[Skip to content](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html" \l "container)

* [*Safari*](https://www.safaribooksonline.com/home/)
* [Recommended](https://www.safaribooksonline.com/r/)
* [Queue](https://www.safaribooksonline.com/s/)
* [Search](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html)
* [Expand Nav](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html)
  + [Recent](https://www.safaribooksonline.com/recent/)
  + [Topics](https://www.safaribooksonline.com/t/)
  + [Highlights](https://www.safaribooksonline.com/u/p253751/)
  + [Settings](https://www.safaribooksonline.com/u/)
  + [Feedback](mailto:feedback@safaribooksonline.com)
  + [Sign Out](https://www.safaribooksonline.com/accounts/logout/)
  + [Settings](https://www.safaribooksonline.com/u/)**10** days left in your trial. [Subscribe](https://www.safaribooksonline.com/subscribe/).
  + [Feedback](mailto:feedback@safaribooksonline.com)
  + [Sign Out](https://www.safaribooksonline.com/accounts/logout/)

[Software Test Engineering with IBM Rational Functional Tester: The Definitive Resource](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html)

* [Search in book...](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html)

Top of Form

Bottom of Form

* [Toggle Font Controls](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html)
* [Share this](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html)
  + [Twitter](https://twitter.com/share?url=http://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch04.html&text=Software%20Test%20Engineering%20with%20IBM%20Rational%20Functional%20Tester%3A%20The%20Definitive%20Resource&via=safari)
  + [Facebook](https://www.facebook.com/sharer/sharer.php?u=http://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch04.html)
  + [Google Plus](https://plus.google.com/share?url=http://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch04.html)
  + [Email](mailto:?subject=Safari:%20Chapter%204.%20XML%20and%20Rational%20Functional%20Tester&body=http://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch04.html%0D%0Afrom%20Software%20Test%20Engineering%20with%20IBM%20Rational%20Functional%20Tester%3A%20The%20Definitive%20Resource%0D%0A)

[Prev](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch14.html)

[Chapter 14. Developing Scripts in the VB.NET Environment](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch14.html)

[Next](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch16.html)

[Chapter 16. Internationalized Testing with Rational Functional Tester](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch16.html)

**Chapter 15. Using Rational Functional Tester in a Linux Environment**

**Chip Davis, Lee B. Thomas, Marc van Lint**

*This chapter describes how to use Rational Functional Tester (RFT) on a Linux system. There is one significant difference: There is no “Red button” to record. This chapter describes the steps to install Rational Functional Tester and create a basic script. For details on all other topics, use the Java examples that are provided throughout this book.*

*Although this chapter uses the term “Linux,” IBM does not provide support for using Rational Functional Tester on all Linux distributions. At the time of this writing, particular versions of Red Hat Enterprise Linux and SUSE Linux (SLES and SLED) are supported. This chapter shows examples for Red Hat Enterprise Linux.*

**Installation**

You should already have installed a Linux distribution that is a supported operating system (OS) for Rational Functional Tester. In that installation, you need to proceed with the following steps:

**1.** Install additional Linux software required by Rational Functional Tester.

**2.** Install Java and enable it in Firefox.

**3.** Install IBM Installation Manager.

**4.** Install Rational Functional Tester via IBM Installation Manager.

All these actions must be performed with root access.

**Installing Additional Linux Software Required by Rational Functional Tester**

A full and up to date description of the components needed on the Linux operating system can be found in the installation documentation for Rational Functional Tester. Still, there are two components that require attention:

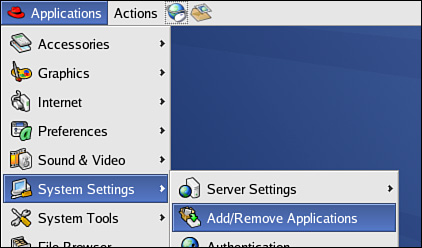
• Graphical Internet

• Legacy Software Development

To add these components, follow these steps:

**1.** Select **System Settings > Add/Remove Applications**. See [Figure 15.1](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig01).

**Figure 15.1** The Linux Add/Remove programs



**2.** Select the **Graphical Internet** group and select **Details**. See [Figure 15.2](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig02).

