



IBM Software Group | Rational software

IBM® Rational® TestManager

Setting up and troubleshooting the TestManager and Rational Functional Tester integration

Rational software



@business on demand.

© 2008 IBM Corporation

Updated December 29, 2008

This module covers the setup and troubleshooting for the integration of IBM Rational TestManager and Rational Functional Tester, versions 7.0 and above.

Module objectives

- This module covers:
 - ▶ The benefits of associating a Rational Functional Tester (RFT) project with a TestManager (TM) project
 - ▶ The steps to associate a TM project with an RFT project
 - ▶ What can cause the integration to fail
 - ▶ The steps to make the TM and RFT integration work again after a failure
- After completion of this module, you will:
 - ▶ Understand the advantages of associating a TM project with an RFT project
 - ▶ Understand why the integration might fail
 - ▶ Be able to fix the integration if it stops working



This module discusses the benefits of associating a TestManager (TM) project with a Rational Functional Tester (RFT) project. It includes the steps to integrate TM and RFT, causes for integration failure, and steps to get the TM/RFT integration working after a failure.

After completion of this module, you will understand the advantages of associating a TM project with an RFT project, will understand why the integration might fail, and will know how to fix the integration if it stops working.

Associating a TM project with an RFT project

- Benefits
 - ▶ Manage all testing assets from one TM project
 - ▶ Record and playback all RFT scripts from TestManager
 - ▶ Can playback an RFT script on different computers
 - ▶ Log management
 - ▶ Reporting capabilities
 - ▶ Can run several RFT scripts at once



There are several benefits when associating a TM project with an RFT project. It is an easy way to manage all testing assets from one Rational project. Association allow you to record and play back all Functional Test scripts from TM. It also allows play back of an RFT script on different systems using Rational Test Agents set up through TM. You can manage logs more easily through association, including RFT logs, using TM's Test Log windows. Finally, you can apply TM reports to gather information about the results of running RFT scripts and obtain coverage reports for a testing project. This is in addition to using a TM suite to coordinate the way that scripts run. Note that several scripts and computers can be involved in a test, and you can employ suites to run scripts in parallel on available computers allowing tests to run more quickly.

Associating a TM project with an RFT project

- To associate a TM project with an RFT project:
 - ▶ Create a Rational TM project using Rational Administrator before associating it with an RFT project.
 - ▶ Start RFT and log in to a TM project.



Rational Test Login

User Name:
admin

Password:

Project:
Datapool

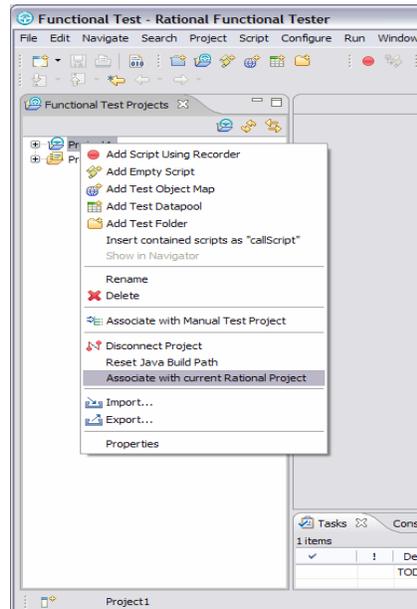
Location:
C:\Datapool\Datapool.rsp

An RFT project can only be associated with a TM project if it is currently not associated with any other project. Accordingly, “Associate with Current Rational Project” is only available if a user has administrative privileges with the project.

Before associating a TM project with an RFT project, ensure that a TM project has been created using Rational Administrator. To begin the process, start RFT and log in to a TM project.

Associating a TM project with an RFT project

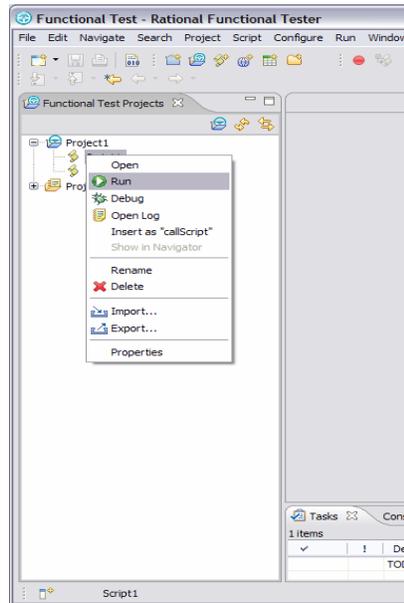
- To associate a TM project with an RFT project:
 - ▶ From the RFT Projects view, right-click an RFT project, and select “Associate with current Rational Project.”



From the RFT Projects view, right-click an RFT project, and select “Associate with current Rational Project.”

