

TestDirector

Requirements Synchronizer for Rational RequisitePro Add-in Guide

Version 8.0

TestDirector Requirements Synchronizer for Rational RequisitePro Add-in Guide, Version 8.0

This manual, and the accompanying software and other documentation, is protected by U.S. and international copyright laws, and may be used only in accordance with the accompanying license agreement. Features of the software, and of other products and services of Mercury Interactive Corporation, may be covered by one or more of the following patents: United States: 5,511,185; 5,657,438; 5,701,139; 5,870,559; 5,958,008; 5,974,572; 6,137,782; 6,138,157; 6,144,962; 6,205,122; 6,237,006; 6,341,310; 6,360,332; 6,449,739; 6,470,383; 6,477,483; 6,549,944; 6,560,564; 6,564,342; 6,587,969; 6,631,408; 6,631,411; 6,633,912; 6,694,288; 6,738,813; 6,738,933; 6,754,701; 6,792,460 and 6,810,494. Australia: 763468 and 762554. Other patents pending. All rights reserved.

Mercury, Mercury Interactive, the Mercury Interactive logo, LoadRunner, LoadRunner TestCenter, Mercury Business Process Testing, Mercury Quality Center, Quality Center, QuickTest Professional, SiteScope, SiteSeer, TestDirector, Topaz and WinRunner are trademarks or registered trademarks of Mercury Interactive Corporation or its subsidiaries, in the United States and/or other countries. The absence of a trademark from this list does not constitute a waiver of Mercury Interactive's intellectual property rights concerning that trademark.

All other company, brand and product names are registered trademarks or trademarks of their respective holders. Mercury Interactive Corporation disclaims any responsibility for specifying which marks are owned by which companies or which organizations.

Mercury Interactive Corporation
379 North Whisman Road
Mountain View, CA 94043
Tel: (650) 603-5200
Toll Free: (800) TEST-911
Customer Support: (877) TEST-HLP
Fax: (650) 603-5300

© 2004 Mercury Interactive Corporation, All rights reserved

If you have any comments or suggestions regarding this document, please send them via e-mail to documentation@mercury.com.

Table of Contents

Welcome to the TestDirector Requirements Synchronizer for Rational RequisitePro Add-in	v
Using This Guide.....	vii
Documentation Updates.....	viii
Typographical Conventions.....	viii
Chapter 1: What’s New in the TestDirector Requirements Synchronizer	1
Support of Multiple RequisitePro Projects.....	1
Support of RequisitePro Packages and Multiple Requirement Types ...	2
TestDirector Copies RequisitePro Hierarchical Structure.....	2
Improved Logging in HTML Format.....	2
Support of RequisitePro Views.....	2
Chapter 2: Installing the Requirements Synchronizer for Rational RequisitePro Add-in	3
Chapter 3: Building Tasks	7
Working with the Synchronizer.....	7
What the Synchronizer Adds to Your Tools.....	9
Understanding Task Building.....	9
Synchronizing with a RequisitePro View.....	11
Building a Task.....	13
Modifying a Task.....	20
Removing a Task.....	21
Chapter 4: Working with the Synchronizer	23
Changing the Synchronization Settings.....	23
Running a Task Manually.....	27
Running the Synchronizer Service.....	29
Synchronization Example.....	31

Table of Contents

Welcome to the TestDirector Requirements Synchronizer for Rational RequisitePro Add-in

Welcome to the TestDirector Requirements Synchronizer for Rational RequisitePro Add-in. This add-in supports Rational RequisitePro 2002 and 2003 with TestDirector 8.0 Enterprise Edition Service Pack 2 or later.

Note: To use the Synchronizer, you must have TestDirector administrator privileges.

The TestDirector Requirements Synchronizer enables you to manage project requirements using the most suitable tool for your needs by organizing and synchronizing the transfer of requirements data between TestDirector and RequisitePro.

For example, if the development team uses RequisitePro and the quality assurance team uses TestDirector, the Synchronizer enables the teams to integrate their projects so that they both work with the same data. This enables the quality assurance team to use one tool to handle the testing process, while the development team uses its preferred requirements management tool.

The Synchronizer employs multiple transfer options to satisfy varying user needs. This enables users to create requirements that do not yet exist in the receiving tool, update existing requirements, or remove deleted requirements.

You begin working with the Synchronizer by building a *synchronization task* in the Tasks Builder. A synchronization task represents data transfer between one TestDirector project and one RequisitePro project, which may contain multiple packages and requirements.

After you build a synchronization task, you can work with your requirements in TestDirector or RequisitePro.

As required, you can synchronize your projects by performing update tasks or delete tasks. An *update task* searches for and updates new or modified attributes. A *delete task* searches for and updates or removes deleted or moved attributes.

You can run the Synchronizer manually or as a service. Running the Synchronizer as a service enables you to run your tasks automatically according to predetermined time intervals.

Note: This guide explains how to use Rational RequisitePro with TestDirector. For additional information on using TestDirector, refer to the *TestDirector User's Guide*.

Using This Guide

This guide contains the following chapters:

Chapter 1 What's New in the TestDirector Requirements Synchronizer

Describes the new features available in the latest version of the Synchronizer.

Chapter 2 Installing the Requirements Synchronizer for Rational RequisitePro Add-in

Describes how to install the add-in on your TestDirector client machine. It also describes additional important post installation steps that are required if you are working with a TestDirector project that was upgraded, or if you have worked with a previous version of the Synchronizer.

Chapter 3 Building Tasks

Describes how to build, modify, and delete a synchronization task.

Chapter 4 Working with the Synchronizer

Describes how to run a synchronization task, either manually or as a service. Also includes an example of working with the Synchronizer.

Documentation Updates

Mercury Interactive is continuously updating its product documentation with new information. You can download the latest version of this document from the Customer Support Web site (<http://support.mercury.com>).

To download updated documentation:

- 1** In the Customer Support Web site, click the **Documentation** link.
- 2** Under **Select Product Name**, select TestDirector.

Note that if TestDirector does not appear in the list, you must add it to your customer profile. Click **My Account** to update your profile.

- 3** Click **Retrieve**. The Documentation page opens and lists the documentation available for the current release and for previous releases. If a document was recently updated, **Updated** appears next to the document name.
- 4** Click a document link to download the documentation.

Typographical Conventions

This book uses the following typographical conventions:

- | | |
|-------------------|--|
| 1, 2, 3 | Bold numbers indicate steps in a procedure. |
| ► | Bullets indicate options and features. |
| > | The greater than sign separates menu levels (for example, File > Open). |
| Stone Sans | The Stone Sans font indicates names of interface elements in a procedure that you perform actions upon (for example, “Click the Run button.”). |
| <i>Italics</i> | <i>Italic</i> text indicates variable names, or introduces a new term. |
| Arial | The Arial font is used for examples and statements that are to be typed in literally. |

1

What's New in the TestDirector Requirements Synchronizer

This chapter describes what's new in the TestDirector Requirements Synchronizer.

