



HP Business Process Testing

Software Version: 12.50

User Guide

Document Release Date: September 2015
Software Release Date: September 2015

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 - 2015 Hewlett-Packard Development Company, L.P.

Trademark Notices

Adobe™ is a trademark of Adobe Systems Incorporated.

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

UNIX® is a registered trademark of The Open Group.

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

Documentation Updates

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

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

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

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to <https://softwaresupport.hp.com> and click **Register**.

Support

Visit the HP Software Support Online web site at: <https://softwaresupport.hp.com>

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: <https://softwaresupport.hp.com> and click **Register**.

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

HP Software Solutions & Integrations and Best Practices

Visit **HP Software Solutions Now** at <https://h20230.www2.hp.com/sc/solutions/index.jsp> to explore how the products in the HP Software catalog work together, exchange information, and solve business needs.

Visit the **Cross Portfolio Best Practices Library** at <https://hpln.hp.com/group/best-practices-hpsw> to access a wide variety of best practice documents and materials.

Contents

HP Business Process Testing User Help	8
Part 1: Business Process Testing Basics	9
Chapter 1: Business Process Testing Introduction	11
Overview	12
Identifying Roles	13
Choosing Design Approaches	15
Choosing Methodologies	18
Categorizing Components	22
Working with Business Process Testing	24
Integration with other HP Products	24
Working with Unified Functional Testing	25
Working with Sprinter	27
Working with BPT Packaged Apps Kit	28
Part 2: Business Components	33
Chapter 2: Introducing Business Components	35
Business Components Overview	36
Components with Manual Content	37
Components with Automated Content	38
Application Area Overview	39
How to Create Business Components in ALM	41
Business Components User Interface	45
Business Components Module Window	46
Business Components Module Menus and Buttons	48
Business Process Testing Icons	52
Business Components Module Fields	54
New Component/Component Details Dialog Box	56
Snapshot Tab	58
Paste Components/Component Folders to Target Project Dialog Box	58
Select Destination Folder Dialog Box	59
Chapter 3: Components with Manual Implementation	61
Manual Implementation Overview	62
How to Work with Manual Implementation	62
Manual Implementation User Interface	64
Manual Implementation Tab	65
Manual Implementation Pane/Component Step Details Dialog Box	65
Chapter 4: Components with Keyword GUI Automation	69
Keyword GUI Automation Overview	70
How to Work with Keyword GUI Automation	73
Keyword GUI Automation User Interface	77

Automation Tab	78
Keyword View Pane	78
Select Application Area Dialog Box	83
Keyword View Options Dialog Box	83
Select Test Object Dialog Box	85
Output Options Dialog Box	85
Value Configuration Options Dialog Box	86
Checkpoint Properties Dialog Box	87
Parameterization / Properties Dialog Box	88
Output Value Properties Dialog Box	89
About Dialog Box	90
Location Information Dialog Box	90
Chapter 5: Components with Scripted GUI or API Automation	93
Scripted GUI and API Automation Overview	94
How to Work with Scripted GUI and API Automation	94
Scripted GUI and API Automation User Interface	95
Script Launch Pane	96
Part 3: Business Process Tests and Flows	97
Chapter 6: Introducing Business Process Tests and Flows	99
Business Process Test and Flow Overview	100
How to Plan Business Process Tests and Flows	100
Chapter 7: Business Process Test and Flow Specification	103
Business Process Test and Flow Specification Overview	104
How to Create Business Process Tests and Flows	104
How to Record Components into Flows and Business Process Tests	108
How ALM Calculates Business Process Test and Flow Statuses	111
Business Process Test and Flow Specification User Interface	112
Test Plan Module Window for Business Process Testing	113
Test Plan Module Menu and Buttons for Business Process Testing	114
Test Plan Module Icons for Business Process Testing	115
Test Plan Module Fields for Business Process Testing	116
Test Script Tab for Business Process Testing	117
Test Details Dialog Box for Business Process Testing	126
Test Validation Results Dialog Box	127
Chapter 8: Learn Process	129
Learn Process Overview	130
How to Learn Flows and Business Process Tests	130
Criteria for Determining if a Component Can Be Reused	134
Learn Flow Process User Interface	136
Available Components for Reuse Dialog Box	137
Learn Wizard	137
Learn Toolbar Page	138
Learn Summary Page	139
Chapter 9: Requirement Coverage	143
Requirement Coverage Overview	144

How to Create Coverage by Criteria	145
Chapter 10: Run Conditions	147
Run Conditions Overview	148
How to Set Run Conditions	148
Run Conditions User Interface	149
Run Conditions Dialog Box	150
Chapter 11: Business Process Test Configurations	153
Business Process Test Configurations Overview	154
Chapter 12: Test and Flow Debugging	157
Test and Flow Debugging Overview	158
How to Debug Tests and Flows Containing Automated Components	158
Test and Flow Debugging User Interface	160
Run or Debug Test Dialog Box	161
Debug Run Information Dialog Box	161
Part 4: Data Handling	163
Chapter 13: Introducing Data Handling (Parameters)	165
Data Handling Overview	166
Parameter Categories	169
How to Handle Data in Business Process Testing	170
Chapter 14: Parameters	173
Parameters Overview	174
How to Create Parameters	174
How to Set Parameter Values	175
Methods for Creating Parameters	177
Parameters User Interface	178
Parameters Page	179
Set Values Dialog Box	182
Dynamic Date Area	184
Parameters Dialog Box	186
New Parameter / Parameter Details Dialog Box	187
Promotion Results Dialog Box	189
Edit Table Parameter Dialog Box	190
Chapter 15: Parameter Linkage	193
Parameter Linkage Overview	194
Linkage and Iterations	194
Data Linkage Examples	195
How to Link Data	198
Parameter Linkage User Interface	200
I/O Parameters Dialog Box	201
Select Output Parameter Dialog Box	201
Chapter 16: Parameter Promotion	203
Parameter Promotion Overview	204
How to Promote Parameters	205
Parameter Promotion User Interface	206
Promote Parameters Dialog Box	207

Test/Flow Parameters Promotion Status Dialog Box	208
Chapter 17: Iterations	209
Iterations Overview	210
Group Iterations Overview	212
How to Set Data for Iterations	214
How to Import and Export (Save) Parameter Values	216
Iterations User Interface	218
Iteration Pages	219
Select Iterations Dialog Box	221
Map Imported Data Dialog Box	222
Create Flow/Test Parameters Dialog Box	223
Part 5: Test Runs	225
Chapter 18: Introducing Test and Flow Execution	227
Test and Flow Execution Overview	228
How to Run Business Process Tests and Flows Manually	228
How to Run Automated Business Process Tests and Flows	229
Run Progress Pane Hierarchy	232
Test and Flow Execution User Interface	234
Manual Runner Wizard for Business Process Testing	235
Manual Runner: Run Details Page for Business Process Testing	235
Manual Runner: Step Details Page for Business Process Testing	236
Test Properties Dialog Box	238
Chapter 19: Change Detection and Resolution	239
Change Detection and Resolution Overview	240
How to Run Tests in Change Detection Mode	241
How to View and Resolve Detected Changes	243
Change Detection and Resolution User Interface	244
Change Detection Report Screen	245
Part 6: Appendices	247
Appendix A: BPT Packaged Apps Kit App-specific Information	249
BPT Packaged Apps Kit App-specific Information Overview	250
How to Work with BPT Packaged Apps Kit with Support for SAP Apps	250
How BPT Packaged Apps Kit Creates Components when Learning SAP GUI Flows	251
How BPT Packaged Apps Kit Creates Components when Learning SAP Fiori Flows	253
BPT Packaged Apps Kit App-specific User Interface	253
An Example of a SAP-specific Change Detection Report	254
Send Us Feedback	257

HP Business Process Testing User Help

Welcome to HP Business Process Testing. Business Process Testing works within HP Application Lifecycle Management (ALM) as a component-based testing framework. Working with a testing framework provides many advantages to enterprises, including streamlining the creation and maintenance of both manual and automated tests, and maximizing efficiency for testing complete business processes.

Because of its flexibility, Business Process Testing can be used by various personas with varying levels of programming experience, subject matter knowledge, and business expertise. The framework can be implemented differently based on enterprise needs.

This help describes how to use the ALM application to work with HP Business Process Testing. This help presents some basic implementation methods, but for more details, see the Business Process Testing technical white papers available under **Resources** at the Business Process Testing [official site](#).

Part 1: Business Process Testing Basics

Welcome to HP Business Process Testing. Business Process Testing works within HP Application Lifecycle Management (ALM) as a component-based testing framework. Working with a testing framework provides many advantages to enterprises, including streamlining the creation and maintenance of both manual and automated tests, and maximizing efficiency for testing complete business processes.

Because of its flexibility, Business Process Testing can be used by various personas with varying levels of programming experience, subject matter knowledge, and business expertise. The framework can be implemented differently based on enterprise needs.

The *HP Business Process Testing User Guide* describes how to use the ALM application to work with HP Business Process Testing. This help presents some basic implementation methods, but for more details, see the Business Process Testing technical white papers available under **Resources** at the Business Process Testing [official site](#).

Chapter 1: Business Process Testing

Introduction

This chapter includes:

- Overview 12
- Identifying Roles 13
- Choosing Design Approaches 15
- Choosing Methodologies 18
- Categorizing Components 22
- Working with Business Process Testing 24
- Integration with other HP Products 24

Overview

This chapter introduces Business Process Testing, provides information about basic implementation techniques, and describes workflows for working with Business Process Testing.

This chapter also introduces BPT Packaged Apps Kit for learning components and detecting changes in packaged apps, such as SAP products.

Note:

- For details on configuring user permissions, see the *HP Application Lifecycle Management Administrator Guide*.
- For details on installing and setting up BPT Packaged Apps Kit, 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 more information about HP Business Process Testing editions and their functionality, see the *HP Application Lifecycle Management User Guide*. To find out what edition of HP Business Process Testing you are using, ask your ALM site administrator.

The most efficient way to manage functional testing activities is to work in a consistent way using a testing framework. Business Process Testing provides you with a customizable, component-based testing framework that supports:

Automated and manual testing

- You can create both manual and automated steps for each component, and if desired, synchronize between the steps.
- You can migrate from manual to automated testing.
- You can use a keyword GUI method for creating automation without coding. This method generates easy-to-understand test documentation automatically.

Component reuse and modularization

Component reuse and modularization keep costs low by speeding up test creation, maintenance, and execution.

Creation of tests for both simple and complex applications

An application under test can be a simple, HTML-based web application or a complex business process involving packaged applications such as SAP or Oracle, and back-end services and databases.

Collaboration between various personas

The testing framework is flexible enough to meet the needs of various personas, such as manual testers, automation engineers, and subject matter experts.

Business Process Testing helps you document your components and tests, including screenshots illustrating how they should be used, and so on. This makes it easy for people with different roles and skill sets to share each others' assets.

Examples

- Business process tests can be created by non-technical subject matter experts who best understand the business processes that need to be tested, without the need for a programming background.
- Automation engineers can prepare object hierarchies and application-specific functions and then package these hierarchies and functions as building blocks for use in business process tests by subject matter experts.

Management of the parts of a test

Managing parts of a test includes component documentation, test execution results, version control, reporting, and history. Additionally, you can generate documents containing information about the tests, flows, and components in a project.

Quick and easy design of tests early in the development cycle and in a script-free environment

You can define keywords and components before the application has been delivered for testing, so you can design tests earlier in the software development lifecycle.

You can test applications manually before automated tests are ready. This facilitates quick implementation of business process tests.

Integration with HP's test management and testing tools, such as ALM, HP Sprinter, UFT, and HP Service Test

Automation engineers can access Business Process Testing functionality directly from their testing tool, in an environment with which they are familiar.

ALM users can create business process tests in an environment with which they are familiar.

Identifying Roles

Business Process Testing can be used by several different personas, each with varying levels of experience and different goals.

When working with Business Process Testing, roles are flexible. There are no product-dictated rules controlling which types of users can perform which Business Process Testing tasks (provided that users have the correct permissions).

Note: For details on assigning permissions, see the *HP Application Lifecycle Management Administrator Guide*.

The following table describes various roles that can be used when working with Business Process Testing.

Role	Description
Subject Matter Experts	<p>Subject matter experts have specific knowledge of the application under test's 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 are likely to:</p> <ul style="list-style-type: none"> • Determine the business processes to test. • Identify activities common to multiple processes (such as a login procedure, which is used in many business process tests or flows). • Create business components and manual steps. • Define automated keyword GUI component steps. • Create flows and business process tests.
Automation Engineers, also known as Test Automation Experts	<p>Automation engineers are experts in automated testing using a testing tool such as UFT. The automation engineer is only needed if the Business Process Testing framework includes automated tests.</p> <div style="background-color: #f0f0f0; padding: 5px; margin: 10px 0;"> <p>Note: Automation engineers can work directly in Business Process Testing or access Business Process Testing functionality from within UFT.</p> </div> <p>Automation engineers are likely to:</p> <ul style="list-style-type: none"> • Define automated keyword GUI component steps. • Define automated GUI scripted components. • Define automated API scripted components. • Prepare resources required for testing features: <ul style="list-style-type: none"> • Application areas, which can be defined within ALM and other testing tools. • Function libraries with general scripts, which are encapsulated into general operation keyword GUI steps. • Shared object repositories, representing the objects in the application being tested. You can use these objects to create steps in business components with keyword GUI automation. <p>Automation engineers may also be responsible for some of the tasks listed for the subject matter expert.</p> <p>Automation engineers can also create, debug, and modify business components in the testing tool.</p>
QA Testers	<p>QA testers are likely to:</p> <ul style="list-style-type: none"> • Define data for iterations and configurations. • Run tests to verify that they are designed appropriately and run as expected. • Review test results.
Test Architects	<p>Test architects design and implement the testing framework. Test architects are likely to:</p> <ul style="list-style-type: none"> • Determine the design approach. • Decide how to categorize, name, organize, and use components. • Set up standards for using Business Process Testing in the various ALM modules.
ALM Administrators	<p>Set up and configure Business Process Testing and its users.</p>

Choosing Design Approaches

The Business Process Testing testing framework 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.

Available approaches include:

- ["Bottom-up Approach" below](#)
- ["Top-down Approach" on the next page](#)
- ["Agile Development-centric Approach" on page 17](#)

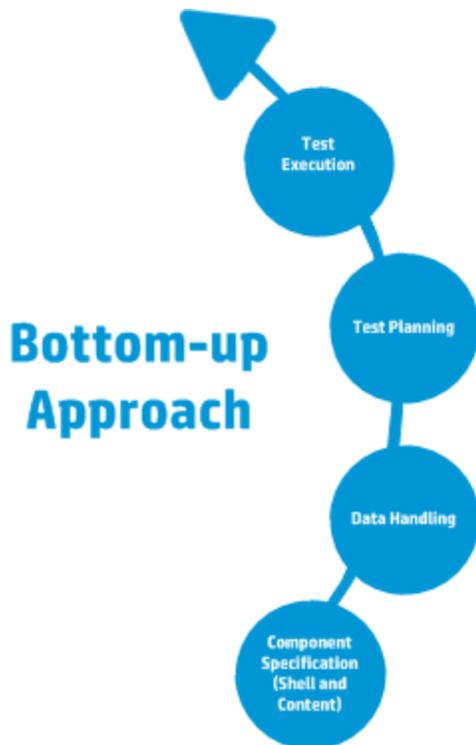
Note: The chapters in this guide are structured according to the bottom-up approach.

Bottom-up Approach

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

- For regression testing
- When the business processes in the organization are clearly defined
- When users are new to Business Process Testing

The bottom-up approach includes the following phases in the following order:



For a description of each phase, see ["Working with Business Process Testing" on page 24](#).

For an example of a common workflow using BPT Packaged Apps Kit, see "[Working with BPT Packaged Apps Kit](#)" on page 28.

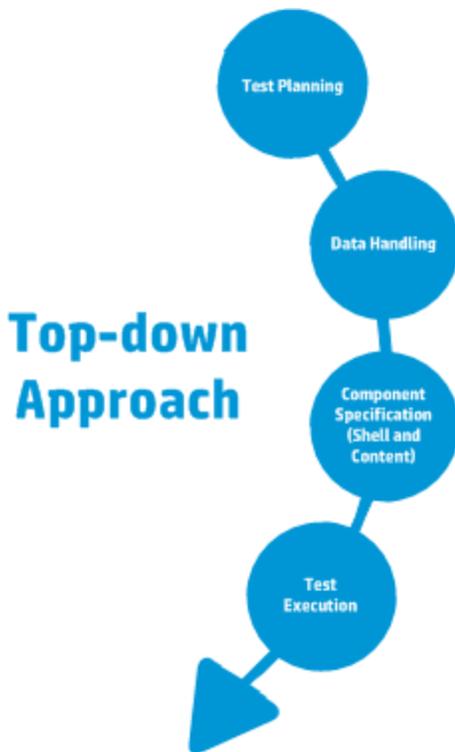
Top-down Approach

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

The top-down approach advocates the creation of business process testing entities for regression testing 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 approach includes the following phases in the following order:



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

Design Phase	Description
High-level design	<p>Includes creating business process tests, and determining the test configurations needed for different use-cases.</p> <p>When designing at the high-level:</p> <ul style="list-style-type: none">• Facilitate automation by designing modularly. Design tests to use smaller, reusable components.• Use reusable components, which makes maintaining tests easier.• Keep in mind that automation engineers can use the design as a structure for creating methods that

Design Phase	Description
	<p>facilitate automation, instead of for creating scripted components.</p> <p>This part of the design phase is often done by both the subject matter expert and the automation engineer together. For example, creating the business process tests and their configurations might be performed by the subject matter expert, while designing automated components might be performed by the automation engineer.</p>
<p>Mid-level design</p>	<p>Includes the:</p> <ul style="list-style-type: none"> • 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). 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. • Parameterizing and promoting parameters. • Adding business components to business process tests and flows. <p>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.</p>
<p>Low-level implementation</p>	<p>Includes the low-level implementation of business component content by:</p> <ul style="list-style-type: none"> • 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) • Linking parameters <p>This part of the design phase can be performed by the subject matter expert, the automation engineer, or both together.</p>

Agile Development-centric Approach

This approach is based on using Business Process Testing to provide testing in sprints, as developers code features for the application under test. Components and tests are created and updated in parallel with development.

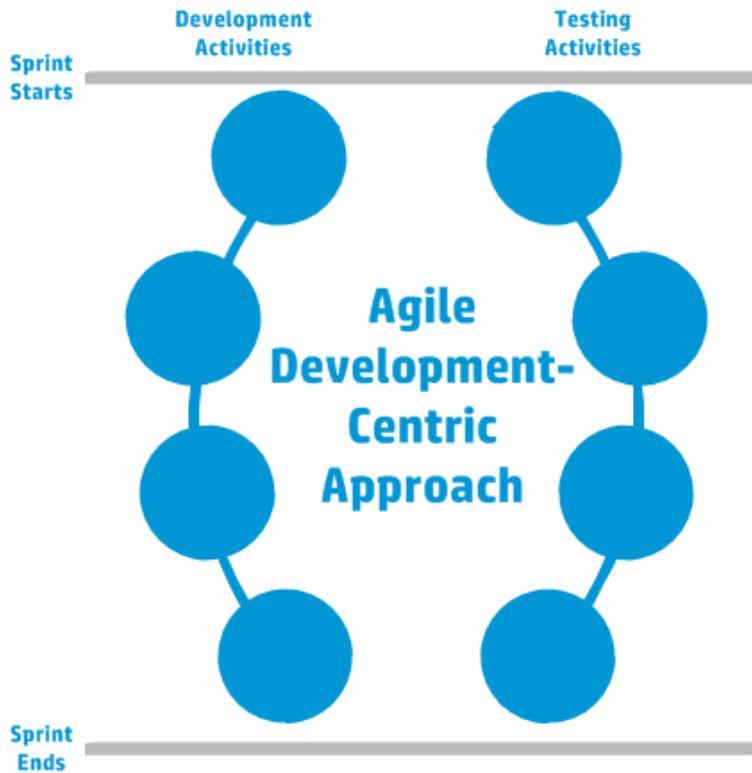
Example

If the application under test is implemented in Java, components might be grouped by the classes that represent certain groups of UI elements, such as toolbar buttons. Each time a button is added to the toolbar, the component representing that class is updated.

This approach encourages:

- **Automation.** Because sprints are short, it is important to automate as much as possible.
- **Component reuse.** Component reuse can be designed in the same way that the developers implement modularly for reuse.

The following presents the Agile Development-centric approach.



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.

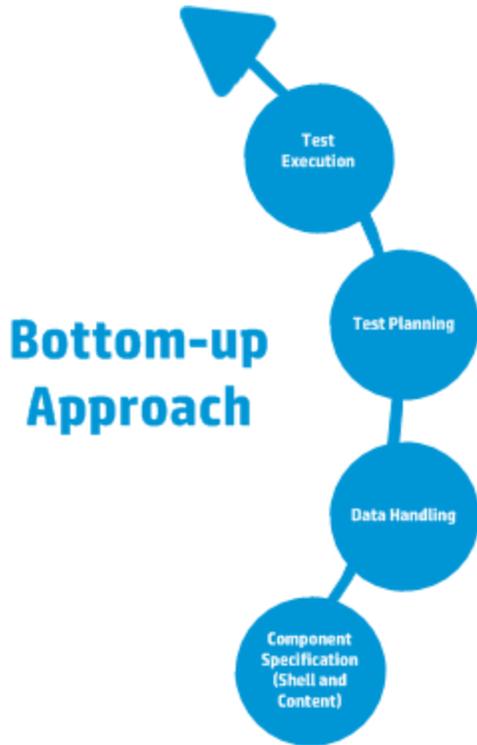
The chapters in this guide are structured according to the bottom-up methodology.

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 bottom-up methodology includes the following phases in the following order:



Phase	Description
Component Specification	Develop a component tree with components. Create the component shell by adding basic details. Create component content by adding manual and/or automated implementations. Component content can contain: <ul style="list-style-type: none"> • Manual implementation for manual components • Automation, for automated components • Both manual implementation and automation For task details on creating components, see "How to Create Business Components in ALM" on page 41 .
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 100 .
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 170 .
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 228 and "How to Run Automated Business Process Tests and Flows" on page 229 .

For an example of a common workflow using BPT Packaged Apps Kit, see ["Working with BPT Packaged Apps Kit" on page 28](#).

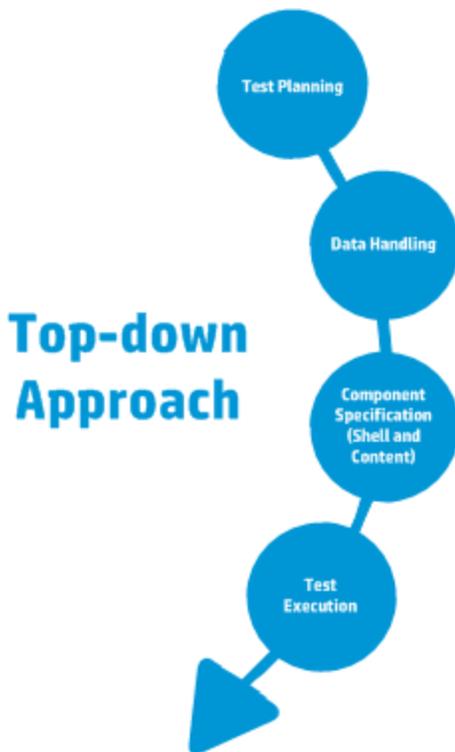
Top-down Methodology

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 includes the following phases in the following order:



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

Design Phase	Description
High-level design	<p>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.</p> <p>When designing at the high-level, facilitate automation:</p> <ul style="list-style-type: none"> • 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. <p>This part of the design phase is often done by both the subject matter expert and the automation engineer together.</p>

	Creating the structure for business process tests and determining the needed test configurations are typically performed by the subject matter expert.
Mid-level design	<p>Includes the:</p> <ul style="list-style-type: none"> • 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. <p>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.</p>
Low-level implementation	<p>Includes the low-level implementation of business component content by:</p> <ul style="list-style-type: none"> • 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 <p>This part of the design phase can be performed by the subject matter expert, the automation engineer, or both together.</p>

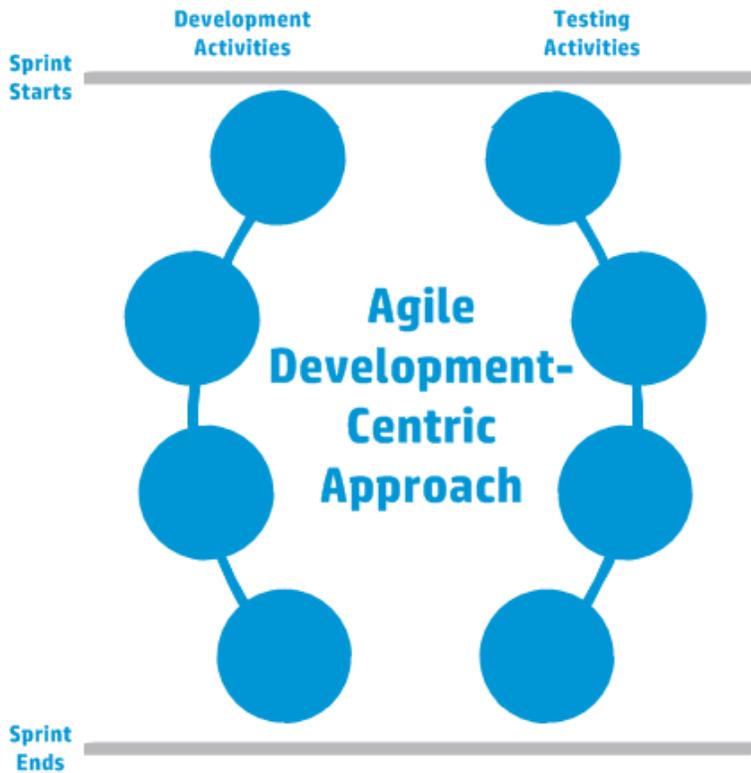
Agile Methodology

The Agile methodology is based on using Business Process Testing to providing testing in sprints, as developers code features for the application under test.

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

- Components are created corresponding to features or UI elements developed during the sprint. For example, if developers design and create a new window that has a menu, a navigation pane, and a main data area, you might create three components corresponding to these UI elements.
- Tests and flows, which contain the sprint's business components, are maintained so at the end of any sprint, the developed features can be tested immediately.

The Agile methodology includes the phases by sprint.



Categorizing Components

Because Business Process Testing is a component-based testing framework, components are largely responsible for driving the system being tested. The framework encourages component design and reuse, so the method you use to categorize your components has a large impact on the ability of your framework to manage your testing abilities successfully.

Consider the following methods for categorizing components .

- ["Logical Components" on the next page](#)
- ["Application Object Components" on the next page](#)
- ["Generic Components" on the next page](#)

Logical Components

A logical component represents the use of a part of the screen with one or more controls, or a set of API calls which combine to perform some application logic. This category is based on a specific context in the application under test.

Examples

- A **Login** component represents the login process, based on a login window that allows you to enter a user name and password, and then click a **Login** button.
- A **Search** component represents search for an entity in the application under test. You can enter a string for which to search, indicate capitalization and/or whole word options, and click a **Search** button.

Application Object Components

An application object component might represent an object on the screen or a call to a single API.

This category is usually independent of the context within the application under test, and can be used in many situations. You decide the level of granularity that most encourages reuse.

Examples

- A **Button** component represents the button object.
- A **Grid** component represents a grid object in a pane or window.
- A **Pane** component represents a pane in a window or screen.
- An **Interrogate** component represents the interrogation of the application under test's backend database.

Generic Components

A generic component performs actions outside of the context of the application under test. It can be reused in tests of different applications.

Example

- A **Launch** component represents the launching of a browser.

Note: Flows can be thought of as complex components or small business component tests. Flows comprise a set of components in a fixed sequence to perform a specific task. A flow can be part of a test just like any other component, but when the flow runs, Business Process Testing executes the components that the flow contains.

