NI Requirements Gateway

Getting Started with NI Requirements Gateway

January 2009
375035A-01

NATIONAL INSTRUMENTS™

Worldwide Technical Support and Product Information

ni.com

National Instruments Corporate Headquarters

11500 North Mopac Expressway Austin, Texas 78759-3504 USA Tel: 512 683 0100

Worldwide Offices

Australia 1800 300 800, Austria 43 662 457990-0, Belgium 32 (0) 2 757 0020, Brazil 55 11 3262 3599, Canada 800 433 3488, China 86 21 5050 9800, Czech Republic 420 224 235 774, Denmark 45 45 76 26 00, Finland 358 (0) 9 725 72511, France 01 57 66 24 24, Germany 49 89 7413130, India 91 80 41190000, Israel 972 3 6393737, Italy 39 02 41309277, Japan 0120-527196, Korea 82 02 3451 3400, Lebanon 961 (0) 1 33 28 28, Malaysia 1800 887710, Mexico 01 800 010 0793, Netherlands 31 (0) 348 433 466, New Zealand 0800 553 322, Norway 47 (0) 66 90 76 60, Poland 48 22 328 90 10, Portugal 351 210 311 210, Russia 7 495 783 6851, Singapore 1800 226 5886, Slovenia 386 3 425 42 00, South Africa 27 0 11 805 8197, Spain 34 91 640 0085, Sweden 46 (0) 8 587 895 00, Switzerland 41 56 2005151, Taiwan 886 02 2377 2222, Thailand 662 278 6777, Turkey 90 212 279 3031, United Kingdom 44 (0) 1635 523545

For further support information, refer to the *Technical Support and Professional Services* appendix. To comment on National Instruments documentation, refer to the National Instruments Web site at ni.com/info and enter the info code feedback.

Important Information

Warranty

The media on which you receive National Instruments software are warranted not to fail to execute programming instructions, due to defects in materials and workmanship, for a period of 90 days from date of shipment, as evidenced by receipts or other documentation. National Instruments will, at its option, repair or replace software media that do not execute programming instructions if National Instruments receives notice of such defects during the warranty period. National Instruments does not warrant that the operation of the software shall be uninterrupted or error free.

A Return Material Authorization (RMA) number must be obtained from the factory and clearly marked on the outside of the package before any equipment will be accepted for warranty work. National Instruments will pay the shipping costs of returning to the owner parts which are covered by warranty.

National Instruments believes that the information in this document is accurate. The document has been carefully reviewed for technical accuracy. In the event that technical or typographical errors exist, National Instruments reserves the right to make changes to subsequent editions of this document without prior notice to holders of this edition. The reader should consult National Instruments if errors are suspected. In no event shall National Instruments be liable for any damages arising out of or related to this document or the information contained in it.

EXCEPT AS SPECIFIED HEREIN, NATIONAL INSTRUMENTS MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. CUSTOMER'S RIGHT TO RECOVER DAMAGES CAUSED BY FAULT OR NEGLIGENCE ON THE PART OF NATIONAL INSTRUMENTS SHALL BE LIMITED TO THE AMOUNT THERETOFORE PAID BY THE CUSTOMER. NATIONAL INSTRUMENTS WILL NOT BE LIABLE FOR DAMAGES RESULTING FROM LOSS OF DATA, PROFITS, USE OF PRODUCTS, OR INCIDENTAL OR CONSEQUENTIAL DAMAGES, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. This limitation of the liability of National Instruments will apply regardless of the form of action, whether in contract or tort, including negligence. Any action against National Instruments must be brought within one year after the cause of action accrues. National Instruments shall not be liable for any delay in performance due to causes beyond its reasonable control. The warranty provided herein does not cover damages, defects, malfunctions, or service failures caused by owner's failure to follow the National Instruments installation, operation, or maintenance instructions; owner's modification of the product; owner's abuse, misuse, or negligent acts; and power failure or surges, fire, flood, accident, actions of third parties, or other events outside reasonable control.

Copyright

Trademarks

Patents

For patents covering National Instruments products/technology, refer to the appropriate location: **Help»Patents** in your software, the `patents.txt` file on your media, or the *National Instruments Patent Notice* at `ni.com/patents`.

WARNING REGARDING USE OF NATIONAL INSTRUMENTS PRODUCTS

(1) NATIONAL INSTRUMENTS PRODUCTS ARE NOT DESIGNED WITH COMPONENTS AND TESTING FOR A LEVEL OF RELIABILITY SUITABLE FOR USE IN OR IN CONNECTION WITH SURGICAL IMPLANTS OR AS CRITICAL COMPONENTS IN ANY LIFE SUPPORT SYSTEMS WHOSE FAILURE TO PERFORM CAN REASONABLY BE EXPECTED TO CAUSE SIGNIFICANT INJURY TO A HUMAN.

(2) IN ANY APPLICATION, INCLUDING THE ABOVE, RELIABILITY OF OPERATION OF THE SOFTWARE PRODUCTS CAN BE IMPAIRED BY ADVERSE FACTORS, INCLUDING BUT NOT LIMITED TO FLUCTUATIONS IN ELECTRICAL POWER SUPPLY, COMPUTER HARDWARE MALFUNCTIONS, COMPUTER OPERATING SYSTEM SOFTWARE FITNESS, FITNESS OF COMPILERS AND DEVELOPMENT SOFTWARE USED TO DEVELOP AN APPLICATION, INSTALLATION ERRORS, SOFTWARE AND HARDWARE COMPATIBILITY PROBLEMS, MALFUNCTIONS OR FAILURES OF ELECTRONIC MONITORING OR CONTROL DEVICES, TRANSIENT FAILURES OF ELECTRONIC SYSTEMS (HARDWARE AND/OR SOFTWARE), UNANTICIPATED USES OR MISUSES, OR ERRORS ON THE PART OF THE USER OR APPLICATIONS DESIGNER (ADVERSE FACTORS SUCH AS THESE ARE HEREAFTER COLLECTIVELY TERMED "SYSTEM FAILURES"). ANY APPLICATION WHERE A SYSTEM FAILURE WOULD CREATE A RISK OF HARM TO PROPERTY OR PERSONS (INCLUDING THE RISK OF BODILY INJURY AND DEATH) SHOULD NOT BE RELIANT SOLELY UPON ONE FORM OF ELECTRONIC SYSTEM DUE TO THE RISK OF SYSTEM FAILURE. TO AVOID DAMAGE, INJURY, OR DEATH, THE USER OR APPLICATION DESIGNER MUST TAKE REASONABLY PRUDENT STEPS TO PROTECT AGAINST SYSTEM FAILURES, INCLUDING BUT NOT LIMITED TO BACK-UP OR SHUT DOWN MECHANISMS. BECAUSE EACH END-USER SYSTEM IS CUSTOMIZED AND DIFFERS FROM NATIONAL INSTRUMENTS' TESTING PLATFORMS AND BECAUSE A USER OR APPLICATION DESIGNER MAY USE NATIONAL INSTRUMENTS PRODUCTS IN COMBINATION WITH OTHER PRODUCTS IN A MANNER NOT EVALUATED OR CONTEMPLATED BY NATIONAL INSTRUMENTS, THE USER OR APPLICATION DESIGNER IS ULTIMATELY RESPONSIBLE FOR VERIFYING AND VALIDATING THE SUITABILITY OF NATIONAL INSTRUMENTS PRODUCTS WHENEVER NATIONAL INSTRUMENTS PRODUCTS ARE INCORPORATED IN A SYSTEM OR APPLICATION, INCLUDING, WITHOUT LIMITATION, THE APPROPRIATE DESIGN, PROCESS AND SAFETY LEVEL OF SUCH SYSTEM OR APPLICATION.

Conventions

The following conventions are used in this manual:

»	The **»** symbol leads you through nested menu items and dialog box options to a final action. The sequence **File»Page Setup»Options** directs you to pull down the **File** menu, select the **Page Setup** item, and select **Options** from the last dialog box.
	This icon denotes a note, which alerts you to important information.
bold	Bold text denotes items that you must select or click in the software, such as menu items and dialog box options. Bold text also denotes parameter names.
italic	Italic text denotes variables, emphasis, a cross-reference, or an introduction to a key concept. Italic text also denotes text that is a placeholder for a word or value that you must supply.
`monospace`	Text in this font denotes text or characters that you should enter from the keyboard, sections of code, programming examples, and syntax examples. This font is also used for the proper names of disk drives, paths, directories, programs, subprograms, subroutines, device names, functions, operations, variables, filenames, and extensions.
`monospace italic`	Italic text in this font denotes text that is a placeholder for a word or value that you must supply.
Platform	Text in this font denotes a specific platform and indicates that the text following it applies only to that platform.

Contents

Chapter 5
Customizing Types

Chapter 6
Using NI Requirements Gateway with NI TestStand

Chapter 7
Using NI Requirements Gateway with MATRIXx

Chapter 8
Using NI Requirements Gateway with LabVIEW

Chapter 9
Using NI Requirements Gateway with LabWindows/CVI

Chapter 10
Using NI Requirements Gateway with Telelogic DOORS

Appendix A
Third-Party Types Overview

Appendix B
Technical Support and Professional Services

1

Introduction to NI Requirements Gateway

NI Requirements Gateway is a requirements traceability solution that links development and verification documents with formal requirements stored in documents and databases. Requirements Gateway improves the quality of the development process by effectively managing requirements traceability and impact analysis throughout the life cycle of a project.

Most engineering projects initially define high-level specifications and later define more detailed specifications. Specifications contain technical and procedural requirements that guide the product through each engineering phase. In addition, working documents, such as hardware schematics, simulation models, software source code, and test specifications and procedures, must adhere to and cover the requirements that the specifications define.

Requirements Gateway enhances project management by linking traceability information from any source. Requirements Gateway allows you to:

- Manage project documents and graphically create traceability relationships between documents
- Customize types for importing various sources of data from National Instruments and external products
- Use coverage analysis, impact analysis, and graphical views to visualize and analyze traceability relationships between documents
- Create filters to customize analysis and views
- Capture and compare project snapshots to determine changes in requirements and coverage
- Generate reports using default and custom templates

Requirements Gateway interacts with external products, as shown in Figure 1-1. You can use Requirements Gateway to complete the following types of tasks:

- Configure the specification and working documents to process
- Configure the type of traceability information to capture from each document
- Specify the traceability relationship between the specification and working documents
- Navigate between documents in Requirements Gateway and the external products

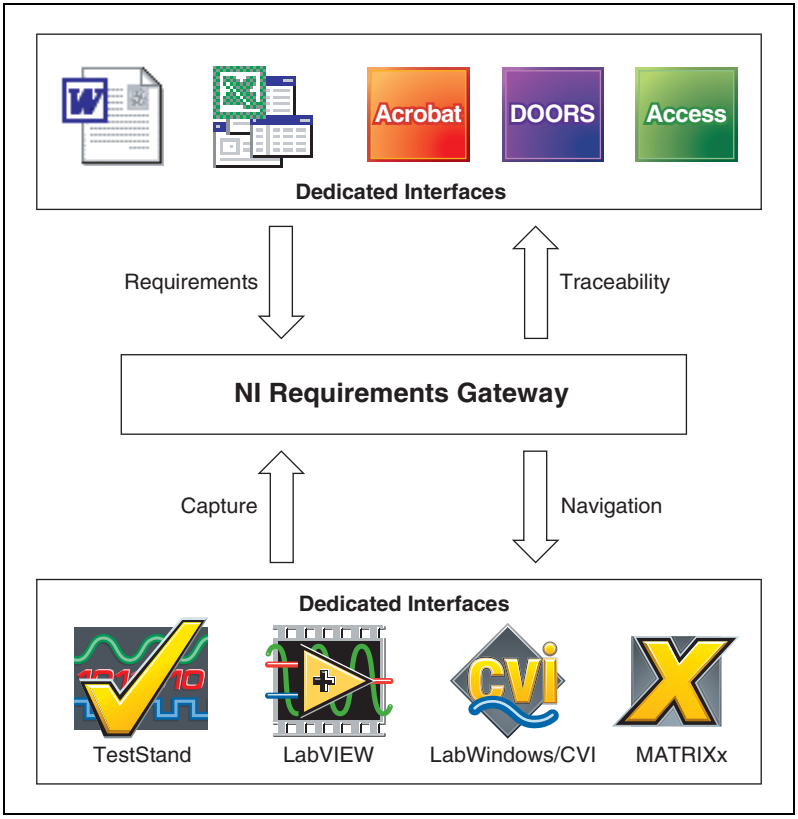


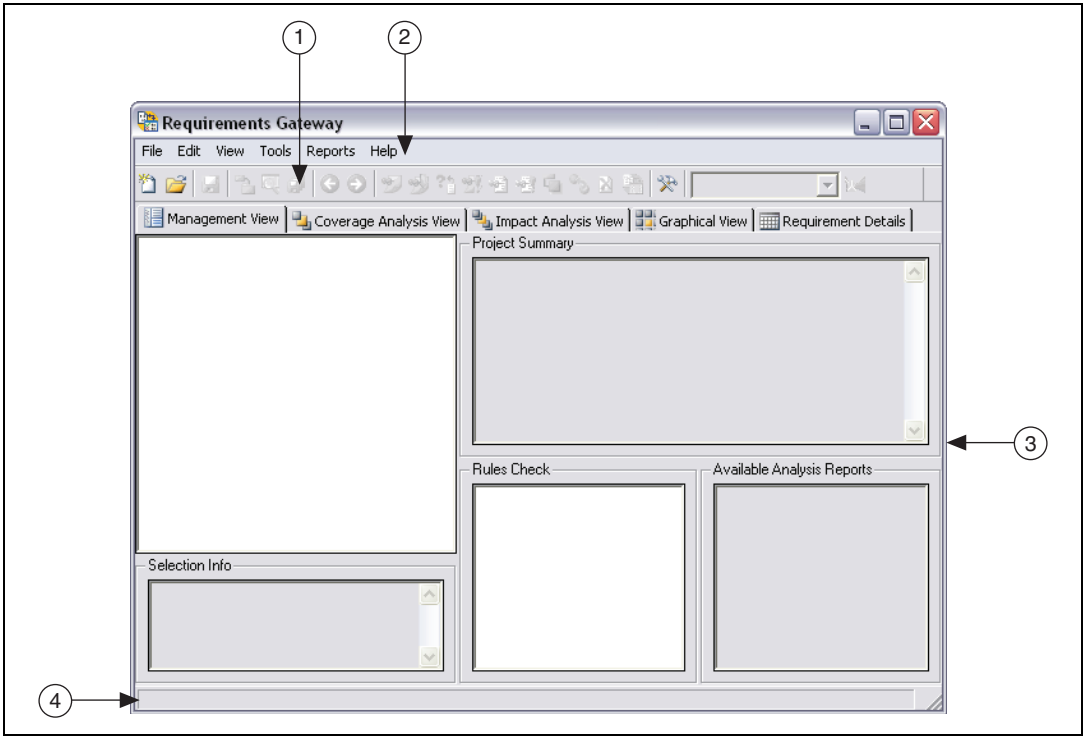
Figure 1-1. Requirements Gateway Overview

Starting Requirements Gateway

The Requirements Gateway interface includes two basic components for managing projects: a main window, in which you perform coverage and impact analysis, and a Configuration dialog box, in which you create and modify project documents, types, filters, and other options.

Main Window

When you launch Requirements Gateway, the main window contains four basic components, as shown in Figure 1-2.



1 Toolbar	3 Project Workspace
2 Menu Bar	4 Status Bar

Figure 1-2. Requirements Gateway Main Window

Toolbar

The toolbar contains options for common tasks and includes six sections, as shown in Figure 1-3.

1	Standard	3	Navigation	5	Filtering
2	Configuration	4	View Options	6	Third Party

Figure 1-3. Requirements Gateway Toolbar

- **Standard**—Contains buttons for creating, loading, and saving project files.
- **Configuration**—Contains buttons for editing projects, types, and snapshots.
- **Navigation**—Contains buttons to apply navigation commands previously performed within the coverage information of the Coverage Analysis View and Impact Analysis View.
- **View Options**—Contains buttons to control the visible traceability elements in the Management View, Coverage Analysis View, and Impact Analysis View.
- **Filtering**—Contains the Filter ring control to configure and apply filters that specify the conditions for including requirements in an analysis or view.
- **Third Party**—Contains buttons for exporting traceability information to Telelogic DOORS and IBM Rational RequisitePro. You must install and have access to DOORS and RequisitePro on the same computer as Requirements Gateway for the button to become available.

Menu Bar

The menu bar contains the File, Edit, View, Tools, Reports, and Help menus. Browse the menus in the menu bar of the main window to familiarize yourself with their contents.

Project Workspace

The project workspace is the main part of the Requirements Gateway application and displays the project information and analysis for the loaded project. The project workspace contains tabs that display the content of the project in different views. Each view contains one or more panes.

The project workspace contains the following views:

- **Management View**—Displays the documents in the project, the elements of each document, and a summary of coverage information for the project.
- **Coverage Analysis View**—Displays one level of covering elements at the immediate downstream level, `N–1`, for a selected element of a document, and one level of covered elements at the immediate upstream level, `N+1`, from other documents as the project defines.
- **Impact Analysis View**—Displays all levels of covering elements at downstream levels, `N–m`, for a selected element of a document, and all levels of covered elements at upstream levels, `N+p`, from other documents as the project defines.
- **Graphical View**—Displays each document graphically using a tree view, in which lines connect requirement elements in documents and covering elements in other documents.
- **Requirement Details**—Displays each requirement and its attributes for a document in a table.

Status Bar

The status bar displays common application information, such as status while performing analysis or a brief explanation of a menu item when you hover over the item in the menu bar.

Refer to the *NI Requirements Gateway Help* for more information about the components of the main window.

Configuration Dialog Box

Requirements Gateway launches the Configuration dialog box, as shown in Figure 1-4, when you select a menu item or toolbar button to complete the following tasks:

- Create and configure a new project or edit an existing project
- Create and manage types
- Create and manage snapshots
- Create and manage advanced analysis display filters

- Customize reports
- Test expressions that types use to capture traceability information from source documents
- Specify additional project options

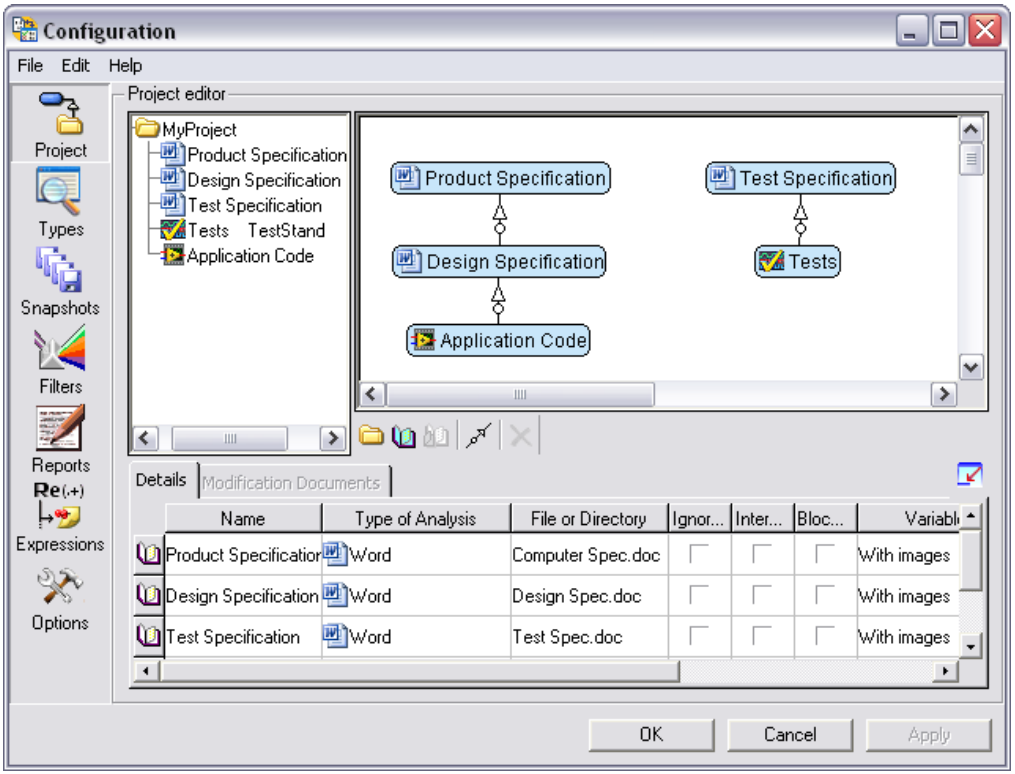


Figure 1-4. Requirements Gateway Configuration Dialog Box

The Configuration dialog box contains the following panes:

- **Project**—Use this pane to configure the project by specifying the documents to include, the type of each document, and the covering relationship between documents. Refer to Chapter 2, *Managing Requirements*, and to the *NI Requirements Gateway Help* for more information about configuring projects.
- **Types**—Use this pane to create new types or customize existing types for the project. Refer to Chapter 5, *Customizing Types*, of this manual and to Chapter 2, *Customizing Types and Type Elements*, of the *NI Requirements Gateway Customization Guide* for more information about customizing types.

- **Snapshots**—Use this pane to create, manage, and compare snapshots of the project. A snapshot is a stored image of analysis results that you can use to trace modifications, additions, or deletions of information throughout the lifetime of the project. Refer to the *NI Requirements Gateway Help* for more information about using snapshots.
- **Filters**—Use this pane to define custom filters to analyze or display certain requirements from documents that meet specific criteria. You can enable filters using the Filter ring control on the toolbar in the Requirements Gateway main window. Refer to the *NI Requirements Gateway Help* for more information about using filters.
- **Reports**—Use this pane to define new custom reports. You can use the options in the Reports menu on the toolbar in the main window to generate a default or custom report. Refer to Chapter 4, *Generating Reports*, and to the *NI Requirements Gateway Help* for more information about generating reports. Refer to Chapter 4, *Customizing Reports*, of the *NI Requirements Gateway Customization Guide* for more information about creating custom reports.
- **Expressions**—Use this pane to test regular expressions. You can specify source text and a regular expression, and the pane displays the captured text returned by the regular expression. Refer to Chapter 3, *Using and Testing Regular Expressions*, of the *NI Requirements Gateway Customization Guide* for more information about testing regular expressions.
- **Options**—Use this pane to set the default font for the text in the application, set the password for the project, define environmental variables, and specify other miscellaneous settings for the application.

Refer to the *NI Requirements Gateway Help* for more information about the options in the Configuration dialog box panes and menus.

Requirements Gateway Directory Structure

To comply with Windows Vista restrictions on writing to the `Program Files` directory and to improve usability for Windows 2000/XP users who do not have permission to write to the `Program Files` directory, Requirements Gateway installs files in the following directories:

- **<Requirements Gateway>**—Located by default at `C:\Program Files\National Instruments\Requirements Gateway x.x` on Windows 2000/XP and Windows Vista (32-bit) and at `C:\Program Files (x86)\National Instruments\Requirements Gateway x.x` on Windows Vista (64-bit).

  The `<Requirements Gateway>` directory is the location where you installed Requirements Gateway on the computer and contains the read-only Requirements Gateway program files.

- **<Requirements Gateway Public>**—Located by default at `C:\Documents and Settings\All Users\Documents\National Instruments\Requirements Gateway x.x` on Windows 2000/XP and at `C:\Users\Public\Documents\National Instruments\Requirements Gateway x.x` on Windows Vista.

  The `<Requirements Gateway Public>` directory contains the project directories and files, modifications, customizations, and other files you can directly edit, including the tutorials in this manual and other examples. Refer to the *NI Requirements Gateway Release Notes* for more information about the subdirectories of the `<Requirements Gateway Public>` directory.

Refer to the *NI Requirements Gateway Help* for information about the types of files located in these directories.

2

Managing Requirements

You can use the different windows and views in Requirements Gateway to create a project, add existing documents to the project, and review the content of the documents.

A project specifies the documents Requirements Gateway analyzes and displays. A project also specifies the type to use for each document. A type defines how Requirements Gateway completes the following tasks:

- Selects external files that represent a document
- Reads the content of the external files
- Interprets the content as elements for managing requirements
- Displays the elements of the document

Requirements Gateway contains a set of predefined types for the following data sources:

- LabVIEW VIs
- NI TestStand sequence files and XML reports
- LabWindows™/CVI™ source code and function panel files
- MATRIXx SystemBuild catalogs
- Microsoft Word documents
- Microsoft Excel spreadsheets
- Microsoft Access databases
- Microsoft Visio project files
- Telelogic DOORS databases
- IBM Rational RequisitePro databases
- Acrobat PDF files
- Generic text files
- Generic source code files

Creating a Project

Complete the following steps to start the Requirements Gateway application and create a new project.

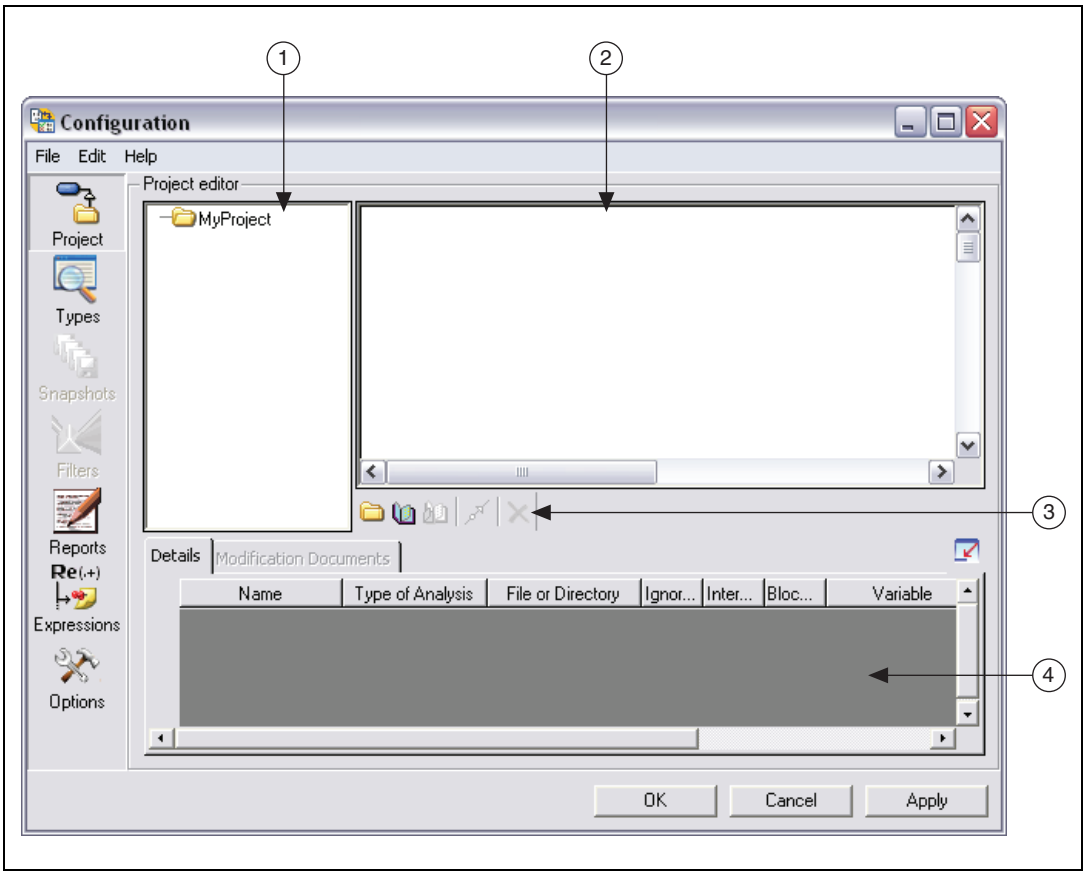
Note The exercises in this manual modify the tutorial files located in the `<Requirements Gateway Public>\Tutorial` directory. You can restore the tutorial files to their original state by copying the files from the `<Requirements Gateway Public>\Tutorial\Original` directory into the `<Requirements Gateway Public>\Tutorial` directory.

