

Rational Integration Tester



Reference Guide for Software AG webMethods

Version 8.0.0



Note

Before using this information and the product it supports, read the information in “Notices” on page 43.

This edition applies to version 8.0.0 of Rational Integration Tester and to all subsequent releases and modifications until otherwise indicated in new editions.

© Copyright IBM Corporation 2001, 2012.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

About this Publication	v
Intended Audience	vi
Scope	vi
Typographical Conventions	vi
Contacting IBM Support	vi
Requirements	1
webMethods Versions	2
webMethods Libraries	3
Adding the GH Processor JAR File	5
Installing the Integration Server Adapter Package	8
Managing webMethods Recorders	10
Network Settings	11
webMethods Broker Transport	12
Creating the webMethods Broker Transport	13
Configuring the webMethods Broker Transport	14
Publishing and Capturing Broker Documents	18
Using the webMethods Broker Transport	19
Configuring Messaging Actions in Rational Integration Tester Tests	20
Integration Server Transport	24
Creating the Integration Server Transport	25
Configuring the Integration Server Transport	32

Testing Integration Server Services.....	36
Overview	37
Test Actions Using the Integration Server Transport.....	38
Troubleshooting.....	40
Error When Synchronizing Packages	41
Errors When Recording and Fields Discarded	41
Glossary	42
Notices	43
Trademarks and service marks	46

About this Publication

Contents

Intended Audience

Scope

Typographical Conventions

Contacting IBM Support

This guide describes how to configure and run IBM® Rational® Integration Tester with the Software AG webMethods plugin, which provides support for connectivity to Software AG's webMethods Broker and Integration Server resources.

Intended Audience

This document intended to be read by those with a fair understanding and exposure to the concepts involved in both testing and development and in enterprise integration.

Scope

This document is concerned only with the configuration and use of IBM Rational Integration Tester with Software AG's webMethods Broker and Integration Server technologies. For more information about webMethods technologies, please refer to the relevant webMethods documentation.

Typographical Conventions

The following typographical conventions are observed throughout this document:

Type	Usage
Constant Width	Program output, listings of code examples, file names, commands, options, configuration file parameters, and literal programming elements in running text.
<i>Italic</i>	Document title names in statements that refer you to other documents. Also used to highlight concepts when first introduced.
Bold	Menu items in graphical user interface windows (such as Microsoft Windows-based or UNIX X Window applications) from which you select options or execute macros and functions. Submenus and options of a menu item are indicated with a "greater than" sign, such as Menu > Submenu or Menu > Option .

Contacting IBM Support

To contact IBM Support, see: www.ibm.com/contact/us/en/

Requirements

Contents

[webMethods Versions](#)

[webMethods Libraries](#)

[Adding the GH Processor JAR File](#)

[Installing the Integration Server
Adapter Package](#)

[Managing webMethods Recorders](#)

[Network Settings](#)

This chapter describes software and system requirements for using the webMethods plugin for Rational Integration Tester.

1.1 webMethods Versions

The following sections provide details about the versions of webMethods Broker and Integration Server that are supported in Rational Integration Tester.

1.1.1 webMethods Broker

To use Rational Integration Tester with webMethods Broker, one or more of the following should be installed and configured properly:

- webMethods Broker 6.1
- webMethods Broker 6.5
- webMethods Broker 7.1
- webMethods Broker 8.0
- webMethods Broker 8.2

1.1.2 webMethods Integration Server

To use Rational Integration Tester with webMethods Integration Server, one or more of the following should be installed and configured properly:

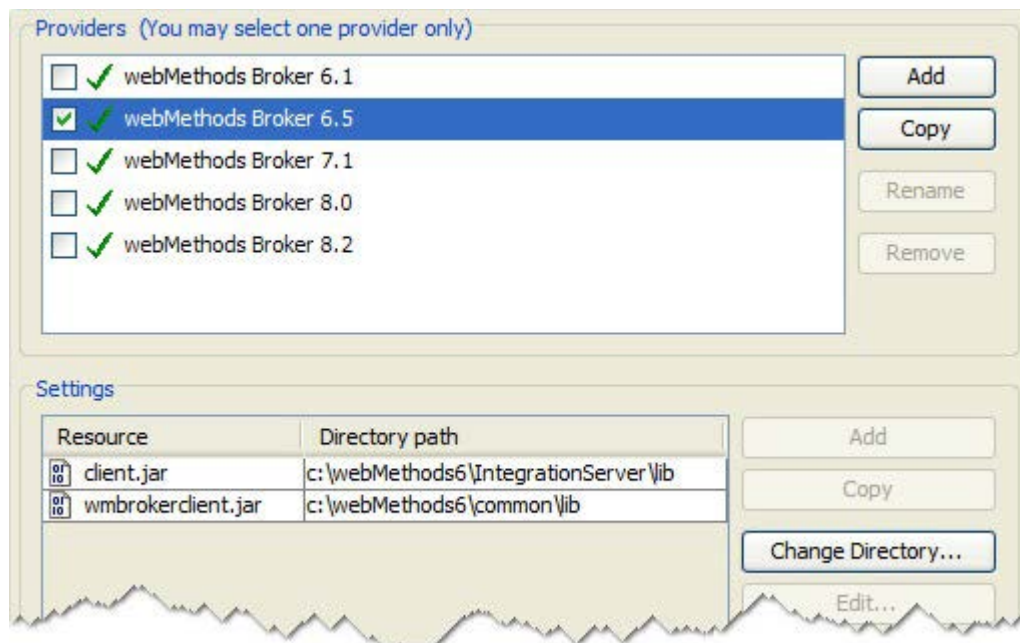
- webMethods Integration Server 6.1
- webMethods Integration Server 6.5
- webMethods Integration Server 7.1
- webMethods Integration Server 8.0
- webMethods Integration Server 8.2

1.2 webMethods Libraries

Depending on the version of webMethods Broker or Integration Server in use, specific product libraries are required. The following sections describe the libraries that are required along with their default location.

1.2.1 webMethods Broker Libraries

Broker Version	Library	Default Location
6.1	client.jar client61.jar	C:\webMethods6\common\lib
6.5	client.jar wmbrokerclient.jar	C:\webMethods6\common\lib
7.1, 8.0, 8.2	wm-brokerclient.jar wm-g11nutils.jar wm-isclient.jar	C:\webMethods7\common\lib C:\webMethods8\common\lib C:\SoftwareAG\

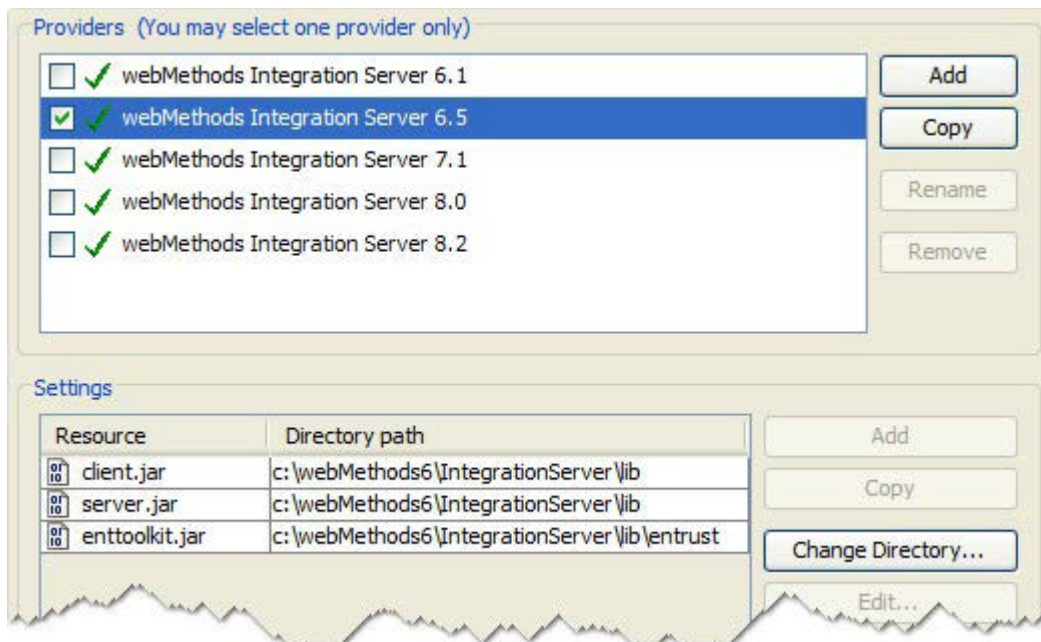


NOTE: Default locations for the specified libraries can be modified using the Library Manager. For more information, refer to *IBM Rational Integration Tester Installation Guide*.

1.2.2 webMethods Integration Server Libraries

IS Version	Library	Default Location
6.1	client.jar	C:\webMethods6\IntegrationServer\lib
	server.jar	C:\webMethods6\IntegrationServer\lib
6.5	entbase.jar	C:\webMethods6\IntegrationServer\lib\enttrust
	entmisc.jar	C:\webMethods6\IntegrationServer\lib\enttrust
	entssl.jar	C:\webMethods6\IntegrationServer\lib\enttrust
7.1, 8.0, 8.2	client.jar	C:\webMethods6\IntegrationServer\lib
	server.jar	C:\webMethods6\IntegrationServer\lib
	enttoolkit.jar	C:\webMethods6\IntegrationServer\lib\enttrust
	mail.jar	C:\webMethods6\IntegrationServer\lib\enttrust

By default, <wm-root> is C:\webMethods7, C:\webMethods8, or C:\SoftwareAG, respectively.



