Rational TestManager

A Test Management Tool
What is Rational TestManager?

• Rational Testmanager is a testing tool that encompasses all aspects of test analysis from test management to execution to reporting.

• It supports all kinds of testing from pure manual test approaches to various automated paradigms including unit testing, functional regression testing and performance testing.
Who uses Rational TestManager?

TestManager is designed to be used by all members of a project team.

- **Testers:** TestManager automates and simplifies crucial tasks. This lets the testers design all kinds of tests to run on the software.
- **Developers:** Helps developers keep track of what features have been tested, and ensure no requirement goes untested.
- **Managers:** Testing allows project managers to determine whether the system the team is building matches the design model and to keep track of deadlines.
Rational TestManager Workflow

- Planning tests
- Designing tests
- Implementing tests
- Executing tests
- Evaluating tests
Rational TestManager Workflow

1. Planning Tests

The activity of planning tests involves answering the following questions:

- **What and Where?** – Requirements, visual models, and other test inputs tell you what to test and where to run the tests.

- **Why** – Test inputs tell you why you are going to do certain tests. For example, tests may be performed to validate system requirements.

- **When?** – Iteration plans tell you when the tests must be run and must pass.

- **Who?** – Test plans, iteration plans, or project plans tell you who performs the testing activities.
Rational TestManager Workflow

2. Designing Tests

The activity of designing tests answers the question, “How am I going to perform the testing?”

A complete test design informs readers about what actions need to be taken with the system and what behaviors and characteristics they should expect to observe if the system is functioning properly.
Rational TestManager Workflow

3. Implementing Tests

• The activity of implementing tests involves the design and development of reusable test scripts that implement your test case.
• After you create the implementation, you can associate it with the test case.
Rational TestManager Workflow

4. Executing Tests

The activity of executing your tests involves running the test implementations to ensure that the system functions correctly.

In TestManager, you can run any of the following:

- An individual test script
- One or more test cases
- A suite, which runs any combination of test cases and test scripts across one or more computers and virtual testers.
Rational TestManager Workflow

5. Evaluating Tests

The activity of evaluating tests involves:

- Determining the validity of the actual test run. Did it complete? Did it fail because preconditions weren’t met?
- Analyzing the test output to determine the result. In performance testing, you look at reports on the generated data to see if the performance is acceptable.
- Looking at aggregate results to check coverage against test plans, test inputs, configurations, and so on. This can also be used to measure test progress and to do trend analysis.
Rational Robot Tutorial
Setup For Tutorial

- Creating a Rational Administrator Project
Create a Rational Administrator Project

• Start Rational Administrator by clicking:
  Start > Programs > Rational Software > Rational Administrator
Create a Rational Administrator Project

- Create a new Project: Choose File > New Project
- Enter a project name (TestManagerTutorial) and path (C:\TestManagerTutorial)
Create a Rational Administrator Project

- Click okay on the warning window and do not set a password, just click next.
- Ensure that **Configure Project Now** is checked on the summary page.
Create a Rational Administrator Project

- To configure the test datastore that will be part of your project, click Create... in the Test Assets group.

- For this Tutorial, MS Access will be sufficient, however for projects involving more than one person, Rational recommends using Sybase SQL Anywhere.

- Accept the default and close Rational Administrator.
Overview of Sample Application

A Quick tour of Classics Online
A Quick tour of Classics Online

• Classics Online is a sample application we will be using to plan and run our Tests on.

• You need not have the sample installed for this tutorial, but if you want to, you can install it by clicking
  Start > Programs > Rational Software > Rational Test > Set up Rational Test Samples
A Quick tour of Classics Online

- Classics Online is a simulation of an online store where you can buy classic CDs.
- (Start Classics Online by clicking: Start > Rational Test Samples > ClassicsA)
- As soon as you start the application, a log in screen is displayed.

After logging in, the following Main Window is brought up:

![Image of the Classics Online Main Window]
A Quick tour of Classics Online

- If you select a CD and click the "Press here to order" button, the following Order Window is displayed:

  ![Make An Order Window](image)

  - Sub-Total: $16.99
  - S+H: $2.00
  - Total: $18.99

- After placing an order, you can view a summary of your orders from the main window by clicking Order > View Existing Order Status:

  ![View Existing Orders Window](image)
Rational TestManager

Creating The Test Plan
Creating The Test Plan
Specifying Inputs to Test Inputs

- Rational TestManager can retrieve Requirements from both Microsoft Excel as well as Rational RequisitePro. As implementing requirements through RequisitePro is beyond the scope of this tutorial, we will be using Excel.

