update component to the changed screen.

Used By Tab

The Used By tab is located at the bottom right of the Change Detection Report screen.

This tab is displayed only for flows and components.

User interface elements are described below:

UI Element	Description
Used By table	For a flow, displays the tests in which the flow is used. For a component, displays the tests and flows in which the component is used.
Reused Component Options	For reused components, enables modifying the way in which change resolutions are handled. **Available for:** Components that contain a change These options apply only to resolutions that affect the component. For example, the **Reused Component Options** are not relevant for the **Open a defect** resolution. The **Reused Component Options** area contains the following options: • **Apply changes to the original component.** All flows and tests sharing this component will reflect the changes. Resolutions to the component will be reflected in all flows and tests using the component. • **Apply the changes to a new, local copy of the component and use the new component in this flow**. All other flows and tests using the original component will be unaffected by the resolution. The flow will contain a local copy of the component and will not reuse the existing component. The copy of the component is saved in the default folder for the flow. For details on component reuse, see "Criteria for Determining if a Component Can Be Reused" on page 175

Screenshot Tab

The Screenshot tab is located at the bottom right of the Change Detection Report screen.

This tab is displayed only for components that have changes detected.

The Screenshot tab displays a screenshot of your application for the selected component in the flow before and after the change. For a step, displays the screenshots for the component containing the step.

Button Area

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

UI Element	Description
Previous Change	Navigates to the previous unresolved change in the Changes tree.

UI Element	Description
Next Change	Navigates to the next unresolved change in the Changes tree.
Execute	Executes the resolution option for the change that you selected in the Changes tab. Version Control: The component is checked out.
Save	Closes the Change Detection Report screen and implements the change resolutions that you selected. Version Control: The component is checked in.
Cancel	Closes the Change Detection Report screen without implementing the change resolutions that you selected.

Appendices

Appendix A

Business Process Testing Enterprise Extension Environment-specific Information

This appendix includes:

Business Process Testing Enterprise Extension Environment-specific Information Overview 340

How to Work with Business Process Testing Enterprise Extension with Support for SAP 340

How Business Process Testing Enterprise Extension Creates Components when Learning SAP Flows 341

Business Process Testing Enterprise Extension Environment-specific User Interface 343

Business Process Testing Enterprise Extension Environment-specific Information Overview

This provides environment-specific user information for Business Process Testing Enterprise Extension.

If you do not have a license for Business Process Testing Enterprise Extension, the features described in this chapter are not relevant to you.

Supported Environments

In this version, Business Process Testing Enterprise Extension supports SAP GUI and SAP CRM. Additional environments will be supported in future versions.

How to Work with Business Process Testing Enterprise Extension with Support for SAP

The following steps describes how to work with Business Process Testing Enterprise Extension if your supported enterprise environments include SAP.

- "Prerequisites" below
- "Configure UFT to work with Business Process Testing Enterprise Extension features with support for SAP" below
- "Understand how Business Process Testing Enterprise Extension creates components when learning SAP flows" on next page

This task is part of a higher-level task. For details, see "Business Process Testing Enterprise Extension Workflow" on page 31.

1. **Prerequisites**
   - Make sure that the SAP add-in for UFT is installed.
2. **Configure UFT to work with Business Process Testing Enterprise Extension features with support for SAP**
   a. Open UFT.
   b. Select **Tools > Options** and click the **SAP** node. In the SAP pane:
      - Under **SAP GUI for Windows > Record settings** ensure that:
        - the **Auto-parameterize table and grid controls** check box is selected. This option ensures that Business Process Testing can learn table values during the learn flow process.
        - the **Record only the selected value when using a Possible Entries (F4) list** check box is checked.

- Under **SAP GUI for Windows > Run settings** ensure that the **Session cleanup** check box is not selected.

c. If your SAP application contains objects from environments other than SAP and Web, you may configure Business Process Testing Enterprise Extension to recognize these objects.

> **Note:** Only users who are comfortable editing XML files should perform this operation.

i. Open the **Bpt4ErpCrmCfg.xml** file in the **Dat** folder of your Unified Functional Testing installation folder.

ii. Add the following line to the **Settings** section:

<Config name="additional_addins" value="**add-in name**" />

where the **add-in name** is the name of the add-in as it appears in the **Add-in Manager** of Unified Functional Testing. You separate multiple add-ins with a semicolon, with no additional spaces.