NOTE: Default locations for the specified libraries can be modified using the Library Manager. For more information, refer to *IBM Rational Integration Tester Installation Guide*.

1.3 Adding the GH Processor JAR File

Rational Integration Tester 5.2.11.30 introduced a new mechanism for integrating a Rational Integration Tester installation with a webMethods Integration Server.

Therefore, you cannot use versions of Rational Integration Tester earlier than 5.2.11.30 against a webMethods Integration Server installation configured with the GH Processor JAR file from 5.2.11.30 or later.

NOTE: The correct version of the GH Processor JAR file must be configured on **each** webMethods Integration Server where you intend to run Rational Integration Tester tests.

If you are upgrading from a version of Rational Integration Tester earlier than 5.2.11.30 to version 5.2.11.30 or later, refer to [Rational Integration Tester 5.2.11.30 or Later](#).

If you are not sure which instructions you should be following, please contact IBM Support.

1.3.1 Rational Integration Tester 5.2.11.29 or Earlier

The GH Processor JAR file must be configured on each webMethods Integration Server where you intend to run Rational Integration Tester tests.

NOTE: Integration Server versions 7.1, 8.0, and 8.2 require a different JAR file than earlier versions. For 7.1, 8.0, and 8.2 installations, use the `gh-processor-71-82.jar` file. For 6.1 and 6.5 installations, use the `gh-processor-61-70.jar` file.

NOTE: If the JAR file for your webMethods version has changed, you must first delete the old file from `<wM_HOME>\IntegrationServer\lib\jars` before copying over the updated file.

To add the GH Processor JAR file:

1. Copy the appropriate file from `<Rational Integration Tester Installation Directory>\tools\SoftwareAG\webMethods` to `<wM_HOME>\IntegrationServer\lib\jars` under the webMethods installation directory.
2. Shut down the Integration Server and open the following file for editing:

`<wM_HOME>\IntegrationServer\config\invokemanager.cnf`

-
3. Directly above the line ending with `</array>`, add the following line:

```
<value>com.ghc.a3.wmis.invoke.GHProcessor</value>
```
 4. Save and close the file, then restart the Integration Server.
 5. View `<wM_HOME>\IntegrationServer\config\invokemanager.cnf` and make sure the changes you made have not been overwritten.
 6. Install the adapter package described in [Installing the Integration Server Adapter Package](#).

1.3.2 Rational Integration Tester 5.2.11.30 or Later

The GH Processor JAR file must be configured on each webMethods Integration Server where you intend to run Rational Integration Tester tests.

NOTE: Integration Server versions 7.1, 8.0, and 8.2 require a different JAR file than earlier versions. For 7.1, 8.0, and 8.2 installations, use the `gh-processor-71-82.jar` file. For 6.1 and 6.5 installations, use the `gh-processor-61-70.jar` file.

NOTE: If the JAR file for your webMethods version has changed, you must first delete the old file from `<wM_HOME>\IntegrationServer\lib\jars` before copying over the updated file. This includes upgrading from version of Rational Integration Tester earlier than 5.2.11.30.

To add the GH Processor JAR file:

1. Copy the `protobuf-javame-1.1.1.jar` file and the `gh-processor-61-70.jar` file or the `gh-processor-71-82.jar` (whichever is applicable) from *<Rational Integration Tester Installation Directory>\tools\SoftwareAG\webMethods to* `<wM_HOME>\IntegrationServer\lib\jars` under the webMethods installation directory.
2. Shut down webMethods Integration Server and open the following file for editing:

```
<wM_HOME>\IntegrationServer\config\invokemanager.cnf
```
3. Directly above the line ending with `</array>`, add the following line (unless it already exists):

```
<value>com.greenhat.wm.pipeline.PipelineHandler</value>
```

-
4. If you are upgrading from a version of Rational Integration Tester earlier than 5.2.11.30, you must delete the following line:

```
<value>com.ghc.a3.wmis.invoke.GHProcessor</value>
```

5. Save and close the file.
6. Restart the webMethods Integration Server.
7. View `<WM_HOME>\IntegrationServer\config\invokemanager.cnf` and make sure the changes you made have not been overwritten.
8. Install the adapter package described in [Installing the Integration Server Adapter Package](#).

1.4 Installing the Integration Server Adapter Package

NOTE: The Integration Server Adapter Package was updated in the Rational Integration Tester 5.2.11 release. If you have upgraded from an earlier release, all webMethods Integration Servers must be updated and the IBM package will have to be imported again.

NOTE: If you have upgraded from a version of Rational Integration Tester earlier than 5.2.11.30 to 5.2.11.30 or later, you must install the Integration Server adapter package (or reinstall it if you have previously installed it) because the package definition changed at Rational Integration Tester 5.2.11.30.

The webMethods Integration Server adapter package must be installed on every Integration Server on which you intend to run Rational Integration Tester tests. The adapter package file is called `GreenHat61_82.zip`.

NOTE: If the adapter package filename for your version has changed, the old package must be removed from the inbound directory (see step 1) and unloaded by means of the webMethods administrator before proceeding.

To install the Integration Server adapter package:

1. Copy the adapter package from *<Rational Integration Tester Installation Directory>\tools\SoftwareAG\webMethods to <wM_HOME>\IntegrationServer\replicate\inbound* (under the webMethods installation directory).
2. Ensure the webMethods Integration Server is running and install the inbound package with the Integration Server Administrator (for more information, refer to *Integration Server Administrator's Guide*).

NOTE: If the package does not auto-install, create an `IBM` subdirectory under *<wM_HOME>\IntegrationServer\replicate\inbound* on the Integration Server. Extract the package into that directory and you should be able to load the package from the new directory.

3. In webMethods Developer, select **File > Sync Document Types > All Out-of-Sync**.

The broker documents that are out of sync will be listed.

-
4. Select all of the **IBM** items and synchronize the broker documents.

1.5 Managing webMethods Recorders

To configure the integration internally, modify the following property in the `GreenHat.properties` file (by default, this file is located in `<IS Install Directory>/packages/IBM/config/`):

- `publishLocal`

This property controls how documents representing captured messages are managed.

If set to `true` (the default setting), documents are managed locally within webMethods Integration Server before being sent to Rational Integration Tester.

If set to `false`, messages are sent through webMethods Broker so that they can be managed explicitly by end-users.

1.6 Network Settings

Rational Integration Tester application preferences in the Library Manager utility include two settings that are specific to webMethods Integrations Server – WMIS Server Port and WMIS Server Bind Address. These settings may be important in your specific webMethods deployment:

- **WMIS Server Port** is the port on which Rational Integration Tester should listen for webMethods Integration Server requests (used when firewall configuration is necessary).
- **WMIS Server Bind Address** is the host name or IP address to which Rational Integration Tester should be bound for webMethods Integration Server requests (for use on systems with multiple network interfaces, or to prevent Rational Integration Tester from binding to a network address that may not be accessible to remote systems).

For more information about Library Manager, refer to *Rational Integration Tester Installation Guide*.

webMethods Broker Transport

Contents

**[Creating the webMethods Broker
Transport](#)**

**[Configuring the webMethods
Broker Transport](#)**

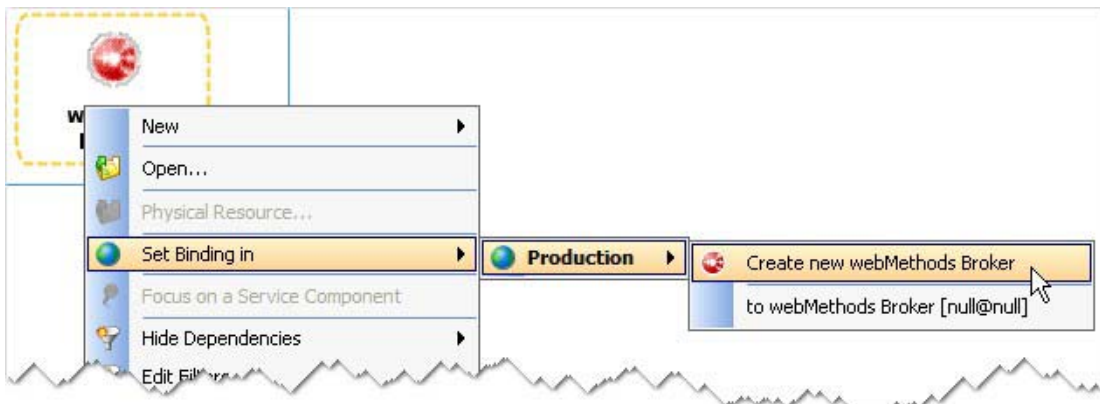
This chapter provides an overview of how to create and configure the webMethods Broker transport.

2.1 Creating the webMethods Broker Transport

The webMethods Broker transport is created when you create a physical webMethods Broker resource in Rational Integration Tester's Architecture School.

In Architecture School, you can create a new resource using either of the following methods:

- In the Logical View, right-click a webMethods Broker Domain and select the **Set Binding in > [environment] > Create new webMethods Broker** option.



