Rational. Rhapsody

IBM.

IBM® Rational® Rhapsody® TestConductor Add On

Adapter for Rational Quality Manager - Howto

Rhapsody®

IBM® Rational® Rhapsody® TestConductor Add On

Adapter for Rational Quality Manager - Howto

Release 2.8.0

License Agreement

No part of this publication may be reproduced, transmitted, stored in a retrieval system, nor translated into any human or computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual or otherwise, without the prior written permission of the copyright owner, BTC Embedded Systems AG.

The information in this publication is subject to change without notice, and BTC Embedded Systems AG assumes no responsibility for any errors which may appear herein. No warranties, either expressed or implied, are made regarding Rhapsody software and its fitness for any particular purpose.

Trademarks

IBM® Rational® Rhapsody®, IBM® Rational® Rhapsody® Automatic Test Generation Add On, and IBM® Rational® Rhapsody® TestConductor Add On are registered trademarks of IBM Corporation.

All other product or company names mentioned herein may be trademarks or registered trademarks of their respective owners.

(1) Introduction

This document gives a short overview about the Rhapsody TestConductor Adapter for Rational Quality Manager.

It describes (shortly) how to configure and start the adapter (2), how to create TestConductor Tests in Rhapsody (3.I), how to create Test Cases (using the TestConductor Adapter) in RQM (3.II) and how to execute and evaluate these tests (4). Section (5) lists the available options for Test Scripts which influence the Test Case execution and the options needed options to access models on Design Manager. This document concludes with a troubleshooting and FAQ section (6).

The Rhapsody TestConductor Adapter for Rational Quality Manager supports version 4.x, 5.x and 6.x of the Rational Quality Manager.

(2) Configuring and Starting the RQM TestConductor Adapter

The TestConductor Adapter is located inside the folder `TestConductor/RQM_TestConductorAdapter` in your Rhapsody installation (per default in the user profile). The adapter can either be started directly from the command line (a) or by using the adapter settings dialog (b).

(a) Start from command line

The adapter is started via the file `start.bat`[1] analogous to the CommandLine Adapter shipped with RQM[2]:

```
start.bat -repository https://<rqmserver>:<port>/qm -user <userid>
-password <password> -adapter <adapter> \
     [-projectArea <project area>][-adapterName <adapter name>] \
     [-sleepTime <sleep time>] [-configFile <configuration file>]\
     [-proxy <proxy server>] [-proxyPort <port>] \
     [-proxyUser <proxy user>] [-proxyPassword <password>]

where:
     rqmserver: is the hostname/IP address of the RQM server
     port: is the port where the RQM server is running
     userid: is a registered user ID within RQM that has the license to
          run an adapter
     password: is the password of the userID used
     adapter: is a user given adapter identifier.  This value must be
          unique as it is used by the RQM server to identify this adapter.
     project area: is the name or alias of the project being logged into,
               default is "Quality Manager".  Surround with double quotes
               if the name contains spaces.
     adapter name: is a user given adapter name.
     sleep time: is the polling interval between polling for tasks, default
          is 30 seconds
     log file: the file to which to log the output messages, default is
               CommandLineAdapter.log
     configuration file: file to store and read the settings for this
          adapter, default is Config.ini
     proxy: is the hostname/IP address of the proxy server
```
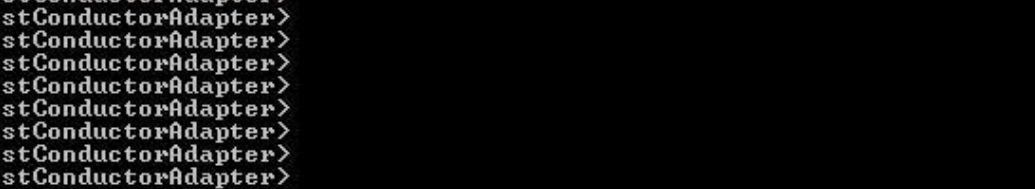
---

1   On Linux use `start.sh`
2   The following parameter description is extracted from the readme.txt of the original commandline adapter.

```
        proxy port: is the port where the proxy server is running
        proxy user: is a registered user within proxy server
        proxy password: is the password of the proxy user used
```

The adapter will generate a configuration file (Config2.ini file or the file specified via -configFile option) that will contain the registration information. This will be reused when the adapter is started the next time.