1. **(Windows 2000/XP)** Select **Start»All Programs»National Instruments»Requirements Gateway *x.x*»Requirements Gateway** to launch Requirements Gateway and display the main window.

   (Windows Vista) Select **Start»Programs»National Instruments» Requirements Gateway x.x»Requirements Gateway Programs** to launch Requirements Gateway and display the main window.

2. Select **File»New** or click the **New** button, shown at left, on the toolbar in the main window to launch the Create a New Project and Save As dialog box, and navigate to the `<Requirements Gateway Public>\Tutorial` directory.
3. Enter `MyProject` in the File name control and click **Save**.

Requirements Gateway creates a new project file in memory and displays the Project pane of the Configuration dialog box, as shown in Figure 2-1. A project file defines the documents that Requirements Gateway reads, the type of each document, and the covering relationship between the documents. If you do not add documents to the project file, Requirements Gateway does not save the project file to disk and discards the file in memory. After you add documents to the project and apply the changes, Requirements Gateway saves the project file in the `<Requirements Gateway Public>\Tutorial` directory.



1	Project Overview	3	Action Buttons
2	Traceability Description Zone	4	Document Details

Figure 2-1. New Project on the Project Pane of the Configuration Dialog Box

The Project pane contains the following sections:

- **Project Overview**—Tree view of all the documents and folders the root project directory contains. Each document in the project tree includes the document name and an icon to indicate the document type.
- **Traceability Description Zone**—Graphical overview of the documents and folders in the project. You can place and arrange documents and folders, and you can add covering links between the project documents to help you visualize the relationship between the documents.

- **Action Buttons**—Contains options for adding folders, documents, modification documents, and covering links to a project, and options for removing selected items from the project. Refer to the *NI Requirements Gateway Help* for more information about adding folders and modification documents to a project.
- **Document Details**—Contains options for configuring the settings of project documents, such as the name, type of analysis, file source, variables, and other settings for a selected document.

Refer to the *NI Requirements Gateway Help* for more information about the options on the Project pane of the Configuration dialog box.

Adding a Document

Complete the following steps to add a specification document to the project you created in the *Creating a Project* section of this chapter.

1. Click the **Add a document** button, shown at left. The cursor automatically moves to the Traceability Description Zone and outlines a document object. Click within the Traceability Description Zone to place the document.

   When you place the document, Requirements Gateway adds the document to the Project Overview pane. The Document Details pane displays the settings for the document on the Project Overview pane.
2. On the Document Details pane, click in the **Name** column to select the **Document1** text. Enter `Product Specification` and press <Enter> to rename the document. The name in the document object now displays the new name.
3. Click in the **Type of Analysis** column and select **Text** from the ring control to analyze the document using the Text type.

4. Click in the **File or Directory** column. The File Browse button, shown at left, becomes visible on the right side of the control. Click the **File Browse** button and select `<Requirements Gateway Public>\Tutorial\ProductSpec.txt`. Figure 2-2 shows the Configuration dialog box after you add the Product Specification document.

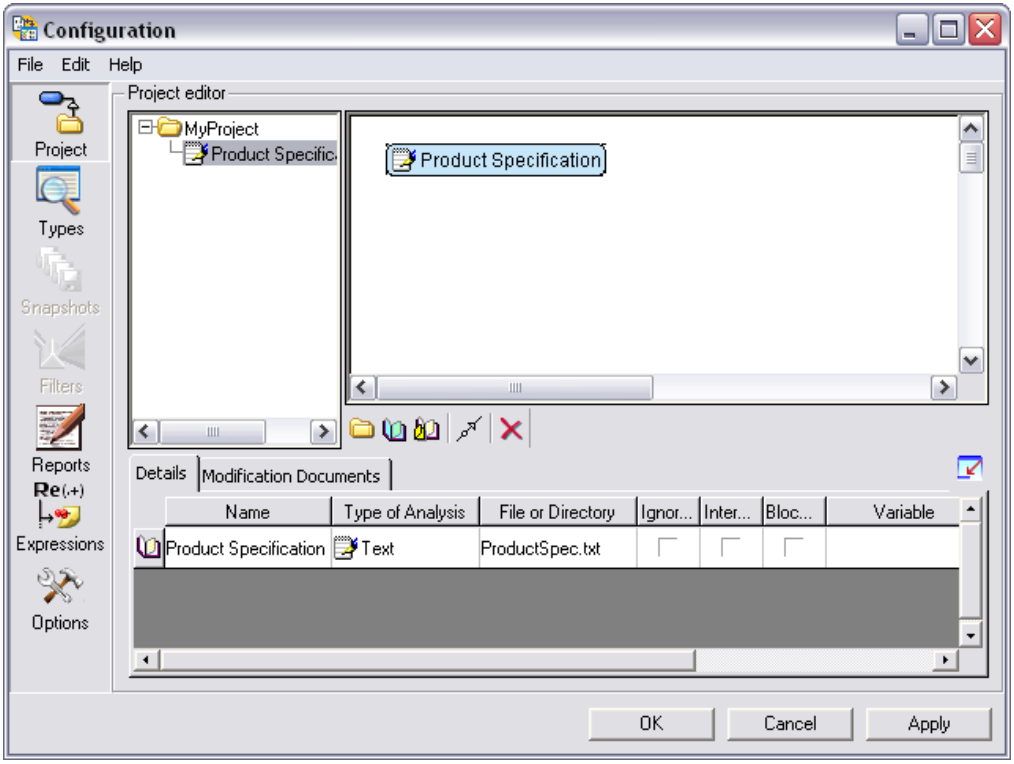


Figure 2-2. Adding a Document to a New Project

Adding a Covering Document

A covering document is a document that contains references to requirements that another document defines. A covering document can be a source document from any supported external tool, including code files such as a TestStand sequence file, a LabVIEW VI, or a C file. For example, a step in a TestStand sequence might cover requirements stated in a Specification text document.

Note This tutorial adds a covering text document. Refer to Chapters 6 through 10 of this manual for tutorials on adding TestStand, MATRIXx, LabVIEW, LabWindows/CVI, and Telelogic DOORS documents to a Requirements Gateway project. Refer to the individual coupling documents for more information about adding documents from each supported external tool.

Complete the following steps to add a new document to the project that covers the Product Specification document.

1. Click the **Add a document** button to add a second document in the Traceability Description Zone. Place the document below the Product Specification document.
2. On the Document Details pane, enter `Covering Specification` in the Name control, select **Text** from the Type of Analysis ring control, and browse to `<Requirements Gateway Public>\Tutorial\CoveringSpec.txt` in the **File or Directory** control.

3. Click the **Add a cover** button, shown at left. The cursor moves to the Traceability Description Zone. Click the **Covering Specification** document and then click the **Product Specification** document. An arrow appears between the two documents, as shown in Figure 2-3. The arrow indicates that the Covering Specification document covers the Product Specification document.

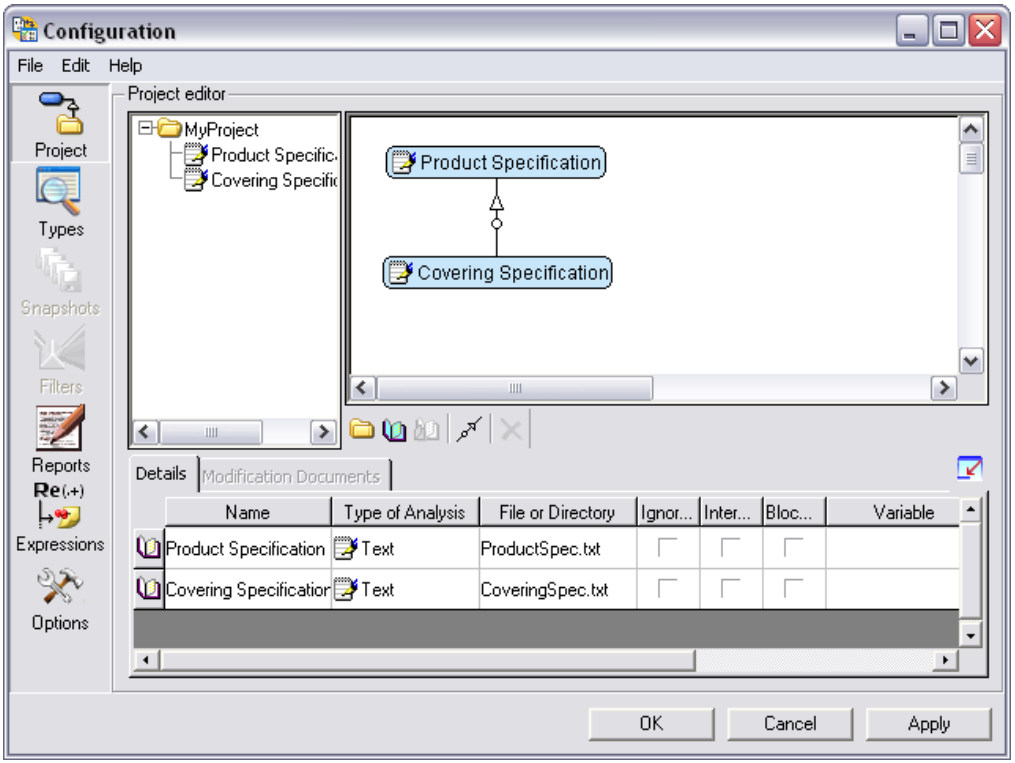


Figure 2-3. Covering Document in Project

You can also select the downstream document, press the mouse button and hover over the upstream document, and release the mouse button to add a covering document.

4. Click **OK** to close the Configuration dialog box.

Using the Management View

After you close the Configuration dialog box in step 4 of the *Adding a Covering Document* section, the Management View appears in the main window, as shown in Figure 2-4.

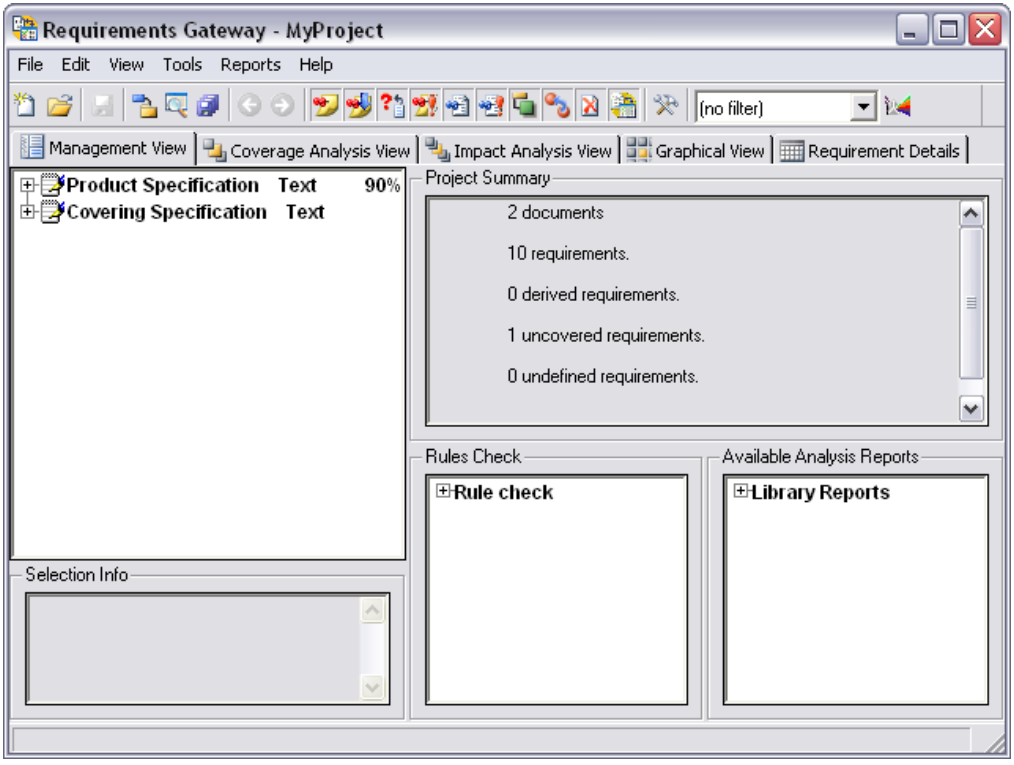
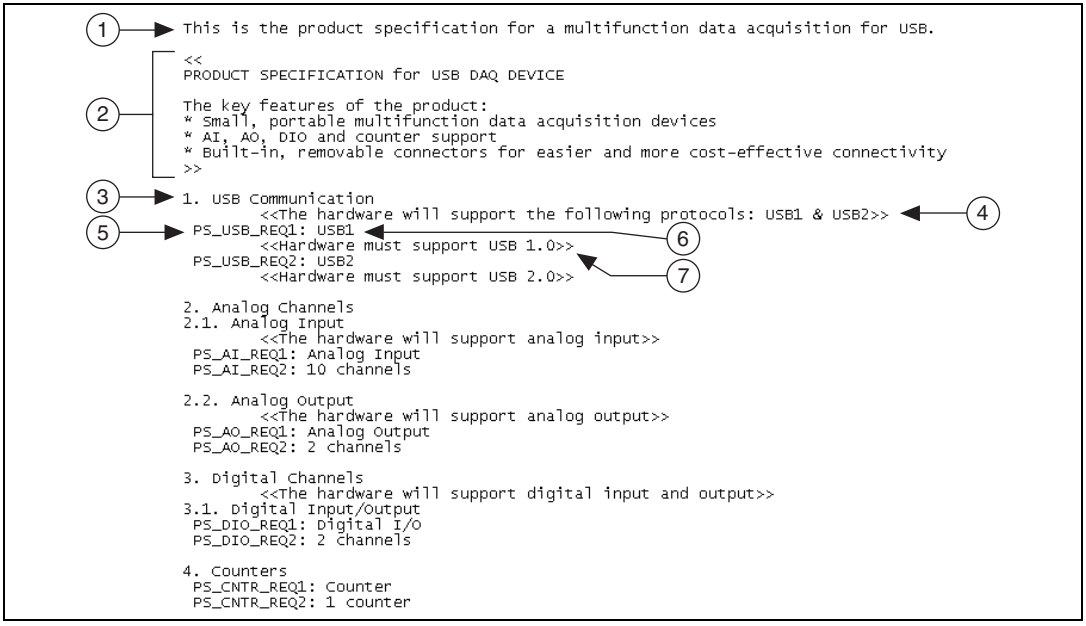


Figure 2-4. Management View

The tree view pane in the Management View includes a list of the documents the project defines and includes a root node for each document added to the project. The Project Summary pane indicates that the project has two documents with ten defined requirements and one uncovered requirement. Refer to the *NI Requirements Gateway Help* for more information about the panes in the Management View.

Complete the following steps to familiarize yourself with the Management View and the documents you included in the project.

1. Right-click the **Product Specification** document on the tree view pane and select **Navigate** from the context menu. Requirements Gateway displays `ProductSpec.txt` in the external application associated with `.txt` files for your computer, as shown in Figure 2-5. The default application for Windows is Microsoft Notepad.



1	General Text	3	Section	5	Requirement ID	7	Requirement Text
2	Document Text	4	Section Text	6	Requirement Label		

Figure 2-5. ProductSpec.txt Document

2. Review the content of the text file in the external application. `ProductSpec.txt` specifies ten requirements.

   The file contains sections, requirements, and text elements. The default Text type in Requirements Gateway interprets the elements of the file in the following ways:

   - **Section**—Defined by numeric heading characters such as `2.1`. The text after the numeric heading is the text of the section.
   - **Requirement**—Defined by an identifier that contains a set of arbitrary characters, followed by the characters, `REQ`, and ending with a numeric value. The label for the requirement is located after the identifier and is delimited by the colon character.

- **Text**—When delimited by the << and >> characters, the text is associated with the previously specified element. If an initial text element is specified at the beginning of the file, the text is associated with the document, otherwise the text is associated with a section or traceability element, such as a requirement.

3. Exit the application that displays `ProductSpec.txt`.
4. In Requirements Gateway, right-click the **Covering Specification** document on the tree view pane and select **Navigate** from the context menu. Requirements Gateway displays `CoveringSpec.txt` in an external application.
5. Review the content of the text file in the external application. `CoveringSpec.txt` contains section and text elements. Instead of requirements, the file contains references to requirements. A reference is defined by the prefix `[Covers:`, followed by a set of characters that represent the requirement identifier, and completed with a closing bracket character.

   `CoveringSpec.txt` specifies nine requirement references. Notice that the 1.2. USB 2.0 Speeds section of the document does not contain a requirement reference.
6. Exit the application that displays `CoveringSpec.txt`.
7. In Requirements Gateway, select the **Product Specification** document on the tree view pane. The tree view pane displays the percentage of covered requirements for a document, which is 90% for this document. The Selection Info pane in the lower left corner of the Management View indicates that the document defines ten requirements, and one of the requirements is uncovered.
8. Expand the child elements of the **Product Specification** document on the tree view pane of the Management View.

Note Press <Shift> while clicking the plus icon to expand the parent element and all its child elements.

The tree view pane displays the section headings from the text file as parent elements and displays the requirements as child elements, as shown in Figure 2-6.

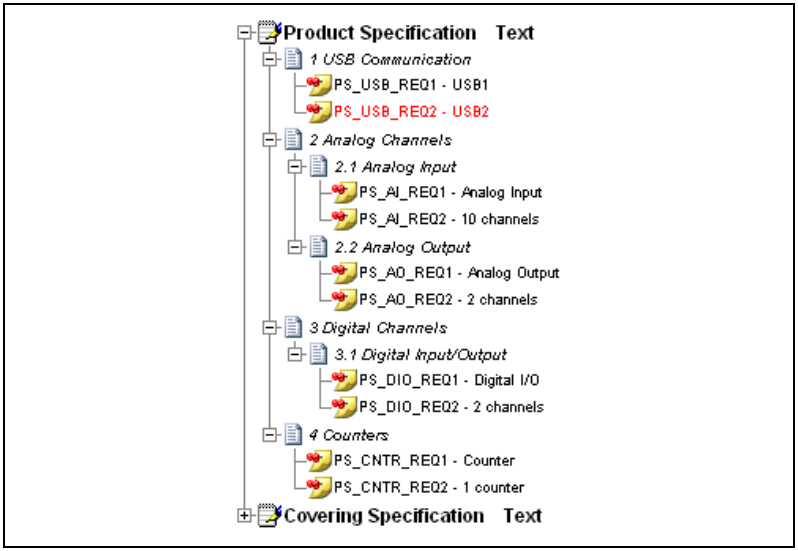


Figure 2-6. Product Specification in Management View

9. Select the **PS_USB_REQ1** element on the tree view pane. The PS_USB_REQ1 element represents a requirement the document specifies. The Selection Info pane indicates that the PS_USB_REQ1 element is a requirement and that the requirement is covered.
10. Select the **PS_USB_REQ2** element on the tree view pane. The Selection Info pane indicates that the PS_USB_REQ2 element is also a requirement but the requirement is not covered. Requirements Gateway highlights the element name in red.
11. Collapse the **Product Specification** document on the tree view pane of the Management View.
12. Select the **Covering Specification** document on the tree view pane. The Selection Info section indicates that the document contains references to nine requirements.

13. Expand the child elements of the **Covering Specification** document on the tree view pane, as shown in Figure 2-7.

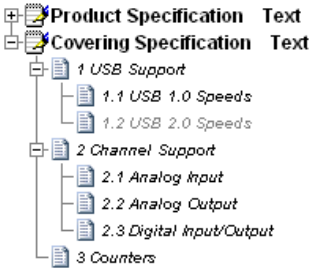


Figure 2-7. Covering Specification in Management View

14. Select the **1.1 USB 1.0 Speeds** element. The element represents a section that the document specifies. The Selection Info pane indicates that the section contains a reference to one requirement.
15. Select the **1.2 USB 2.0 Speeds** element. The Selection Info pane for this element indicates that the section is empty and contains no references to any requirements.
16. Collapse the **Covering Specification** document on the tree view pane of the Management View.
17. Expand the elements on the **Rules Check** pane of the Management View, as shown in Figure 2-8. The Rules Check pane contains a summary of the rules flagged from analyzing the project. For this project, the section indicates that the PS_USB_REQ2 requirement is uncovered. Refer to the *NI Requirements Gateway Help* for more information about the rule checks.

Figure 2-8. Rules Check Section of the Management View

3

Analyzing Requirements

The Management View summarizes the documents in the project, the structure and requirement-related information in each document, and any rules related to the project.

Use the Coverage Analysis View, the Impact Analysis View, and the Graphical View to obtain details about the requirements and the references that cover the requirements.

Using the Coverage Analysis View

The Coverage Analysis View displays traceability information from only the immediate downstream and upstream documents from a selected element of a document.

The upper half of the Coverage Analysis View contains the following columns:

- **Upstream Coverage Information**—For a selected element in the Selection column, the Upstream Coverage Information column contains a tree view that displays the immediate upstream elements that the selected element covers.
- **Selection**—Displays the content of the documents in the project. When you select an element in the Selection column, the information in the Upstream Coverage Information and Downstream Coverage Information columns changes depending on the element selected.
- **Downstream Coverage Information**—For a selected element in the Selection column, the Downstream Coverage Information column contains a tree view that displays the immediate downstream elements that cover the selected element.

The lower half of the Coverage Analysis View contains the following tabs. Each tab displays details about the selected element in the upper half of the view.

- **Texts and Reference Attributes**—Displays the text for the selected element and any reference attributes for references that link the selection in the Selection column to the corresponding element in the

Upstream Coverage Information or Downstream Coverage Information column.

- **Attributes**—Displays the attributes for the selected element.
- **Messages**—Displays helpful information, including rule violation details, for the selected element in the Selection column.

Complete the following steps to use the Coverage Analysis View to analyze requirement coverage for the project you created in Chapter 2, *Managing Requirements*.

1. Open `<Requirements Gateway Public>\Tutorial\MyProject.rqtf` if it is not already open.
2. Click the **Coverage Analysis View** tab in the main window. Check to make sure that the Product Specification and the Covering Specification documents in the Selection column are collapsed, as shown in Figure 3-1.

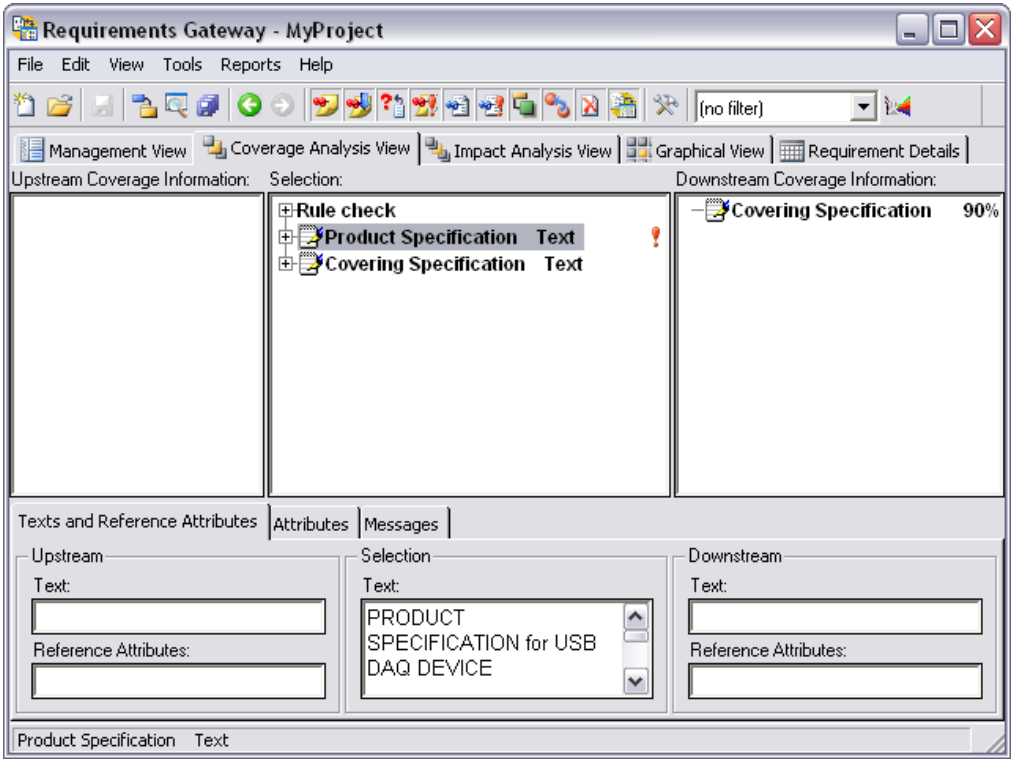


Figure 3-1. Coverage Analysis View

3. Select the **Product Specification** document in the Selection column. The Downstream Coverage Information column indicates that the Covering Specification document covers 90% of the requirements in the Product Specification document.
4. Expand the child elements of the **Product Specification** document in the Selection column.
5. Select the **PS_USB_REQ1** requirement, as shown in Figure 3-2.

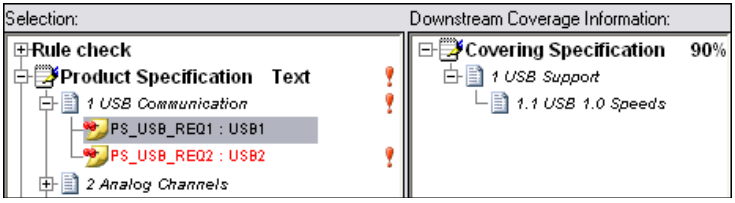


Figure 3-2. PS_USB_REQ1 Selected in Coverage Analysis View

The Downstream Coverage Information column displays the 1.1 USB 1.0 Speeds section as a covering element because this section of the Covering Specification document contains a reference to the requirement. The Downstream Coverage Information column also includes the parent 1 USB Support section of the covering element.

6. Select the **PS_USB_REQ2** requirement, as shown in Figure 3-3.

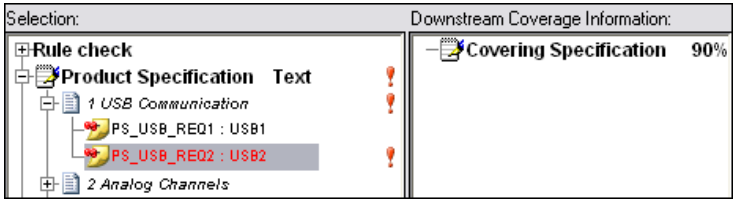


Figure 3-3. PS_USB_REQ2 Selected in Coverage Analysis View

The Downstream Coverage Information column does not display any covering elements because the Covering Specification document does not contain a reference to the requirement.

7. Hover over the **exclamation** icon, shown at left, to the right of the PS_USB_REQ2 requirement to display a tooltip that contains the text `1 uncovered requirement`. The Selection column displays the exclamation icon for an uncovered requirement and all the parent elements of the uncovered requirement.

