Rational Integration Tester

Integration Guide for Software AG CentraSite

Version 8.0.0

Note

Before using this information and the product it supports, read the information in "Notices" on page 33.

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.

Contents

About this Publication

Contents

This document provides details about how to incorporate governance in your SOA by utilizing the integration of IBM® Rational® Integration Tester with Software AG CentraSite.

Intended Audience

This document is 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 covers the integration of Rational Integration Tester with Software AG CentraSite. It is assumed that you are already familiar with the version of CentraSite that you will be using.

Users of this document are assumed to have some experience using Rational Integration Tester and some of its core functions such as creating physical resources, creating and running tests and test suites, and creating stubs. Please refer to *IBM Rational Integration Tester Getting Started Guide* and *IBM Rational Integration Tester Reference Guide* for more information.

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.
Italic	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

This chapter describes the requirements for using Rational Integration Tester with CentraSite and provides details about specific product libraries that are required by Rational Integration Tester.

1.1 System Requirements

Rational Integration Tester integrates with CentraSite 8.0 and 8.2.

The same client requirements for using Rational Integration Tester apply to the CentraSite integration, and the same requirements provided by Software AG for using CentraSite Control apply.

1.2 CentraSite Libraries

To connect to a CentraSite server and utilize the features of the integration with Rational Integration Tester, the following libraries must be configured in Rational Integration Tester's Library Manager.

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

CentraSite 8.0 and 8.2

All CentraSite 8.x files can be found in `<CentraSite Root>\redist`, by default.

The following files are common to CentraSite 8.0 and 8.2:

activation.jar	ant.jar	ant-apache-log4j.jar
ant-launcher.jar	CentraSiteCommons.jar	CentraSiteDownloadDocuments.jar
CentraSiteDynLoader.jar	CentraSiteImportExport.jar	CentraSiteJAXR-API.jar
CentraSiteLCM.jar	CentraSiteLCM-api.jar	CentraSiteLCM-L10N.jar
CentraSitePolicy-API.jar	CentraSiteResourceAccess-API.jar	CentraSiteUtils.jar
CentraSiteUtils-L10N.jar	CentraSiteVMS.jar	CentraSiteVMS-L10N.jar
commons-codec.jar	commons-httpclient.jar	commons-lang.jar
commons-logging.jar	cstUtils.jar	groovy-all-1.6.0.jar
INMConfiguration.jar	INMConfiguration-L10N.jar	INMConfiguration-L10N_en.jar
inmUtil.jar	inmUtilConf.jar	jaxen.jar
jaxr-api.jar	jaxrpc.jar	jdom.jar
log4j.jar	PolicyLogBindings.jar	saaj.jar
saxpath.jar	sin-common.jar	sin-misc.jar
sin-ssx.jar	sin-xmlserver.jar	stax-api.jar
TaminoAPI4J.jar	TaminoAPI4J-l10n.jar	uddiKeyConverter.jar
wsdl4j.jar	wstx-asl.jar	wvcm.jar
xmlbeans.jar	xqj-ino-api.jar	

The following files are specific to CentraSite 8.0:

axis.jar	CentraSiteLCM-L10N_en.jar	CentraSiteLMS-api.jar
CentraSiteLMS-L10N.jar	CentraSiteLMS-L10N_en.jar	CentraSiteLogPurging.jar
CentraSiteUtils-L10N_en.jar	CentraSiteVMS-L10N_en.jar	CSIntegrator.jar
LMS.jar	TaminoAPI4J-l10n_en.jar	WSIComplianceBindings.jar
xqj-api.jar		

The following files are specific to CentraSite 8.2:

CentraSiteDynLoaderExt.jar	xqjapi.jar

1.3 CentraSite 8.0 Action Templates

IBM supplies three action templates for CentraSite Policies. The action templates (.zip files) are provided in the Rational Integration Tester installation and can be found in `<Rational Integration Tester Installation Directory>\tools\SoftwareAG\CentraSite\8.0\ActionTemplates`. These .zip files have to be defined in CentraSite and then added to a Policy. The Policy is associated with Services when the Pre-State Change event occurs. This is how CentraSite knows when to enforce the policy.

The supplied action templates are described below:

Test Run Age Validator	This action template checks if the certification suite in Rational Integration Tester that is linked to the Service has been executed in the last *n* days. If not, the lifecycle state change will be denied.
Test Successful Validator	This action checks if the certification suite in Rational Integration Tester that is linked to the Service has passed or not. If the suite did not pass the last time it ran, then the lifecycle state change will be denied.
Test Up To Date Validator	This action checks if the certification suite in Rational Integration Tester that is linked to the Service has been run since the last time the service has been modified. Any change to the service counts as a modification and will change the service modification date. So if this action template is in the policy file, the very last thing to be done before changing the lifecycle state is to execute the test. Whether it passed or failed is irrelevant to this action template. If the Service has been modified since the test was last run, the lifecycle change will be denied.