Working with Business Process Testing

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

Phase	Description
Component Specification	<p>Develop a component tree with components.</p> <p>Create the component shell by adding basic details.</p> <p>Create component content by adding manual and/or automated implementations. Component content can contain:</p> <ul style="list-style-type: none">• Manual implementation for manual components• Automation, for automated components• Both manual implementation and automation <p>For task details on creating components, see "How to Create Business Components in ALM" on page 41.</p>
Data Handling	<p>Design the data that each business process test, flow, or component uses when run.</p> <p>For task details, see "How to Handle Data in Business Process Testing" on page 170.</p>
Test Planning	<p>Build test plans and design business process tests and flows.</p> <p>For task details, see "How to Plan Business Process Tests and Flows" on page 100.</p>
Test Execution	<p>Create a subset of the business process tests in your project and run them.</p> <p>For task details, see "How to Run Business Process Tests and Flows Manually" on page 228 and "How to Run Automated Business Process Tests and Flows" on page 229.</p>

For an example of a common workflow using BPT Packaged Apps Kit, see ["Working with BPT Packaged Apps Kit" on page 28](#).

Integration with other HP Products

Business Process Testing integrates with other HP products:

Product	Integration
HP Unified Functional Testing (UFT)	<p>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.</p> <p>HP Unified Functional Testing (UFT) comprises the product formerly known as HP QuickTest Professional and the product known as HP Service Test.</p> <ul style="list-style-type: none">• Functionality provided by QuickTest is now known as GUI testing in UFT.• Functionality provided by Service Test is also known as API testing in UFT. <p>For an overview of how Business Process Testing works with UFT, see "Working with Unified Functional Testing" on the next page.</p> <p>Tip: Business Process Testing and UFT are so tightly integrated that much of Business Process Testing's functionality is available directly within UFT.</p>

Product	Integration
HP Service Test	Automate business components using API testing. These components can then be included in automated tests that use the Service Test testing tool, which is a functional testing tool for testing headless applications, such as web services. The Service Test testing tool is also available with Unified Functional Testing (UFT).
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 "Working with Sprinter" on page 27 .
BPT Packaged Apps Kit	BPT Packaged Apps Kit adds additional functionality to ALM and Business Process Testing that enables you to test packaged applications. This functionality includes features such as learning flows and change detection. For an overview of how Business Process Testing works with BPT Packaged Apps Kit, see "Working with BPT Packaged Apps Kit" on page 28 .

Working with Unified Functional Testing

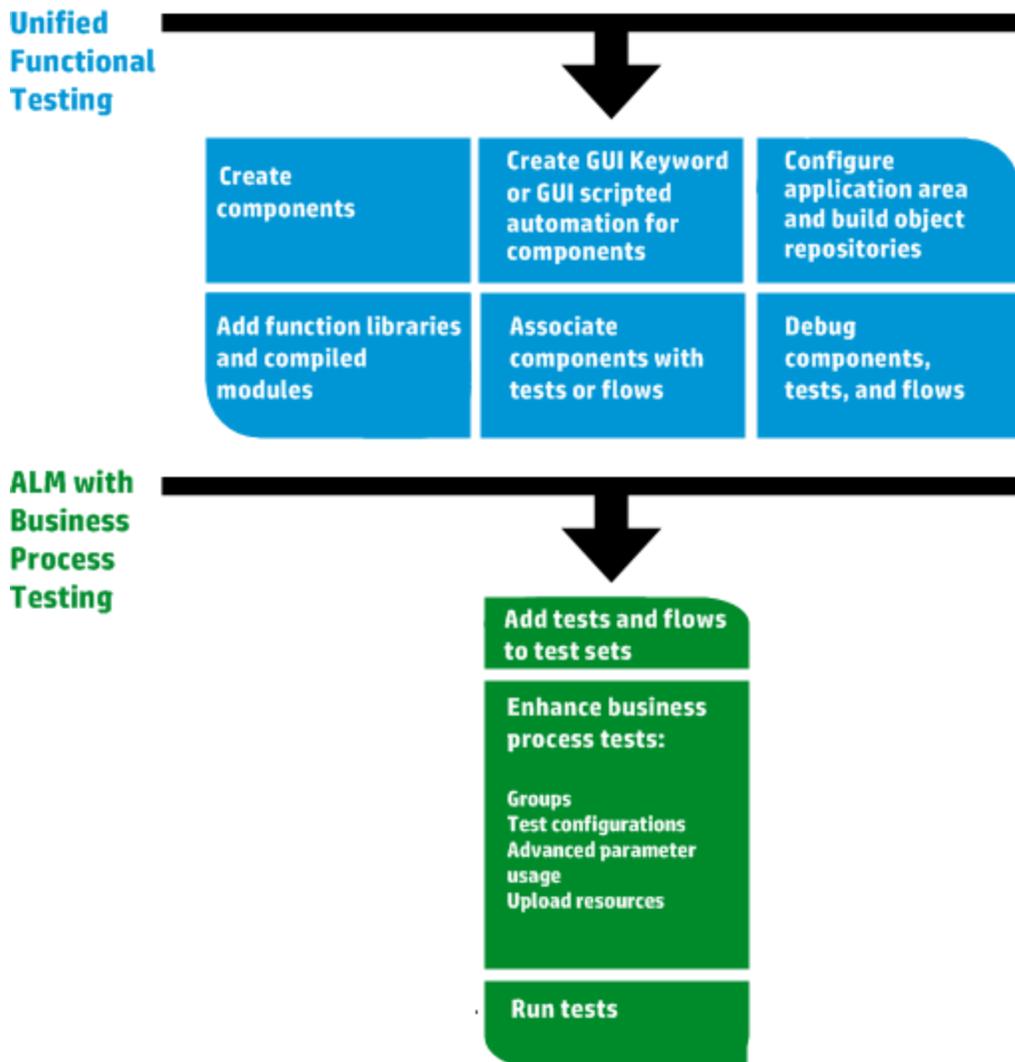
There are many ways to work with the UFT testing tool and Business Process Testing.

This section includes:

- ["Automation Engineers Working Primarily in UFT" below](#)
- ["Users Working in Their Respective Tools, UFT and ALM" on the next page](#)

Automation Engineers Working Primarily in UFT

This example demonstrates the ease with which Automating Engineers can access Business Process Testing functionality from directly within UFT, the tool with which they are most familiar.

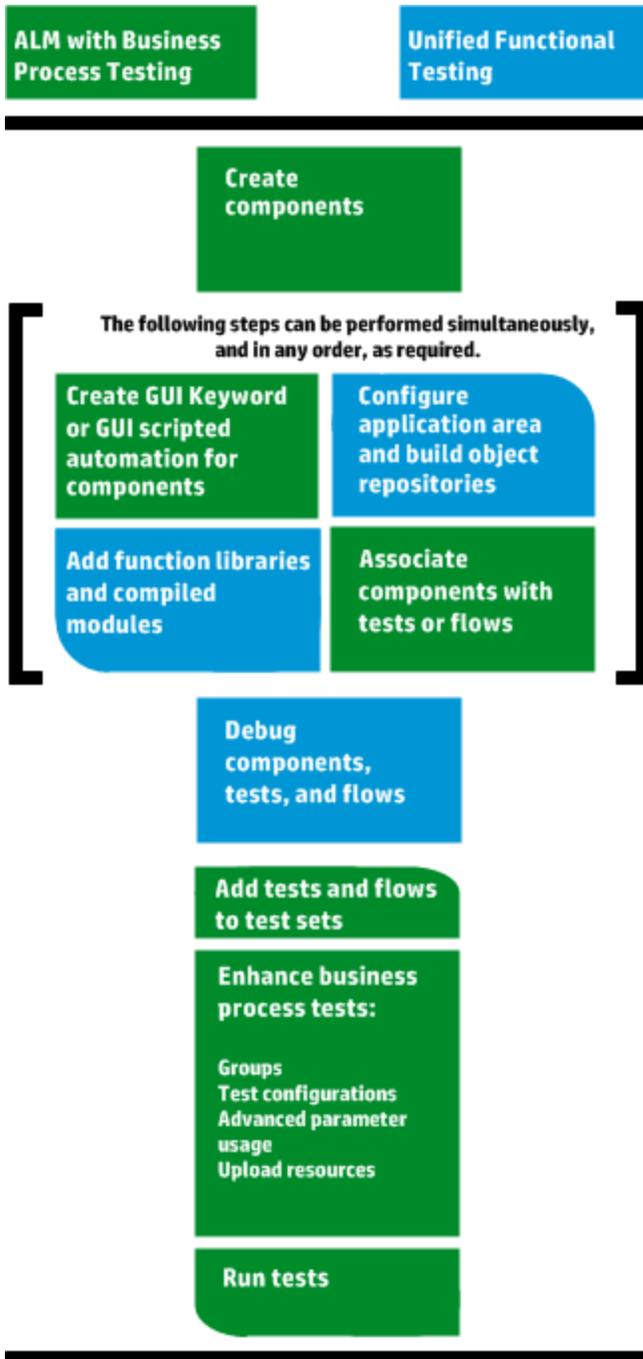


Users Working in Their Respective Tools, UFT and ALM

This example demonstrates the collaboration between Automating Engineers working primarily in UFT and subject matter experts working in Business Process Testing.

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.

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



Working with Sprinter

You can run tests manually from Business Process Testing using HP Sprinter.

For full details on installing and working with Sprinter, refer to the *HP Sprinter README* and the *HP Sprinter User Guide*, available on the HP Application Lifecycle Management Add-ins page.

This section describes the following topics:

- ["General Sprinter Advantages" below](#)
- ["Sprinter Advantages when Working with Business Process Testing" below](#)
- ["Sprinter Limitations when Working with Business Process Testing" below](#)

General Sprinter Advantages

Because both Sprinter and Business Process Testing are fully integrated with ALM, you can benefit from all Sprinter functionality, including:

- creating and annotating screen captures
- capturing movies of your run
- recording and running macros on your test application
- automatically entering data into fields in your application
- automatically including the list of your steps or user actions in any defect you submit
- replicating your user actions on multiple machines with different configurations

You can use all this functionality without interrupting the flow of your manual business process test.

Sprinter Advantages when Working with Business Process Testing

Running Business Process Testing manual tests with Sprinter provides the following additional benefits:

- You can view step information, navigate steps, and modify step information.
- The test hierarchy, including components and iterations, is displayed.
- You can view, and edit, the parameters in your steps.
- You can advance to previous/next components in a business process test.
- You can create (author) manual components directly in Sprinter.

Note: For a list of ALM and Sprinter versions that support creation of manual components from Sprinter, see the *ALM What's New* document.

Sprinter Limitations when Working with Business Process Testing

- Unlike other ALM tests, you cannot export, print, or email steps in a business process test.

Working with BPT Packaged Apps Kit

Available for BPT Packaged Apps Kit Users

BPT Packaged Apps Kit adds additional functionality to ALM and Business Process Testing that enables you to test packaged applications. This functionality includes features such as learning flows and business process tests, and also change detection.

This section describes the following topics:

- ["Prerequisites" below](#)
- ["Benefits" below](#)
- ["Roadmap " on the next page](#)

Prerequisites

To work with BPT Packaged Apps Kit:

- Make sure Unified Functional Testing is installed on the client machine along with its SAP add-in and ALM plug-in.
- BPT Packaged Apps Kit is enabled in ALM Customization.

For details about enabling BPT Packaged Apps Kit, see the *HP Application Lifecycle Management Administrator Guide*.

Note: When using BPT Packaged Apps Kit, it is best to work with only once instance of ALM open. Do not open multiple instances of ALM in multiple browsers or tabs.

Benefits

BPT Packaged Apps Kit includes the following benefits:

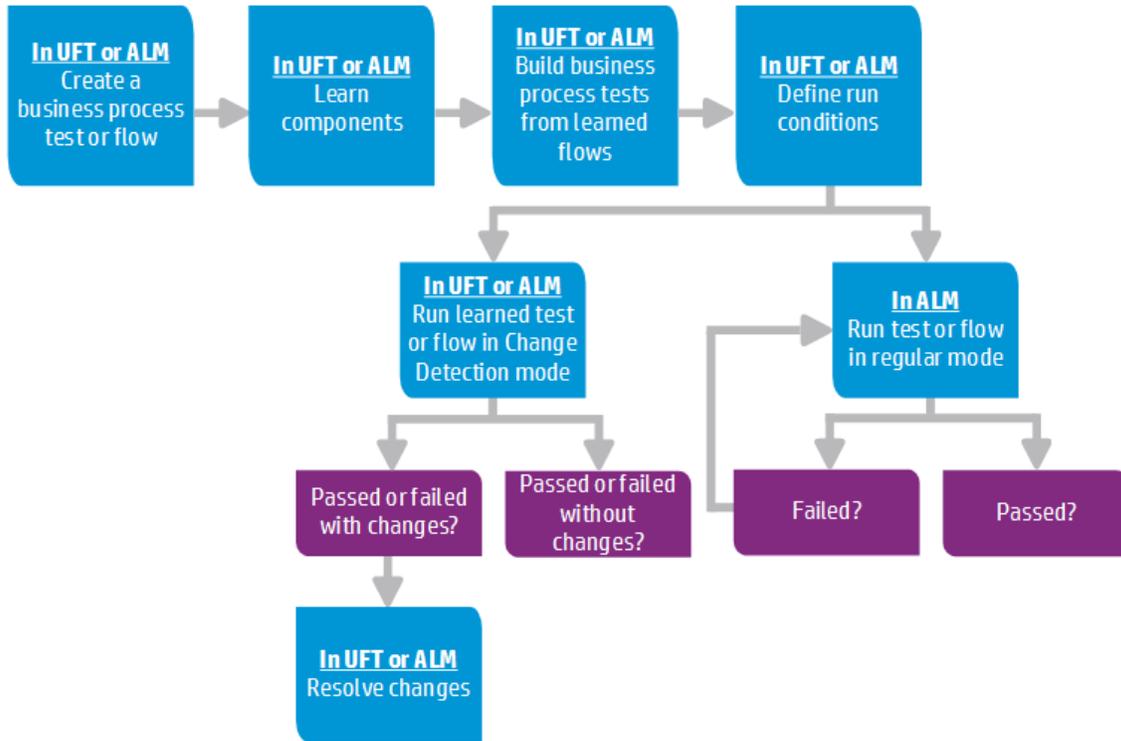
- BPT Packaged Apps Kit can automatically learn the actions you perform on your applications, and then generate a flow or business process test containing components based on the screens and transactions within your applications. You can reuse learned components instead of creating new components.
- BPT Packaged Apps Kit 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 BPT Packaged Apps Kitapps, see ["BPT Packaged Apps Kit App-specific Information Overview" on page 250](#).

For details about enabling BPT Packaged Apps Kit, see the *HP Application Lifecycle Management Administrator Guide*.

Roadmap

The roadmap with BPT Packaged Apps Kit 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 and Business Process Tests	You create flows and business process tests in the Test Plan module. For task details, see "How to Create Business Process Tests and Flows" on page 104.
Learn Components	BPT Packaged Apps Kit enables you to learn the steps that comprise a transaction as you navigate your packaged app. BPT Packaged Apps Kit creates automated business components corresponding to screens and transactions in your app, and creates automated component steps based on the operations you perform. These components are then added to the flow or business process test. During the Learn process, you can insert checkpoints and output values. For task details, see "How to Learn Flows and Business Process Tests" on page 130.
Build Tests from Learned Flows	In addition to business process tests that you learned, you can also build a business process test from the flows that were created in the Learn 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 104.
Define Run	You can use run conditions to determine whether to run a component or flow, skip to the next component

Phase	Description
Conditions	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 148.
Run Tests in Regular Mode	You run a business process test containing learned components and flows as you would a regular business process test. For task details, see "How to Run Business Process Tests and Flows Manually" on page 228.
Run Tests in Change Detection Mode	If your test includes flows learned automatically, or your test itself was learned automatically, you can run the test in Change Detection mode. BPT Packaged Apps Kit checks whether elements in your packaged app changed since the flow or test was learned. This enables you to later update your flows and tests to reflect these changes. For task details, see "How to Run Tests in Change Detection Mode" on page 241.
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 test, flow, business component, and component step in your test, including whether objects used in your steps have changed in your app. If changes were detected during the test run, you can view the changes in UFT and ALM. 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 app. For task details, see "How to View and Resolve Detected Changes" on page 243.

Part 2: Business Components

Chapter 2: Introducing Business Components

This chapter includes:

- Business Components Overview36
- Components with Manual Content 37
- Components with Automated Content38
- How to Create Business Components in ALM 41
- Business Components User Interface45

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 41](#).
- **Content** (steps or scripts). Low-level, detailed information, such as the component's manual steps and/or automation. The contents provide detailed instructions for performing business process tasks in the application. Component content can be manual, automated, or both—depending on whether you create a manual implementation and/or automation for the component.
 - For task details on creating manual components, see ["How to Work with Manual Implementation" on page 62](#).
 - For task details on creating automated components, see ["How to Work with Keyword GUI Automation" on page 73](#) and ["How to Work with Scripted GUI and API Automation" on page 94](#).

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 148](#).

When BPT Packaged Apps Kit learns a flow or test, it creates a new business component for each screen or tab through which you navigate in your applications. BPT Packaged Apps Kit also enables you to identify and reuse learned components instead of creating new components in your flow/test. For task details on learning, see ["How to Learn Flows and Business Process Tests" on page 130](#).

Components with Manual Content

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

Manual 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 Implementation" on page 62](#).

For user interface details, see ["Manual Implementation User Interface" on page 64](#).

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" on the next page](#)

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 the 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 73 . 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 <i>HP Unified Functional Testing User Guide</i> .
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 94 .
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 94 . Note: API scripted automation is supported for HP API 11.10 or later. For details, see the <i>HP Service Test User Guide</i> or the information about API testing in the <i>HP Unified Functional Testing User Guide</i> .

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 the 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, available from the HP Application Lifecycle Management Add-ins page (**Help > Add-ins**).

The 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.

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 (manual implementation or automation) in ALM.

Note:

- This task is part of a higher-level task. For details, see ["Working with Business Process Testing" on page 24](#).
- **Product Feature Movie:** To view a movie that demonstrates basic information about Business Process Testing, select **Help > Movies** in the ALM main window.

This task includes the following steps:

- ["Prerequisites" below](#)
- ["Create the component tree" below](#)
- ["Create the component shell" below](#)
- ["Update the component shell" on the next page](#)
- ["Create component content" on the next page](#)
- ["Synchronize component content - optional" on the next page](#)
- ["Handle component requests - optional" on page 44](#)
- ["Convert manual tests - optional" on page 44](#)
- ["Create components by recording - optional" on page 44](#)
- ["Create components by learning packaged apps, such as SAP - optional" on page 44](#)
- ["Create components using HP Sprinter - optional" on page 44](#)
- ["Analyze components" on page 44](#)

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 prepare the component for the creation of manual 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 46](#).

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 56](#).

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 the next page](#).
- When learning flows using BPT Packaged Apps Kit. For user interface details, see ["Learn Flow Process User Interface" on page 136](#).
- Copying components from other projects. For user interface details, see ["Paste Components/Component Folders to Target Project Dialog Box" on page 58](#)
- Handling component requests. For task details, see ["Handle component requests - optional" on the 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 assuming a manual implementation is needed for adding manual steps. You can change this default if you prefer that the component be created without preparation for manual implementation. 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, manual 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 56](#).

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 18](#).

For task details on adding manual steps to the component content, see ["How to Work with Manual Implementation" on page 62](#).

For task details on adding keyword GUI automation to the component content, see ["How to Work with Keyword GUI Automation" on page 73](#).

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 94](#).

6. Synchronize component content - optional

If your business component contains both manual implementation and automation, you can synchronize the business component's manual implementation steps and its automated steps. This guarantees that the two implementations are consistent. When synchronized, you cannot edit manual implementation steps.

For user interface details, see the **Sync to Automation** button in ["Manual Implementation Tab" on page 65](#).

7. 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.
- You cannot automate a component request. After the request becomes a component, you can add automation.
- If you add a component request, which by default only has a manual implementation, into an automated business process test, the business process test can still be run automatically. The component request is ignored during the automatic run.

For task details on requesting components, see the task ["How to Create Business Process Tests and Flows" on page 104](#).

8. 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 59](#).

9. Create components by recording - optional

You can record your actions in any application. These actions are then used as steps in newly-created business components. For details, see ["How to Record Components into Flows and Business Process Tests" on page 108](#).

10. Create components by learning packaged apps, such as SAP - optional

You can learn actions in a packaged app, such as SAP. These actions are then created as components in newly-created business process tests or flows. For details, see ["Learn Process Overview" on page 130](#).

11. Create components using HP Sprinter - optional

You can use Sprinter to define manual business components. For details, see the Sprinter documentation.

12. 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 example, to see the

development status of your components, you could view the Components Progress graph. For task details, see the information about analyzing data in the *HP Application Lifecycle Management User Guide*.

Business Components User Interface

This section includes:

- [Business Components Module Window](#) 46
- [Business Components Module Menus and Buttons](#) 48
- [Business Process Testing Icons](#) 52
- [Business Components Module Fields](#) 54
- [New Component/Component Details Dialog Box](#) 56
- [Snapshot Tab](#) 58
- [Paste Components/Component Folders to Target Project Dialog Box](#) 58
- [Select Destination Folder Dialog Box](#) 59

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	<ul style="list-style-type: none"> • "How to Create Business Components in ALM" on page 41 • "How to Work with Manual Implementation" on page 62 • "How to Work with Keyword GUI Automation" on page 73 • "How to Work with Scripted GUI and API Automation" on page 94
See also	"Business Components Overview " on page 36

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

UI Element	Description
<Business Components module common UI elements>	<ul style="list-style-type: none"> • Business Component fields. For field definitions, see "Business Components Module Fields" on page 54. • Business Component menus and buttons. For command and button descriptions, see "Business Components Module Menus and Buttons" on page 48. • Business Component icons. For icon descriptions, see "Business Process Testing Icons" on page 52. • ALM masthead, sidebar, and menus. For details, see the information about the ALM Common User Interface in the <i>HP Application Lifecycle Management User Guide</i>.
<filter status bar>	Describes the filter currently applied to the tree. Located directly above the tree.
<component tree>	<p>Located on the left side of the window, organizes and displays your business components hierarchically.</p> <p>The tree contains the following special folders:</p> <ul style="list-style-type: none"> • Components. For user interface details, see "Components root folder" below. • Obsolete. For user interface details, see "Obsolete folder" on the 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	<p>Contains requests for new components that were submitted from the Test Plan module.</p> <p>This folder cannot be renamed or deleted.</p> <p>In this folder, component requests can be edited but cannot be copied.</p> <p>Component requests can also be edited in the Test Plan module.</p> <p>When importing libraries, components in the Component Requests folder are not included in the process.</p> <p>Version Control: Version control is supported. For details on working with version control in ALM, see the <i>HP Application Lifecycle Management User Guide</i>.</p> <p>For task details on requesting components, see "How to Create Business Process Tests and Flows" on page 104.</p>

UI Element	Description
Obsolete folder	<p>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.</p> <p>Components in this folder:</p> <ul style="list-style-type: none"> • 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 <i>HP Application Lifecycle Management User Guide</i>. <p>Tip:</p> <ul style="list-style-type: none"> • 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 "Business Components Module Menus and Buttons" on the next page.
Details	Lists the component fields. For details on component fields, see "Business Components Module Fields" on page 54 .
Snapshot	<p>Captures and attaches a snapshot image from the application. For user interface details, see "Snapshot Tab" on page 58.</p> <p>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 <i>HP Unified Functional Testing User Guide</i>.</p>
Manual Implementation	<p>Enables you to create or view the manual steps of your business component.</p> <p>For user interface details, see "Manual Implementation Tab" on page 65.</p>
Automation	<p>Enables you to create or view the automation of your business component.</p> <p>For user interface details, see "Automation Tab" on page 78.</p>
Parameters	Defines input and output component parameters and default parameter values for the business component. For user interface details, see "Parameters Page" on page 179 .
Dependencies	Displays the dependency relationships between components, tests, flows, and resources (including application areas). For user interface details, see the <i>HP Application Lifecycle Management User Guide</i> .
History	Displays a log of changes made to the component. For user interface details, see the <i>HP Application Lifecycle Management User Guide</i> .
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 <i>HP Application Lifecycle Management User Guide</i> .

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	<p>Tip: You can access the Business Components menu bar from the Business Components module by pressing the shortcut key ALT.</p> <p>Version Control: Additional menu commands and buttons are available in a version control enabled project. For user interface details, see the <i>HP Application Lifecycle Management User Guide</i>.</p>
Relevant tasks	<ul style="list-style-type: none"> • "How to Create Business Components in ALM" on page 41 • "How to Work with Manual Implementation" on page 62 • "How to Work with Keyword GUI Automation" on page 73 • "How to Work with Scripted GUI and API Automation" on page 94
See also	<ul style="list-style-type: none"> • "Business Process Testing Icons" on page 52 • "Business Components Overview " on page 36

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 site parameter in the Site Configuration tab in Site Administration. For details, see the <i>HP Application Lifecycle Management Administrator Guide</i> .
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 <i>HP Application Lifecycle Management User Guide</i> .
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 54 .
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.

UI Elements (A-Z)	Available from	Description
		For user interface details, see " New Component/Component Details Dialog Box " on page 56.
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.
Copy/Paste	Edit	<p>Copies selected components and/or folders (including subfolders) within the same project or across projects.</p> <div style="background-color: #f0f0f0; padding: 10px;"> <p>Note:</p> <ul style="list-style-type: none"> 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 58. Version Control: Before copying checked out entities, follow the guidelines described in the <i>HP Application Lifecycle Management User Guide</i>. </div>
Cut/Paste	Edit	<p>Moves selected components or folders to a different location in the component tree.</p> <p>The root folder cannot be moved.</p> <div style="background-color: #f0f0f0; padding: 10px;"> <p>Tip: You can also move a component to a new location in the component tree by dragging it.</p> </div>
Delete 	Edit	<p>Removes the selected folders or business components from the component tree. The contents of a removed folder are also removed.</p> <p>You cannot delete the root folder.</p> <p>Any removed component that is unused is permanently deleted.</p> <p>Any removed component that is used by one or more business process tests or flows is moved to the Obsolete folder in the tree.</p>

UI Elements (A-Z)	Available from	Description
		<p>Tip: Business components can be restored from the Obsolete folder. For details, see "Business Components Module Menus and Buttons" on page 48.</p> <p>Note: To check whether any business process tests or flows use a component, select the component and click the Dependencies tab.</p> <p>Version Control: Deleting a business component deletes all previous versions of the component.</p>
Expand All	View	Expands all branches in the component tree.
Favorite bar Favorite: (private) ReadyComponents:	View	Displays the Favorite bar.
Filter bar Filter: Status[Ready]	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 <i>HP Application Lifecycle Management User Guide</i> . For a list of fields available for filtering and sorting, see "Business Components Module Fields" on page 54.
Find	Edit	Searches for a component in the Components module. For user interface details, see the <i>HP Application Lifecycle Management User Guide</i> .
Find Next	Edit	Searches for the next item that meets the previously defined search criteria.
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. <p>Note: You can only go to components that are in the current filter.</p>
Graphs	Analysis	Lists the graphs you can generate for component data. Select a predefined graph, or launch the graph wizard. <p>For details, see the <i>HP Application Lifecycle Management User Guide</i>.</p>
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 <i>HP Application Lifecycle Management User Guide</i> .

UI Elements (A-Z)	Available from	Description
		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 56.
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 <i>HP Application Lifecycle Management User Guide</i> .
Private	Favorites	Lists the favorite views that are accessible only to the user who created them.
Project Reports	Analysis	Lists the predefined reports you can produce on component data. For details, see the <i>HP Application Lifecycle Management User Guide</i> .
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.
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 <i>HP Application Lifecycle Management User Guide</i> .
Report Selected	<right-click menu>	Displays a report of the selected components.
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

UI Elements (A-Z)	Available from	Description
		<p>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.</p> <p>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.</p>
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 <i>HP Application Lifecycle Management User Guide</i> .
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 <i>HP Application Lifecycle Management User Guide</i> .
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 <i>HP Application Lifecycle Management User Guide</i> .