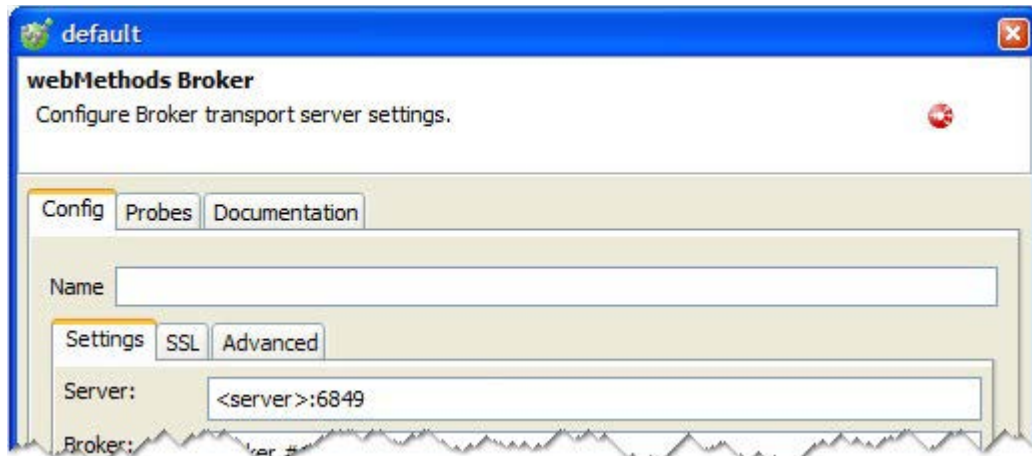
- In the Physical View, select the **Software AG > webMethods Broker** option.



Each physical webMethods Broker resource will represent a webMethods Broker transport that can be selected and configured later on.

2.2 Configuring the webMethods Broker Transport

To configure a webMethods Broker transport, double-click the appropriate webMethods Broker resource in Architecture School's Physical View.



If desired, enter a name for the transport in the **Name** field (to help identify it when multiple webMethods Broker transports are available).

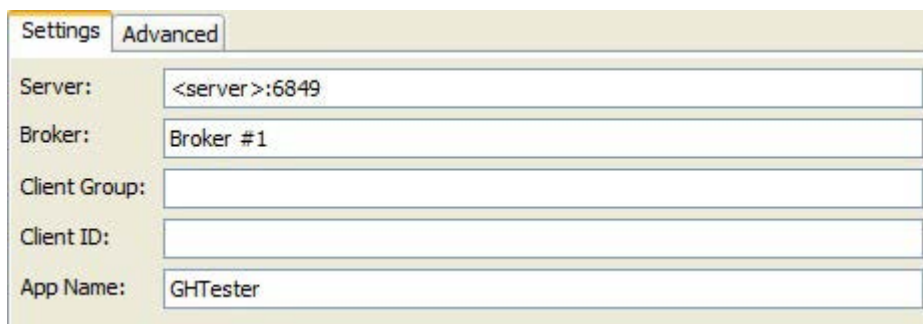
The transport settings are managed under the **Config** tab, and are broken into [Basic Settings](#) (**Settings** tab), [SSL Settings](#), and [Advanced Settings](#).

NOTE: All of the configuration fields support the use of tags, which can be entered manually or by selecting them from the context menu.

NOTE: You can test the connection parameters at any time by clicking **Test Transport**.

2.2.1 Basic Settings

The basic webMethods Broker settings are configured under the **Settings** tab.



The available configuration options are described in the following table:

Server	The server and port where webMethods Broker is running.
Broker	The name of the webMethods Broker (if left blank, “Broker” will be used). “Broker #1” is the default in webMethods.
Client Group	The name of the client group for the webMethods Broker client.
Client ID	A string containing a unique identifier for the webMethods Broker client to be used when disconnecting or reconnecting. If left blank, the broker will generate a unique client identifier.
App Name	The name of the application that will contain the webMethods Broker client. The default is “Rational Integration Tester”.

2.2.2 SSL Settings

Settings used to connect to the server using SSL are configured under the **SSL** tab.



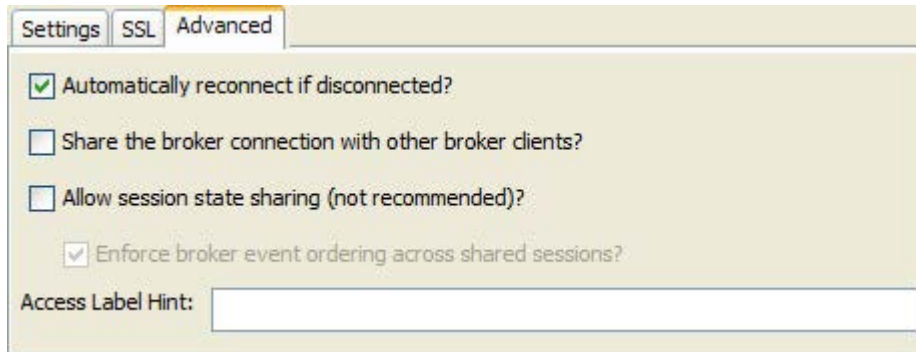
The available configuration options are described in the following table:

Use SSL	Select this check box to use SSL for the connection to the server.
Distinguished Name	This is the portion of a certificate that identifies either the owner of the certificate or the issuer of the certificate.
Trust Store	An existing Rational Integration Tester Identity store, required by webMethods to be a JKS (JAVA Key Store) file. Once the JKS is added in Rational Integration Tester as an Identity Store, it can be selected from the combo box. The trust store contains certificates that are trusted by the client.
Identity Store	The client's public key, combined with a certificate, that identifies the client to the server. This is required by webMethods to be a P12 file (Personal Information Exchange File). This does not need to be added to Rational Integration Tester, but it must be available at runtime.
Encrypt Traffic	Select this check box to encrypt the communication between Rational Integration Tester and the server. The encryption of the handshake is used for both encrypted and non-encrypted (enabled and disabled) communications. Disabling this option will provide faster communication between Rational Integration Tester and the server, but the connection will be less secure.

NOTE: SSL is not supported for webMethods Broker 6.1.

2.2.3 Advanced Settings

More advanced settings for connecting to a webMethods Broker are configured under the **Advanced** tab.



Settings SSL **Advanced**

☒ Automatically reconnect if disconnected?

☐ Share the broker connection with other broker clients?

☐ Allow session state sharing (not recommended)?

☒ Enforce broker event ordering across shared sessions?

Access Label Hint:

The available configuration options are described in the following table:

Automatically reconnect if disconnected?	Enables or disables the automatic re-connection feature for the Broker client's connection descriptor. If enabled, the Broker client associated with this descriptor will be automatically reconnected after the connection to the Broker is lost. This option is enabled by default.
Share the broker connection with other broker clients?	Enables or disables the sharing of this connection by multiple Broker clients. This option is disabled by default.
Allow session state sharing?	Enables or disables the sharing of the client state associated with this connection by multiple Broker clients. This option is disabled by default.
Enforce broker event ordering across shared sessions?	Available only if the previous option is enabled, this option sets the shared event ordering status for the Broker client's connection descriptor. If available, this option is enabled by default.
Access Label Hint	The access label hint for the Broker client's connection descriptor.

Publishing and Capturing Broker Documents

Contents

[Using the webMethods Broker Transport](#)

[Configuring Messaging Actions in Rational Integration Tester Tests](#)

This chapter provides an overview of how to use messaging actions in Rational Integration Tester to subscribe to capture broker documents and publish documents to the broker.

3.1 Using the webMethods Broker Transport

Broker events are primarily used in two ways in Rational Integration Tester. You can subscribe to a broker event (document type) to capture the broker document (data and structure).

Additionally, you can publish a captured broker document back the broker to initiate downstream processing. The data in the broker document can be manipulated for the purpose of testing different scenarios.

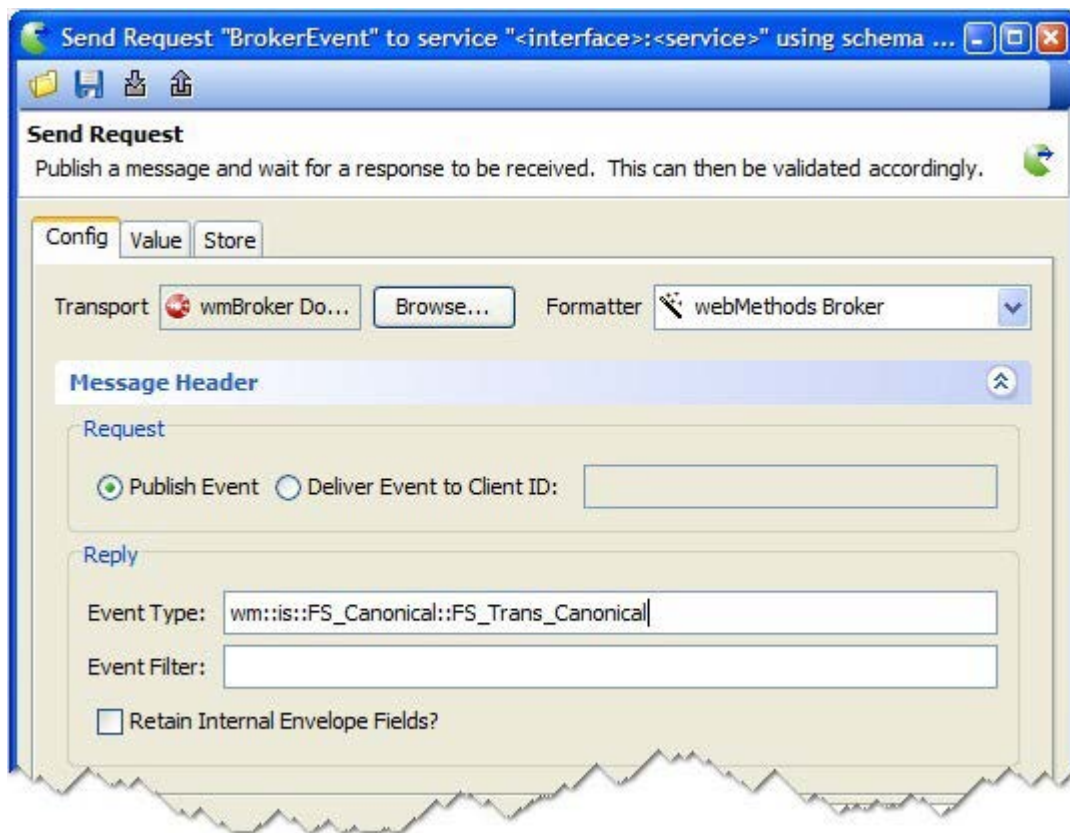
3.2 Configuring Messaging Actions in Rational Integration Tester Tests

This section describes the configuration options for the actions used in Rational Integration Tester tests for publishing and capturing (subscribing to events) broker documents. For more information about this, refer to *IBM Rational Integration Tester Reference Guide*.