This chapter describes:

- ▶ Support of Multiple RequisitePro Projects
- ▶ Support of RequisitePro Packages and Multiple Requirement Types
- ▶ TestDirector Copies RequisitePro Hierarchical Structure
- ▶ Improved Logging in HTML Format
- ▶ Support of RequisitePro Views

Support of Multiple RequisitePro Projects

The Synchronizer now enables you to link multiple RequisitePro projects to one TestDirector project. You do this by building multiple tasks in which you link individual RequisitePro projects to the same TestDirector project. In TestDirector, this enables you to move requirements between various RequisitePro projects after all projects are synchronized.

Support of RequisitePro Packages and Multiple Requirement Types

After you synchronize your projects, TestDirector now displays RequisitePro projects and packages as parent requirements. In TestDirector, this enables you to move child requirements from one requirement representing a package to another. In addition, each requirement name now includes its requirement type, as is done in RequisitePro.

TestDirector Copies RequisitePro Hierarchical Structure

TestDirector now simulates the hierarchy of synchronized RequisitePro projects, ensuring that requirements remain grouped logically and appear similarly in both tools. When you synchronize projects, the hierarchical structure of your RequisitePro project is simulated in TestDirector. This ensures that projects, packages, and requirements appear similarly in both tools.

Improved Logging in HTML Format

The Synchronizer now provides improved logging that contains more detailed information about the synchronization process. The easy-to-read log files are displayed in HTML format. Every row in the log file is also color-coded to represent a specific information type, such as debugging information or errors.

Support of RequisitePro Views

The Synchronizer now provides support for synchronizing a TestDirector project with a RequisitePro view. When you build a task, you are able to choose to synchronize your TestDirector project with a RequisitePro requirement type or a specific view.

2

Installing the Requirements Synchronizer for Rational RequisitePro Add-in

To synchronize Rational RequisitePro and TestDirector projects, you must install the TestDirector Requirements Synchronizer for Rational RequisitePro Add-in on your TestDirector client machine.

Notes:

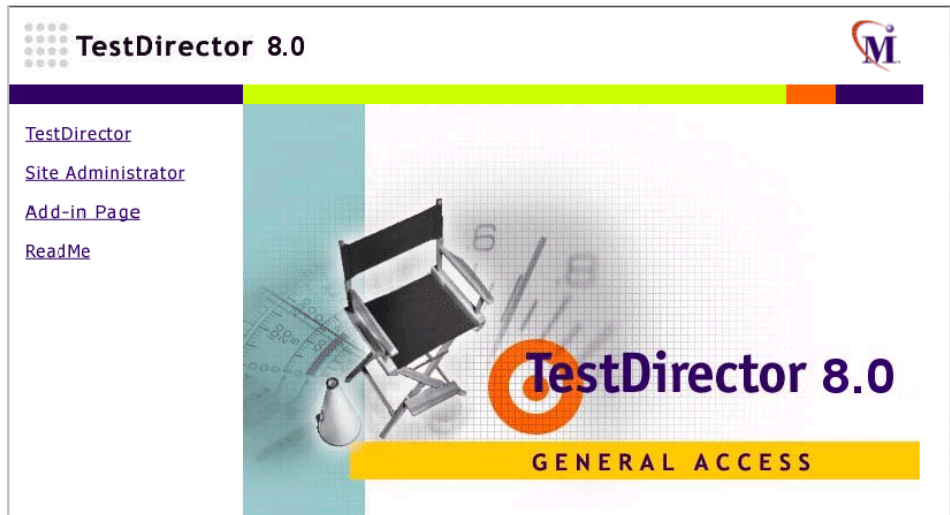
- ▶ If you built synchronization tasks using a previous version of the Synchronizer, you must upgrade them.
 - ▶ If your TestDirector project was upgraded to TestDirector 8.0, TestDirector 8.0 SP1, or to the same version of TestDirector, you must perform additional important post-installation steps. For more information, contact Mercury Customer Support.
-

To install the Requirements Synchronizer for Rational RequisitePro Add-in:

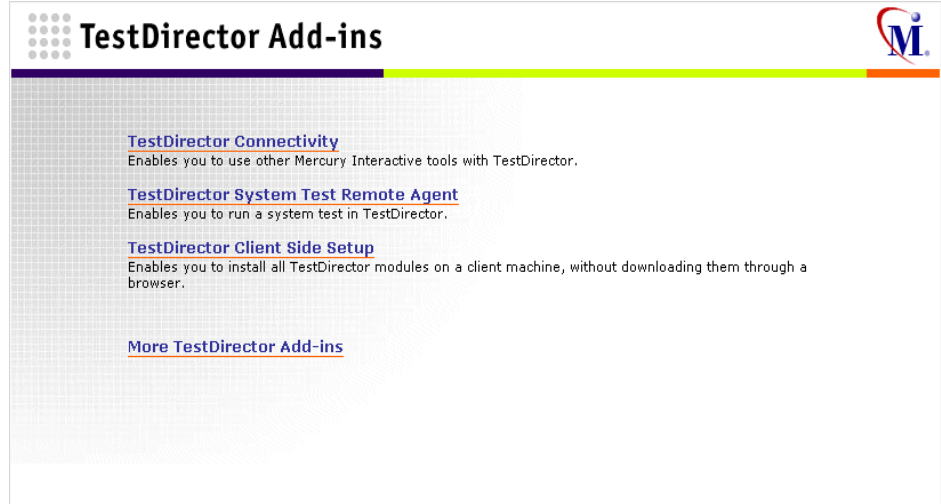
- 1 Install a RequisitePro Windows client on your TestDirector client machine. Check that both clients are communicating successfully with their servers.

Note: You must install a RequisitePro Windows client to work with the Synchronizer, even if you are working with a RequisitePro Web client.

- 2 Uninstall any previous versions of the TestDirector Requirements Synchronizer for Rational RequisitePro Add-in. To uninstall, choose **Start > Settings > Control Panel > Add/Remove Programs** and follow the instructions on your screen.
- 3 Open your Web browser and type your TestDirector URL:
`http://<TestDirector server name>/<virtual directory name>/default.htm`.
The TestDirector Options window opens.



- 4 Click the **Add-ins Page** link. The TestDirector Add-ins page opens.



- 5 Click the **More TestDirector Add-ins** link. The More TestDirector Add-ins page opens.
- 6 Under **Synchronization Tool Add-ins**, click the **Requirements Synchronizer for Rational RequisitePro Add-in** link. The TestDirector Requirements Synchronizer for Rational RequisitePro Add-in page opens.
- 7 Click **Download Add-in** and install the Synchronizer on your client machine. Follow the on-screen installation instructions.

Note: To install this add-in, you must log in with administrator privileges.

- 8 Navigate back to the TestDirector Add-ins page and click the **TestDirector Connectivity Add-in** link. The TestDirector Connectivity Add-in opens.
- 9 Click **Download Add-in** and install the Connectivity Add-in on your TestDirector client machine.

3

Building Tasks

You use the TestDirector Synchronizer Tasks Builder to build tasks. Each synchronization task enables you to transfer data between one TestDirector project and one Rational RequisitePro project. You can also build additional tasks to link multiple RequisitePro projects to your TestDirector project.