8. Select the **2 Analog Channels** section, as shown in Figure 3-4.

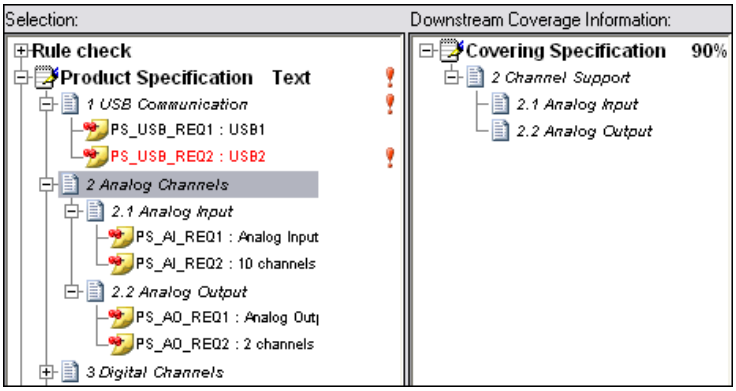


Figure 3-4. 2 Analog Channels Section Selected in Coverage Analysis View

The child elements of the Analog Channels section contain four requirements: PS_AI_REQ1, PS_AI_REQ2, PS_AO_REQ1, and PS_AO_REQ2. The Downstream Coverage Information column displays the 2.1 Analog Input and the 2.2 Analog Output sections as covering elements. In the Covering Specification document, the 2.1 Analog Input section contains references to the PS_AI_REQ1 and PS_AI_REQ2 requirements, and the 2.2 Analog Output section contains references to the PS_AO_REQ1, and PS_AO_REQ2 requirements.

9. Double-click the **2.1 Analog Input** section of the Covering Specification document in the Downstream Coverage Information column. Requirements Gateway navigates to the 2.1 Analog Input section of the Covering Specification document in the Selection column, as shown in Figure 3-5.

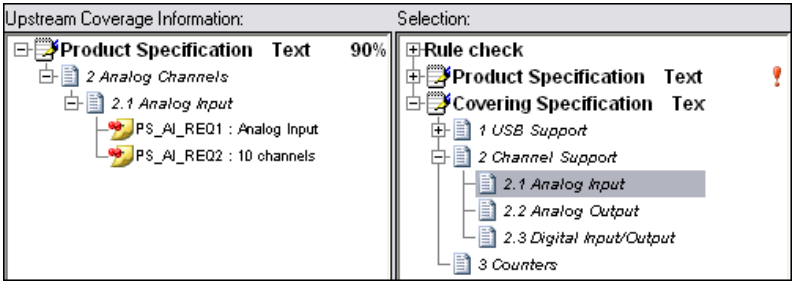


Figure 3-5. 2.1 Analog Input Section Selected in Coverage Analysis View

The Upstream Coverage Information column indicates that the Covering Specification document covers 90% of the requirements in the Product Specification document, and that the references from the 2.1 Analog Input section of the Covering Specification document specifically cover the PS_AI_REQ1 and PS_AI_REQ2 requirements.

10. Expand the **1 USB Support** section of the Covering Specification document in the Selection column to view the 1.2 USB 2.0 Speeds element.
11. Double-click the **1.2 USB 2.0 Speeds** element in the Selection column to launch `CoveringSpec.txt` in an external application.
12. Edit the file by adding a reference to the PS_USB_REQ2 requirement below the 1.2. USB 2.0 Speeds section, as shown in Figure 3-6.

```
1. USB Support
1.1. USB 1.0 Speeds
     <<Hardware supports USB 1.0>>
     [Covers: PS_USB_REQ1]

1.2. USB 2.0 Speeds
     <<Hardware supports USB 2.0>>
     [Covers: PS_USB_REQ2]

2. Channel Support
2.1. Analog Input
     <<Hardware supports analog input>>
     [Covers: PS_AI_REQ1]
     <<10 channels>>
     [Covers: PS_AI_REQ2]
```

Figure 3-6. PS_USB_REQ2 Reference in Covering Document

13. Save the changes to `CoveringSpec.txt` and exit the application.

14. Return to Requirements Gateway, which prompts you with a dialog box indicating that the Covering Specification document was modified. Click **Yes** to reload the file. Requirements Gateway analyzes the new document and updates the view, as shown in Figure 3-7.

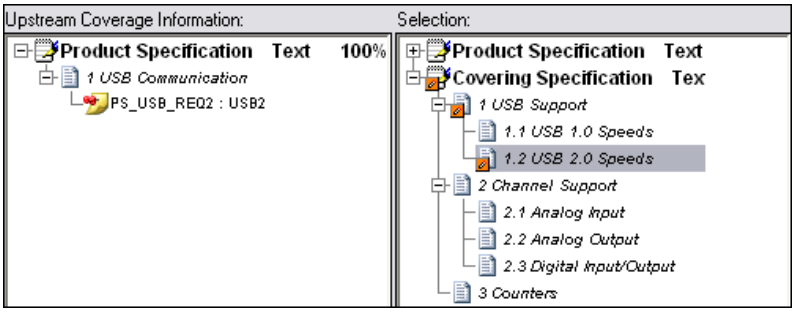


Figure 3-7. Covering Specification Document After Reload

Note When you reload a project after making changes, Requirements Gateway highlights the added, deleted, modified, and moved elements with orange icons. The icons are temporary and disappear when you reload the project again without making any additional changes. If you want to permanently keep the change information in the project, use snapshots or marks. Refer to the *NI Requirements Gateway Help* for more information about the orange icons, snapshots, and marks.

The Upstream Coverage Information column now indicates that the Covering Specification document covers 100% of the requirements in the Product Specification document. The Upstream Coverage Information column also indicates that the 1.2 USB 2.0 Speeds element now covers the PS_USB_REQ2 requirement. The icon for the 1.2 USB 2.0 Speeds element and its parent elements indicate a change occurred.

Refer to the *NI Requirements Gateway Help* for more information about using the Coverage Analysis View.

Using the Impact Analysis View

The Impact Analysis View contains the same columns and tabs as the Coverage Analysis View, but displays traceability information from all downstream and upstream documents.

Adding a Second Downstream Document

Complete the following steps to add a third document to the project you created in Chapter 2, *Managing Requirements*.

1. Open `<Requirements Gateway Public>\Tutorial\MyProject.rqtf` if it is not already open.
2.  Select **File»Edit Project** or click the **Edit Project** button, shown at left, on the toolbar in the main window to launch the Project pane of the Configuration dialog box.
3. Select the **Covering Specification** document in the Traceability Description Zone.
4. On the Document Details pane, enter `Design Specification` in the Name control, select **Text** in the Type of Analysis ring control, and browse to `<Requirements Gateway Public>\Tutorial\DesignSpec.txt` in the **File or Directory** control.
5. Click the **Add a document** button to add a third document in the Traceability Description Zone. Place the document below the Design Specification document.
6. On the Document Details pane, enter `Test Specification` in the Name control, select **Text** in the Type of Analysis ring control, and browse to `<Requirements Gateway Public>\Tutorial\TestSpec.txt` in the **File or Directory** control.
7. Click the **Add a cover** button to begin adding a covering link. Click the **Test Specification** document and then click the **Design Specification** document to create an arrow between the two documents, as shown in Figure 3-8.

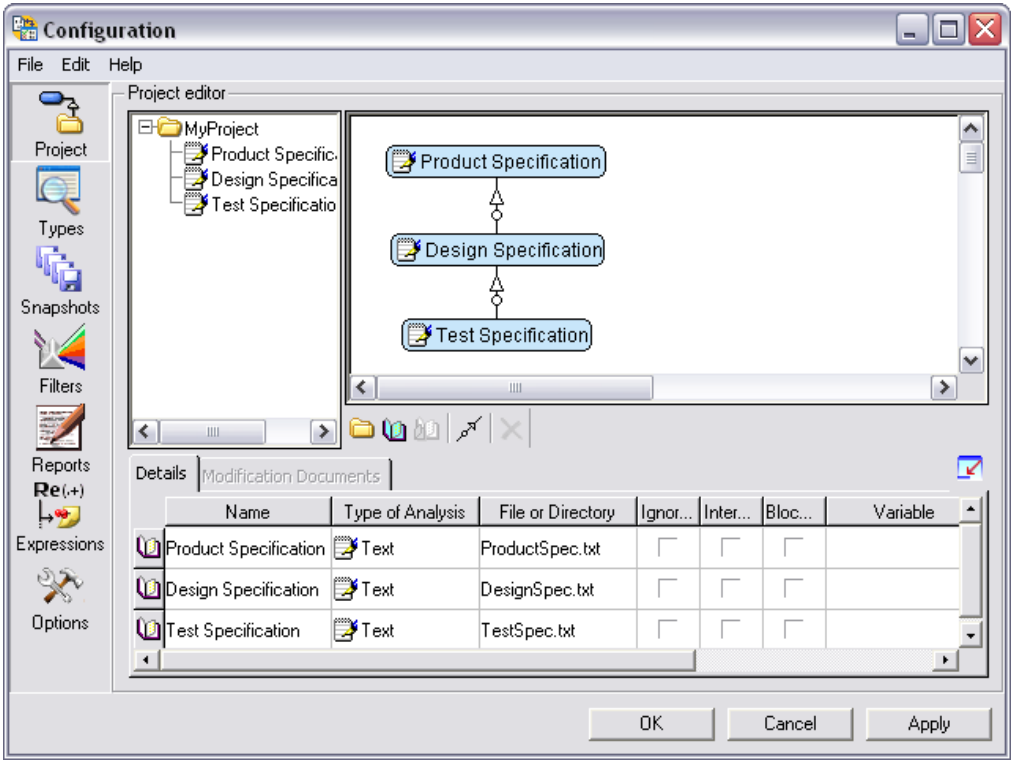


Figure 3-8. Two Levels of Covering Documents in Project

8. Click **OK** to close the Configuration dialog box.
9. Click **Yes** when Requirements Gateway prompts you to reanalyze the project. Your new project is now properly configured to analyze the new documents.

Reviewing the New Documents

Complete the following steps to review the content of the two new documents.

1. In the Requirements Gateway main window, click the **Impact Analysis View** tab.
2. Double-click the **Design Specification** document in the Selection column to display `DesignSpec.txt` in an external application.
3. Review the content of the text file in the external application. `DesignSpec.txt` contains similar sections and text elements as those in `CoveringSpec.txt`, but `DesignSpec.txt` contains 25 additional requirements that references in the Test Specification document cover.

Figure 3-9 shows the 1.1 USB 1.0 Speeds section of the file, which contains a reference to the PS_USB_REQ1 requirement and contains two additional derived requirements, DS_USB1_REQ1 and DS_USB1_REQ2. A derived requirement is a requirement that a document defines but that is not directly associated with the coverage of an upstream document.

```
1. USB Support
1.1. USB 1.0 Speeds
     <<Hardware supports USB 1.0>>
     [Covers: PS_USB_REQ1]

DS_USB1_REQ1: Low Speed
     <<1.5 Mbps>>
DS_USB1_REQ2: High Speed
     <<12 Mbps>>

1.2. USB 2.0 Speeds
     <<Hardware supports USB 2.0>>
```

Figure 3-9. Derived Requirements in Design Specification

To associate a requirement with the coverage of an upstream document, you must specify the requirement immediately before the reference that covers the upstream document, as shown in Figure 3-10. Such requirements are referred to as non-derived requirements.

```
1.2. USB 2.0 Speeds
     <<Hardware supports USB 2.0>>

DS_USB2_REQ1: Low Speed
     <<1.5 Mbps>>
     [Covers: PS_USB_REQ2]
DS_USB2_REQ2: Med Speed
     <<12 Mbps>>
     [Covers: PS_USB_REQ2]
DS_USB2_REQ3: High Speed
     <<480 Mbp>>
     [Covers: PS_USB_REQ2]
```

Figure 3-10. Non-Derived Requirements in Design Specification

Requirements Gateway displays derived and non-derived requirements in the Impact Analysis View.

4. Exit the application that displays `DesignSpec.txt`.
5. Double-click the **Test Specification** document in the Selection column to launch `TestSpec.txt` in an external application.

6. Review the content of the text file in the external application. `TestSpec.txt` contains 25 references for the requirements the Design Specification document specifies.
7. Exit the application that displays `TestSpec.txt`.

Performing Impact Analysis

Complete the following steps to use the Impact Analysis View to analyze the new documents.

1. Expand the **Design Specification** document in the Selection column to display the derived requirements, DS_USB1_REQ1 and DS_USB1_REQ2, and the non-derived requirements, DS_USB2_REQ1, DS_USB2_REQ2, and DS_USB2_REQ3, as shown in Figure 3-11. Requirements Gateway displays different icons for derived and non-derived requirements and displays a navigation arrow icon to the right side of the requirement and all the parent elements of the requirement to indicate the derived requirements.

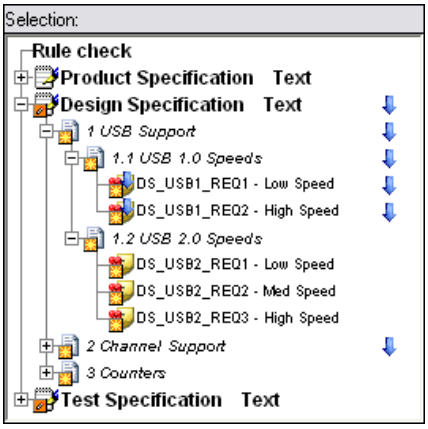


Figure 3-11. Derived and Non-Derived Requirements

2. Expand the **Product Specification** document in the Selection column and select the **PS_USB_REQ1** requirement. The Downstream Impact Information column displays only the covering 1.1 USB 1.0 Speeds section from the Design Specification document, as shown in Figure 3-12, because the 1.1 USB 1.0 Speeds section does not contain any non-derived requirements that the Test Specification document covers.

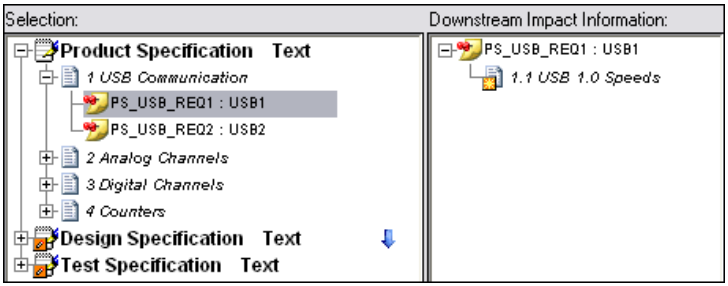


Figure 3-12. PS_USB_REQ1 Downstream Impact Analysis

3. Select the **PS_USB_REQ2** requirement. The Downstream Impact Information column displays the non-derived requirements from the covering Design Specification document and the sections from the Test Specification document that cover the non-derived requirements, as shown in Figure 3-13.

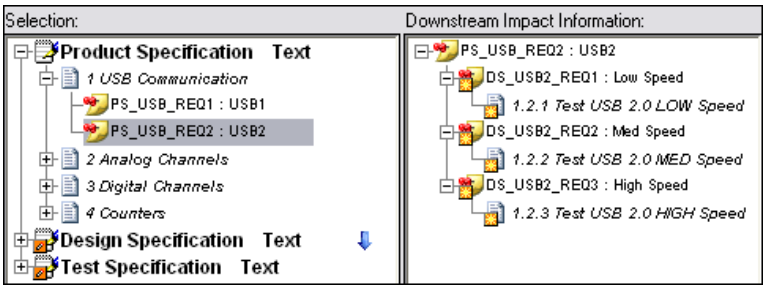


Figure 3-13. PS_USB_REQ2 Downstream Impact Analysis

Refer to the *NI Requirements Gateway Help* for more information about using the Impact Analysis View.

Using the Graphical View

The Graphical View displays each document as an object with its traceability elements displayed in a tree view within the object. Thin, black lines represent covering references between requirements elements of a document and elements in another document. Dotted lines between objects represent links other than references, such as support, allocation, or validation information for requirements. You can also reposition documents, adjust the width of the documents, pan, zoom, and resize the containing page.

Complete the following steps to graphically view project documents.

1. Open `<Requirements Gateway Public>\Tutorial\MyProject.rqtf` if it is not already open.
2. Click the **Graphical View** tab.
3. Select the **PS_USB_REQ1** requirement in the Product Specification document. The Graphical View highlights the PS_USB_REQ1 requirement, the covering 1.1 USB 1.0 Speeds section, and the line between the elements, as shown in Figure 3-14.

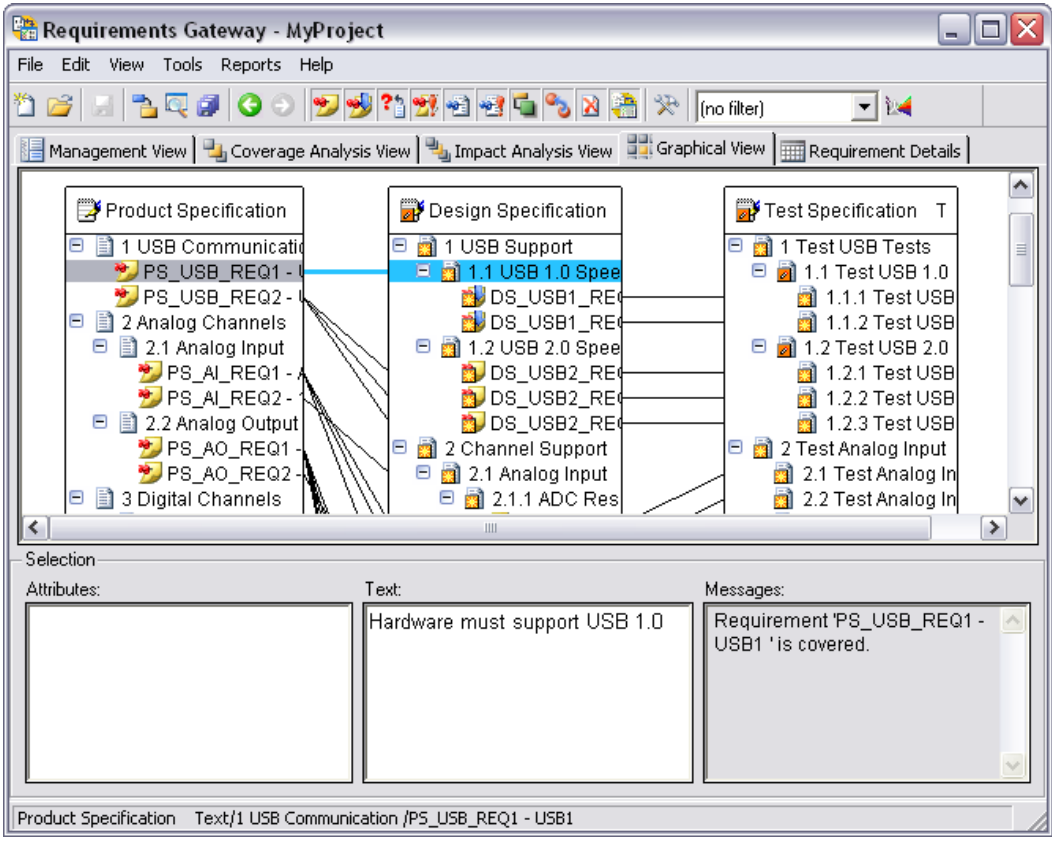


Figure 3-14. PS_USB_REQ1 Selected in Graphical View

4. Select the **PS_USB_REQ2** requirement. The Graphical View highlights the PS_USB_REQ2 requirement and additional elements from the two downstream documents.

5. Right-click in the **Graphical View** and select **View Graph for Selection** from the context menu. When you select this option, the Graphical View displays only the highlighted elements from the three documents, as shown in Figure 3-15.

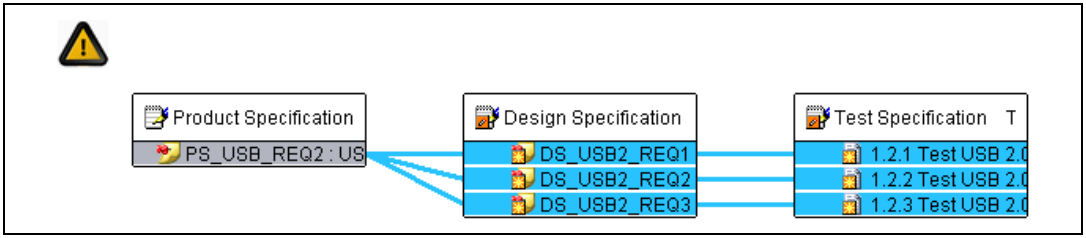


Figure 3-15. PS_USB_REQ2 Selected in Graphical View

The yellow warning icon in the Graphical View indicates that the graph is only a partial view of the entire project.

6. Right-click in the Graphical View and select **Show All Elements** from the context menu to display all the elements of the documents again.
7. Click the header of the Test Specification document to select the entire document.
8. Right-click the **Test Specification** document and select **Hide Selected Documents** from the context menu. The Graphical View hides the Test Specification document and displays the traceability information for the remaining two documents.
9. Right-click in the Graphical View and select **Show Hidden Documents** in the context menu to launch a dialog box in which you can select a previously hidden document to show again in the Graphical View.
10. Select **Test Specification** from the list of documents and click **OK** to close the dialog box and show the Test Specification document again.

As your project becomes more complex, you can perform the following tasks to control the Graphical View:

- Increase or decrease the number of elements shown in each document by expanding or collapsing sections within the document.
- Resize the Graphical View by selecting the lower right corner of the main window and dragging the window corner to increase or decrease the window size.
- Move documents within the Graphical View by selecting the document header and dragging the header to a new location.

- Resize the width of a document by selecting the document header and dragging the resize handles that appear on either side of the document.
- Zoom in and out by pressing <Ctrl> while rolling the mouse wheel up or down or by selecting **Zoom»100%** or **Zoom»Fit in page** from the context menu.

Refer to the *NI Requirements Gateway Help* for more information about using the Graphical View and about derived requirements and links.

4

Generating Reports

You can generate built-in or custom reports that contain project analysis information. Requirements Gateway distinguishes between the following types of reports:

- **Library Reports**—Default read-only report types available for all projects, stored in `<Requirements Gateway>\config\doc_models\Library Reports.xml`.
- **Public Reports**—Custom reports available for all projects, stored in `<Requirements Gateway Public>\Config\doc_models\Public Reports.xml`.
- **Project Reports**—Custom reports available only for a specific project, stored in the same directory as the project in `doc_models\Project Reports.xml`.

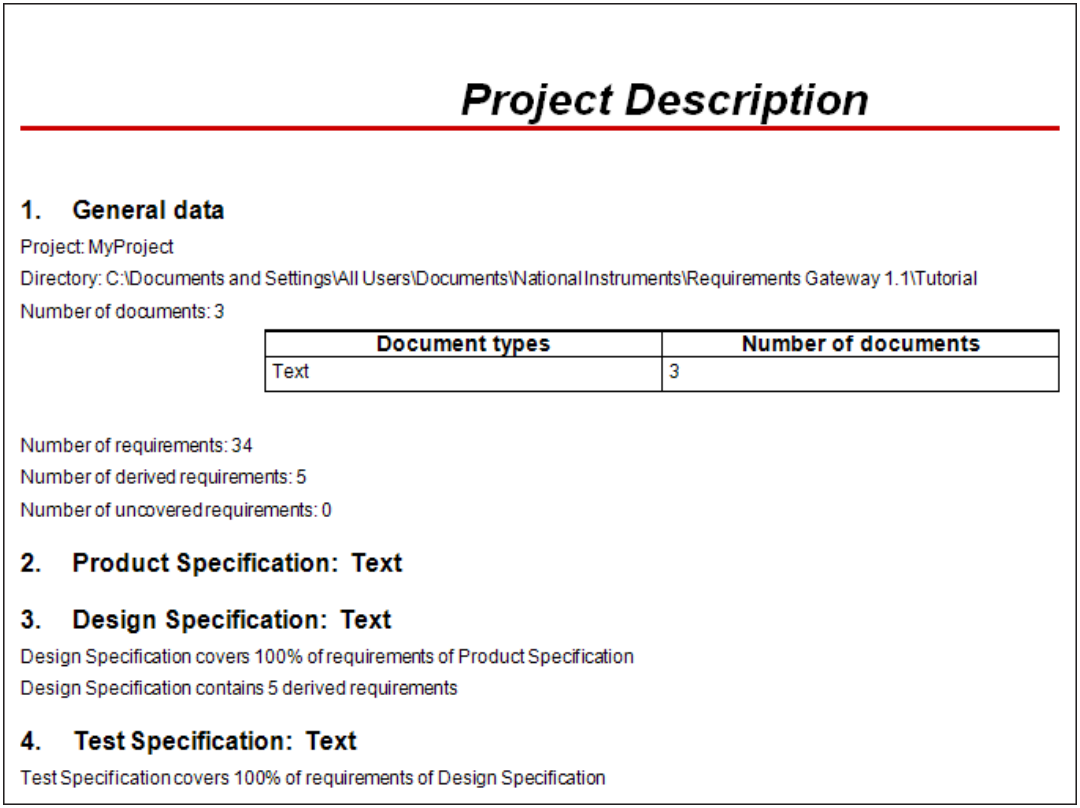
Generating a Library Report

Requirements Gateway installs the following library reports:

- **Traceability Matrix**—Lists the upstream-to-downstream covered requirements and the downstream-to-upstream covering references.
- **Analysis Results**—Summarizes the coverage analysis of the entire project.
- **Project Description**—Describes the project and the project documents.
- **Upstream Impact Analysis**—Lists the upstream traceability information for selected elements of the project.
- **Downstream Impact Analysis**—Lists the downstream traceability information for selected elements of the project.
- **Synthesis of Added Information**—Summarizes any added attributes, references, and text in the project.
- **Rules Checking**—Contains a summary of any rules the project violates.

Complete the following steps to generate a Project Description report for the project you created in Chapter 2, *Managing Requirements*, and modified in Chapter 3, *Analyzing Requirements*.

1. Open `<Requirements Gateway Public>\Tutorial\MyProject.rqtf` if it is not already open.
2. Select **Reports»Library Reports»Project Description** to launch the Save As dialog box.
3. Browse to the `<Requirements Gateway Public>\Tutorial` directory, enter `ProjectDescription` in the **File name** control, and click **Save**. Requirements Gateway generates and displays the report in Microsoft WordPad or Microsoft Word, as shown in Figure 4-1.



