

Teamcenter 11.1 Systems Engineering and Requirements Management

Systems Architect/ Requirements Management User's Manual

Manual History

Manual Revision	Teamcenter Requirements Version	Publication Date
A	3.0	March 2003
B	3.0.1	May 2003
C	4.0	December 2003
D	4.1	February 2004
E	5.0	July 2004
F	6.0	March 2005
Manual Revision	Teamcenter Systems Engineering and Requirements Management Version	Publication Date
G	2005	September 2005
H	2005 SR1	June 2006
I	2007	December 2006
J	2007.1	April 2007
K	2007.2	September 2007
L	2007.3	January 2008
M	8	January 2009
N	8.0.1	June 2009
O	8.1	October 2009
P	8.2	October 2010
Q	9	July 2011
R	9.1	May 2012
R1	9.1.5	January 2014

Manual Revision	Teamcenter Systems Engineering and Requirements Management Version	Publication Date
S	10.0	January 2015
T	10.1	September 2016
U	11.1	March 2018

This edition obsoletes all previous editions.

Contents

- Manual History 4**
- Contents 7**
- Preface..... 17**
 - Audience 17
 - Conventions 17
 - Revision Marks 17
 - Browser and Dialog Window Examples 17
 - Names and Values 18
 - Submitting Comments..... 18
 - Proprietary and Restricted Rights Notice..... 18
- Chapter 1: Introducing Architect/Requirements..... 19**
 - Requirements Identification and Allocation 20
 - Identification..... 20
 - Allocation..... 21
 - Requirements Management..... 22
 - Object Types 23
 - Object Properties..... 25
 - Project Alignment 26
- Chapter 2: Installing the Architect/Requirements Client with Office Integration..... 27**
 - Introduction..... 27
 - Architect/Requirements Components 28
 - Prerequisites for Installing the Architect/Requirements Client with Office Integration..... 28
 - Limitations of Architect/Requirements Support for Microsoft Office Integration 30
 - Supporting Multiple Installations of Microsoft Office 30
 - Constraints for Supporting Microsoft Office Integration 30
 - Windows Client Hardware..... 31
 - New Visio 2013 and Visio 2016 Stencil and Diagram Formats Not Supported..... 31
 - Installing the Architect/Requirements Client with Office Integration 33
 - Troubleshooting the ICE_JNIRegistry.DLL Java console error..... 35
 - Using the Live Office Interface 37
 - Prerequisites for Using the Live Office Interface 37
 - Post Installation Procedures for Using the Live Office Interface 38
 - Using the Live Office Diagnosis..... 42

Uninstalling the Architect/Requirements Client	43
Chapter 3: Using the Architect/Requirements Main Window	45
Starting a New Architect/Requirements Session	45
Elements of the Architect/Requirements Main Window	48
Customizing the Main Window Display	50
Toolbar	51
Rearranging Button Groups Within the Toolbar Area	55
Displaying Button Groups in Floating Windows	55
Module Bar	57
Disabling or Enabling the Module Bar	57
Hiding or Showing the Module Bar	57
Resizing the Module Bar	57
Address Bar	58
Disabling or Enabling the Address Bar	58
Hiding or Showing the Address Bar	59
Resizing the Address Bar	59
Navigation Tree	60
Expanding a Project or a Folder	60
Disabling or Enabling the Navigation Tree	61
Disabling or Enabling the Root Node	61
Hiding or Showing the Navigation Tree	61
Switching to the Navigation Tree	61
Resizing the Navigation Tree	62
Content Table	63
Hierarchical Content Table	64
Architect/Requirements Recycle Bin	68
Disabling or Enabling the Content Table	70
Hiding or Showing the Content Table	70
Switching to the Content Table	70
Resizing the Content Table	71
Notebook Pane	72
Properties Tab	74
Attachments Tab	75
Links Tab	77
Connectivity Tab	83
Preview Tab	85
Where Used Tab	86
Versions Tab	88
Using Tabs in Floating Windows	90
Disabling or Enabling the Notebook Pane	93
Hiding or Showing the Notebook Pane	93
Switching to the Notebook Pane	93
Resizing the Notebook Pane	94
Viewing Objects in a Hierarchy	95
Filtering Objects in a Hierarchy	97