This chapter describes:

- ▶ Working with the Synchronizer
- ▶ What the Synchronizer Adds to Your Tools
- ▶ Understanding Task Building
- ▶ Building a Task
- ▶ Modifying a Task
- ▶ Removing a Task

Working with the Synchronizer

When working with the Synchronizer, make sure that you follow these guidelines:

- ▶ Do not use the following characters when naming a requirement in TestDirector, or when naming a project, package, or requirement in RequisitePro: \ ^ * " ' : ' "
- ▶ The first time that you synchronize projects, you must run a full scan. This ensures that project data are identical in both tools.

- ▶ In TestDirector, if you want to move a requirement and make it a child of another requirement, make sure that you move it under a requirement of the same type. This is required because RequisitePro does not allow you to move a requirement of one type under a requirement of another type. If you do, the Synchronizer will move the requirement back when you synchronize projects.
- ▶ In RequisitePro, do not add more than 676 requirements per hierarchical level as this is the maximum supported by TestDirector.
- ▶ After you synchronize projects from RequisitePro to TestDirector, requirement names are displayed as **<Requirement Type>; <Requirement Name>**. Do not delete the requirement type, the semi-colon, or the space after the semi-colon from the requirement name in TestDirector.
- ▶ In TestDirector, you cannot create, rename, or move packages.
- ▶ After you synchronize projects, if you rename a package in RequisitePro, the next time you synchronize the projects a new package will be created in TestDirector. All the requirements of the same requirement type will be moved to the new package. If the original project contained one requirement type and no child packages, the original package will be removed from the TestDirector project after you run a delete task (for more information, see “Running a Task Manually” on page 27). If, however, the original package contained requirements of another type, or if it contained child packages, the original package will not be removed from TestDirector.
- ▶ After you build a task, do not modify the project login connection information. Specifically, for each linked project, do not modify the TestDirector server name, the TestDirector domain name, the TestDirector/RequisitePro project name, your user name, or password.

This is because the Synchronizer connects to each project using the login information you specified when you built the task. If you change the login information, you must delete the task using your file management application. For more information, see “Removing a Task” on page 21.

- ▶ If you define masks in TestDirector, make sure that the data you enter in RequisitePro matches the input mask string. For more information on defining input masks in TestDirector, refer to the *TestDirector Administrator's Guide*.

What the Synchronizer Adds to Your Tools

The Synchronizer adds the following to RequisitePro and TestDirector:

- ▶ In RequisitePro, the Synchronizer adds two new fields:
 - **SYNC_TIME_STAMP**: a text field that is used to determine whether the Rational RequisitePro requirement was updated.
 - **TD_KEY**: a text field that is used to store information about the corresponding TestDirector requirement.
- ▶ In the TestDirector Requirements table, the Synchronizer reserves the first available user-defined requirement field to store information about the corresponding Rational RequisitePro requirement. The Synchronizer assigns this field the name **RP_SYNCH_INFO**.

Note: If your tasks were built using either TestDirector 7.6 or TestDirector 8.0 with Service Pack 1 versions of the Synchronizer, the Synchronizer uses the **RQ_USER_12** system field instead, and changes it to the **RP_SYNCH_INFO** user-defined field.

When you run a task, the Synchronizer stores information about corresponding requirements in the **TD_KEY** field in your RequisitePro project and the **RP_SYNCH_INFO** field in your TestDirector project.

Understanding Task Building

Building a new synchronization task involves the following four steps. For additional information, see “Building a Task” on page 13.

- ▶ **Set the TestDirector project information:** Connect to the TestDirector server and project.
- ▶ **Set the RequisitePro project information:** Connect to the RequisitePro project and either choose to synchronize a requirement type or a view. For more information about synchronizing a RequisitePro view, see “Synchronizing with a RequisitePro View” on page 11. If you want to

synchronize a project containing multiple requirement types, you can build a task for each requirement type.

Note: For each task, you can choose one requirement type only. If applicable, you can instruct the Synchronizer to create additional RequisitePro fields to enable more convenient mappings with the TestDirector projects.

- ▶ **Set the project field mappings:** Map the fields in your TestDirector project to fields in the RequisitePro project. For each field, you can specify which tool takes precedence and overrides the value in the other tool when the projects are synchronized. If needed, you can map additional fields from the TestDirector project to the RequisitePro project.
 - ▶ **Set the deletion rules:** Specify the record removal rules when a delete task is performed.
-

Note: You can link multiple RequisitePro projects to one TestDirector project. You do this by building additional tasks in which you link individual RequisitePro projects to the same TestDirector project.

After you synchronize your projects, the TestDirector project contains all the RequisitePro projects. In TestDirector, each RequisitePro project is displayed hierarchically as a root level parent requirement. This enables you to view the parent-child relationship between requirements and to move requirements between what were originally separate RequisitePro projects.

Synchronizing with a RequisitePro View

You can synchronize your TestDirector project with a RequisitePro view. If you have already synchronized requirements between your TestDirector project and the RequisitePro view, or synchronized requirements that were previously part of the view, the Synchronizer will continue to update these requirements in subsequent synchronizations.

- ▶ If you synchronize a RequisitePro view with a TestDirector project, then modify the query of the RequisitePro view and synchronize again, any requirements that were synchronized in the old view that are now in the new view are synchronized.
- ▶ If you modify a query in your RequisitePro view and rename the view, you must select the new name in the **Requirement View** box of the RequisitePro Login dialog box in order for it to be synchronized when you run a task. For more information, see “Building a Task” (step 8) on page 16.

The following sections provide additional information for:

- ▶ Adding Requirements
- ▶ Modifying Requirements
- ▶ Deleting Requirements

Adding Requirements

Before adding a requirement to your TestDirector project or to your RequisitePro view, you should consider the following issues:

- ▶ If you add a new requirement in your TestDirector project, it must be created under the root requirement that is mapped to the RequisitePro view. Only requirements that are part of the RequisitePro project can be synchronized.
- ▶ If you add a new requirement in your TestDirector project, it will be synchronized with the RequisitePro project. The Synchronizer has no way of determining if a requirement should be part of the RequisitePro view, so all requirements are added when you run a task.
- ▶ If you add a requirement with a child requirement in your RequisitePro project, but only the child requirement is in the view, both the parent requirement and its child are created in your TestDirector project when you synchronize.

Modifying Requirements

Before modifying a requirement in your TestDirector project or in your RequisitePro view, you should consider the following issues:

- ▶ If you modify a requirement in the RequisitePro project that is not part of the view, or does not already have a corresponding requirement in your TestDirector project, it will not be synchronized with the TestDirector project when you run a task.
- ▶ A RequisitePro view is part of a requirement type. If you modify a requirement in your TestDirector project that belongs to the requirement type that the RequisitePro view is a part of, it will be synchronized when you run a task.

Deleting Requirements

Before deleting a requirement from your TestDirector project or from your RequisitePro view, you should consider the following issues:

- ▶ The following applies if you synchronize your TestDirector project with a RequisitePro view and then modify the query in the RequisitePro view. If you delete a requirement in the TestDirector project and then run a delete task, the Synchronizer does not delete the corresponding requirement in the RequisitePro project while that requirement is not a part of the view.

The Synchronizer clears the **TD_KEY** field in your RequisitePro project. This means that subsequent modifications to the requirement in the RequisitePro project are not synchronized with your TestDirector project until the requirement once again becomes part of the view.