Associating a TM project with an RFT project

- To associate a TM project with an RFT project:
 - ▶ Play back a script from RFT:
 - Click “Run Functional Tester Script” from RFT toolbar
 - Right-click the script, then select “Run.”
 - The results can be viewed in TM.
 - Use Solution Explorer if using VB.NET scripting

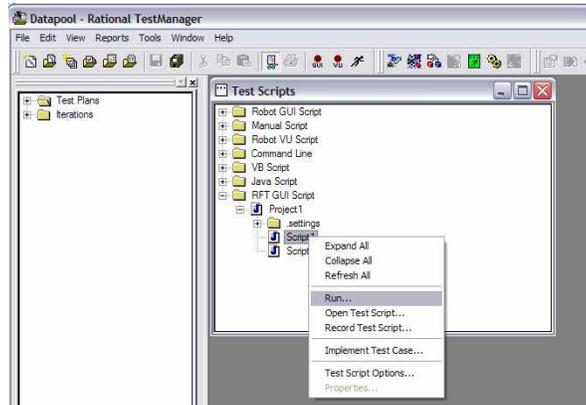


Before running a script from RFT, configure the application for testing by setting the appropriate Java™ environment or Web browser to run the application.

To run an RFT script, in the RFT projects view, select a script. In the Functional Tester toolbar, click “Run Functional Tester Script.” In the Functional Tester Projects view, right-click a script and select Run. If using VB.NET scripting instead, use the Solution Explorer.

Associating a TM project with an RFT project

- To play back a script from TM:
 - ▶ Log in to the TM project.
 - ▶ Select View->Test Scripts, then select the test script type.
 - For RFT scripts, click "Functional Test GUI Script," then select the script
 - ▶ Right-click the script, select Run.



To play back a script from TM, first log in to the TM project. Select View->Test Scripts, then select the test script type. For RFT scripts, click "Functional Test GUI Script," then select the script. Finally, right-click the script, and select Run.

Additionally, there is a second way to run an RFT script from TM. First, log in to the TM project. Select File->Run Test Script->GUI-Functional Tester-> <name of datastore>. Select an RFT script, click Open, then click OK.

Reasons the integration can fail

- The TM project is moved from one server to another and the paths are not updated. To verify if the path is incorrect:
 - ▶ Select Tools->Manager->TestScript Types
 - ▶ Click Functional Tester GUI Script
 - ▶ Click Edit. In the Sources tab, one entry is visible for every Functional Tester datastore
 - ▶ Click Edit
 - ▶ In the Connection Data tab, verify that the path is correct



There are several ways that the integration can fail. This section of the module discusses the reasons involved. The remedies for these situations are further detailed in the next section of this module, titled “Getting the TM/RFT integration working again after a failure.”

One reason the integration might fail is if the TM project is moved from one server to another and the paths are not updated. To fix this, verify that the path is correct.

Select Tools->Manager->TestScript Types. Click Functional Tester GUI Script, then click Edit. In the Sources tab, one entry should be visible for every Functional Tester datastore. Click Edit. In the Connection Data tab, verify that the path is correct.

Reasons the integration can fail

- The .NET script does not recognize the Test Script Service Library.
 - ▶ You cannot associate the TM Test Script Option with RFT .NET scripting
 - ▶ This makes TM unable to pass values to RFT scripts

Another possibility for failure might be that the .NET script does not always recognize the Test Script Service Library. Here, you are not able to associate the TM Test Script Option with RFT .NET scripting. This event causes TM to be unable to pass values to RFT scripts.

Reasons the integration can fail

- The RFT datastore definition file no longer contains the TM project association.
 - ▶ RFT and TM integration will fail with this error:
“Failed to create script source in current Rational project because of the name conflicts. Reconnect this project using different logical name.”
 - ▶ TM stores old RFT project information in the Test datastore.

There is the possibility that the RFT datastore definition file no longer contains the TM project association. If this is the case, then RFT and TM integration will fail with the error listed on the slide. This can happen because TM stores the old RFT project information in the Test datastore.

Reasons the integration can fail

- The RFT datastore definition file no longer contains the TM project association.
 - ▶ The datastore definition file is located:
C:\Documents and Settings\username\IBM\rational\sdp7.0\workspace\ProjectName\resources
 - ▶ The datastore definition file does not contain a project path. This indicates that there is no association.
 - ```
<?xml version="1.0" encoding="UTF-8"?>
<DatastoreDefinition L=".DatastoreDefinition">
<ProjectName/><ProjectPath/><ScriptSourceUID/>
<DefaultMapName/><MajorVersion>2</MajorVersion>
<MinorVersion>2</MinorVersion>
<ProjectSubType>java</ProjectSubType>
<Name>Project1</Name><RMTProjectName/>
<RMTProjectPath/></DatastoreDefinition>
```

The datastore definition file is located at the address listed on the slide. Additionally, the datastore definition file does not contain a project path, which indicates that there is no association. An example of this is also listed on the slide.