Showing and Hiding Filters	97
Filter Toolbar	98
Filter Choices	99
Filter Operators	101
Filtering Data	101
Saved Views	102
Printing a Selected Panel.....	102
Viewing Architect/Requirements System Properties.....	103
Working With the Client Log File	105
Setting Client Debugging Options	107
Changing Your Architect/Requirements Password.....	110
Exiting Architect/Requirements.....	110
Chapter 4: Maintaining a Project.....	111
Overview of Projects.....	111
Opening a Project.....	112
Opening a Folder.....	113
Creating a Folder.....	115
Copying Objects.....	119
Copying Object URLs.....	122
Moving Objects.....	123
Renaming an Object.....	125
Using Groups to Maintain Objects.....	126
Creating a Group.....	127
Adding Group Members	128
Removing Group Members.....	129
Working With Shortcuts	131
Creating a Shortcut	133
Navigating From a Shortcut to the Master Object	134
Hiding and Showing the Children of a Master Object.....	134
Replacing Shortcuts With Copies of the Master Objects.....	135
Sending Object Data by E-Mail	135
Exporting Objects	138
Exporting Objects to an AP233 STP File	138
Exporting Objects to an AP233 XML File	140
Exporting Objects to an Architect/Requirements XML File	142
Exporting Objects to Microsoft Office Excel	143
Exporting Objects to Microsoft Office Word	146
Exporting Objects for importing into Teamcenter.....	150
Importing Objects	152
Importing Objects From an AP233 STP File.....	152
Importing Objects From an AP233 XML File.....	154
Importing Objects From an Architect/Requirements XML File.....	156
Importing Objects From Microsoft Office Excel.....	158
Deleting Objects.....	162
Shortcut Objects.....	163
Deleting a Child Object	163

Restoring Objects.....	163
Emptying Your Architect/Requirements Recycle Bin.....	165
Chapter 5: Managing Requirements.....	167
Overview of Requirements.....	167
Creating a Requirement Object.....	168
Requirement Subtype Assignment and Inheritance.....	168
Creating a Requirement at the Top Level of a Folder.....	169
Creating a Sibling of an Existing Requirement.....	169
Creating a Child of an Existing Requirement.....	170
Creating Requirements and Content by Import From Microsoft Office Word.....	171
Document Parsing by Outline Levels Only.....	171
Document Parsing by Outline Levels and Keywords.....	171
Style Sheet of the Import Folder.....	173
To import requirements from Microsoft Office Word:.....	173
Entering and Changing Requirement Content in Microsoft Office Word.....	175
Content Elements.....	175
Content Formatting.....	175
Page Setup.....	177
Editing Requirement Content in Microsoft Office Word.....	177
Using OLE Objects in Word.....	178
Referencing Information in Requirement Text Edited in Microsoft Word.....	178
Editing Requirement Content in Text Property Cells.....	179
Enabling Text Property Editing for a Requirement.....	182
Organizing Requirements in a Hierarchy.....	184
Positions of Promoted Requirements.....	184
Positions of Demoted Requirements.....	184
Number Property Values of Promoted and Demoted Requirements.....	185
Promoting and Demoting Requirements.....	185
Viewing Requirement Content.....	186
Comparing the Content of Different Requirements.....	187
Setting the Comparison Mode.....	187
Comparing Two Requirements.....	188
Comparing the Requirements in Two Folders.....	189
Chapter 6: Constructing System Views With Building Blocks and Diagrams.....	191
Building Blocks.....	191
Creating a Building Block.....	192
Organizing Building Blocks in a Hierarchy.....	195
Live Visio Diagrams.....	197
User Actions in Live Visio Diagrams.....	197
Creating a Diagram.....	201
Creating a Child Diagram.....	203
Editing a Diagram.....	204
Copying a Microsoft Office Visio Diagram to Live Visio.....	209
Changing a Diagram Type.....	211

Disconnecting a Live Visio Diagram.....	213
Viewing a Diagram if Microsoft Office Visio Is Not Installed	214
Data Dictionaries and Data Definitions	215
Creating a Data Dictionary	215
Attaching a Data Dictionary to a Diagram.....	216
Creating Data Definitions	217
Ports and Connections.....	220
Creating Ports	220
Creating Connections in the Architect/Requirements Client.....	221
Working With Connections in a Live Visio Diagram.....	223
Attaching a Data Definition to a Connection.....	228
Detaching a Data Definition From a Connection.....	230
Viewing the Connections for an Object.....	231
Working With Trace Links in a Live Visio Diagram.....	231
Creating a Trace Link in a Live Visio Diagram.....	232
Copying a Trace Link to a Live Visio Diagram.....	232
Chapter 7: Showing Object Relationships With Trace Links	233
Overview of Trace Links	233
Creating Trace Links.....	234
Linking Objects Within Architect/Requirements	234
Linking to an Object in Another Teamcenter Product.....	237
Viewing Object Relationships	238
Viewing Objects in a Defining or Complying Path	238
Viewing Trace Links for an Object	241
Navigating to a Linked Object	242
Deleting Trace Links for an Object.....	243
Generic Links.....	244
Customizing Generic Links	244
Chapter 8: Recording Supplementary Information With Notes	245
Overview of Notes	245
Attaching a Note to an Object.....	246
Creating a Plain Text Note.....	247
Creating a Rich Text Note	248
Creating a Note in Microsoft Office Word.....	250
Editing a Note	252
Enabling Text Property Editing	255
Editing Text Property Cells.....	257
Viewing Note Content	259
Chapter 9: Working With Object Properties.....	261
Overview of Object Properties.....	261
System-Defined Properties	261
User-Defined Properties	263
Property Values.....	263
Releasing Reservation of Objects	264