The defined policies will be enforced when a user tries to change the lifecycle state of a web service in CentraSite. The lifecycles are user-defined, so an IBM policy needs to be created. The intent is that when the Service is published for public consumption, the IBM policy will be enforced.

1.3.1 Installing CentraSite 8 ActiveSOA Action Templates

The following sections provide information about installing the CentraSite ActiveSOA Action Templates.

Prerequisites

1. You must be using ActiveSOA version of CentraSite – not the Community Edition. A CentraSite 8 license is required to turn Community Edition into ActiveSOA.

2. You must have created a user in CentraSite with the appropriate Roles to manage lifecycles and Policies – Organization Administrator and Policy Administrator.

3. An "IBM" organization containing a lifecycle model for Services must exist.

Adding the Action Templates

The first step in defining the IBM policy is to create each of the action templates (that is, the "rules" that enforce the policy).

1. Log in to CentraSite with the appropriate permissions.

2. Navigate to **Policies > Action Templates**.

3. If there is no category called "IBM," create one by clicking **Add Action Category...** and name the new category IBM.

4. Click the **Add Action Template** button.

5. Choose **IBM** as the category and enter the name as "Test Successful."

6. Select **Java** as the implementation.

7. Click **Browse** to upload the .zip file and navigate to the action templates in `<Rational Integration Tester Installation Directory>\tools\SoftwareAG\CentraSite\8.0\ActionTemplates`.

8. Select `TestSuccessfulValidator.zip`.

9. Locate the **Service for Object Type** option and enable it (put a check next to it).

10. Choose **Pre-State Change** as the Event type.

    NOTE: If you have just added the Test Run Age template, see Special Instructions for Test Run Age before saving.

11. Click **Save**.

12. Repeat the steps above for the other two template .zip file in the tools directory (changing the name and file where appropriate).

Special Instructions for Test Run Age

The Test Run Age template has a parameter that must be defined, otherwise an exception is thrown by CentraSite.

1. Click on the **Parameter Templates** tab.

2. Click the **Add Parameter Template** button.

3. Enter the name as "Time-in-Days" – exactly as shown without the quotes.

4. Select **Number** as the type.

5. Provide a default positive integer value, which is the time in days that the certification suite has to have been executed in order for the lifecycle state change to be accepted. One (1) is sufficient, but the value can be any positive integer.

6. Click **Save**.

Creating the IBM Policy

1. Navigate to **Design/Change-Time**.

2. If a IBM policy does not already exist, click the **Add Policy** button.

3. Enter "IBM" as the name.

4. Optionally enter a description and version, and leave priority at **11**.

5. Select **Service** as the Object Type, **Pre-State Change** as the Event Type, and **IBM** as the Organization.

6. Click **Next**.

7. Click the **Add States** button.

8. Select the lifecycle model created above from the list and click **OK**.

9. Click **Next**.

10. Locate the action templates you created in the **Available Actions** list and place a check next to each of them.

11. Click the right arrow button to move the checked action templates into the **Selected Actions** list.

12. Press the **Finish** button.

13. Click the **Change State...** button, select **Productive**, then click **OK**.

The IBM policies are now ready to use.

1.3.2 Disabling CentraSite 8 ActiveSOA Action Templates

If necessary, follow the steps below to disable the action templates:

1. Navigate to **Design/Change-Time**.

2. Locate the IBM policy and click to edit it.

3. Click the **Change State...** button, select **Suspended**, then click **OK**.

Certifying CentraSite Services with Rational Integration Tester

Contents

This chapter describes how to configure a CentraSite server in your Rational Integration Tester project and use tests, test suites, and stubs to certify web services in the CentraSite registry.

2.1 Overview

Software AG's CentraSite is an SOA registry/repository that helps promote the re-use of services, assure quality, and keep services aligned with business needs. It also enables organizations to enforce standard processes and conventions through the application of policies.

Through the integration with Rational Integration Tester, CentraSite users can test and validate the services in the repository, enabling everyone to use and re-use assets that are certified.

The features provided by the integration are powerful, yet they are easy to use and understand. Once you have defined a CentraSite server in Rational Integration Tester, you can immediately begin to import and test the services that are stored on it.

The following sections describe how to enable the integration and work in Rational Integration Tester to certify CentraSite services.