Project Description

1. General data

Project: MyProject

Directory: C:\Documents and Settings\All Users\Documents\National Instruments\Requirements Gateway 1.1\Tutorial

Number of documents: 3

Document types	Number of documents
Text	3

Number of requirements: 34

Number of derived requirements: 5

Number of uncovered requirements: 0

2. Product Specification: Text

3. Design Specification: Text

Design Specification covers 100% of requirements of Product Specification

Design Specification contains 5 derived requirements

4. Test Specification: Text

Test Specification covers 100% of requirements of Design Specification

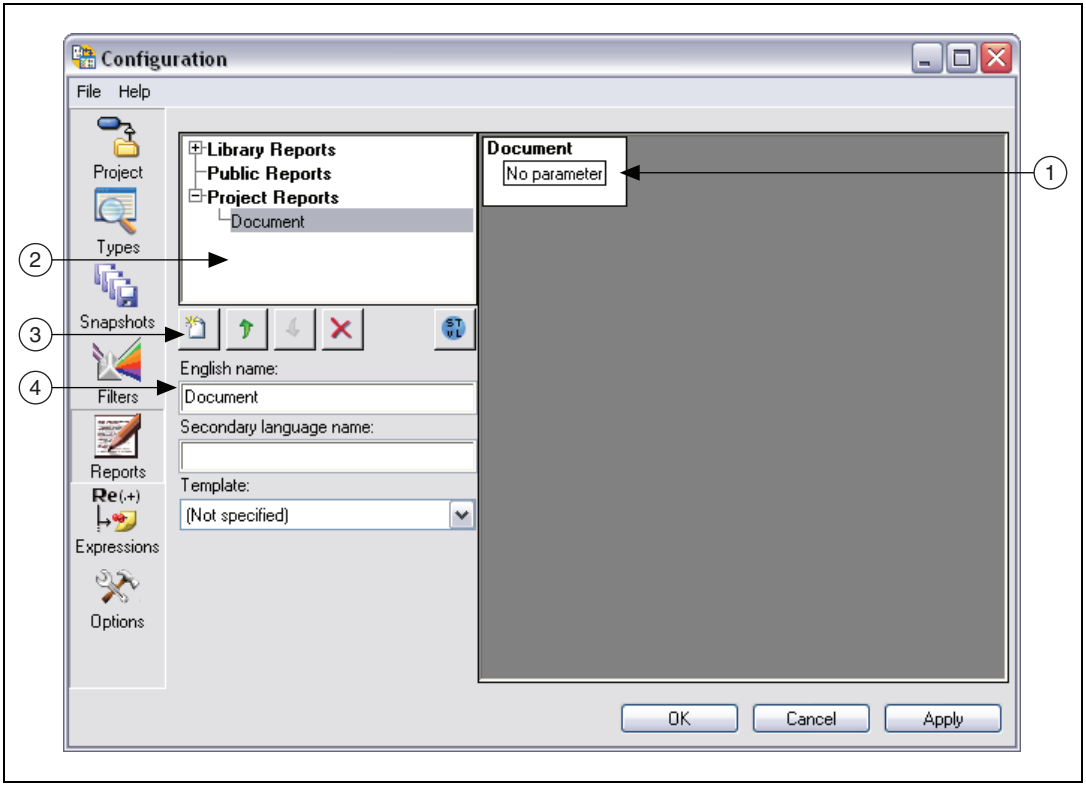
Figure 4-1. Project Description Report

4. Review the content of the report.
5. Exit the application that displays the report.

Creating a Custom Report

Complete the following steps to create a custom report that lists the requirements defined in each document of the project.

1. Open `<Requirements Gateway Public>\Tutorial\MyProject.rqtf` if it is not already open.
2. Select **Reports»Edit Reports** to launch the Reports pane of the Configuration dialog box.
3. Click the **New report** button, shown at left, to add a new report under Project Reports, as shown in Figure 4-2.



1 Report	2 Report List	3 Toolbar	4 Properties

Figure 4-2. Reports Pane of Configuration Dialog Box

The Reports pane contains the following sections:

- **Report**—A graphical structure of the active report selected in the Report List.
- **Report List**—Lists all the available library, project, and public reports.
- **Toolbar**—Contains buttons to create, rearrange, and delete reports from the Report List. In addition, the **Report elements** button, shown at left, toggles the controls below the Report List to display the properties of the selected item in the report or the elements available to insert in the report. You can also double-click the Report elements button to launch a modal Report Elements window.

- **Properties**—Contains the following panes that display separately when you toggle the Report elements button:
  - **Selection Properties**—Options for configuring the item you select in the Report List or in the graphical report structure. The options available depend on the element you select.
  - **Report Elements**—Structures, data blocks, and other parameters that you can add to the report. When you select an element of the report, valid elements you can insert into the selected element are highlighted with bold text. Refer to Chapter 4, *Customizing Reports*, of the *NI Requirements Gateway Customization Guide* for more information about the elements you can insert into other elements when customizing a report structure.

4. Enter `Requirements` in the **English name** control.
5. Select **portrait.rtf** from the Template ring control.
6. Select the **Requirements** root object of the report. Requirements Gateway highlights the Requirements object.
7. Click the **Report elements** button to display the Report Elements pane, which contains the following tabs:
   - **Structures**—Elements that define the structure of the report, such as text, paragraph, list, or table.
   - **Data**—Elements associated with the object you select in the report, such as the project, a document, or a requirement.
   - **Parameters**—Data type elements you can pass to a structure element.
8. Click the **Data** tab.

9. Select the **project»Contents»analyzed documents** element and drag the element to the selected Requirements object in the report, as shown in Figure 4-3.

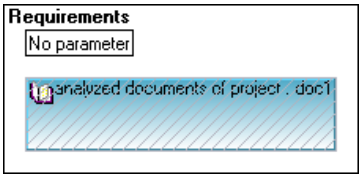


Figure 4-3. Inserted Analyzed Documents Element

The analyzed documents element instructs the report to loop on all the documents in the project.

10. Click the **Report elements** button to display the Selection Properties pane. Notice that the Variable name control value defaults to `doc1`. This variable represents the active document while looping on all documents in the project.
11. Click the **Report elements** button and click the **Structures** tab.
12. Select the **paragraph** element and drag the element to the analyzed documents element. Requirements Gateway adds a paragraph to the report for each document in the project. Steps 13 through 25 define the structure and content of the paragraphs.
13. Select the **text** element and drag the element to the paragraph element.
14. Select the **Report elements** button and enter `Document:` in the **English name** control.
15. Click to the right of the **Document:** text element in the report to highlight the entire paragraph element.
16. Click the **Report elements** button and click the **Data** tab.
17. Select the **doc1»Identification information»name** element and drag the element to the right of the Document: text element, as shown in Figure 4-4.

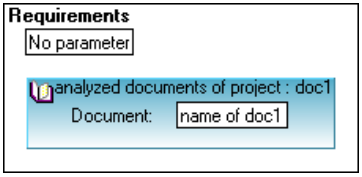


Figure 4-4. Inserted Name of Doc1 Element

18. Select the **analyzed documents of project : doc1** element in the report.
19. Click the **Structures** tab.
20. Select the **table** element and drag the element to the bottom of the analyzed document of project element.
21. Drag a **text** element to each of the elements in the upper half of the table and assign `Requirement` and `Text` to each of the table header elements.
22. Select the area below the column headers, as shown in Figure 4-5.

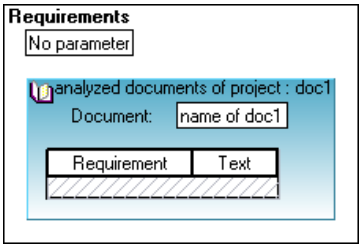


Figure 4-5. Inserted Column Text Elements

23. Click the **Data** tab and drag the **doc1»Contents»requirements** element to the lower half of the table.
24. Click the **Structures** tab and drag the **table row** element to the requirements of doc1 element.
25. Click the **Data** tab and drag the **req1»Identification information»display** element and the **req1»Identification information»text** element to each of the cells in the new table row, as shown in Figure 4-6.

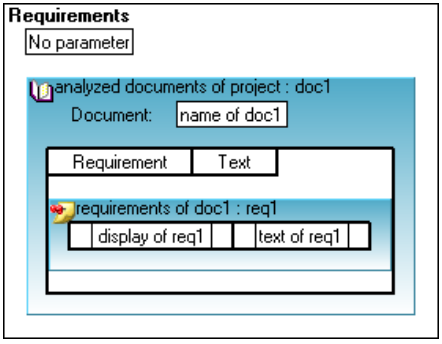


Figure 4-6. Final Custom Report

26. Click **OK** to close the Configuration dialog box.
27. Select **Reports»Project Reports»Requirements** from the main window to launch the Save As dialog box.
28. Enter `Requirements` in the **File name** control and click **Save** to generate and display the report file in Microsoft WordPad or Microsoft Word, as shown in Figure 4-7.

Requirements

Document:Product Specification

Requirement	Text
PS_USB_REQ1 - USB1	Hardware must support USB 1.0
PS_USB_REQ2 - USB2	Hardware must support USB 2.0
PS_AI_REQ1 - Analog Input	
PS_AI_REQ2 - 10 channels	
PS_AO_REQ1 - Analog Output	
PS_AO_REQ2 - 2 channels	
PS_DIO_REQ1 - Digital I/O	
PS_DIO_REQ2 - 2 channels	
PS_CNTR_REQ1 - Counter	
PS_CNTR_REQ2 - 1 counter	

Document:Design Specification

Requirement	Text
DS_USB1_REQ1 - Low Speed	1.5 Mbps
DS_USB1_REQ2 - High Speed	12 Mbps

Figure 4-7. Final Generated Report

29. Review the content of the report.
30. Exit the application that displays the report.

Refer to Chapter 4, *Customizing Reports*, of the *NI Requirements Gateway Customization Guide* and the *NI Requirements Gateway Help* for more information about customizing reports.

5

Customizing Types

The organization and formatting of data can vary between documents and does not always adhere to the format the default implementation of a type requires. You can modify the document to conform to the required format, or you can create a custom type that processes the format the source document defines.

This tutorial explains how to create a custom type based on the default Text type to capture traceability information from text files that use unique syntax for defining traceability information.

Reviewing File Formats

For most types, Requirements Gateway captures potential traceability information by translating an external file, such as a Microsoft Word document, into an intermediate text or XML file. The type analyzes the intermediate file and captures the required structure and traceability information. Refer to Chapter 1, *Capturing Information from External Products*, of the *NI Requirements Gateway Customization Guide* and to the individual coupling documents for more information about the content and format of the intermediate text or XML file.

The Text type directly processes the content of a text file without using a translated intermediate file.

Complete the following steps to evaluate the format of custom text files so you can create a custom type to capture the traceability information from the files.

1. Select **File»Open** to launch the Open dialog box and navigate to the `<Requirements Gateway Public>\Tutorial` directory.
2. Select **CustomTypeProject.rqtf** and click **Open**. Requirements Gateway opens the project file and displays the documents in the main window.
3. Click the **Management View** tab. Requirements Gateway does not list any sections or requirements for the documents in the project because the documents use a syntax for requirements, references, and text elements that the default Text type cannot recognize.

4. Double-click the **Product Specification** document on the tree view pane to launch `CustomProductSpec.txt` in an external application.
5. Review the content of the text file in the external application. Figure 5-1 shows a portion of `CustomProductSpec.txt`.

```
- USB Communication            # The hardware will
    [REQ: USB1] USB1            # Low & High Speeds
    [REQ: USB2] USB2            # Low, Med, and High

- Analog Channels
  - Analog Input               # The hardware will
    [REQ: AI] Analog Input      # +/- 12 Volts
    [REQ: AI_COUNT] 10 channels
```

Figure 5-1. Text from CustomProductSpec.txt

6. Double-click the **Covering Specification** document on the tree view pane to launch `CustomCoveringSpec.txt` in an external application.
7. Review the content of the text file in the external application. Figure 5-2 shows a portion of `CustomCoveringSpec.txt`.

```
- USB Support
  - USB 1.0 Speeds             #Hardware supports USB
    [REF: USB1]

  - USB 2.0 Speeds             #Hardware supports USB
    [REF: USB2]
```

Figure 5-2. Text from CustomCoveringSpec.txt

The Product Specification and Coverage Specification documents contain the following traceability formalisms. Documents must adhere to the formalisms the types define.

- Each document specifies two levels of sections. A minus character precedes a first-level section, and two space characters and a minus character precede a second-level section.
- The syntax `[REQ: id]`, where `id` is a set of characters that represent the requirement, specifies requirements. An optional text label follows the requirement.

- The syntax `[REF: id]`, where `id` is a set of characters that represents the requirement, specifies references.
- The `#` character specifies text associated with a section or requirement.

8. Exit the external applications.

Creating a Custom Type

When you customize a type, you can duplicate and modify an existing type or create a new type. Because the Product Specification and Coverage Specification documents adhere to a very different formalism than what the default Text type defines, you must create a new type.

Complete the following steps to create a new custom Text type:

1. Select **File»Edit Types** or click the **Edit Types** button, shown at left, on the toolbar in the main window to launch the Types pane of the Configuration dialog box.

   The Types pane contains the following sections and panes:

   - **Types List**—The list of types available in Requirements Gateway. The Types List groups the types in folders that correspond to the read-only directories in the `<Requirements Gateway>\config\types` directory.

     The Types List also contains a public folder. You can place custom types and duplicates of the default types in the public folder to make the types available for all projects. Requirements Gateway stores the type definitions for the public folder in `<Requirements Gateway Public>\Config\types\public\public.types`.

     The final folder in the list represents the custom types defined for any projects in the directory of the current project file.
   - **Type Toolbar**—Buttons to copy, delete, and add new elements to the types list or to the elements of a specific type.
   - **Selection Properties**—The settings associated with the selected type or type element in the Types List. Depending on the element you select in the Types List, the Selection Properties pane can display Analysis, Advanced Options, and Attributes tabs. Refer to the *NI Requirements Gateway Help* for more information about the options on the tabs and toolbar of the Types pane.

2. Select the **Tutorial** folder in the tree view.

3. Click the **Add new type** button, shown at left, on the Type Toolbar to add a new element, Type, to the `Tutorial` folder in the Types List.

Note Use the **Add XML type** and **Add a type for added elements** buttons on the Type Toolbar to create custom types for specific purposes. Refer to Chapter 2, *Customizing Types and Type Elements*, of the *NI Requirements Gateway Customization Guide* for more information about XML types. Refer to the *NI Requirements Gateway Help* for more information about types for added elements.

4. Enter `My Type` in the **Name** control.
5. Click the **Analysis** tab on the Selection Properties pane, if it is not already selected.
6. Select **Text** from the **Convert tool** ring control for Requirements Gateway to interpret any document that uses the type as a general text file, similar to the default Text type.
7. Select **Text** from the **Edit tool** ring control for Requirements Gateway to launch the application associated with the file type when you navigate to the document.

Figure 5-3 shows the new type with the Name, Convert tool, and Edit tool options completed.

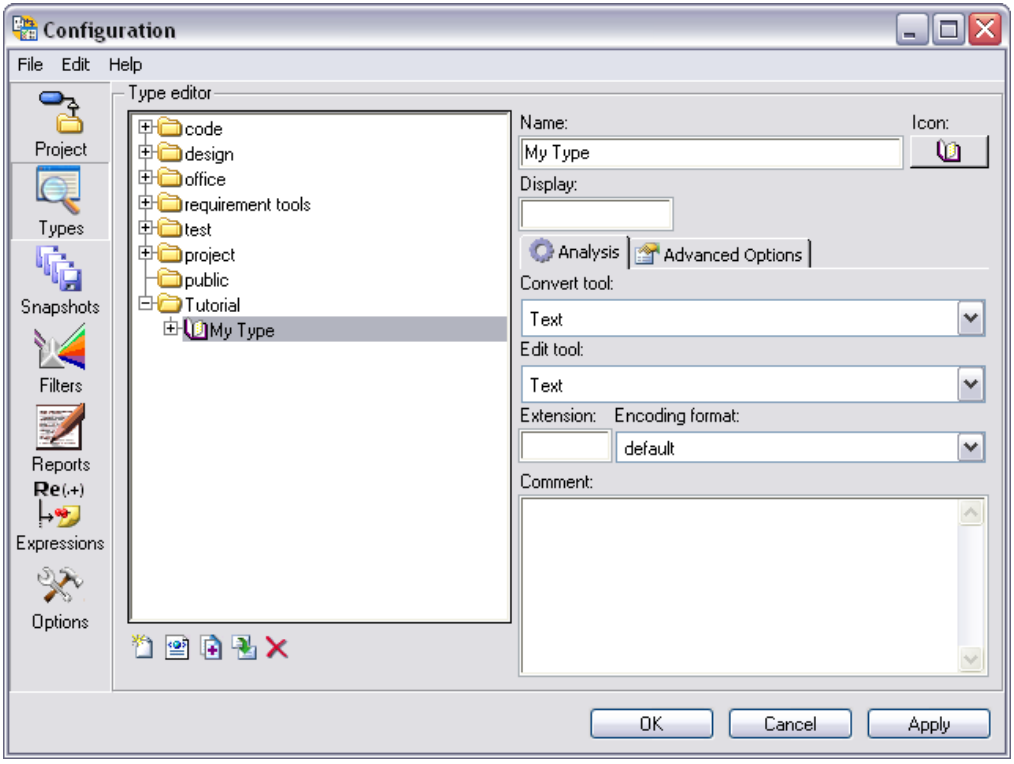


Figure 5-3. New Type Created

8. Expand the child elements under the **My Type** element in the Types List.

   The elements under a type in the Types List define how Requirements Gateway identifies traceability information in an intermediate text or XML file that represents the data extracted from a project document. Type elements use regular expressions to specify patterns in the text of the intermediate file for Requirements Gateway to identify instances of an element. Types that use intermediate XML files also specify the structure of the XML data to identify traceability information.

Table 5-1 lists the elements that a type might define and the purpose of each element. Chapters 6 through 10 and Appendix A of this manual contain similar tables with information specific to the type the chapter discusses.

Table 5-1. Type Elements

Element	Description
Section	Specifies how to identify and display the structural elements for a document. Examples of sections include headings in a text or Microsoft Word file, files in a directory, rows in a database, sequences, step groups, and steps in a TestStand sequence file, or front panels, diagrams, controls, and indicators in a LabVIEW VI.
Macro-Requirement	Specifies how to identify and display macro-requirements for a document. A macro-requirement element defines a starting location and an ending location, where Requirements Gateway associates any requirement element between the locations with the macro-requirement. For example, a macro-requirement element might interpret the syntax `[MacroReq_reqid]`, where `reqid` is the macro-requirement identifier, as a starting location, and the syntax `[End_of_MacroReq]` as an ending location.
Requirement	Specifies how to identify and display requirements for a document. For example, a requirement element might interpret `[REQ: reqid]` as a requirement, where `reqid` is the requirement identifier.
Entity	Specifies how to identify and display an entity for a document. An entity element is similar to a section element except that Requirements Gateway triggers a rule violation if a reference element does not follow an entity element. Refer to the *NI Requirements Gateway Help* for more information about rule violations.
Reference	Specifies how to identify references to requirements for a document. Requirements Gateway does not display reference elements but associates the reference with a preceding section or entity element. For example, an element might interpret `[Covers: reqid]` as a requirement, where `reqid` is the identifier of the covered requirement.
Attribute	Specifies how to identify and display attributes of requirement elements for a document. An attribute element has a name and a Boolean, string, or numeric value. An attribute element might define the priority or owner of a requirement.

Table 5-1. Type Elements (Continued)

Element	Description
Reference Attribute	Specifies how to identify and display attributes of reference elements for a document. A reference attribute element has a name and a Boolean, string, or numeric value. A reference attribute element might define the type of coverage for a reference.
Link	Specifies how to identify and display a non-covering reference to a requirement or a reference to a section or entity. The link element must refer to the identifier of the requirement, section, or entity. You can navigate from the link element to the requirement, section, or entity element.
Text	Specifies how to identify text associated with a preceding section, entity, requirement, or attribute element.
Picture	Specifies how to identify pictures associated with a preceding section or entity element. Requirements Gateway supports images with BMP, PNG, Scalable Vector Graphics (SVG), and Windows Metafile (WMF) file formats.

Refer to Chapter 2, *Customizing Types and Type Elements*, of the *NI Requirements Gateway Customization Guide* for more information about configuring elements when creating a custom type, including the options available on the Types pane for each element.

9. Select the **section** element in the Types List.
10. Click the **Add new type** button to add a new section element named `Section` under the custom type.

11. Enter `^-[ \t]*([^#\n\r]*).*$` in the **Regular expression** control on the Analysis tab, as shown in Figure 5-4.

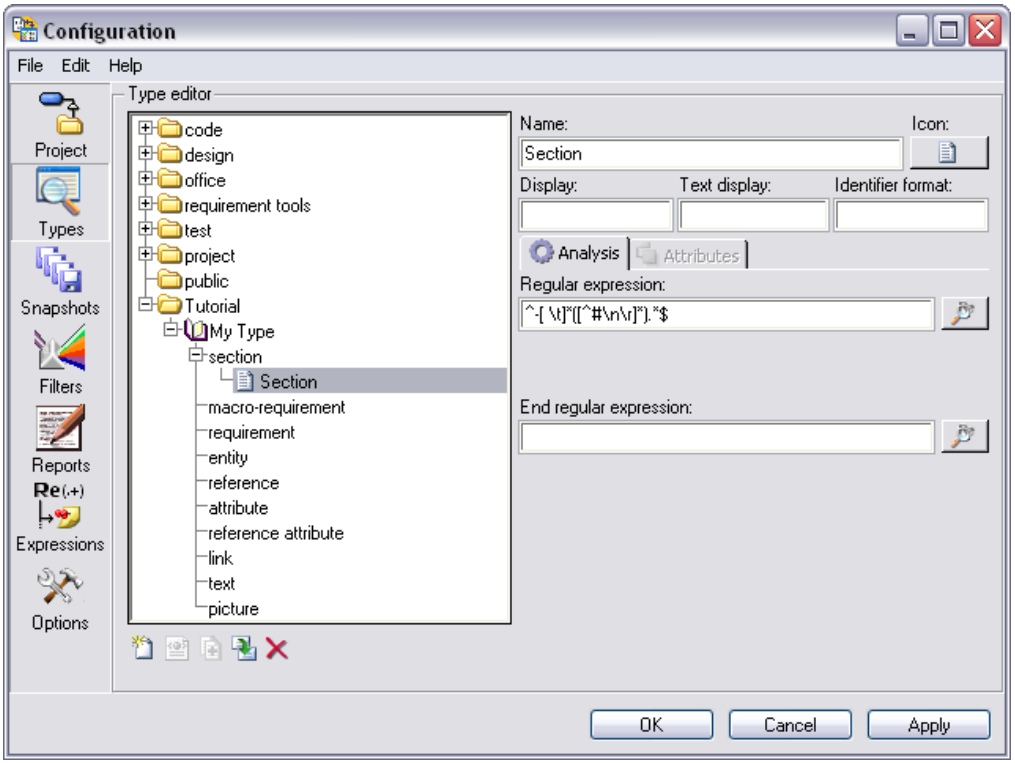


Figure 5-4. Section Element of the Custom Type

The regular expression of a section element specifies the pattern of text to match in the source document to locate section elements and text of section elements. Table 5-2 explains each component of the expression you entered in step 11.

Table 5-2. Section Regular Expression Components

Expression Component	Purpose
`^`	Specifies to start matching text patterns at the beginning of a line.
`-`	Specifies that the next character must be the minus character.
`[ \t]*`	The surrounding bracket characters specify that a character must match a space or a tab character, and the asterisk specifies that the text must contain zero or more matches.

Table 5-2. Section Regular Expression Components (Continued)

Expression Component	Purpose
`( )`	Specifies that any text that matches the inside pattern is returned as a field. This returns the text associated with the section.
`[^#\n\r]*`	The `^` character specifies that the pattern cannot match any of the following specified characters: `#`, new line, or a carriage return. This forces the pattern to stop at the `#` character, if the `#` character exists.
`.*`	Specifies to match zero or more of any of the characters. This pattern process the `#` character, if it exists, and any subsequent text.
`$`	Specifies to match the end of a line.

12. With the new Section element still selected in the tree view, click the **Add new type** button to add a new section element named `Section1` under the Section element.
13. Enter `^[ \t][ \t]-[ \t]*([^#\n\r]*).*$` in the **Regular expression** control on the Analysis tab. This regular expression is similar to the expression in step 11, with two additional required spaces or tabs that precede the minus character.
14. Select the **requirement** element in the tree view.
15. Click the **Add new type** button to add a new requirement element named `Requirement`.
16. Enter `\[REQ: ([^\]]+)\][ \t]*([^#\r\n]*)` in the **Regular expression** control. Table 5-3 explains each component of this expression.

Table 5-3. Requirement Regular Expression Components

Expression Component	Purpose
`\[REQ:`	Specifies to match the characters `[REQ:` followed by a space character.
`([^\]]+)`	Specifies to return one or more characters that do not include the closing bracket character. This expression component returns the requirement identifier.
`\]`	Specifies to match the closing bracket character.
`[ \t]*`	Specifies to match zero or more space or tab characters.
`([^#\r\n]*)`	Specifies to return zero or more characters that do not include the `#` character, new line, or carriage return. This expression component returns the requirement label.

17. Select the **reference** element in the tree view.
18. Click the **Add new type** button to add a new reference element named `Reference`.
19. Enter `\[REF: ([^\]]+)` in the **Regular expression** control on the Analysis tab. This regular expression specifies to return one or more characters between the space character after `[REF:` and the closing bracket character. This expression returns the requirement identifier for the reference.
20. Select the **text** element in the tree view.
21. Click the **Add new type** button to add a new text element named `Text1`.
22. Enter `#[\t ]*(.+)$` in the **Regular expression** control. This regular expression specifies to return as the text value all characters between the `#` character and the end of line, excluding any preceding space or tab characters after the `#` character.
23. Click **Apply** to save the new type.
24. Click **Yes** when Requirements Gateway prompts you to reanalyze the project.

Applying a Custom Type

Complete the following steps to configure the documents in the project to use the new type:

1. Open `<Requirements Gateway Public>\Tutorial\CustomTypeProject.rqtf` if it is not already open.
2. Click the **Project** button in the Configuration dialog box to launch the Project pane.
3. For each document in the project, select **My Type** from the **Type of Analysis** ring control on the Document Details pane.
4. Click **OK** to close the Configuration dialog box.
5. Click **Yes** when Requirements Gateway prompts you to reanalyze the project.
6. Click the **Management View** tab and review the content of each document. The Project Summary section indicates that the two documents have ten requirements and zero uncovered requirements.

Chapters 6 through 10 of this manual demonstrate how to use Requirements Gateway with TestStand, MATRIXx, LabVIEW, LabWindows/CVI, and Telelogic DOORS, respectively. Refer to Appendix A, *Third-Party Types Overview*, for a general overview of other types that Requirements Gateway supports. Refer to the *NI Requirements Gateway Help* for more information about the options on the Types pane.

6

Using NI Requirements Gateway with NI TestStand

Use the TestStand type to add TestStand documents to a project. When you configure a TestStand document, you can select sequence, project, or workspace files. Typically, you use a TestStand document to cover requirements in a specification document. For example, the steps in a sequence can cover the requirements for testing a product.

Use the TestStand XML Report type to show the execution coverage of the steps in TestStand sequences. When you configure a TestStand XML Report document, you select an XML report file.

Note You must have TestStand 3.5 or later installed on your computer to complete the tutorials in this chapter.

TestStand Type

The TestStand type analyzes traceability information specified in TestStand files. The default TestStand type searches for traceability information stored in the comment or requirements properties of steps, sequences, sequence files, project files, and workspace files. You can customize the definition of the TestStand type to conform to the data format that a file contains.

The TestStand type defines the following document settings that you configure for a TestStand document on the Document Details pane of the Project pane of the Configuration dialog box:

- **File or Directory**—Select a sequence or workspace file, or a directory that contains sequence files. For a directory, you can specify to include subdirectories.
- **Variables**—None.