- ▶ The following applies if you delete a requirement in your TestDirector project that has a child in the RequisitePro project which is not part of the view. When you run a delete task, the child in the RequisitePro project becomes a child of the package that the parent belonged to.
- ▶ The following applies if you delete a requirement in your RequisitePro project that is not part of the view, but already has a corresponding TestDirector requirement. When you run a delete task, the requirement is deleted from your TestDirector project.

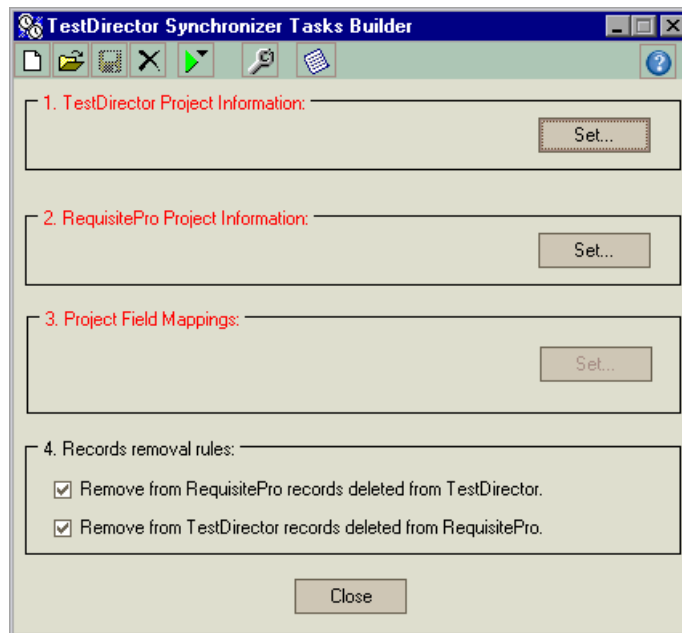
Building a Task

This section describes how to build a synchronization task. The procedure comprises four main steps:

- Defining TestDirector project information
- Defining RequisitePro project information
- Mapping TestDirector fields with RequisitePro fields
- Specifying the records removal rules

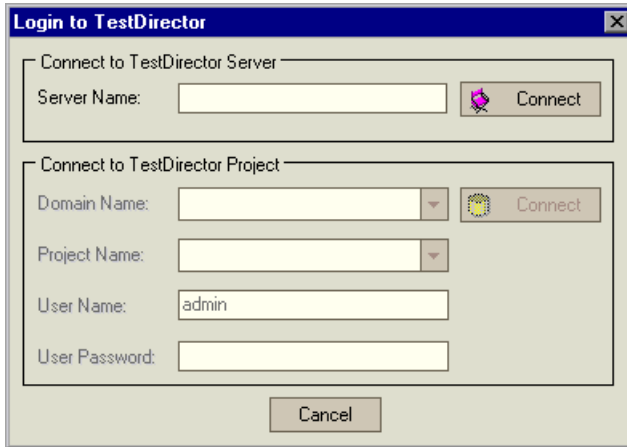
To build a synchronization task:

- 1** Choose **Start > Programs > TestDirector 8.0 > Integrations > Requirements Synchronizer > Tasks Builder**. The TestDirector Synchronizer Tasks Builder dialog box opens.



- 2** Click the **New Task** button.

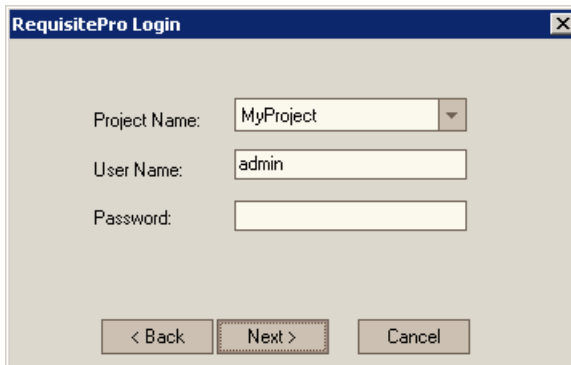
- Under **TestDirector Project Information**, click **Set**. The Login to TestDirector dialog box opens.



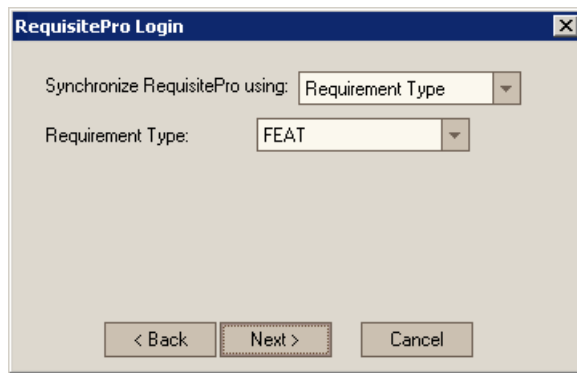
- Log in to the TestDirector server and project. Note that the **Server Name** must be entered as: `http://<server name>/...` or `https://<server name>/...`

If the connection is successful, **TestDirector Project Information** changes from red to blue.

- Under **RequisitePro Project Information**, click **Set**. The RequisitePro Login dialog box opens.



- 6 Log in to the RequisitePro project and click **Next**. If the connection is successful, the next dialog box opens.

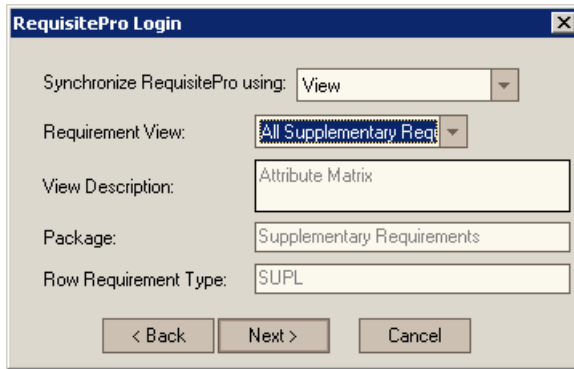


- 7 In the **Synchronize RequisitePro using** box, choose whether you want to synchronize your RequisitePro project according to Requirement Type or View:
 - ▶ **Requirement Type:** Synchronizes the requirements in your TestDirector project with your RequisitePro project according to requirement type.
 - ▶ **View:** Synchronizes the requirements in your TestDirector project with a RequisitePro project view.

Click **Next**.

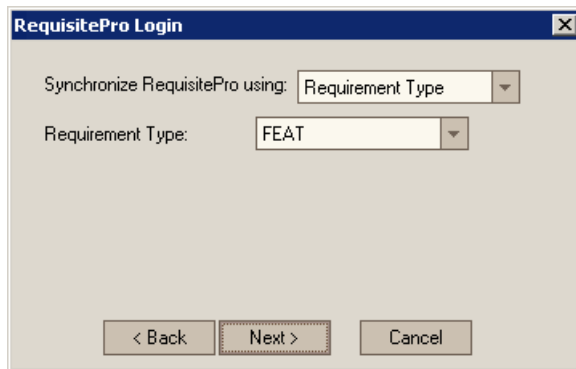
If you chose **Requirement Type**, go to step 11.