- Define the CentraSite Server

- Manage Rational Integration Tester Attributes in the CentraSite Registry

- Manage CentraSite Publication Fields

- Add a Registered WSDL to Rational Integration Tester

- Certification Test Suites

- Publishing Test Suite Reports to CentraSite
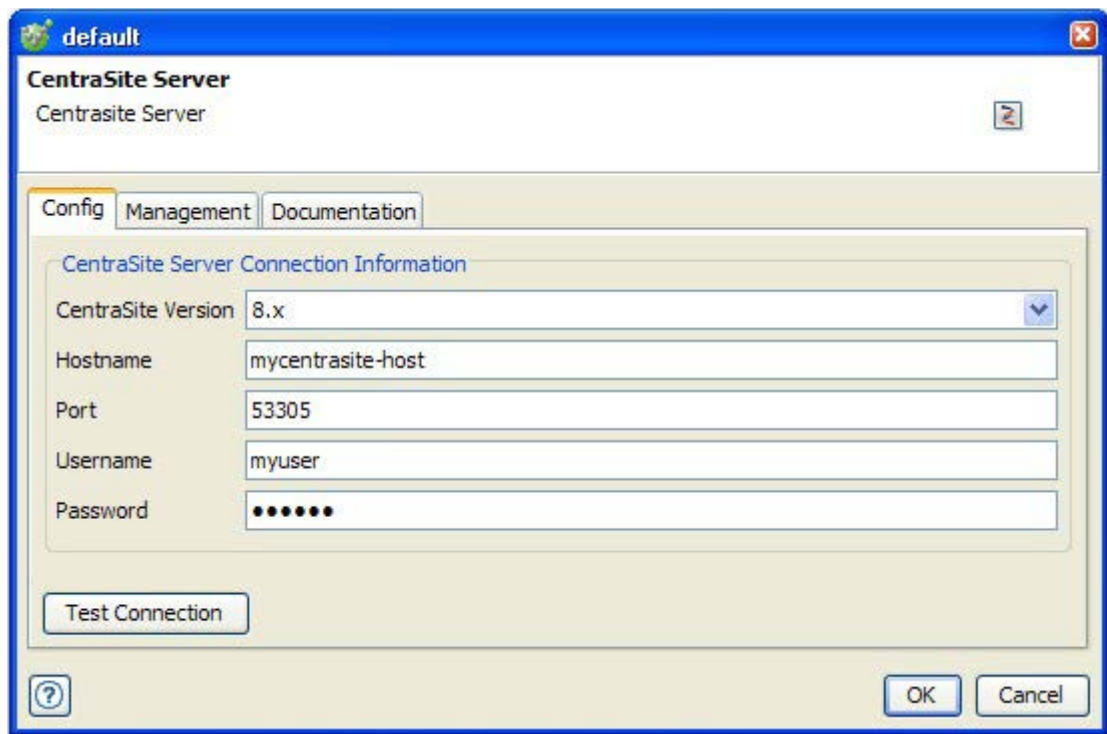
- Publish a Stub to CentraSite

2.2 Define the CentraSite Server

Before you can access a CentraSite server from Rational Integration Tester, you must define the server and configure the appropriate connection parameters in the Physical View of Rational Integration Tester's Architecture School.

1. In Rational Integration Tester, open the Architecture School perspective (**F7**) and select the Physical View.

2. Click the **Software AG** menu and select the **CentraSite Server** option.

   A new server is created under "Unconnected Resources."

3. Double-click the new server to view and configure its properties. 53305

4. Under the **Config** tab, enter the version, host, port, and user name/password combo of the CentraSite server to which you want to connect.

   NOTE: The user name and password should be the same as those used to log in to the CentraSite Control page (web-based admin portal).

5. Click **Test Connection** to ensure that the configuration is valid, and click **OK** when finished.

2.3   Manage Rational Integration Tester Attributes in the Centra-Site Registry
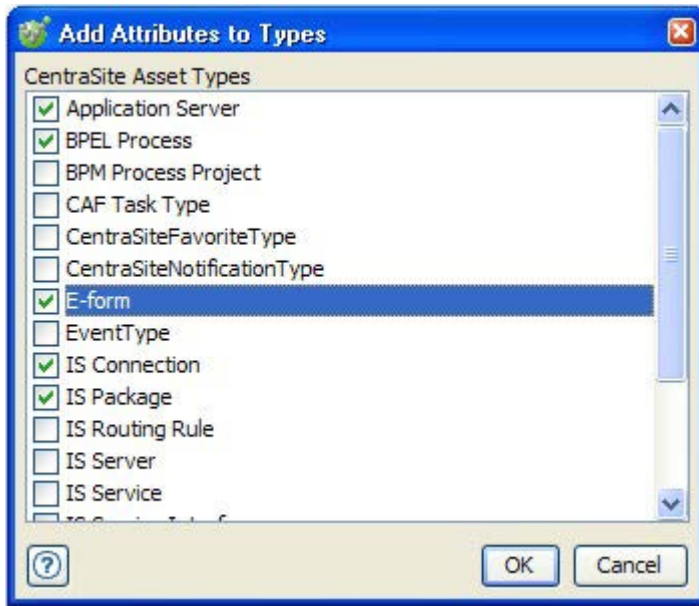
Once you have created and successfully connected to a CentraSite server in the Physical View of Rational Integration Tester's Architecture School, you must publish the Rational Integration Tester attributes to the server (that is, define the attributes to update on services tested by Rational Integration Tester, viewable under the **GH Tester** tab for each service). If you are no longer using a specified CentraSite server with Rational Integration Tester, you may also want to remove the registry attributes from the server.

To publish or remove the registry attributes, open the CentraSite Server in the Physical View of Architecture School and select the **Management** tab.