3.2.1 Send Request and Publish Actions

The following information is used for the Send Request and Publish actions:

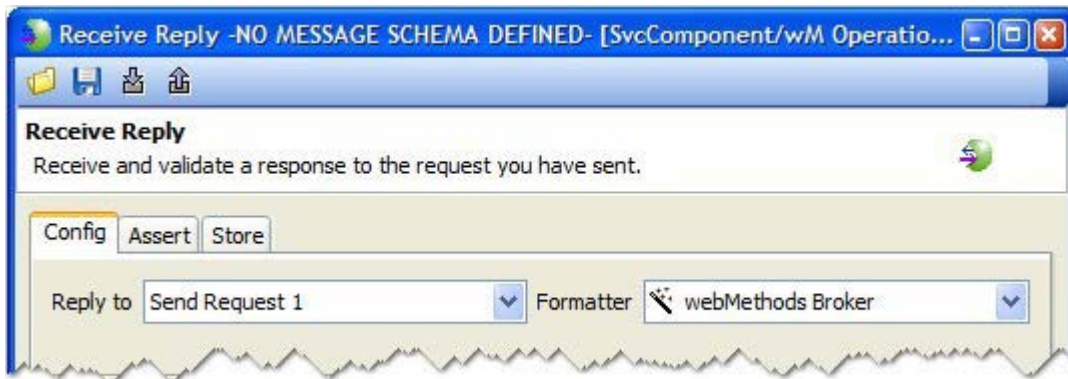
- Whether to publish or deliver the broker event. If delivering, you must specify the message and destination client ID to which it will be delivered.
- The event type name and optional event filter string (for more information about event filter strings, refer to *webMethods Broker Client Java API Programmer's Guide*).
- Whether or not to retain internal envelope fields (disabled by default).



NOTE: A message cannot be intercepted if it has been published.

3.2.2 Receive Reply Action

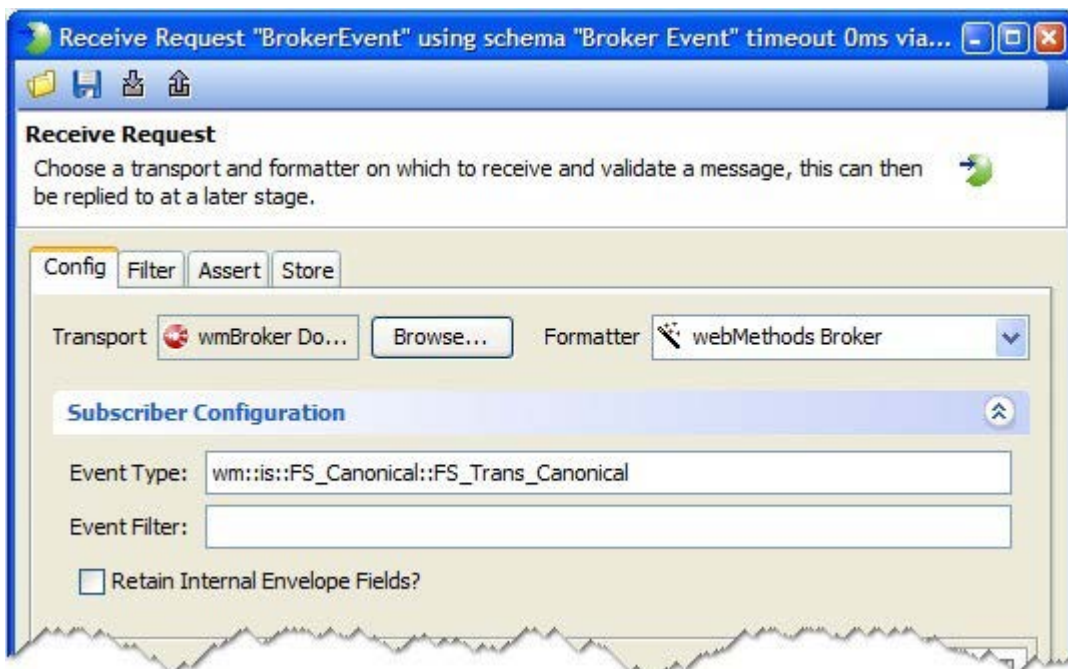
The Receive Reply is used in conjunction with a Send Request action. Therefore, the only information required is the ID of the request to which you wish to reply.



3.2.3 Receive Request

The following information is used for the Receive Request action:

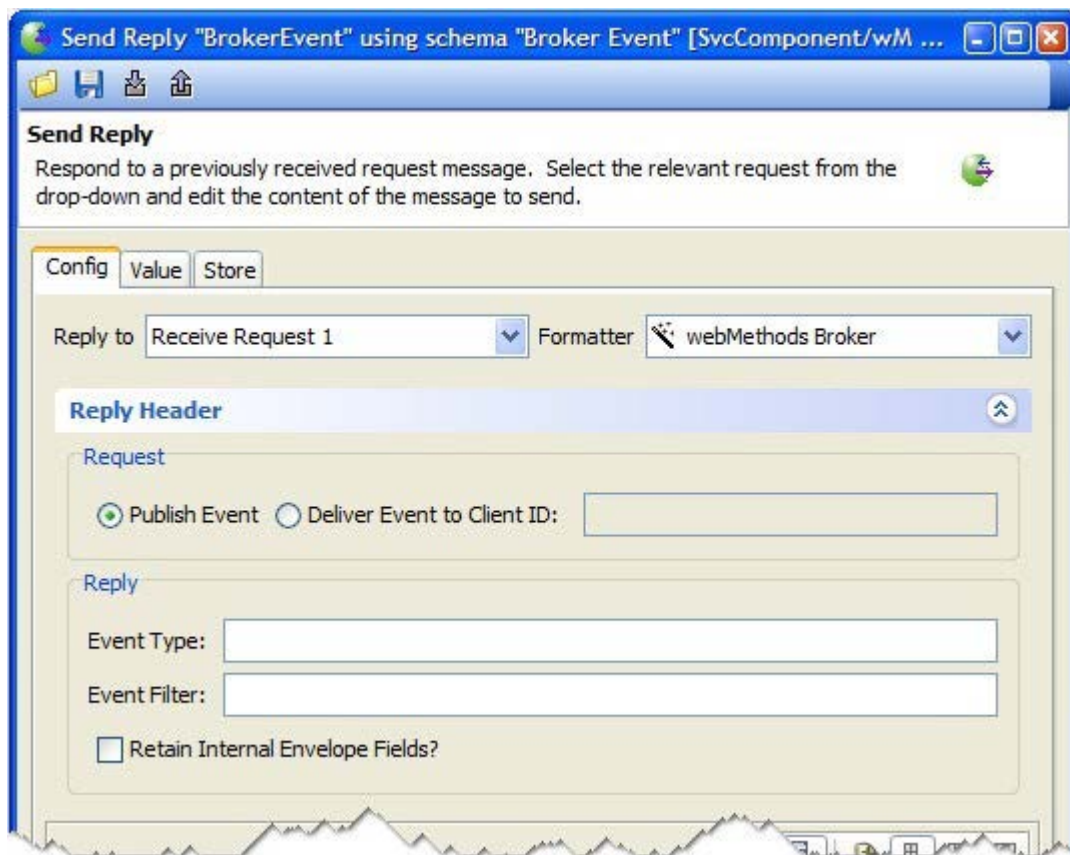
- The event type name and optional event filter string (for more information about event filter strings, refer to *webMethods Broker Client Java API Programmer's Guide*).
- Whether or not to retain internal envelope fields (disabled by default).



3.2.4 Send Reply

The Send Reply is used in conjunction with a Receive Request action. In addition to the ID of the request to which you are replying, the following information is required:

- Whether to publish or deliver the broker event. If delivering, you must specify the message and destination client ID to which it will be delivered.
- The event type name and optional event filter string (for more information about event filter strings, refer to *webMethods Broker Client Java API Programmer's Guide*).
- Whether or not to retain internal envelope fields (disabled by default).