The TestStand type defines the following default elements:

- **Section**—The sequences and step groups of sequence files, the hierarchy of workspaces, and the files in directories. The Section element of the TestStand type also displays steps that cannot generate results as a section element.
- **Macro-Requirement**—Not defined.
- **Requirement**—The steps that can generate results as requirements that a document of the TestStand XML Reports type can cover. The TestStand XML Reports type uses the unique ID of the step as the requirement identifier.
- **Entity**—Not defined.
- **Reference**—Specified by entering a requirement identifier in the Requirements List property of a step, sequence, sequence file, project file, or workspace file.

  You can also enter `[Covers: reqid]`, where `reqid` is the requirement identifier, in a comment of a step, sequence, or sequence file to specify references.
- **Attribute**—The comments, paths, and versions of files.
- **Reference Attribute**—Not defined.
- **Link**—Not defined.
- **Text**—Not defined.
- **Picture**—Not defined.

When you create a custom TestStand type, you do not need to customize any type definition settings on the Analysis tab of the Types pane of the Configuration dialog box. Refer to Chapter 5, *Customizing Types*, for more information about creating custom types.

Refer to the *Coupling NI TestStand with NI Requirements Gateway* document for more information about using the TestStand type in Requirements Gateway.

TestStand XML Reports Type

The TestStand XML Reports type determines the execution coverage of the steps that a TestStand document specifies. The default TestStand type defines steps that can generate results as requirements using the unique ID of a step as the requirement identifier. The default TestStand XML Reports type uses the same identifier that is stored with a step result as a requirement reference.

Note The TestStand XML Reports type can analyze only XML results, not Automatic Test Markup Language (ATML), HTML, or ASCII reports. In addition, TestStand does not allow you to generate two report files at the same time. Refer to the National Instruments website at ni.com/info and enter the info code rd2rit to access the TestStand support document, *Generating Two Reports in TestStand*, which contains information about how to alter the process model to generate two report files. Refer to TestStand documentation for more information about ATML reports.

The TestStand XML Reports type defines the following document settings that you configure for a TestStand XML Report document on the Document Details pane of the Project pane of the Configuration dialog box:

- **File or Directory**—Select an XML report file or a directory that contains XML report files. For a directory, you can specify whether to include subdirectories.
- **Variables**—Not defined.

The TestStand XML Reports type defines the following default elements:

- **Section**—The results of an XML report file, the files within directories, and the critical failure stack for failing Units Under Test (UUTs).
- **Macro-Requirement**—Not defined.
- **Requirement**—Not defined.
- **Entity**—Not defined.
- **Reference**—A step result in an XML report file. The requirement identifier is the unique step ID stored with the result.
- **Attribute**—Indicates UUT and step result information.
- **Reference Attribute**—Indicates the status of the step result.
- **Link**—A critical stack failure element that refers to the associated step result.
- **Text**—Not defined.
- **Picture**—Not defined.

When you create a custom TestStand XML Reports type, you do not need to customize any type definition settings on the Analysis tab of the Types pane of the Configuration dialog box. Refer to Chapter 5, *Customizing Types*, for more information about creating custom types.

Refer to the *Coupling NI TestStand with NI Requirements Gateway* document for more information about using the TestStand XML Reports type in Requirements Gateway.

Adding TestStand Documents to Projects

Complete the following steps to add a TestStand document to a project:

1. Open `<Requirements Gateway Public>\Tutorial\TestStandProject.rqtf`.
2. Click the **Management View** tab if it is not already selected.
3. Double-click the **Specification** document in the tree view to launch `TestStandSpec.txt` in an external application.
4. Review the content of the text file in the external application. The Specification document contains four requirements. The requirement in the Product section specifies the overall test sequence. The three requirements in the Parts section specify individual tests for the sequence.
5. Exit the application that displays `TestStandSpec.txt`.
6. In Requirements Gateway, launch the Project pane of the Configuration dialog box to edit the project.
7. Click the **Add a document** button to add a new document in the Traceability Description Zone. Place the document below the Specification document.
8. On the Document Details pane, enter `TestStand Files` in the **Name** control and select **TestStand** from the **Type of Analysis** ring control.
9. Click in the **File or Directory** column. The File Browse button becomes visible on the right side of the control. Click the **File Browse** button to launch the Select Files to Include in Document dialog box.

   Use the Select Files to Include in Document dialog box to include multiple sequence files in a TestStand document in a Requirements Gateway project. You can add individual sequence files, all the sequences in a TestStand workspace file, or all the sequence files in a directory. In addition, when you select a directory added in the list view of the Select Files to Include in Document dialog box, you can specify to include all the subdirectories.

   Refer to the *Coupling NI TestStand with NI Requirements Gateway* document for more information about the Select Files to Include in Document dialog box for a TestStand document.
10. Click **Add Sequence File** and select `<Requirements Gateway Public>\Tutorial\TestStandWidgetTests.seq`.
11. Click **Open** to add the file to the list view of the Select Files to Include in Document dialog box.
12. Click **OK** to close the Select Files to Include in Document dialog box.

13. Click the **Add a cover** button to begin adding a covering link. Click the **TestStand Files** document and then click the **Specification** document to create an arrow between the two documents.
14. Click **OK** to close the Configuration dialog box.
15. Click **Yes** when Requirements Gateway prompts you to reanalyze the project.
16. Expand the two documents in the Management View, as shown in Figure 6-1. Requirements Gateway displays `TestStandWidgetTests.seq` under the TestStand Files document.

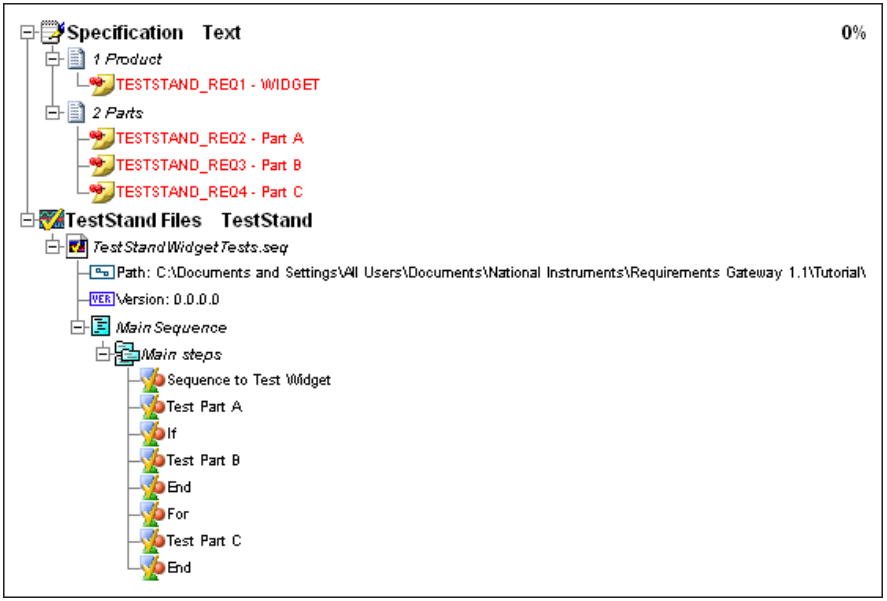


Figure 6-1. TestStand Document Expanded in the Management View

Adding References to TestStand Files

Complete the following steps to specify references in a TestStand sequence that cover the requirements in the Specification document.

1. Select the **TESTSTAND_REQ1** requirement in the Specification document.
2. Right-click the **TESTSTAND_REQ1** requirement and select **Copy For»TestStand Requirement List** to copy the required covering syntax for TestStand to cover the selected requirement.

3. Double-click the **MainSequence»Main steps** section of `TestStandWidgetTests.seq` to launch TestStand and open the sequence file.
4. In TestStand, select **Edit»Sequence Properties** to launch the Sequence Properties dialog box, the title of which is specific to the sequence you select.
5. Select **<insert new item>** in the Requirements List and press <Ctrl-V> to paste the text from the clipboard. TestStand adds `TESTSTAND_REQ1` to the Requirements List, as shown in Figure 6-2.

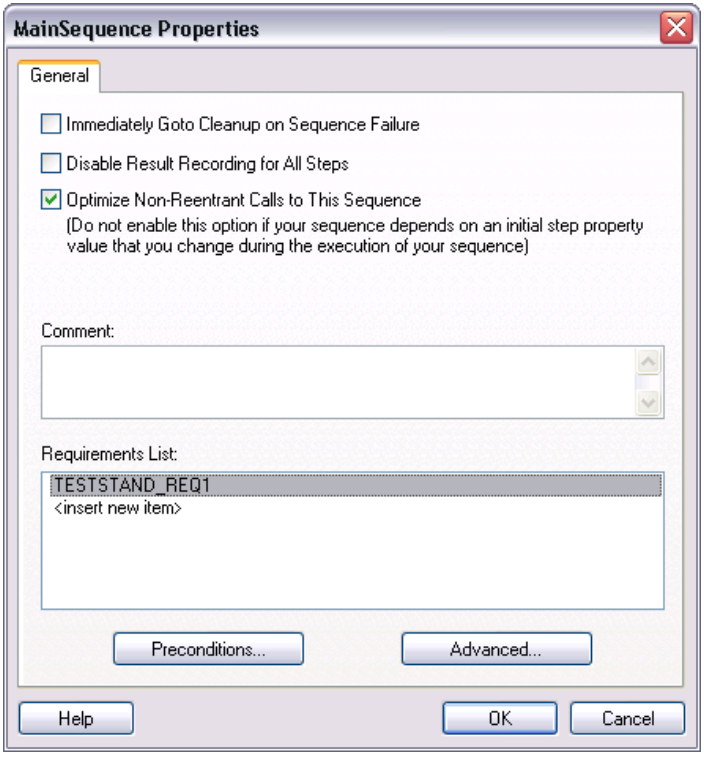


Figure 6-2. TestStand Sequence Properties Dialog Box

6. Click **OK** to close the Sequence Properties dialog box.

7. **(TestStand 4.0 and later)** Complete the following steps to add requirement text to a step.
   a. Select the **Test Part A** step in the Steps list on the Steps pane to display the Properties tab of the Step Settings pane.
   b. Click **Requirements** on the Properties tab of the Step Settings pane to display the Requirements panel.
   c. Select **<insert new item>** on the Requirements panel and enter `TESTSTAND_REQ2`.

   (TestStand 3.5) Complete the following steps to add requirement text to a step.
   a. Double-click the **Test Part A** step to launch the Step Properties dialog box, the title of which is specific to the step that you select.
   b. Click the **Requirements** tab of the Step Properties dialog box.
   c. Select **<insert new item>** in the Requirements List and enter `TESTSTAND_REQ2`, and click **OK** to close the Step Properties dialog box.
8. Repeat step 7 to add `TESTSTAND_REQ3` to the Requirements List for the Test Part B step, and `TESTSTAND_REQ4` to the Requirements List for the Test Part C step.
9. Select **File»Save** to save the changes to the sequence file. Exit TestStand and return to Requirements Gateway. Click **Yes** when Requirements Gateway prompts you to reload the TestStand Files document.
10. Expand the two documents in the Management View. Requirements Gateway indicates that the coverage is now 100%.
11. Click the **Coverage Analysis View** tab and the **Graphical View** tab to review the coverage of requirements in the Specification document by the elements of the TestStand Files document.

Adding TestStand XML Report Documents to Projects

Complete the following steps to add a TestStand XML Report document to the project and review the execution coverage.

1. Open `<Requirements Gateway Public>\Tutorial\TestStandProject.rqtf` if it is not already open.
2. Launch the Project pane of the Configuration dialog box to edit the project.

3. Click the **Add a document** button to add a new document in the Traceability Description Zone. Place the document below the TestStand Files document.
4. On the Document Details pane, enter `XML Report` in the **Name** control and select **TestStand XML Reports** from the **Type of Analysis** ring control.
5. Click in the **File or Directory** column. The File Browse button becomes visible on the right side of the control. Click the **File Browse** button to launch the Select Files to Include in Document dialog box.

   Use the Select Files to Include in Document dialog box to include multiple reports in the TestStand XML Report document in a Requirements Gateway project. You can add individual report files or all report files in a directory. In addition, when you select a directory added in the list view of the Select Files to Include in Document dialog box, you can specify to include all the subdirectories.
6. Click **Add XML Report File** and select `<Requirements Gateway Public>\Tutorial\TestStandWidgetTestsReport.xml`.
7. Click **Open** to add the file to the list view of the Select Files to Include in Document dialog box.
8. Click **OK** to close the Select Files to Include in Document dialog box.
9. Click the **Add a cover** button to begin adding a covering link. Click the **XML Report** document and then click the **TestStand Files** document to create an arrow between the two documents.
10. Click **OK** to close the Configuration dialog box.
11. Click **Yes** when Requirements Gateway prompts you to reanalyze the project.

12. Expand the XML Report document in the Management View. Requirements Gateway displays `TestStandWidgetTestsReport.xml` under the XML Report document, as shown in Figure 6-3.

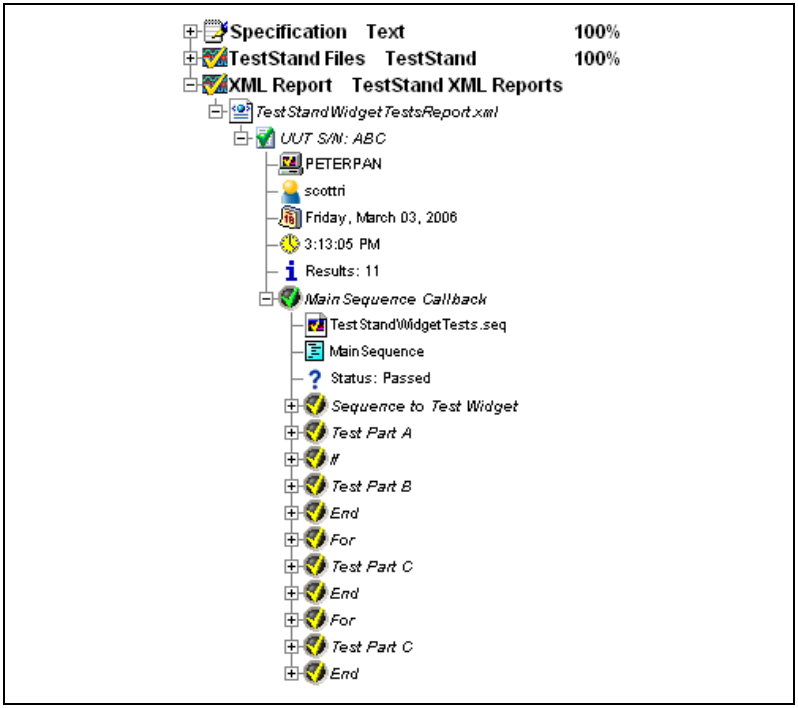


Figure 6-3. TestStand XML Report Document Expanded in the Management View

13. Click the **Coverage Analysis View** tab. Expand the **TestStand Files** document to display the steps in the Main steps section of `TestStandWidgetTests.seq`.
14. Select the **Main steps** section. The Downstream Coverage Information column displays the status for all the steps in the Main steps section in the XML Report document.
15. Select the **Test Part A** step in the Main steps section. The Downstream Coverage Information column displays a status of `Done` for the Test Part A step in the XML Report document.
16. Click the **Impact Analysis View** and the **Graphical View** tabs to review the coverage of requirements in the Specification document by the TestStand Files and XML Report documents.

7

Using NI Requirements Gateway with MATRIXx

Use the MATRIXx type to add MATRIXx documents to a project. When you configure a MATRIXx document, you select a SystemBuild catalog file or a specific model in a catalog file. You typically use a MATRIXx document to cover requirements in a specification document. For example, the blocks in a superblock can cover the requirements for a simulation model. Requirements Gateway 1.1 supports MATRIXx 7.1.6, 7.1.7, and 7.1.8.

MATRIXx Type

The MATRIXx type analyzes traceability information specified in the models that SystemBuild catalog files define. The default MATRIXx type searches for traceability information stored in the comments or the **Requirements_s** user parameter of a model object. You can customize the definition of the MATRIXx type to conform to the data format that a file contains.

The MATRIXx type defines the following document settings you configure for a MATRIXx document on the Document Details pane of the Project pane of the Configuration dialog box:

- **File or Directory**—Selects a catalog file, individual superblocks from a catalog file, or a directory that contains catalog files. For a directory, you can specify to include subdirectories.
- **Variables**—Not defined.

The MATRIXx type defines the following default elements:

- **Section**—Captures the superblocks or models in a catalog file to display the hierarchy of the superblocks or models, similarly to SystemBuild. The display contains a list of models, superblocks, state diagrams, and datastores.
- **Macro-Requirement**—Not defined.

- **Requirement**—Specified by `[REQ: reqid]`, where `reqid` is the requirement identifier, in a comment field of an object or the **Requirements_s** user parameter.
- **Entity**—Not defined.
- **Reference**—Specified by `[Covers: reqid]`, where `reqid` is the requirement identifier, in a comment field of an object or the **Requirements_s** user parameter.

  You can specify references in the comments of superblocks, blocks, datastores, state diagrams, bubbles, and transitions. You can also specify references in the objects that support user parameters, such as superblocks, blocks, and state diagrams.
- **Attribute**—Defines the following attribute elements:
  - **Block Type**—The type of block.
  - **Attribute**—Specified by `(#name)`, where `name` is the name of the attribute, in a comment field of an object or the **Requirements_s** user parameter. The attribute must appear after the corresponding requirement.
- **Reference attribute**—Defines the following reference attribute elements:
  - **Attribute**—Specified by `(*name)`, where `name` is the name of the attribute, in a comment field of an object or the **Requirements_s** user parameter. The attribute must appear after the corresponding reference.
- **Link**—A link to the corresponding superblock, state diagram, or datastore element in the object lists.
- **Text**—Not defined.
- **Picture**—Not defined.

When you create a custom MATRIXx type, you do not need to customize any type definition settings on the Analysis tab of the Types pane of the Configuration dialog box. Refer to Chapter 5, *Customizing Types*, for more information about creating custom types.

Refer to the *Coupling NI MATRIXx with NI Requirements Gateway* document for more information about using the MATRIXx type in Requirements Gateway.

Adding MATRIXx Documents to Projects

Complete the following steps to add a MATRIXx document to a project.

1. Open `<Requirements Gateway Public>\Tutorial\MATRIXxModel.rqtf`.
2. Click the **Management View** tab if it is not already selected.
3. Double-click the **Specification** document in the tree view to launch `MATRIXxSpec.txt` in an external application.
4. Review the content of the text file in the external application. The Specification document contains four requirements. The requirement in the Model section specifies the purpose of the superblock. The three requirements in the Blocks section specify the key blocks the model must include.
5. Exit the application that displays `MATRIXxSpec.txt`.
6. In Requirements Gateway, launch the Project pane of the Configuration dialog box to edit the project.
7. Click the **Add a document** button to add a new document in the Traceability Description Zone. Place the document below the Specification document.
8. On the Document Details pane, enter `MATRIXx Model` in the **Name** control and select **MATRIXx** in the **Type of Analysis** ring control.
9. Click in the **File or Directory** column. The File Browse button becomes visible on the right side of the control. Click the **File Browse** button to launch the Select Files to Include in Document dialog box.

   Use the Select Files to Include in Document dialog box to include multiple models in a MATRIXx document in a Requirements Gateway project. You can add individual superblocks, catalog files, or all the catalog files in a directory. In addition, when you select a directory added in the list view of the Select Files to Include in Document dialog box, you can specify to include all the subdirectories.

   Refer to the *Coupling NI MATRIXx with NI Requirements Gateway* document for more information about the Select Files to Include in Document dialog box for a MATRIXx document.
10. Click **Add MATRIXx File** and select `<Requirements Gateway Public>\Tutorial\MATRIXxModel.cat`.
11. Click **Open** to add the file to the list view of the Select Files to Include in Document dialog box.
12. Click **OK** to close the Select Files to Include in Document dialog box.

13. Click the **Add a cover** button to begin adding a covering link. Click the **MATRIXx Model** document and then click the **Specification** document to create an arrow between the two documents.
14. Click **OK** to close the Configuration dialog box.
15. Click **Yes** when Requirements Gateway prompts you to reanalyze the project.
16. Expand the two documents in the Management View, as shown in Figure 7-1. Requirements Gateway displays the superblocks under the MATRIXx Model document.

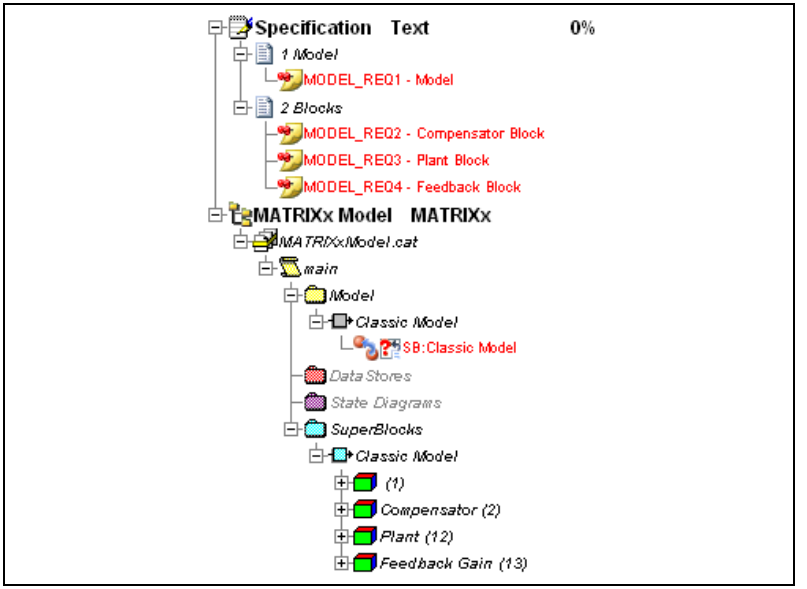


Figure 7-1. MATRIXx Document Expanded in the Management View

Adding References to MATRIXx Blocks

Complete the following steps to specify references in a MATRIXx model that cover the requirements in the Specification document.

1. In the Management View, right-click the **MODEL_REQ1** requirement of the Specification document and select **Copy For» MATRIXx Comment** to copy the required covering syntax for a MATRIXx comment to cover the selected requirement.
2. Double-click the **MATRIXxModel.cat»Main»SuperBlocks» Classic Model** element of the MATRIXx Model document to launch MATRIXx SystemBuild and open the SuperBlock window.

3. In the SuperBlock window, select **File»SuperBlock Properties** to launch the SuperBlock Properties dialog box.
4. Click the **Comment** tab.
5. Press <Ctrl-V> to paste the text from the clipboard in the control. MATRIXx adds `[Covers: MODEL_REQ1]` to the comment, as shown in Figure 7-2.

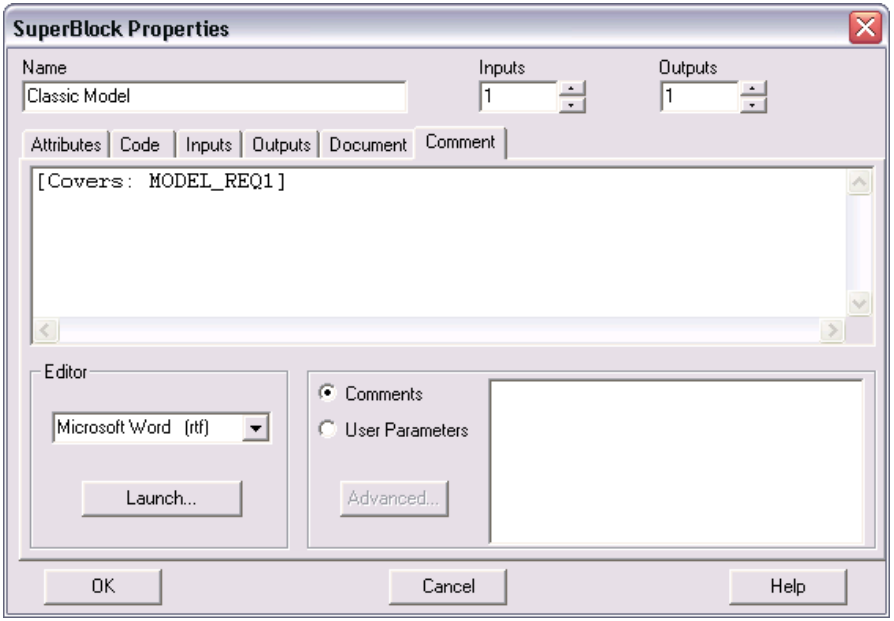


Figure 7-2. SuperBlock Properties Dialog Box

6. Click **OK** to close the SuperBlock Properties dialog box.
7. In the SuperBlock window, select the **Compensator** block.
8. Select **Edit»Block Properties** to launch the Block Properties dialog box.
9. Click the **Comment** tab.
10. Enter `[Covers: MODEL_REQ2]` in the control.
11. Click **OK** to close the Block Properties dialog box.
12. Repeat steps 7 through 11 to enter `[Covers: MODEL_REQ3]` to the comment for the Plant block and `[Covers: MODEL_REQ4]` to the comment for the Feedback Gain block.
13. Select **File»Update** to save the changes to memory.
14. Select **File»Close Window** to close the superblock.

15. In the SystemBuild Catalog Browser, select **File»Save As** to save the changes to disk. Select `<Requirements Gateway Public>\Tutorial\MATRIXxModel.cat` and overwrite the file on disk.
16. Exit MATRIXx and return to Requirements Gateway. Click **Yes** when Requirements Gateway prompts you to reload the MATRIXx Model document.
17. Expand the two documents in the Management View. Requirements Gateway indicates that the coverage is now 100%.
18. Click the **Coverage Analysis View** tab and the **Graphical View** tab to review the coverage of requirements in the Specification document by the elements of the MATRIXx Model document.

8

Using NI Requirements Gateway with LabVIEW

The LabVIEW type allows you to add LabVIEW documents to a project, including VIs and project and library files. When you configure a LabVIEW document, you select one or more VI, project, or library files. You typically use a LabVIEW document to cover requirements in a specification document. For example, the VIs in an LLB can cover the requirements for functions in an instrument driver, or the VIs in an application can cover requirements for the graphical user interface and its underlying logic.

LabVIEW Type

The LabVIEW type analyzes traceability information specified in the description fields of a VI or the controls and indicators of a VI. You can also enable the Analyze Diagrams variable for a LabVIEW document to analyze descriptions and labels of block diagram structures, functions, and wires.

The LabVIEW type defines the following document settings you configure for a LabVIEW document on the Document Details pane of the Project pane of the Configuration dialog box:

- **File or Directory**—Select one or more VI or LLB files or directories that contain VI or LLB files. You can specify whether to include all the subVIs of a selected VI or LLB or to include subdirectories of a selected directory.
- **Variables**—Defines the following document variables:
  - **Analyze Diagrams**—Instructs Requirements Gateway to analyze the block diagrams of the VIs in the document. The variable is disabled by default. Enabling the Analyze Diagrams variable for a document increases the time required for Requirements Gateway to analyze the document. Enable the option in the **Value** column to activate this variable.
  - **With Images**—Captures the front panel and diagram images of selected VIs. Enable the option in the **Value** column to activate this variable.