**Figure 15.2** Graphical Internet group in the list of optional components

image

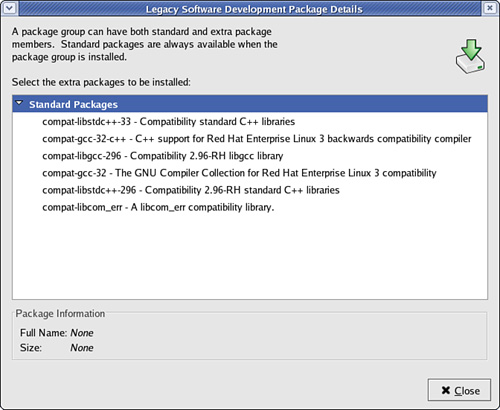
**3.** Confirm that the Firefox browser is installed. See [Figure 15.3](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig03).

**Figure 15.3** Legacy Software Development group

image

**4.** Confirm that the GNU Image Manipulation Program Toolkit (GTK+), Version 2.2.1 or later, and its associated libraries (Glib and Pango) are installed. See [Figure 15.4](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig04).

**Figure 15.4** The needed compatibility libraries for Rational Functional Tester



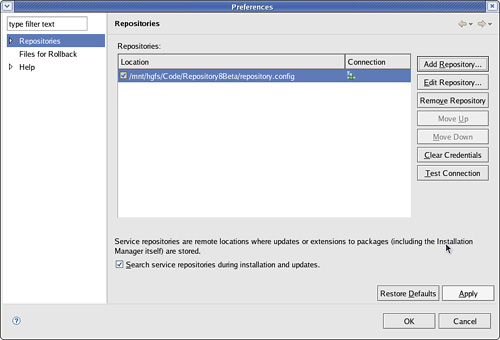
**Installing Java and Enabling Java in Firefox**

Rational Functional Tester requires a Java Runtime Environment (JRE), which can be obtained from [www.java.com](http://www.java.com/). You should refer to the instructions on that site for how to download and install the JRE, and refer to your Rational Functional Tester online documentation for version compatibility.

**Installing IBM Installation Manager**

Eclipse-based Rational products are installed from IBM Installation Manager (IIM). IBM Installation Manager takes care of any dependencies among packages to be installed and the environment already available. IBM Installation Manager operates in the same manner on both Windows and Linux. You enter a Linux path to identify the location of the repository, as shown in [Figure 15.5](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig05).

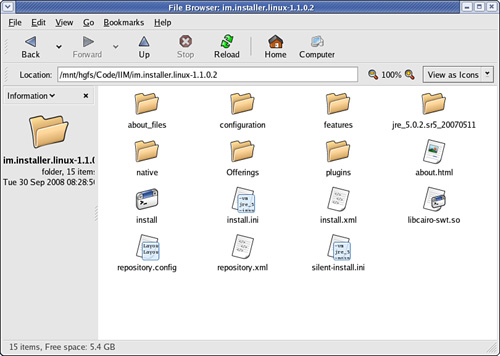
**Figure 15.5** The directory where the code for IIM is available



To install the IBM Installation Manager, follow these steps:

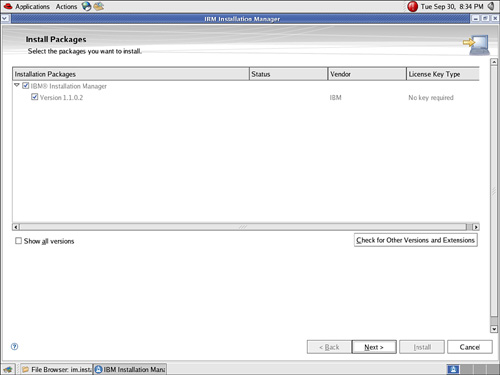
**1.** Double-click the **install icon**, as shown in [Figure 15.6](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig06).

**Figure 15.6** The available installation manager



**2.** Select **Next**. See [Figure 15.7](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig07).

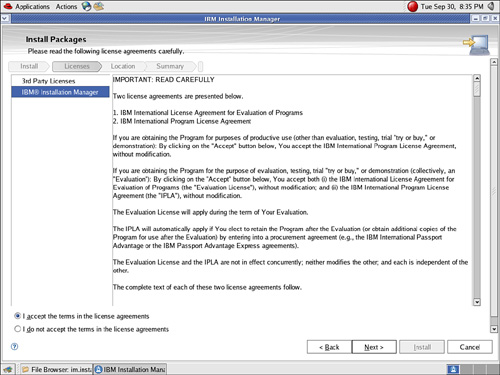
**Figure 15.7** Accept installation of IBM Installation Manager.