Example:

<Config name="additional_addins" value="Java;ActiveX" />

3. **Understand how Business Process Testing Enterprise Extension creates components when learning SAP flows**

Familiarize yourself with how Business Process Testing Enterprise Extension creates components. For reference details, see "How Business Process Testing Enterprise Extension Creates Components when Learning SAP Flows" below.

How Business Process Testing Enterprise Extension Creates Components when Learning SAP Flows

When Business Process Testing Enterprise Extension learns a flow, it creates a new business component for each screen or tab through which you navigate in your SAP application. Each screen or tab within a particular transaction is represented as a separate business component.

The name of a learned component is of the form **<Transaction code>-<Screen name>**. If the transaction code or screen name contains spaces or characters that are not valid in the names of business components, these characters are replaced by underscores (_). For example, if a learned component corresponds to the screen **Create Sales Order: Initial Screen** in the **VA01** transaction, the component is named **VA01-Create_Sales_Order_Intial_Screen**.

If you navigate from one screen to a different screen or tab, and then return to the original screen, a new component is created with the same name as the component corresponding to the original screen, with a numerical suffix appended to differentiate between the components. For example, suppose the component corresponding to a screen is named **VA01_Test**. If you return to the screen, a new component is created named **VA01_Test_01**. Reusing existing components in your learned flow minimizes the number of learned components for similar or identical screens.

For each user operation within a screen that requires data specified by the user, an input component parameter is created for that step. the input component parameters that are created during the Learn

Flow process are automatically set as flow parameters. The value input during the Learn Flow process is used as the default parameter value. For example, a parameter would be created for a value entered in a text field.

The name of an automatically generated parameter is the same as the name of the relevant field in the application. For example, in a Display Sales Order transaction, the parameter name of the order number would be **Order**. If a flow contains several components that have the same parameter name, an incremental suffix is added to subsequent parameter names (for example, **Order_1**, **Order_2**) in order to differentiate between them.

Note: In previous versions of Business Process Testing Enterprise Extension, the name of an automatically generated parameter was in the format **<name of object_class type_name of method_name of argument>**. For example, the parameter used to set the value of the Sales Organization edit box was named Sales_organization_Edit_Box_Set_Text. Old parameter names are not changed to the new format during the upgrade process.

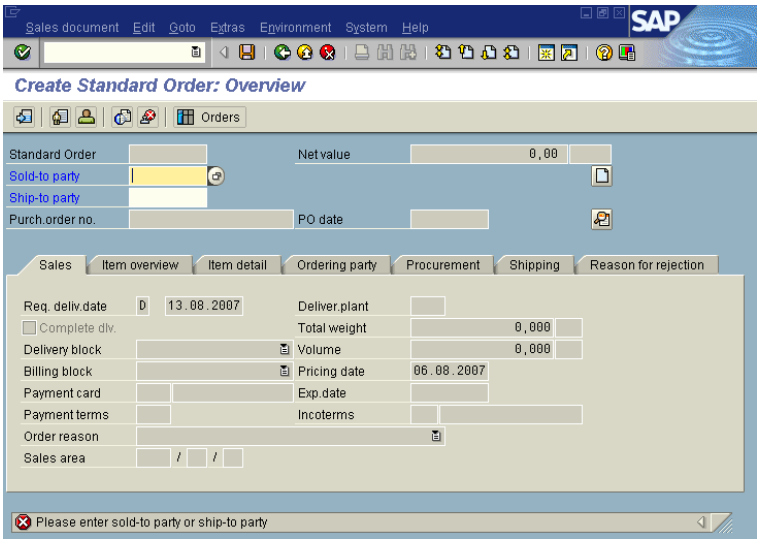
The description of a generated parameter includes the type of control and the name of the screen in which it appears, for example, **"The 'Order' control is of type 'Edit Box'. The control is a part of the 'Display Sales Order: Initial Screen' screen."**

If you add data to a table in your SAP application while learning a component, a component table parameter is created. Table parameters enable you to use one parameter for an entire table, rather than using separate parameters for each table cell. For more information on working with table parameters, see "Specify a table value" on page 240.