Click the **Publish Attributes** button to publish Rational Integration Tester attributes to the registry. If all available attributes have already been published, you will be

notified. If any available attributes remain unpublished, however, the **Add Attributes to Types** dialog is displayed.

Select one or more available (that is, unpublished) attributes to publish, then click **OK**.

To remove any attributes, click **Remove Attributes**. If there are no attributes to remove, you will be notified. If any attributes have been published, however, the **Remove Attributes from Types** dialog is displayed.
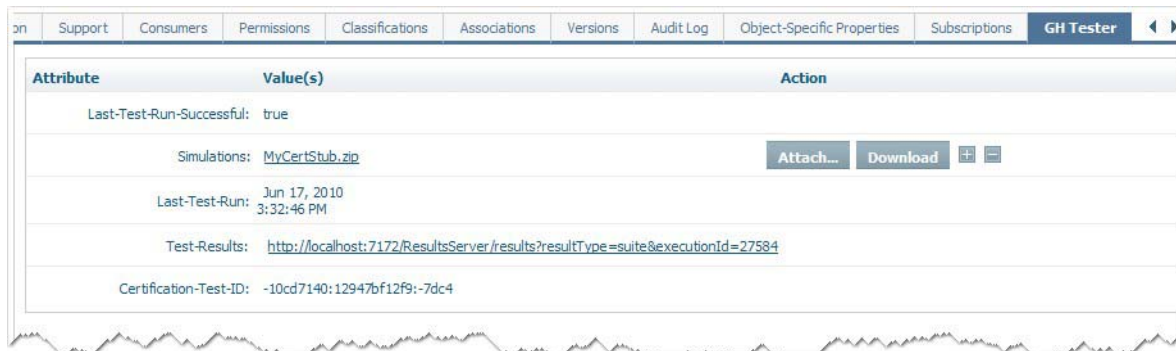

Select one or more previously published attributes to remove, then click **OK**.

NOTE: For CentraSite 8.0, only **Service** type attributes are available. For CentraSite 8.2, all available types will be displayed.

NOTE: If you run a a certification suite (see Certification Test Suites) that contains unpublished attributes, those attributes will be published automatically.

The Rational Integration Tester attributes can be seen when viewing any of the registered services on the CentraSite server. Select one of the services from the Asset Catalog and click the **GH Tester** tab, which should be the last tab on the left.

The Rational Integration Tester attributes include the following information:

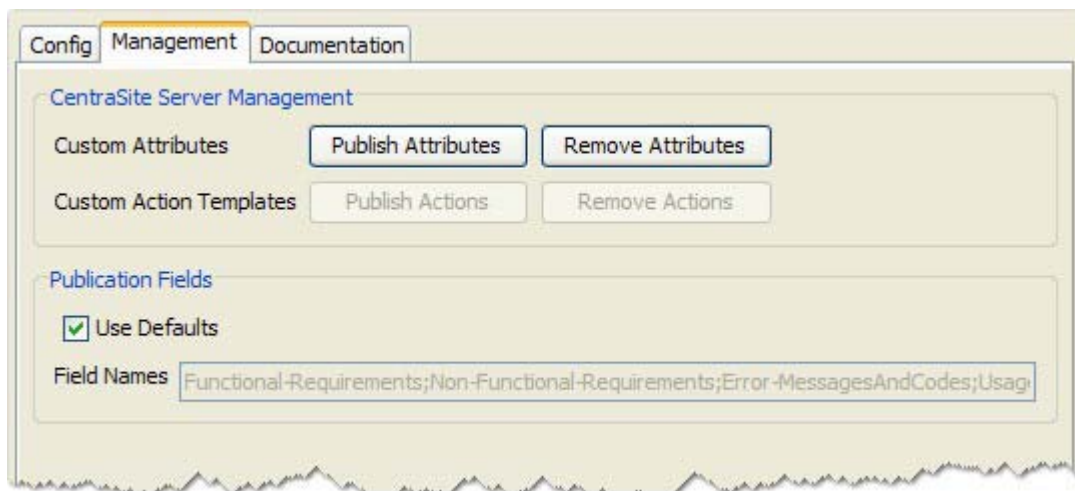| Attribute	Value
Last Test Run Successful	Indicates whether or not the most recent run of the certification test suite for the service was successful.
Simulations	Lists any stubs that might be needed by other users to test the same service in their own environment.
Last -Test-Run	The date and time when the service's certification test suite was last executed.
Test-Results	The Rational Integration Tester Results Server URL where the execution results for the certification test suite can be viewed.
Certification-Test-ID	The internal Rational Integration Tester ID for the service's certification test suite.

2.4 Manage CentraSite Publication Fields

Publication fields define report types that can be published to the CentraSite server for marked test suites in Rational Integration Tester. The published reports are made available under the **Specification** tab when viewing a service in CentraSite's Asset Catalog.