**3.** Select the **I accept the terms in the license agreements** radio button.

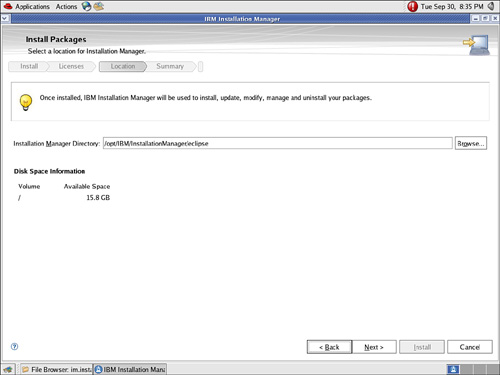
**4.** Select **Next**. See [Figure 15.8](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig08).

**Figure 15.8** Accept the terms and conditions before IBM Installation Manager.



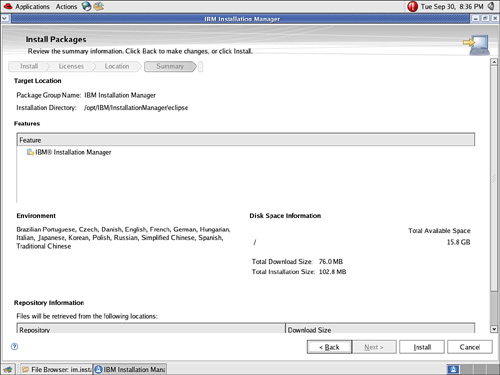
**5.** Assuming this default directory is acceptable, Select **Next**. See [Figure 15.9](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig09).

**Figure 15.9** A check on the settings for the installation of IBM Installation Manager



**6.** Select **Install**. See [Figure 15.10](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig10).

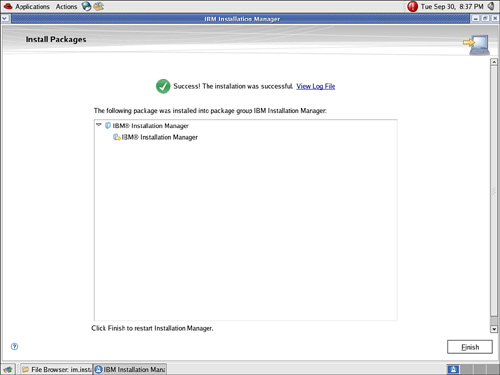
**Figure 15.10** Check for settings and install.



**7.** Select **Finish**.

You can start IBM Installation Manager from the menu. Select **Applications > System Tools > IBM Installation Manger**, as shown in [Figure 15.11](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig11).

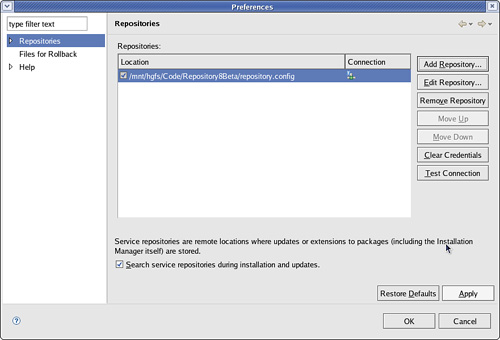
**Figure 15.11** A successful IBM Installation Manager installation



**Installing Rational Functional Tester**

With IBM Installation Manager installed, you can install Rational Functional Tester. Define the location of the repository of the code to be used. Select **File > Preferences**. See [Figure 15.12](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig12).

**Figure 15.12** The definition of the source of the installation files



**Using Rational Functional Tester**

This section gives you the basic steps for creating an automated test script in Rational Functional Tester in a Linux environment:

• [Start Rational Functional Tester](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15sec2lev5).

• [Enable environments for testing](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15sec2lev6).

• [Configure applications for testing](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15sec2lev7).

• [Create an empty script](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15sec2lev8).

• [Insert a Test Object](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15sec2lev9).

• [Insert a Verification Test Object](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15sec2lev10).

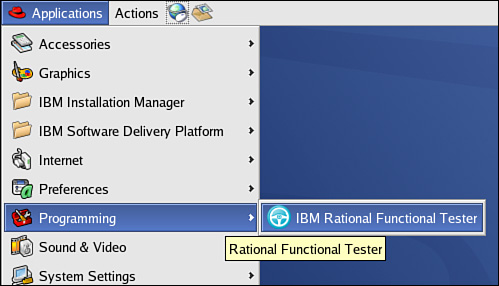
• [Insert waitForExistence](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15sec2lev11).