The LabVIEW type defines the following default elements:

- **Section**—A selected VI, VIs within a directory or LLB, and the controls and indicators of each VI that contains traceability information.
- **Macro-Requirement**—Not defined.
- **Requirement**—Specified by `[REQ: reqid]`, where `reqid` is the requirement identifier, in a description field of a VI, control, or indicator.
- **Entity**—Not defined.
- **Reference**—Specified by `[Covers: reqid]`, where `reqid` is the requirement identifier, in a description field of a VI, control, or indicator, or in the descriptions and labels of block diagram structures, functions, and wires if you enable the Analyze Diagrams variable for the LabVIEW document.
- **Attribute**—Defines the following attribute elements:
  - **Attribute**—Specified by `(#name)`, where `name` is the name of the attribute, in a description field of a VI, control, or indicator. The attribute must appear after the corresponding requirement.
- **Reference Attribute**—Defines the following reference attribute elements:
  - **Attribute**—Specified by `(*name)`, where `name` is the name of the attribute, in a description field of a VI, control, or indicator, or in the descriptions and labels of block diagram structures, functions, and wires if you enable the Analyze Diagrams variable for the LabVIEW document. The attribute must appear after the corresponding reference.
- **Link**—Not defined.
- **Text**—Not defined.
- **Picture**—Not defined.

When you create a custom LabVIEW type, you do not need to customize any type definition settings on the Analysis tab of the Types pane of the Configuration dialog box. Refer to Chapter 5, *Customizing Types*, for more information about creating custom types.

Refer to the *Coupling LabVIEW with NI Requirements Gateway* document for more information about using the LabVIEW type in Requirements Gateway.

Adding LabVIEW Documents to Projects

Complete the following steps to add a LabVIEW document to a project.

1. Open `<Requirements Gateway Public>\Tutorial\LabVIEWProject.rqtf`.
2. Click the **Management View** tab if it is not already selected.
3. Double-click the **Specification** document in the tree view to launch `LabVIEWSpec.txt` in an external application.
4. Review the content of the text file in the external application. The Specification document contains five requirements. The requirement in the Logic section specifies the purpose of the VI. The three requirements in the Front Panel section specify controls and indicators for the front panel. The requirement in the Block Diagram section specifies nodes of the block diagram.
5. Exit the application that displays `LabVIEWSpec.txt`.
6. In Requirements Gateway, launch the Project pane of the Configuration dialog box to edit the project.
7. Click the **Add a document** button to add a new document in the Traceability Description Zone. Place the document below the Specification document.
8. On the Document Details pane, enter `LabVIEW Files` in the **Name** control and select **LabVIEW** from the **Type of Analysis** ring control.
9. Click in the **File or Directory** column. The File Browse button becomes visible on the right side of the control. Click the **File Browse** button to launch the Select Files to Include in Document dialog box.

   Use the Select Files to Include in Document dialog box to include multiple VI files in a LabVIEW document in a Requirements Gateway project. You can add individual VI files, all the VIs from an LLB, or all the VIs in a directory. In addition, when you select a directory added in the list view of the Select Files to Include in Document dialog box, you can specify to include all of the subdirectories, and you can specify to include the subVIs of a selected VI.

   Refer to the *Coupling LabVIEW with NI Requirements Gateway* document for more information about the Select Files to Include in Document dialog box for a LabVIEW document.
10. Click **Add LabVIEW File** and select `<Requirements Gateway Public>\Tutorial\LabVIEWSignalAdder.vi`.
11. Click **Open** to add the file to the list view of the Select Files to Include in Document dialog box.

12. Click **OK** to close the Select Files to Include in Document dialog box.
13. With the LabVIEW document selected in the Traceability Description Zone, select **Analyze Diagrams** from the Variable ring control and enable the option in the Value column to activate the variable, as shown in Figure 8-1.

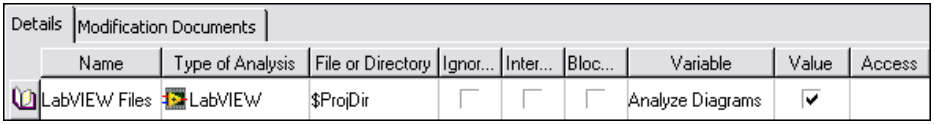


Figure 8-1. Analyze Diagrams Variable Activated for the LabVIEW Document

The Analyze Diagrams variable instructs the LabVIEW type to analyze the block diagrams of the VIs in the document to identify traceability information specified in the descriptions and labels of block diagram structures, functions, and wires.

14. Click the **Add a cover** button to begin adding a covering link. Click the **LabVIEW Files** document and then click the **Specification** document to create an arrow between the two documents.
15. Click **OK** to close the Configuration dialog box.
16. Click **Yes** when Requirements Gateway prompts you to reanalyze the project.
17. Expand the two documents in the Management View, as shown in Figure 8-2. Requirements Gateway displays the `LabVIEWSignalAdder.vi` file under the LabVIEW Files document.

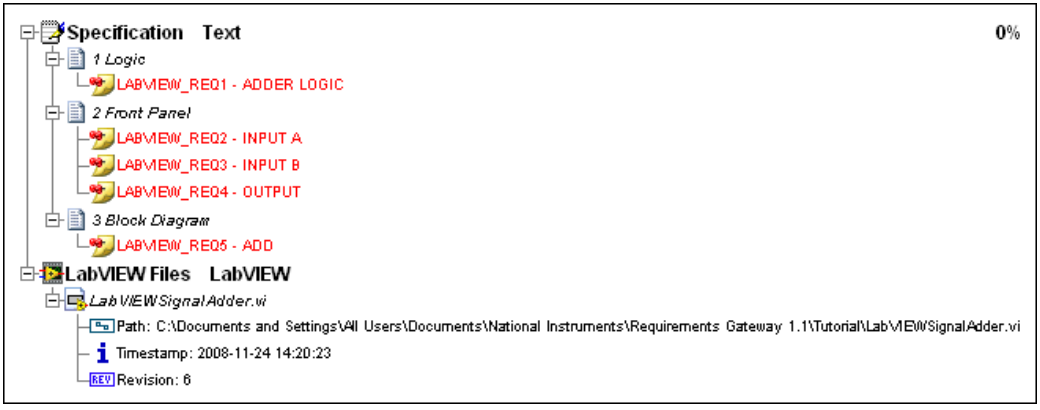


Figure 8-2. LabVIEW Document Expanded in the Management View

Adding References to LabVIEW VIs

Complete the following steps to specify references in a LabVIEW VI that cover the requirements in the Specification document.

1. Select the **LABVIEW_REQ1** requirement in the Specification document.
2. Right-click the **LABVIEW_REQ1** requirement and select **Copy For»LabVIEW Text Reference** to copy the required covering syntax for a LabVIEW comment to cover the selected requirement.
3. Double-click the **LabVIEWSignalAdder.vi** file under the LabVIEW Files document to launch LabVIEW and open the VI front panel.
4. In LabVIEW, select **File»VI Properties** to launch the VI Properties dialog box.
5. Select **Documentation** from the **Category** ring control.
6. Press <Ctrl-V> to paste the text from the clipboard in the **VI description** control after the existing comments. LabVIEW adds `[Covers: LABVIEW_REQ1]` to the description, as shown in Figure 8-3.

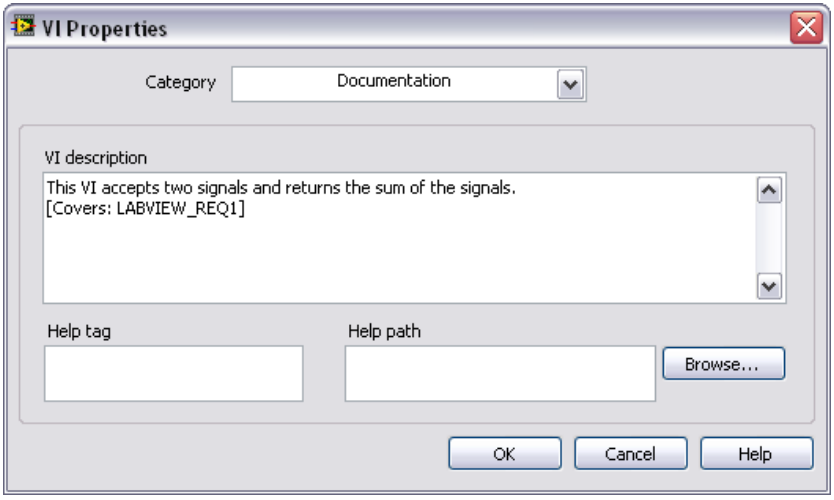


Figure 8-3. LabVIEW VI Properties Dialog Box

7. Click **OK** to close the VI Properties dialog box.
8. Right-click the **Input A** control on the front panel and select **Description and Tip** to launch the Description and Tip dialog box.

9. Enter `[Covers: LABVIEW_REQ2]` in the Description control, as shown in Figure 8-4.

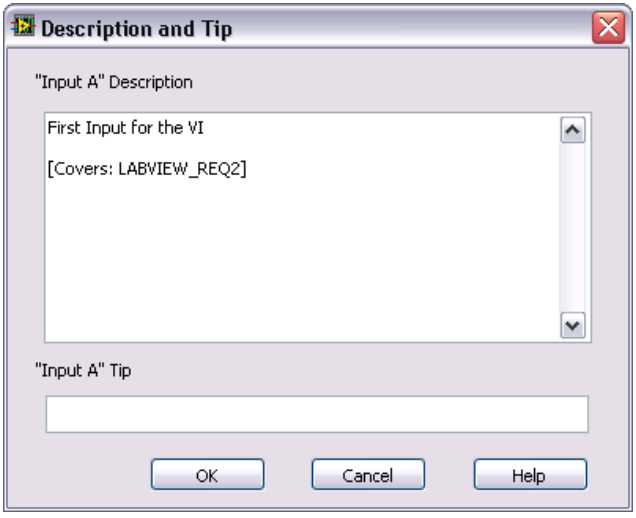


Figure 8-4. LabVIEW Description and Tip Dialog Box

10. Click **OK** to close the Description and Tip dialog box.
11. Repeat steps 8 through 10 to add `[Covers: LABVIEW_REQ3]` to the comment for the **Input B** control, and `[Covers: LABVIEW_REQ4]` to the comment for the **Output** indicator.
12. Select **Window»Show Block Diagram** to open the block diagram.
13. Right-click the **Add** node and select **Description and Tip** from the context menu to launch the Description and Tip dialog box.
14. Enter `[Covers: LABVIEW_REQ5]` in the Description control.
15. Click **OK** to close the Description and Tip dialog box.
16. Select **File»Save** to save the changes to memory.
17. Exit LabVIEW and return to Requirements Gateway. Click **Yes** when Requirements Gateway prompts you to reload the LabVIEW Files document.
18. Expand the two documents in the Management View. Requirements Gateway displays the controls, indicators, and objects that contain references under the `LabVIEWSignalAdder.vi` file, and indicates that the coverage is now 100%.
19. Click the **Coverage Analysis View** tab and the **Graphical View** tab to review the coverage of requirements in the Specification document by the elements of the LabVIEW Files document.
20. Close the project file.

9

Using NI Requirements Gateway with LabWindows/CVI

Use the LabWindows/CVI type to add LabWindows/CVI documents to a project. When you configure a LabWindows/CVI document, you select one or more source files. You typically use a LabWindows/CVI document to cover requirements in a specification document. For example, the functions in source files can cover the requirements for a library or tests or the requirements for a graphical user interface and its underlying logic.

LabWindows/CVI Type

The LabWindows/CVI type analyzes traceability information specified in source code files or in the help for function panel files. The default LabWindows/CVI type searches `.c`, `.h`, and `.fp` files and displays the function with C source and function panel files. The default type recognizes C and C++ style comments in source files.

The LabWindows/CVI type defines the following document settings you configure for a LabWindows/CVI document on the Document Details pane of the Project pane of the Configuration dialog box:

- **File or Directory**—Selects individual source files, a directory that contains source files, or a LabWindows/CVI workspace or project file. For a directory, you can specify to include subdirectories.
- **Variables**—Not defined.

The LabWindows/CVI type defines the following default elements:

- **Section**—Defines the following section elements:
  - **Directory**—The selected directory and subdirectories.
  - **File**—A file located in a directory or project.
  - **Function**—A function in a file.
- **Macro-Requirement**—Not defined.
- **Requirement**—Not defined.
- **Entity**—Not defined.

- **Reference**—Specified by `Implements reqid`, where `reqid` is the requirement identifier, within a C or C++ comment or in the help of a function panel.

  If the comment is in the function or immediately precedes the function, references are associated with the function. If the comment is not in the function or does not immediately precede the function, the references are associated with the file.
- **Attribute**—Not defined.
- **Reference Attribute**—Not defined.
- **Link**—Not defined.
- **Text**—Not defined.
- **Picture**—Not defined.

When you create a custom LabWindows/CVI type, you do not need to customize any type definition settings on the Analysis tab of the Types pane of the Configuration dialog box. Refer to Chapter 5, *Customizing Types*, for more information about creating custom types.

Refer to the *Coupling LabWindows/CVI with NI Requirements Gateway* document for more information about using the LabWindows/CVI type in Requirements Gateway.

Adding LabWindows/CVI Documents to Projects

Complete the following steps to add a LabWindows/CVI document to a project.

1. Open `<Requirements Gateway Public>\Tutorial\CVILibrary.rqtf`.
2. Click the **Management View** tab if it is not already selected.
3. Double-click the **Specification** document in the tree view to launch `CVISpec.txt` in an external application.
4. Review the content of the text file in the external application. The Specification document contains five requirements. The requirement in the Header section specifies a requirement for the header file. The four requirements in the Source Code section specify functions for the library.
5. Exit the application that displays `CVISpec.txt`.
6. In Requirements Gateway, launch the Project pane of the Configuration dialog box to edit the project.

7. Click the **Add a document** button to add a new document in the Traceability Description Zone. Place the document below the Specification document.
8. On the Document Details pane, enter `CVI Library` in the **Name** control and select **LabWindows/CVI** from the **Type of Analysis** ring control.
9. Click in the **File or Directory** column. The File Browse button becomes visible on the right side of the control. Click the **File Browse** button to launch the Select Files to Include in Document dialog box.

   Use the Select Files to Include in Document dialog box to include multiple files in a LabWindows/CVI document in a Requirements Gateway project. You can add individual source files, a workspace or project file, or all the files in a directory. In addition, when you select a directory added in the list view of the Select Files to Include in Document dialog box, you can specify to include all of the subdirectories.

   Refer to the *Coupling LabWindows/CVI with NI Requirements Gateway* document for more information about the Select Files to Include in Document dialog box for a LabWindows/CVI document.
10. Click **Add Workspace or Project File** and select `<Requirements Gateway Public>\Tutorial\CVILibrary.cws`.
11. Click **Open** to add the file to the list view of the Select Files to Include in Document dialog box.
12. Click **OK** to close the Select Files to Include in Document dialog box.
13. Click the **Add a cover** button to begin adding a covering link. Click the **CVI Library** document and then click the **Specification** document to create an arrow between the two documents.
14. Click **OK** to close the Configuration dialog box.
15. Click **Yes** when Requirements Gateway prompts you to reanalyze the project.

16. Expand the two documents in the Management View, as shown in Figure 9-1. Requirements Gateway displays the LabWindows/CVI workspace under the CVI Library document.

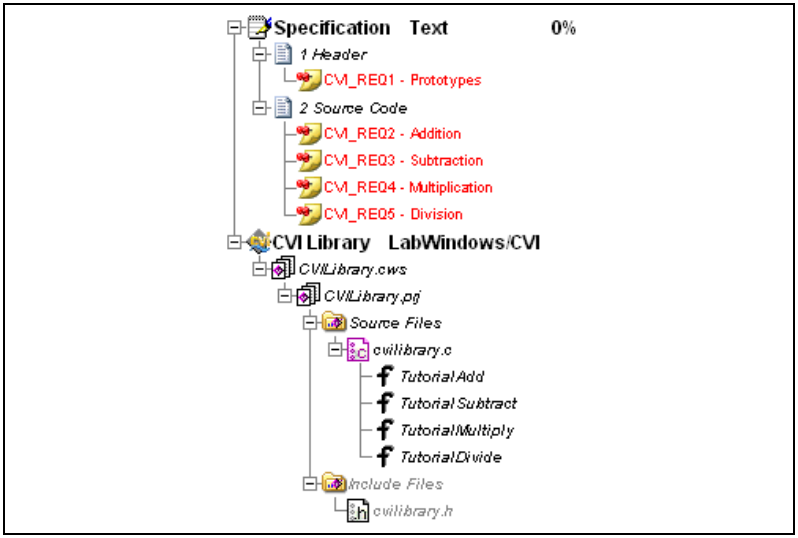


Figure 9-1. LabWindows/CVI Document Expanded in the Management View

Adding References to LabWindows/CVI Files

Complete the following steps to specify references in a LabWindows/CVI source file that cover the requirements in the Specification document:

1. Select the **CVI_REQ1** requirement in the Specification document.
2. Right-click the **CVI_REQ1** requirement and select **Copy For» LabWindows/CVI Comment** to copy the required covering syntax for a LabWindows/CVI comment to cover the selected requirement.
3. Double-click the **CVILibrary.cws** file under the CVI Library document to launch LabWindows/CVI and open the workspace.
4. In LabWindows/CVI, double-click the **CVILibrary.h** file in the tree view to open the file.
5. Place the cursor at the beginning of the second line in the file and press <Ctrl-V> to paste the text from the clipboard. LabWindows/CVI adds `// Implements CVI_REQ1` to the file functions, as shown in Figure 9-2.

```
// This is the header for the static library.
// Implements CVI_REQ1

// Function prototypes
float TutorialAdd (float a, float b);
float TutorialSubtract (float a, float b);
float TutorialMultiply (float a, float b);
float TutorialDivide (float a, float b);
```

Figure 9-2. CVILibrary.h

6. Save and close the `CVILibrary.h` file but do not exit LabWindows/CVI.
7. Double-click the **CVILibrary.c** file in the tree view to open the file.
8. Enter the comments for each function, as highlighted in Figure 9-3.

```
// This is the source file for the static library.


// Function Add
// Implements CVI_REQ2
float TutorialAdd (float a, float b)
{
     return a + b;
}

// Function Subtract
// Implements CVI_REQ3
float TutorialSubtract (float a, float b)
{
     return a - b;
}

// Function Multiply
// Implements CVI_REQ4
float TutorialMultiply (float a, float b)
{
     return a * b;
}

// Function Divide
// Implements CVI_REQ5
float TutorialDivide (float a, float b)
{
     return a / b;
}
```

Figure 9-3. CVILibrary.c

9. Save and close the `CVILibrary.c` file.
10. Exit LabWindows/CVI and return to Requirements Gateway. Click **Yes** when Requirements Gateway prompts you to reload the LabWindows/CVI Files document.
11. Expand the two documents in the Management View. Requirements Gateway indicates that the coverage is now 100%.
12. Click the **Coverage Analysis View** tab and the **Graphical View** tab to review the coverage of requirements in the Specification document by the elements of the CVI Library document.

10

Using NI Requirements Gateway with Telelogic DOORS

Use the Telelogic DOORS types to add DOORS documents to a project. You typically use a DOORS document to define requirements for a Requirements Gateway project. When you configure a DOORS document, you select a formal module in DOORS as the source of traceability information.

A DOORS module contains a tree of objects similar to numeric headings in a Microsoft Word document. An object specifies header text, contains descriptive text, and contains a unique ID within the module. Each object in a module can define its own attribute values. The data types for attributes include Boolean, numeric, date, and string values.

Requirements Gateway includes DOORS Basic and DOORS Advanced types, each of which adhere to different formalisms for identifying requirements. The following tutorial uses the DOORS Advanced type.

You can also refer to the `<Requirements Gateway Public>\Examples\Doors` directory for an additional example of using the DOORS types in Requirements Gateway.

DOORS Types

The DOORS types analyze traceability information specified in DOORS modules. Requirements Gateway supports DOORS 9.0 and earlier on Windows 2000/XP and only DOORS 9.0 on Windows Vista.

Note Known issues exist between DOORS 8.0 and Requirements Gateway. If you are using DOORS 8.0, National Instruments recommends installing the 8.0.5 or later patch.

DOORS Basic Type

The DOORS Basic type uses a Requirement Boolean object attribute to determine if objects in modules are requirements. In addition, the type uses the ID of the object as the requirement identifier. The organization and format of the module data might not adhere to the format that the default DOORS types require. You can customize the default definition of the DOORS Basic type to conform to the formalism of the modules in a DOORS database.

Figure 10-1 shows an example of a DOORS module that contains the default requirement traceability information for the DOORS Basic type.

ID		Requirement
REQ_89	**1 Example Computer Specification**	False
REQ_90	**1.1 *Power Supply***	False
REQ_91	**1.1.1 Voltages**	False
REQ_92	**1.1.1.1 PowerInputVoltages** The power supply must support input voltages from 110V-125VAC, both 50Hz and 60 Hz. The maximum allowed current draw is 6 amps.	True
REQ_93	**1.1.1.2 PowerOutputVoltages** ABCThe power supply must support the following output voltages: 3.3 volts +/-5 volts +/-12 volts +/-9 volts	True

Figure 10-1. DOORS Basic Module

The DOORS Basic type defines the following document settings you configure on the Document Details pane of the Project pane of the Configuration dialog box:

- **File or Directory**—Selects a module in the DOORS database.
- **Variables**—Defines the following document variables:
  - **Capture diagrams**—Specifies to import DOORS images. Enable the option in the **Value** column to activate this variable.
  - **Extract only defined attributes**—Specifies to extract only the attributes from objects that the type defines to improve performance when collecting data from DOORS. Enable the option in the **Value** column to activate this variable.

  - **Baseline**—Specifies the baseline you want to extract from the DOORS database. Enter the baseline in the **Value** column. You can also specify a baseline for a database in the **Baseline** control in the Select DOORS module dialog box.
  - **Server**—Specifies the server you want to consider for the DOORS database. Enter the server name in the **Value** column. You can also specify a server for the database in the **Server** control in the Select DOORS module dialog box.
  - **Extract objects from**—Specifies that you want to use a DOORS Extension Language (DXL) regular expression to extract objects from a DOORS module. Enter the DXL regular expression in the **Value** column. When a parent object satisfies requirements, all sub-objects are extracted. Refer to DOORS documentation for more information about DXL expressions.

  Refer to the *Adding DOORS Documents to Projects* section and to the *Coupling Telelogic DOORS with NI Requirements Gateway* document for more information about the Select DOORS module dialog box.

The DOORS Basic type defines the following default elements:

- **Section**—Indicates the objects in a module.
- **Macro-Requirement**—Not defined.
- **Requirement**—Specified by a value of `True` in the Requirement Boolean object attribute. The type uses the ID of the object as the requirement identifier.
- **Entity**—Not defined.
- **Reference**—An out-link from an object in the module to another object.
- **Attribute**—Not defined.
- **Reference Attribute**—Not defined.
- **Link**—Not defined.
- **Text**—Not defined.
- **Picture**—Not defined.

When you create a custom DOORS Basic type, you might need to customize the following type definition settings on the Analysis tab of the Types pane of the Configuration dialog box:

- **Convert Tool**—Supports the Doors and DoorsXML options.

Refer to Chapter 5, *Customizing Types*, for more information about creating custom types.

Refer to the *Coupling Telelogic DOORS with NI Requirements Gateway* document for more information about the Convert Tool options and using the DOORS Basic type in Requirements Gateway.

DOORS Advanced Type

The default DOORS Advanced type uses the value of the ObjectType object attribute in DOORS to determine if objects in modules are requirements. In addition, the type uses the value of the ReqID object attribute as the requirement identifier. You can customize the default definition of the DOORS Advanced type to conform to the formalism of the modules in a DOORS database.

Figure 10-2 shows an example of a DOORS module that contains the default requirement traceability information for the DOORS Advanced type.

ID	ReqID		ObjectType	Priority
1		**1 Calling the elevator**		
2	REQ1	A potential passenger can be on any of the floors and can call an elevator by pressing either the up or button to call the elevator.	Requirement	High
3	REQ2	The potential passenger waits for the doors to open before entering into the elevator. The potential passenger now becomes a passenger	Requirement	Medium
6		**2 In the elevator**		
7	REQ3	Once in an elevator, a passenger can select the floor or a number of floors where he wants to go to. Modif	Requirement	High
8		Each elevator will have a list of floors to visit : Once the elevator has been called by a potential passenger or a passenger has selected a destination, then the elevator will move to the appropriate floor.	Comment	

Figure 10-2. DOORS Advanced Module

The DOORS Advanced type defines the following document settings you configure on the Document Details pane of the Project pane of the Configuration dialog box:

- **File or Directory**—Selects a module in the DOORS database.
- **Variables**—Defines the same document variables as the DOORS Basic type. Refer to the *DOORS Basic Type* section for information about the variables of the DOORS Basic type.

The DOORS Advanced type defines the following default elements:

- **Section**—Indicates the objects in a module.
- **Macro-Requirement**—Not defined.
- **Requirement**—Specified by a value of `Requirement` in the ObjectType object attribute. The type uses the value of the ReqID object attribute as the requirement identifier.
- **Entity**—Not defined.
- **Reference**—Specified by an out-link or in-link for an object in the module.
- **Attribute**—Defines the following attribute elements:
  - **Priority**—Specified by the Priority object attribute.
- **Reference Attribute**—Not defined.
- **Link**—Not defined.
- **Text**—Not defined.
- **Picture**—Not defined.

When you create a custom DOORS Advanced type, you might need to customize the following type definition settings on the Analysis tab of the Types pane of the Configuration dialog box:

- **Convert Tool**—Supports the Doors and DoorsXML options.

Refer to Chapter 5, *Customizing Types*, for more information about creating custom types.

Refer to the *Coupling Telelogic DOORS with NI Requirements Gateway* document for more information about the Convert Tool options and using the DOORS Advanced type in Requirements Gateway.

Defining Requirements in DOORS

Complete the following steps to import a tutorial project into DOORS.

1. Launch DOORS and log in to the database. Your user login must have the authority to create projects.
2. Select **File»Restore»Project** to launch the Restore Project dialog box.
3. Click **Browse** and open `<Requirements Gateway Public>\Tutorial\RequirementsGatewayTutorial.dpa`.
4. Select **RequirementsGatewayTutorial.dpa** and click **Open**.
5. Click **OK** in the Restore Project dialog box. DOORS displays the content of the project in the Restore Project dialog box. Click **OK** to create the RequirementsGatewayTutorial project in the database.
6. Select the **RequirementsGatewayTutorial** project in the tree view to display the Requirements module in the list view, as shown in Figure 10-3.