The default fields that can be made available to any CentraSite server are defined in Rational Integration Tester's Library Manager utility. Under **Application** settings, all possible fields from all servers should be included in **CentraSite Publication Fields** (see the *IBM Rational Integration Tester Installation Guide* for more information).

Under the **Management** tab of the physical CentraSite server component (in Architecture School), you can specify whether all of the default fields should be available – or customize which ones to display – for the specific server instance.

To manage which fields should be available when using the specified server, open the desired CentraSite Server in the Physical View of Architecture School and select the **Management** tab.

Under **Publication Fields**, leave **Use Defaults** checked to display all of the field names that have been defined in Library Manager. To customize the fields that are displayed (that is, remove one or more of the default fields, or add new ones), clear the **Use Defaults** check box and add or remove the names of the desired fields.

NOTE: Fields names should be separated with a semicolon, and no semicolon should be used after the last field.

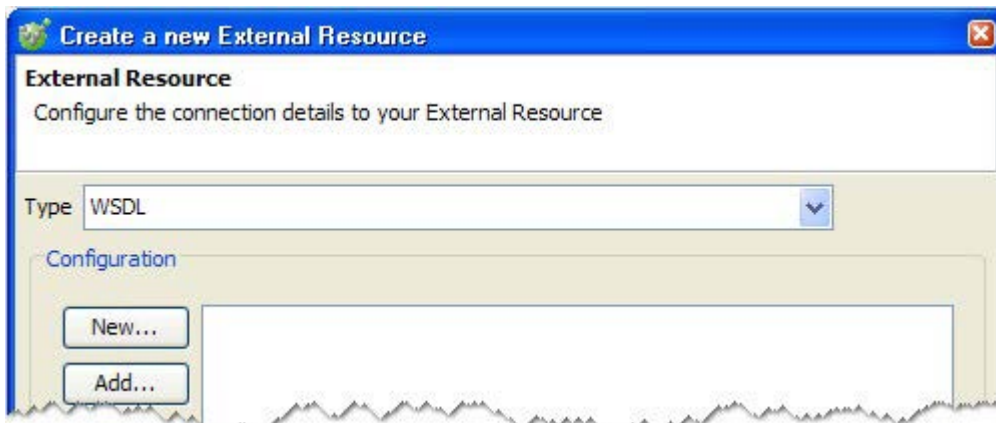## 2.5 Add a Registered WSDL to Rational Integration Tester

The integration between Rational Integration Tester and CentraSite relies on the testing of the services (WSDLs) that are in the CentraSite registry using Rational Integration Tester. The registered WSDLs can be imported into Rational Integration Tester from CentraSite (by means of UDDI), then tests and test suites can be created and executed as usual.
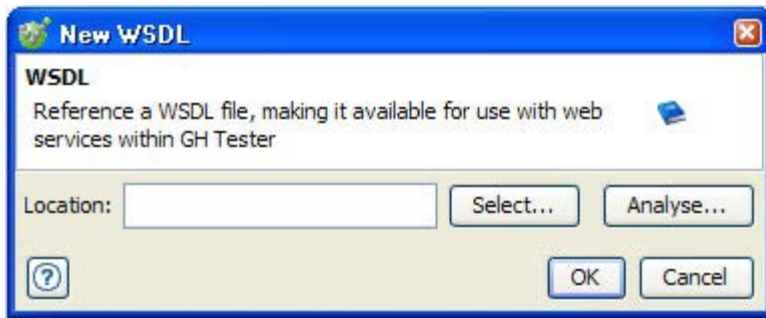
Follow the steps below to add a registered WSDL to Rational Integration Tester:

1. Open the Logical View of Architecture School (**F7**) and select **WSDL** from the **General** or **Web** menu.

2. In the **External Resource** wizard, ensure **WSDL** is selected in the **Type** field, then click **New** under **Configuration**.

3. In the **New WSDL** dialog, click **Select**.

4. In the **Select Location** dialog, select the **UDDI** tab and click **Browse**.

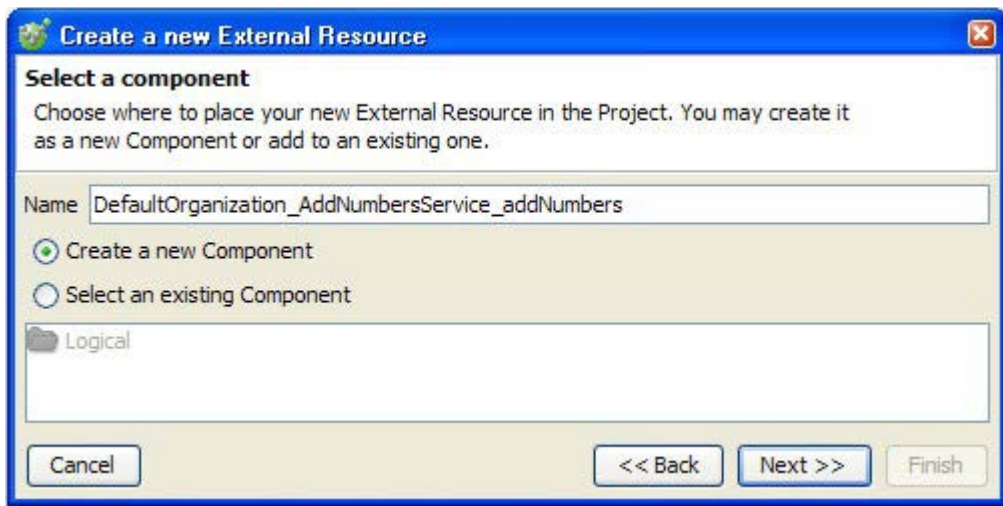5. In the **Browse UDDI** dialog, ensure that the correct CentraSite server is selected under **UDDI Inquiry URL**.