• [Create an entire script](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15sec2lev12).

**Starting Rational Functional Tester**

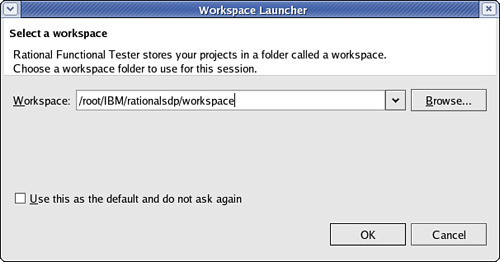
**1.** Start Rational Functional Tester by selecting **Applications > Programming > IBM Rational Functional Tester**. See [Figure 15.13](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig13).

**Figure 15.13** Starting Rational Functional Tester



**2.** Rational Functional Tester prompts for the location of the workspace, which is the area where temporary settings are stored. See [Figure 15.14](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig14).

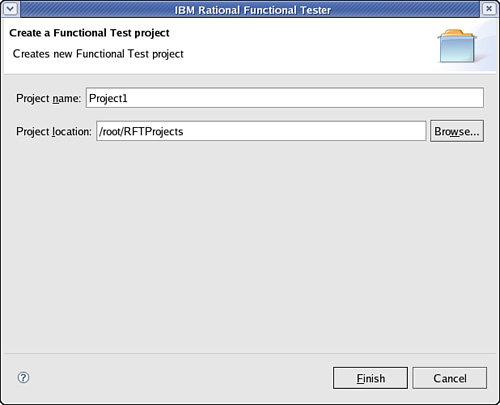
**Figure 15.14** Defining the workspace



**3.** If it appears, you can close the Welcome view. However, it is highly recommended that you return to it by the menu **Help > Welcome**, to review the material there. If you do not see the Functional Test perspective at this point, open it via **Window > Perspective > Other > Functional Test.**

**4.** Create a project, which is the area where all artifacts are stored. Select **File > New > Functional Test Project**. See [Figure 15.15](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig15).

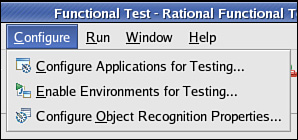
**Figure 15.15** Creating the project



**Enabling Environments for Testing**

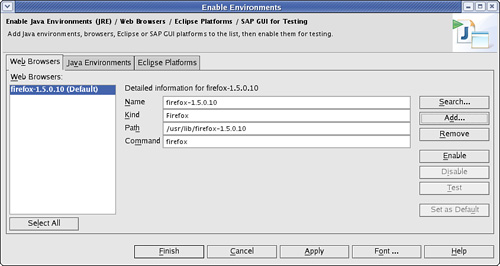
You have to set your environments or at least check to see if they are set correctly. See [Figure 15.16](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig16).

**Figure 15.16** The configuration of applications and environments



**1.** If you are testing a browser-based application, you need to define the browsers used. See [Figure 15.17](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig17).

**Figure 15.17** Define the browsers that can be used.

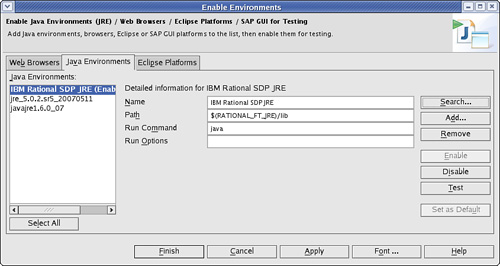


**2.** Select the **Search** option to automatically search for all browsers.

**3.** Select the **Search All** and **Search** to add any browsers.

**4.** Select the **Java Environments** tab. See [Figure 15.18](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig18).

**Figure 15.18** Define the Java environments that can be used.



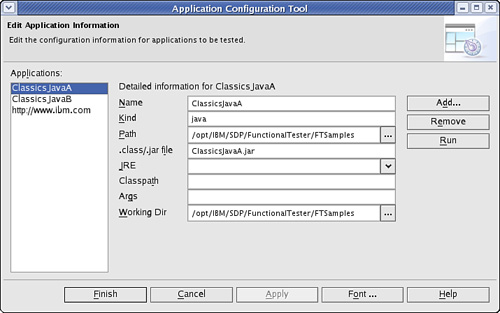
**5.** Select the **Search** option again.