Fig. 1: An appropriate –dependent from your environment and configuration– call would be: "start.bat -repository https://qm.ibm.com:9443/qm -user testuser -password testpassword -adapter myTCAdapter -projectArea Project2"

Note that inside the RQM installation in the adapters folder there is a RQMCommandLineAdapter.zip file which contains a readme.txt file describing the mandatory and optional parameters in more detail.

If you want to start the adapter using the SP800-131A protocol, the `start.bat` contains an option USE_SP800 which is commented out by default.

(b) Start via settings dialog

Alternatively a settings dialog is available which provides the possibility to enter the required adapter settings data and to connect and to terminate the adapter. The status as well as the output of the adapter are shown in the GUI.

The settings dialog can be started via the file `startSettingsGUI.bat`. A configuration file *.ini can be specified as argument. In this case, the values defined in the configuration file are already entered in the appropriate setting fields (except the password). If no configuration file is given and the default configuration file Config2.ini exists, the values defined in this file are used.

The following settings can be specified in the adapter settings dialog.

- Server URL: https://<rqmserver>:<port>/qm
- User ID: Registered RQM user
- User Password: Password of the given RQM user
- Adapter ID: User given adapter identifier which value must be unique as it is used by the RQM server to identify this adapter
- Project Area: Name of the project being logged into
- Adapter Name: User given adapter name
- Sleep Time: Polling interval between polling for tasks

Fig. 2: The settings dialog for the TestConductor RQM adapter. Settings have been entered and the adapter has been connected.

The dialog also offers an option that the SP800-131A protocol should be used.

(3) Executing Rhapsody Testconductor Tests via RQM

To execute Rhapsody TestConductor Tests in RQM at first Tests have to be specified in a Rhapsody model using TestConductor (I). Then in Rational Quality Manager TestScripts can be created (II) which are linked via the TestConductor Adapter with the tests in the Rhapsody model. The TestScripts are again part of a (RQM) TestCase which can be executed in RQM.

I. Preparing the Tests (the Rhapsody Part)

(It is advisable to have a basic knowledge about Rhapsody and TestConductor before creating TestConductor tests. Both tools are shipped with documentation and tutorial which are accessible via the help tab of Rhapsody.)

1. At first you need a Rhapsody model with a design model which you want to test. Let 's assume you are using the sample model CppCashRegister
    (`Samples/CppSamples/TestConductor/CppCashRegister`).
2. Inside the model rightclick the class `CashRegister` in the package `CashRegisterPkg` and select "Create TestArchitecture" from the context menu.
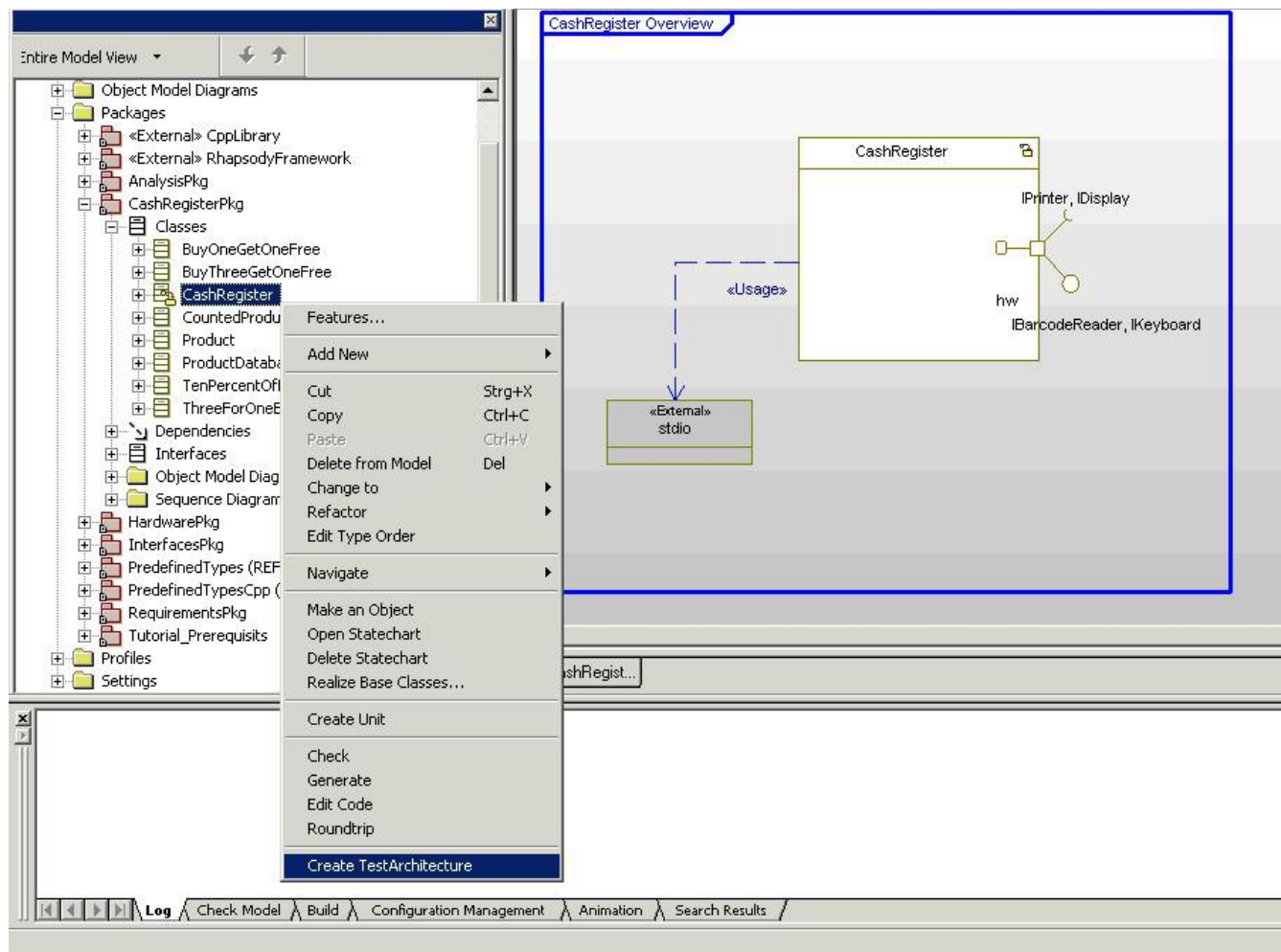