Customizing Views of Property Columns.....	265
Conditions That Determine the Column View in the Content Table.....	266
Applying a View in the Content Table	267
Applying a View in the Notebook Pane	268
Creating a Column View	269
Setting the Default View for a Folder.....	270
Setting the Default View for Your User Name.....	271
Modifying a View Locally.....	272
Modifying a View in the Database	276
Filtering Property Tables	277
Editing the Properties of a Selected Object.....	279
Property Editing for Multiple Object Selections.....	282
Conditions of Property Display for Multiple Selections.....	282
Editing Properties for Multiple Objects.....	283
Editing Properties in Table View Cells.....	286
Calculating Numeric Property Values With Formulas	288
Updating Properties From an AP 233 STP File	290
Using the Live Excel Interface.....	292
Types of Live Excel Sessions	292
User Actions in Live Excel Files	294
Editing Properties in a Live Excel Session With the Architect/Requirements Client	295
Editing Properties in an Independent Live Excel Session	302
Creating Objects in Live Excel.....	305
Creating Trace Links in Live Excel	306
Disconnecting a Live Excel Spreadsheet.....	307
Attaching a Spreadsheet to an Object	308
Storing Property Values Through Reference Links	309
Reference Link Concepts.....	309
Creating a Plain Text Reference Link.....	318
Creating a Full Content Reference Link	319
Viewing Reference Links	321
Deleting Reference Links	323
Chapter 10: Working With Versions	325
Overview of Versions and Variants.....	325
Enabling Versions for a Project	326
Filtering Versions by Effectivity Rule	328
Freezing Objects	329
Unfreezing Objects	330
Creating Versions.....	331
Creating Variants	332
Viewing a Version Tree	333
Deleting a Version or a Variant	334
Creating a Baseline	335
Viewing a Baseline	337
Chapter 11: Using the Search Module	339

Overview of the Search Module	339
Activating the Search Module.....	340
Search Results	341
Search Results Dialog Window	342
Saving, Recalling, and Running Searches and Reports	344
Exporting a Report to Microsoft Office Excel.....	347
Exporting a Report to Microsoft Office Word.....	349
Exporting a Report to Microsoft Office Visio	350
Order of Search Results	352
Basic Search View	353
Intermediate Search View.....	358
Advanced Search View	362
Switching to the Advanced Search View.....	364
Basic Pane.....	364
Query View Pane.....	364
Modify Pane.....	365
Query Edit Pane.....	366
Constructing Advanced Queries	367
Advanced Query Statements.....	370
Chapter 12: Using the Change Management Package	379
Overview of Change Management.....	379
Submitting Changes for Approval	380
Approving or Rejecting a Change.....	381
Modifying a Change Approval Object.....	383
Viewing a Change Approval Object	384
Viewing a Change Log	386
Appendix A: Glossary.....	389
Appendix B: System-Defined Properties in the Systems Engineering and Requirements Management Module.....	395
Overview of System-Defined Properties	395
Table of System-Defined Properties	396
Appendix C: Live Office Interface Frequently Asked Questions.....	403
General.....	403
What are the prerequisites to use the Live Office Interface?	403
Can I install Microsoft Office 2013, and Office 2016 on the same machine?.....	403
What are the Architect/Requirements default addins?.....	403
How do I disable Architect/Requirements addins in Microsoft Office application?	404
What are the registry entries created by Architect/Requirements for the Live Office Interface?.....	404
What should I do if the Architect/Requirements Live Office Interface does not work on my machine after I install the latest version of the Architect/Requirements client? OR How to fix the communication error message when I export to live Excel from Architect/Requirements?	405
What should be the security settings in the Microsoft Office Excel, Microsoft Office Word, and Microsoft Office Visio in order to use the Architect/Requirements Live Office Interface?.....	405