**6.** Enable the appropriate Java Runtime and make it the default.

**Configuring Applications for Testing**

In Rational Functional Tester, you can define the applications under test. This can be done via the menu **Configure > Configure Applications for Testing**. You can add the application to test. For this exercise, you use the Java application ClassicsJavaA and the browser-based application [www.ibm.com](http://www.ibm.com/). See [Figure 15.19](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig19).

**Figure 15.19** Define the application under test.

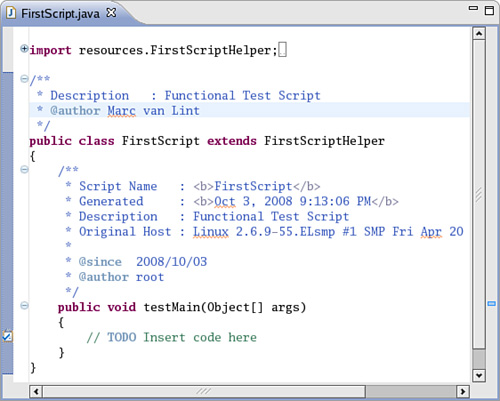


**Creating an Empty Script**

In the Linux version of IBM Rational Functional Tester, there is no “Record” button available. You create the script by entering code into an empty script.

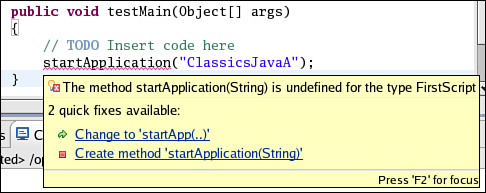
**1.** Create an empty script by selecting the menu item **File > New**. See [Figure 15.20](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig20).

**Figure 15.20** An empty script



**2.** To gain familiarity with the Eclipse-editing features, deliberately add an erroneous startApplication statement. If you make a coding error, the syntax and logic checking of Eclipse assist you. You can immediately see that the statement is incorrect if you look at the highlighting feature. Rational Functional Tester proposes quick fixes to correct the error. See [Figure 15.21](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig21).

**Figure 15.21** An example of assistance in correcting errors



The majority of statements in a test script consist of two parts: identifying the object to be acted upon and specifying the action to perform. In more detail, the parts are:

**1. Insert an object**—Points to the object in the application under test. This results in an entry in the Test Object Map, which enables actions against that object.

**2. Action on an object**—The object in the script complemented with an action such as a click action.

**Inserting a Test Object**

Creating a test script is done by inserting a Test Object from the Test Object Map into the script, and then selecting a method for the action, such as click(). This paragraph describes the steps to add a Test Object to the Test Object Map.

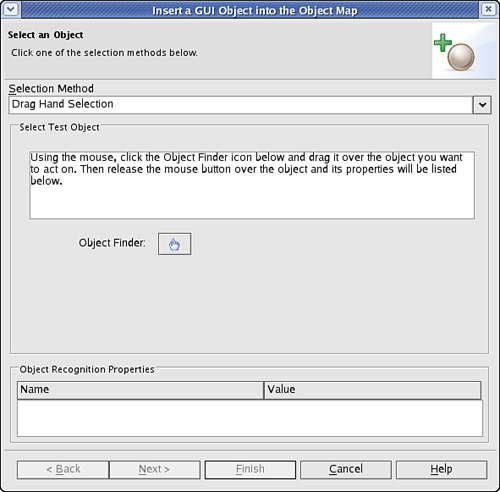
**1.** The Test Objects are added with the Insert Test Object icon. See [Figure 15.22](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig22).

**Figure 15.22** Insert Test Object.

image

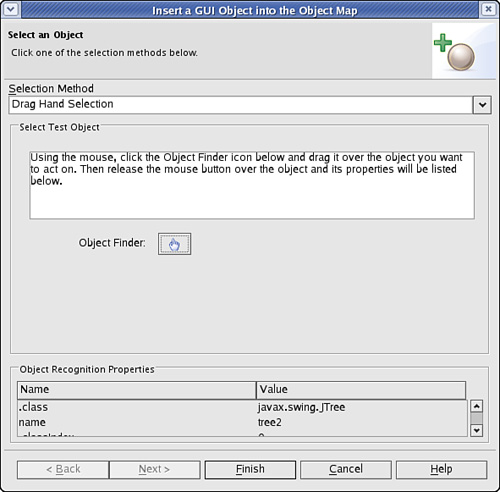
**2.** Drag the hand cursor to the object to be activated (for example, a tree or a button). By releasing the hand cursor, you can select the object that is highlighted. See [Figure 15.23](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig23).