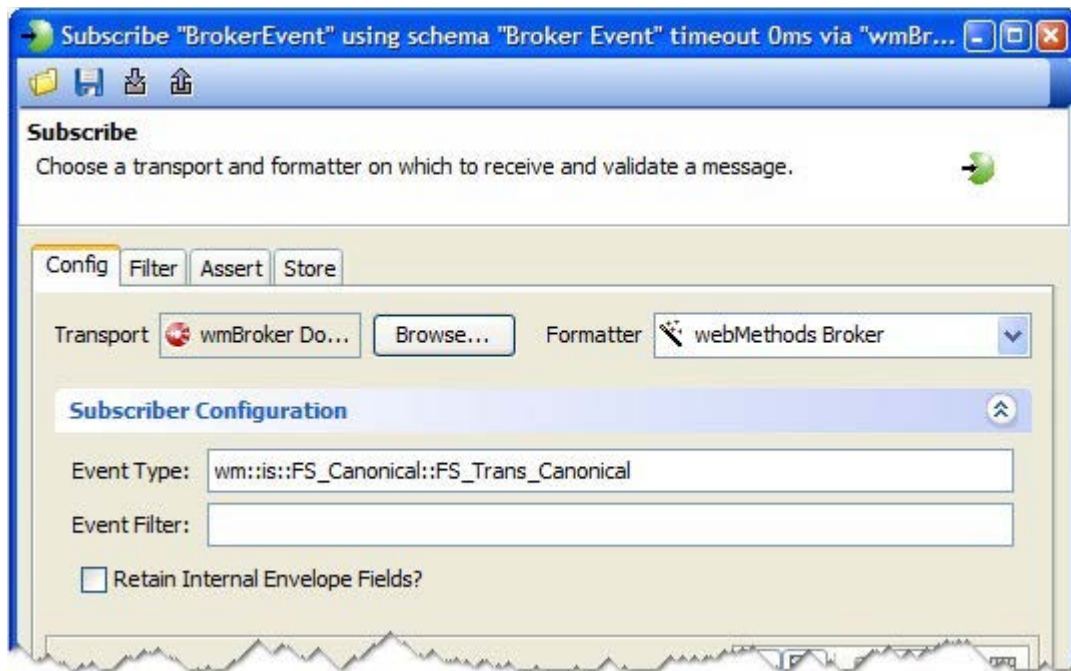


3.2.5 Subscribe

When using the Subscribe action to intercept broker events, the following information is required:

- The event type name to which to subscribe.
- An optional event filter string (for information about event filter strings, refer to *webMethods Broker Client Java API Programmer's Guide*).

- Whether or not to retain internal envelope fields (disabled by default).



Integration Server Transport

Contents

**[Creating the Integration Server
Transport](#)**

**[Configuring the Integration Server
Transport](#)**

This chapter provides an overview of how to create and configure the webMethods Integration Server transport.

4.1 Creating the Integration Server Transport

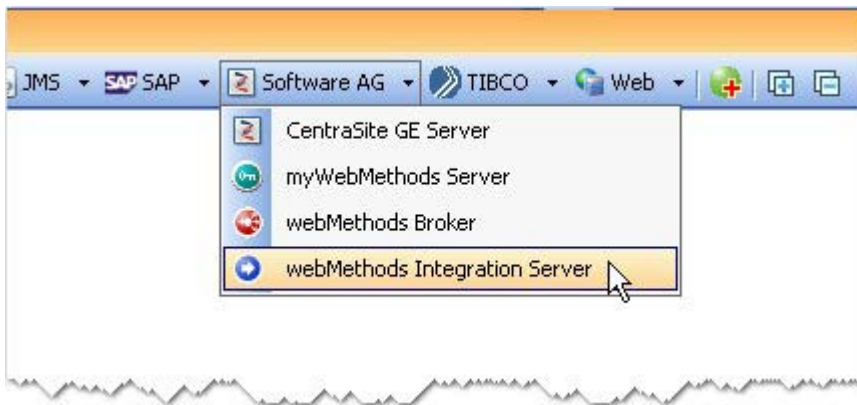
The webMethods Integration Server transport is created when you create a physical webMethods Integration Server resource in Rational Integration Tester's Architecture School.

NOTE: Before the transport can be used in a test, the physical resource must be bound to a logical webMethods Integration Server Domain, which is added as an external resource in Architecture School's Logical or Synchronisation views.

4.1.1 Creating a Physical webMethods Integration Server Resource

In Architecture School, you can create a new webMethods Integration Server resource as follows:


1. Select the Physical View in Architecture School.
2. Right-click the Physical folder and select **New > Software AG > webMethods Integration Server**, or select **webMethods Integration Server** from the **Software AG** menu.

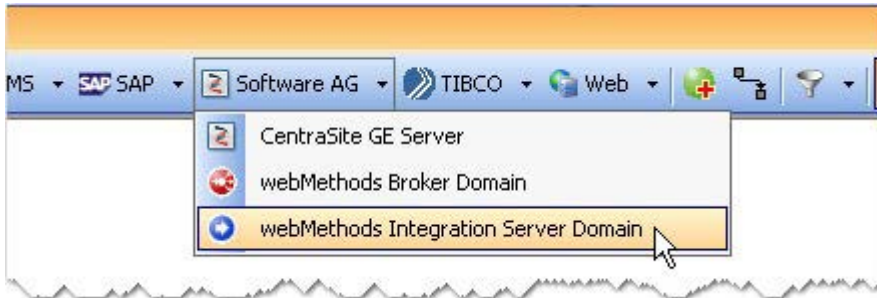


The physical webMethods Integration Server resource can be selected when adding an external webMethods Integration Server Domain (see [Adding a webMethods Integration Server Domain](#)). The details of the physical resource (that is, the transport) can be configured subsequently.

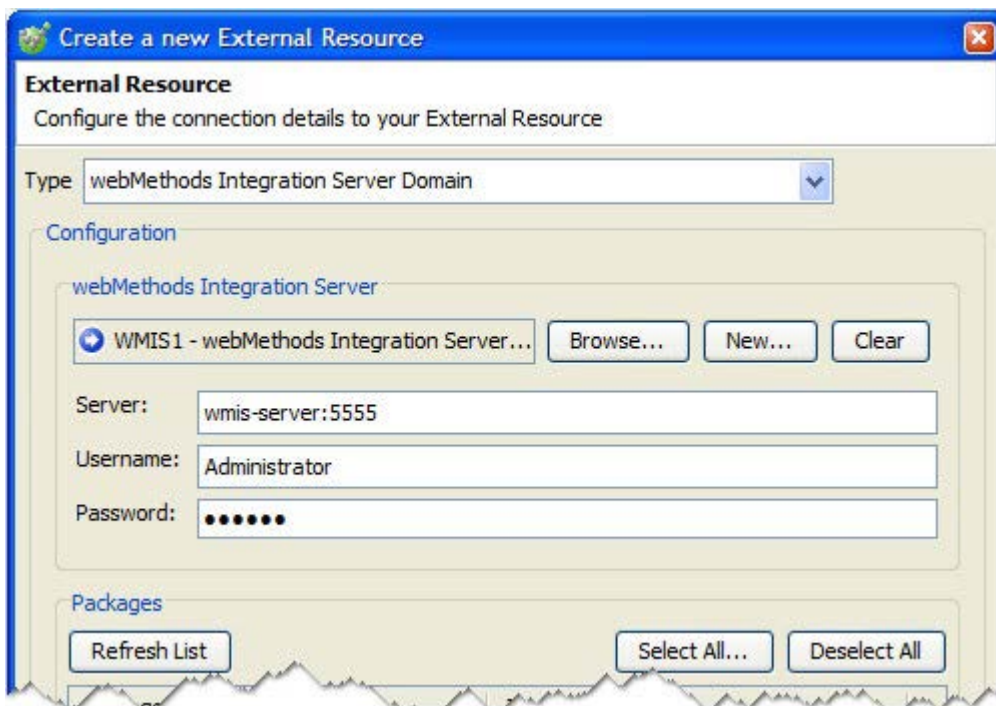
4.1.2 Adding a webMethods Integration Server Domain

Follow the steps below to add an external webMethods IS Domain to your project.

1. In the Logical or Synchronisation view, add a reference to the webMethods Integration Server Domain by clicking **Add Item** . Alternatively, you can select **webMethods Integration Server Domain** from the **Software AG** menu or from the context menu.

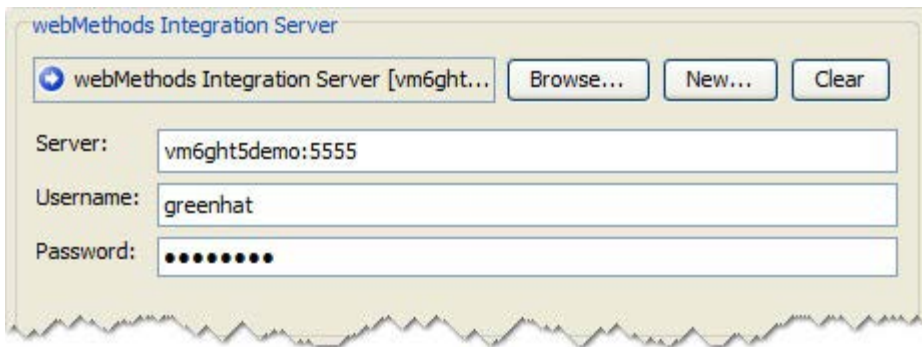


The **Create a new External Resource** wizard is displayed.



2. Ensure that **webMethods Integration Server Domain** is selected in the **Type** field.
3. Click **Browse** to select an existing Integration Server from the project, or click **New** to create a new physical resource (see [Configuring the Integration Server Transport](#)).