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	<ul style="list-style-type: none"> 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 <i>HP Application Lifecycle Management Administrator Guide</i>.
See also	<ul style="list-style-type: none"> "Business Components Module Menus and Buttons" on page 48 "Business Components Module Window" on page 46

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.

UI Element	Description
	A business component that does not have a manual implementation or automation.
	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. BPT Packaged Apps Kit created the business component using the Learn process.
	Maintenance. The yellow triangle indicates that this business component was previously implemented and is now being modified.
	Manual. A manual business component.
<no icon>	<p>Not Implemented. 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.</p> <p>Note: This status is available from the Test Plan module.</p>
	Obsolete. The component is obsolete, cannot be modified, and should not be used in business process tests or flows.
	<p>Ready.</p> <p>For components: A green-circled checkmark indicates that this business component is fully implemented and ready to be run.</p> <p>For business process tests and flows: A green checkmark indicates that a business process test or flow is ready, meaning, all its components are ready.</p>
	<p>Under Development. The yellow triangle indicates that this business component is currently under development. This status is assigned to:</p> <ul style="list-style-type: none"> • New components • Component requests dragged or moved out of the Component Requests folder into a component folder
	<p>Comment. The information on the current row in the automated component's Keyword View is a comment.</p> <p>Available from: UFT Keyword View in the Automation tab</p>
	<p>Missing Object. The automated component contains an object that is missing from the shared object repository.</p> <p>Available from: UFT Keyword View in the Automation tab</p>
	<p>Constant. The automated component contains an output value configured as a constant.</p> <p>Available from: UFT Keyword View in the Automation tab</p>
	<p>Component Parameter. The automated component contains an output value configured as a component parameter.</p> <p>Available from: UFT Keyword View in the Automation tab</p>
	<p>Local Parameter. The automated component contains an output value configured as a local parameter.</p> <p>Available from: UFT Keyword View in the Automation tab</p>
	<p>Version Control. The component is checked out.</p> <ul style="list-style-type: none"> • Green lock. The component is checked out by the current user. • Red lock. The component is checked out by another user.

UI Element	Description
	For details on working with version control in ALM, see the <i>HP Application Lifecycle Management User Guide</i> .

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	<ul style="list-style-type: none"> 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 <i>HP Application Lifecycle Management Administrator Guide</i>. You can use the Script Editor to restrict and dynamically change the fields and values in the Business Components module. For details, see the <i>HP Application Lifecycle Management Administrator Guide</i>. Version Control: Additional fields are available in a version control enabled project. For user interface details, see the <i>HP Application Lifecycle Management User Guide</i>.
Relevant tasks	<ul style="list-style-type: none"> "How to Create Business Components in ALM" on page 41 "How to Work with Manual Implementation" on page 62 "How to Work with Keyword GUI Automation" on page 73 "How to Work with Scripted GUI and API Automation" on page 94
See also	"Business Components Module Window" on page 46

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.

UI Element (A-Z)	Description
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
Description	<p>A description of the business component. You can include information such as:</p> <ul style="list-style-type: none"> • 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. <p>Examples of pre-conditions:</p> <ul style="list-style-type: none"> • 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 <p>Examples of post-conditions:</p> <ul style="list-style-type: none"> • 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. <p>This field displays a toolbar for formatting and spell checking the text.</p> <p>UFT:</p> <ul style="list-style-type: none"> • 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. <p>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.</p>
Manual Implementation	Indication that the component has manual 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
Status	<p>The status of the business component.</p> <p>This field is read-only. The status of the business component is determined by the status of its automation. If no automation exists, business component status is determined by the status of its manual implementation.</p> <ul style="list-style-type: none"> • Ready. The automation or manual implementation of the component is complete and ready for use in a business process test or flow.

UI Element (A-Z)	Description
	<ul style="list-style-type: none"> • Maintenance. The component automation (or, if no automation, the manual implementation) is being modified, and has an Under Development or Maintenance status. • Not Implemented. A component request has been initiated in the Test Plan module. • Under Development. One of the following applies: <ul style="list-style-type: none"> • The component has just 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. • The component is obsolete (meaning, it was deleted but other entities, such as business process tests, still refer to it). • Error. The component automation (or, if no automation, the manual implementation) has errors and needs to be debugged. <p>For icons representing business component statuses, see "Business Process Testing Icons" on page 52.</p> <p>Additional statuses can be defined by updating the project list for business component statuses. For details, see the <i>HP Application Lifecycle Management Administrator Guide</i>.</p> <p>The status can also be modified by the automation engineer in the Business Component Settings dialog box in UFT.</p>
Subtype ID	<p>The implementation of the component.</p> <p>The Subtype ID is read-only, and is displayed in the Manual Implementation and Automation tabs.</p> <p>Valid subtype IDs are:</p> <ul style="list-style-type: none"> • MANUAL. Manual steps. This value is available in the Manual Implementation 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 or SERVICE-TEST-11-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.

To access	<ul style="list-style-type: none"> • 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	<ul style="list-style-type: none"> • "How to Create Business Components in ALM" on page 41 • "How to Work with Manual Implementation" on page 62 • "How to Work with Keyword GUI Automation" on page 73 • "How to Work with Scripted GUI and API Automation" on page 94
See also	<ul style="list-style-type: none"> • "Components with Manual Content" on page 37 • "Manual Implementation Overview" on page 62

- ["Components with Automated Content" on page 38](#)

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 <i>HP Application Lifecycle Management User Guide</i> .
	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 54 .
Snapshot	Captures and attaches a snapshot image from the application. For user interface details, see "Snapshot Tab" on the 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 <i>HP Unified Functional Testing User Guide</i> . Available from: Component Details dialog box
Manual Implementation	Enables you to create or view the manual steps of your business component. For user interface details, see "Manual Implementation Tab" on page 65 . 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 78 . 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 179 . Available from: Component Details dialog box
Dependencies	Displays the dependency relationships between components, tests, flows, and resources (including application areas). For user interface details, see the <i>HP Application Lifecycle Management User Guide</i> . Available from: Component Details dialog box
History	Displays a log of changes made to the component. For user interface details, see the <i>HP Application Lifecycle Management User Guide</i> .

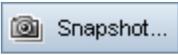
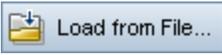
UI Element	Description
	Available from: Component Details dialog box

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	<ul style="list-style-type: none"> 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	<ul style="list-style-type: none"> "How to Work with Manual Implementation" on page 62 <i>HP Application Lifecycle Management User Guide</i>
See also	<i>HP Application Lifecycle Management User Guide</i>

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

UI Element	Description
 Snapshot...	Snapshot. Opens the Snapshot dialog box. For user interface details, see the <i>HP Application Lifecycle Management User Guide</i> .
 Load from File...	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	<ol style="list-style-type: none"> In the source project, right-click a component or component folder and select Copy. Open the target project as the same user in a separate browser window. 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	<ul style="list-style-type: none"> 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

	<p>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.</p> <ul style="list-style-type: none"> 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 46

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.

When choosing the destination folder, you can select a folder, a single test, or multiple tests.

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	<ul style="list-style-type: none"> "How to Work with Manual Implementation" on page 62 <i>HP Application Lifecycle Management User Guide</i>

Considerations when Converting ALM Manual Tests to Components

Entity	Considerations
Parameters	<ul style="list-style-type: none"> 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.

Entity	Considerations
Steps	In the Manual Implementation tab of the new component: <ul style="list-style-type: none"> • 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 (unlabeled elements are shown in angle brackets):

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.
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. <div style="background-color: #e0e0e0; padding: 5px;"> 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). </div> 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 3: Components with Manual Implementation

This chapter includes:

- [Manual Implementation Overview](#)62
- [How to Work with Manual Implementation](#)62
- [Manual Implementation User Interface](#)64

Manual Implementation Overview

In the Business Components module, you can create or modify manual business component content by adding manual 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 manual steps defined in the component are performed.

Tip:

- You can use your manual steps as guidelines for creating automated steps at a later time, using an applicable testing tool.
- You can use your manual 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 manual steps in the Manual Implementation tab of each manual component. You run manual tests using the Manual Runner or Sprinter.

For task details, see ["How to Work with Manual Implementation" below](#).

How to Work with Manual Implementation

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

To learn more about creating manual components, see ["Manual Implementation 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 41](#).

This task includes the following steps:

- ["Select a component" below](#)
- ["Add steps" below](#)
- ["Parameterize steps - optional" below](#)
- ["Export steps - optional" below](#)
- ["Results" on the 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 Manual Implementation tab. If there are existing steps, select the step below which you want to add a new step. For user interface details, see ["Manual Implementation Tab" on page 65](#).

Click **New Step** . The Component Step Details dialog box is displayed. For user interface details, see ["Manual Implementation Pane/Component Step Details Dialog Box" on page 65](#).

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 174](#).

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.

4. Export steps - optional

From the Manual Implementation 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 Manual Implementation tab/Component Step Details dialog box, see the **Export** UI element in the "[Manual Implementation Pane/Component Step Details Dialog Box](#)" on the next page.

5. Results

The steps are listed in the Manual Implementation tab.

Manual Implementation User Interface

This section includes:

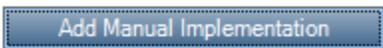
- [Manual Implementation Tab](#)65
- [Manual Implementation Pane/Component Step Details Dialog Box](#)65

Manual Implementation Tab

This tab enables you to create and edit the manual implementation for components.

To access	From the Business Components module, click the Manual Implementation tab. From the Component Details dialog box, click the Manual Implementation sidebar entry.
Relevant tasks	<ul style="list-style-type: none"> • "How to Work with Manual Implementation" on page 62
See also	<ul style="list-style-type: none"> • "Business Components Overview " on page 36 • "Components with Manual Content" on page 37 • "Manual Implementation Overview" on page 62

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

UI Element	Description
 	Enables you to add or delete manual steps to/from the business component.
Keep Editable	Manual implementation steps and automation are not synchronized. You can edit the manual implementation steps. Available from: Manual Implementation tab, when Keyword GUI testing automation also exists
Sync to Automation	Manual implementation steps and automation are synchronized. You cannot edit the manual implementation steps. Available from: Manual Implementation tab, when Keyword GUI testing automation also exists
< Business Component fields >	For field definitions, see "Business Components Module Fields" on page 54 .
< Manual Implementation pane >	Enables you to create and edit manual steps. For user interface details, see "Manual Implementation Pane/Component Step Details Dialog Box" below .

Manual Implementation Pane/Component Step Details Dialog Box

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

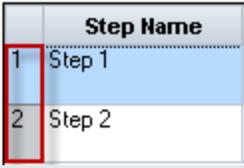
To access	<p>Manual Implementation pane: In the Business Components module, select a component in the component tree. Click the Manual Implementation tab.</p> <p>Component Step Details dialog box: In the Business Components module, select a component in the component tree. In the Manual Implementation tab, do one of the following in the Manual Implementation pane:</p>
------------------	---

	<ul style="list-style-type: none"> Right-click the step grid, or a step, and select New Step. Right-click a step and select Edit Step.
Important information	<ul style="list-style-type: none"> This pane or dialog box enables you to create manual 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 <code>password</code>, you must italicize the entire string <<<<code>password</code>>>> and not just the word <code>password</code>. 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 Implementation" on page 62
See also	"Manual Implementation Overview" on page 62

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 54 . Available from: Manual Implementation pane
	Allows you to browse through manual steps. Available from: Component Step Details dialog box
	New Step. Creates a new step. <ul style="list-style-type: none"> From the Manual Implementation 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: Manual Implementation pane
	Renumbr Steps. Renumbers the steps sequentially. Use this command after you rearrange steps. Available from: Manual Implementation pane
	Delete Selected. Deletes the selected step(s). Available from: Manual Implementation pane and Component Details dialog box
	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 58 . Note: Parameters included in the step are copied to the target.

UI Element	Description
	Available from: Manual Implementation 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 174 .
	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 <i>HP Application Lifecycle Management User Guide</i> .
	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: Manual Implementation 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: Manual Implementation 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: Manual Implementation pane
	Move Down. Moves the selected step down in the list. Available from: Manual Implementation 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: Manual Implementation pane
<Step grid>	Displays steps that are included in the selected component. Each row in the step grid is numbered sequentially. Example

UI Element	Description
	 <p>Available from: Manual Implementation pane</p>
Step Name	<p>Displays the name of each step.</p> <p>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.</p>
Description	<p>Displays the description of each step.</p> <p>Parameters included in the Description column are surrounded by <<< >>> brackets.</p> <p>Tip:</p> <ul style="list-style-type: none"> • 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	<p>Displays the expected state after the step instructions are performed.</p> <p>Note: Parameters included in the Expected Result column are surrounded by <<< >>> brackets.</p> <p>Tip:</p> <ul style="list-style-type: none"> • 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 4: Components with Keyword GUI Automation

This chapter includes:

- [Keyword GUI Automation Overview](#) 70
- [How to Work with Keyword GUI Automation](#) 73
- [Keyword GUI Automation User Interface](#) 77

Keyword GUI Automation Overview

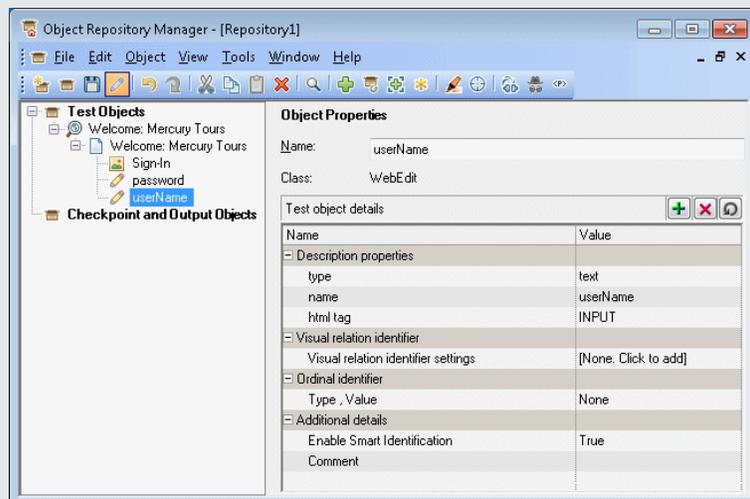
Keyword GUI automated components are reusable modules that perform a specific task when testing your application. The tasks are defined as keyword steps. These steps can be viewed and edited both in ALM and in UFT.

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 78](#).

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 ["Working with Unified Functional Testing" on page 25](#).

This chapter describes how to use several Business Process Testing features. HP BPT Packaged Apps Kit provides additional, automated features that enable you to test packaged apps. Therefore, some procedures described in this chapter may not be part of the recommended best practice when using BPT Packaged Apps Kit to test your packaged app.

This section includes the following topics:

- ["Object Overview" on the next page](#)
- ["Operation Overview" on the next page](#)
- ["Operation Values and Their Arguments" on the next page](#)
- ["Keyword View Parameterization" on page 72](#)

- ["Output Value Overview" on the next page](#)
- ["Comment Overview" on the next page](#)
- ["Checkpoint Overview" on page 73](#)

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 78](#) using the ["Value Configuration Options Dialog Box" on page 86](#) for input parameters and the ["Output Options Dialog Box" on page 85](#) 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 179](#) of the Business Components module. For task details on working with component parameters, see ["How to Handle Data in Business Process Testing" on page 170](#).

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 231](#).

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.

BPT Packaged Apps Kit users can also add output values to a component during the Learn 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 231](#).

BPT Packaged Apps Kit users can also add object property checkpoints to a component during the Learn 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 ["Working with Business Process Testing" on page 24](#).
- For a sample workflow of using Business Process Testing and keyword GUI automation together, see ["Working with Unified Functional Testing" on page 25](#).

This task includes the following steps:

- ["Prerequisites" below](#)
- ["Create an application area" on the next page](#)
- ["Select a component" on the next page](#)
- ["Add keyword GUI automation" on the next page](#)
- ["Select the application area" on page 75](#)
- ["Add content \(steps\)" on page 75](#)
- ["Parameterize steps" on page 76](#)
- ["Enhance steps - optional" on page 76](#)
- ["Results" on page 76](#)

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 available from the HP Application Lifecycle Management Add-ins page (**Help > Add-ins**).
- To be able to debug or run automated keyword GUI components, or view test results, make sure that UFT 12.00 or later is installed.

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 choose when created automated steps.
 - 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 78](#).

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 83.

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.
- You can record steps in other applications, creating components at the same time. For details, see "[How to Record Components into Flows and Business Process Tests](#)" on page 108.
- You can scan objects into the local object repository directly from within ALM instead of only from within UFT. For details, see **Scan Kwd** in "[Keyword View Pane](#)" on page 78.

For conceptual details on the entities used to create steps, such as items and operations, see "[Keyword GUI Automation Overview](#)" on page 70.

For user interface details on the Keyword View pane, see ["Keyword View Pane" on page 78](#).

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 86
Output values for a step	"Output Value Properties Dialog Box" on page 89
Expected property values from the Checkpoint Properties dialog box or the Output Value Properties dialog box	"Output Value Properties Dialog Box" on page 89

Save the steps.

8. Enhance steps - optional

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

Enhancement	Description
Comments	<p>Comments provide additional information in the form of free-text. You can insert comments between, and as placeholders for, step operations.</p> <p>For conceptual details about comments, see "Keyword GUI Automation Overview" on page 70.</p> <p>For user interface details, see "Keyword View Pane" on page 78.</p>
Checkpoints	<p>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 "Keyword GUI Automation Overview" on page 70.</p> <p>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 87.</p>
Output Values	<p>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.</p> <p>For conceptual details about output values, see "Output Value Overview" on page 72.</p> <p>For user interface details, see "Output Value Properties Dialog Box" on page 89.</p>

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 52](#).
- 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	78
• Keyword View Pane	78
• Select Application Area Dialog Box	83
• Keyword View Options Dialog Box	83
• Select Test Object Dialog Box	85
• Output Options Dialog Box	85
• Value Configuration Options Dialog Box	86
• Checkpoint Properties Dialog Box	87
• Parameterization / Properties Dialog Box	88
• Output Value Properties Dialog Box	89
• About Dialog Box	90
• Location Information Dialog Box	90

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	<ul style="list-style-type: none"> • "How to Work with Keyword GUI Automation" on page 73 • "How to Work with Scripted GUI and API Automation" on page 94
See also	<ul style="list-style-type: none"> • "Business Components Overview " on page 36 • "Components with Automated Content" on page 38 • "Keyword GUI Automation Overview" on page 70 • "Scripted GUI and API Automation Overview" on page 94

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

UI Element	Description
	<p>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:</p> <ul style="list-style-type: none"> • Keyword GUI • Scripted GUI • API <p>Note: If you click Add Automation without selecting the down arrow, ALM adds the type of automation used the last time you added automation to a component. If you have not yet added automation, ALM adds keyword GUI automation (assuming the UFT add-in or the UFT for Business Process Testing add-in is installed on your machine. These add-in are available from the HP Application Lifecycle Management Add-ins page (Help > Add-ins)).</p> <p>For details on the automation types, see "Automation Types" on page 39.</p> <p>Available when: At least one testing tool is installed on your machine.</p>
<Business Component fields>	For field definitions, see "Business Components Module Fields" on page 54 .
<Keyword View pane>	Enables you to create and edit keyword steps. For user interface details, see "Keyword View Pane" below .

Keyword View Pane

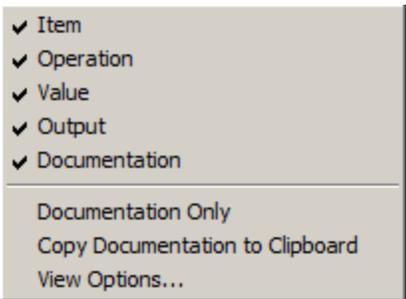
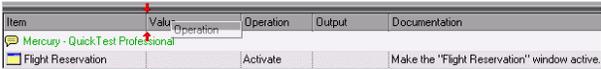
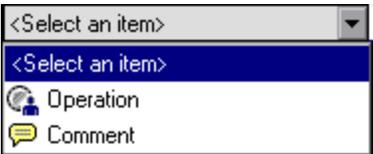
This pane enables you to create and modify keyword GUI steps in a keyword view.

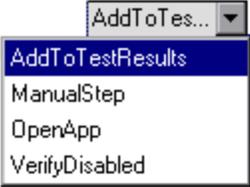
To access	<ol style="list-style-type: none"> 1. In the Business Components module, select a component with Keyword GUI automation. 2. Click the Automation tab. The Keyword View pane is displayed in the lower half of the window.
------------------	--

Important information	<ul style="list-style-type: none"> 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 <i>HP Unified Functional Testing User Guide</i>. 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 83. 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 73
See also	<ul style="list-style-type: none"> "Keyword GUI Automation Overview" on page 70 <i>HP Unified Functional Testing User Guide</i>

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

UI Element	Description
 Record Steps	Record Steps. Records keyword steps into business components. For details, see " How to Record Components into Flows and Business Process Tests " on page 108.
 Scan KWD	Scan Keyword. Captures objects in your application and stores them in the local object repository. For details, see the <i>Unified Functional Testing User Guide</i> .
	Save. Saves the component steps.
	<p>Add Step. Adds a step after the currently-selected row and opens the Select an Item list.</p> <p>Tip: This button can be used to add a comment.</p>
	<p>Delete Step. Deletes the current step.</p> <p>Before you delete a step, make sure that removing it does not prevent the component from running correctly.</p> <p>You cannot delete a step if one of its cells is in edit mode.</p> <p>Note: This button can be used to delete a comment. When deleting a comment, a warning message is not displayed.</p>
	View Options. Enables you to set how ALM displays the keyword view. For user interface details, see " Keyword View Options Dialog Box " on page 83.
	Select Application Area. Enables you to select an application area for the automation from the application area resources uploaded in the Test Resources module.

	<p><column name context menu options>. Hides or displays the column with the same name as the context menu option.</p>
<p>Column header row</p>	<p>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.</p> 
<p>Item column</p>	<p>Lists items, such as:</p> <ul style="list-style-type: none"> • Objects, from the shared object repository, on which you perform an operation. For conceptual details, see "Keyword GUI Automation Overview" on page 70. • 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 "Keyword GUI Automation Overview" on page 70. • Comments, which can be used to: <ul style="list-style-type: none"> • Add a manual step • Add other text information between steps • Add notes about the component • Provide information about adjacent steps
	<p>Select an Item list. Enables you to select the following:</p> <ul style="list-style-type: none"> • 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 85. <p>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.</p> <div style="background-color: #e0e0e0; padding: 5px;"> <p>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.</p> </div>

<p>Operation column</p>	<p>Lists the operation to be performed on the object selected in the Item column.</p> <p>Tip: When you position the cursor over an operation in the list, a tooltip describes the operation.</p> <p>For conceptual details, see "Keyword GUI Automation Overview" on page 70.</p>
	<p>Select an Operation list. Enables you to select the operation to be performed on the object selected in the Item column.</p> <p>The available operations vary according to the item selected:</p> <ul style="list-style-type: none"> 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 <i>HP Unified Functional Testing User Guide</i>. <p>The default operation (most commonly-used operation) for the item is automatically displayed in the Operation column.</p> <p>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.</p>
<p>Value column</p>	<p>Lists argument values for the selected operation (if relevant). The value is partitioned according to the number of possible arguments of the selected operation.</p> <p>For conceptual details, see "Keyword GUI Automation Overview" on page 70.</p> <p>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.</p>  <p>The Value column cell also contains a parameterization button  for parameterizing the value.</p> <p>Available when: An operation has been selected that needs additional arguments specified.</p>
	<p>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.</p> <p>Available from: An argument partition in a Value cell</p>

	<p>Predefined constant argument value. Enables you to select a value from the list.</p> <p>Available from: An argument partition in a Value cell</p>
	<p>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 89.</p> <p>Available from: An argument partition in a Value cell</p>
	<p>Cancel button. Cancels stored output values. For user interface details on storing output values, see "Output Value Properties Dialog Box" on page 89.</p> <p>Available from: An Output cell for a step with an output parameter</p>
	<p>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 87.</p> <p>Available from: An argument partition in a Value cell for a checkpoint step</p>
	<p>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 85.</p>
<p>Output column</p>	<p>Lists the parameter in which output values for the step are stored (if relevant).</p>
<p>Documentation column</p>	<p>Provides a read-only, plain-language textual description of what the step does.</p> <div data-bbox="667 1024 1370 1178" style="background-color: #f0f0f0; padding: 10px;"> <p>Examples:</p> <ul style="list-style-type: none"> • Click the "Sign-in" image • Select "San Francisco" in the "toPort" list. </div> <p>The documentation is automatically displayed after you have entered the item, operation, and any required values for the selected operation.</p> <p>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 228.</p>
<p><Comment Row></p>	<p>Comments span an entire row in the Keyword View.</p> <p>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.</p>
<p>Documentation Only context menu option</p>	<p>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.</p>
<p>Insert Comment context menu option</p>	<p>Adds a comment after the currently-selected row.</p>
<p>Copy / Cut / Paste / Delete context menu options</p>	<p>Performs these standard operations on the currently-selected rows in the Keyword View.</p>

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	<ol style="list-style-type: none"> 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	<ul style="list-style-type: none"> • 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 <i>HP Unified Functional Testing User Guide</i>. • 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 73
See also	"Application Area Overview" on page 39

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	<ol style="list-style-type: none"> 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 73
See also	<ul style="list-style-type: none"> • "Keyword View Pane" on page 78 • <i>HP Unified Functional Testing User Guide</i>

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. <ul style="list-style-type: none"> • 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	<ol style="list-style-type: none"> In the Business Components module, select a component with Keyword GUI automation. Click the Automation tab. Click in an Item cell, and then click the arrow to display the Select an Item list. <div style="border: 1px solid gray; background-color: #f0f0f0; padding: 5px; margin: 5px 0;"> <p>Note: If you have just created a new step, the list opens automatically as soon as you create the new step.</p> </div> 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 73
See also	<i>HP Unified Functional Testing User Guide</i>

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.

To access	<ol style="list-style-type: none"> In the Business Components module, select a component with Keyword GUI automation. Click the Automation tab. Click in an Output cell of a step that returns an output value. Click Specify parameter for output .
Important information	<ul style="list-style-type: none"> 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.

	<ul style="list-style-type: none"> 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 170.
Relevant tasks	"How to Work with Keyword GUI Automation" on page 73
See also	<i>HP Unified Functional Testing User Guide</i>

User interface elements are described below:

UI Element	Description
Output Types	<p>Enables you to specify the type of the output parameter:</p> <ul style="list-style-type: none"> 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	<p>Displays additional fields for specifying local parameters.</p> <p>Available when: Local Parameter is selected in the Output Types field.</p>
Name	<p>Enables you to enter a name for a new local parameter or select an existing local parameter from the list.</p> <p>Available when: Local Parameter is selected in the Output Types field.</p>
Description	<p>Enables you to enter a brief description for the local parameter.</p> <p>Available when: Local Parameter is selected in the Output Types field.</p>
Parameter	<p>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.</p> <p>Available when: Component Parameter is selected in the Output Types field.</p>

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.

To access	<ol style="list-style-type: none"> In the Business Components module, select a component with Keyword GUI automation. Click the Automation tab. Click in a Value cell. Click Configure the value .
Important information	<p>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.</p>

	<p>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 173.</p> <p>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.</p>
Relevant tasks	"How to Work with Keyword GUI Automation" on page 73
See also	<i>HP Unified Functional Testing User Guide</i>

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	<ol style="list-style-type: none"> 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). Click the Automation tab. Click Checkpoint Properties  in the Value cell of a checkpoint step.
Relevant tasks	"How to Work with Keyword GUI Automation" on page 73

See also	<ul style="list-style-type: none"> • "Keyword GUI Automation Overview" on page 70 • <i>HP Unified Functional Testing User Guide</i>
-----------------	---

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

UI Element	Description
Name	The name assigned to the checkpoint.
Property	<p>The property name, and an icon indicating the expected value type of the property.</p> <ul style="list-style-type: none"> • 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>	<p>You can accept the selected checks or modify them accordingly.</p> <ul style="list-style-type: none"> • 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" below.