**Figure 15.23** Insert GUI object.



**3.** You can verify you’ve selected the correct object by investigating the Object Recognition Properties (see [Figure 15.24](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig24)). Select **Finish.**

**Figure 15.24** Verify the object recognition properties.



**4.** Reference the object in the test script. For example, you can enter the following:

placeOrder()

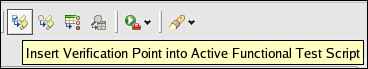
As you type, the Code Assist feature proposes the methods that are available. Methods such as click, focus, and so on are shown, and if you select one, you are shown the patterns for arguments that can be passed.

**Inserting a Verification Point**

You can create Verification Points much like you create actions on objects.

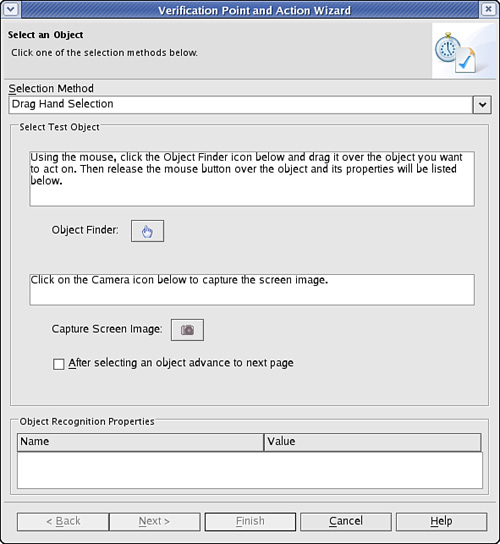
**1.** Place the cursor at the correct location in the script and select the **Insert Verification Point** icon. See [Figure 15.25](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig25).

**Figure 15.25** The Insert Verification Point icon



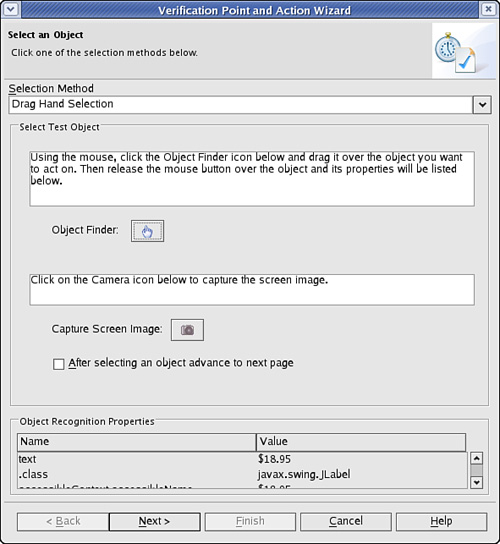
**2.** The Insert Verification Point Action Wizard displays, as shown in [Figure 15.26](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig26).

**Figure 15.26** The Insert Verification Point and Action Wizard



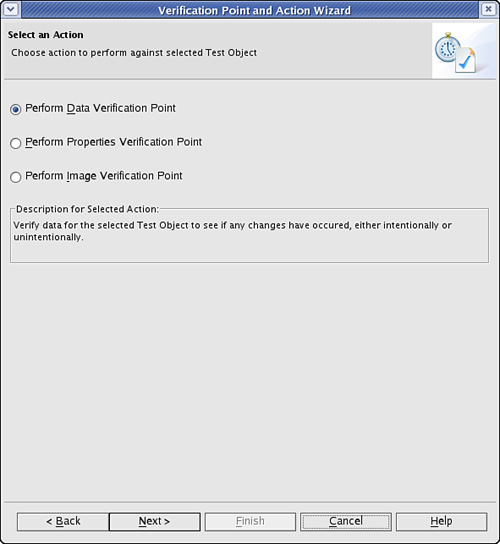
**3.** Drag the hand to the object to test by holding down the left mouse button. Rational Functional Tester draws a red square around the object that is selected when you release the mouse button. Selection is done when the cursor is released. See [Figure 15.27](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig27).