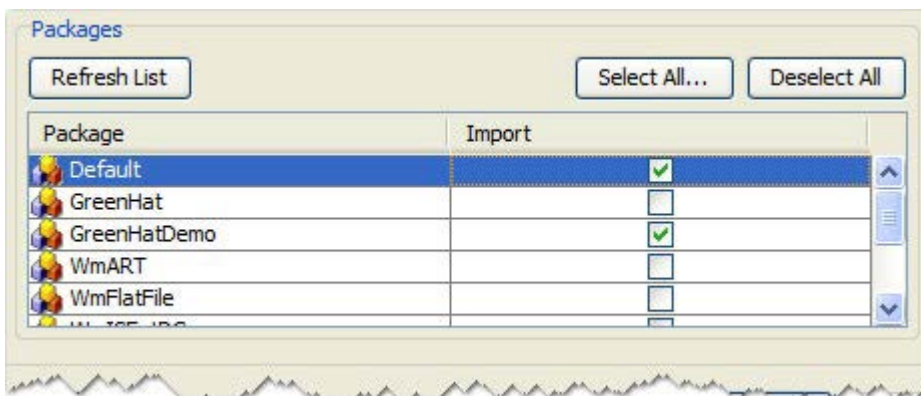
The connection details of the server (from the physical resource) are displayed.



The screenshot shows a window titled "webMethods Integration Server". At the top, there is a dropdown menu showing "webMethods Integration Server [vm6ght...]" with buttons "Browse...", "New...", and "Clear" to its right. Below this, there are three input fields: "Server:" with the value "vm6ght5demo:5555", "Username:" with the value "greenhat", and "Password:" with a masked password represented by nine dots.

4. Click **Refresh List** under **Packages** to see the list of packages available on the server.

NOTE: To more quickly find a specific package, you can click column names to sort the contents in ascending or descending order.



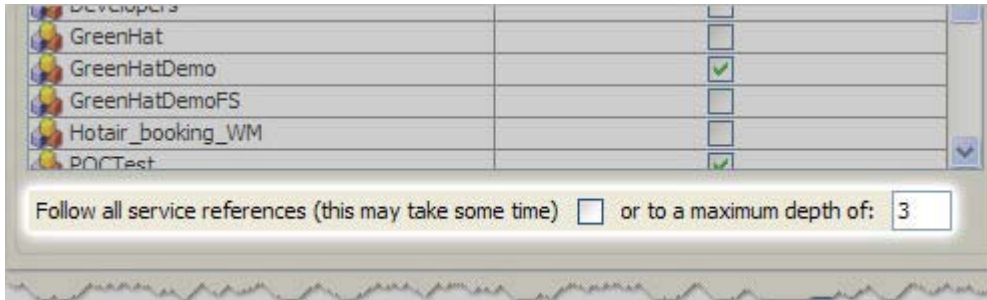
The screenshot shows a window titled "Packages". At the top, there are buttons "Refresh List", "Select All...", and "Deselect All". Below these buttons is a table with two columns: "Package" and "Import". The table contains the following data:

Package	Import
Default	<input checked="" type="checkbox"/>
GreenHat	<input type="checkbox"/>
GreenHatDemo	<input checked="" type="checkbox"/>
WmART	<input type="checkbox"/>
WmFlatFile	<input type="checkbox"/>
WmJMS	<input type="checkbox"/>

5. Select the **Import** check box for each package you want to import, and clear the check box for each package that should not be imported.

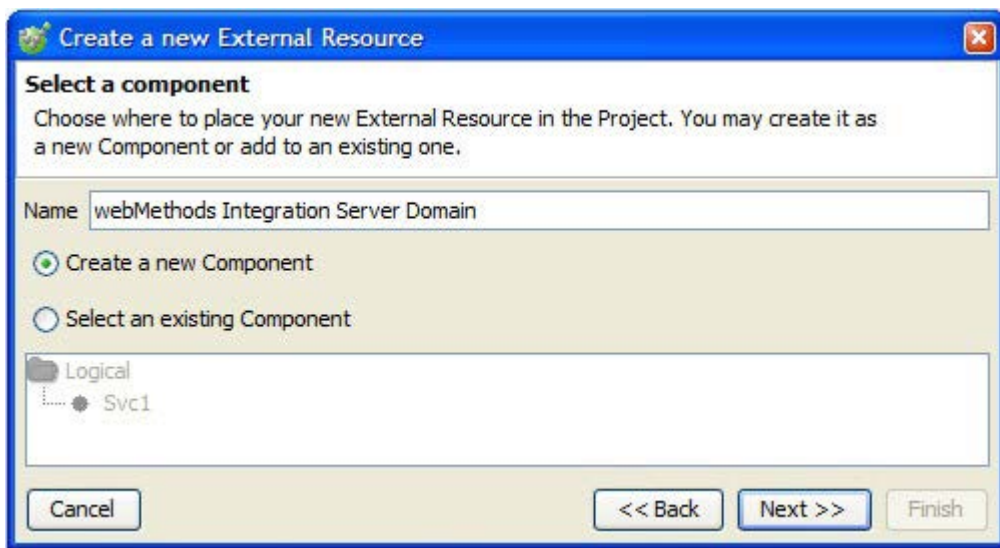
NOTE: To quickly select or deselect all packages, click **Select All** or **Deselect All**, as desired. When selecting all, you will have the option to include all built-in packages.

-
6. To limit the number of referenced services that are imported for each selected package, clear the **Follow all service references** check box and enter the maximum level of referenced services in the available field (integer values are supported).



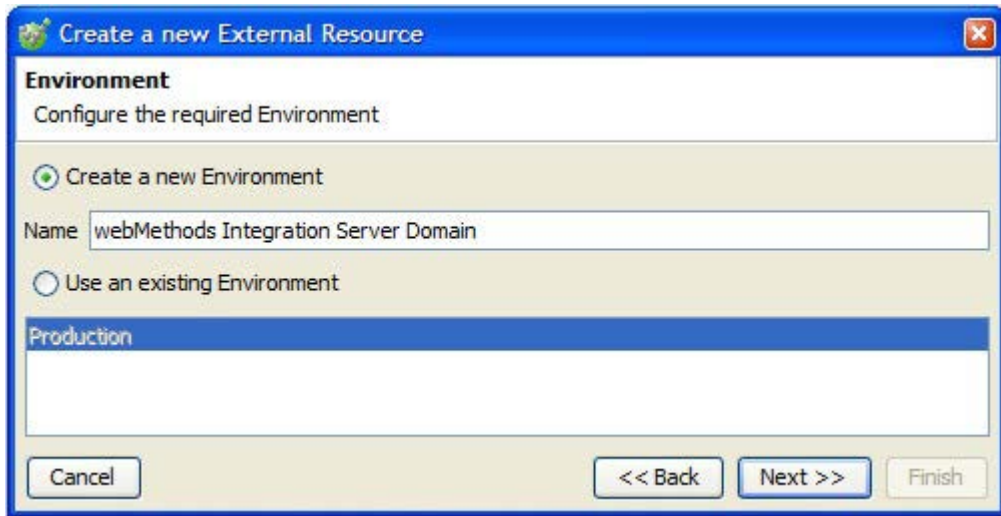
7. Click **Next** to proceed.

In the **Select a Component** screen, you can specify where to create the Integration Server artifact in your project (that is, in a new or existing Service Component).



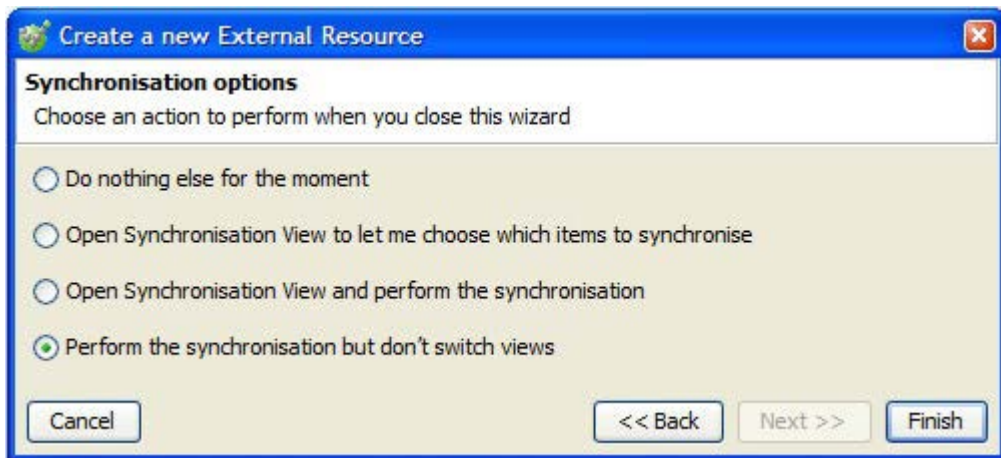
8. Select the desired option (providing a name for a new component or selecting the desired existing component), then click **Next**.

If one or more environments exist in your project, the **Environment** screen is displayed, letting you create a new environment or use an existing one for the variables and bindings that will be imported.



9. Select the desired option (providing a name for a new environment or selecting an existing environment), then click **Next**.