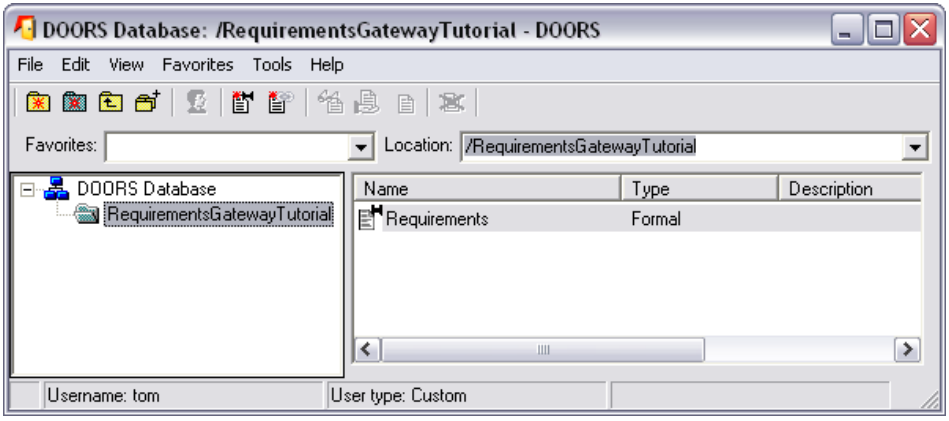


Figure 10-3. DOORS Database

7. Double-click the **Requirements** module to open the Formal Module window, as shown in Figure 10-4.

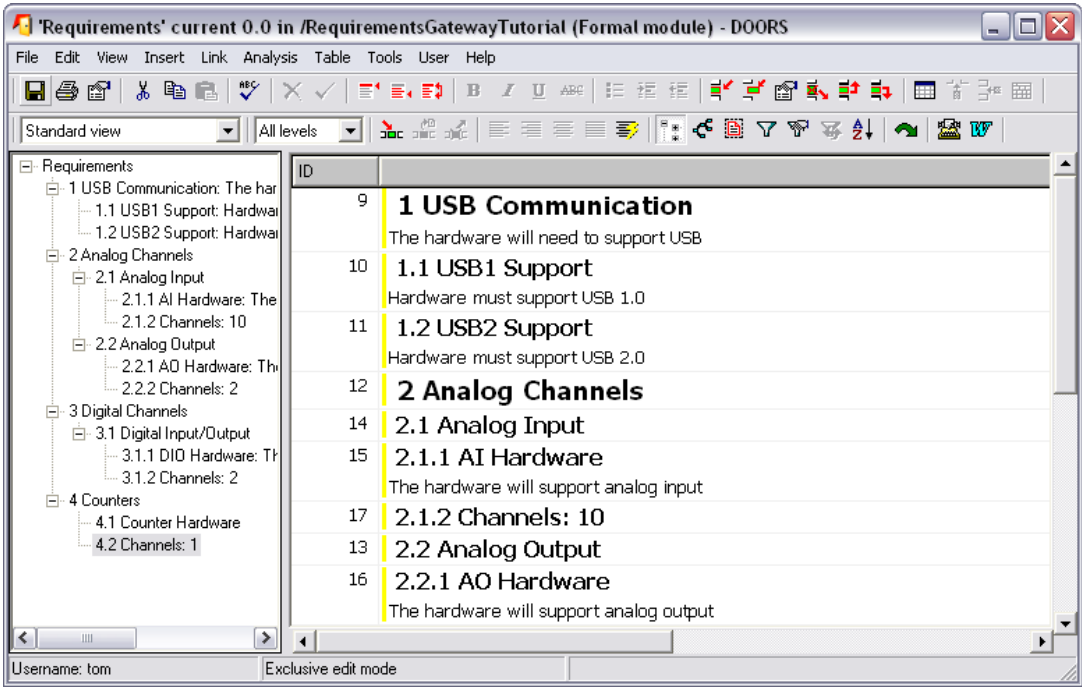


Figure 10-4. DOORS Formal Module Window

The Requirements module contains similar information to the text specification document, `ProductSpec.txt`, used in Chapter 2, *Managing Requirements*. The DOORS Advanced type uses attribute values to determine if an object is a requirement and to specify the requirement identifier.

8. In the Formal Module window, select **Insert»Column** to launch the New Column dialog box.
9. Select **ObjectType** from the **Attribute** ring control and enter `ObjectType` in the **Column Title** control.
10. Click **OK** to close the New Column dialog box. DOORS adds a new column to the view.
11. Repeat steps 8 through 10 to add another column in the display for the ReqID attribute. The DOORS Advanced type interprets objects with the ObjectType attribute set to Requirement as a requirement, and uses the ReqID attribute value as the requirement identifier.

Figure 10-5 shows the Formal Module window with the added columns.

ID		ObjectType	ReqID
9	**1 USB Communication** The hardware will need to support USB		
10	1.1 USB1 Support Hardware must support USB 1.0	Requirement	PS_USB_REQ1
11	1.2 USB2 Support Hardware must support USB 2.0	Requirement	PS_USB_REQ2
12	**2 Analog Channels**		
14	2.1 Analog Input		
15	2.1.1 AI Hardware The hardware will support analog input	Requirement	PS_AI_REQ1
17	2.1.2 Channels: 10	Requirement	PS_AI_REQ2
13	2.2 Analog Output		
16	2.2.1 AO Hardware The hardware will support analog output	Requirement	PS_AO_REQ1
18	2.2.2 Channels: 2	Requirement	PS_AO_REQ2

Figure 10-5. New Columns in Formal Module Window

12. Select **View»Save As** to launch the Save As dialog box.
13. Enter `Requirements Gateway Advanced` in the **Name** control and click **OK** to save the changes to the view and close the Save As dialog box.
14. Select **File»Close** to close the Formal Module window.

Adding DOORS Documents to Projects

Complete the following steps to create a project with a DOORS document.

1. In Requirements Gateway, select **File»New** or click the **New** button on the toolbar in the main window to launch the Create a New Project and Save As dialog box, and navigate to the `<Requirements Gateway Public>\Tutorial` directory.
2. Enter `DOORSProject` in the **File name** control and click **Save**. Requirements Gateway launches the Project pane of the Configuration dialog box.
3. Click the **Add a document** button to place a document in the Traceability Description Zone.

4. On the Document Details pane, enter `Product Specification` in the **Name** control and select **Doors Advanced** in the **Type of Analysis** ring control.
5. Click in the **File or Directory** column. The File Browse button becomes visible on the right side of the control. Click the **File Browse** button to launch the Select DOORS module dialog box.

   Use the Select DOORS module dialog box to log in and navigate to a DOORS database to select a module.
6. Enter your DOORS username and password in the DOORS login section and click the **Update DOORS tree** button, shown at left. Requirements Gateway accesses the DOORS database and lists the projects and modules in the tree view.

7. Select the **DOORS Database»RequirementsGatewayTutorial» Requirements** element in the tree view.
8. Click **OK** to close the Select DOORS module dialog box.
9. Click the **Add a document** button to add a second document object in the Traceability Description Zone. Place the document below the Product Specification document.
10. On the Document Details pane, enter `Covering Specification` in the **Name** control, select **Text** in the **Type of Analysis** ring control, and navigate to `<Requirements Gateway Public>\Tutorial\CoveringSpec.txt` in the File or Directory control.
11. Click the **Add a cover** button to begin adding a covering link. Click the **Covering Specification** document and then click the **Product Specification** document to create an arrow between the two documents.
12. Click **OK** to close the Configuration dialog box.
13. Click **Yes** when Requirements Gateway prompts you to reanalyze the project.

14. Expand the two documents in the Management View, as shown in Figure 10-6.

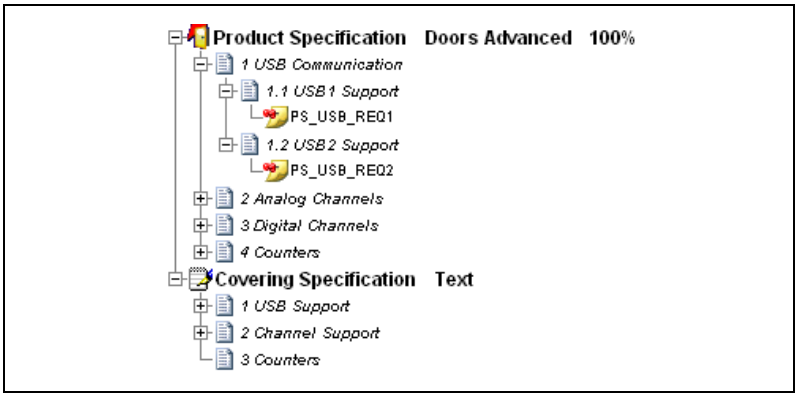


Figure 10-6. DOORS Document Expanded in the Management View

Requirements Gateway displays the content of the DOORS module and indicates that the coverage is now 100%.

Note If you complete this tutorial before you complete the tutorial in Chapter 2, *Managing Requirements*, Requirements Gateway displays the coverage as 90%.

15. Click the **Coverage Analysis View** tab and the **Graphical View** tab to review the coverage of requirements in the Specification document by the elements of the DOORS module.

Exporting Documents to DOORS

You can export documents and traceability links from covering documents in a Requirements Gateway project back to the DOORS database so you can access the Requirements Gateway analysis directly in DOORS. The DOORS types create a new module for the covering document and links the references in the document to the requirements in DOORS. Refer to the *Coupling Telelogic DOORS with NI Requirements Gateway* document for more information about exporting traceability information to DOORS.

Complete the following steps to export the Covering Specification document to the DOORS database.

1. In Requirements Gateway, select the **Covering Specification** document on the tree view pane of the Management View.
2. Click the **Export Document to DOORS** button, shown at left, in the toolbar in the main window to launch the Export document to DOORS dialog box.
3. Select the **RequirementsGatewayTutorial** element in the **Target** tree view.
4. Enter `Covering Module` in the **New module** control, as shown in Figure 10-7.

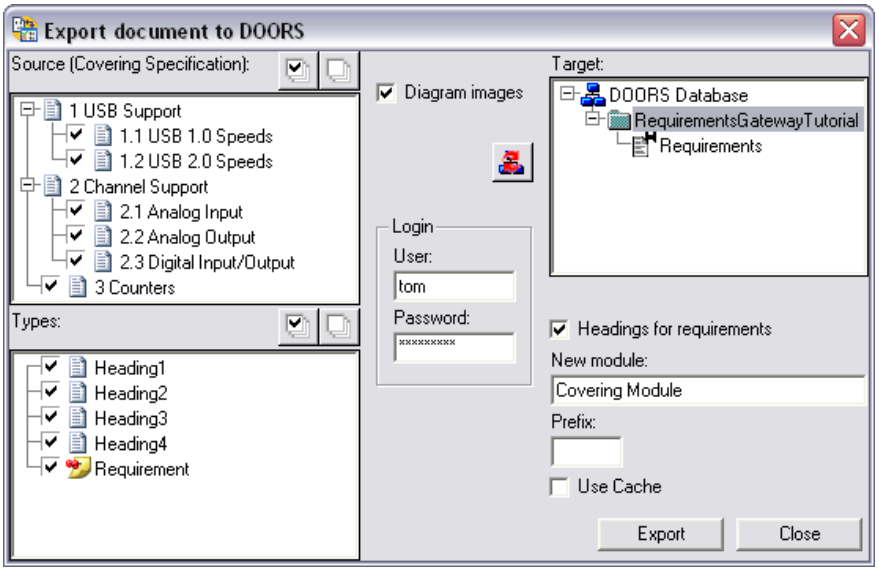


Figure 10-7. Export Document to DOORS Dialog Box

5. Click **Export** to launch a confirmation dialog box that lists the new elements to export to DOORS, as shown in Figure 10-8.

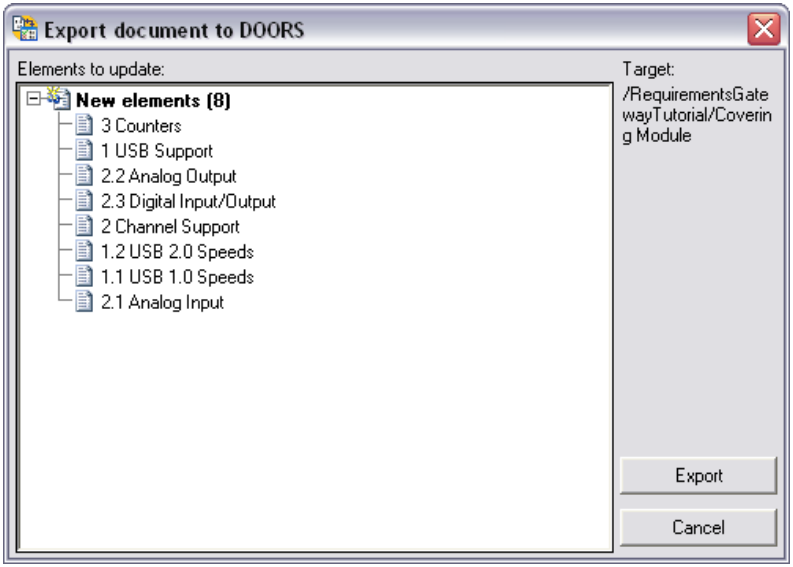


Figure 10-8. Confirmation for Export Document to DOORS

6. Click **Export** again.
7. Click **OK** when Requirements Gateway displays a message that the operation completed.

Note If you did not close the Formal Module window for the Requirements Module in DOORS, Requirements Gateway cannot export the Covering Specification document and create links. If the export operation fails, close the Formal Module window and try again.

8. Click **Close** to close the Export document to DOORS dialog box.
9. In DOORS, select **View»Refresh** to display the Covering Module in the database.

10. Double-click **Covering Module** to display the module in the Formal Module window, as shown in Figure 10-9.

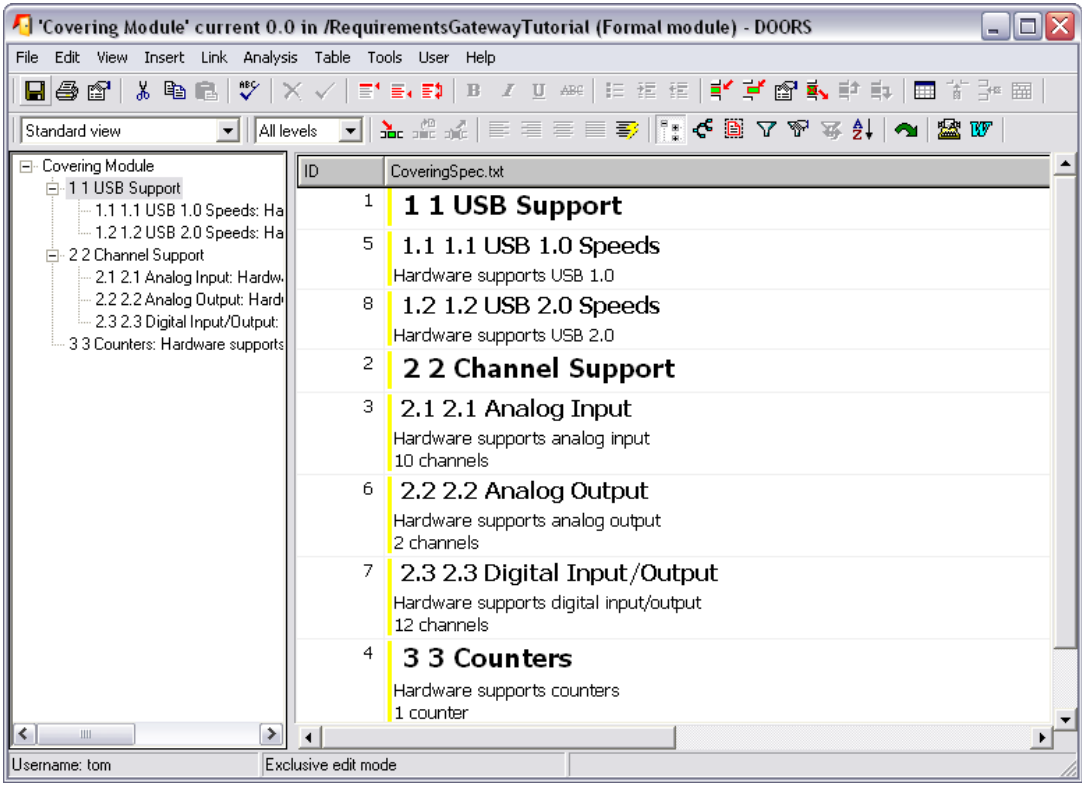


Figure 10-9. CoveringSpec.txt in DOORS

The content of the module mirrors the traceability information that Requirements Gateway displays, and links exist between the Covering Module and the Requirements Module.

11. Select **File»Close** to close the Formal Module window and then select **File»Exit** to exit DOORS.
12. Close the project file in Requirements Gateway.

A

Third-Party Types Overview

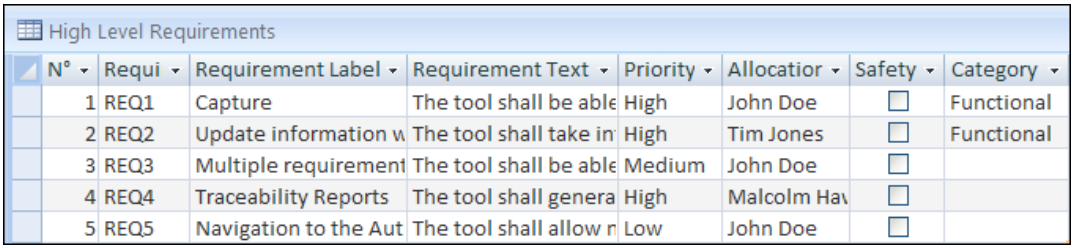
In addition to the TestStand, MATRIXx, LabVIEW, LabWindows/CVI, and DOORS types discussed in Chapters 6 through 10, Requirements Gateway supports the following types:

- Microsoft Access
- Adobe Acrobat PDF
- Checksum
- Code
- Code C
- Microsoft Excel
- Large Code
- IBM Rational RequisitePro
- Text
- Microsoft Word
- Microsoft Visio

Access

Use the Access type to analyze traceability information that tables of a Microsoft Access database specify. The default type analyzes the text located in fields with specific names.

Figure A-1 shows an example of a table that contains the default requirement traceability information for the Access type.



High Level Requirements

N°	Requi	Requirement Label	Requirement Text	Priority	Allocatior	Safety	Category
1	REQ1	Capture	The tool shall be able	High	John Doe	☐	Functional
2	REQ2	Update information w	The tool shall take in	High	Tim Jones	☐	Functional
3	REQ3	Multiple requirement	The tool shall be able	Medium	John Doe	☐	
4	REQ4	Traceability Reports	The tool shall genera	High	Malcolm Hav	☐	
5	REQ5	Navigation to the Aut	The tool shall allow r	Low	John Doe	☐	

Figure A-1. Access Table

The Access type defines the following document settings you configure on the Document Details pane of the Project pane of the Configuration dialog box:

- **File or Directory**—Selects an Access database file (`.mdb`).
- **Variables**—Defines the following document variables:
  - **Table**—Specifies that you want to capture traceability information from one or more specific tables in an Access document. Enter the names of the tables, separated by an exclamation point (`!`), in the **Value** column.

The Access type defines the following default elements:

- **Section**—Defines the following section elements:
  - **Table**—Identifies the tables within the database.
- **Macro-Requirement**—Not defined.
- **Requirement**—The requirement identifier in the Requirement_ID field of a table, and the requirement label in the Requirement_Label field of a table.
- **Entity**—Not defined.
- **Reference**—Specified by entering the list of requirement identifiers, separated by commas, in the Covered_Requirements field of a table.
- **Attribute**—Defines the following attribute elements:
  - **Priority**—A text value in the Priority field of a table.
  - **Allocation**—A text value in the Allocation field of a table.
  - **Safety**—A Boolean value in the Safety field of a table.
  - **Category**—A text value in the Category field of a table.
- **Reference Attribute**—Not defined.

- **Link**—Not defined.
- **Text**—Text in the Requirement_Text field of a table.
- **Picture**—Not defined.

Note When Requirements Gateway opens an Access database, Access might display a Security Warning dialog box that prompts you for permission to open the database. If you do not give permission to open the database, or if Access already has the database open with exclusive access, Requirements Gateway fails to open the database and displays an error for the Access document in the main window.

The organization and format of a database table can vary and might not adhere to the format the default Access type requires. You can customize the definition of the Access type to conform to the data format that a table contains. Refer to Chapter 5, *Customizing Types*, for more information about creating custom types.

When you create a custom Access type, you do not need to customize any type definition settings on the Analysis tab of the Types pane of the Configuration dialog box.

Refer to the *Coupling Microsoft Access with NI Requirements Gateway* document for more information about using the Access type in Requirements Gateway.

Acrobat PDF

Use the Acrobat PDF type to analyze traceability information specified in Adobe Acrobat PDF files.

Figure A-2 shows an example of a PDF file that contains the default requirement traceability information for the Acrobat PDF type.

```
This is an example PDF document

1. Heading 1 Text
1.1. Heading 2 Text
    [Covers: PS_REQ1]

    DS_REQ1: Label
    <<Derived Requirement Text>>
    Priority: High

1.1.1. Heading 3 Text
[MacroReq_DS_ALL]
    <<Macro Requirement Text>>
    [Covers: PS_REQ2]

    DS_REQ2: Requirement Label
    <<Requirement Text>>
    DS_REQ3: Requirement Label
    <<Requirement Text>>
[End_of_MacroReq]
```

Figure A-2. Acrobat PDF Document

The Acrobat PDF type defines the following document settings you configure on the Document Details pane of the Project pane of the Configuration dialog box:

- **File or Directory**—Selects an Acrobat PDF file (`.pdf`).
- **Variables**—Defines the following document variables:
  - **Notes format**—Captures sticky notes added to the PDF document. You can capture the text of the note and the body text to which the note applies. Enter a regular expression in the **Value** column to specify the content you want to capture. Refer to the *Coupling Adobe Acrobat with NI Requirements Gateway* document for more information about capturing notes in a PDF document.

The Acrobat PDF type defines the following default elements:

- **Section**—Specified by `n.n.n.n. text`, where `n` is a numeric value and `text` is the label for the section. The type supports up to four numeric levels.
- **Macro-Requirement**—Specified by `[MacroReq_reqid]`, where `reqid` is the requirement identifier, and ends with `[End_of_MacroReq]`.
- **Requirement**—Specified by `prefixREQnn: label`, where `prefix` is non-spaced alphanumeric characters, `nn` is a numeric value, and `label` is any text.
- **Entity**—Not defined.
- **Reference**—Specified by `[Covers: reqid1, reqid2]`, where `reqid` is the identifier of the covered requirement.
- **Attribute**—Defines the following attribute elements:
  - **Priority**—Specified by `Priority: text`, where `text` must be one of the following: Low, Med, or High. The attribute must appear after the corresponding requirement.
  - **Allocation**—Specified by `Allocated to: text`, where `text` is the attribute value.
- **Reference Attribute**—Not defined.
- **Link**—Not defined.
- **Text**—Specified by `<<text>>`, where `text` is the requirement text value. The requirement text must appear on a new line after the corresponding section or requirement element.
- **Picture**—Not defined.

The organization and format of text data can vary between files and might not adhere to the format that the default Acrobat PDF type requires. You can customize the definition of the Acrobat PDF type to conform to the data format that a file contains.

When you create a custom Acrobat PDF type, you do not need to customize any type definition settings on the Analysis tab of the Types pane of the Configuration dialog box. Refer to Chapter 5, *Customizing Types*, for more information about creating custom types.

Refer to the *Coupling Adobe Acrobat with NI Requirements Gateway* document for more information about using the Acrobat PDF type in Requirements Gateway.

Checksum

Use the Checksum type to provide a high-level analysis of multiple directories and files. The Checksum type does not capture any specific traceability information from source files. When you apply a Checksum type to a document, specify the files and directories for the Checksum document in the project and use the Graphical View to create internal coverage links to add traceability information to the document. Requirements Gateway creates modification documents for these links.

The Checksum type defines the following document settings you configure on the Document Details pane of the Project pane of the Configuration dialog box:

- **File or Directory**—Selects an individual file of any type or a directory that contains multiple files of any type.
- **Variables**—Defines the following document variables:
  - **File Filters**—Specifies that you want to include specific files when you select a directory for the Checksum document. In the **Value** column, enter the filenames and extensions, separated by semicolons and using wildcards, that you want to include. For example, enter `*.h` to include all files with the `.h` extension.

The Checksum type defines the following default elements:

- **Section**—Defines the following section elements:
  - **Folder**—The selected directory and any subdirectories.
  - **File**—A file located in a directory.
  - **Root File**—An individual file not included in an entire specified directory.
- **Macro-Requirement**—Not defined.
- **Requirement**—Not defined.
- **Entity**—Not defined.
- **Reference**—Not defined.

- **Attribute**—Defines the following attribute elements:
  - **Date Time**—Specified by the `date_time` XML attribute of the `<file>` tag of the intermediate file. The Checksum type generates an XML intermediate file that contains file modification information for all the files included in the Checksum document. The Date Time attribute specifies the date and time at which the file was last modified.
  - **MD5**—Specified by the `md5` XML attribute of the `<file>` tag of the intermediate file. The MD5 attribute specifies a Message-Digest algorithm 5 (MD5) signature for the file.
- **Reference Attribute**—Not defined.
- **Link**—Not defined.
- **Text**—Not defined.
- **Picture**—Not defined.

The organization and format of a directory and its files might not adhere to the format that the default Checksum type requires. You can customize the definition of the Checksum type to conform to the data format that a file contains.

When you create a custom Checksum type, you do not need to customize any type definition settings on the Analysis tab of the Types pane of the Configuration dialog box. Refer to Chapter 5, *Customizing Types*, for more information about creating custom types.

Refer to the *Coupling Checksum with NI Requirements Gateway* document for more information about using the Checksum type in Requirements Gateway.

Code

Use the Code type to analyze traceability information specified in source code files. The default Code type searches for C++ style comments in `.c`, `.h`, and, `.cpp` source files. The type does not list the functions defined in the files.

Figure A-3 shows an example of a Code file that contains the default requirement traceability information for the Code type.

```
// This is example code
// Implements REQ_LIBRARY

// Implements REQ_FUNC1
int Function1(int a, int b)
{
    return a + b;
}
```

Figure A-3. Code Type Source Code Document

The Code type defines the following document settings you configure on the Document Details pane of the Project pane of the Configuration dialog box:

- **File or Directory**—Selects a directory that contains source files (`.c`, `.h`, `.cpp`).
- **Variables**—Defines the following document variables:
  - **Prompt when files change**—Specifies if Requirements Gateway prompts you to reload the project when you modify the source file. Enable the option in the **Value** column to activate this variable.

The Code type defines the following default elements:

- **Section**—Defines the following section elements:
  - **Directory**—The selected directory and any subdirectories.
  - **File**—A file located in a directory.
- **Macro-Requirement**—Not defined.
- **Requirement**—Not defined.
- **Entity**—Not defined.
- **Reference**—Specified by `// Implements reqid`, where `reqid` is the requirement identifier, within a source file.
- **Attribute**—Not defined.

- **Reference Attribute**—Not defined.
- **Link**—Not defined.
- **Text**—Not defined.
- **Picture**—Not defined.

The organization and format of a code file might not adhere to the format that the default Code type requires. You can customize the definition of the Code type to conform to the comment style that a file contains.

When you create a custom Code type, you might need to customize the following type definition settings on the Analysis tab of the Types pane of the Configuration dialog box:

- **Encoding**—The type of file to read. The default options include:
  - **Default**—ANSI encoding (ISO 8859-1).
  - **Shift_JIS**—Character encoding for Japanese.
  - **UCS2**—2-byte Unicode Little Endian format.
  - **UCS2be**—2-byte Unicode Big Endian format.
  - **UTF8**—8-bit Unicode transformation format.
- **Filters**—The file types to analyze.
- **Include subdirectories**—Specifies to search for files in the subdirectories of the selected directory you specify.

Refer to Chapter 5, *Customizing Types*, for more information about creating custom types.

Refer to the *Coupling Code Files with NI Requirements Gateway* document for more information about using the Code type in Requirements Gateway.

Code C

Use the Code C type to analyze traceability information specified in C and C++ source code files. The default Code C type searches `.c`, `.h`, and, `.cpp` files and displays the functions defined in the `.c` and `.cpp` source files. The default type recognizes C and C++ style comments.