Note:

- During the Learn Flow process the table parameter saves the data only after you move the focus out of the table. If the table data is the last step in a learn process, ensure that you move the focus out of the table before stopping the learn.
- To create table parameters during the Learn Flow process, you must enable UFT to create table parameters. In UFT, select **Tools > Options > SAP** node. Under **SAP GUI for Windows > Record settings** ensure that the **Auto-parameterize table and grid controls** check box is checked. For more information, see "Specify a table value" on page 240.

If a screen contains a tab strip, one component is created for the common area on the screen, and a separate component is created for each tab opened. The status bar is included in each component created for the screen. For example, consider the screen below:



When you learn a flow containing this screen, operations performed on the upper area of the screen containing the menus, toolbars, and common fields (such as **Standard Order** and **Net value**) comprise a single business component. Operations performed in each of the **Sales, Item overview, Item detail, Ordering party, Procurement, Shipping**, and **Reason** for rejection tabs result in the creation of separate business components.

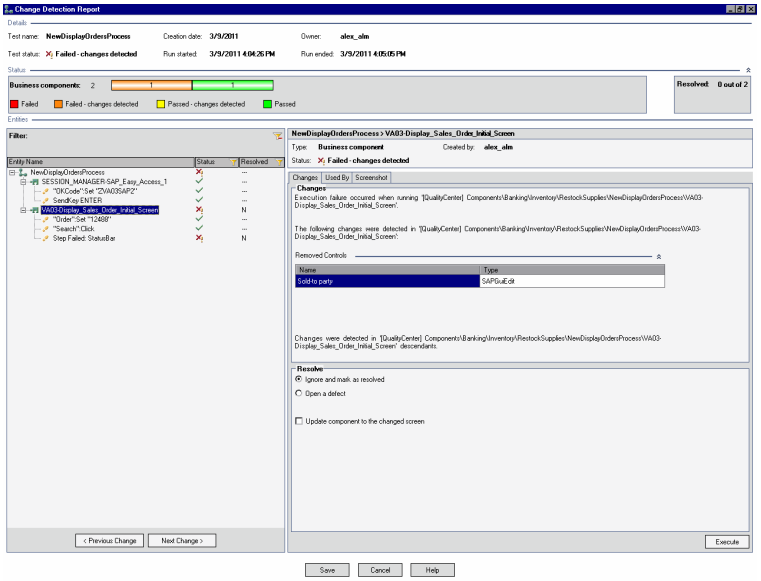
Business Process Testing Enterprise Extension Environment-specific User Interface

This section includes:

An Example of a SAP-specific Change Detection Report 344

An Example of a SAP-specific Change Detection Report

This report enables you to view the results of a test run in Change Detection mode in an SAP environment and resolve the detected changes.



To access	For access details, see "Change Detection Report Screen" on page 328.
Important information	Note the **Resolved** column in the **Filter** area, which indicates components with detected changes that have not yet been resolved.
Relevant tasks	• "How to Run Tests in Change Detection Mode" on page 321 • "How to Resolve Detected Changes" on page 323
See also	• "Learn Flow Process Overview " on page 170 • "Change Detection and Resolution Overview" on page 320

User interface elements are described under "Change Detection Report Screen" on page 328.

Business Process Testing Glossary

This glossary lists terminology that is used throughout Business Process Testing.

See also the HP Application Lifecycle Management (ALM) glossary in the *HP Application Lifecycle Management User Guide*.

A

Term	Description
API test	An automated, functional test for headless applications, such as Web services. These tests are created in UFT and can be viewed in ALM. The UFT term **API test** is also known as an automated HP Service Test scripted test.
application area	Application areas contain all of the settings and resources required to create the content of keyword-driven business components for a particular application or part of an application. These include representations of the objects from your application contained in the shared object repository, and user-defined operations, contained in function libraries.
automation engineer	An expert in an automated testing product, such as Unified Functional Testing (UFT).