Where can I find the log file for Live Office Interface?	405
Microsoft Office Word Troubleshooting	406
What do I do if the formatting is lost while exporting requirements to Microsoft Word?	406
How to fix problems Importing Microsoft Word Document?	406
Microsoft Office Excel Troubleshooting	407
What should I do if the live Excel functionality does not work on my system? The diagnostic tool does not suggest any errors. The security and registry settings are correct.	407
How to fix the missing Architect/Requirements toolbar in Microsoft Office Excel? OR How to view the Architect/Requirements menu when I open a saved live Excel workbook? OR What should I do if the Microsoft Office Excel Add-ins do not load after I open a saved live Excel workbook?	407
I made changes in the saved independent live Excel workbook after I connected to the Architect/Requirements server. I cannot see these changes in the Architect/Requirements client. What could be the reason?	408
In what format should I save my live Excel workbook to retain the live behavior after I open the saved live Excel sheet?	408
I edited a requirement (a Microsoft Office Word document) from a live Excel workbook using the Architect/Requirements Open menu in the live Excel sheet. However, the changes are not reflecting in the Architect/Requirements client. What could be the issue?.....	408
Microsoft Office Visio Troubleshooting.....	408
I open a saved live Visio diagram from my local disk, connected to Architect/Requirements and changed the property of object, but I cannot see the property value updated in Architect/Requirements client. What could be the issue?.....	408
What could be the issue if adding a master shape in live Visio diagram does not create type or subtype in Architect/Requirements?	408
What should I do if I cannot open at least one stencil object when I try to create a new live Visio diagram?.....	409
How can I map more than one Architect/Requirements property to a Microsoft Office Visio master shape?.....	409
How can I identify the sequence of the sub shapes in a group shape?.....	409
While creating or opening Visio diagram, some of the shapes appear outside the Visio page. What should I do?.....	409
Troubleshooting for the Architect/Requirements Client.....	410
The client installation of Systems Architect/Requirements Management is failing. What could be the possible cause?.....	410
How do I unregister and register the Architect/Requirements Client and Microsoft Office?.....	410
What should I do if I cannot create a live Visio diagram from the Architect/Requirements client and the Visio Live—>Create Diagram menu is disabled?	411
I am not able to open a Microsoft Office Word document from the Architect/Requirements client. I get several errors related to com.inzoom.comjni.ComJniException.eComError. How I can get rid of those?	411
How to fix the several pop up messages I get related to vtable call when I launch the Architect/Requirements client?.....	411
Where can I find the log file for the Architect/Requirements client?	412
What are the log files that I can use for troubleshooting?	412

Appendix D: Changing the maximum memory available for rich client.....413
Index.....415

Preface

This manual is a user's reference for Teamcenter Systems Architect/Requirements Management 11.1. Systems Architect/Requirements Management belongs to the Siemens PLM Software portfolio of digital product lifecycle management software and services.

Audience

This manual is for project team members who use Systems Architect/Requirements Management (Architect/Requirements) to create, manage, trace, and analyze requirements information throughout the entire life of a product. This manual provides both conceptual information and step-by-step instructions for specific tasks.

This manual assumes that you are familiar with your project and your product development process, that you understand general computer terminology and the Microsoft Windows operating system, and that you have experience with Microsoft Word, Microsoft Excel, Microsoft Internet Explorer, and Microsoft Office Visio.

Conventions

This manual uses the conventions described in the following sections:

Revision Marks

Technical changes are marked by a bar adjacent to the changed text.

Browser and Dialog Window Examples

The examples of browsers and dialog windows in this manual may appear different from those you see on your screen:

- The examples reflect Systems Architect/Requirements Management as initially installed at your site. Your enterprise may customize the browsers and dialog windows such that they appear different from those in the examples.
- The examples reflect individual Systems Architect/Requirements Management modules. If you install additional modules, your dialog windows and browsers reflect the additional modules.
- The examples reflect Systems Architect/Requirements Management installed on a Windows platform.

Names and Values

This manual represents system names, file names, and values in fonts that help you interpret the name or value. For example:

Change or add the parameter to the **initsid.ora** file.

The conventions are:

Bold	Bold font represents unvarying text or numbers within a name or value. Capitalization is as it appears.
<i>Italic</i>	Italic font represents text or numbers that vary within a name or value. The characters in italic text describe the entry. Letters are shown in lowercase, but the varying text may include uppercase letters. In initsid.ora , <i>sid</i> identifies a varying portion of the name (a unique system ID). For example, the name of the file might be: initBlue5.ora
<i>text-text</i>	A hyphen separates two words that describe a single entry.

Submitting Comments

Portions of Teamcenter software are provided by third-party vendors. Special agreements with these vendors require Siemens PLM Software to handle all problem reports concerning the software they provide. Please submit all comments directly to Siemens PLM Software.