6. If you want to filter your search of the registry, enter the search criteria and conditions using the name, sort, and match options, then click **Find Organizations** or **Find Services** to display a list of matching items.

7. In the list of organizations or services that is returned, locate and select the desired service, then click **OK**.

8. Click **OK** in the **Select Location** and **New WSDL** dialogs to return to the **External Resource** wizard.

9. Ensure that the desired WSDL appears under the **Configuration** panel.

10. If you want to add another WSDL, repeat steps 2 through 9. When you are ready to proceed, click **Next**.

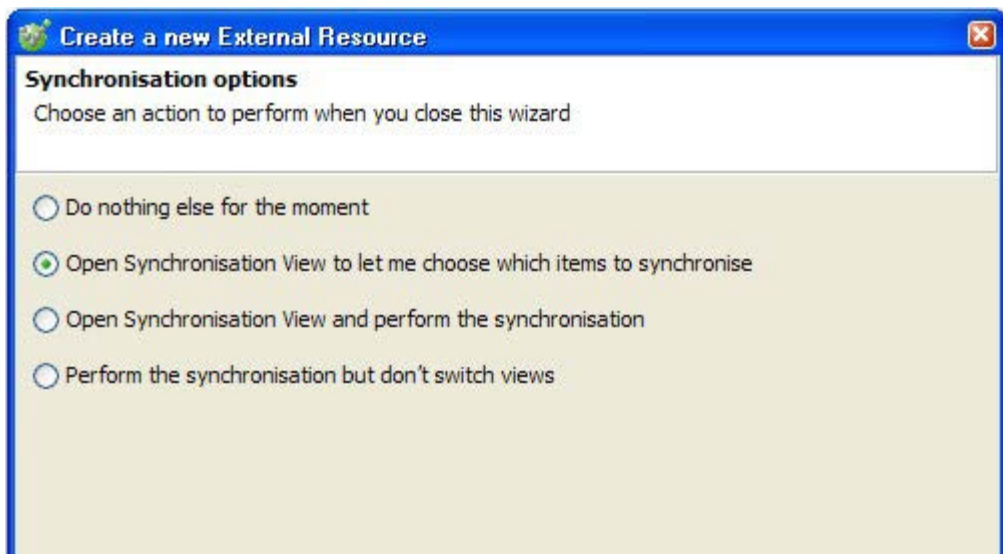The **Select a Component** dialog is displayed.

11. Select the desired option for adding the WSDL to a project component – add it to a new component with a user-defined name, or add it to an existing component – then click **Next**.

    If at least one environment already exists in the project, the **Environment** dialog is displayed. If no environments exist, this dialog is skipped and a new environment is created automatically, named according to WSDL file name.

12. If the dialog is displayed, select the desired option for your environment – create a new environment with a user-defined name, or use an existing environment – then click **Next**.

13. The **Synchronisation Options** dialog is displayed.

14. Select the desired synchronisation option, then click **Finish**.

> **NOTE:** You should synchronize the WSDL to ensure that all of the required test artifacts are created in the Rational Integration Tester project.

The WSDL is now a component in your project and you can use it to create tests, test suites, and stubs that will be used for certification.

2.6   Publishing Test Suite Reports to CentraSite

When viewing registered services from the Asset Catalog in the CentraSite registry, the **Specification** tab contains a list of reports. The available report categories may differ from server to server, but there is typically a core set of fields or categories that are common among all servers.

NOTE: Users can define their own fields for publishing, so long as they are "file type" attributes and are associated with the "service object" type.

A summary execution report for any Rational Integration Tester test suite executed against a registered service can be published to one of these fields. The fields that are available for publishing are managed in Rational Integration Tester's Library Manager and under the Management tab of the physical CentraSite resource (in Architecture School). See Manage CentraSite Publication Fields for more information.