Parameterization / Properties Dialog Box

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

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 73
See also	<i>HP Unified Functional Testing User Guide</i>

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

UI Element	Description
Constant	<p>A value that is defined directly in the step and remains unchanged when the component runs.</p> <p>If you select Constant, you can edit the value directly in the Constant box.</p> <p>Available from: Parameterization dialog box</p>
Parameter	<p>A value that is defined or generated separately from the step and is retrieved when the specific step runs.</p> <p>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.</p> <p>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 170.</p> <p>Available from: Parameterization dialog box</p>
	Parameter Options. Opens the Parameter Options dialog box, which enables you to parameterize the value.

UI Element	Description
	Available from: Parameterization dialog box
<output definition>	<p>Pane containing the output definition (type and name) for the selected property. You can change the output definition by clicking the Modify button.</p> <p>Available from: Properties dialog box</p>
Modify	<p>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 85.</p> <p>Available from: Properties dialog box</p>

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	<ol style="list-style-type: none"> 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 73
See also	<ul style="list-style-type: none"> • "Keyword View Pane" on page 78 • "Checkpoint Properties Dialog Box" on page 87 • <i>HP Unified Functional Testing User Guide</i>

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>	<p>The expected value type of the property.</p> <ul style="list-style-type: none"> • 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 the previous page .

About 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  .
Important information	The UFT Professional Add-in for ALM is available from the HP Application Lifecycle Management Add-ins page (Help > Add-ins).
Relevant tasks	"How to Work with Keyword GUI Automation" on page 73
See also	<i>HP Unified Functional Testing User Guide</i>

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.														
	<p>Displays additional product information in a browser window.</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p>Example</p> <p style="text-align: center; color: #4F81BD; font-weight: bold;">Product Information</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;">Product name:</td> <td>Unified Functional Testing Add-in for ALM/QC</td> </tr> <tr> <td>Product version:</td> <td>11.5.0</td> </tr> <tr> <td>Product ID:</td> <td></td> </tr> <tr> <td>Product build:</td> <td>4693</td> </tr> <tr> <td>Operating system:</td> <td>Service Pack 1 (Build 7601)</td> </tr> </table> <p>Patch Information:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 50%;">Name</th> <th style="width: 50%;">Readme</th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> </tr> </tbody> </table> <p style="text-align: center; font-size: small;">© Copyright 1992–2010 Hewlett-Packard Development Company, L.P.</p> <p style="text-align: center;"></p> </div>	Product name:	Unified Functional Testing Add-in for ALM/QC	Product version:	11.5.0	Product ID:		Product build:	4693	Operating system:	Service Pack 1 (Build 7601)	Name	Readme		
Product name:	Unified Functional Testing Add-in for ALM/QC														
Product version:	11.5.0														
Product ID:															
Product build:	4693														
Operating system:	Service Pack 1 (Build 7601)														
Name	Readme														

Location 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 73

See also	<i>HP Unified Functional Testing User Guide</i>
-----------------	---

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 5: Components with Scripted GUI or API Automation

This chapter includes:

- [Scripted GUI and API Automation Overview](#) 94
- [How to Work with Scripted GUI and API Automation](#) 94
- [Scripted GUI and API Automation User Interface](#) 95

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 "[Working with Business Process Testing](#)" on page 24.

This task includes the following steps:

- "[Prerequisites](#)" below
- "[Create an application area](#)" on the next page
- "[Select a component](#)" on the next page
- "[Implement scripted automation](#)" on the next page
- "[Results](#)" on the 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 73 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](#)" on the next page.
- 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 52.
- 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](#)96

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	<ol style="list-style-type: none"> 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 94
See also	<ul style="list-style-type: none"> • "Scripted GUI and API Automation Overview" on page 94 • <i>HP Unified Functional Testing User Guide</i>

User interface elements are described below:

UI Element	Description
 Record Steps	Records steps into business components. For details, see "How to Record Components into Flows and Business Process Tests" on page 108 .
 Launch HP Unified Functional Testing	Opens the applicable testing tool so you can view and edit the automation script.
 Application Area: BusinessComponentTemplate10 [Change]	Displays, and lets you change, the application area. For details on changing the application area, see "Select Application Area Dialog Box" on page 83 .

Part 3: Business Process Tests and Flows

Chapter 6: Introducing Business Process Tests and Flows

This chapter includes:

- [Business Process Test and Flow Overview](#)100
- [How to Plan Business Process Tests and Flows](#)100

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). 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.

Using BPT Packaged Apps Kit, you can create a flow or test by navigating your packaged app. This process is called "learning." If you learned 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 Process](#)" on page 129.

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 "[Working with Business Process Testing](#)" on page 24.

This task includes the following steps:

- "[Create a test plan tree](#)" on the next page
- "[Define parameters](#)" on the next page
- "[Define iterations](#)" on the next page
- "[Define test configurations](#)" on the next page
- "[Create requirement coverage](#)" on the next page
- "[Link a test to a defect](#)" on the next page
- "[Analyze test plan data](#)" on page 102

- ["Debug a test or flow" on the next page](#)
- ["Establish a baseline" on the 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 104](#).

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 174](#).

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 214](#).

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 the *HP Application Lifecycle Management User Guide*.

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 145](#).

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 160](#).

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 7: Business Process Test and Flow Specification

This chapter includes:

- Business Process Test and Flow Specification Overview 104
- How to Create Business Process Tests and Flows 104
- How ALM Calculates Business Process Test and Flow Statuses 111
- Business Process Test and Flow Specification User Interface 112

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*.
- BPT Packaged Apps Kit provides additional, automated features that enable you to test packaged apps. Therefore, some procedures described in this chapter may not be part of the recommended best practice when using Business Process Testing to test your packaged app.
- 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 100.

This task includes the following steps:

- ["Create test subjects"](#) on the next page
- ["Create business process tests and flows in the test plan tree"](#) on the next page
- ["Update business process tests and flows"](#) on the next page
- ["Add content to business process tests and flows"](#) on the next page

- ["Record components for business process tests and flows - optional" on the next page](#)
- ["Learn business process tests and flows - optional" on the next page](#)
- ["Request components - optional" on the next page](#)
- ["Set run conditions" on page 107](#)
- ["Set failure conditions for components and flows" on page 107](#)
- ["Group components and flows" on page 107](#)
- ["Validate tests" on page 108](#)
- ["Run manual tests from the Test Plan module–optional" on page 108](#)
- ["Debug automated tests and flows from the Test Plan module" on page 108](#)

1. Create test subjects

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

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

Create business process tests and flows under test subject folders.

- In the Test Plan tree, right-click a folder, and select **New Test**.
- 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
- Fill in the remaining fields in the New Test dialog box.
- 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 126](#).

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.

Tip: In the Test Script Tab Grid, you can view information in an extended view. This lets you see a more comprehensive view of the test at a higher level, without drilling down by clicking various links.

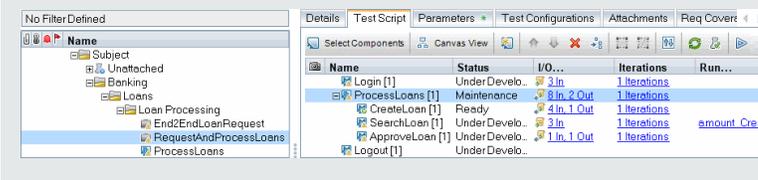
For user interface details, see ["Test Script Tab for Business Process Testing"](#) on page 117.

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. Record components for business process tests and flows - optional

Users can create business process tests and flows automatically by navigating through applications. Actions taken while navigating are recorded as business components.

For task details, see ["How to Record Components into Flows and Business Process Tests"](#) on page 108.

6. Learn business process tests and flows - optional

BPT Packaged Apps Kit users can create business process tests and flows automatically by navigating through packaged apps.

For task details, see ["How to Learn Flows and Business Process Tests"](#) on page 130.

7. 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 41.

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.)

Note: You cannot automate component requests. You can add automation only after creating the corresponding component.

8. Set run conditions

You can define whether a:

- Business component that does not meet a condition advances to the next component in the flow.
- Business component that does not meet a condition advances to the next component in the test.
- Flow that does not meet a condition advances to the next flow in the test.

You can set the status of the component or flow to either:

- **Ignored** (for manual tests) or **Not Run** (for automated tests)
- **Failed**. If you set the status to **Failed**, depending on the defined failure conditions, test or flow execution may be stopped. For details, see the next step.

In the Test Script tab, click  for a business component in a flow, a business component in a test, or a flow in a test. The Run Condition dialog box opens. For user interface details, see "[Run Conditions Dialog Box](#)" on page 150.

9. 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 117.

To set the default **On Failure** value, see the *HP Application Lifecycle Management Administrator Guide*.

10. 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 components 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 117](#).

For conceptual details about group iterations, see ["Iterations Overview" on page 210](#).

For user interface details about group iterations, see ["Iteration Pages" on page 219](#).

For task details on iterating groups, see ["How to Set Data for Iterations" on page 214](#).

11. 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 127](#).

12. Run manual tests from the Test Plan module—optional

You can run business process tests manually from a printout. The printout details the relevant parameter values for each flow, group, iteration, component and step in the business process test. To generate the printout, select the business process test and run either of the following project reports from the Analysis menu in the Test Plan module:

- Business Process Tests with Component Steps
- Business Process Tests with Component Steps and Component Details

Tip: You can also run these reports for auditing purposes.

13. Debug automated 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 158](#).

How to Record Components into Flows and Business Process Tests

This task describes how to record components into a flow or business process test automatically while navigating any application.

Recording enables you to create component steps, or even a full business process test or flow, in your application without the need to manually create separate components and their associated application

areas before starting to create steps. When recording, you perform user actions and create additional components as necessary.

Note:

- This task is part of a higher-level task. For details, see ["How to Create Business Process Tests and Flows" on page 104](#).

This task includes the following steps:

1. ["Prerequisites" below](#)
2. ["Open and log into the application to record" below](#)
3. ["Create or select a flow or business process test in ALM" below](#)
4. ["Start recording in ALM" below](#)
5. ["Working with the Recording toolbar" on the next page](#)
6. ["Record the application" on the next page](#)
7. ["Add checkpoints while recording" on the next page](#)
8. ["Stop recording" on page 111](#)
9. ["Results" on page 111](#)

1. Prerequisites

Consider the following:

- Make sure Unified Functional Testing is installed on the client machine along with its ALM plug-in.
- We recommend you close UFT before recording.
- 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.

2. Open and log into the application to record

Log into the application you want to record, and navigate to the point at which you want to start recording.

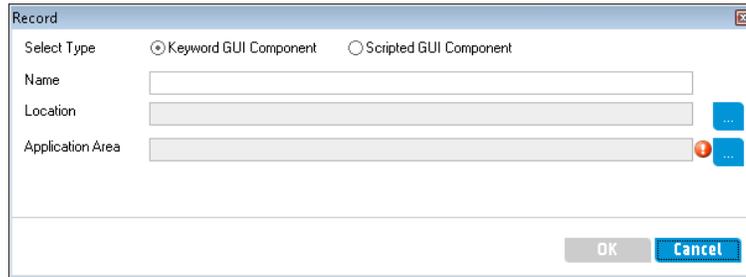
3. Create or select a flow or business process test in ALM

- a. In the Test Plan module, before recording, do one of the following:
 - Create a new flow or business process test in which to insert recorded components. For task details on creating flows, see ["How to Create Business Process Tests and Flows" on page 104](#).
 - Select an existing flow or test in which to insert recorded components.

4. Start recording in ALM

- a. In the Test Script tab toolbar, click  **Record Steps** to start recording.

The Record dialog box opens.



Enter the type of automated component to record (keyword GUI or scripted), the name of the component to record, the location, and the application area. Click **OK**. The Recording toolbar opens.

5. Working with the Recording toolbar

- a. When recording, the title bar flashes.



The activities you can perform from the toolbar include:

- Stopping / pausing the recording
- Assigning the recorded steps to different actions (GUI tests only)
- Assigning the steps to a different component in the current Test Lab Mini App
- Adding new business components
- Capturing objects in your application
- Changing the recording mode
- Inserting checkpoints or object values

For details on all Recording toolbar buttons and activities, see the *Unified Functional Testing User Guide*.

6. Record the application

Perform the operations that you want to record. In general, the operations you perform for a single flow or test should represent those of a single screen or transaction in your application. These operations are saved as actions, and you can see the actions listed in the Record toolbar.

As you perform actions, the toolbar provides a count of the number of steps performed in the application.

7. Add checkpoints while recording

You can add object property checkpoints and object property output value steps to a component while recording.

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.

Adding checkpoints and output values eliminates the need for you to add these steps after recording components.

- a. While performing user actions in your application, in the toolbar, click **Insert** and select the type of checkpoint or output value to insert.
- b. If necessary, select the object on which you want to insert the checkpoint or output value. For user interface details on the Output Value Properties dialog box, see "[Output Value Properties Dialog Box](#)" on page 89.
- c. Select the test object properties to check and click **OK**.
- d. Continue performing operations in your application to continue recording.

8. Stop recording

When you finish performing the operations that you want to record, click **STOP** in the toolbar.

9. Results

Business Process Testing 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 or test.
- Inserts the checkpoints and output values you defined.

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](#)113
- [Test Plan Module Menu and Buttons for Business Process Testing](#) 114
- [Test Plan Module Icons for Business Process Testing](#)115
- [Test Plan Module Fields for Business Process Testing](#) 116
- [Test Script Tab for Business Process Testing](#) 117
- [Test Details Dialog Box for Business Process Testing](#) 126
- [Test Validation Results Dialog Box](#) 127

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 <i>HP Application Lifecycle Management User Guide</i> .
Relevant tasks	<ul style="list-style-type: none"> • "How to Plan Business Process Tests and Flows" on page 100 • "How to Create Business Process Tests and Flows" on page 104
See also	<ul style="list-style-type: none"> • "Business Process Test and Flow Specification Overview" on page 104 • "Test Plan Module Fields for Business Process Testing" on page 116 • <i>HP Application Lifecycle Management User Guide</i>

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>	<ul style="list-style-type: none"> • Test Plan for Business Process Testing fields. For field definitions, see "Test Plan Module Fields for Business Process Testing" on page 116. • 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 the next page. • Test Plan for Business Process Testing icons. For icon descriptions, see "Test Plan Module Icons for Business Process Testing" on page 115.
<Test plan tree>	<p>Organizes and displays your tests hierarchically.</p> <p>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.</p>
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 145.
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 116. For details on other test fields, see the <i>HP Application Lifecycle Management User Guide</i> .
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

UI Element (A - Z)	Description
	details, see "How to Handle Data in Business Process Testing" on page 170.
Test Configurations	Enables you to define and update test configurations. For details specific to Business Process Testing, see "Business Process Test Configurations Overview" on page 154. For task details, see the <i>HP Application Lifecycle Management User Guide</i> .
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 117.

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.
Relevant tasks	"How to Create Business Process Tests and Flows" on page 104
See also	<ul style="list-style-type: none"> <i>HP Application Lifecycle Management User Guide</i> "Test Plan Module Window for Business Process Testing" on the previous page

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

UI Elements	Available from	Description
Detect Changes 	Tests	<p>Opens the Automatic Runner dialog box, which enables you to run learned tests and flows in change detection mode.</p> <p>Available for: Learned business process test and flows, when BPT Packaged Apps Kit is enabled in Customization.</p>
Convert to Component 	Tests	<p>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 59.</p> <p>Available for: ALM manual tests</p>
Delete 	Edit	<p>Deletes the selected business process tests, flows, or folders.</p> <p>Caution:</p> <p>Additional cautions for Business Process Testing include:</p> <ul style="list-style-type: none"> 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.

UI Elements	Available from	Description
		<ul style="list-style-type: none"> 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>	<p>Opens the Test Details dialog box, displaying the details of the selected test or flow.</p> <p>For user interface details specific for Business Process Testing, see "Test Details Dialog Box for Business Process Testing" on page 126.</p>

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	<ul style="list-style-type: none"> 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	<ul style="list-style-type: none"> "Test Plan Module Menu and Buttons for Business Process Testing" on the previous page "Test Plan Module Window for Business Process Testing" on page 113

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.
	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 "Test Plan Module Fields for Business Process Testing" on the next page .
	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 "Test Plan Module Fields for Business Process Testing" on the next page .
	Maintenance status for a test or flow, or Design status for a flow. For details about this status, see "Test Plan Module Fields for Business Process Testing" on the next page .
	Error status for a test or flow. This is the second-most severe status for a test or flow. For details about this status, see "Test Plan Module Fields for Business Process Testing" on the next page .
	Outdated status for a test or flow. The test or flow contains obsolete components. A more up-to-date component may be required. This is the most severe status for a test or flow. For details about this status, see "Test Plan Module Fields for Business Process Testing" on the next page .
	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 "How to Create Business Process Tests and Flows" on page 104 .

UI Element	Description
	Learned. Components created by the BPT Packaged Apps Kit Learn process are identified by this icon. For task details about learning flow, see "How to Learn Flows and Business Process Tests" on page 130.
	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 Summary Page" on page 139.
	Unresolved Changes in Flow. The test or flow was run in change detection mode and its changes are unresolved. A test or flow indicates unresolved changes if it contains components (or flows) with unresolved changes.
	Version Control. The test or flow is checked out. <ul style="list-style-type: none"> • 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 <i>HP Application Lifecycle Management User Guide</i> .

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	<ul style="list-style-type: none"> • <i>HP Application Lifecycle Management User Guide</i> • "Test Plan Module Window for Business Process Testing" on page 113

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

UI Element	Description
Status	<p>The status of the business process test or flow.</p> <p>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 111.</p> <ul style="list-style-type: none"> • 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. <p>For icons representing business component statuses, see "Test Plan Module Icons for Business Process Testing" on the previous page.</p>

UI Element	Description
	<p>For component statuses, see "Status" on page 55.</p> <p>For details on test or flow statuses, see "How ALM Calculates Business Process Test and Flow Statuses" on page 111.</p>

Test Script Tab for Business Process Testing

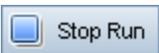
This tab enables you to create and organize business components into business process tests and flows, and flows into business process tests. The Test Script tab has a grid view and a canvas view.

To access	In the Test Plan tree, select a business process test or flow and click the Test Script tab.
Important information	<ul style="list-style-type: none"> For user interface details on the Test Script tab when the selected test is not a business process test or flow, see the <i>HP Application Lifecycle Management User Guide</i>. For BPT Packaged Apps Kit: <ul style="list-style-type: none"> In addition to manually adding existing business components to a flow or business process test, you can create a flow or business process test comprising components created automatically by recording your operations as you navigate a packaged app. For details, see "How to Learn Flows and Business Process Tests" on page 130. 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 BPT Packaged Apps Kit. <p>Caution: If you are using dynamic data and are working with an external data table in Microsoft Excel to change parameter values and add iterations, the information listed in the Test Script Tab is not accurate. To know how your test will actually run, see the data defined in the Excel file.</p>
Relevant tasks	"How to Create Business Process Tests and Flows" on page 104

User interface elements are described below:

UI Element	Description
 Select Components	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 125 .
 	<p>Canvas/Grid View. Toggles between the grid view and the canvas view.</p> <p>For user interface details on the grid view, see "Grid View" on page 120.</p> <p>For user interface details on the canvas view, see "Canvas View" on page 122.</p>
	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 "How to Create Business Process Tests and Flows" on page 104 .
 Record Steps	<p>Record Steps. Enables you to record components automatically by navigating any application. For task details, see "How to Record Components into Flows and Business Process Tests" on page 108.</p> <p>Available when: A flow or business process test is selected.</p> <p>You must have the correct permissions to record, For details on the permissions needed to</p>

UI Element	Description
	record, see "Prerequisites" on page 131 , or see the <i>HP Application Lifecycle Management Administrator Guide</i> .
 Learn	<p>Learn Flow. Enables you to learn a flow automatically by navigating your packaged app. For task details, see "How to Learn Flows and Business Process Tests" on page 130.</p> <p>Available for: BPT Packaged Apps Kit users with a flow selected.</p> <ul style="list-style-type: none"> • BPT Packaged Apps Kit must be enabled for the current project. Access to BPT Packaged Apps Kit must be provided by your HP Application Lifecycle Management (ALM) Project Administrator using Project Customization. For details, see the <i>HP Application Lifecycle Management Administrator Guide</i>. • You must have the correct permissions to learn a flow, For details on the permissions needed to learn a flow, see "Prerequisites" on page 131, or see the <i>HP Application Lifecycle Management Administrator Guide</i>.
	<p>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.</p> <p>Tip: You can also change the testing order by dragging and dropping selected items.</p>
	<p>Delete from Test. Removes the selected business component, group, or flow from the business process test (or flow).</p> <p>Considerations</p> <ul style="list-style-type: none"> • 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. • You can also change the testing order by dragging and dropping selected items. • If you are deleting the last component in a group, the entire group is deleted. • If automatic deletion is enabled in Customization, when prompted for confirmation, a Delete unused promoted parameters from the test level? checkbox appears. If checked, unused parameters in the selected component and/or flow instances are deleted from the test and/or flow levels. (Similarly, if a group contains components and flows, and the group is deleted, the group's component and flow instances' parameters are deleted from the test and/or flow levels.) <p>Note:</p> <ul style="list-style-type: none"> • If the parameter is used at higher levels, automatic deletion does not delete it. • If there are other unused parameters, not related to this delete operation, at the higher levels, automatic deletion does not delete them. <p>For details on enabling automatic deletion, see the <i>HP Application Lifecycle Management Administrator Guide</i>.</p> <p>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.</p>
	<p>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.</p>
	<p>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.</p> <p>A group node is created above the grouped items and is identified by the group icon . The</p>

UI Element	Description
	<p>group is named, by default, Group followed by a unique number.</p> <p>Tip:</p> <ul style="list-style-type: none"> 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.
	<p>Ungroup. Ungroups components and/or flows.</p> <p>To completely remove a group, including its members, select the group and click the Remove from Test  button in the toolbar.</p> <p>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.</p>
	<p>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 208.</p>
	<p>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.</p>
	<p>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 127.</p>
	<p>Add/Edit Run Condition. Enables you to add run conditions to your flow or test.</p> <p>For user interface details, see "Run Conditions Dialog Box" on page 150.</p> <p>Available when: A business component or a flow is selected.</p>
	<p>Delete Run Condition. Enables you to remove an existing run condition from your flow or test.</p> <p>Available when: A flow or a business process test is selected.</p>
	<p>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 161.</p>
	<p>Stop Run. Enables you to stop the current run or debug session.</p> <p>Available when: Running or debugging a test from the Test Plan module.</p>
	<p>Extended View. Enables you view more information in the test script grid. This enables you to see a more comprehensive view of the test component from the grid without clicking additional links. The additional information you can see include:</p> <ul style="list-style-type: none"> Snapshot thumbnails, instead of an icon. Separate columns for input and output parameters, which include parameter names and if only one iteration, the value for the parameter. Links between input and output parameters, without having to open the I/O Parameters dialog box. <p>For additional details, see "Input Parameters" on page 121 and "Output Parameters" on page 121 in the "Grid View" on the next page.</p> <p>Not available from: The canvas.</p>

UI Element	Description
	Regular View. Enables you to switch back to the regular grid view from the extended view. Not available from: The canvas.
	Adjust Row Height. Adjusts the row size in extended view of the grid. This enables you to view all the text in the if the text is long. Available when: In grid view's extended view.
Iterations option	Opens the iterations dialog box for the selected entity. For user interface details, see "Iteration Pages" on page 219 . Available from: <right-click menu>
	Hides or shows the tabs at the bottom of the Test Script tab.
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
	Snapshot icon, indicating that the business component or flow has a snapshot. Click the icon to see the snapshot. Available when: In the grid's regular view.
	Snapshot thumbnail. Click the thumbnail to see the snapshot. Available when: In the grid's extended view.
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.

UI Elements	Description
	<p>Example</p> <p> CheckCreditHistory [1]  CheckCreditHistory [2]</p> <p>Syntax Exceptions:</p> <ul style="list-style-type: none"> • Flows and business process tests: Do not include two consecutive semicolons (; ;) or any of the following characters: \ / : " ' ? ` < > * %
Status	Displays the status of the business component or flow.
I/O Parameters	<p>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 201.</p> <p>Available when: Working in regular view, by clicking .</p>
Input Parameters	<p>Displays the name of any input 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 201.</p> <p>If only one iteration is defined, the parameter value is also displayed.</p> <p>Tip: As you add iterations, the parameter value disappears and only parameter names are displayed. If you remove iterations until only iteration is left, the parameter value for that iteration reappears.</p> <p>You cannot edit values directly in this column.</p> <p>Available when: Working in extended view, by clicking .</p>
Output Parameters	<p>Displays the name of any 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 201.</p> <p>If only one iteration is defined, the parameter value is also displayed.</p> <p>Tip: As you add iterations, the parameter value disappears and only parameter names are displayed. If you remove iterations until only iteration is left, the parameter value for that iteration reappears.</p> <p>You cannot edit values directly in this column.</p> <p>Available when: Working in extended view, by clicking .</p>
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 214 .
Run Conditions	<p>Displays a link to the run conditions defined for the business component or flow.</p> <p>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.</p>
On Failure	Enables you to define whether a run continues or ends if a specific business component or flow in the test

UI Elements	Description
	fails. <ul style="list-style-type: none"> • 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. To set the default On Failure value, see the <i>HP Application Lifecycle Management Administrator Guide</i> .
Comments	Displays the comments for the business component, group, or flow.

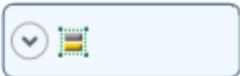
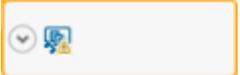
Canvas View

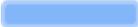
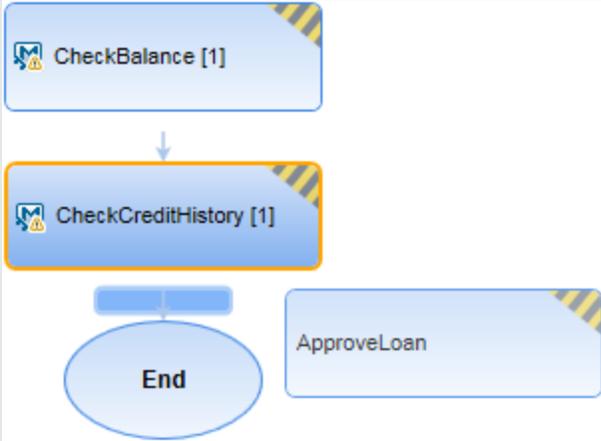
The canvas view enables you to modify your test scripts using a graphical display.