**Figure 15.27** The identified object can be verified by checking the object properties.



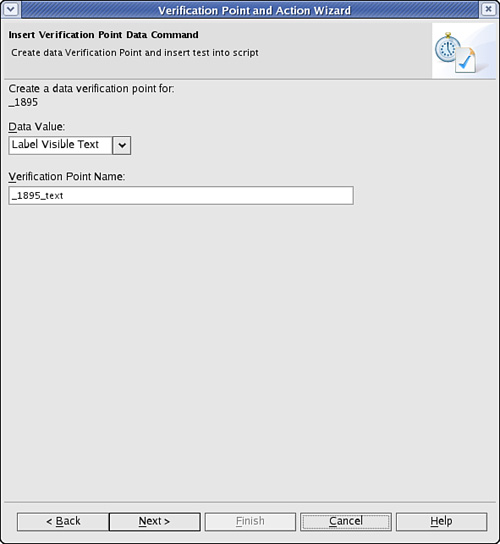
**4.** Select **Next,** or reselect the object to test. See [Figure 15.28](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig28).

**Figure 15.28** The various types of Verification Points. Note that waitForExistence is not available for Linux through this dialog.



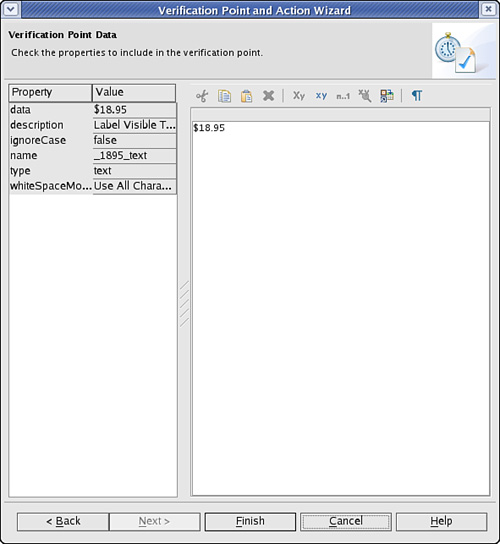
**5.** Select the appropriate type of test. Here a Data Verification Point is selected. See [Figure 15.29](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig29).

**Figure 15.29** The panel to give the Verification Point a sensible name. This name is not meaningful as a datapool reference.



**6.** Give the Verification Point an appropriate name and select **Next**. See [Figure 15.30](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig30).

**Figure 15.30** The Verification Point Wizard where the baseline of the Verification Point can be defined in hard-coded manner as a regular expression or as a datapool reference.



**7.** Select **Finish.** The following line is inserted into the code:

placeOrder().performTest( PlaceOrder\_textVP() );

**Inserting waitForExistence**

You can add waitForExistence for a Test Object by inserting the object, and adding the method via Eclipse’s Code Assist feature. The way to create is as follows:

**1.** Insert the Test Object into your script that Rational Functional Tester should wait for to ensure that it exists.

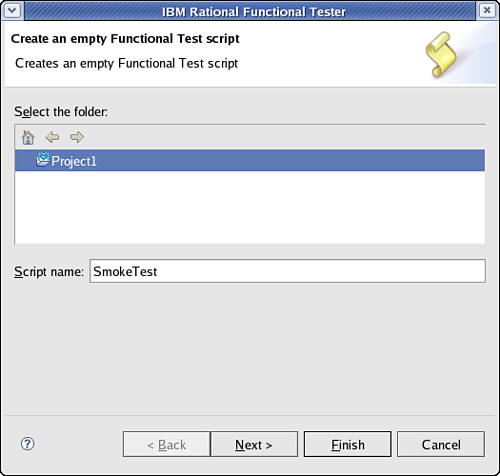
**2.** Add the .waitForExistence() method to the object’s reference.

**Creating an Entire Script**

On Linux, Rational Functional Tester does not provide a way for you to record your script. This section shows how you create a script from start to finish, using the ClassicsJavaA application.

**1.** Create an empty functional test script via the Rational Functional Test menu by selecting **File > New > Empty Functional Test Script**. See [Figure 15.31](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig31).