Fig. 3: Select "Create TestArchitecture" on class CashRegister

Then you have to agree to have the TestingProfile added to your model and a new

(Test)package "`Tpkg_CashRegister`" is created in the model.
3. Rightclick the TestContext "`TCon_CashRegister`" in the TestPackage and select "Create SDTestCase".
An SDTestCase is created. Now fill the TestScenario of the TestCase with content (e.g. Fig. 4).
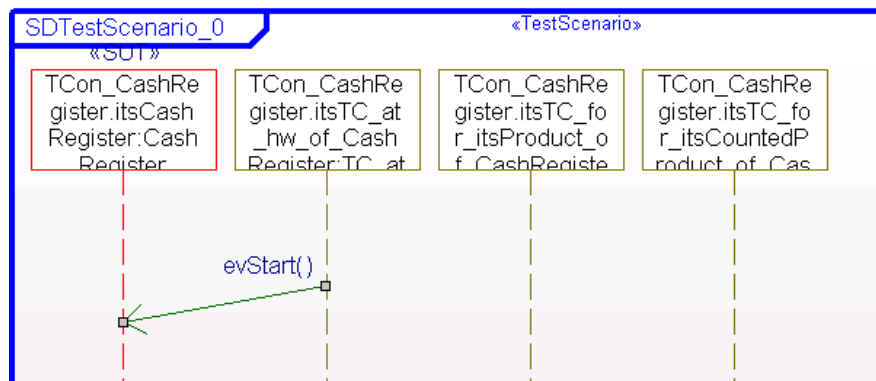

Fig. 4: A very simple TestScenario for a SDTestCase

4. Create more TestCases if you like.
5. Rightclick on the TestContext and select "Update TestContext" (the test model is populated with necessary driver operations)
6. Rightclick on the TestContext and select "Build TestContext" (the TestContext and the TestCases are built and afterwards ready to be executed).
7. If the preceding steps were successful save the model. Be sure remember the names of the TestCases (and TestContext and TestPackage) as you need these later to reference them from inside of RQM.

II. Preparing the Tests (the RQM Part)

(It is advisable to have basic knowledge about Rational Quality Manager integrating Rhapsody TestConductor Tests into RQM. RQM is shipped with online help files and sample files.)