- Open up Microsoft Excel and create 4 columns called “Requirement Name”, “Description”, “Status” and “Last Modified” and set the formatting on the last column to a format supporting Data and Time. Fill up the table as shown in the screenshot below.

![Excel Table Example]

- Save this Excel document as TestManagerTutorial_Requirement in the same location as the project (C:\TestManagerTutorial)
Creating The Test Plan
What is a Test Plan

- A test plan is the method of organizing and creating the test cases in Rational TestManager. A test plan can be created stand-alone, i.e., without reference to test inputs, or starting with test inputs and creating test cases for each input.
- The advantage of linking test cases and requirements (test inputs) is that we have the ability to report on test progress against the requirements.
Creating The Test Plan
Starting Rational TestManager

- Start Rational TestManager by clicking “Start > Programs > Rational Software > Rational TestManager”

- You must log into a Rational Administrator Project to continue. Select the “TestManagerTutorial” Project we created earlier. Projects are created with Admin user with a blank password. For this tutorial this will suffice, however on actual projects you will want to create a username and password for each member of the team. Just click “OK” to use the Admin username.
Creating The Test Plan
Creating The Test Plan Structure

- When TestManager opens, click “Plans” on the bottom of left pane.
- Expand the Test Plans folder and right click “Test Plan 1” and rename it to “Classics Online”
- Double click on the Test Plan to open it
- Right click on the default folder to delete it
Creating The Test Plan
Creating The Test Plan Structure

• Right click on the Test Plan name and click “Insert Test Case folder”
• Name the folder “Verify Customer Name”, enter a description and click okay

• Repeat the previous steps to create the following Test Plan Structure
Creating The Test Plan
Connecting to the Excel Spreadsheet

- Since we used Excel to document our requirements, we need to establish the test inputs from TestManager.
- Click Tools > Manage > Test Input Types
- Select Microsoft Excel and click Edit
- Click the Sources Tab and Click Insert
Creating The Test Plan
Connecting to the Excel Spreadsheet

- In the New Input Source window, type the name Application Requirements.
- Click the Connection Data tab and in the Data Path field, click browse, and browse to the Excel Spreadsheet we created earlier.
- Click the “Set Configuration” button and click yes on the Save dialog box that pops up.
- Fill the dialog window with the values shown below and press okay.

![New Test Input Source](image1)

![Set Configuration](image2)
Creating The Test Plan
Viewing the Test Inputs

- To view the test inputs, Click View > Test Inputs.
- Click the “+” next to the Application Requirements to view the requirements.
- You can right click on a requirement and click properties to view its details.
Creating The Test Plan

Creating the Test Cases

- Test Cases hold information about the tests to execute. Right click on the “Customer Name” requirement and click “Insert Test Case”.

- Type in a name (Customer Name Verification) and select the “Verify Customer Name” Test Case Folder.

- Ensure that the “Edit Properties Now” checkbox is checked and click OK.
Creating The Test Plan
Creating the Test Cases

• Enter a brief description of the test case and click OK to save the test case.

• The test case opens beneath the requirement. Repeat this procedure for the other two requirements to have something like the following screenshot.
Rational TestManager

Creating a Test Script
Creating the Test Script
Test Implementation Type

- TestManager allows you to attach a whole array of Test Scripts to your test cases.
- Test Scripts can be automated scripts written in Rational Robot or Functional tester, or even custom tests written in Java, Visual Basic, or any language that can be executed from the command line.
- Test Scripts can also be Manual scripts that contain a description of steps to perform and verification points that determine whether the test passed or failed.
- We will be using Rational ManualTest for this tutorial.
Creating the Test Script

Manual Test Script


- Type in the steps required to verify the UserID in Classics Online. (Refer to the screenshot below)
- When a step ends with a “?” it is automatically converted to a verification point.
- To change type manually, click on the type field.

- Steps/Procedures are indicated with black text with a shoe-step icon on the “Type” column, and Verification points have blue text with a check mark on the Type Column.
Creating the Test Script

Manual Test Script

• Save the script by clicking File > Save and enter the name as “Customer Name Verification”
• Now write a new script (like the one shown below), and save it as “Order Status Verification”
Creating the Test Script
Implementing the Test Case

• Now, we can attach or Test Script to a test case. This is known as implementing the Test Case.
• Open the Test Plan Window again
• Right click on the Test Case “Customer Name Verification” and select properties.

• Click on the Implementation Tab and click Select in the Manual Implementation group.
Creating the Test Script
Implementing the Test Case

• In the Select Script window, click on the “Customer Name Verification” script and click OK.