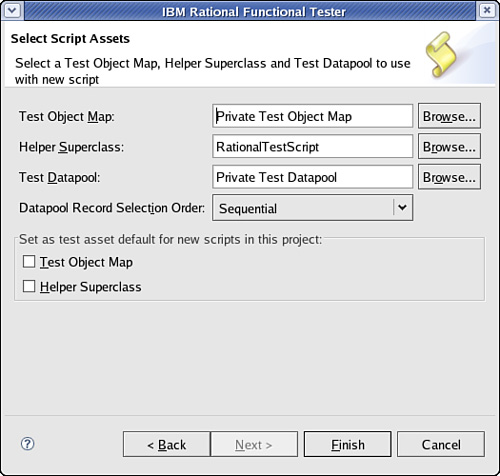
**Figure 15.31** Starting a new script



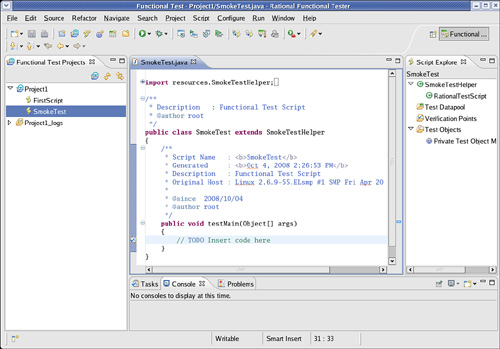
**2.** Enter a meaningful script name and Select **Next**.

**3.** Accept the script assets. Select **Finish.** See [Figure 15.32](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig32). The RFT main screen displays. See [Figure 15.33](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15fig33).

**Figure 15.32** Acceptance of default script assets

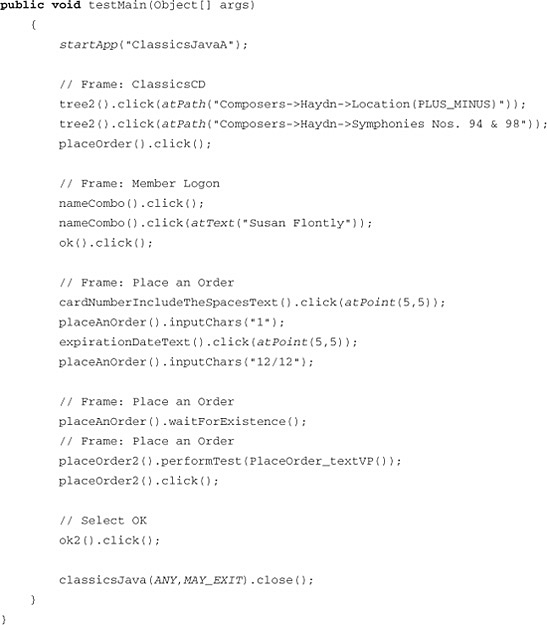


**Figure 15.33** Rational Functional Tester overview



**4.** Start your application from the Enable Application for Testing menu.

**5.** Type the following code, making use of Eclipse’s Code Assist feature to help you. Create the Test Objects and Verification Points as described in the sections “[Inserting a Test Object](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15sec2lev9)” and “[Inserting a Verification Point](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html#ch15sec2lev10)” earlier in this chapter.



**Summary**

This chapter showed how to use Rational Functional Tester to create automated scripts on a Linux system. The Code Assist feature of Eclipse and the many wizards in Rational Functional Tester make this a straightforward task.

[Prev](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch14.html)

[Chapter 14. Developing Scripts in the VB.NET Environment](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch14.html)

[Next](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch16.html)

[Chapter 16. Internationalized Testing with Rational Functional Tester](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch16.html)

**Welcome to Safari**. Remember, your free trial will end on March 7, 2015, but you can [subscribe at any time](https://www.safaribooksonline.com/subscribe/)

* [Recommended](https://www.safaribooksonline.com/r/)
* [Queue](https://www.safaribooksonline.com/s/)
* [Recent](https://www.safaribooksonline.com/recent/)
* [Topics](https://www.safaribooksonline.com/t/)
* [Settings](https://www.safaribooksonline.com/u/)
* [Blog](http://blog.safaribooksonline.com/)
* [Support](http://support.safaribooksonline.com/?prod=flow)
* [Support](http://msupport.safaribooksonline.com/?prod=flow)
* [Feedback](mailto:feedback@safaribooksonline.com)
* [Sign Out](https://www.safaribooksonline.com/accounts/logout/)

© 2015 [Safari](http://www.safaribooksonline.com/).   
[Terms of Service](https://www.safaribooksonline.com/terms/) / [Membership Agreement](https://www.safaribooksonline.com/membership-agreement/) / [Privacy Policy](https://www.safaribooksonline.com/privacy/)

[**Make font larger Make font smaller**](https://www.safaribooksonline.com/library/view/software-test-engineering/9780137036455/ch15.html)