## Reasons the integration can fail

- The integration uses duplicate RFT project names.
  - ▶ When two computer specific script sources have the same name (from two different test datastores), their registered datapaths overwrite each other.
  - ▶ The connection between TM and RFT is lost after associating the TM project with multiple RFT projects

Always use unique RFT project names before establishing the integration with the TM project, or the integration will fail. Since datapaths of all computer specific script sources are registered in the system registry, when two computer specific script sources have the same name (from two different test datastores), their registered datapaths overwrite each other. This causes the connection between TM and RFT to be lost when the TM project is associated with multiple RFT projects.

## Get the TM/RFT integration working after a failure

- The TM project is moved from one server to another and the paths are not updated.
  - ▶ Check the paths in the TM files and manually edit them to reflect the new path.
  - ▶ The files where paths require modification are located:
    - ..\ProjectFolder\Project.rsp
    - ..\ProjectFolder\Testdatastore\project.dat
    - ..\ProjectFolder\Testdatastore\DefaultTestScriptDatastore\project.dat
    - ..\ProjectFolder\TestUsersAndGroups\project.dat
    - ..\ProjectFolder\TestDatastore\datastore.dat
    - ..\ProjectFolder\TestDatastore\DefaultTestScriptDatastore\datastore.dat
    - ..\ProjectFolder\TestUsersAndGroups\datastore.dat
    - ..\TestDatastore\TMS\_TestScriptSources\<ScriptSource>.TestScriptSource.rxml
  - ▶ After editing, disassociate and re-associate the TM project from the RFT project



This section of the module discusses how to get the TM/RFT integration working again after a failure.

If the TM project is moved from one server to another, the paths might not be updated, First, check the paths in the TM files, and manually edit them to reflect the new path. The files with paths that require modifications are listed on the slide.

After editing the paths to reflect their new locations, dissociate the TM project from the RFT project, then re-associate. This applies the new path information.

## Get the TM/RFT integration working after a failure

- To start RFT and TestManager and re-associate by way of RFT:
  - ▶ Select Start->All Programs->IBM Rational Functional Tester
  - ▶ Log in to the RFT workspace
  - ▶ When the TM dialog box opens, log in to the project to associate it with the RFT project
  - ▶ Right-click the RFT project and select “Associate with Current Rational Project”



Some situations require you to re-associate RFT and TM after a new workspace has been created. First, start RFT and TM, and re-associate by way of RFT. Select Start->All Programs->IBM Rational Functional Tester. Log in to the RFT workspace. When the TM dialog box opens, log in to the project to associate it with the RFT project. Right-click the RFT project and select “Associate with Current Rational Project.”

## Get the TM/RFT integration working after a failure

- If the RFT upgrade is unsuccessful, reinstall RFT and TestManager:
  - ▶ Uninstall RFT and TM
  - ▶ Install RFT
  - ▶ Install TM
  - ▶ Test to determine if test cases and suites can run with an RFT script

There is a chance that the RFT upgrade might be unsuccessful. To fix this, uninstall and reinstall RFT and TM in that order. Afterwards, test the programs to determine if test cases and suites can run with an RFT script.

## Get the TM/RFT integration working after a failure

- To apply a Hotfix if the .NET script does not recognize the Test Script Service Library:
  - ▶ Extract the contents of the Hotfix associate with defect RATLC119232
  - ▶ From the extracted contents, double-click RunHotfix.bat, which installs:
    - policy.7.0.rtxftnet.dll
    - rtxftnetpolicy.config
    - rtxftnet.dll
    - RtxIEDomain.dll
    - rxivsys.dll
    - SiebelDomainProxies.dll
    - SiebelIEHelper.dll
    - SiebelNotificationListener.dll

To apply a Hotfix if the .NET script does not recognize the Test Script Service Library, first extract the contents of the Hotfix associate with defect RATLC119232. From the extracted contents, double-click RunHotfix.bat, which installs the files listed on the slide.

## Get the TM/RFT integration working after a failure

- To apply a Hotfix if the .NET script does not recognize the Test Script Service Library:
  - ▶ After the batch file is complete, select Start->Run, and type "assembly"
  - ▶ Check the version of these DLLs, which should be version 7.0.0.61234:
    - rtxftnet.dll
    - SiebelDomainProxies.dll
    - SiebelEHHelper.dll
    - SiebelNotificationListener.dll

After the files have been installed, select Start->Run, and type "Assembly." Check the version of the dll files listed on the slide, which should be at version 7.0.0.61234. This ensures that the Hotfix is properly installed and is ready to use.