Please feel free to share with us your opinion on the usability of this manual, to suggest specific improvements, and to report errors. Mail your comments to:

Siemens PLM Software Technical Communications
5939 Rice Creek Parkway
Shoreview, MN 55126
U.S.A.

To submit an incident report, you can use the Siemens PLM Software GTAC online support tools at the following URL:

http://www.plm.automation.siemens.com/en_us/support/gtac/

Proprietary and Restricted Rights Notice

This software and related documentation are proprietary to Siemens Product Lifecycle Management Software Inc.

© 2018 Siemens Product Lifecycle Management Software Inc. All Rights Reserved.

All trademarks belong to their respective holders.

Chapter 1: Introducing Architect/Requirements

This chapter presents an overview of Architect/Requirements, discussing the basic concepts of its purpose and application.

Insufficient planning and misunderstood expectations affect more than two-thirds of all product development projects.¹ Architect/Requirements solves both problems in the most crucial phase of development, the decision-making stage, when plans and expectations are consolidated as requirements.

Requirements describe the product that the customer will buy. They communicate the customer's specifications to the various disciplines involved in the product's development. To ensure that the finished product meets those specifications, developers follow the requirements throughout the development process. When the product conforms to all requirements, it is ready for delivery, and more importantly, it has the functions and the quality that the customer demands.

Projects that inadequately communicate requirements to developers are prone to run over budget and behind schedule, with late-cycle changes and heroic integration efforts. In turn, these conditions can lead to functional deviations and inferior quality, and ultimately to a product that the customer may reject.

In avoiding such difficulties, two principles are vital. Requirements must be identified at the project's inception, so that problems are revealed and understood before the actual development begins. After identification, requirements must be connected to the product design, and that connection must be maintained through all successive stages—while drawings, parts, and assemblies are developed, tested, and changed.

Incorporating both principles, Architect/Requirements provides for:

- Identification of requirements in the initial stage of product development.
- Early allocation of requirements to development teams and product components.
- Ongoing management of requirements.

¹ Forrester Report, March 2000, in Collaborative Solutions for Product Lifecycle Management brochure (M-1000), published March 2002 by EDS PLM Solutions.

Requirements Identification and Allocation

Architect/Requirements extends customer involvement to all phases of the development process. Through Teamcenter integration, the customer can actively influence the design from requirements identification and allocation to product testing and acceptance.

The following sections describe the requirements identification and allocation process.

Identification

The first step in requirements identification is to gather information for source requirements. Such information may be gathered from telephone conversations, meetings, regulatory agencies, standards organizations, and external documents.

The next step is to analyze that information, looking for issues, ideas, and keywords expressed by the customer. By those guidelines, irrelevant material can be separated and discarded. The remaining information becomes the content for the source requirements.

In Architect/Requirements, content can be added to a new requirement with Microsoft Word. In addition, documents can be imported from Word to create multiple requirements and content simultaneously. For more information, see [Creating a Requirement Object, Entering and Changing Requirement Content in Microsoft Office Word](#), and [Creating Requirements and Content by Import From Microsoft Office Word](#) in chapter 5, *Managing Requirements*.

Then, additional requirements are derived from the source requirements. This step involves creating more detailed requirements that can be traced back to the originating sources. Trace links, from source requirements to derived requirements to other derived requirements, establish the order of precedence among requirements. For more information, see [Overview of Trace Links](#) in chapter 7, *Showing Object Relationships With Trace Links*.

Also in the identification process, requirements are organized according to their intended implementation. This organization partitions the requirements around designs, so that records can be generated for reference in comparing changes with previous structures. For more information, see chapter 4, [Maintaining a Project](#).

To record the organizational structure:

- Table views in Architect/Requirements can be exported to Microsoft Excel. For more information, see [Exporting Objects to Microsoft Office Excel](#) in chapter 4, *Maintaining a Project*.
- Requirements can be exported to Microsoft Word to capture their content and their structure within a folder. For more information, see [Exporting Objects to Microsoft Office Word](#) in chapter 4, *Maintaining a Project*.

Allocation

Following identification, requirements are ready for allocation among the development teams and product components. Allocation connects requirements to the product design through trace links to objects in Teamcenter products:

- In Architect/Requirements, requirements link to one another within a project.
- In Engineering Process Management, requirements link to diagrams, parts, and assemblies.
- In Teamcenter Enterprise, requirements link to product modules.
- In Project, requirements link to tasks in a program schedule.

For more information, see [Creating Trace Links](#) and [Linking to an Object in Another Teamcenter Product](#) in chapter 7, *Showing Object Relationships With Trace Links*.