Figure A-4 shows an example of a Code C file that contains the default requirement traceability information for the Code C type.

```
// This is example code
// Implements REQ_LIBRARY

// --------------------
// Implements REQ_FUNC1
// --------------------
int Function1(int a, int b)
{
    // Implements REQ_MAIN
    return a + b;
}

/* -------------------------
   Implements REQ_FUNC2
-------------------------*/
int Function2(int a, int b)
{
    return a + b;
}
```

Figure A-4. Code C Type Source Code Document

The Code C type defines the following document settings you configure on the Document Details pane of the Project pane of the Configuration dialog box:

- **File or Directory**—Selects a directory that contains source files (`.c`, `.h`, `.cpp`).
- **Variables**—Defines the following document variables:
  - **Prompt when files change**—Specifies if Requirements Gateway prompts you to reload the project when you modify the source file. Enable the option in the **Value** column to activate this variable.
  - **Includes function body**—Captures the text associated with functions in a C source file so that you can track changes to traceability information stored in the associated text. Enable the option in the **Value** column to activate this variable.

The Code C type defines the following default elements:

- **Section**—Defines the following section elements:
  - **Directory**—The selected directory and any subdirectories.
  - **File**—A file located in a directory.
  - **Function**—A function in a file.
- **Macro-Requirement**—Not defined.
- **Requirement**—Not defined.
- **Entity**—Not defined.
- **Reference**—Specified by `Implements reqid`, where `reqid` is the requirement identifier, within a C or C++ comment of a source file. A reference is associated with a function if the comment is in the function or immediately precedes the function, otherwise the references is associated with the file.
- **Attribute**—Not defined.
- **Reference Attribute**—Not defined.
- **Link**—Not defined.
- **Text**—Not defined.
- **Picture**—Not defined.

The organization and format of a code file might not adhere to the format that the default Code C type requires. You can customize the definition of the Code C type to conform to the comment style that a file contains.

When you create a custom Code C type, you might need to customize the following type definition settings on the Analysis tab of the Types pane of the Configuration dialog box:

- **Filters**—Specifies the file types to analyze.
- **Include subdirectories**—Specifies to search for files in the subdirectories on the selected directory you specify.

Refer to Chapter 5, *Customizing Types*, for more information about creating custom types.

Refer to the *Coupling Code Files with NI Requirements Gateway* document for more information about using the Code C type in Requirements Gateway.

Excel

Use the Excel type to analyze traceability information specified in Microsoft Excel worksheets. The default Excel type searches worksheets for traceability information that adheres to a specific format with specific columns.

Note The Excel type does not support Excel 2007 (`.xlsx`) files, but you can use Excel 2007 in compatibility mode to open Excel 2003 or earlier (`.xls`) files. Use the ExcelX type for Excel 2007 files. Refer to the *ExcelX* section for more information about the ExcelX type.

Figure A-5 shows an example of a worksheet that contains the default requirement traceability information for the Excel type.

REQ1	Capture	The tool shall be able to capture semi-automatically the requirements included in a document and/or in a model. "Semi-automatic" means the text has to be formalized beforehand by the user or another dedicated tool.	High	John Doe
REQ2	Update information when source changes	The tool shall take into account the successive versions of the documents and models and update automatically the traceability information presented to the user.	High	Tim Jones

Figure A-5. Excel Worksheet

The Excel type defines the following document settings you configure on the Document Details pane of the Project pane of the Configuration dialog box:

- **File or Directory**—Selects an Excel file (`.xls`).
- **Variables**—Defines the following document variables:
  - **Worksheet**—Captures traceability information from one or more specific worksheets in an Excel document. Enter the names of the sheets, separated by commas, in the **Value** column.

The Excel type defines the following default elements:

- **Section**—Indicates the sheets within a worksheet file.
- **Macro-Requirement**—Not defined.

- **Requirement**—Specified by `REQnn`, where `nn` is a numeric value in the requirement identifier, in the first column of a sheet. The text in the second cell specifies the label for the requirement.
- **Entity**—Not defined.
- **Reference**—Specified by `[Covers: reqid1, reqid2]`, where `reqid` is the identifier of the covered requirement, within a cell of a sheet.
- **Attribute**—Defines the following attribute elements:
  - **Priority**—Specified by the text located in the fourth cell in a row that contains a requirement. The value must be one of the following: Low, Med, or High.
  - **Allocated**—Specified by the text located in the fifth cell in a row that contains a requirement.
- **Reference Attribute**—Not defined.
- **Link**—Not defined.
- **Text**—Specified by the text located in the third cell in a row that contains a requirement.
- **Picture**—Not defined.

The organization and format of worksheet data can vary between documents and might not adhere to the format the default Excel type requires. You can customize the definition of the Excel type to conform to the data format that a worksheet contains.

When you create a custom Excel type, you might need to customize the following type definition settings on the Analysis tab of the Types pane of the Configuration dialog box:

- **Convert Tool**—Supports the Excel, ExcelWithColNumbers, and ExcelX options.

Refer to Chapter 5, *Customizing Types*, for more information about creating custom types.

Refer to the *Coupling Microsoft Excel with NI Requirements Gateway* document for more information about the Convert Tool options and using the Excel type in Requirements Gateway.

ExcelX

Use the ExcelX type to capture traceability information from Excel 2007 (`.xlsx`) worksheets. The ExcelX type defines the same document settings and default elements as the Excel type. Refer to the *Coupling Microsoft Excel with NI Requirements Gateway* document for more information about using the ExcelX type in Requirements Gateway.

Large Code

Use the Large Code type to analyze traceability information specified in a large number of source code files. The default Large Code type searches for C++ style comments in `.c`, `.h`, and, `.cpp` source files. The type does not list the functions defined in the files. When you configure a document that uses the Large Code type, you must specify a Filter Expression variable that the type uses to select the lines in source files to process.

Figure A-6 shows an example of a Large Code file that contains the default requirement traceability information for the Large Code type.

```
// This is example code
// Implements REQ_LIBRARY

// Implements REQ_FUNC1
int Function1(int a, int b)
{
    return a + b;
}
```

Figure A-6. Large Code Type Source Code Document

The Large Code type defines the following document settings you configure on the Document Details pane of the Project pane of the Configuration dialog box:

- **File or Directory**—Selects a directory that contains source files (`.c`, `.h`, `.cpp`).
- **Variables**—Defines the following document variables:
  - **Filter Expression**—Specifies that you want to use a qualifying regular expression for the type to locate lines within the source code files. Enter the regular expression in the **Value** column. The Large Code type analyzes the specified directory for lines that match the specified expression. For the default type, you typically specify `(.*Implements.*)` as the filter expression.

The filter expression applies only to the selected Large Code project document. Other documents that use the Large Code type can specify unique filter expressions.

The Large Code type defines the following default elements:

- **Section**—Defines the following section elements:
  - **Directory**—The selected directory and any subdirectories.
  - **File**—A file located in a directory.
- **Macro-Requirement**—Not defined.
- **Requirement**—Not defined.
- **Entity**—Not defined.
- **Reference**—Specified by `// Implements reqid`, where `reqid` is the requirement identifier, within a source file.
- **Attribute**—Not defined.
- **Reference Attribute**—Not defined.
- **Link**—Not defined.
- **Text**—Not defined.
- **Picture**—Not defined.

The organization and format of code files might not adhere to the format that the default Large Code type requires. You can customize the definition of the Large Code type to conform to the comment style that files contain.

When you create a custom Large Code type, you might need to customize the following type definition settings on the Analysis tab of the Types pane of the Configuration dialog box:

- **Encoding**—The type of file to read. The default options include:
  - **Default**—ANSI encoding (ISO 8859-1)
  - **UTF8**—8-bit Unicode transformation format
  - **UCS2**—2-byte Unicode Little Endian format
  - **UCS2be**—2-byte Unicode Big Endian format
- **Filters**—The file types to analyze.
- **Include subdirectories**—Specifies to search for files in the subdirectories of the selected directory you specify.

Refer to Chapter 5, *Customizing Types*, for more information about creating custom types.

Refer to the *Coupling Code Files with NI Requirements Gateway* document for more information about using the Large Code type in Requirements Gateway.

RequisitePro

Use the RequisitePro type to analyze traceability information specified in the packages of IBM Rational RequisitePro projects.

Figure A-7 shows an example of a RequisitePro package that contains the default requirement traceability information for the RequisitePro type.

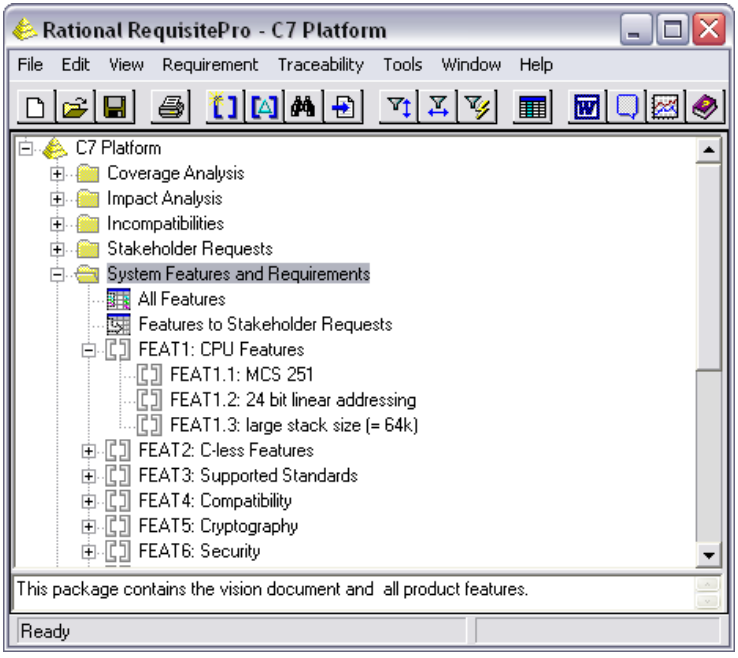


Figure A-7. IBM Rational RequisitePro Project

The RequisitePro type defines the following document settings you configure on the Document Details pane of the Project pane of the Configuration dialog box:

- **File or Directory**—Selects an IBM Rational RequisitePro project file (`.rgs`).
- **Variables**—Defines the following document variables:
  - **Package name**—Specifies that you want to analyze a specific package in the RequisitePro database. Enter the package name in the Value column. If you selected a package in the Choose RequisitePro Package dialog box, the package name already appears in the **Value** column. If you do not specify a package name, the type analyzes all packages in the project. Refer to the *Coupling IBM Rational RequisitePro with NI Requirements Gateway* document for more information about the Choose RequisitePro Package dialog box.

The RequisitePro type defines the following default elements:

- **Section**—Displays the packages with a project.
- **Macro-Requirement**—Not defined.
- **Requirement**—Specified by requirements within a package. The identifier for the requirement in RequisitePro is the requirement identifier in Requirements Gateway.
- **Entity**—Not defined.
- **Reference**—Specified by any inverse references.
- **Attribute**—Defines the following attribute elements:
  - **Priority**—The value of the priority attribute associated with a requirement.
  - **Difficulty**—The value of the difficulty attribute associated with a requirement.
- **Reference Attribute**—Not defined.
- **Link**—Not defined.
- **Text**—The text associated with a requirement or package in a project.
- **Picture**—Not defined.

The organization and format of package information might not adhere to the format that the default RequisitePro type requires. You can customize the definition of the RequisitePro type to conform to the data format that a package contains.

When you create a custom RequisitePro type, you do not need to customize any type definition settings on the Analysis tab of the Types pane of the Configuration dialog box. Refer to Chapter 5, *Customizing Types*, for more information about creating custom types.

Refer to the *Coupling IBM Rational RequisitePro with NI Requirements Gateway* document for more information about using the RequisitePro type in Requirements Gateway.

Text

Use the Text type to analyze traceability information specified in text files.

Figure A-8 shows an example of a text file that contains the default requirement traceability information for the Text type.

```
This is an example text document

1. Heading 1 Text
1.1. Heading 2 Text
    [Covers: PS_REQ1]

DS_REQ1: Label
    <<Derived Requirement Text>>
    - Priority: High

1.1.1. Heading 3 Text
[MacroReq_DS_ALL]
    <<Macro Requirement Text>>
    [Covers: PS_REQ2]

DS_REQ2: Requirement Label
    <<Requirement Text>>
DS_REQ3: Requirement Label
    <<Requirement Text>>
[End_of_MacroReq]
```

Figure A-8. Text Document

The Text type defines the following document settings you configure on the Document Details pane of the Project pane of the Configuration dialog box:

- **File or Directory**—Selects a text file (`.*`).
- **Variables**—Not defined.

The Text type defines the following default elements:

- **Section**—Specified by `n.n.n.n. text`, where `n` is a numeric value and `text` is the label for the section. The type supports up to four numeric levels.
- **Macro-Requirement**—Specified by `[MacroReq_reqid]`, where `reqid` is the requirement identifier, and completed with `[End_of_MacroReq]`.
- **Requirement**—Specified by `prefixREQnn: label`, where `prefix` is non-spaced alphanumeric characters, `nn` is a numeric value, and `label` is any text.
- **Entity**—Not defined.
- **Reference**—Specified by `[Covers: reqid1, reqid2]`, where `reqid` is the identifier of the covered requirement.
- **Attribute**—Defines the following attribute elements:
  - **Priority**—Specified by `-Priority: text`, where `text` must be `Low`, `Med`, or `High`. The attribute must appear after the corresponding requirement.
- **Reference attribute**—Not defined.
- **Link**—Not defined.
- **Text**—Specified by `<<text>>`, where `text` is the requirement text value. The requirement text must appear on a new line after the corresponding section or requirement element.
- **Picture**—Not defined.

The organization and format of text data can vary between files and might not adhere to the format that the default Text type requires. You can customize the definition of the Text type to conform to the data format that a file contains.

When you create a custom Text type, you might need to customize the following type definition settings on the Analysis tab of the Types pane of the Configuration dialog box:

- **Encoding**—The type of file to read. The default options include:
  - **Default**—ANSI encoding (ISO 8859-1)
  - **UTF8**—8-bit Unicode transformation format
  - **UCS2**—2-byte Unicode Little Endian format
  - **UCS2be**—2-byte Unicode Big Endian format

Refer to Chapter 5, *Customizing Types*, for more information about creating custom types.

Word

Use the Word type to analyze traceability information specified in a Microsoft Word file. Use the MultiWord type to analyze all Word files in a directory. The default Word types use styles to identify sections, requirements, text, and references and uses formatting to identify attributes and macro-requirements.

Note The Word type does not support Word 2007 (`.docx`) files, but you can use Word 2007 in compatibility mode to open Word 2003 or earlier (`.doc`) files. Use the WordX type for Word 2007 files. Refer to the *WordX* section for more information about the WordX type.

Figure A-9 shows an example of a Word file that contains the default requirement traceability information for the Word types.

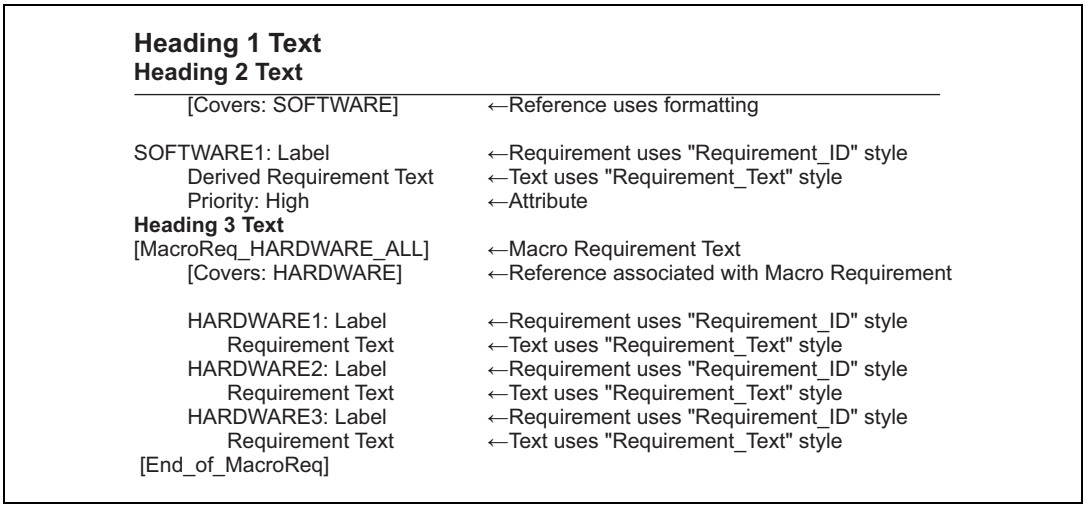


Figure A-9. Microsoft Word Document

The Word types define the following document settings you configure on the Document Details pane of the Project pane of the Configuration dialog box:

- **File or Directory**—Selects a Word file (`.doc`) when using the Word type or a directory using the MultiWord type.
- **Variables**—Defines the following document variables. Enable the option in the **Value** column to activate the selected variable.
  - **With images**—Captures images in the Word document. You can capture most drawings or images inserted in a Word file, including vectorized images in Windows Metafile (WMF) format, even if the images are inserted directly in a Word paragraph. If the image is larger than 10 MB, Requirements Gateway does not capture the image.
  - **With Character styles**—Adds information to the intermediate file about the Word character styles associated with the captured text. Enable this variable if you want to capture traceability information that uses specific character styles.
  - **With Bookmarks**—Converts bookmark content to a form that Requirements Gateway can read from the intermediate file and include in the analysis.
  - **With Annotations**—Converts annotation content, such as review comments, to a form that Requirements Gateway can read from the intermediate file and include in the analysis. This variable is not available for the WordX type.
  - **One Cell Per Line**—Adds a carriage return between each column of a table in the intermediate file. This variable is not available for the WordX type.

The Word types define the following default elements:

- **Section**—The text that uses the Heading style or numbering to indicate sections of a document.
- **Macro-Requirement**—Specified by `[MacroReq_reqid]`, where `reqid` is the requirement identifier and ends with `[End_of_MacroReq]`.
- **Requirement**—Specified by using the Requirement_ID style for the text on an entire line. A colon character delimits the label for a requirement.
- **Entity**—Not defined.
- **Reference**—Specified by `[Covers: reqid1, reqid2]`, where `reqid` is the identifier of the covered requirement.

- **Attribute**—Defines the following attribute elements:
  - **Priority**—Specified by `Priority: value`, where `value` must be one of the following: Low, Med, or High. The attribute must appear after the corresponding requirement.
  - **Allocation**—Specified by `Allocated to: text`, where `text` is the allocation identifier. The attribute must appear after the corresponding requirement.
- **Reference Attribute**—Not defined.
- **Link**—Not defined.
- **Text**—Specified by using the Requirement_Text style for the text that appears after the corresponding section or requirement element.
- **Picture**—Specified by inserting a picture into the document from the clipboard or from a file. You cannot specify a picture element that is linked in a Word document.

The organization and format of text data can vary between files and might not adhere to the format that the default Word types define. You can customize the definition of the Word types to conform to the data format that a file contains.

When you create a custom Word type, you might need to customize the following type definition settings on the Analysis tab of the Types pane of the Configuration dialog box:

- **Convert Tool**—Supports the Word, WordWithStyles, and WordX options.

Refer to Chapter 5, *Customizing Types*, for more information about creating custom types.

Refer to the *Coupling Microsoft Word with NI Requirements Gateway* document for more information about the Convert Tool options and using the Word type in Requirements Gateway.

WordX

Use the WordX type to capture traceability information from Word 2007 (`.docx`) files. The WordX type defines the same document settings and default elements as the Word type. Refer to the *Coupling Microsoft Word with NI Requirements Gateway* document for more information about using the WordX type in Requirements Gateway.

Visio

Use the Visio type to analyze traceability information specified in Microsoft Visio files. The default Visio type uses a Requirements Traceability custom property to identify requirements and references for pages and shapes.

Note Visio 2007 refers to custom properties as shape data.

Figure A-10 shows an example of a Visio drawing file that contains the default requirement traceability information for the Visio type. The Custom Properties window contains the custom properties for the drawing, including the Requirements Traceability custom property.

Note Visio 2007 refers to the Custom Properties window as the Shape Data window. Refer to the *Coupling Microsoft Visio with NI Requirements Gateway* document for more information about capturing traceability information from custom properties and shape data.

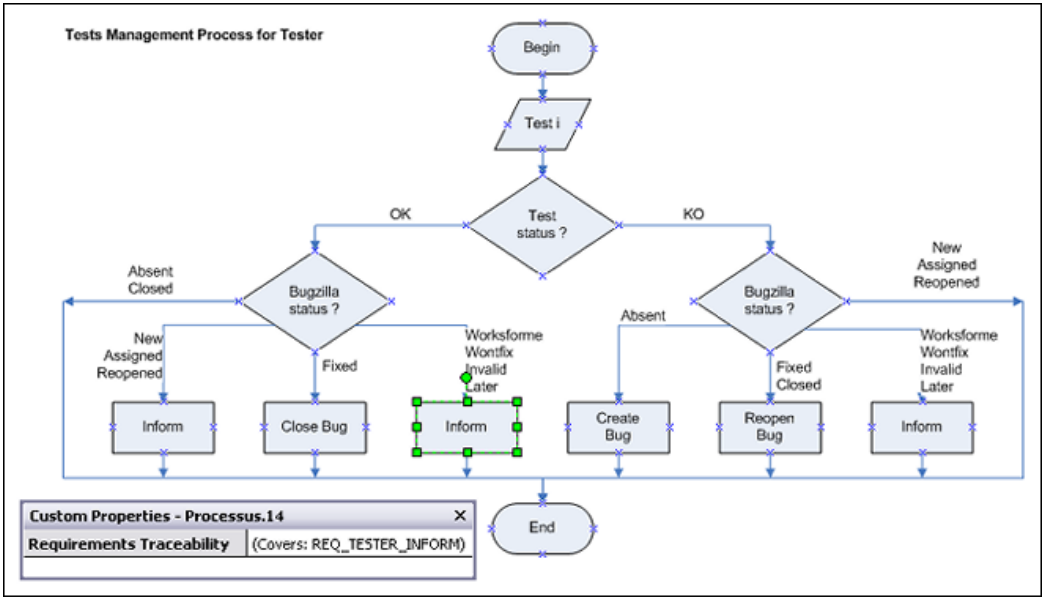


Figure A-10. Microsoft Visio Document

The Visio type defines the following document settings you configure on the Document Details pane of the Project pane of the Configuration dialog box:

- **File or Directory**—Selects a Visio file (`.vsd`).
- **Variables**—Not defined.

The Visio type defines the following default elements:

- **Section**—The pages in a file and the shapes on a page.
- **Macro-Requirement**—Not defined.
- **Requirement**—Specified by `VISIOnn`, where `nn` are numeric characters in the requirement identifier, in the Requirements Traceability custom property of a page or shape.
- **Entity**—Not defined.
- **Reference**—Specified by `[Covers: reqid1, reqid2]`, where `reqid` is the requirement identifier, in the Requirements Traceability custom property of a page or shape.
- **Attribute**—Not defined.
- **Reference Attribute**—Not defined.
- **Link**—Not defined.
- **Text**—Not defined.
- **Picture**—Captures the image of each page in a drawing file.

You can customize the definition of the Visio type to capture traceability information from a custom property with a name other than Requirements Traceability.

When you create a custom Visio type, you do not need to customize any type definition settings on the Analysis tab of the Types pane of the Configuration dialog box. Refer to Chapter 5, *Customizing Types*, for more information about creating custom types.

Refer to the *Coupling Microsoft Visio with NI Requirements Gateway* document for more information about using the Visio type in Requirements Gateway.

B

Technical Support and Professional Services

Visit the following sections of the award-winning National Instruments Web site at ni.com for technical support and professional services:

- **Support**—Technical support at ni.com/support includes the following resources:
  - **Self-Help Technical Resources**—For answers and solutions, visit ni.com/support for software drivers and updates, a searchable KnowledgeBase, product manuals, step-by-step troubleshooting wizards, thousands of example programs, tutorials, application notes, instrument drivers, and so on. Registered users also receive access to the NI Discussion Forums at ni.com/forums. NI Applications Engineers make sure every question submitted online receives an answer.
  - **Standard Service Program Membership**—This program entitles members to direct access to NI Applications Engineers via phone and email for one-to-one technical support as well as exclusive access to on demand training modules via the Services Resource Center. NI offers complementary membership for a full year after purchase, after which you may renew to continue your benefits.

    For information about other technical support options in your area, visit ni.com/services, or contact your local office at ni.com/contact.
- **Training and Certification**—Visit ni.com/training for self-paced training, eLearning virtual classrooms, interactive CDs, and Certification program information. You also can register for instructor-led, hands-on courses at locations around the world.
- **System Integration**—If you have time constraints, limited in-house technical resources, or other project challenges, National Instruments Alliance Partner members can help. To learn more, call your local NI office or visit ni.com/alliance.

If you searched ni.com and could not find the answers you need, contact your local office or NI corporate headquarters. Phone numbers for our worldwide offices are listed at the front of this manual. You also can visit the Worldwide Offices section of ni.com/niglobal to access the branch office Web sites, which provide up-to-date contact information, support phone numbers, email addresses, and current events.

Glossary

A

attribute	Describes a requirement. An attribute has a name and a Boolean, numeric, or string value. A requirement can only have one attribute with a specific name. You can use attributes and their values when you define filters, which specify which requirements to analyze or display.

C

covering document	A document that contains references to requirements that are defined in another document.

D

derived requirement	A requirement that is defined in a section or entity, but the section or entity does not cover requirements at a higher level.
downstream document	A document that contains references that cover the requirements in a selected document. When you configure a covering link in a project, a downstream document points to an upstream document.

E

entity	An element that must contain a reference to a requirement. If a defined entity does not contain any reference, Requirements Gateway displays a warning message.

F

formalism	Structural elements and syntax that define traceability information in a document. Documents must adhere to the formalism the types define in Requirements Gateway

L

link — A non-covering reference to a requirement or reference to a section or entity.

M

macro-requirement — Super-requirement that encapsulates requirements and passes its attributes, text, or links onto those requirements. If the macro-requirement is directly referenced by a forward element, all the requirements that it contains are considered referenced by this element.

R

reference — Indicates the coverage of a requirement. A reference points to a macro-requirement, requirement, or derived requirement.

reference attribute — Describes the type of coverage for a reference, such as partial coverage or provisional coverage. An attribute has a name and a Boolean, numeric, or string value. A reference can have only one attribute with a specific name.

regular expression — A mechanism to select specific text from within a string. An expression contains literal characters, wildcard characters, and operators to locate text patterns in the string. For example, the expression `REQ[0-9]+` matches any text that contains the characters `REQ` followed by one or more digits. Requirements Gateway implements regular expression pattern matching using the same syntax and semantics as Perl.

requirement — Expresses a need or constraint, such as a technical constraint, cost, or deadline.

S

section — A hierarchical element that represents structure within a document. Sections represent heading levels in a text or Microsoft Word file, objects within a file or database, or files and directories on disk.

T

text	Descriptive wording associated with a section, entity, requirement, or attribute.
type	Defines how to select external files that represent a document, how to read the content of the external files, how to interpret the content as elements for managing requirements, and how to display the elements of the document.

U

Units Under Test (UUTs)	The devices or components you are testing.
upstream document	A document that contains requirements that are covered by a selected document. When you configure a covering link in a project, a downstream document points to an upstream document.

Index

A

C

D

E

F

G

H

I

K

L

M

N

O

P

R

S

T

U

V

W