## Get the TM/RFT integration working after a failure

- To apply a Hotfix if the .NET script does not recognize the Test Script Service Library:
  - ▶ After applying the Hotfix, associate the Test Suite and Test Cases with the RFT Script.
    - From Rational TM, choose View->Test Scripts
    - Open the "RFT GUI Script" folder
    - Open the RFT script
    - Right-click the script, select "Test Script Options"
    - Type in "Option name" and "Option Value"
    - Click "OK"

After applying the Hotfix, associate the test suite and test cases with the RFT script. In TM, choose View->Test Scripts, and open the "RFT GUI Script" folder. Open the RFT script. Right-click the script, and select "Test Script Options." Type in "Option name" and "Option value," then click OK.

## Get the TM/RFT integration working after a failure

- Delete the current path to the TM project:
  - ▶ In TestManager, go to Tools->Manage->Test Script Types->RFT Tester GUI Script, and click "Edit..."
  - ▶ In the "Test Script Type Properties" window, click the "Sources" tab
  - ▶ Highlight the source that points to the problem RFT Tester datastore and delete it
  - ▶ Close TM, and restart RFT
  - ▶ At the logon window, click "OK" to associate with the TM's project

If there is a problem RFT path in TM, delete the current path to the TM project. In TM, go to Tools->Manager->Test Script Types->RFT Tester GUI Script. Click Edit. In the Test Script Type Properties window, click the Sources tab. Highlight the source that points to the problem RFT datastore and delete it. Close TM, and restart RFT. At the logon window, click "OK" to associate with the TM's project.

## Get the TM/RFT integration working after a failure

- Resolving corruption when there are two RFT projects with the same name:
  - ▶ Backup the project
  - ▶ Determine the current UID for the RFT project:
    - From the system folder, open the file **..\Project\Testdatastore\TMS\_TestScriptSources\project.TestScriptSource.rtxml**
    - Find the tag of <UID>
  - ▶ Determine the original UID for RFT project:
    - Go to the folder of a test case associated with an RFT script, such as: **..\Project\Testdatastore\TMS\_TestPlans\TestPlan1.TestPlan\Default.CaseF\RFT.CaseF**
    - Open the *testcase name.rtxml* file, and find the tag <AutoImplementation Name>.



If there are two RFT projects with the same name, resolve the corruption by first backing up the project. Next, determine the current UID for the RFT project. From the system folder, open the file listed on this slide, and find the tag of UID. Afterwards, determine the original UID for the RFT project. Go to the folder of a test case associated with an RFT script, such as the example file listed on this slide. Open the *testcase name.rtxml* file, and find the tag <AutoImplementation Name>.

## Get the TM/RFT integration working after a failure

- Resolving corruption when there are two RFT projects with the same name:
  - ▶ Perform a global search and replace the original UID with the current UID for all the files under the folder, “TMS\_TestPlans.”
  - ▶ Verify that the UIDs have been updated correctly.
  - ▶ Run Datastore Doctor to synchronize the database and the file system.

Perform a global search and replace the original UID with the current UID for all files under the folder, “TMS\_TestPlans.” Verify that the UIDs have been updated correctly. Finally, run Datastore Doctor to synchronize the database and the file system.

## Module summary

- This module covered:
  - ▶ The benefits of associating a TM project with an RFT project
  - ▶ How to associate a TM project with an RFT project
  - ▶ Integration failure causes
  - ▶ How to make the TM and RFT integration work again after a failure

In summary, this module discussed the benefits of associating a TM project with an RFT project, the steps to associate a TM project with an RFT project, what can cause the integration to fail, and the steps to make the TM and RFT integration work again after a failure.

## Feedback

### Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send e-mail feedback:

[mailto:iea@us.ibm.com?subject=Feedback\\_about\\_RTM\\_RFT\\_Integration.ppt](mailto:iea@us.ibm.com?subject=Feedback_about_RTM_RFT_Integration.ppt)

This module is also available in PDF format at: [../RTM\\_RFT\\_Integration.pdf](http://../RTM_RFT_Integration.pdf)



You can help improve the quality of IBM Education Assistant content by providing feedback.



## Trademarks, copyrights, and disclaimers

IBM, the IBM logo, ibm.com, and the following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:  
Rational

If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of other IBM trademarks is available on the Web at "Copyright and trademark information" at <http://www.ibm.com/legal/copytrade.shtml>

Rational is a trademark of International Business Machines Corporation and Rational Software Corporation in the United States, Other Countries, or both.

Java, and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. 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 in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

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

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2008. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.