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. <p>The following describes how to work in the canvas view.</p> <ul style="list-style-type: none"> • 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 125. • 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. <div style="background-color: #f0f0f0; padding: 5px; margin: 5px 0;"> <p>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.</p> </div> <ul style="list-style-type: none"> • 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
	<p>Zoom Out. Decreases the zoom level of the canvas.</p> <p>Tip: You can press and hold this button for quicker zooming.</p>
	<p>Zoom Level. Increases/decreases the zoom level of the canvas using a slider.</p>
	<p>Zoom In. Increases the zoom level of the canvas.</p> <p>Tip: You can press and hold this button for quicker zooming.</p>
	<p>Reset Zoom. Resets the zoom level to the default zoom settings.</p>
	<p>Show Minimap. Toggles the display of the overview in the canvas view.</p> <p>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.</p>

UI Elements	Description
	<p>Example</p>  <p>Tip:</p> <ul style="list-style-type: none"> • 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.
<p>Parameter Presentation</p>	<p>Enables you to select the detail level for parameters and links in the canvas. The following levels are available:</p> <ul style="list-style-type: none"> • None • Links only • Links and names • All <p>User defined is displayed if you manually change the detail level, such as by expanding or collapsing the parameter details for a few nodes.</p>
	<p>Start node. Represents the start of the business process test or flow. For display purposes only.</p>
	<p>Component node. Represents a component entity.</p>
	<p>Component Request node. Represents a component request entity.</p>
	<p>Group node. Represents a group entity.</p> <p>Tip:</p> <ul style="list-style-type: none"> • 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.
	<p>Flow node. Represents a flow entity.</p> <p>Tip: Click the expand and collapse buttons to show/hide components in the flow.</p>

UI Elements	Description
	<p>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.</p> <p>The ApproveLoan component is being pasted after the CheckCreditHistory component.</p> 
	<p>Status. Located at the upper right of an entity's node, indicates the status of the entity.</p> <ul style="list-style-type: none"> • 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.
	<p>Collapse. Hides nodes for the selected entity.</p>
	<p>Expand. Shows nodes for the selected entity.</p>
	<p>Entity Order. Indicates the order of the entities in the business process test or flow.</p>
	<p>Input parameters. Indicates the number of input parameters for the entity.</p> <p>Tip: Click to display a scrollable list of the entity's input parameters.</p>
	<p>Output parameters. Indicates the number of output parameters for the entity.</p> <p>Tip: Click to display a scrollable list of the entity's output parameters.</p>
	<p>Promotion/Linkage Arrow. Indicates from where parameters were promoted or to where parameters are linked.</p> <p>Note: When many links exist between parameters, click the triangle next to a parameter name to display that parameter's link in yellow.</p> <p>Tip: Hover over the arrow to see the name of the target node of the link.</p>

UI Elements	Description
[<instance>]	<p>Instance. Indicates the instance of the entity in the test or flow, if the entity was selected multiple times.</p> <p>Examples: [1] [2]</p>
	<p>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.</p> <p>Click to open the Iterations dialog box for the entity.</p> <p>Hover over to see a tooltip that displays which iterations are selected.</p> <p>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).</p> <p>For user interface details, see "Iteration Pages" on page 219.</p>
	<p>Exit. Indicates the component has an On Failure condition that is set to exit.</p>
	<p>Run condition. Indicates the component or flow has a run condition.</p> <p>Click to open the Run Condition dialog box.</p> <p>Hover over to see a tooltip that displays the run condition.</p> <p>For user interface details, see "Run Conditions Dialog Box" on page 150.</p>
	<p>Notification. Click to see the notification, such as (iterations mismatch).</p>
	<p>End node. Node representing the end of the business process test or flow. For display purposes only.</p>

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.
	<p>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.</p> <ul style="list-style-type: none"> • Add and Automatically Promote All Parameters. All parameters are promoted to the next level as the component or flow is added.

UI Element	Description
	<ul style="list-style-type: none"> • 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 207. <p>For task details on promoting parameters, see "How to Promote Parameters" on page 205.</p>
	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.
	<p>Find. Searches for a specific test in the tree.</p> <p>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.</p> <p>For more details, see <i>HP Application Lifecycle Management User Guide</i>.</p>
	Filter/Sort. Filters and sorts the test in the tree. For more details, see <i>HP Application Lifecycle Management User Guide</i> .
	<p>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.</p> <p>Note: You can only go to components or flows that are in the current filter.</p>
	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	<p>The Test Details dialog box for business process tests and flows is similar to the Test Details dialog box for other test types.</p> <p>Note: The Test Details dialog box for business process tests and flows does not have a Manual Implementation tab.</p>
Relevant tasks	"How to Create Business Process Tests and Flows" on page 104
See also	<i>HP Application Lifecycle Management User Guide</i> .

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 <i>HP Application Lifecycle Management User Guide</i> .
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 117 .
Test Configurations	<p>Displays configurations of a selected test.</p> <p>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.</p> <p>Not available when: A flow is selected.</p> <p>For details specific to Business Process Testing, see "Business Process Test Configurations Overview" on page 154</p> <p>For user interface details, see the <i>HP Application Lifecycle Management User Guide</i>.</p> <p>For task details, see the <i>HP Application Lifecycle Management User Guide</i>.</p>
Criteria	<p>Displays criteria of the selected business process test.</p> <p>Not available when: A flow is selected.</p> <p>For user interface details, see the <i>HP Application Lifecycle Management User Guide</i>.</p> <p>For task details, see "How to Create Coverage by Criteria" on page 145.</p>

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	<p>If no validation errors are found, a message is displayed indicating that the validation was successfully completed.</p> <p>In the case of an iteration range mismatch, the error is indicated in the second component.</p>
Relevant tasks	"How to Create Business Process Tests and Flows" on page 104

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 8: Learn Process

Available for BPT Packaged Apps Kit Users

This chapter includes:

- [Learn Process Overview](#)130
- [How to Learn Flows and Business Process Tests](#)130
- [Criteria for Determining if a Component Can Be Reused](#)134
- [Learn Flow Process User Interface](#) 136

Learn Process Overview

Available for BPT Packaged Apps Kit Users

This chapter describes how BPT Packaged Apps Kit users can learn flows and business process tests automatically by navigating through packaged apps.

During the Learn process, BPT Packaged Apps Kit:

- Learns the actions performed when navigating packaged apps.
- Breaks down the learned actions into a sequence of automated UFT keyword and/or scripted GUI business components, each representing a screen, tab, or transaction in your app.
- Creates steps in each component based on the operations you perform.
- Populates an existing flow or test 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 process.
- Automatically creates output parameters.

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

- Automatically creates screenshots.
- Enables you to take snapshots as you learn.
- Enables you to insert checkpoints and output values into the flow or test.
- Analyzes each of the components in the flow or test 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.

Note: During the Learn 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 app-specific details on how BPT Packaged Apps Kit breaks up a learned flow or test into components and creates parameters, see ["BPT Packaged Apps Kit App-specific Information" on page 249](#).

For task details, see ["How to Learn Flows and Business Process Tests" below](#).

How to Learn Flows and Business Process Tests

Available for BPT Packaged Apps Kit Users

This task describes how to learn a flow or business process test automatically while navigating your application.

Note: This task is part of a higher-level task. For details, see ["How to Create Business Process Tests and Flows" on page 104](#).

This task includes the following steps:

1. ["Prerequisites" below](#)
2. ["Open and log into the packaged app to learn" on the next page](#)
3. ["Create or select a flow or business process test in ALM" on the next page](#)
4. ["Start the Learn wizard in ALM" on the next page](#)
5. ["Learn the packaged app" on the next page](#)
6. ["Add checkpoints while learning the packaged app" on page 133](#)
7. ["How to Learn Flows and Business Process Tests" on the previous page](#)
8. ["Stop the Learn wizard" on page 133](#)
9. ["Review the Learn summary" on page 133](#)
10. ["Reuse components - optional" on page 134](#)
11. ["Save the learned components into the flow or test" on page 134](#)
12. ["Results" on page 134](#)

1. Prerequisites

Consider the following:

- Make sure BPT Packaged Apps Kit is enabled. Access to BPT Packaged Apps Kit must be provided by your HP Application Lifecycle Management (ALM) Project Administrator using Project Customization. An administrator can enable BPT Packaged Apps Kit in ALM by clicking , then selecting **Customize > Business Process Testing** and selecting the **Enable BPT Packaged Apps Kit** checkbox. For details, see the *HP Application Lifecycle Management Administrator Guide*.
- Make sure Unified Functional Testing is installed on the client machine along with its SAP add-in and ALM plug-in.
- We recommend you close UFT before learning a flow or business process test.
- 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 BPT Packaged Apps Kit reuse options. For details, see the *HP Application Lifecycle Management Administrator Guide*.

Note: When using BPT Packaged Apps Kit, it is best to work with only once instance of ALM open. Do not open multiple instances of ALM in multiple browsers or tabs.

For additional details on configuring UFT to work with applications, setting up BPT Packaged Apps Kit, and on configuring user permissions, see the *HP Application Lifecycle Management Administrator Guide*.

2. Open and log into the packaged app to learn

Log into the app for which you want to learn the flow or business process test, and navigate to the point at which you want to start learning.

Note: BPT Packaged Apps Kit is not able to learn the login procedure.

3. Create or select a flow or business process test in ALM

- a. In the Test Plan module, before learning, do one of the following:
 - Create a new flow or business process test in which to insert learned components. For task details on creating flows, see "[How to Create Business Process Tests and Flows](#)" on [page 104](#).
 - Select an existing flow or test in which to insert learned components.
- b. If there are already components in the flow or test, 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 (before existing components), you must first insert the new components to a different location, and then change the order of the components manually after the Learn process is finished.

Example:

Suppose a 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 wizard in ALM

- a. In the Test Script tab toolbar, click the **Learn** button to start the Learn wizard.
- b. If there are already business components in the flow or test, you are asked if you want to remove the existing component. If you select **No**, the learned components are inserted after the selected component.
- c. If other messages appear related to granting access to other assets, scripts, or applications, resolve the issues and click **Retry**.
- d. The Learn toolbar opens. When Learning, the title bar flashes. For user interface details on the Learn toolbar, see "[Learn Toolbar Page](#)" on [page 138](#).

5. Learn the packaged app

Perform the operations that you want to learn. In general, the operations you perform for a single flow or test should represent those of a single screen or transaction in your application. These operations are saved as actions, and you can see the actions listed in the Learn toolbar.

Tip: You can set the recording mode to use while learning using the toolbar.

As you perform actions, the Learn toolbar provides a count of the number of steps performed in the application.

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 or test. 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 the next page.
- Using keyboard input to navigate your app instead of clicking buttons outside your screen or tab results in fewer learned components in your flow or test. For example, clicking the **Enter** button during the Learn 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 app-specific details, see "[BPT Packaged Apps Kit App-specific Information](#)" on page 249.

6. Add checkpoints while learning the packaged app

You can add object property checkpoints to a component during the Learn 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.

Adding checkpoints and output values while learning eliminates the need for you to add these steps after learning components.

- a. While performing user actions in your app, in the Learn toolbar, click **Insert** and select the type of checkpoint to insert.
- b. If necessary, in the Object Selection dialog box, select the object on which you want to insert the checkpoint or output value.
- c. In the Checkpoint Properties dialog that opens, select the test object properties to check and click OK. The Learn toolbar counter changes to note that you added a checkpoint or output value step. In addition, this checkpoint step will be part of the learned component that is created after you stop learning the app.
- d. Continue performing operations in your application to continue learning.

7. Stop the Learn wizard

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

8. Review the Learn summary

After stopping the Learn process, the Learn Summary dialog box opens.

Click any learned component to see its steps.

For user interface details on the Learn Summary Page, see "[Learn Summary Page](#)" on page 139.

9. Change the default location for learned components - optional

In the Learn Summary dialog box, click **Change Path** to change the default location in which learned business components are stored.

10. Reuse components - optional

If BPT Packaged Apps Kit detects that a learned component is similar or identical to an existing component, it displays the number of similar components next to the component name in the component tree.

In Unified Functional Testing, you can reuse an existing component in your learned flow or test in place of creating a new component. Select a component that can be substituted with a reused component from the flow / test tree in the Learn summary. For user interface details, see the *Unified Functional Testing User Guide*.

11. Save the learned components into the flow or test

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

You see the learned components in Unified Functional Testing. Return to ALM and click  to see the learned components with Business Process Testing.

Note: Creation of business components for a learned flow or test may take some time.

12. Results

BPT Packaged Apps Kit performs the following:

- Creates automated business components in the Business Components module that correspond to screens, transactions, and so on, in your app.
- Creates component steps based on the operations you perform.
- Adds the created components to the flow or test.
- 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 process.
- Automatically creates screenshots.

Criteria for Determining if a Component Can Be Reused

Available for BPT Packaged Apps Kit Users

When BPT Packaged Apps Kit 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, BPT Packaged Apps Kit 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" below](#)
- ["Percentage Similarity" below](#)
- ["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.

Entity	Similarity
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, have the same comments, and contain the same number of arguments. The values of the arguments are not compared.

Learn Flow Process User Interface

Available for BPT Packaged Apps Kit Users

This section includes:

- [Available Components for Reuse Dialog Box](#) 137
- [Learn Wizard](#) 137

Available Components for Reuse Dialog Box

Available for BPT Packaged Apps Kit Users

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 Summary dialog box, select a component and click the Available Components for Reuse tab.
Relevant tasks	"How to Learn Flows and Business Process Tests" on page 130
See also	<i>Unified Functional Testing User Guide</i>

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 the "Criteria for Determining if a Component Can Be Reused" on page 134 .
General area	Displays general information about the selected component.
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 134 .
Steps tab	Displays the steps in the component selected in the Available Components area.

Learn Wizard

Available for BPT Packaged Apps Kit Users

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

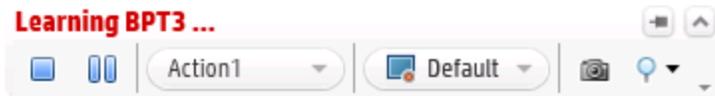
To access	In the Test Plan module, in the Test Script tab, select the flow in which you want to insert learned components. Click  Learn .
Wizard map	This wizard contains: "Learn Toolbar Page" on the next page > "Learn Summary Page" on page 139
Important information	This wizard is available to BPT Packaged Apps Kit users with a flow or business process test selected. This wizard is not available when: <ul style="list-style-type: none"> BPT Packaged Apps Kit is not enabled for the current project. Access to BPT Packaged Apps Kit must be provided by your HP Application Lifecycle Management (ALM) Project Administrator using Project Customization. An administrator can enable BPT Packaged Apps Kit in ALM by clicking  and

	<p>selecting Customize > Business Process Test, then selecting the Enable BPT Packaged Apps Kit checkbox.</p> <ul style="list-style-type: none"> You do not have the correct permissions to learn, For details on the permissions needed to learn, see "How to Learn Flows and Business Process Tests" on page 130, or see the <i>HP Application Lifecycle Management Administrator Guide</i>.
Relevant tasks	"How to Learn Flows and Business Process Tests" on page 130
See also	"Learn Process Overview " on page 130

Learn Toolbar Page

Available for BPT Packaged Apps Kit Users

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



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 .
Wizard map	The "Learn Wizard" on the previous page contains: "Learn Toolbar Page" above > "Learn Summary Page" on the next page
Important information	<p>This wizard is available to BPT Packaged Apps Kit users with a flow or business process test selected. This wizard is not available when:</p> <ul style="list-style-type: none"> BPT Packaged Apps Kit is not enabled for the current project. Access to BPT Packaged Apps Kit must be provided by your HP Application Lifecycle Management (ALM) Project Administrator using Project Customization. An administrator can enable BPT Packaged Apps Kit in ALM by clicking  and selecting Customize > Business Process Test, then selecting the Enable BPT Packaged Apps Kit checkbox. You do not have the correct permissions to learn, For details on the permissions needed to learn, see "How to Learn Flows and Business Process Tests" on page 130, or see the <i>HP Application Lifecycle Management Administrator Guide</i>. <p>As you perform actions, the Learn toolbar provides a count of the number of steps performed in the application.</p>
See also	"How to Learn Flows and Business Process Tests" on page 130

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

UI Element	Description
<Screens Learned>	In the title bar, indicates the number of screens learned during the Learn process. Available when: BPT Packaged Apps Kit is learning a flow or test.
	Stop Learn. Stops learning the actions in the packaged app.

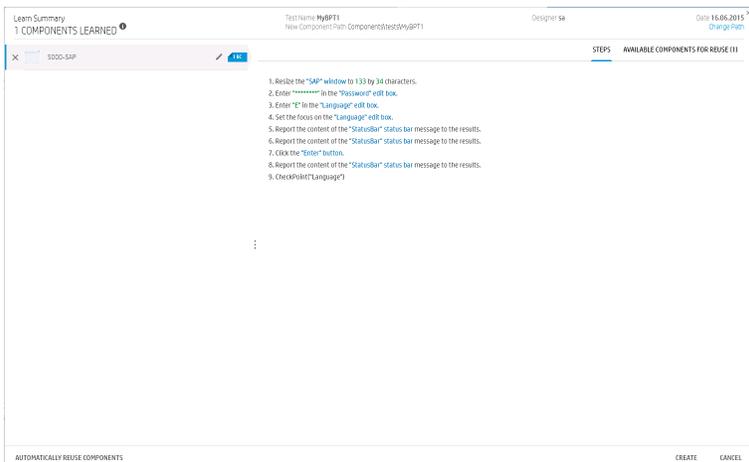
UI Element	Description
	<p>Pause. Pauses or resumes learning. When the Learn process is paused, the button has a gray background, and any actions you perform on your app are not learned. To resume learning actions, click the Pause button again.</p>
	<p>Component. Name of the component to learn. You can switch to a different existing component to re-learn it. You can also edit the name to create a new compoent.</p>
	<p>Recording Mode. Choose the type of recording you want to use, such as normal, analog, and Insight. For details, see the information about recording modes in the <i>Unified Functional Testing User Guide</i>.</p>
	<p>Snapshot. Take a snapshot of the app. For details, see the information about snapshots in the <i>Unified Functional Testing User Guide</i>.</p>
	<p>Insert Checkpoint or Output Value. Inserts a standard checkpoint or a standard output value into the automated component by choosing one of the standard options from the drop-down list.</p> <ul style="list-style-type: none"> • 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 app 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 87 • 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 89.
	<p>Pin / Unpin. Pin, or unpin, the toolbar to the side of the screen.</p>
	<p>Collapse / Expand. Collapse or expand the toolbar to a see more or less buttons.</p>

Learn Summary Page

Available for BPT Packaged Apps Kit Users

The Learn Summary page enables you to:

- View summary information on the Learn process.
- View the details of the steps performed during the Learn process.
- View information about, and save, the learned components.
- Reuse the component (in Unified Functional Testing).



Wizard map	The "Learn Wizard" on page 137 contains: "Learn Toolbar Page" on page 138 > "Learn Summary Page" on the previous page
Important information	<ul style="list-style-type: none"> General information about this wizard is available here: "Learn Wizard" on page 137.
See also	"How to Learn Flows and Business Process Tests" on page 130

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

UI Element	Description
<number of learned components>	Displays the number of components that the Learn process learned.
Test name	Displays the name of the flow or test.
New Component Path	Displays the location where the learned components are stored.
Designer	Displays the ALM user name of the user who designed the flow or test.
Date	Displays the date on which the summary was created.
Change Path	Enables you to change the path / location where the learned components are stored..
	Remove Component from Flow/Test. Deletes the selected component from the learned flow or test.
<tree>	Displays the business components learned. When you select a component you can view its steps in the Steps 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 115 .
	Thumbnail of snapshot for the component.
<component name>	Displays the full name of the component including its flow or test.

UI Element	Description
	Lets you edit the component name.
Steps tab	<ul style="list-style-type: none"> • 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.
	Displays a screenshot of your application as it appeared at the beginning of the selected component in the flow or test. For a reused component, the Screenshot tab displays the screenshot of the reused component.
Available Components for Reuse tab	Displays the reuse status of the selected component in the tree.
Reuse	On the Available Components for Reuse tab, lets you reuse a component.
Unuse	On the Available Components for Reuse tab, lets you change a reused component back to its regular state ("un-used").
Automatically reuse components	Any components found that can be reused are automatically reused when creating.
Create	Saves the new components to the Business Component module, and adds them to the flow or test. The flow / test is displayed in UFT, but you can switch to ALM to see the flow or test in Business Process Testing.
Cancel	Cancels learning. All the learned components are deleted.

Chapter 9: Requirement Coverage

This chapter includes:

- Requirement Coverage Overview144
- How to Create Coverage by Criteria 145

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 the 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 criteria 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 100](#).
- To learn more about Requirement Coverage, see ["Requirement Coverage Overview" on the 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 the next page](#)
- ["Analyze Coverage" on the 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 <i>HP Application Lifecycle Management User Guide</i> .
Criterion Results Tab	Displays results of the last test criteria run for the selected business process test.	Perform one of the following: <ul style="list-style-type: none"> • 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 Runs 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.
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 <i>HP Application Lifecycle Management User Guide</i> <div style="background-color: #e0e0e0; padding: 5px; margin-top: 10px;"> <p>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.</p> </div>
Create Test Criteria Reports	You can create a template-based report for tests that includes test criteria coverage.	For task details, see the <i>HP Application Lifecycle Management User Guide</i> .

Chapter 10: Run Conditions

This chapter includes:

- [Run Conditions Overview](#)148
- [How to Set Run Conditions](#)148
- [Run Conditions User Interface](#)149

Run Conditions Overview

You can use run conditions to insert condition statements into your flows and business process tests.

A **run condition** checks the current value of parameters before running a:

- Component in a flow
- Component in a business process test
- Flow in a business process test

Based on the parameter value and the run condition definition, HP Application Lifecycle Management (ALM) determines whether to:

- Run the component or flow
- Skip to the next component or flow
- Set the component\flow status to **Failed** and skip to the next component

When you run business process tests containing run conditions (either directly or indirectly through a flow), the test run results display the results of run conditions in the test, and lists the entities 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 why the entity run failed or did not run.

Tip: 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 an entity or change the order within a flow or a test, 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 Create Business Process Tests and Flows" on page 104](#).

This task includes the following steps:

- ["Prerequisites" on the next page](#)
- ["Add run conditions" on the next page](#)
- ["Test run conditions" on the next page](#)

1. Prerequisites

Verify that one of the following is true to use run conditions:

- A component step uses at least one flow parameter or component parameter
- There is at least one output parameter in a component that is located before the current component or flow
- Input test parameters are defined

2. Add run conditions

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

The Run Condition dialog box closes and the run condition is added to the component or 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 the next page](#).

3. Test run conditions

Run the test set/test/flow and verify that the components and flows 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 228](#) and ["How to Run Automated Business Process Tests and Flows" on page 229](#) 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.

Run Conditions User Interface

This section includes:

- [Run Conditions Dialog Box](#) 150

Run Conditions Dialog Box

This dialog box enables you to define which attributes must match for a component or flow to run.

To access	<p>In the Test Plan module, select a flow or a business process test.</p> <p>Select the Test Script tab.</p> <p>Select the component or flow for which you want to add the run condition and click Add/Edit Run Condition .</p>
Important information	<ul style="list-style-type: none"> You can only define one run condition per business component or flow. 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. <p>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.</p>
Relevant tasks	"How to Learn Flows and Business Process Tests" on page 130
See also	"Criteria for Determining if a Component Can Be Reused" on page 134

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

UI Elements	Description
Run If	<p>Lists the parameter types defined in that entity. The following types are available:</p> <ul style="list-style-type: none"> Input parameter. Available for components and flows, only if one or more input parameters are defined for this entity. Output parameter. Available for components and flows, only if one or more output parameters are defined for one or more of the previous entities. Flow parameter. Available for components, only if one or more flow input parameters is defined. Test parameter. Available for components and flows, only if one or more test parameters is defined. <p>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.</p>
<parameter name>	<p>Lists the available parameter.</p> <p>Encrypted parameters are not listed.</p> <p>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.</p>
Is	<p>Lists the operators that need to be met for the component to run. The following conditions are available:</p> <ul style="list-style-type: none"> equal to not equal to less than less than or equal greater than greater than or equal

UI Elements	Description
<value>	Enables you to enter the valid value for the condition.
Else	<p>Specifies what to do if the condition is not met. The following options are available:</p> <ul style="list-style-type: none"> • Skip to next component and continue. If the condition is not met, the entity for which the run condition is set does not run, and the test/flow run continues with the next entity. <ul style="list-style-type: none"> • When running with the Manual Runner, the test results list the run status for the component/flow with the run condition as Ignored. • When running with the Automatic Runner, the test results list the run status for the component/flow with the run condition as Not Run. • End component/flow run and fail. If the condition is not met, the entity for which the run condition is set does not run, and the test/flow run continues with the next entity. The status of the run as Failed. Because the status of the run is set as Failed, the execution of the flow or test might be stopped, depending on the Failure condition set for the entity. For details, see "Set failure conditions for components and flows" on page 107. <p>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.</p> <p>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 114.</p>
Summary	Displays a text description of the run condition you have defined for the component or flow.

Chapter 11: Business Process Test Configurations

This chapter includes:

- [Business Process Test Configurations Overview](#) 154

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.

In addition to basic test configuration functionality available in ALM, a business process test configuration can be associated with multiple iterations, based on the number of sets of data supplied for that test configuration.

Note: You cannot define configurations for flows.

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

This chapter includes the following topics:

- ["Test Configurations Associated with Static Data" below](#)
- ["Test Configurations Associated with Dynamic Data" below](#)
- ["Configuration Example with Iterations" on the next page](#)

Test Configurations Associated with Static Data

Static data are sets of values provided for each parameter directly from within ALM.

When creating test configurations for manual and automated business process tests, you 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, see the *HP Application Lifecycle Management User Guide*.

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 available from the HP Application Lifecycle Management Add-ins page (**Help > Add-ins**).

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

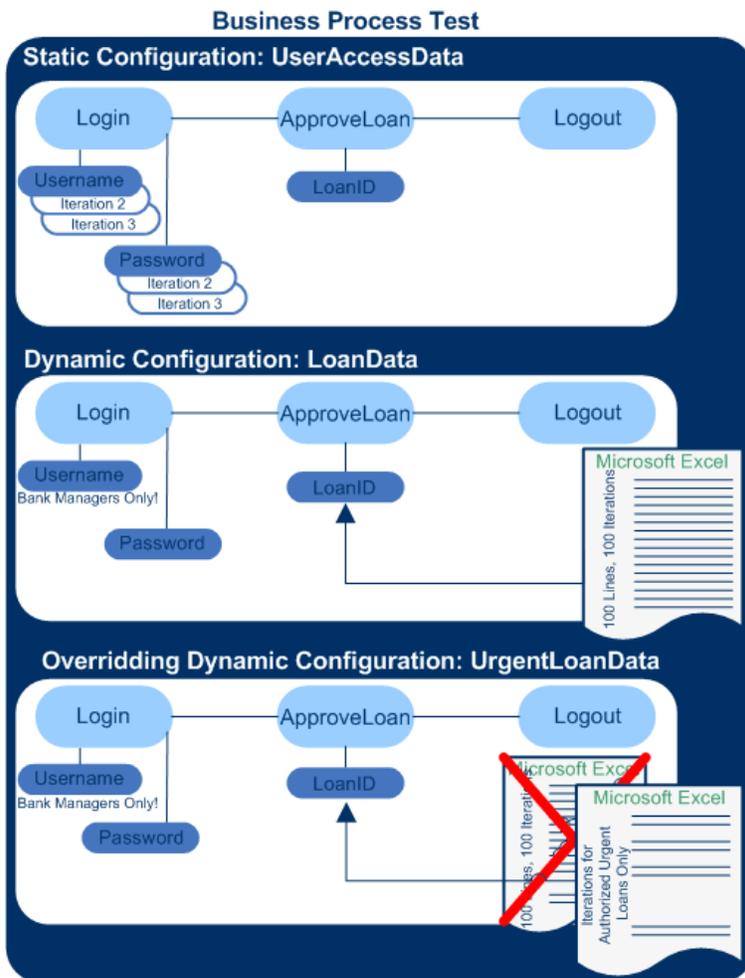
configuration in the Parameters tab of the Test Plan module.

- **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.

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

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. Configurations can supply data values for test iterations and 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).

Chapter 12: Test and Flow Debugging

This chapter includes:

- [Test and Flow Debugging Overview](#) 158
- [How to Debug Tests and Flows Containing Automated Components](#) 158
- [Test and Flow Debugging User Interface](#) 160

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.

Note: Debugging components with API automation is not supported. You can debug components with keyword GUI automation and Scripted GUI automation.

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 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 161](#).

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 100](#).

This task includes the following steps:

- ["Prerequisites" below](#)
- ["Debug automated tests and flows" below](#)
- ["Monitor the debug run" below](#)
- ["Results" on the next page](#)

1. Prerequisites

- Make sure that the flow or business process test that you intend to debug contains only automated components.
- Ensure that the components you intend to debug have either scripted GUI or keyword GUI automation.
- Ensure that default values are set for all parameters. For task details on setting values, see ["How to Set Parameter Values" on page 175](#).
- **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 11.00
 - UFT version 11.50 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 page 161](#).