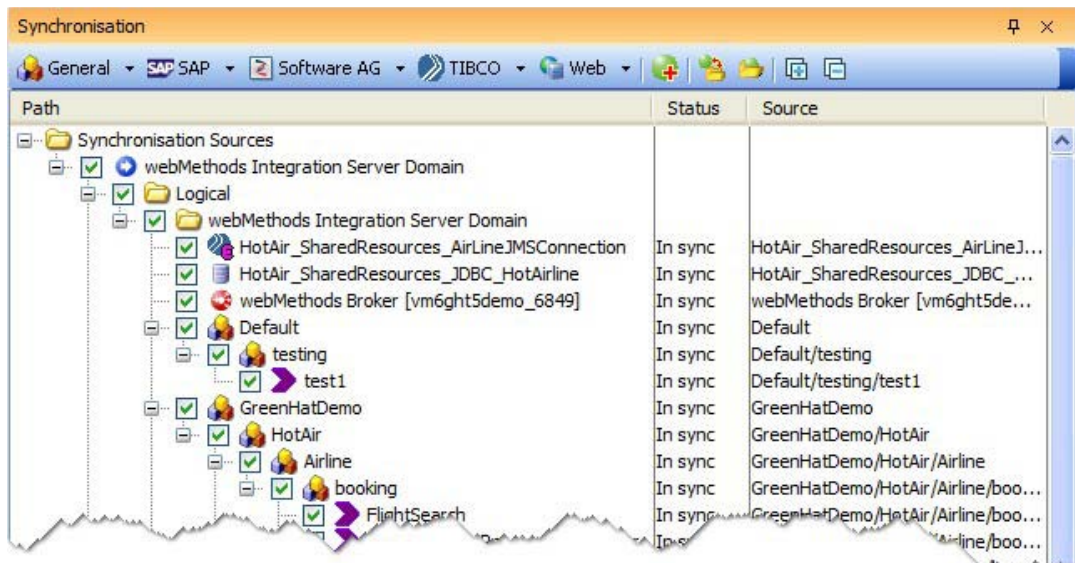
The **Synchronisation Options** screen is displayed.



10. Select the desired synchronisation option and click **Finish**.

NOTE: Depending on how many packages were selected and the contents of each package, synchronisation could take several minutes to complete.

When complete, the required artifacts will be created in the project.



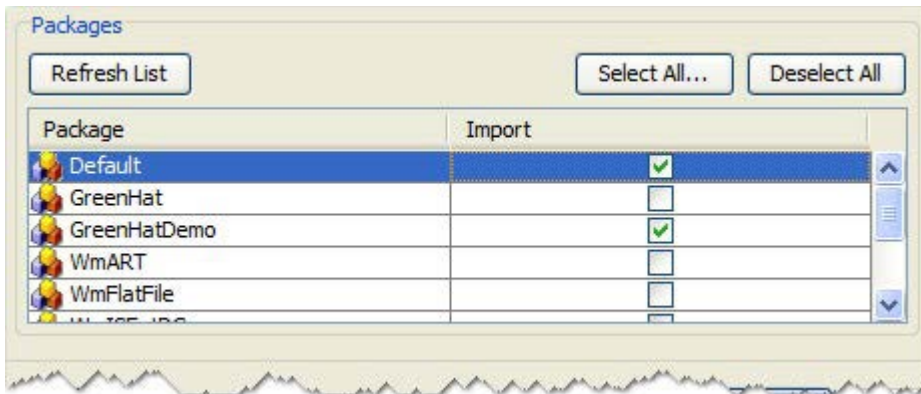
Additionally, the environment will be populated with the required variables and bindings that have been converted from the Integration Server packages.

4.1.3 Adding or Removing Synchronized Packages

After a webMethods Integration Server Domain has been added to a Rational Integration Tester project, the server packages that are synchronized can be modified by selecting/clearing them from the logical component (that is, the webMethods IS Domain in the Logical View of Architecture School).

1. Open Architecture School (**F7**) and go to the **Logical View**.
2. Double-click the webMethods Integration Server Domain component.
3. Click **Refresh List** under **Packages** to see the list of packages available on the server.

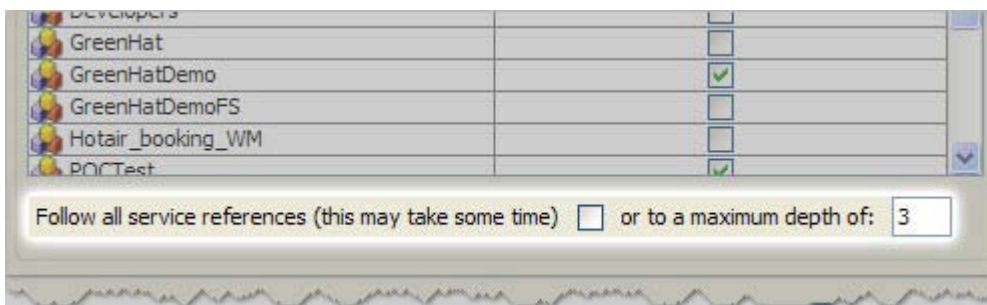
NOTE: To more quickly find a specific package, you can click column names to sort the contents in ascending or descending order.



4. Select the **Import** check box for each of the packages you want to import, and clear the check box for each package that should not be imported.

NOTE: To quickly select or deselect all packages, click **Select All** or **Deselect All**, as desired. When selecting all, you will have the option to include all built-in packages.

5. To limit the number of referenced services that are imported for each selected package, clear the **Follow all service references** check box and enter the maximum level of referenced services in the available field (integer values are supported).



6. Click **OK** when finished.
7. Synchronize your project to enable any changes you have made to take effect.

4.2 Configuring the Integration Server Transport

To configure a webMethods Integration Server transport, double-click a webMethods Integration Server resource in Architecture School's Physical View. If desired, enter a name for the transport in the **Name** field (to help identify it when multiple Integration Servers are available).

The transport settings are managed under the **Config** tab, and are broken into [Basic Settings](#) (**Settings** tab), [Advanced Settings](#), and [SSL Settings](#).

NOTE: All of the configuration fields support the use of tags, which can be entered manually or by selecting them from the context menu. Tags must be entered manually in the **Password** field (for example, `%%wm_password%%`).

NOTE: You can test the connection parameters at any time by clicking **Test Transport**.

4.2.1 Basic Settings

The basic Integration Server settings are configured under the **Settings** tab.

The screenshot shows the 'Settings' tab selected. It contains three input fields: 'Server:' with the value 'my_is_server:5555', 'Username:' with the value 'greenhat', and 'Password:' with a masked password represented by ten dots.

The available configuration options are described in the following table:

Server	The host name and port on which the Integration Server is installed and listening. The default port is 5555.
Username	The user name to use for connecting to the Integration Server. The default is Administrator .
Password	The password to use for connecting to the Integration Server. The default is manage .

4.2.2 Advanced Settings

More advanced settings for connecting to an Integration Server are configured under the **Advanced** tab.

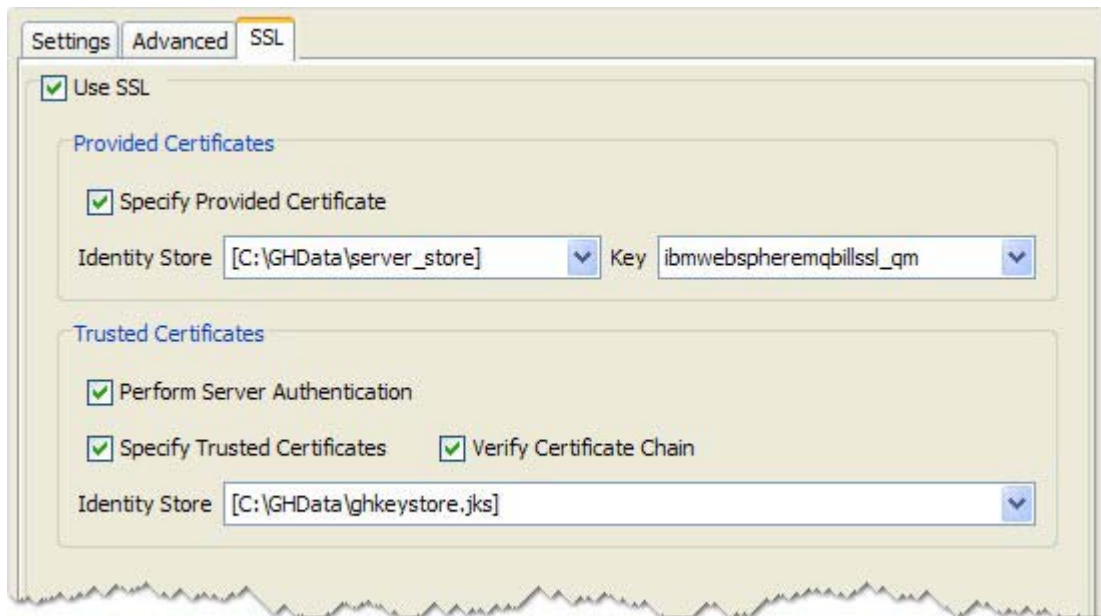
The screenshot shows the 'Advanced' tab selected. It includes a checked checkbox for 'Allow redirects within the IS server cluster?'. Below this are two empty text input fields labeled 'URI for remote web server CGI script:' and 'Retry server in case primary server is unavailable:'. A section titled 'Proxy Server' contains an unchecked checkbox for 'Use an HTTP proxy server?'. Below this are three empty text input fields for 'Server:', 'Username:', and 'Password:'.

The available configuration options are described in the following table:

Allow redirects within the IS server cluster?	This option enables or disables the redirection of requests through the connection (that is, whether or not requests can be passed to another server in the target server's cluster).
URI for remote web server CGI script	If the Integration Server is configured for it, this is the Uniform Resource Identifier (URI) of the CGI script that must be used to access the server.
Retry server in case primary server is unavailable	The host name and port of the retry server, which is used when a client connects to a server in a cluster and the requested server is down. If a retry server is specified, an attempt will automatically be made to connect the client to the retry server.
Proxy Server	<p>These options are used only if you want to configure the transport to work with an HTTP proxy server. If so, select the Use an HTTP proxy server? check box and configure the following fields:</p> <p>Server: The host name of the proxy server.</p> <p>Username: The user name to use for connecting to the proxy server.</p> <p>Password: The password to use for connecting to the proxy server.</p>

4.2.3 SSL Settings

The webMethods transport SSL settings are configured under the **SSL** tab, as shown below:



Specify the desired options under **Provided Certificates** (client settings) and **Trusted Certificates** (server settings), and select the desired Identity Store to use for each one.