- 8 If you selected to synchronize according to **View**, the following dialog box opens.



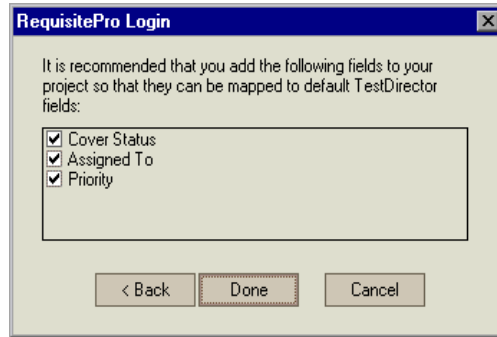
In the **Requirement View** box, choose the view that you want to synchronize with your TestDirector project. The details that correspond to the view you have selected are displayed in the **View Description**, **Package** and **Row Requirement Type** boxes.

- 9 Click **Next**. The TestDirector Synchronizer Tasks Builder dialog box opens.
- 10 If you need to map fields between your RequisitePro project and your TestDirector project, go to step 13. Otherwise, go to step 18.
- 11 If you selected to synchronize according to **Requirement Type** in step 7, select a requirement type.

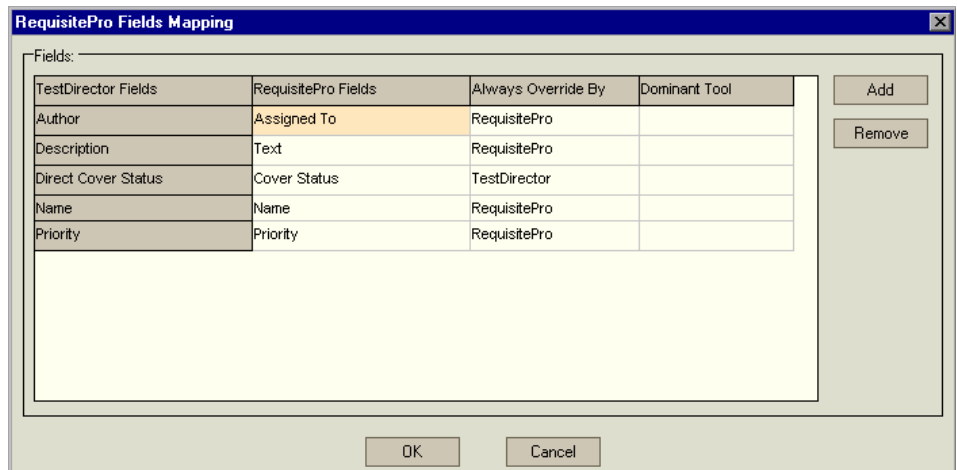


Note that for each synchronization task, you can synchronize only one requirement type. Click **Next**.

If applicable, the next dialog box opens, displaying a list of recommended TestDirector fields that are not in your RequisitePro project. Adding these fields to RequisitePro maps them automatically to your TestDirector project, ensuring that the database structures contain logical field mappings.



- 12** Click **Done**. The Synchronizer adds the fields to your RequisitePro project and maps them to TestDirector fields. The TestDirector Synchronizer Tasks Builder dialog box opens. **RequisitePro Project Information** changes from red to blue.
- 13** Under **Project Field Mappings**, click **Set**. The RequisitePro Fields Mapping dialog box opens and displays the recommended field mappings.



The mapping grid contains the following columns:

- ▶ **TestDirector Fields:** The field label, as defined in TestDirector.
- ▶ **RequisitePro Fields:** The field (also known as an *attribute*), as defined in RequisitePro, that maps to the TestDirector field in the same row.

Notes:

- ▶ Ensure that the lookup list of possible values is the same for fields that are mapped to each other. For example, if the **Assigned To** field in RequisitePro lists project members by user name, ensure that the field that it is mapped to in your TestDirector project, **Author**, also lists project members by user name.
- ▶ You can map a field from one tool to another only once.
- ▶ If you create a requirement in your TestDirector project that you want to be a part of your RequisitePro view, you need to ensure that all the fields that relate to the query are mapped between your TestDirector project and your RequisitePro project.
- ▶ Do not map RequisitePro numeric fields (of type **Real**) to TestDirector numeric fields, as TestDirector numeric fields support integers only.

-
- ▶ **Always Override By:** This defines the dominant tool that overrides data during synchronization, RequisitePro or TestDirector. For example, if the value for **Always Override By/Name** is RequisitePro, when the tools are synchronized, **Name** in RequisitePro remains the same while **Name** in TestDirector is updated to the same value as in RequisitePro.

Always Override By can also be defined as Bidirectional Update, which works together with the **Dominant Tool** column. Bidirectional Update specifies that the tool that overrides is the tool in which data was updated.

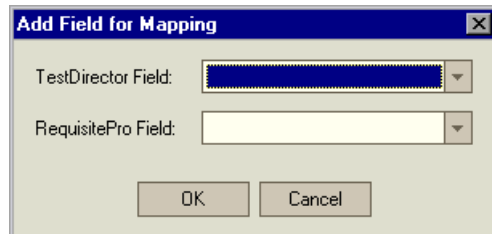
- ▶ **Dominant Tool:** If you define **Always Override By** as Bidirectional Update, then you must specify the **Dominant Tool** as either RequisitePro or TestDirector. When data is changed in both tools, the Synchronizer performs an update according to the dominant tool defined.

For example, if **Name** is set to Bidirectional Update and **Dominant Tool** is set to RequisitePro, the following occurs:

If a requirement **Name** was changed in your TestDirector project but not in your RequisitePro project, when the tools are synchronized, the value for **Name** in TestDirector overrides the value in RequisitePro.

If the requirement **Name** was changed in both your TestDirector project and your RequisitePro project, when the tools are synchronized, the RequisitePro field value overrides the TestDirector value as it is the dominant tool.

- 14** To add a new field for mapping, click **Add**. The Add Field for Mapping dialog box opens.



- 15** Select a **TestDirector Field** and a **RequisitePro Field**, and click **OK**. The Synchronizer adds the fields to the mapping grid.
- 16** To remove a field mapping, click in the row containing the mapped fields and click **Remove**. Click **Yes** to confirm.
- 17** When you have completed mapping the fields, click **OK**. The TestDirector Synchronizer Task Builder dialog box opens. **Project Field Mappings** changes from red to blue.
- 18** Under **Records removal rules**, you can specify the deletion rules for synchronization.
- Select the **Remove from RequisitePro records deleted from TestDirector** check box to instruct the Synchronizer to delete from the RequisitePro project any records that were deleted in the TestDirector project. If this option is not selected, when the tools are synchronized, the Synchronizer recreates any records deleted from the TestDirector project by copying them from the relevant requirement in the RequisitePro project.

- ▶ Select the **Remove from TestDirector records deleted from RequisitePro** check box to instruct the Synchronizer to delete from the TestDirector project any records that were deleted in the RequisitePro project. If this option is not selected, when the tools are synchronized, the Synchronizer recreates any records deleted from the RequisitePro project by copying them from the relevant requirement in the TestDirector project.