The trace links to and from each Teamcenter product show how requirements affect the individual objects, the overall development in that discipline, and the dependencies among the disciplines. These complementary relationships increase efficiency by:

- Reducing latency of design and decisions.
- Minimizing the need for changes late in the development process.
- Preventing repetition of past mistakes.
- Enabling process measurement and improvement.
- Promoting accurate design documentation.
- Notifying downstream disciplines of decisions based on the requirements, immediately and in context.

Teamcenter integration delivers maximum effectiveness in the development process. Throughout the process, the customer's influence ensures that problems are discovered before the customer receives the product, and that the finished product is one which the customer will buy. Because requirements are communicated without delay, development disciplines can adjust to design changes as they occur. With requirements actuating development decisions, organizations that use Teamcenter products can respond rapidly to fluctuations in market demand.

Requirements Management

To manage requirements, an organization may use spreadsheets, linked documents, custom databases, and document-oriented tracing tools. Typical problems in such methods are that requirements are isolated on individual computers with limited access, stored in databases with little resemblance to the product structure, or maintained through complicated user interfaces with significant learning curves.

Architect/Requirements simplifies requirement development and access, and substantially reduces the learning curve. Requirements are developed with Microsoft Office Word, Microsoft Office Excel, and Microsoft Office Visio, tools that are readily available in most organizations. For authorized users, requirements are always accessible in a central database that matches the product structure. In addition, the Architect/Requirements user interface is modeled on Microsoft Windows Explorer, with parallel concepts and user actions. Team members who are familiar with Word, Excel, Visio, and Windows Explorer can quickly become productive with Architect/Requirements.

Table 1-1 relates key concepts of Windows Explorer to Architect/Requirements.

Table 1-1. Microsoft Windows Explorer Model in Architect/Requirements

Windows Explorer	Architect/Requirements
Disk volumes Physical data storage devices to hold folders and files	Projects Logical data boundaries to hold folders, requirements, building blocks, and groups
Folders Organization of files within a disk volume	Folders Organization of requirements, building blocks, and groups within a project
Files Units of content managed within folders	Requirements, building blocks, and groups Units of content managed within folders
Permissions Rules for user access to disk volumes, folders, and files	Security profile Rules for user access to projects, folders, requirements, building blocks, and groups
Properties Named values defining the characteristics of a disk volume, folder, or file	Properties Named values defining the characteristics of a project, folder, requirement, building block, or group
Search Find objects that meet specified criteria, such as name, text content, type, or other properties	Search Find objects that meet specified criteria, such as name, text content, type, or other properties
Create Create new folders and files	Create Create new folders, requirements, building blocks, and groups

Table 1-1. Microsoft Windows Explorer Model in Architect/Requirements

Windows Explorer	Architect/Requirements
Cut, Copy, Paste Move or duplicate folders and files	Cut, Copy, Paste Move or duplicate folders, requirements, building blocks, and groups
Undo Reverse previous actions	Undo Reverse previous actions

Though it supports many aspects of systems engineering, Architect/Requirements is mainly a requirements management tool. Therefore, to most users the principal object is the requirement, one of the built-in object types in Architect/Requirements. For each object type, a distinct set of properties defines all objects of that type.

Object Types

Projects are superior to all other objects in Architect/Requirements and represent the highest level of organization. Each project prescribes a boundary for user access and for customization of object types and properties. Furthermore, each project defines a hierarchy in which the other types of objects reside.

In the Systems Engineering and Requirements Management module, those object types are the following:

- *Folder*

In a project hierarchy, the primary level is reserved for folders. As folders contain files and other folders in Microsoft Windows, folders contain requirements, building blocks, groups, and other folders in Architect/Requirements. For more information, see chapter 4, [Maintaining a Project](#).

- *Requirement*

Requirements are created and organized in folders. Within each folder, multiple levels of organization allow flexibility in structuring requirements. For example, all requirements in a folder can reside at the top level. Or, they can be organized in a hierarchy of parent, child, and sibling requirements. Each requirement is a separately managed object, with specific properties and access control. Content is created and edited in Microsoft Word. For more information, see chapter 5, [Managing Requirements](#).

- *Building block*

Building blocks are created and organized in folders. As requirements can be structured within a folder, building blocks can be organized in multiple levels of parents, children, and siblings, or they can all reside at the top level. Building blocks graphically illustrate the hierarchical relationships of elements in a product or system. For example, such hierarchies may reflect design elements in a product, tasks in a work breakdown structure, or job functions in an organizational chart. Diagrams attached to building blocks are created and edited in live Visio, the Architect/Requirements interface with Microsoft Office Visio. For more information, see chapter 6, [Constructing System Views With Building Blocks and Diagrams](#).

- *Group*

A group consists of references to existing objects within a Architect/Requirements project. Each reference associates one object as a *member* of the group. Member objects remain in their various locations and can be maintained from the group without the need to switch from one location to another.