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 the next page.

Test and Flow Debugging User Interface

This section includes:

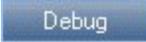
- [Run or Debug Test Dialog Box](#) 161
- [Debug Run Information Dialog Box](#) 161

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 235 . UFT and HP Service Test: After the run, the test results display in the HP Run Results Viewer.
Relevant tasks	<ul style="list-style-type: none"> • "How to Create Business Process Tests and Flows" on page 104 • "How to Run Automated Business Process Tests and Flows" on page 229
See also	"Test and Flow Debugging Overview " on page 158

User interface elements are described below:

UI Element	Description
	Sets all the keyword GUI and scripted GUI components to run in Debug mode. Note: API components cannot run in Debug mode.
	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: <ul style="list-style-type: none"> • 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 <i>HP Unified Functional Testing User Guide</i>. • 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 158
See also	"Run or Debug Test Dialog Box" on the previous page

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

UI Element	Description
<Information area>	Displays the following: <ul style="list-style-type: none">• 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.

Part 4: Data Handling

Chapter 13: Introducing Data Handling (Parameters)

This chapter includes:

- [Data Handling Overview](#) 166
- [How to Handle Data in Business Process Testing](#) 170

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 170](#).

Product Feature Movie: To view a movie that demonstrates how to handle Business Process Testing parameters, select **Help > Movies** in the ALM main window.

This section includes:

- ["Parameter Categories" below](#)
- ["Parameter Linkage" on the next page](#)
- ["Parameter Promotion" on the next page](#)
- ["Iterations and Test Configurations" on page 168](#)
- ["Integration with Parameters in Other Testing Tools" on page 169](#)

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 169](#).

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	<ul style="list-style-type: none">• LoginName, entered as input by the banker while logging in• AccountNo, entered by the banker, perhaps from a written inquiry
Output Component Parameters	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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 198](#).

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 input 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 205](#).

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 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 210](#).
- 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 154](#).

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
<p>Input / Output Input and output parameters can be linked to make data available between components or flows within the same business process test.</p> <p>See "Data Linkage Examples" on page 195.</p>	<p>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.</p> <p>An input parameter can receive:</p> <ul style="list-style-type: none"> • 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. <p>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.</p> <p>Note: You cannot set a default value for an output parameter.</p>
<p>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.</p> <p>See "Example: Parameter Promotion" on page 204.</p>	<p>Component parameters are parameters defined within a component. Component parameters can be input or output parameters.</p> <p>These parameters are available to:</p> <ul style="list-style-type: none"> • All subsequent steps in the same component. • Subsequent components in a flow or a test, provided that: <ul style="list-style-type: none"> • 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. <p>Flow parameters are parameters defined within a flow. These parameters</p>

Parameter Categories	Parameter and Description
	<p>are available to all components in the flow. Like component parameters, flow parameters can be input or output parameters.</p> <p>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.</p>
Local Local parameter values are not available outside of the component in which they are defined.	<p>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.</p> <p>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 73.</p>

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 ["Working with Business Process Testing" on page 24](#).

This task includes the following steps:

- ["Design data" below](#)
- ["Create parameters and set default values" below](#)
- ["Link parameters" on the next page](#)
- ["Promote parameters" on the next page](#)
- ["Set data values for each iteration" on the next page](#)
- ["Set data values for each test configuration" on the next page](#)
- ["Set data values for test instances" on the next page](#)
- ["Import and save parameters - optional" on the 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 194](#).
- Determine which parameters should be available at the component, flow, and test levels. For conceptual details, see ["Parameter Promotion Overview" on page 204](#).
- 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 210](#) and ["Business Process Test Configurations Overview" on page 154](#).

2. Create parameters and set default values

Define input and output parameters. For task details, see ["How to Create Parameters" on page 174](#).

Set default values for input parameters. For task details, see ["How to Set Parameter Values" on page 175](#).

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 198](#).

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 205](#).

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.

When working with dynamic configurations, business components within the same business process test or flow can be iterated a different number of times. To do this, set up component iterations using multiple sheets in the data table from within Microsoft Excel.

For task details on iterating components, see ["How to Set Data for Iterations" on page 214](#).

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 the *HP Application Lifecycle Management User Guide*.

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 214](#).

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 216](#).

Chapter 14: Parameters

This chapter includes:

- Parameters Overview 174
- How to Create Parameters 174
- How to Set Parameter Values 175
- Methods for Creating Parameters 177
- Parameters User Interface 178

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 177](#).

For conceptual details on parameters, see ["Data Handling Overview" on page 166](#).

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 the 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 177](#).

For task details on setting values for parameters, see ["How to Set Parameter Values" on the next page](#).

This task includes the following steps:

- ["Create parameters" below](#)
- ["Create parameters while defining manual steps" below](#)
- ["Generate many parameters at one time while defining iterations - optional" on the next page](#)
- ["Create parameters by promotion" on the next page](#)
- ["Results" on the 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 187](#).

Create parameters while defining manual steps

ALM enables you to create input and output component parameters directly in the Manual

Implementation tab as you define manual steps.

1. In the Business Components module > Manual Implementation 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 187](#) and ["Parameters Page" on page 179](#). 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 223](#).

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 205](#). For user interface details, see ["Promote Parameters Dialog Box" on page 207](#).

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 201](#).

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 \(for BPT Packaged Apps Kit users\)" on the next page](#)
- ["Specify a date value" on the next page](#)
- ["Specify a dynamic date value" on the next page](#)
- ["Encrypt the value" on the 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 182](#).

Specify a table value (for BPT Packaged Apps Kit users)

This parameter type, which represents a packaged app table (such as exists in SAP apps), is only available when a flow or business process test is learned using BPT Packaged Apps Kit. 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 190](#).

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 182](#).

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 182](#).

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 182](#).

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 179 . For task details, see "How to Create Parameters" on page 174 .
Component Step Definition	Provides the ability to define parameters "on the fly" directly when defining component steps.	Business Component module > Manual Implementation tab / Component Step Details dialog box	For task details, see "How to Create Parameters" on page 174 .
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 182 .
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 223 .
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	Test Plan module > Test Script tab > Select Components and Flows pane >  button	For user interface details, see "Promote"

Method	Purpose / Use	Available from	See Also
	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.	<p>Note: Also, if automatic promotion is enabled in Customization: Business Component / Test Plan modules > Parameters tab > New Parameter dialog box > Promote automatically to test level checkbox</p>	<p>Parameters Dialog Box on page 207.</p> <p>For task details, see "How to Promote Parameters" on page 205.</p>
Learn	BPT Packaged Apps Kit generates flow (or test) parameters automatically during the Learn process.	Test Plan module > Test Script tab >  Learn	For task details, see "How to Learn Flows and Business Process Tests" on page 130

Parameters User Interface

This section includes:

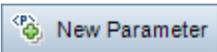
- [Parameters Page](#) 179
- [Set Values Dialog Box](#) 182
- [Dynamic Date Area](#) 184
- [Parameters Dialog Box](#) 186
- [New Parameter / Parameter Details Dialog Box](#) 187
- [Promotion Results Dialog Box](#) 189
- [Edit Table Parameter Dialog Box](#) 190

Parameters Page

This dialog box or tab enables you to view, select, and insert parameters for business components, flows, and business process tests.

To access	<p>Use one of the following:</p> <ul style="list-style-type: none"> 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	<ul style="list-style-type: none"> You can insert component parameters into your steps in the Manual Implementation 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 manual steps" on page 174. 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 <i>HP Unified Functional Testing User Guide</i>. 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	<ul style="list-style-type: none"> "How to Create Parameters" on page 174 "How to Set Parameter Values" on page 175
See also	<p>"Parameters Overview" on page 174</p>

User interface elements are described below:

UI Element	Description
	<p>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 187.</p> <p>Note: This button is disabled when selecting the Parameters tab from the Test Script tab in the Test Plan module.</p> <p>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.</p>
	<p>Delete. Deletes the currently selected parameter.</p> <p>Considerations</p> <ul style="list-style-type: none"> 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>

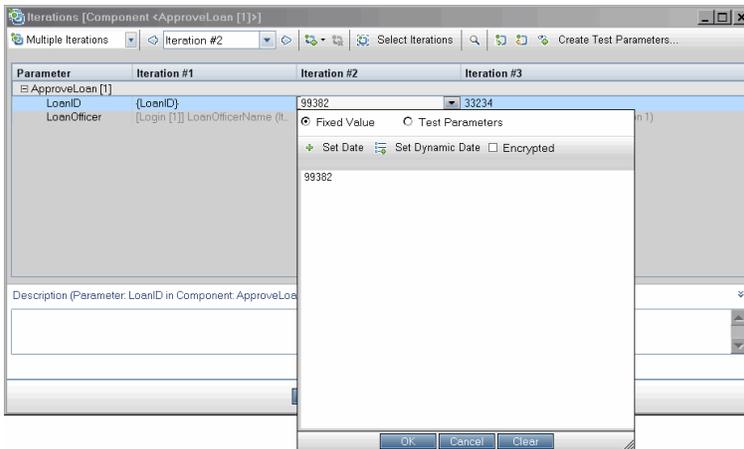
UI Element	Description
	<ul style="list-style-type: none"> If a business process contains the component parameter that you are deleting, the test may not run successfully. If automatic deletion is enabled in Customization, when prompted for confirmation, a Delete unused promoted parameters from the test level checkbox appears. If checked, the parameter is deleted from the test and/or flow levels if it is unused at those levels. <div style="background-color: #f0f0f0; padding: 5px; margin: 10px 0;"> <p>Notes</p> <ul style="list-style-type: none"> If the parameter is used at higher levels, automatic deletion does not delete it. If there are other unused parameters, not related to this delete operation, at the higher levels, automatic deletion does not delete them. </div> <p>For details on enabling automatic deletion, see the <i>HP Application Lifecycle Management Administrator Guide</i>.</p> <p>Not available when:</p> <ul style="list-style-type: none"> When selecting the Parameters tab from the Test Script tab in the Test Plan module. From the Parameters dialog box.
	<p>Refresh. Refreshes the component parameters so that they display the most up-to-date information.</p> <p>Not available when:</p> <ul style="list-style-type: none"> When selecting the Parameters tab from the Test Script tab in the Test Plan module. From the Parameters dialog box.
	<p>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 <i>HP Application Lifecycle Management User Guide</i>.</p> <p>Not available when: Selecting the Parameters tab from the Test Script tab in the Test Plan module.</p>
	<p>Select Columns. Opens the Select Columns dialog box, enabling you to determine which fields are displayed and their order. For more details, see the <i>HP Application Lifecycle Management User Guide</i>.</p> <p>Note: This option is not available when selecting the Parameters tab from the Test Script tab in the Test Plan module.</p>
	<p>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 187.</p>
	<p>Move Up. Moves the selected parameter up in the list.</p> <div style="background-color: #f0f0f0; padding: 5px; margin: 10px 0;"> <p>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.</p> </div>
	<p>Move Down. Moves the selected parameter down in the list.</p> <div style="background-color: #f0f0f0; padding: 5px; margin: 10px 0;"> <p>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.</p> </div>
	<p>Go To Resource. For use with configurations that access dynamic data. Opens the resource in the Test Resources module.</p> <p>Available from: Parameters tab from Test Plan module. This field is not available when</p>

UI Element	Description
	selecting the Parameters tab from the Test Script tab in the Test Plan module.
	<p>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.</p> <p>The Resource Preview dialog box is similar to the Resource Viewer tab in the Test Resources module. For user interface details, see the <i>HP Application Lifecycle Management User Guide</i>.</p> <p>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.</p>
	<p>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 <i>HP Application Lifecycle Management User Guide</i>.</p> <p>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.</p>
Sort By	<p>Specifies the order in which the parameters are listed, as defined in the Set Filter dialog box. For user interface details, see <i>HP Application Lifecycle Management User Guide</i>.</p>
Data Resource	<p>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 details, see <i>HP Application Lifecycle Management User Guide</i>.</p> <p>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.</p>
<p><grid filters></p> 	<p>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.</p> <p>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 <i>HP Application Lifecycle Management User Guide</i>.</p>
Filters	<p>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.)</p>
Used	<p>When checked, indicates that the component parameter is included in a step.</p>
Name	<p>Displays a descriptive name for the parameter.</p> <ul style="list-style-type: none"> 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	<p>Lists all defined input parameters.</p> <p>Available from: Parameters tabs</p>

UI Element	Description
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	<ol style="list-style-type: none"> 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	<ul style="list-style-type: none"> • The structure and fields of the dialog box change depending on: <ul style="list-style-type: none"> • 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 packaged app tables, are only available when a flow is learned using BPT Packaged Apps Kit. For details, see "Edit Table Parameter Dialog Box" on page 190. • You can provide a default value in case a value is not supplied for the run.
Relevant tasks	<ul style="list-style-type: none"> • "How to Create Parameters" on page 174 • "How to Set Parameter Values" on page 175
See also	"Parameters Overview" on page 174

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	<ol style="list-style-type: none"> 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	<ul style="list-style-type: none"> • 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. <p>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 ###-###-####.</p>

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

UI Element	Description
	<p>Displays a calendar area, enabling you to select a fixed date.</p> <p>Select Encrypted to encrypt the date—optional.</p>
	<p>Displays the dynamic date area, enabling you to select a date that is relative to a fixed date (such as Today, WeekAgo).</p> <p>For user interface details, see "Dynamic Date Area" on the next page.</p>
Encrypted	<p>Encrypts the fixed value (including dates).</p> <p>The characters you enter are encrypted and displayed as asterisks. You can modify an encrypted value. It remains encrypted.</p> <p>When cleared, erases the fixed value.</p> <p>Once a value is encrypted, it can only be modified in the Set Value tab, and not directly in the grid.</p> <p>Available when: Entering fixed values, and if it appears, Fixed Value is selected.</p>
<text area>	<p>Displays the current value for the parameter, and enables you to enter and modify the value.</p> <p>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.</p>

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	<ol style="list-style-type: none"> 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	<ul style="list-style-type: none"> 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 187

User interface elements are described below:

UI Element	Description
Promoted to:	<p>Displays promotion information for the selected parameter. Located at the bottom of the Test Parameters view.</p> <p>For conceptual details about parameter promotion, see "Parameter Promotion Overview" on page 204.</p> <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;"> <p>Example</p> <p>Promoted to: MyTestParam1</p> </div>

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 175
See also	"Set Values Dialog Box" on page 182

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>	<p>Contains a list of relative time frames:</p> <ul style="list-style-type: none"> Today Yesterday Tomorrow WeekAgo WeekFromToday

UI Element	Description
	<ul style="list-style-type: none"> • MonthAgo • MonthFromToday • YearAgo • YearFromToday <p>Date strings are not case-sensitive. For example, both [WeekAgo] and [Weekago] are valid values.</p>
<+/->	<p>Determines whether the <offset> and <time period> values are added to (+) or subtracted from (-) the <relative time frame> value.</p>
<offset>	<p>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.</p> <p>The maximum offset values that apply to time periods and used to modify date strings are:</p> <ul style="list-style-type: none"> • 5000 for years • 60000 for months • 240000 for weeks • 1680000 for days <div style="background-color: #f0f0f0; padding: 10px; margin-top: 10px;"> <p>Example</p> <p>[today + 5000 years] is valid.</p> <p>[today + 5001 years] is not valid.</p> </div>
<time period>	<p>Contains a list of time periods, enabling you to adjust the relative time frame. The options are:</p> <ul style="list-style-type: none"> • 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. <p>Time periods used to modify date strings are not case-sensitive. For example, both [YearFromToday + 5 Months] and [YearFromToday + 5 months] are valid.</p> <p>Spaces are not required in dynamic date values. For example, both [MonthFromToday + 3 w] and [MonthFromToday+3w] are valid.</p> <div style="background-color: #f0f0f0; padding: 10px; margin-top: 10px;"> <p>Examples</p> <ul style="list-style-type: none"> • [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] </div>

UI Element	Description
<expression area>	<p>Displays the expression created from selecting the <relative time frame>, <offset>, and <time period>.</p> <p>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.</p>

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	<ol style="list-style-type: none"> In the Business Components module, select a component with manual steps from the Components tree. In the Manual Implementation tab > Manual Implementation pane, do one of the following: <ul style="list-style-type: none"> 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 Manual Implementation pane. Place the cursor in a cell in the Descriptions column or the Expected Result column. Click Insert Parameter .
Relevant tasks	"How to Create Parameters" on page 174

User interface elements are described below:

UI Element	Description
	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 the 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 <i>HP Application Lifecycle Management User Guide</i> .
	Select Columns. Opens the Select Columns dialog box, enabling you to determine which fields are displayed and their order. For more details, see the <i>HP Application Lifecycle Management User Guide</i> .
	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 the next page .
	<p>Move Up. Moves the selected parameter up in the list.</p> <p>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.</p>
	<p>Move Down. Moves the selected parameter down in the list.</p> <p>Tip: In other tabs and dialog boxes, such as the in Iteration pages, parameters are displayed</p>

UI Element	
	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	<p>A descriptive name for the parameter.</p> <ul style="list-style-type: none"> 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	<p>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.</p> <p>Default values are saved and compared as strings, but can be specified according to different value types.</p>
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
- Promote parameters to higher levels, if enabled in Customization

The Parameter Details dialog box enables you to view the details for parameters.

To access	<ul style="list-style-type: none"> • 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. <p>Tip: You can also access the New Parameter dialog box from:</p> <ul style="list-style-type: none"> • 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 Manual Implementation tab's pane, click the Insert Parameter button and then click the New parameter button.
Important information	<p>Parameters are inserted after the currently-selected parameter.</p> <p>Parameter values are stored, compared, and processed as characters strings (even if values are specified as numeric or date values).</p> <p>If automatic parameter promotion is enabled in Customization, and you choose to automatically promote a parameter while creating the component, the Promotion Results dialog box is displayed. For details, see the "Promotion Results Dialog Box" on page 189.</p> <p>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</p>

	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	<ul style="list-style-type: none"> • "How to Create Parameters" on page 174 • "How to Set Parameter Values" on page 175
See also	<ul style="list-style-type: none"> • "Parameters Overview" on page 174 • "Parameters Page" on page 179

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: <ul style="list-style-type: none"> • 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.
Promote Automatically to Test Level	If checked, promotes the parameter to either the flow or the test levels. For details, see the tip under "How to Promote Parameters" on page 205 . Available when: <ul style="list-style-type: none"> • Creating a new parameter (not when editing an existing parameter). • Automatic parameter promotion is enabled in Customization. For details on enabling, see the <i>HP Application Lifecycle Management Administrator Guide</i>.
ID	The ID of the parameter.
Parameter Name	A descriptive name for the parameter. <ul style="list-style-type: none"> • Parameter names must be in English. • Parameter names must begin with a letter and cannot include spaces or any of the following characters: ! @ # \$ % ^ & * () + = \ { } ; ' : " , . / < > ? ~ ` [] - ~ <p>If the parameter is in a manual test that was converted to a business component, any invalid character</p>

UI Element	Description
	<p>is automatically replaced by an underscore (_).</p> <ul style="list-style-type: none"> 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.
Details tab	Enables you to modify details for the parameter.
Mapping Status	<p>The name of the corresponding external data resource parameter (column) to which the parameter is mapped.</p> <p>This UI element is blank if no parameter is mapped.</p> <p>Available from: Test Parameter Details dialog box</p>
Order	<p>The position of the parameter in the set of parameters.</p> <p>Available for: Test and flow parameters</p>
Default Value tab	<p>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.</p> <p>For user interface details, see "Set Values Dialog Box" on page 182.</p> <p>Available from: Input Parameters sub-tabs</p>
Description tab	A description of the parameter's purpose, or other descriptive text. This tab enables you to modify details for the parameter.

Promotion Results Dialog Box

The Promotion Results dialog box enables you to export the results of automatic parameter promotion while creating a new component or flow.

To access	<ul style="list-style-type: none"> New Test dialog box: When creating a flow with automatic promotion of its parameters, the dialog box is displayed after you click OK if at least one parameter could not be promoted. New Component dialog box: When creating a component with automatic promotion of its parameters, the dialog box is displayed after you click OK if at least one parameter could not be promoted.
Important information	Automatic parameter promotion is enabled in Customization. For details on enabling, see the <i>HP Application Lifecycle Management Administrator Guide</i> .
Relevant tasks	"How to Create Parameters" on page 174
See also	"New Parameter / Parameter Details Dialog Box" on page 187

User interface are described below:

UI Element	Description
<Successful results>	Information about the parameters that were successfully promoted to the next level.
<Unsuccessful result grid>	<p>Information about the parameters that were not promoted to the next level, in a grid.</p> <ul style="list-style-type: none"> Test ID. The ID of the test or flow to which the parameter could not be promoted.

UI Element	Description
	<ul style="list-style-type: none"> • Test Name. The name of the test or flow to which the parameter could not be promoted. • Reason. The reason why the parameter could not be promoted.
Export	Exports the information from <unsuccessful result grid> to a file using the Export All Grid Data dialog box in one of the following formats: Microsoft Excel, Microsoft Word, HTML, or text.

Edit Table Parameter Dialog Box

This dialog box enables you to modify table values for parameters created during the Learn process. The table parameter definition includes the table structure and default values.

To access	<p>You can access the Edit Table Parameter dialog box on the component, flow, test, or test set level.</p> <ul style="list-style-type: none"> • 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	<ul style="list-style-type: none"> • This dialog box is only available for BPT Packaged Apps Kit users, and only available for learned components that contain app-specific table parameters. • Table parameters are available for use in: <ul style="list-style-type: none"> • Components • Flows • Tests • Test sets • Iterations • Run conditions
Relevant tasks	"How to Set Parameter Values" on page 175
See also	"How to Learn Flows and Business Process Tests" on page 130

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).

UI Element (A-Z)	Description
Fixed value	<p>Displays the Fixed Values view.</p> <p>Available when: Accessing the Edit Table Parameter dialog box via an iterations dialog box in the Test Plan module.</p>
Flow parameter	<p>Displays the Flow/Test Parameter view, enabling you to select a different flow table parameter.</p> <p>Available when: Accessing the Edit Table Parameter dialog box via an iterations dialog box in the Test Plan module.</p>
Test parameter	<p>Displays the Flow/Test Parameter view, enabling you to select a different test table parameter.</p> <p>Available when: Accessing the Edit Table Parameter dialog box via an iterations dialog box in the Test Plan module.</p>
Add Row	<p>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.</p> <p>Available in: Fixed Value view</p>
Delete Row	<p>Deletes the selected row from the table parameter grid.</p> <p>Available in: Fixed Value view</p>
Add Column	<p>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.</p> <p>Available in: Fixed Value view</p>
Delete Column	<p>Deletes the selected column from the table parameter grid.</p> <p>Available in: Fixed Value view</p>
Import	<p>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.</p> <p>Available in: Fixed Value view</p>
Export	<p>Exports the table parameter definition to a .csv (comma separated values) file.</p> <p>Available in: Fixed Value view</p>
<column header row>	<p>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.</p> <p>Available in: Fixed Value view</p>
<table cell>	<p>Enables editing of the default value for a table cell. Select the cell and enter the default value.</p> <p>Available in: Fixed Value view</p>

Chapter 15: Parameter Linkage

This chapter includes:

- [Parameter Linkage Overview](#)194
- [How to Link Data](#) 198
- [Parameter Linkage User Interface](#) 200

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 > 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 198](#).

This section also includes the following topics:

- ["Linkage and Iterations" below](#)
- ["Data Linkage Examples" on the 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 214](#).

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 196](#).
- **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 197](#).

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 127](#).

Data Linkage Examples

This topic provides examples that demonstrate how to link data.

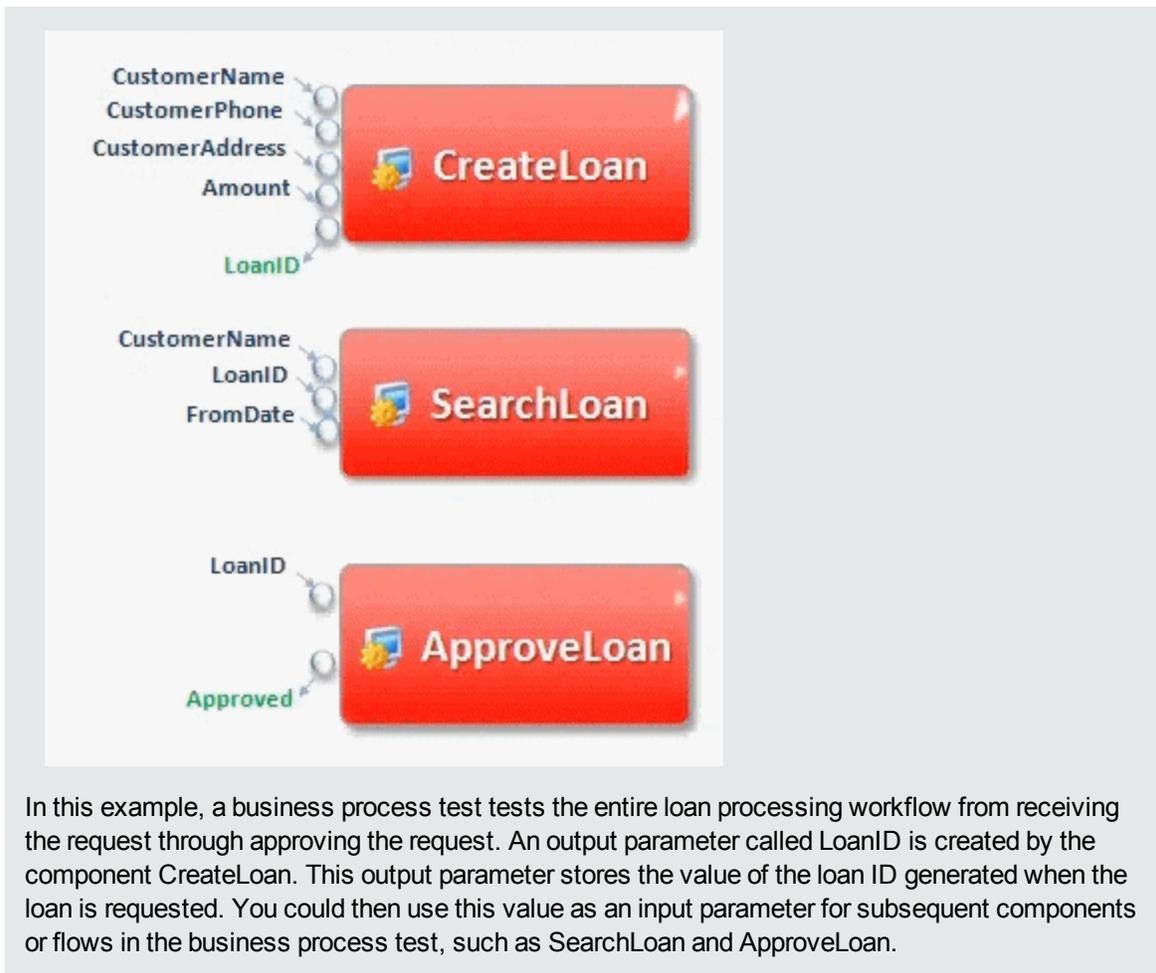
The following examples are provided:

- ["Example: Linking Input and Output Component Parameters—Graphical Representation" below](#)
- ["Iterations Example \(“1-to-n” Relationship\)—Graphical Representation" on the next page](#)
- ["Iterations Example \(“n-to-n” Relationship\)—Graphical Representation" on page 197](#)
- ["Iterations Examples—Representation in Iterations Dialog Box" on page 198](#)

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.