B

Term	Description
business component (or **component**)	A business component is the building block on which business process testing is based. A business component is an easily-maintained, reusable unit comprising one or more steps that perform a specific task and their component steps describe the condition or state of the application before and after that task. You can use business components in multiple tests and flows. A business component can be defined as a manual or automated component. Business components may require input values from an external source or from other components, and they can return output values to other components.
Business Components module	Functional area of ALM accessible by selecting Business Components on the ALM sidebar under Testing. You use the reusable business process components created in this module to create business process tests and flows in the Test Plan module. Access to this module is dependent on your ALM license.

Term	Description
business process test	A scenario comprising a sequence of business components or flows, designed to test a specific business process of an application.
Business Process Testing Enterprise Extension	An HP product that adds additional functionality to ALM and Business Process Testing that enables you to test enterprise applications. This functionality includes features such as learning flows and change detection.

C

Term	Description
canvas view	A view that enables you to modify test scripts using a graphical display.
change detection	Using Business Process Testing Enterprise Extension while running a business process test to determine if an enterprise application has changed in such a way that modifications are needed to previously-learned flows. See also: **Business Process Testing Enterprise Extension**
component	See **business component**.
component request	A type of a component that represents a request for the creation of a new business component, when no existing component answers the needs of a business process test or flow. Component requests are submitted in the Test Plan module and can be viewed in the Component Requests folder in the Business Components module.
content	Low-level, detailed information about a component, such as the component's steps. See also: **step**
criteria	When creating coverage between requirements and business process tests, instead of covering each requirement only at the level of a test, test criteria enables coverage by defined business components of a test.

D

Term	Description
data table	A type of test resource in the ALM Test Resources module, used for uploading Microsoft Excel files for use with test configurations that access external, dynamic data. See also: **dynamic data**

Term	Description
debug mode	A mode for running an automatic test that ensures that the test runs properly, and locates errors that may have occurred during the building of the test.
design step	A manual component step. See also: **step**
dynamic data	Values for a test configuration specified in an external Microsoft Excel file, which is then uploaded as a test resource and mapped to a test configuration in ALM.

E

Term	Description
encryption	Setting a parameter value so that it is displayed as asterisks and cannot be seen.

F

Term	Description
facet	A container for the content of a business component. A facet contains a set of design steps for manual components or automated steps or scripts for automated components. A component can have a manual facet, an automated facet, or both.
flow	A type of test that comprises a logical set of business components, in a fixed sequence, that performs a specific task. Flows are reusable, run as a unit, and can be thought of as "compound components." You can use flows in multiple business process tests. Flows may require input values from an external source or from other components or flows, and they can return output values to other components or flows.
flow-level input parameter	Variable values that enable you to define data used by a flow that is provided to the flow from an external source. See also: **parameterization**
flow-level output parameter	Values that enable you to define data that is retrieved and stored by a flow and can be used in subsequent components in a flow. See also: **parameterization**

G

Term	Description
group	A combination of contiguous components in a business process test or flow. Flows can also be grouped along with components in a business process test. Groups are usually created for iteration purposes.
GUI test	See **keyword GUI test** and **scripted GUI test**

I

Term	Description
implementation requirements	Details and information, written by the subject matter expert that creates a business component, describing a general overview of the business component's purpose or goals, and the condition of the application before and after a component is run (the pre-conditions and post-conditions).
input component parameter	Variable values that a business component can receive and use as the values for specific, parameterized steps in the component. See also: **parameterization**
iteration	The number of times the following entities run: • An individual business component • A group of components and flows • An individual flow • An entire business process test • A test configuration The iterations for these entities, except for groups, can use different values for their parameters.

K

Term	Description
keyword GUI test	An automated, keyword-driven functional test that can be created in UFT or ALM. The UFT term **keyword GUI test** is also known as an automated HP QuickTest Professional keyword-driven test.
keyword view	The keyword view enables you to create and view the steps of your keyword-driven automated components in a modular, table-like format. Each step is displayed as a row in the keyword view, and is comprised of individual, modifiable parts. You can create and modify steps by selecting items and operations in the keyword view and entering information as required. Each step is automatically documented as you complete it, enabling you to view a description of your action or component in understandable sentences. You can also use these descriptions as instructions for manual testing, if required.