NOTE: The results for any test suite can be published to only one field at a time.
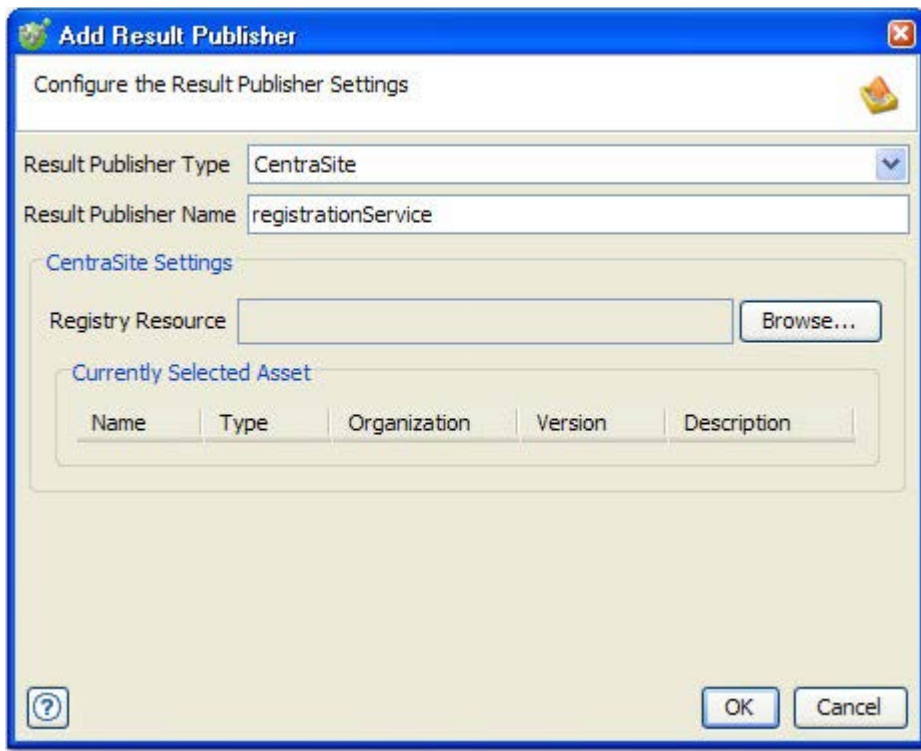
NOTE: General information about publishers and project-wide publication settings can be found in the *IBM Rational Integration Tester Reference Guide*, in the "Project Settings: Results Publishers" section of the "Rational Integration Tester Overview" chapter.

2.6.1   Configure Results Publishers for CentraSite

Test suite reports are published to CentraSite using Results Publishers, managed under the **Results Publishers** tab of Rational Integration Tester's **Project Settings** dialog. Follow the steps below to create CentraSite publisher.

1. Select **Project Settings** from the **Project** menu in Rational Integration Tester's main toolbar.

2. In the **Project Settings** dialog, click the **Results Publishers** tab.

3. Click **Add** to create a new publisher.

The **Add Result Publisher** dialog is displayed.

4. Select **CentraSite** as the publisher type from the **Result Publisher Type** combo box and provide a name for the publisher in the **Result Publisher Name** field (only alpha-numeric characters, hyphens, and underscores are permitted).

5. Under **CentraSite Settings**, click **Browse** next to the **Registry Resource** field to select the CentraSite server and service asset for which the publisher will be used.

The **XML Registry Browser** is displayed.

6. Select an existing CentraSite server from the **Registry Resource** combo-box.

7. Under **Search Criteria**, enter your search criteria and conditions using the organization, name, case, and match options.

To search all services, enable the **All Services** radio button. To search a recently used asset type – most recently used types are listed – enable radio button for the desired type. To search a type that is not listed, enable the **Other** radio button to display the asset type selector.

8. To select a new asset type, select it from the list of those available on the server and click **OK**.

9. When your search criteria are set as desired, click **Search**.

10. From the list of matching assets that is displayed, select the desired asset and click **OK**.

The selected asset will be listed in the result publisher.

11. To create the publisher using the selected asset, click **OK**. Otherwise, browse the registry again to select a new asset.

12. When finished, the new publisher will be displayed in the list of those available under the **Results Publishers** tab.

2.6.2 Configure Publishing Options for a Suite

Once one or more CentraSite publishers have been created in the project, they can be added to a test suite.

NOTE: Detailed information about configuring results publishers for test suites can be found in *IBM Rational Integration Tester Reference Guide.* The steps below only describe the aspects of publishing that are specific to CentraSite.

1. Open the desired test suite for editing in the Test Factory.

2. Select the **Publish Results** tab in the editing panel.

3. Click **Add** to add a publisher to the suite.

   The **Add Publish Results Settings** dialog is displayed.

4. Select the desired CentraSite publisher from the Result Publisher combo-box.

5. Select the report to include from the Custom Report combo-box (see *IBM Rational Integration Tester Reference Guide* for more information about custom reports).

6. Select the desired option for when the report should be published (always, on pass, or on fail).

7. Under the **CentraSite Settings**, select the desired publication field for the report from the **Publication Field** combo-box (see Manage CentraSite Publication Fields for more information).

8. If the current suite should be the certification suite for the selected asset, enable the **Is Certification Test** option (see Certification Test Suites for more information).

> **NOTE:** Once a suite is configured to publish to one of the configured publication fields (see Manage CentraSite Publication Fields) and has been saved, it will be marked with the publication icon ![icon] in Rational Integration Tester.