Members can be folders, requirements, building blocks, notes, and other groups. A given object can belong to any number of groups, and can be removed from one or more groups at any time. For more information, see [Using Groups to Maintain Objects](#) in chapter 4, *Maintaining a Project*.

- *Note*

Notes are well suited for recording information that relates only to certain objects, rather than to all objects of a type. For example, such information may convey the rationale for a decision, the minutes of a meeting, or an informal discussion about a particular requirement.

Notes can be attached to folders, requirements, building blocks, and groups. Content is created and edited in Microsoft Word. For more information, see chapter 8, [Recording Supplementary Information With Notes](#).

- *Diagram*

Diagram objects contain diagrams that are created and edited through live Visio, the Architect/Requirements interface with Microsoft Office Visio. Each live Visio diagram is interactively synchronized with the Architect/Requirements database, and is associated with a special stencil that contains shapes representing the Architect/Requirements object types.

Objects can be created, modified, and deleted in the database by adding, modifying, and deleting the corresponding shapes in the diagram. The diagram is updated dynamically when members of the diagram owner are created, modified, and deleted in the database. Diagram objects are typically attached to building blocks, but can be attached also to folders, requirements, and groups. For more information, see chapter 6, [Constructing System Views With Building Blocks and Diagrams](#).

- *Spreadsheet*

To store equations in the Architect/Requirements database, users can create spreadsheet objects containing worksheets from existing Microsoft Excel files. Equations in worksheet cells are preserved when the spreadsheet is opened, rather than being overwritten by updated property values from the database.

Spreadsheets can be attached to folders, requirements, building blocks, and groups. Equations and other content can be edited in Excel. For more information, see [Attaching a Spreadsheet to an Object](#) in chapter 9, *Working With Object Properties*.

- *Trace link*

A trace link establishes a directional relationship between two objects. Each trace link indicates which object precedes, or defines, the other in the relationship. The defining and complying objects can reside within the same Architect/Requirements project, or the objects can reside in different projects.

Trace links can be created between folders, requirements, building blocks, and groups. Trace links also can be created from objects in Architect/Requirements to objects in other Teamcenter products. For more information, see chapter 7, [Showing Object Relationships With Trace Links](#).

- *Connection*

This object type is used in diagrams created through live Visio, the Architect/Requirements interface with Microsoft Office Visio. For more information, see [Live Visio Diagrams](#) in chapter 6, *Constructing System Views With Building Blocks and Diagrams*.

Object Properties

For each object type, a distinct set of properties governs the nature and behavior of all objects of that type. Architect/Requirements assigns predefined properties automatically to every object. In addition, a project administrator can define custom properties to fit the organization's business needs.

Each property's value is defined as one of the following:

- *Choice*

The value is selected from a predefined list of choices. A list can be *single-choice*, allowing only one selection, or *multiple-choice*, allowing any number and combination of selections.

- *Date*

The value is a calendar date and can be defined to include the time of day.

- *Numeric*

The value is a number and can be defined to allow floating-point formats.

- *Text*

The value is any string of plain text.

For more information, see chapter 9, [Working With Object Properties](#) and appendix B, [System-Defined Properties in the Systems Engineering and Requirements Management Module](#).

Project Alignment

Each site that uses Architect/Requirements can align its projects with its own functional or organizational entities according to its needs. For example, a site might use only one project for all development work. Or, a site might use multiple projects, with each project relating to one of the following:

- Enterprise organizational boundaries: a division, a department, or a work team.
- Product or product line activities that cut across many organizational or functional groups.
- Individual contracts or activities commonly referred to in the local environment as projects.
- Major subsystems of a product.

Although multiple projects allow greater flexibility, each site should evaluate the trade-offs, and then develop a plan for using projects in Architect/Requirements. In defining the scope of each project, the plan should address issues such as the following:

- Which users can modify the information in the project, and which can only view the information.

Users can be granted project access privileges at several broad levels, with corresponding restrictions. In addition, user permissions can be imposed for individual objects through security profiles. Each user can be granted access to any number of projects, with different privileges and permissions for each project.

- Customization of object types and user-defined properties within the project.

At project scope, an Architect/Requirements project administrator defines custom object properties, property values, and object subtypes. Those properties, values, and subtypes apply only within a single project. Therefore, each project can be customized for a particular purpose, process, or product.

Chapter 2: Installing the Architect/Requirements Client with Office Integration

This chapter describes the components, prerequisites, and the installation of the Architect/Requirements Client with Office Integration. It also describes the usage of the Live Office Interface.