- 19** To save the new task, click the **Save** button. In the Task Description dialog box, click **OK** to accept the name assigned to the task. By default, the task name appears as follows:

<TDProjectName>@<RPProjectName>@<RPRequirementType>(<index>).tsk

Note: Although you can change the name of a task, it is recommended to accept the default name provided by the Synchronizer. If you do choose to change the default name, do not use the following characters: \ / : * ? " < > |

- 20** In Rational RequisitePro, if your project is open, choose **View > Refresh** to update it.

Modifying a Task

You can modify an existing synchronization task.

To modify a task:

- 1** Choose **Start > Programs > TestDirector 8.0 > Integrations > Requirements Synchronizer > Tasks Builder**. The TestDirector Synchronizer Tasks Builder window opens.



- 2** Click **Open**. The Open Task dialog box opens.
- 3** Select a synchronization task from the list and click **OK**.
- 4** Click the **Edit** button to modify each step in the task.

Notes:

- After you modify or delete an existing **Project Field Mapping**, click the **Synchronize Now** button and choose **Run Full Scan**.
- After you add a **Project Field Mapping**, save the task, click the **Synchronize Now** button, and choose **Run Full Scan**.

Note that all TestDirector and RequisitePro users must log off before you run a full scan task. For more information, see “Running a Task Manually” on page 27.



- 5 Click **Save** to save your changes.

Removing a Task

If you no longer want to run a synchronization task, you can remove it using the Synchronizer or your file management application. If you remove a task using the Synchronizer, you are removing the linked requirement mappings between TestDirector and RequisitePro. If you remove a task using your file management application, you are preserving the mapped links, enabling you to build new tasks using your existing mappings.

To remove a task and its linked requirement mappings:



- 1 In the TestDirector Synchronizer Tasks Builder, click the **Delete and Unlink a Task** button. The Delete Task dialog box opens.
- 2 Select a synchronization task from the list to remove and click **OK**. A confirmation dialog box opens.
- 3 Click **Yes** to confirm.

To remove a task and preserve its linked requirement mappings:

- 1** In your file management application, navigate to your tasks. By default, they are located under <drive>:\Program Files\Common Files\Mercury Interactive\TDReqSync\Tasks.

4

Working with the Synchronizer

After you create a synchronization task, you can run the task manually or automatically.

This chapter describes:

- ▶ Changing the Synchronization Settings
- ▶ Running a Task Manually
- ▶ Running the Synchronizer Service
- ▶ Synchronization Example

Changing the Synchronization Settings

Each time a synchronization task runs, an HTML log file is written. You can change the settings for the log file by specifying its location, level of detail, and maximum size.

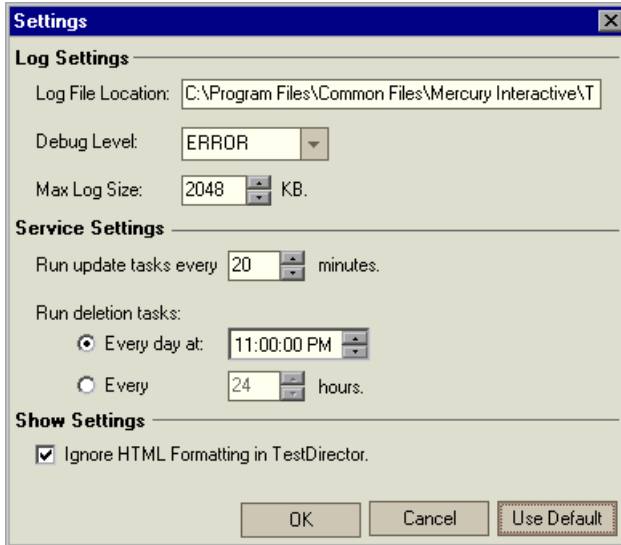
You can also specify how often update and delete tasks are run when synchronization tasks are run as a service.

The settings you define are applied to all synchronization tasks.

To change the task settings:



- 1 In the TestDirector Synchronizer Tasks Builder, click the **Settings** button. The Settings dialog box opens.



- 2 In **Log File Location**, specify an existing folder for the log file.
- 3 In **Debug Level**, select a level of detail for the information written to the log file:
 - ▶ **NONE**: No log file is created.
 - ▶ **ERROR**: Only errors are written to the log file.
 - ▶ **DEBUG**: Every action performed by the Synchronizer is written to the log file.

Note: If you change the **Debug Level** setting, you must restart the TestDirector ReqSync Service to enable the service log to use the new setting. For more information, see “Running the Synchronizer Service,” on page 29.

- 4** In **Max Log Size**, select the maximum size of a synchronization log file. It is recommended that a log file does not exceed the default, 2048 KB. If a log file exceeds the **Max Log Size**, the Synchronizer splits the file and adds a suffix to the subsequent log file(s), for example, <Log name>_Log_2.HTM. This keeps each log file a manageable size.
-

Note: If you do not want the log file to split, select a **Max Log Size** of **0**.

- 5** In **Run update tasks every**, specify the interval of update tasks in minutes.
-

Note: **Run update tasks every** and **Run deletion tasks** are used by the TestDirector ReqSync Service, as described in “Running the Synchronizer Service,” on page 29.

- 6** Under **Run deletion tasks**, select one of the following options:
- **Every day at:** Specifies the time each day that the synchronization of deleted items is performed. It is recommended to synchronize deleted items once a day at a time when users are not connected.
 - **Every ... hours:** Specifies that the deletion process should be run every X hours. This is the minimum time interval between the synchronization of deleted items. Note that you can prevent the delete task from running by selecting this option and entering **0** hours.

Note: Delete tasks can only be run after update tasks. Therefore, the TestDirector ReqSync Service only checks the delete interval after an update task is run.

For example, if you configure an update interval of 20 minutes and a delete interval of 1 hour, each time the update task runs, the service checks if the delete time interval has been reached. The third time that the update task runs, 60 minutes have passed. This means that the delete time interval has been reached and the delete task is then run immediately after the update task.

7 Under **Show Settings**, you can leave the **Ignore HTML formatting in TestDirector** check box selected, by default, or clear the check box.

- ▶ If the check box is selected, the Synchronizer does not preserve formatted text in TestDirector memo fields. After synchronization, all formatted text is displayed in TestDirector as plain text.
 - ▶ If the check box is cleared, the Synchronizer preserves formatted text, such as bold or multi-colored text, in all TestDirector memo fields. After synchronization, all formatted text is displayed in TestDirector as formatted text.
-

Note: RequisitePro does not support formatted text. Therefore, in RequisitePro, formatted text is displayed as plain text with HTML tags.

8 Click **OK**.

Note: You can click **Use Default** to restore the default Synchronizer settings.

Running a Task Manually

You can run a single update, full scan, or delete task manually. Note that you can also set up tasks to run automatically as a service, as described in “Running the Synchronizer Service,” on page 29.

An *update task* identifies requirements that have been created or updated in one of the tools since the last synchronization task. The Synchronizer creates or updates the relevant requirements in the other tool, according to the mapping rules specified for the task.

A *full scan* updates all the requirements according to the updated mapping rules specified for the task. You run a full scan the first time you run a synchronization task and when you add, modify, or delete a project field mapping. For more information on changing project field mappings, see Chapter 3, “Modifying a Task.” Note that running a full scan requires all TestDirector and RequisitePro users to log off.