1. Open the website `https://<rqm_server>:9443/qm` and log in with your user name and password (after initial install only the `ADMIN/ADMIN` exists, you have to create a user with the correct rights before able to work with RQM, please see the RQM documentation for further details)
2. Create a Test Script (Fig. 4)
3. Fill the Test Script (Fig. 5):
   • Specify a name for the Test Script
   • Switch the type of the Test Script from 'Manual' to 'Command Line'
   • Switch the test resources location from 'Shared Location' to 'Local to a test machine'
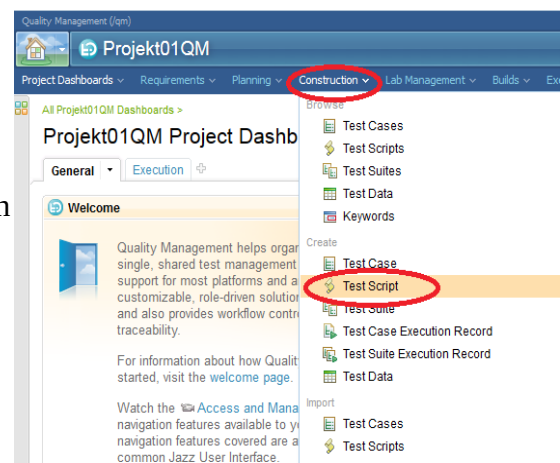

Fig. 4: Create a Test Script

Fig. 5: Specify Test Script details

- Optional: Select the adapter which should execute this Test Script by default (in this step also the model path is specified)
- Specify in the 'Command' text field the full path name of the Rhapsody model on the local machine
- Specify in the 'Arguments' field the full path name of the test element (TestCase, TestContext, TestPackage) which is to be executed. This must be the fully qualified name of the element, which consists of every parent element in hierarchical order and the test element itself, concatenated with double colons. (for example: TPkg_object_1::TCon_object_1_Architecture::TCon_object_1::SD_tc_0)
- After the test element name further options may follow. A list of supported options including particularities when involving Design Manager can be found in Appendix A.

4. In the next step you create a Test Case (Fig. 6) and connect it with the previously created Test Script (Fig. 7):
   - Create and name the Test Case
   - Click on 'Test Script' on the left panel
   - Select the green '+' symbol ('Add Test Scripts') in the Test Scripts window.
   - Select the previously created Test Script
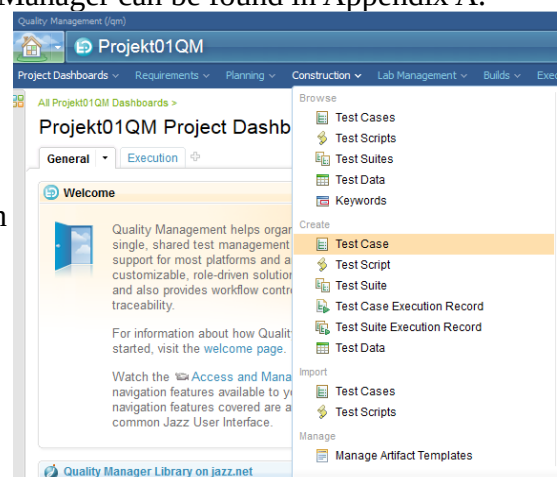   - Save the Test Case
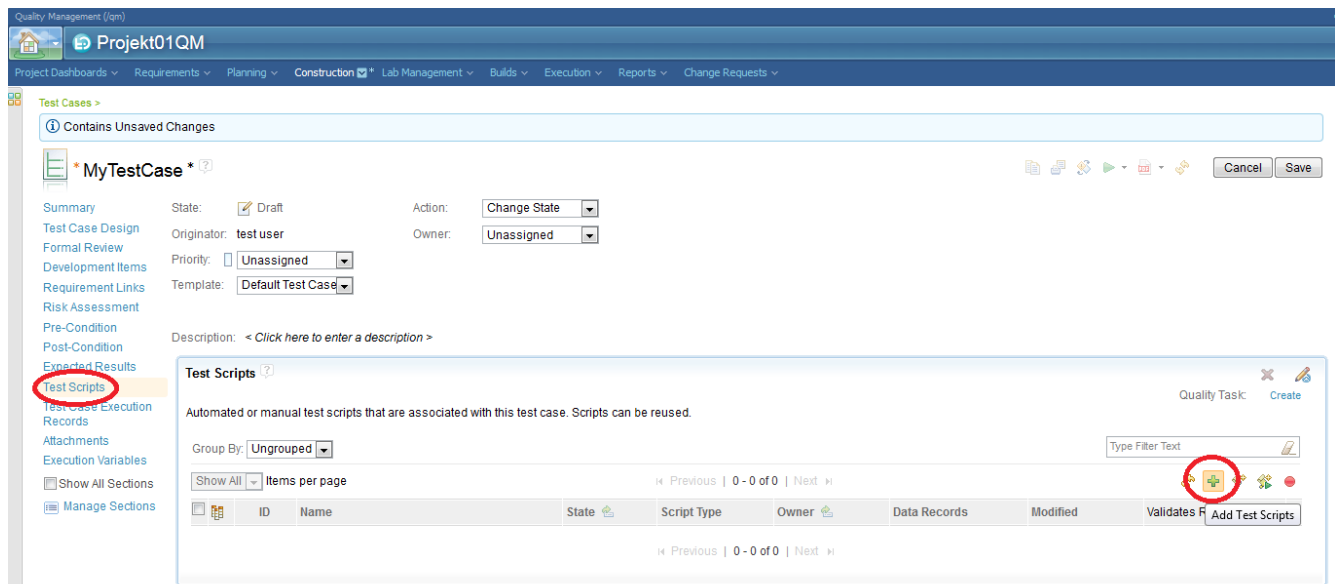