• Now repeat the same for the “Order Status Verification” script.

• You should end up with the Test Plan window looking like the following screenshot.

• The “pointing finger” icon refers to a manual script attached to the test case, while a “gear” icon shows an automated test script.
Rational TestManager

Executing the Test Script
Executing the Test Script
Single Test Case Execution

- You can run a single test script by itself. This is good for a quick execution of a small number of test cases.
- (Do not do this now) You can do this by simply right clicking a test case and clicking Run…
- However, for a more efficient solution to testing your application, you can create a “Test Suite” that will run tests for multiple Test Cases
Executing the Test Script
Creating a Test Suite

• To create a Test Suite, click the “Execution” tab on the left pane.
• Right click on the Suites folder and click “New Suite…”

On the window that opens, select “Functional Testing Wizard” and click OK.

• Click the Select button on the Step 1 Window and on the Test Case Selection Window, select our to test cases with scripts attached to them and click “OK”
Executing the Test Script
Creating a Test Suite

- You should come back to the Step 1 Window looking like the following:

  ![Functional Suite Wizard - Step 1](image)

  - Click “Next” on the Step 2 window as we do not need to attach any stand alone scripts.
  - Click “Finish” on the Step 3 window. The script will be created as follows:

    ![Suite 1](image)

    - Click File > Save and enter the name “Classic Online Test Suite”
Executing the Test Script
Running the Test Suite

- To run the test suite, click on the toolbar button that looks like a running man.

- The following window opens:

- Click the “Options” button.

- Make sure the options are set according to the following screen-shot.

- Click “OK” on the Options window and on the Test Suite window to begin running the suite.
Executing the Test Script
Running the Manual Test

• This brings up the first ManualTest script.

• Normally, we would perform each step with the program to be tested, marking each verification as pass or failed, but for our tutorial, let us simple pretend we have actually done each step.
Executing the Test Script

Running the Manual Test

- On the First ManualTest script, mark each step with a check mark under results, and the verification point as Pass as shown and click done.
Executing the Test Script
Running the Manual Test

- On the Second Manual Test script, mark each step with a check mark under results, but mark the verification point as **Fail** as shown.
- Under Result Details, fill out a detail like “The order history did not match the order”. Click “Done”

- This will bring up the Test Log with the results displayed.
Executing the Test Script
Analyzing Test Results

- The Test Log with the result of the test will come up.
- Since one of our tests passed and the other failed, this will be reflected in the Test Log.
- Even if a single verification point fails in a test, the entire test will reflect as Failed.
Executing the Test Script
Analyzing Test Results

- Click the Details tab at the bottom of the Log.
- You can expand the test cases to see the steps and exactly which Verification point failed.
- You can also right click on a failed verification point and click properties to see the failure details.
Executing the Test Script

Analyzing Test Results

- After reviewing the test results, click the “Test case Results” tab at the bottom of the Log.
- We can now promote the results for reporting. To do this check the “Promoted” check boxes.
- Close the Test Log, and when prompted to save the test case results, select “Yes.”
Report & Monitor Test Progress
Predefined Reports

- TestManager has several predefined reports that can be run to keep track of testing progress and results.

- To see a list of predefined reports available, click the Analysis tab on the Left Pane.
Report & Monitor Test Progress

Test Distribution Report

- Expand the folder called “Test Case Distribution”
- These reports provide information on test progress against test inputs or the test plan.
- Right click on the report titled “Test Input Development Coverage” and select Run

- This will bring up the window shown on the bottom right.
- This report shows the requirements we wrote earlier along with the number of Test Cases for each requirement and the number of these that have been implemented, ie have a Test Script associated with it.
Report & Monitor Test Progress
Test Case Results Report

- Expand the folder called “Test Case Results Distribution”
- These reports provide information on results of the Test Cases we defined.
- Right click on the report titled “Test Plan Execution Coverage” and select Run

- In the Select Test Logs window, expand until you see our Test Suite “Classics Online Test Suite” and move it to the Selected Test Logs group.
Report & Monitor Test Progress
Test Case Results Report

• The following report will be generated for you.

• This report shows you the number of Test Cases implemented in the particular suite as well as whether the passed or failed.

• You can also create custom reports to generate reports according to the requirements of your specific project.
Summary

This tutorial walked you through some of the basic functionalities of TestManager and showed you how to:

- Plan a Test from Requirements (Test Inputs)
- Create a ManualTest Script
- Implement a Test Case by associating a script
- Executing both single tests as well as entire test suites
- Analyse the results of the test using the Test Log.
- Create reports to monitor Test Progress and Results