A *delete task* identifies requirements that have been removed from one tool and deletes them from, or recreates them in, the other tool, according to the deletion rules specified for the task. A delete task also identifies new and updated requirements, as well as requirements that were moved from one package to another in RequisitePro, or from one requirement to another in TestDirector, updating them in the new location.

Note: Running a delete task may take some time because the Synchronizer reviews every record, searching for changes.

After you run a task, the projects are synchronized according to the mapping rules and deletion rules you specified. For more information, see Chapter 3, “Building Tasks.” After synchronization, the TestDirector project replicates the hierarchy of the RequisitePro project, ensuring that projects, packages, and requirements appear similarly in both tools.

Note: The Synchronizer can run only one synchronization task at a time. Therefore, before running a task manually, make sure that the TestDirector ReqSync Service is stopped. For more information, see “Running the Synchronizer Service,” on page 29.

To run a task:

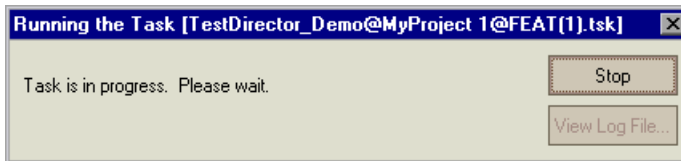


1 In the TestDirector Synchronizer Tasks Builder, click the **Synchronize Now** button.

- To run an update task, choose **Run Update Task Now**.
- To run a full scan task, choose **Run Full Scan**.
- To run a delete task, choose **Run Delete Task Now**.

The Run Task dialog box opens.

2 Select a task to run and click **OK**. The Running the Task dialog box opens.



You can click the **Stop** button to stop the update.

3 After the task is complete, you can click **View Log File** to view a color-coded HTML synchronization log file.

- Green rows indicate general information.
- Purple rows indicate debugging information.
- Yellow rows indicate highlighted information.
- Red rows indicate errors.

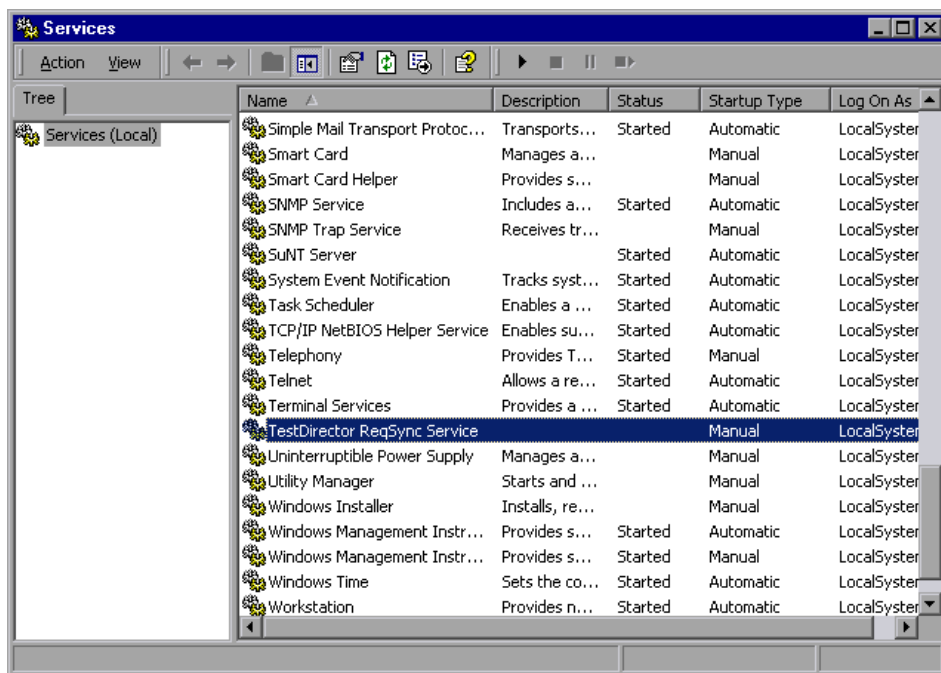
4 In the Running the Task dialog box, click **Close**.

Running the Synchronizer Service

Instead of manually specifying which tasks to run, you can instruct the Synchronizer to run as a service. The service runs in the background and scans for all available tasks created using the TestDirector Synchronizer Tasks Builder. The tasks are then run according to the intervals you specified in the Settings dialog box (see “Changing the Synchronization Settings,” on page 23).

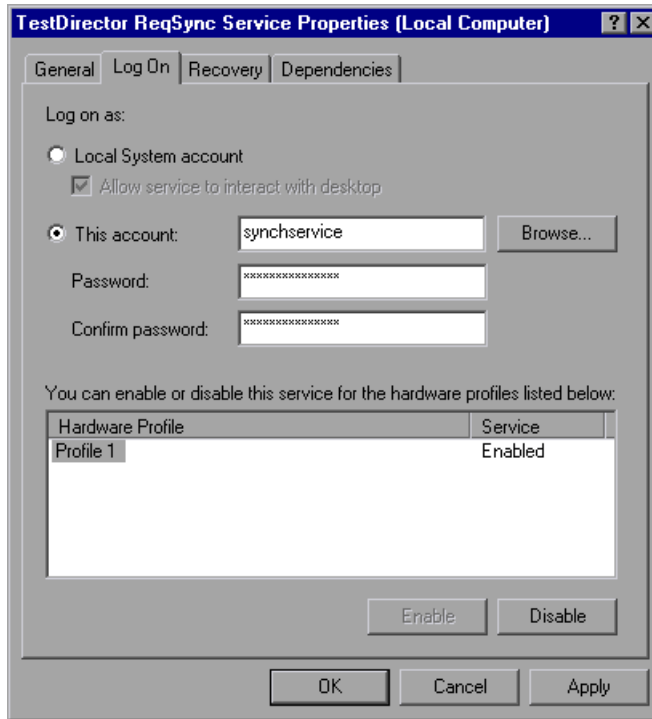
To set up the service:

- 1 Choose **Start > Settings > Control Panel > Administrative Tools** and double-click **Services**. The Services dialog box opens.



- 2 Double-click **TestDirector ReqSync Service**. Alternatively, right-click **TestDirector ReqSync Service** and choose **Properties**. The TestDirector ReqSync Service Properties dialog box opens.