2.7 Certification Test Suites

The main goal of using a service registry/repository like CentraSite is to provide users with the ability to use and re-use services that are certified. In Rational Integration Tester, users can create a battery of tests, stubs, and test suites that can validate these services at any time.

For this purpose, a certification test suite can be designated in Rational Integration Tester. This is the master suite – containing all other tests and test suites – that can be executed to validate the selected service.

Once a test suite is used to certify one of the services registered on the CentraSite server, users can quickly see if the service is valid for use by looking under the **GH Tester** tab for the service (see Manage Rational Integration Tester Attributes in the CentraSite Registry for an example).

NOTE: Only one certification test suite can be designated for each registered service, but a single suite can publish to multiple services.

A test suite in Rational Integration Tester can be designated as a certification suite using the results publisher settings, described in Configure Publishing Options for a Suite. In the results publisher settings, simply enable or disable the Is Certification Suite as desired.

NOTE: If another test suite has already been designated as the certification suite for the asset configured in the selected publisher, a confirmation dialog will be presented where you can cancel the action or replace the existing certification suite.

Once a certification suite has been marked and the test suite is saved, it will be designated as such in Rational Integration Tester, as follows:

2.8 Publish a Stub to CentraSite

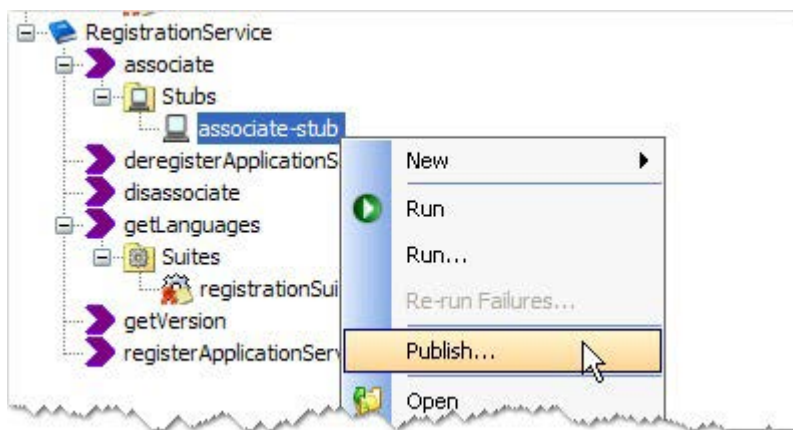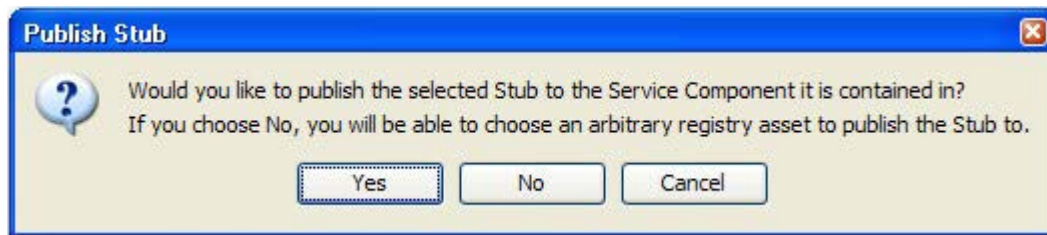If a Rational Integration Tester has created one or more stubs that are used in the validation of all or part of a registered CentraSite service, those stubs can be published as simulations that will appear under the **GH Tester** tab for the specified CentraSite service. Other Rational Integration Tester users can then download the stubs for use in their own project.

A stub that is the child of a service component generated from a CentraSite WSDL can be published to CentraSite by right-clicking it in the Test Factory and selecting **Publish...** from the context menu.

When publishing the stub, you will be asked whether or not you want to publish it to the service component that contains it.

If you choose **Yes**, the stub will be published to the containing service component on the CentraSite server. If you choose **No**, the registry browser will be displayed (see page 24), letting you locate and select a different asset to which the stub should be published.

When published, all of the required Rational Integration Tester components and test artefacts that are required to execute the stub are packaged into a .zip archive. Once the stub is successfully published, a confirmation dialog will be displayed to the user.

In CentraSite, published stubs can be seen under the **GH Tester** tab when viewing a service from the Asset Catalog.

To download any of the published stubs, click the **Download** button. Once saved, the downloaded file can be unloaded, and the Rational Integration Tester components it contains can be copied into another Rational Integration Tester project.

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, below).
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.
Publish-Subscribe	A messaging paradigm for efficient one-to-many communication in which one process (the publisher) sends information to zero or more other processes (subscribers).
Transport	Informally, the messaging software in use. For instance, TIBCO EMS, 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.
Subject	A user-defined, meaningful name for identifying messages on transports. For example, the subject EQ.IBM might identify all pricing data about IBM stocks, while EQ.IBM.N might identify price data from the New York Stock Exchange only.

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

34

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.