Fig. 6: Create a Test Case

Fig. 7: Connect the Test Script and the Test Case

(4) Executing and Evaluating

You can execute a (selected) Test Case (prepared as described in the preceding section) by clicking on the green triangle at the top tool bar of the Test Case (Fig. 8).
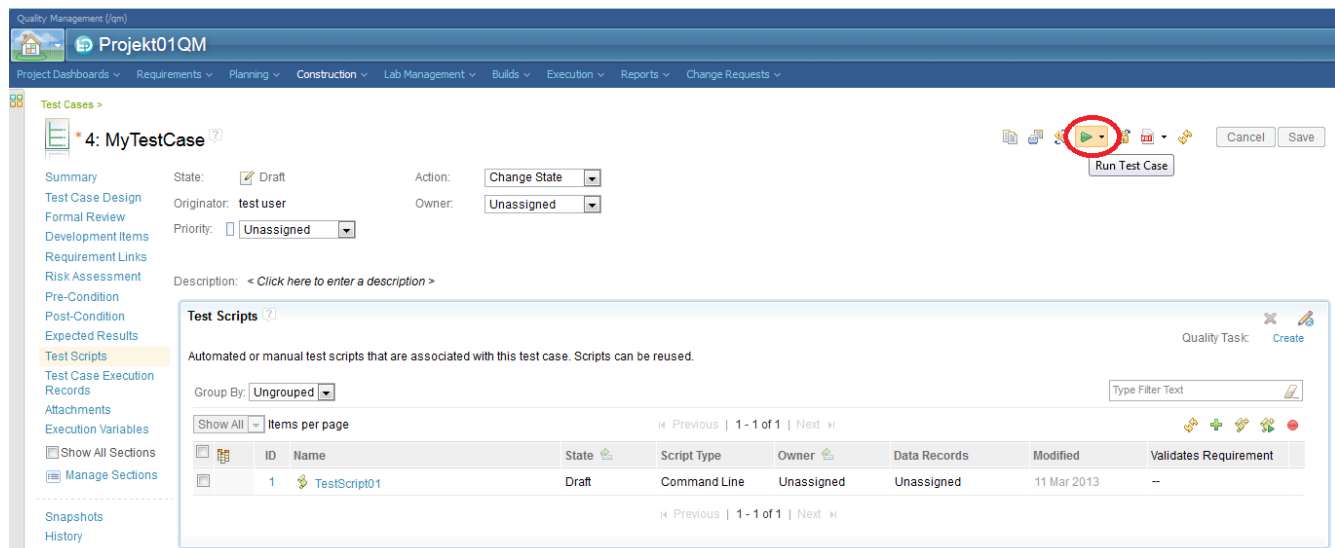

Fig. 8: Execute a Test Case

Before starting the test you have to chose or confirm the TestConductor adapter which is to be used for the execution. Then RQM contacts the corresponding adapter and in the background the adapter starts Rhapsody, loads the model, tries to resolve the specified test, updates, build and executes the test and imports the results into RQM.