3 Click the **Log On** tab.

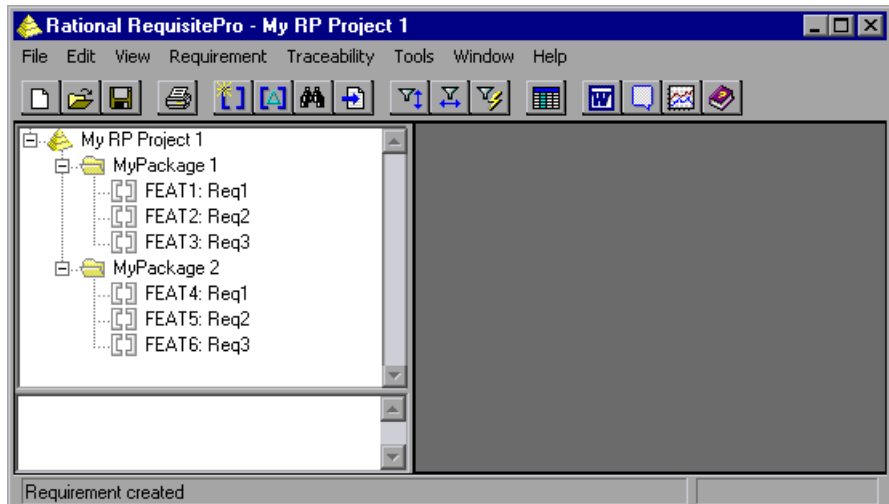


- 4 Select **This account** and click the **Browse** button to select a user account that has administrator permissions on the machine. It must also belong to the same domain.
- 5 Type and confirm the password.
- 6 Click **OK** to close the TestDirector ReqSync Service Properties dialog box.
- 7 In the Services dialog box, right-click **TestDirector ReqSync Service** and choose **Start** to start the service.
- 8 To pause or stop this service, right-click **TestDirector ReqSync Service** and choose **Pause** or **Stop**.
- 9 Close the Services dialog box.

Synchronization Example

This example illustrates how projects are updated when using the Synchronizer with TestDirector and RequisitePro projects.

This is the RequisitePro project. It contains two packages, six requirements, and one requirement type, **FEAT**.



The user synchronizes the RequisitePro project with an empty TestDirector project by running a full scan.

The TestDirector project now contains all the requirements listed in the RequisitePro project. Note that the hierarchical structure of the RequisitePro project is replicated in TestDirector.

The screenshot shows the TestDirector interface for 'Project : My_TD_Project [admin]'. The 'REQUIREMENTS' tab is active. The 'Document View' shows a tree structure of requirements. The 'Direct Cover Status' column indicates that all requirements are 'Not Covered'.

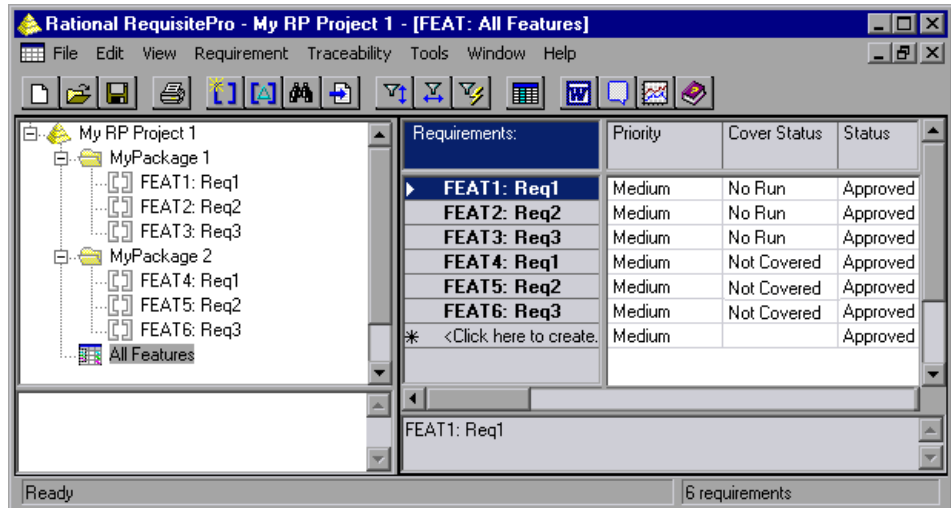
Name	Direct Cover Status	ReqID
My RP Project 1	? Not Covered	[RQ0012]
MyPackage 1	? Not Covered	[RQ0013]
FEAT1; Req1	? Not Covered	[RQ0014]
FEAT2; Req2	? Not Covered	[RQ0015]
FEAT3; Req3	? Not Covered	[RQ0016]
MyPackage 2	? Not Covered	[RQ0017]
FEAT4; Req1	? Not Covered	[RQ0018]
FEAT5; Req2	? Not Covered	[RQ0019]
FEAT6; Req3	? Not Covered	[RQ0020]

In TestDirector, the user assigns tests coverage to **MyPackage 1** and its child requirements.

The screenshot shows the TestDirector interface for 'Project : My_TD_Project [admin]'. The 'REQUIREMENTS' tab is active. The 'Document View' shows the same tree structure as the previous screenshot. The 'Direct Cover Status' column indicates that 'MyPackage 1' and its children are 'No Run', while 'MyPackage 2' and its children remain 'Not Covered'.

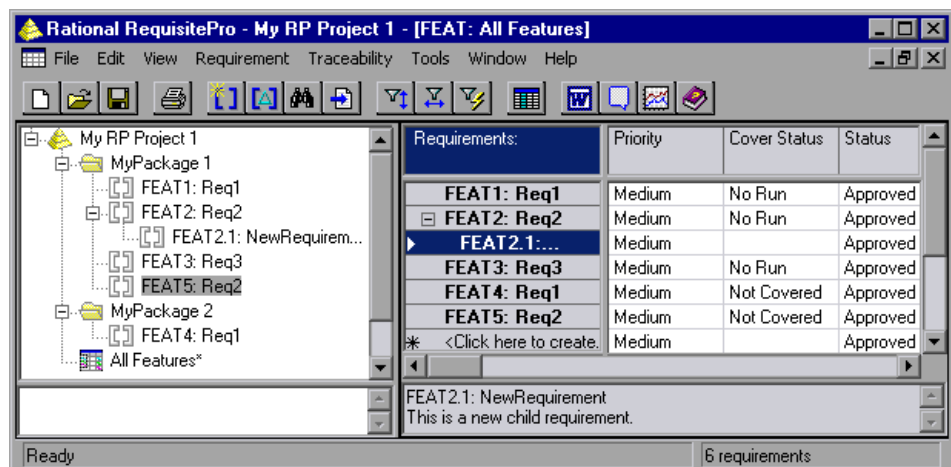
Name	Direct Cover Status	ReqID
My RP Project 1	? Not Covered	[RQ0012]
MyPackage 1	▶ No Run	[RQ0013]
FEAT1; Req1	▶ No Run	[RQ0014]
FEAT2; Req2	▶ No Run	[RQ0015]
FEAT3; Req3	▶ No Run	[RQ0016]
MyPackage 2	? Not Covered	[RQ0017]
FEAT4; Req1	? Not Covered	[RQ0018]
FEAT5; Req2	? Not Covered	[RQ0019]
FEAT6; Req3	? Not Covered	[RQ0020]

After the user runs an update task, both projects are synchronized and the **Cover Status** is updated in the RequisitePro project.



In the RequisitePro project, the user then performs the following actions:

- Moves **FEAT5; Req2** from **MyPackage 2** to **MyPackage 1**
- Deletes **FEAT6; Req3**
- Adds a child requirement and text under **FEAT2; Req2**



After the user runs a delete task:

- The TestDirector project displays **FEAT5; Req2** in its new location under **MyPackage 1**.
- **FEAT6; Req3** is removed from the project.
- **FEAT2; Req2** now has a new child requirement, **FEAT2.1; NewRequirement**, whose description is displayed in the Description tab in the lower part of the window.