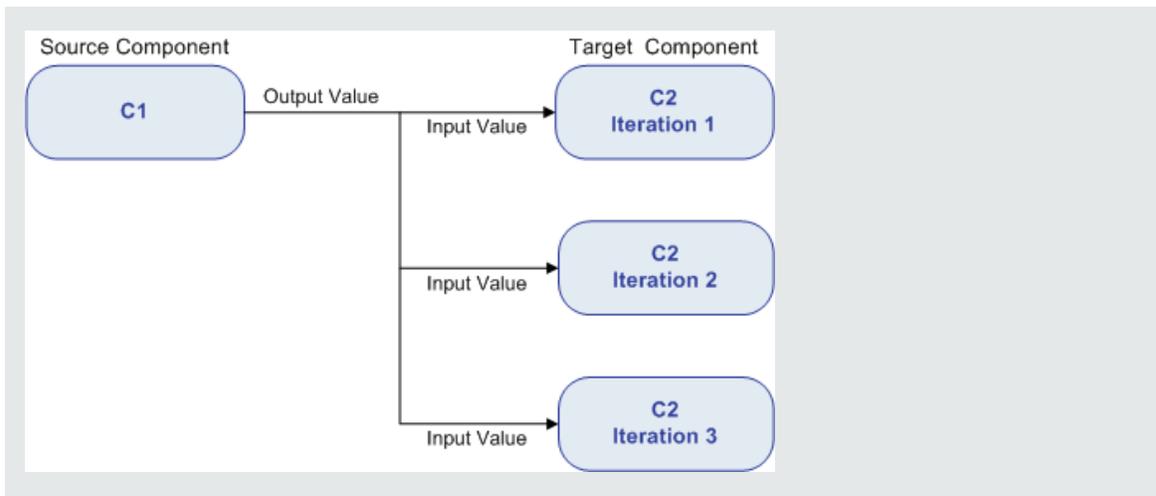


In this example, a business process test tests the entire loan processing workflow from receiving the request through approving the request. An output parameter called LoanID is created by the component CreateLoan. This output parameter stores the value of the loan ID generated when the loan is requested. 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).

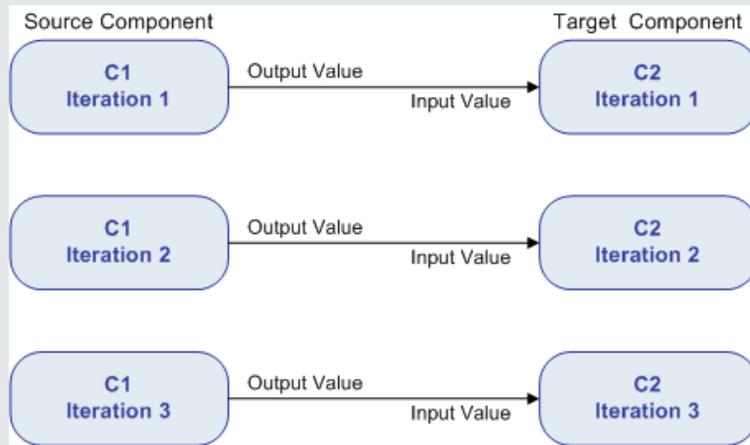
In this case, the same output value of the source component is used in each iteration of the target component.



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.



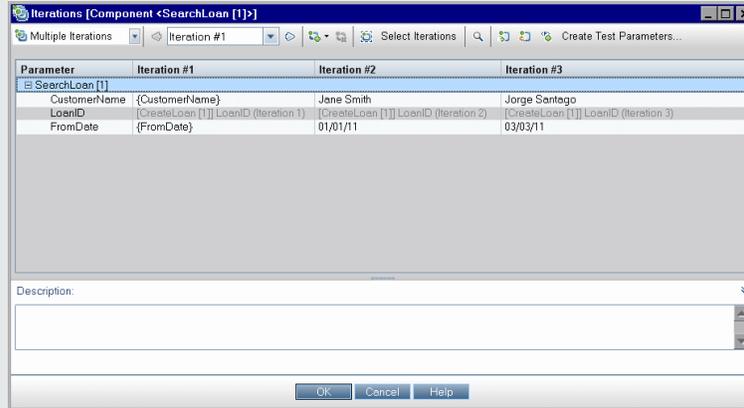
The number of iterations in the relationship between the components can be the number of actual iterations in each component or group, or the number of iterations selected in a partial range.

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 221](#).

Iterations Examples—Representation in Iterations Dialog Box

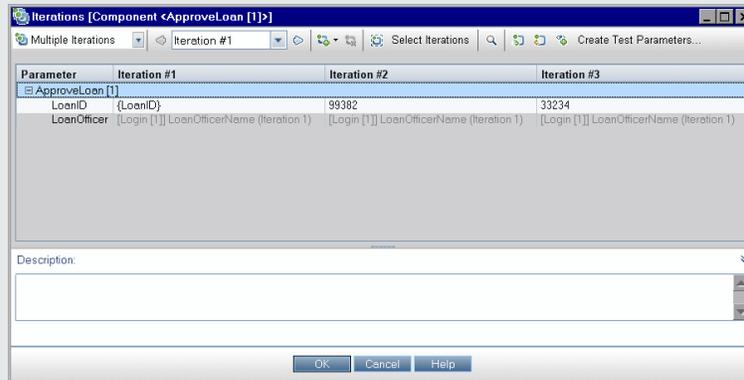
In the example below, the source component CreateLoan has an “n-to-n” (“3 to 3”) relationship with the target component SearchLoan. For each iteration of component input parameter LoanID in SearchLoan, there is a corresponding iteration of component output parameter LoanID from CreateLoan.



The screenshot shows the 'Iterations [Component <SearchLoan [1]>]' dialog box. It features a table with three columns: 'Parameter', 'Iteration #1', 'Iteration #2', and 'Iteration #3'. The 'SearchLoan [1]' component is expanded, showing three parameters: 'CustomerName', 'LoanID', and 'FromDate'. The 'LoanID' parameter is linked to the 'LoanID' output of the 'CreateLoan [1]' component for each iteration. The 'FromDate' parameter is linked to the 'FromDate' output of the 'CreateLoan [1]' component for each iteration. The 'CustomerName' parameter is linked to the 'CustomerName' output of the 'CreateLoan [1]' component for each iteration. The 'LoanID' parameter values are 'Jane Smith', 'Jorge Santiago', and 'Jorge Santiago' for iterations 1, 2, and 3 respectively. The 'FromDate' parameter values are '01/01/11' and '03/03/11' for iterations 1 and 2 respectively. The 'CustomerName' parameter values are 'Jane Smith', 'Jorge Santiago', and 'Jorge Santiago' for iterations 1, 2, and 3 respectively. The dialog box also includes a 'Description' field and 'OK', 'Cancel', and 'Help' buttons.

Parameter	Iteration #1	Iteration #2	Iteration #3
SearchLoan [1]			
CustomerName	{CustomerName}	Jane Smith	Jorge Santiago
LoanID	[CreateLoan [1]] LoanID (iteration 1)	[CreateLoan [1]] LoanID (iteration 2)	[CreateLoan [1]] LoanID (iteration 3)
FromDate	{FromDate}	01/01/11	03/03/11

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.



The screenshot shows the 'Iterations [Component <ApproveLoan [1]>]' dialog box. It features a table with three columns: 'Parameter', 'Iteration #1', 'Iteration #2', and 'Iteration #3'. The 'ApproveLoan [1]' component is expanded, showing three parameters: 'LoanID', 'LoanOfficer', and 'LoanOfficer'. The 'LoanID' parameter is linked to the 'LoanID' output of the 'ApproveLoan [1]' component for each iteration. The 'LoanOfficer' parameter is linked to the 'LoanOfficer' output of the 'Login [1]' component for each iteration. The 'LoanID' parameter values are '99382' and '33234' for iterations 1 and 2 respectively. The 'LoanOfficer' parameter values are '[Login [1]] LoanOfficerName (iteration 1)' for iterations 1, 2, and 3 respectively. The dialog box also includes a 'Description' field and 'OK', 'Cancel', and 'Help' buttons.

Parameter	Iteration #1	Iteration #2	Iteration #3
ApproveLoan [1]			
LoanID	{LoanID}	99382	33234
LoanOfficer	[Login [1]] LoanOfficerName (iteration 1)	[Login [1]] LoanOfficerName (iteration 1)	[Login [1]] LoanOfficerName (iteration 1)

For user interface details on the Iterations dialog box, see ["Iteration Pages" on page 219](#).

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 170](#).

- For user interface details, see "[I/O Parameters Dialog Box](#)" on page 201 and "[Select Output Parameter Dialog Box](#)" on page 201.

This task includes the following steps:

- "[Design linkage](#)" below
- "[Display the parameters for the relevant business process test or flow](#)" below
- "[Link the parameters](#)" below

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 page 201.

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 page 201.
- 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](#) 201
- [Select Output Parameter Dialog Box](#) 201

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 198
See also	"Parameter Linkage Overview " on page 194

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	<ul style="list-style-type: none"> 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 198
See also	"Parameter Linkage Overview " on page 194

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 16: Parameter Promotion

This chapter includes:

- [Parameter Promotion Overview](#)204
- [How to Promote Parameters](#) 205
- [Parameter Promotion User Interface](#)206

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.

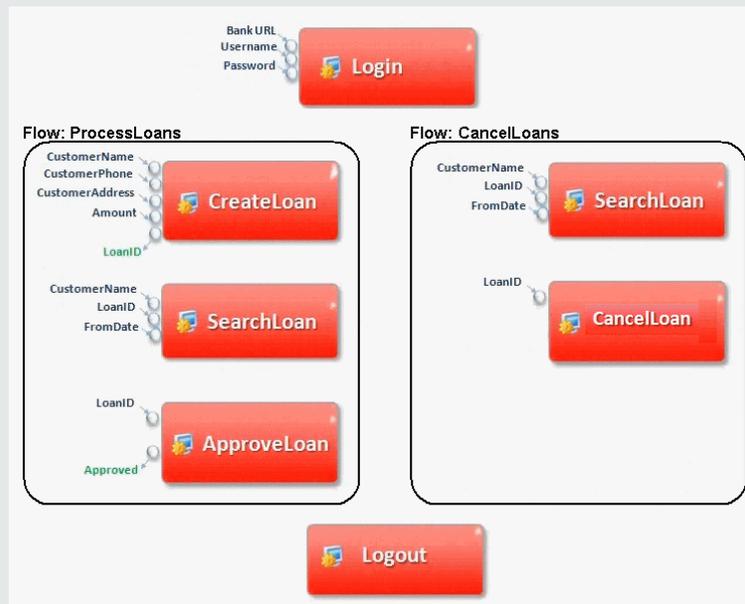
Tip: You can also set Business Process Testing's default behavior to enable you to automatically promote parameters to the next levels when adding a component or flow parameter. For details, see the *HP Application Lifecycle Management Administrator Guide*.

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 link the LoanID flow output parameter to a test level parameter, such as LoanID (with the same name). The test level parameter called LoanID is accessible to every component and flow in the business process test.

For task information, see ["How to Promote Parameters" on the next page](#).

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 165](#).
- For an alternative method for promoting parameters, see the tip below.

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 the 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 ["Select Components and Flows Pane" on page 125](#).
- 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 page 207](#).

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 208.

Tip: If automatic promotion is enabled in Customization, you can also promote parameters automatically to all higher levels when creating a business component or flow parameter.

1. In the Business Component / Test Plan modules, select the component or flow.
2. Then select the Parameters tab and click **New Parameter**.
3. In the New Parameter dialog box, check the **Promote automatically to test level** checkbox before saving.

Parameter Promotion User Interface

This section includes:

- [Promote Parameters Dialog Box](#)207
- [Test/Flow Parameters Promotion Status Dialog Box](#)208

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	<ul style="list-style-type: none"> 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 205
See also	"Parameter Promotion Overview" on page 204

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	<p>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.</p> <p>If checked, the Test/Flow Parameter Name suffix is removed, because the parameter may have been promoted from multiple components.</p> <p>Default: Unchecked.</p>

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).

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 205
See also	"Parameter Promotion Overview" on page 204

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.
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 17: Iterations

This chapter includes:

- Iterations Overview210
- How to Set Data for Iterations214
- How to Import and Export (Save) Parameter Values216
- Iterations User Interface218

Iterations Overview

This chapter describes how to 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 214](#).
- 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 the *HP Application Lifecycle Management User Guide*.
- **Product Feature Movie:** To view a movie that demonstrates how to handle Business Process Testing parameters, select **Help > 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 business component runs in a test with a dynamic data table (using Microsoft Excel).
- A flow runs in a test (using the Flow Iterations dialog box or the Iterations 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 Configurations tab > Data tab)

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.

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.
- You can create iterations for a component that orders books online. Each time the test runs, several book orders are made because the component is iterated in the test. A different number of books are ordered by the same test by specifying different book names for varying numbers of component iterations.

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 221](#).

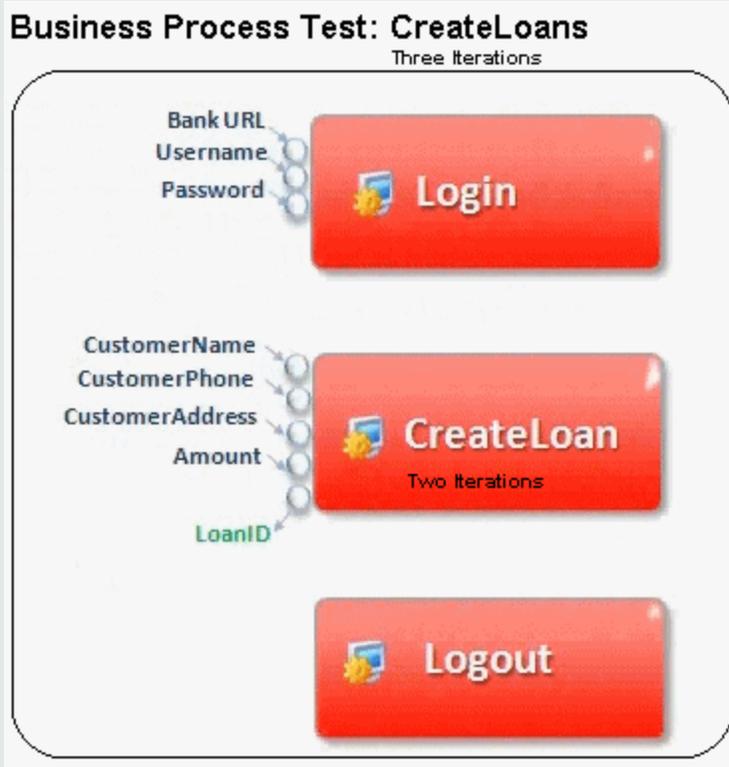
In addition, you could use the same business process test or flow definition for different scenarios. Different combinations of iterations can be run for different test runs. Iteration ranges for test configurations that use dynamic data are specified using one of the following:

- The Data Resource Settings dialog box. For user interface details, see the *HP Application Lifecycle Management User Guide*.
- Microsoft Excel. For details, see the *HP Application Lifecycle Management User Guide*.

When BPT Packaged Apps Kit users run 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 239.

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" below](#).

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.
- Member components cannot be iterated a different number of times than the group to which they belong.

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

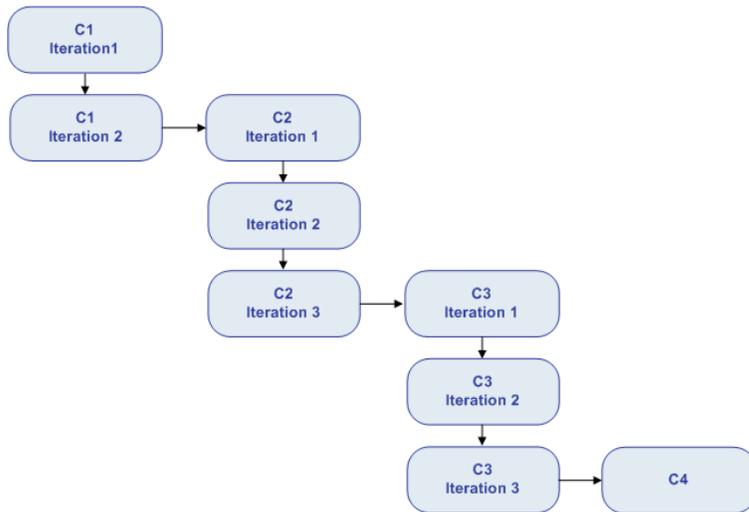
This topic demonstrates iterations for this scenario:

- ["Without Grouping" below](#)
- ["With Grouping" on the 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:

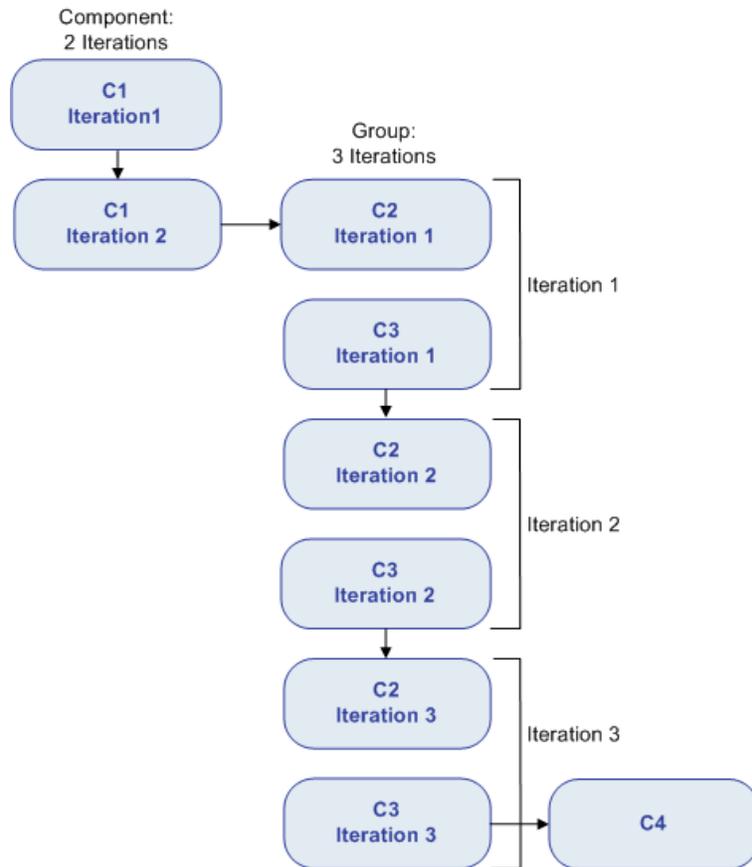


With Grouping

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 170.

This task includes the following steps:

- ["Prerequisites and considerations"](#) on the next page
- ["Iterate components, flows, and groups"](#) on the next page
- ["Iterate flow run instances"](#) on page 216
- ["Iterate test configurations"](#) on page 216
- ["Iterate according to Imported data - optional"](#) on page 216
- ["Results"](#) on page 216

• 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 194.
- 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 must 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 219.

Tip: Alternatively, when working with dynamic data sets, if you prefer working directly in a spreadsheet, you can use Microsoft Excel. For details, see the *HP Application Lifecycle Management User Guide*.

Set the value of each input parameter for each iteration. For user interface details, see "[Set Values Dialog Box](#)" on page 182.

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 221.

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 at runtime. 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 219](#) 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 219](#).

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 214](#).

This task includes the following steps:

- ["Import data from an external file" below](#)
- ["Export \(save\) data to an external file" on page 218](#)

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.

1. 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.

2. 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 219.

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.

3. 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 222.

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.

1. 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 the next page](#).
2. 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](#)219
- [Select Iterations Dialog Box](#)221
- [Map Imported Data Dialog Box](#)222
- [Create Flow/Test Parameters Dialog Box](#)223

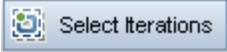
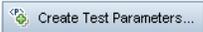
Iteration Pages

These pages enable you to set parameter values for iterations of components, groups, flows, test configurations, and test run instances.

To access	<p>Do one of the following.</p> <ul style="list-style-type: none"> • 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. <div style="background-color: #f0f0f0; padding: 5px; margin: 10px 0;"> <p>Tip: By default, the Iterations column is not displayed in the Execution Grid tab. For details on how to display the column, see the <i>HP Application Lifecycle Management User Guide</i>.</p> </div> <ul style="list-style-type: none"> • To iterate test configurations that access static data: From the Test Plan module > Test Configurations tab, click the Data tab. • To iterate test configurations that access dynamic data: Add rows to a data table in Microsoft Excel. For details, see the <i>HP Application Lifecycle Management User Guide</i>. • To iterate test instances: From the Test Lab module > Test Instance Details—Execution Settings View, click the Test Iterations tab.
Important information	<ul style="list-style-type: none"> • 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	<ul style="list-style-type: none"> • "How to Set Data for Iterations" on page 214 • <i>HP Application Lifecycle Management User Guide</i>
See also	<ul style="list-style-type: none"> • "Iterations Overview" on page 210 • "Group Iterations Overview" on page 212

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

UI Element	Description
	<p>Multiple Iterations View / Single Iteration View. Enables you to toggle between:</p> <ul style="list-style-type: none"> • A view that shows all iterations in one dialog box. • A view that shows only one iteration at a time in the dialog box. <p>Available from: Iterations dialog boxes in the Test Plan module</p>
	<p>Previous / Next Iteration. Displays the next or previous iteration in the dialog box.</p> <p>Available from: Iterations dialog boxes in the Test Plan module</p>
	<p>Iteration. Displays the iteration currently in focus in the dialog box, and enables you to select a different iteration on which to focus.</p> <p>Not available from: Iterations dialog boxes in the Test Plan module</p>
	<p>Add Iteration. Adds an iteration for the entity (component, group, test, or flow).</p> <p>The following options may be available, depending on the context in which you opened the</p>

UI Element	Description
	<p>Iteration page. If these options are not available, iterations are added by default without setting parameter values.</p> <ul style="list-style-type: none"> • 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. <p>Disabled when: The component or flow for which you are adding an iteration is in a group.</p>
	<p>Remove/Delete Iteration. Deletes the selected iteration from the entity (component, group, test, or flow).</p> <p>Disabled when: The component or flow for which you are adding an iteration is in a group.</p>
	<p>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.</p> <p>Disabled when:</p> <ul style="list-style-type: none"> • The component or flow for which you are selecting iterations is in a group • There is only one iteration.
	<p>Find. Enables you to find a parameter value, a parameter, or a component.</p> <p>For details on Find and Replace functionality, see the <i>HP Application Lifecycle Management User Guide</i>.</p> <p>Available from: Iterations dialog boxes</p>
	<p>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.</p> <div style="background-color: #f0f0f0; padding: 10px; border: 1px solid #ccc;"> <p>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 the <i>HP Application Lifecycle Management User Guide</i>.</p> </div>
	<p>Export. Enables you to export data values to a .csv (comma separated value) file from the iteration data values specified in the Iteration page.</p> <div style="background-color: #f0f0f0; padding: 10px; border: 1px solid #ccc;"> <p>Examples</p> <ul style="list-style-type: none"> • 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. </div>
	<p>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.</p> <p>Available from: Test Plan module > Test Configurations tab > Data tab for static configurations</p>
	<p>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.</p>

UI Element	Description
	For user interface details, see "Create Flow/Test Parameters Dialog Box" on page 223 .
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 Iterations 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 182 . 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.
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	<p>Perform one of the following:</p> <ul style="list-style-type: none"> • 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: <ol style="list-style-type: none"> 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 the <i>HP Application Lifecycle Management User Guide</i> .
Relevant tasks	"How to Set Data for Iterations" on page 214
See also	"Iterations Overview" on page 210

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

UI Element	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 216
See also	"How to Set Data for Iterations" on page 214

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	<ul style="list-style-type: none"> • 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 214
See also	"Iterations Overview" on page 210

User interface elements are described below:

UI Element	Description
Flow/Test Parameter Creation Criteria	<ul style="list-style-type: none"> • 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	<p>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.</p> <p>Example. Provides an example of the generated parameter name based on the selected criteria.</p>
Create flow/test parameters for	<ul style="list-style-type: none"> • 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.

Part 5: Test Runs

Chapter 18: Introducing Test and Flow Execution

This chapter includes:

- Test and Flow Execution Overview228
- How to Run Business Process Tests and Flows Manually 228
- How to Run Automated Business Process Tests and Flows229
- Run Progress Pane Hierarchy232
- Test and Flow Execution User Interface234

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 for each iteration, 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 the 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 158.

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 ["Working with Business Process Testing"](#) on page 24.

This task includes the following steps:

1. ["Add iterations to business process tests and flows - optional"](#) below
2. ["Use the Manual Runner or Sprinter to run the manual tests"](#) on the next page
3. ["View and analyze test results"](#) on the 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 214.

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 235.

BPT Packaged Apps Kit users can run tests and flows in change detection mode, to see if changes were made to the packaged app 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 241.

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.

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 "[Working with Business Process Testing](#)" on page 24.

This task includes the following steps:

- "[Prerequisites](#)" on the next page
- "[Debug tests and flows - optional](#)" on the next page
- "[Add iterations to business process tests and flows - optional](#)" on the next page
- "[Determine if tests and flows should be run in change detection mode](#)" on the next page
- "[Use the Automatic Runner to run the tests](#)" on page 231
- "[Resolve detected changes](#)" on page 231
- "[View run results](#)" on page 231

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.

Example

To enable ALM access to a specific UFT client, open UFT on the host computer, select **Tools > Options > GUI Testing > Test Runs**, and ensure that **Allow other HP products to run tests and components** is selected.

- Test results are, by default, saved in ALM. When running an automated test from within ALM you can avoid saving large test results by setting the **UPLOAD_RESULTS_AFTER_TEST_RUN** site parameter. For details, see the *HP Application Lifecycle Management Administrator Guide*.
- When running default test sets on remote hosts, ALM communicates directly with the testing host to initiate the test. See your testing tool documentation for information about what ports must be opened and what other security settings must be adjusted to enable communication.
- To run a UFT component from ALM, you must have the necessary ALM permissions. For details, see the *HP Application Lifecycle Management Administrator Guide*.
- 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*.
- In preparation for running UFT business process tests remotely, you must first register your ALM client in common mode on the remote machine. For more information, refer to the *HP Application Lifecycle Management Installation and Upgrade Guide*.

Note: To run a UFT business process test on a remote Windows 2012 machine, if the COM+ Network Access feature is not installed, you must restart the Windows 2012 remote machine after registration.

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 158.

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 214.

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 BPT Packaged Apps Kit enabled, you can run automated tests containing learned flows in change detection mode. This enables you to determine if the packaged app being tested has been modified. For task details on detecting changes, see ["How to Run Tests in Change Detection Mode" on page 241](#).

Tip: 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**.

Notes

- Dynamic dates are converted to the actual dates they represent.
- If running an automatic test that contains component requests with manual implementations, those component requests are ignored during the run.

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

6. Resolve detected changes

If you have BPT Packaged Apps Kit enabled, and you ran the business process test or flow in change detection mode, you can resolve changes detected in the packaged app during the run. For task details on resolving changes, see ["How to View and Resolve Detected Changes" on page 243](#).

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 UFT reports.** If you have Unified Functional Testing or the Unified Functional Testing Add-in for Business Process Testing 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. The add-ins are available from the HP Application Lifecycle Management Add-ins page (**Help > Add-ins**).

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*.