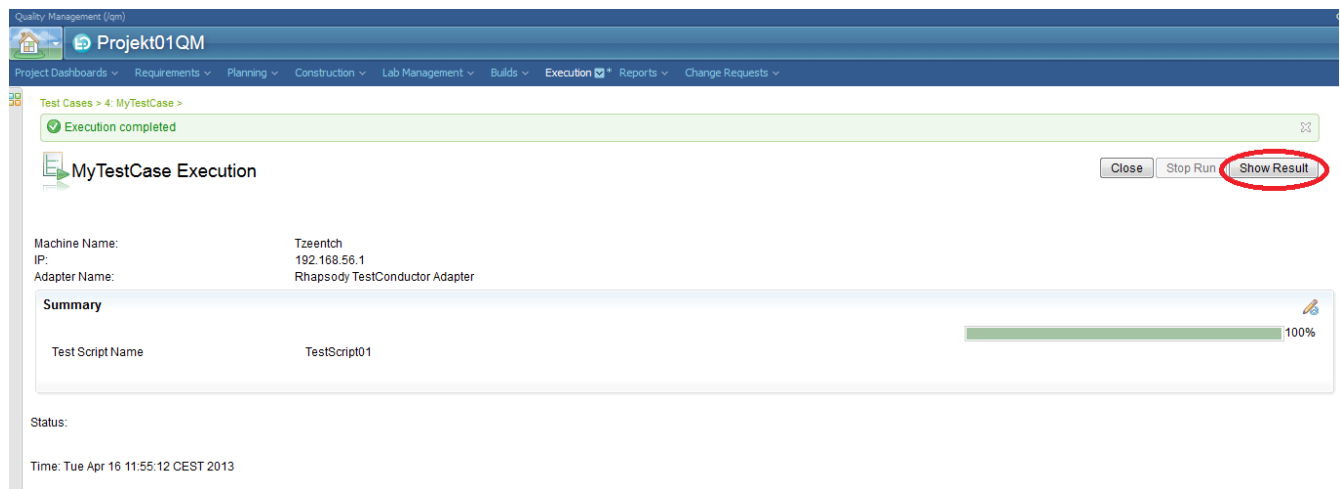After the test has finished, you can evaluate the results.

Fig. 9: Test execution has finished (without error), the results are ready to be reviewed.

If the test was executed successfully (result is passed or failed), there is a test result file in the result file section. If configured so, there are also a model coverage result file and a code coverage result report (as zip so it needs to be extracted). If an error occurred only log files are are imported, of this the file "`TestLog<some_number>.log`" is the most interesting since it contains the course of the test and also possible errors. All files can be shown by clicking on them.