L

Term	Description
learn flow process	The act of using Business Process Testing Enterprise Extension to create flows that represent operations in an enterprise application. Flows are created/recorded automatically by performing the test/scenario once.Learned flows contain automated components. See also: **Business Process Testing Enterprise Extension**
linkage	Parameter linkage enables you to make data available between business components and flows by passing the values of output parameters to input parameters of subsequent business components and flows in a test. The process of setting the value of an input parameter to the value of an output parameter is called linking output and input parameters.

M

Term	Description
mapping	Associating component or flow parameters to column headers of an imported .csv or .xls file.

O

Term	Description
obsolete	Components, folders, or component requests that have been deleted but are still being used by one or more business process tests or flows.
obsolete folder	Special folder that contains obsolete components, folders, and component requests.
output component parameter	Values that a business component can return. These values can be viewed in the business process test results and can also be used as input for a component or flow that is used later in the test. See also: **parameterization**

P

Term	Description
parameter or **parameterization**	Parameters expand the scope of business components, flows, and business process tests by replacing fixed values with input and output parameters. Parameterization enables you to iterate specific components in a business process test, specific components in a flow, test configurations, or entire business process tests, thereby creating data-driven tests. See also: **input component parameter, output component parameter, test-level parameter, flow-level input parameter, flow-level output parameter**
pre-condition and **post-condition**	The state of the application before the first step in a business component, and the state of the application after the last step in a business component. For example, the pre-condition for a component may state that all applications should be closed, or a specific application should be open to a specific screen or with a specific dialog box displayed. A post-condition may state that an application should be open to a particular screen, or be closed. If you want the business component to allow iterations, the post-conditions should specify that the application's end state must match its state at the beginning of the component. The pre-conditions and post-conditions provide a guide when implementing the steps for the business component.
promotion	The act of making a lower-level parameter value available to a higher-level entity. You can promote a flow-level parameter to the test level. You can promote a component-level parameter to the flow level.

Q

Term	Description
QuickTest Professional	A comprehensive functional and regression testing tool. HP Unified Functional Testing (UFT) comprises the products known as HP QuickTest Professional and HP Service Test. • Functionality provided by QuickTest is known as GUI testing in UFT. • Functionality provided by Service Test is known as API testing in UFT.

R

Term	Description
reuse	When learning flows using Business Process Testing Enterprise Extension, existing automated components can be used instead of creating new components. You determine the reuse based on similarity with existing components. See also: **Business Process Testing Enterprise Extension**

Term	Description
roles	The various types of users who are involved in Business Process Testing.
run condition	A conditional statement you can insert into a flow to check the current value of a given parameter before running a component in a flow. The run condition determines whether to run the component, skip the component run, or skip and set the component status to fail.

S

Term	Description
scripted GUI test	An automated, scripted functional test that can be created in UFT and viewed in ALM. The UFT term **scripted GUI test** is also known as an automated HP QuickTest Professional scripted test.
Service Test	A functional testing tool for testing headless applications, such as web services. HP Unified Functional Testing (UFT) comprises the products known as HP QuickTest Professional and HP Service Test. • Functionality provided by QuickTest is known as GUI testing in UFT. • Functionality provided by Service Test is known as API testing in UFT.
shell	Basic information about a business component, such as its name and status.
static data	Values for a test configuration specified from within ALM.
status (component)	A status value that indicates whether a business component is fully implemented, debugged, and ready to be run in a business process test or flow. The component with the most severe status determines the status of the entire test or flow. The status can be modified by the subject matter expert in ALM and the automation engineer in UFT.
status (business process test, or flow)	A status value that is automatically generated based on the status values of all the business components in the business process test or flow.
step	Individual operation in a component. A step can be manual (a design step) or automated, and is designed and implemented in a business component to test the workings of a specific task in an application. Steps are contained in facets, and comprise the content of a component.
subject matter expert	The subject matter expert who uses ALM to create business components, flows, and business process tests.

T

Term	Description
test-level parameters	Variable values that a business process test or flow can receive and then pass to business components or flows for use as input values. See also: **parameterization**

U

Term	Description
Unified Functional Testing (UFT)	A comprehensive functional and regression testing tool. HP Unified Functional Testing (UFT) comprises the products known as HP QuickTest Professional and HP Service Test. • Functionality provided by QuickTest is known as GUI testing in UFT. • Functionality provided by Service Test is known as API testing in UFT.

Java™
COMPATIBLE