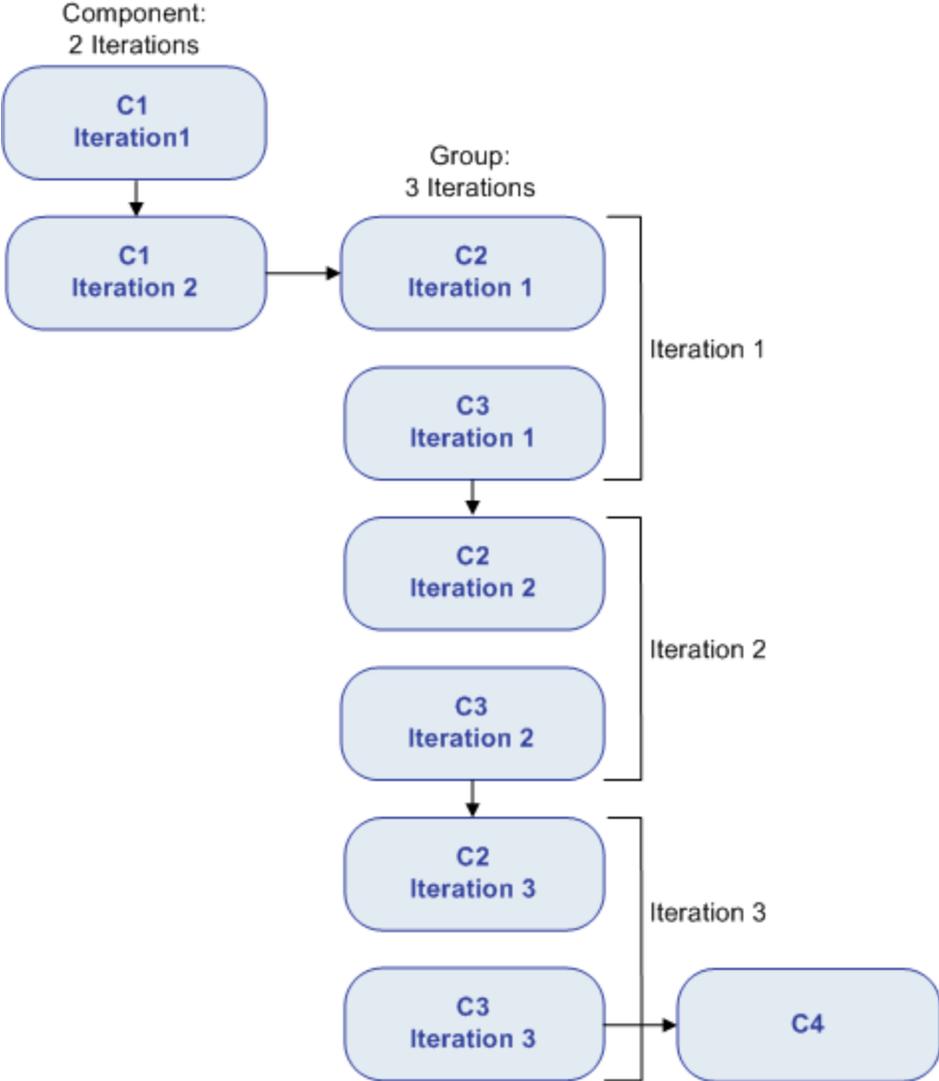
Run Progress Pane Hierarchy

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:

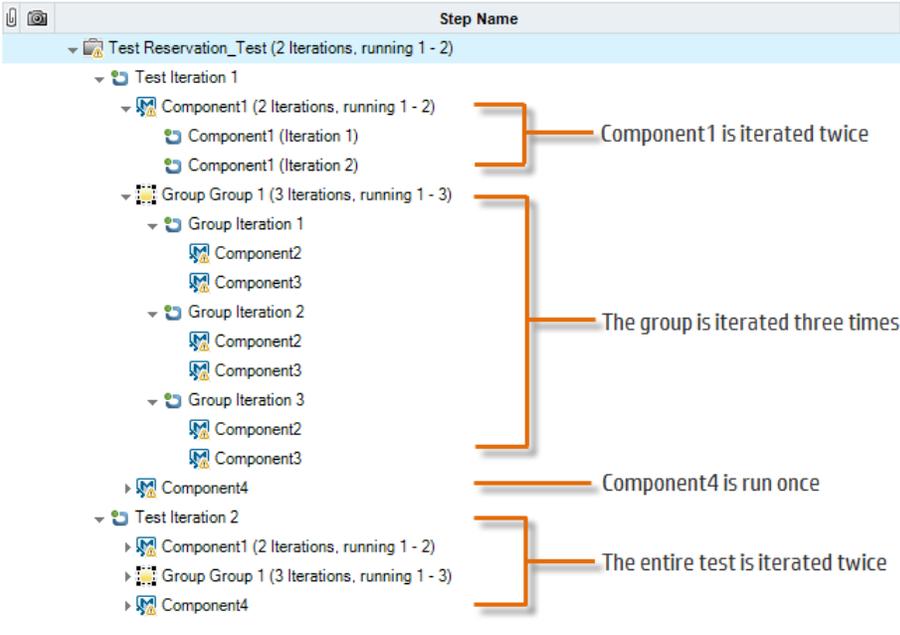
- the first iteration of C1, then the second iteration of C1, followed by
- the first iteration of the group comprising C2 and C3, followed by
- the second iteration of the group, followed by
- the third iteration of the group, followed by the component C4.

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](#) 235
- [Test Properties Dialog Box](#) 238

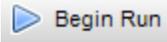
Manual Runner Wizard for Business Process Testing

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 228
Wizard map	This wizard contains: "Manual Runner: Run Details Page for Business Process Testing" below > "Manual Runner: Step Details Page for Business Process Testing" on the next page
See also	"Test and Flow Execution Overview" on page 228 <i>HP Application Lifecycle Management User Guide</i>

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	<ul style="list-style-type: none"> General information about this wizard is available here: "Manual Runner Wizard for Business Process Testing" above. To advance to the next page of this wizard, click  . 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 <i>HP Application Lifecycle Management User Guide</i>.
Wizard map	The "Manual Runner Wizard for Business Process Testing" above contains: "Manual Runner: Run Details Page for Business Process Testing" above > "Manual Runner: Step Details Page for Business Process Testing" on the next page
See also	"Test and Flow Execution Overview" on page 228

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

UI Element	Description
Operating System Info	<p>Opens the Operating System Information dialog box. You can edit the following:</p> <ul style="list-style-type: none"> 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.
Test Details pane	Provides the description of the business process test or flow as defined in the Test Plan module.

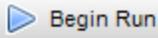
Manual Runner: Step Details Page for Business Process Testing

This wizard page enables you to do the following during the run:

- Set the pass/fail status for each component step
- Rename component steps for display in the Run Results View

After you finish executing the business process test or flow, you can save your component steps with the changes you make.

Note: These changes affect the Run Results View, not the actual step definitions in the Business Components module.

To access	From the Manual Runner: Run Details page, click  .
Important information	<ul style="list-style-type: none"> • General information about this wizard is available here: "Manual Runner Wizard for Business Process Testing" on the previous page. • 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 <i>HP Application Lifecycle Management User Guide</i>.
Wizard map	The "Manual Runner Wizard for Business Process Testing" on the previous page contains: "Manual Runner: Run Details Page for Business Process Testing" on the previous page > "Manual Runner: Step Details Page for Business Process Testing" above
See also	"Test and Flow Execution Overview" on page 228

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

UI Element	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 parameter link you clicked previously.
<Grid fields>	<p>View or edit details about the test steps. The grid displays a hierarchical tree of each test or flow iteration, business component, and component iteration contained in the business process test or flow.</p> <p>For details on the available fields, see <i>HP Application Lifecycle Management User Guide</i>.</p> <p>One of the grid fields is Status. When running a business process test or flow manually, there is one additional status available:</p> <ul style="list-style-type: none"> • Ignored. The business component or flow was ignored because it did not meet its defined run conditions. <p>The status of any steps related to this business component or flow is automatically set to No Run and the steps are skipped when you click Next.</p>

UI Element	Description
	<p>Note: This status cannot be chosen; it is set by Business Process Testing during the manual run.</p>
Description	<p>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.</p>
Expected	<p>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.</p> <p>Input and output parameters within the text in the Description and Expected boxes are indicated by sets of angle brackets:</p> <p>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, <>.</p> <p>Output parameters. Output parameter names are enclosed in three sets of angle brackets, for example, <<<Order_Number>>>.</p>
Actual	<p>Indicates the actual result of running the step during the test or flow.</p>
Parameters area	<p>This area:</p> <ul style="list-style-type: none"> • Displays the input parameter values for you to use when manually running the component selected in the run tree. • Enables you to enter values output by the application that you can use later in the test or flow run. • Enables you to enter values for testing the impact of different parameter values on the application during the run. <p>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 174.</p> <p>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 174.</p>
Input Parameters	<p>Name. 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.</p> <p>Value. Displays the input parameter values defined for the selected component.</p> <p>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.</p> <p>Note: Input parameters that are defined as the default values do not have any associated link.</p>
Output Parameters	<p>Name. 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.</p> <p>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.</p> <p>Source. Input parameters derived from the output parameter of a previous step are shown as links in the Source cells in the Input Parameters section.</p> <p>The name of the link indicates the name of the source component and the name of the relevant parameter, for example, [Component1] Order_Number.</p> <p>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.</p>

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	<ul style="list-style-type: none"> • 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 228
See also	<ul style="list-style-type: none"> • "Test and Flow Execution Overview" on page 228 • "Manual Runner Wizard for Business Process Testing" on page 235

User interface elements are described below:

UI Element	Description
Details tab	Lists the business process test or flow fields. <ul style="list-style-type: none"> • For details on Business Process Testing fields, see "Test Plan Module Fields for Business Process Testing" on page 116. • For other fields common to most test types, see <i>HP Application Lifecycle Management User Guide</i>.
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 19: Change Detection and Resolution

Available for BPT Packaged Apps Kit Users

This chapter includes:

- [Change Detection and Resolution Overview](#)240
- [How to Run Tests in Change Detection Mode](#) 241
- [How to View and Resolve Detected Changes](#) 243
- [Change Detection and Resolution User Interface](#) 244

Change Detection and Resolution Overview

Available for BPT Packaged Apps Kit Users

This chapter describes how you can detect changes in your packaged app that necessitate modifications to your flows and business process tests, and how you can instruct BPT Packaged Apps Kit to make these modifications automatically.

Change detection and resolution simplify component maintenance, thereby reducing the effort needed to invest in automated testing.

Note:

- If BPT Packaged Apps Kit is not enabled for your project, this feature is not available. Access to BPT Packaged Apps Kit must be provided by your HP Application Lifecycle Management (ALM) Project Administrator using Project Customization. An administrator can enable BPT Packaged Apps Kit in ALM by clicking , then selecting **Customize > Business Process Test** and selecting the **Enable BPT Packaged Apps Kit** checkbox. For details, see the *HP Application Lifecycle Management Administrator Guide*.
- Unified Functional Testing must be installed with its SAP plug-in to use this feature.

Change Detection Overview

BPT Packaged Apps Kit enables you to run business process tests and flows on a packaged app in Change Detection mode. This enables you to determine how the packaged app has changed since the business process 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 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 by BPT Packaged Apps Kit, you can resolve the change in Unified Functional Testing. For details, see the section about detecting and resolving changes using Change Detection mode in the *Unified Functional Testing User Guide*.

For a list of supported changes and resolutions, see the *Unified Functional Testing User Guide*.

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, BPT Packaged Apps Kit 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 business process test or flow that includes verification of the additional field.

Similarly, if a field was removed from the screen, BPT Packaged Apps Kit 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

Available for BPT Packaged Apps Kit Users

This task describes how to detect changes made to your packaged app by running business process tests and flows for the application in change detection mode.

You can detect changes to an individual flow, a business process test, or a test set.

Note: This task is part of a higher-level task. For details, see ["How to Run Automated Business Process Tests and Flows" on page 229](#).

This task includes the following steps:

- ["Prerequisites" below](#)
- ["Locate the flow or business process test" on the next page](#)
- ["Start the run " on the next page](#)
- ["Run the flow or business process test" on the next page](#)
- ["Results" on the next page](#)

1. Prerequisites

- Make sure BPT Packaged Apps Kit is enabled. Access to BPT Packaged Apps Kit must be provided by your ALM Project Administrator using Project Customization. An administrator can enable BPT Packaged Apps Kit in ALM by clicking , then selecting **Customize > Business Process Test** and selecting the **Enable BPT Packaged Apps Kit** checkbox. For details, see *HP Application Lifecycle Management Administrator Guide*.
- Make sure Unified Functional Testing is installed on the client machine along with its SAP add-in and ALM plug-in.
- 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 packaged apps, setting up BPT Packaged Apps Kit, and on configuring user permissions, see the *HP Application Lifecycle Management Administrator Guide*.

Note: When using BPT Packaged Apps Kit, it is best to work with only once instance of ALM open. Do not open multiple instances of ALM in multiple browsers or tabs.

2. Locate the flow or business process test

To detect changes, select the relevant test set from the tree in the Test Lab module.

3. Start the run

In the Test Lab Module, in the Execution Grid tab, click **Run**.

The Automatic Runner dialog box is displayed.

4. Run the flow or business process test

In the Automatic Runner dialog box, select the **Detect Changes** checkbox and the flow or business process 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 and Business Process Tests](#)" on [page 130](#).

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 Lab module.

For task details on how to work with this report, see "[How to View and Resolve Detected](#)

[Changes" on the next page.](#)

For user interface details, see ["Change Detection Report Screen" on page 245.](#)

How to View and Resolve Detected Changes

Available for BPT Packaged Apps Kit Users

This task describes how to view and resolve changes detected in applications using the Change Detection Report screen in ALM.

This task is part of a higher-level task. For details, see ["How to Run Automated Business Process Tests and Flows" on page 229.](#)

For detailed information about change detection and resolution, see the *HP Unified Functional Testing User Guide*.

Note: ALM does not support running a learned test of an SAP Fiori application. BPT Packaged Apps Kit SAP Fiori is in technical preview.

This task includes the following steps:

- ["Prerequisites" below](#)
- ["Open the Change Detection Report screen" below](#)
- ["Update the changed components and steps" below](#)
- ["Save the changes" on the next page](#)

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*.
- Make sure Unified Functional Testing version 12.50 or later is installed on the same machine.

2. Open the Change Detection Report screen

For access details, see ["Change Detection Report Screen" on page 245.](#)

3. Update the changed components and steps

Select the element in the changes tree for which you want to view changes.

In the Change Detection Report, ALM displays a number of things:

- Run results for the test or flow, for each component, and for each step in the test.
- Changes in the application for each component
- Screenshots of the different versions of the application

Using the report, you can update your components and steps automatically:

- a. In the component tree, select the component for which you want to resolve changes.
You can see which components in which you need to resolve changes with an exclamation icon in the Changes column of the components tree.

Tip: If you want to see only the components needing changes, in the Changes column, click the down arrow and select the **Open Changes** radio button.

- b. In the right pane, view the details about the needed changes.
- c. If you want to accept the proposed changes, in the lower right corner of the pane, click the **Apply Changes** button. ALM applies the changes as suggested.
In addition, the selected component's report row is updated to show that you have resolved the changes.
If steps in your component require updates due the application change, ALM also updates the Change Detection Report.
- d. In the right pane, select the checkboxes for the steps that require an update.
- e. In the lower right corner of the pane, click **Update Steps**. ALM automatically updates the steps in your components in the background.

Note: If you want to apply the changes in the components for the current test only, you should clear the **Update changes will affect only current test** checkbox. If you do not clear this option, the changes to the components are applied to all tests containing these components.

4. Save the changes

After you have updated all the necessary components, in the lower right corner of the Change Detection report, click **Save**.

Note: After saving changes to a business component, the report becomes read-only for that component.

Change Detection and Resolution User Interface

Available for BPT Packaged Apps Kit Users

This section includes:

- [Change Detection Report Screen](#)245

Change Detection Report Screen

Available for BPT Packaged Apps Kit Users

This screen enables you view results of a business process test or flow run in Change Detection mode and resolve detected changes.

<p>To access</p>	<p>From the Test Lab module:</p> <ol style="list-style-type: none"> 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. <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> <p>Note: Business process tests or flows whose last run was in Change Detection mode are indicated with a Y in the Change Detection column. Business process tests and flows that were run in Change Detection mode, and whose changes are unresolved, are indicated with a delta symbol Δ in the lower right corner of the business process test icon or flow icon. A business process test or flow indicates unresolved changes if it contains any flows or components with unresolved changes. Business process 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 business process test or flow will change its status only when all its flows or components have had their changes resolved.</p> </div> <ol style="list-style-type: none"> 4. In the Last Run Results pane, in the Status column, click the link for the business process test.
<p>Relevant tasks</p>	<ul style="list-style-type: none"> • "How to Run Tests in Change Detection Mode" on page 241 • "How to View and Resolve Detected Changes" on page 243
<p>See also</p>	<ul style="list-style-type: none"> • "Change Detection and Resolution Overview" on page 240

Considerations

Here are some considerations to keep in mind when resolving changes:

- If a business process 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 or business process test, and changes in the duplicate were resolved.
- The component is checked out.
- The component was removed from the flow or business process test.
- 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.

Viewing the Report

The main parts of this report show you:

Area	Description
Header	General information about the business process test or flow that ran such as its name, owner, run date, duration, run status, and more. If changes were detected, the run status indicates " changes detected. "
Tree	A tree of all the components learned. Select one that indicates it has changes.
Changes	Tab that lists changed objects.
Used by	Tab that lists the business process test or flow that has changes.
Screenshot	Tab that shows snapshots of the changes.

When finished viewing and resolving, click **Save** to close the report.

Part 6: Appendices

Appendix A: BPT Packaged Apps Kit App-specific Information

Available for BPT Packaged Apps Kit Users

This appendix includes:

- [BPT Packaged Apps Kit App-specific Information Overview](#) 250
- [How to Work with BPT Packaged Apps Kit with Support for SAP Apps](#) 250
- [How BPT Packaged Apps Kit Creates Components when Learning SAP GUI Flows](#) 251
- [How BPT Packaged Apps Kit Creates Components when Learning SAP Fiori Flows](#) 253
- [BPT Packaged Apps Kit App-specific User Interface](#) 253

BPT Packaged Apps Kit App-specific Information

Overview

Available for BPT Packaged Apps Kit Users

This provides app-specific user information for BPT Packaged Apps Kit.

Supported Apps

In this version, BPT Packaged Apps Kit supports SAP GUI and Fiori.

Note: SAP Fiori is a technical preview. Learning flows and business process tests is supported; Change detection is not yet supported.

Additional apps will be supported in future versions.

How to Work with BPT Packaged Apps Kit with Support for SAP Apps

Available for BPT Packaged Apps Kit Users

The following steps describes how to work with BPT Packaged Apps Kit if your supported packaged apps include SAP apps.

- ["Prerequisites" below](#)
- ["Configure UFT to work with BPT Packaged Apps Kit features with support for SAP apps" below](#)
- ["Understand how BPT Packaged Apps Kit creates components when learning SAP flows" on the next page](#)

This task is part of a higher-level task. For details, see ["Working with BPT Packaged Apps Kit" on page 28](#).

1. Prerequisites

- Make sure that the SAP add-in for UFT is installed.

2. Configure UFT to work with BPT Packaged Apps Kit features with support for SAP apps

- a. Open UFT and add a GUI test to a solution.
- 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 process.
 - The **Record only the selected value when using a Possible Entries (F4) list** check box is checked.
- Under **General > SAP GUI for Windows > Run settings** ensure that the **Session cleanup** check box is not selected.

3. Understand how BPT Packaged Apps Kit creates components when learning SAP flows

Familiarize yourself with how BPT Packaged Apps Kit creates components for SAP GUI and Fiori. For reference details, see:

- ["How BPT Packaged Apps Kit Creates Components when Learning SAP GUI Flows" below](#)
- ["How BPT Packaged Apps Kit Creates Components when Learning SAP Fiori Flows" on page 253](#)

Note: SAP Fiori is a technical preview. Learning flows and business process tests is supported; Change detection is not yet supported.

How BPT Packaged Apps Kit Creates Components when Learning SAP GUI Flows

Available for BPT Packaged Apps Kit Users

When BPT Packaged Apps Kit learns a flow or a business process test, it creates a new business component for each screen or sub-screen through which you navigate in your SAP GUI app. Each screen or sub-screen 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_Initial_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 or business process test 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 process are automatically set as flow parameters. The value input during the Learn 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.

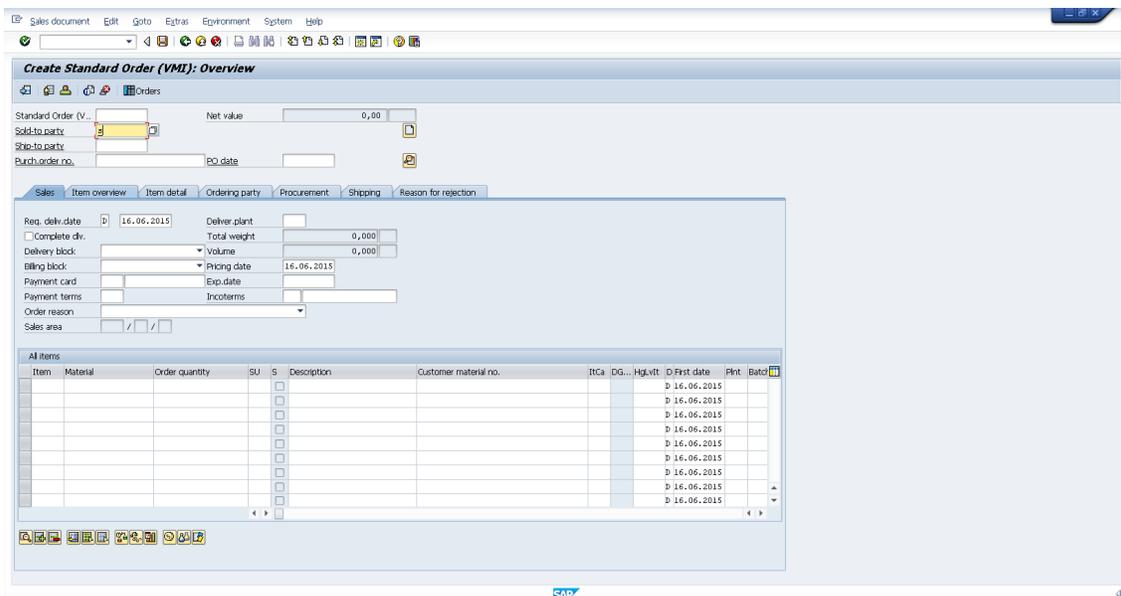
The description of a generated parameter includes the type of control and the name of the screen in which it appears, for example, **"The 'Order' control is of type 'Edit Box'. The control is a part of the 'Display Sales Order: Initial Screen' screen."**

If you add data to a table in your SAP app 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 \(for BPT Packaged Apps Kit users\)" on page 176](#).

Note:

- During the Learn 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 process, you must enable UFT to create table parameters. In UFT, add a GUI test to the solution and select **Tools > Options > GUI Testing > SAP > General** 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 \(for BPT Packaged Apps Kit users\)" on page 176](#).

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 or business process test, 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.

How BPT Packaged Apps Kit Creates Components when Learning SAP Fiori Flows

Available for BPT Packaged Apps Kit Users

Note: SAP Fiori is a technical preview. Learning flows and business process tests is supported; Change detection is not yet supported.

When BPT Packaged Apps Kit learns a business process test or flow based on SAP Fiori technology, it creates a new business component according to the following guidelines:

- The name of the component is the web page name.
- The component ID is based on web page's URL.
- Business components are created based on logic, not on a screen-by-screen basis. You can think of the components as short business flows.
- As usual, a component does not represent a single screen (a screen shot, however, is taken the first time this component is referenced).

Limitations for SAP Fiori

When learning a Fiori app, or running a learned test or flow for a Fiori app, open only one browser and an additional tab in the browser (in addition to the app).

BPT Packaged Apps Kit App-specific User Interface

Available for BPT Packaged Apps Kit Users

This section includes:

- [An Example of a SAP-specific Change Detection Report](#)254

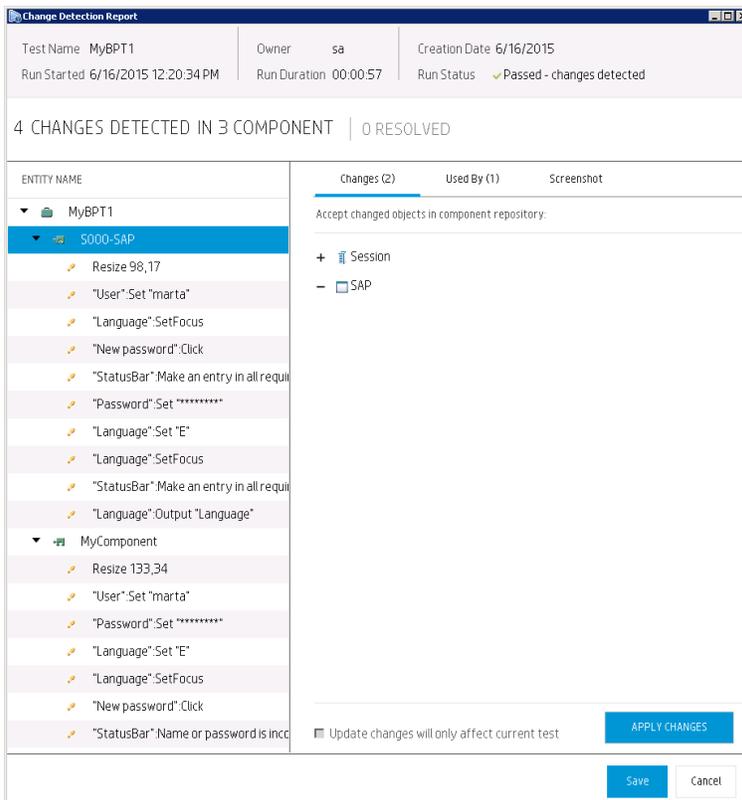
An Example of a SAP-specific Change Detection Report

Available for BPT Packaged Apps Kit Users

This report enables you to view the results of a test run in Change Detection mode in an SAP app. Here is one report, with several of its tabs displayed.

To access	For access details, see "Change Detection Report Screen" on page 245.
Important information	Note the Resolved column in the Filter area, which indicates components with detected changes that have not yet been resolved.
Relevant tasks	<ul style="list-style-type: none"> • "How to Run Tests in Change Detection Mode" on page 241 • "How to View and Resolve Detected Changes" on page 243
See also	<ul style="list-style-type: none"> • "Learn Process Overview " on page 130 • "Change Detection and Resolution Overview" on page 240

Changes tab



Used By tab

Change Detection Report

Test Name: MyBPT1 Owner: sa Creation Date: 6/16/2015
 Run Started: 6/16/2015 12:20:34 PM Run Duration: 00:00:57 Run Status: ✓ Passed - changes detected

4 CHANGES DETECTED IN 3 COMPONENT | 0 RESOLVED

ENTITY NAME	Changes (2)	Used By (1)	Screenshot										
MyBPT1		<table border="1"> <thead> <tr> <th>Name</th> <th>Path</th> <th>Status</th> <th>Owner</th> <th>Creation date</th> </tr> </thead> <tbody> <tr> <td>MyBPT1</td> <td>Subject\MyTes...</td> <td>Ready</td> <td>sa</td> <td>6/16/2015</td> </tr> </tbody> </table>	Name	Path	Status	Owner	Creation date	MyBPT1	Subject\MyTes...	Ready	sa	6/16/2015	
Name	Path	Status	Owner	Creation date									
MyBPT1	Subject\MyTes...	Ready	sa	6/16/2015									

Save Cancel

Screenshot tab

The screenshot shows a 'Change Detection Report' window. At the top, it displays test details: Test Name 'MyBPT1', Owner 'sa', Creation Date '6/16/2015', Run Started '6/16/2015 12:20:34 PM', Run Duration '00:00:57', and Run Status 'Passed - changes detected'. Below this, it states '4 CHANGES DETECTED IN 3 COMPONENT | 0 RESOLVED'. The main area is divided into three tabs: 'Changes (2)', 'Used By (1)', and 'Screenshot'. The 'Screenshot' tab is active, showing two side-by-side windows: 'Screen before change' and 'Screen after change'. The 'Screen before change' window shows a screenshot of a web application with a form. The 'Screen after change' window is greyed out and contains the text 'The screenshot cannot be displayed.' At the bottom right of the window, there are 'Save' and 'Cancel' buttons.

Send Us Feedback



Can we make this User Guide better?

Tell us how: SW-Doc@hp.com