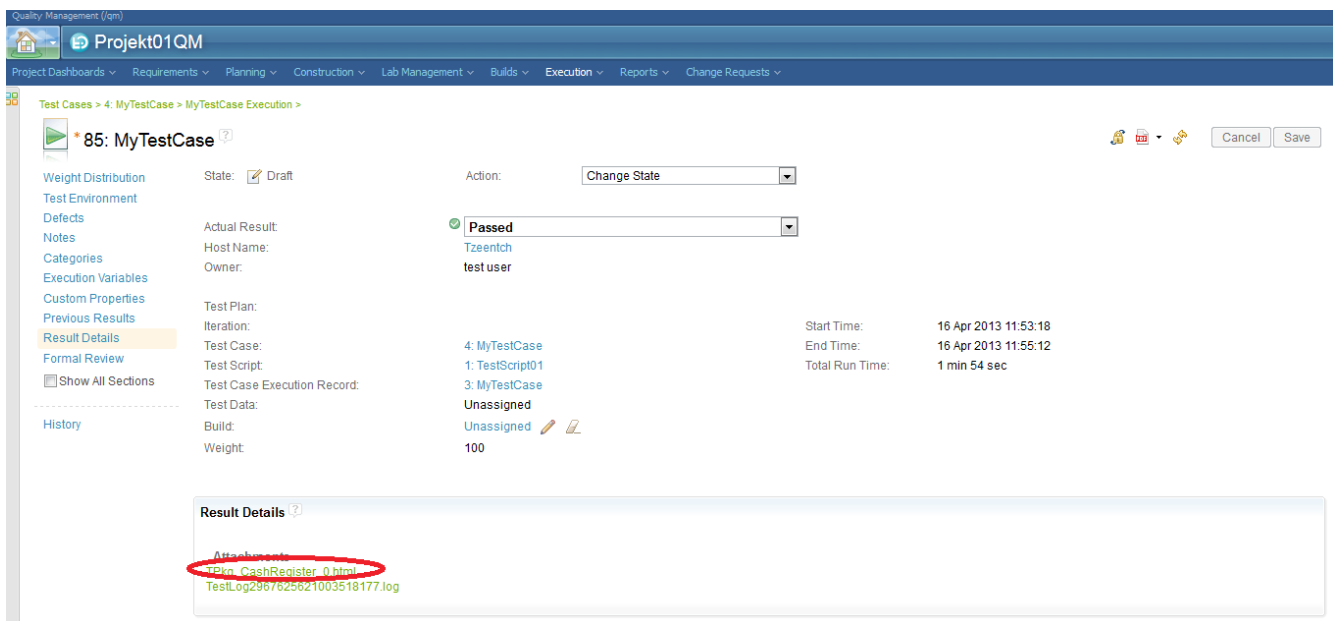*Fig. 10: The Test is passed.*

(5) Execution Options and Design Manager Options

I. Standard Options

You may specify additional options for the test in the arguments text field behind the test name or as execution variables of the test script.

Options supported in Argument Text Field and as Execution Variables

1. LANG=[Cpp|C|Java|Ada] (this setting enforces Rhapsody to be called with the corresponding language option)

2. HideUI=[false|true] (default is false, Rhapsody is started with visible user interface, set to true to hide Rhapsody's user interface during the test run)

3. GrabResultsOnly=[false|true] (default is false, causes the adapter only to import existing TestResults, without executing the test)

4. BuildBeforeExecute=[false|true] (default is true, causes the adapter only to execute the test without updating and building it)

5. CaptureRhapsody=[false|true] This option can be used to gain a speed up if you are executing several Test Cases. If set to true the adapter will overtake an existing Rhapody instance (instead of reporting an error) and will use this instance for the test execution. The Rhapody instance will not be closed after the test and can be used for following tests. Note that this mean that the LANG option has no effect and all Tests will be executed with the same initial Rhapsody instance until the Rhapsody instance has been manually shut down. Note that in this mode also the model is checked, if the name of the model is the same as the model to be loaded it will be reused. This is an important performance issue if the model is stored on Design Manager.

Options supported only as Execution Variable

1. Timeout (expects numeric value specifying the number seconds after which the test execution is to be canceled by the adapter.

II. Design Manager Options

If the mode is stored on Design Manager, only the model name is specified in the 'Command' field (instead of the full path name). The adapter identifies that the model is stored on a Design Manager server by the specification of the following options (in the arguments text field or as execution variables):

6. USER_NAME user name which is to be used to log in to the Design Manager server

7. PASSWORD password for the user specified by user_name to be used to log in to the Design Manager server

8. SERVER_URL the url of the Design Manager server (e.g. https://dm.mydomain.com:9443/dm)

9. PROJECT_AREA_NAME name of project area to be used

10. STREAM_NAME name of stream to be used (if omitted name of project area is also used for stream name)

Note that the Design Manager client extension has to be installed for the Rhapsody installation to be able to use the TestConductor adapter with Design Manager models.

(6) Troubleshooting

- If the **tests in RQM do not finish** or there is some other kind of weird behavior (I) please check in your task manager whether there are Rhapsody processes still active in the system. Since the adapter calls Rhapsody per default with hidden ui, the user does not see any Rhapsody instances which were wrongly not closed after a test run previously.
- If the **tests in RQM do not finish** or there is some other kind of weird behavior (II) please

check in your task manager whether there are unaccounted java processes still active in the system and if so terminate them (after verifying they are really unaccounted). (Currently a java process still remains active after terminating the adapter. This java process may interference with a newly started adapter)

- If you want to have (model) **coverage information** imported to RQM, be sure you have switched Model Coverage on in the TestCase properties (inside Rhapsody).
- **The adapter does not start, complaining about a missing or wrong entry of the Rhapsody API in the class path**: In the file `start.bat` check and adjust the value of the variable `RhapsodyPath` (line 7) according to your Rational Rhapsody installation.
- In the rare case of an ambiguity between a TestCase and a TestContext name (if a TestContext contains another nested TestContext with the same name of a TestCase inside the outer TestContext –this constellation is not advised), the adapter chooses the TestContext before the TestCase. If you want to execute the TestCase you can add "()" behind the Test's name, to indicate you mean the TestCase instead of the equally named TestContext.