Testing Integration Server Services

Contents

Overview

Test Actions Using the Integration Server Transport

This chapter provides an overview of how to use messaging actions in Rational Integration Tester to trigger tests or test Integration Server services.

5.1 Overview

There are several ways that Integration Server publishers and subscribers can be used in Rational Integration Tester.

- Publishers can be used to trigger end-to-end tests, and subscribers can be used to track the invocation of services as a message flows through the integration.
- A send request/receive reply pair can be used to test a webMethods service to trigger an end-to-end test or unit test and check the final response.
- A receive request/send reply pair of test types can be used to simulate (stub) webMethods IS services. Simulated services will override webMethods IS services without the need to place any code in existing flow services.

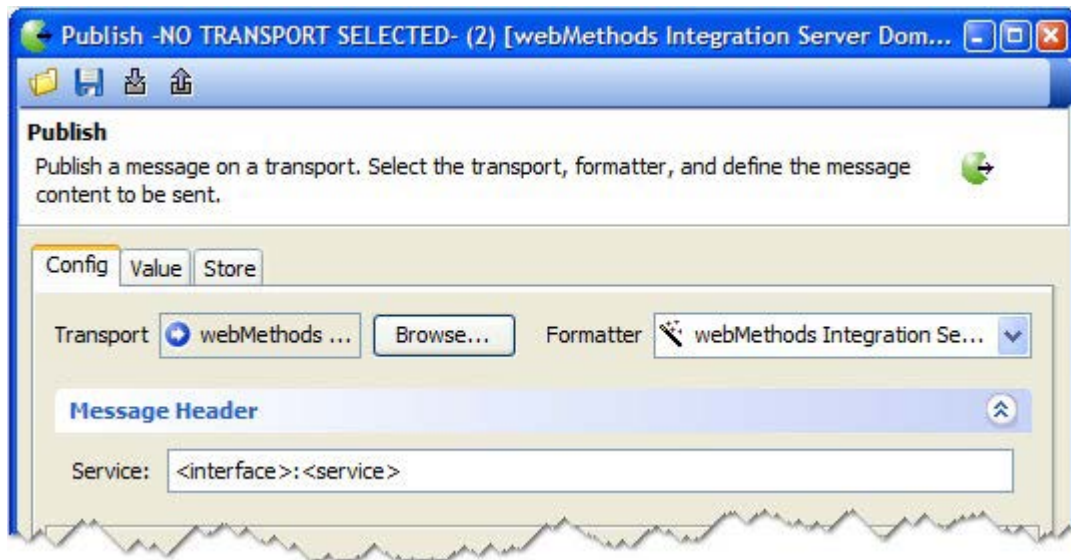
NOTE: webMethods IS messages can be recorded over an active VPN connection, and any stubs created from those recorded messages will simulate the same connection.

NOTE: In the JVM Arguments of the Library Manager utility, users can designate fields in received messages that should be dropped if they contain any of the specified attributes. The argument is `-Dgreenhat.wmis.drop.xml.attributes=`, and the list of field names to drop should be separated by commas. For example, `-Dgreenhat.wmis.drop.xml.attributes=@version,@encoding` will drop fields `@version` and `@encoding`, but not any others.

5.2 Test Actions Using the Integration Server Transport

When choosing an Integration Server transport for a publisher or subscriber, the only available formatter should be “Integration Server IData”, which will transform data between a Rational Integration Tester message and an IData instance.

A service must be specified in the following format: **interface:serviceName**.



With the IData formatter, you can add, edit, and remove fields from the message, including arrays and records.

NOTE: Integration Server documents/schemas are cached locally whenever they are refreshed (for example, when the project is synchronized, when the environment is changed, when a schema is selected in the Schema Library, and so on). This enables the creation of tests (manually and by means of MEP) when the connection to the server is unavailable. When tests are actually executed, however, the connection to the server must be available.

5.2.1 Send Request/Receive Reply

- The Send Request action will construct the IData from the message and send it off to the Integration Server.
- The Receive Reply action will accept the resulting IData.

5.2.2 Receive Request/Send Reply

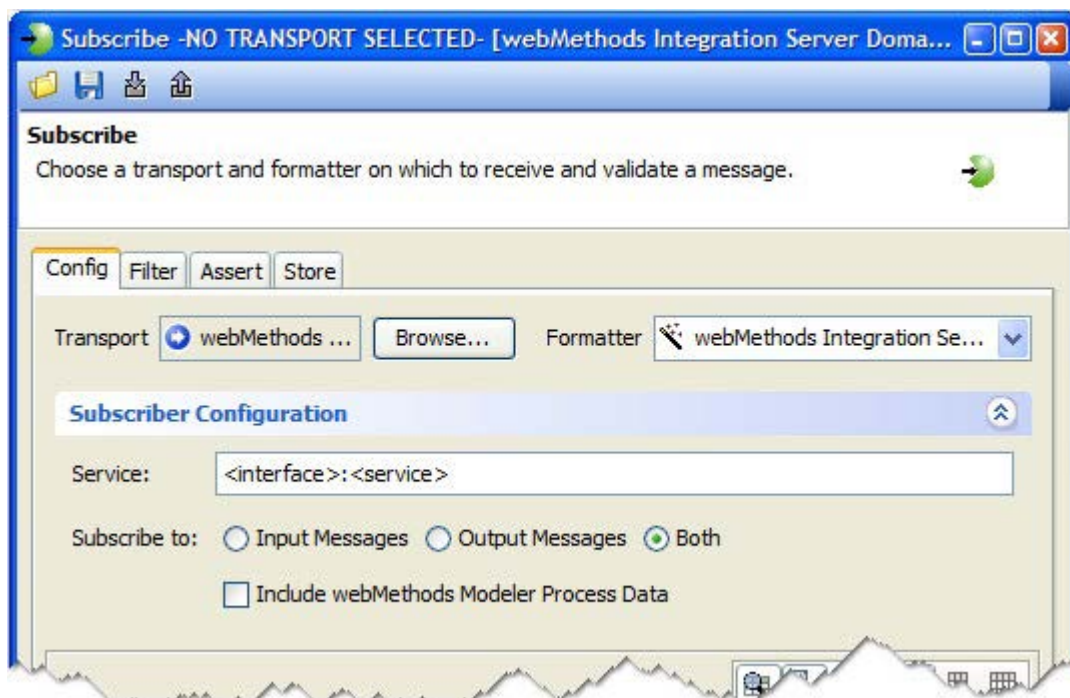
- The Receive Request action will accept the message and the service name.
- The Send Reply action will send the message to the service received in the request that is specified under **Reply To**.

5.2.3 Publish

The Publish action will pass the selected message to the specified service.

5.2.4 Subscribe

The Subscribe action will listen to an Integration Server for activity on a specified service, passing up messages for the input IData and/or the output IData. You can configure whether to pass up messages for the input, the output, or both. You can also include or exclude webMethods Modeler process data.



Troubleshooting

Contents

**Error When Synchronizing
Packages**

**Errors When Recording and Fields
Discarded**

This chapter provides an overview of how to use messaging actions in Rational Integration Tester to trigger tests or test Integration Server services.

6.1 Error When Synchronizing Packages

Error Message:

com.wm.app.b2b.client.ServiceException: com.wm.app.b2b.server.ServiceException:
[ISS.0085.9150] Cannot perform operation without List ACL on package XXXXXX

Possible Solution:

The user that has been specified in the configuration of the physical webMethods server needs to have permissions to read/list the packages that are present on the server.

The current configuration can be seen in Developer by opening the package and viewing the properties; under there will be a List ACL entry with a group specified. The user that you configure in Rational Integration Tester must be a member of this group.

6.2 Errors When Recording and Fields Discarded

XML doc and XML nodes are not supported types when recording webMethods events. These fields are represented as internal references on the server and Rational Integration Tester is unable to produce a string representation for them.

When such fields are encountered, they are discarded and an error is logged on the server.

Glossary

The following table below lists some of the key terms used in this document, and provides a description of each.

Term	Description
Field	A bit of data constituent to a message. Most fields are scalar and therefore unitary, equivalent to data attributes. Vector fields are an aggregation of fields both scalar and vector, and are usually referred to as Messages. See also Message.
Message	A unit of information made up of a header consisting of meta-information and a body consisting of the message data.
Host	The computer on which a software process runs.
Publisher-Subscriber	A messaging paradigm whereby a messaging network consists of Publishers and Subscribers.
Transport	Informally, the messaging software in use. For instance, TIBCO Rendezvous, TIBCO ActiveEnterprise, IBM WebSphere® MQ (JMS).
Publishing	Making a message (data) available on a message channel.
Subscribing	Receiving a stream of messages (data) on a given message channel.
Server	A host computer on a network shared by more than one user.
webMethods Broker	The central webMethods communications component that manages traffic for all other components. Broker clients, including adapters, connect to the broker to send and receive messages (also referred to as events).
webMethods Integration Server	One of the core webMethods application servers that execute services and validate their responses against expected results.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT,

MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM United Kingdom Limited
Intellectual Property Law
Hursley Park
Winchester
SO21 2JN
Hampshire
United Kingdom

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the

capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corporation 2001, 2012.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks and service marks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at www.ibm.com/legal/copytrade.shtml.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