Introduction

The Architect/Requirements Client with Office Integration installation wizard resides on the Architect/Requirements server. You use Microsoft Internet Explorer or Mozilla Firefox to log in to the server and run the wizard. When the installation is complete, the client starts automatically.

Architect/Requirements Components

The Architect/Requirements software consists of the following components:

- The database server component runs the Versant service for the Architect/Requirements database. Your enterprise administrator installs this component.
- The Web server component provides the business logic for Architect/Requirements. Your enterprise administrator installs this component.
- The client component presents the Architect/Requirements user interface on a local computer. You install the client on your computer. For more information, see [Installing the Architect/Requirements Client with Office Integration](#), later in this chapter.

The client component includes the installation of the Architect/Requirements live interface with Microsoft Office. For more information, see [Using the Live Office Interface](#), later in this chapter.

Prerequisites for Installing the Architect/Requirements Client with Office Integration

You must install the following prerequisites before you install the Architect/Requirements Client:

Microsoft .NET 4.6 for Windows 7

The Architect/Requirements client requires .NET 4.0 CLR runtime support. .NET 4.0 CLR runtime is provided with Windows 10, but you need to add a NET 4.6 package for Windows 7 32-bit or 64-bit installation. You can download and install the Microsoft .NET Framework 4.6 package from the following URL:

<https://www.microsoft.com/en-us/download/details.aspx?id=48137>

Primary Interop Assemblies Redistributable

If .NET 4.6 is not included and Microsoft Office is installed before installing .NET, you must install a Primary Interop Assemblies Redistributable package from Microsoft. The **Microsoft Office 2010 Primary Interop Assemblies Redistributable** supports Office 2013, and Office 2016. For more information on installing the Primary Interop Assemblies, see:

[http://msdn.microsoft.com/en-us/library/kh3965hw\(v=vs.100\).aspx](http://msdn.microsoft.com/en-us/library/kh3965hw(v=vs.100).aspx)

Oracle 32-bit JRE

Due to changes in Oracle's JRE redistribution policy, Siemens PLM Software does not redistribute JREs. You can obtain the 32-bit JRE installer from your IT department or download it directly from the Oracle Web site:

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

For information about version of Oracle JRE certified for your platform, see the Siemens PLM Software Certification Database:

http://www.plm.automation.siemens.com/en_us/support/gtac/certifications.shtml and see the **Teamcenter Systems Engineering Software Certifications** section.



You need to install the JRE again even if you have the 32-bit JRE 1.8.0_161 installed.

To install the JRE

1. Uninstall all previously installed JREs.
2. Install 32-bit Java SE 1.8.0_161 .
3. Ensure that the 32-bit Java SE 1.8.0_161 **bin** directory is inserted at the beginning of the user's **PATH** variable.

Other prerequisites

Before you install the Architect/Requirements Client with Office Integration:

- You must know the URL of your site's home page for Architect/Requirements.
- You must be a registered Architect/Requirements user and must know your user name and password.
- You must have Microsoft Internet Explorer or Mozilla Firefox installed.

For information about version of Internet Explorer or Mozilla Firefox, see the Siemens PLM Software Certification Database:

http://www.plm.automation.siemens.com/en_us/support/gtac/certifications.shtml and see the **Teamcenter Systems Engineering Software Certifications** section.

-

You require Power User or Administrator privileges for installing the Architect/Requirements Client with Office Integration.

After the client installation is performed by a user with Power User or Administrator privileges, a different user without these privileges can launch the Architect/Requirements Client and work with it to the extent the user's privileges allow.

-

All the Windows registry entries and the DLL files for the Client with Office Integration are registered in the **HKEY_LOCAL_MACHINE** root keys. Hence, you must have write permissions for these root keys.

- You must remove any previous release version of the live Office interface. Multiple versions of the live Office interfaces on a single machine are not recommended. Every time you start the Architect/Requirements client, the live Office interface with the same version as the Architect/Requirements client is registered for use.



If you have questions about the home page or your user name or password, consult your Architect/Requirements system administrator or project administrator.

You may also need your proxy server settings from Microsoft Internet Explorer. These settings are available through the **Connections** tab in the Internet Options dialog window. Click the **LAN Settings** button to display the Local Area Network (LAN) Settings dialog window, where the settings are displayed under **Proxy server**.

Limitations of Architect/Requirements Support for Microsoft Office Integration

Architect/Requirements provides limited support for Microsoft Office 2013 and Office 2016.

Supporting Multiple Installations of Microsoft Office

Architect/Requirements does not support the installation of multiple versions of Microsoft Office, although it is allowed by Microsoft in certain cases. To prevent stability issues in the Architect/Requirements Live Office interface, avoid installing multiple versions of Office. When more than one version is installed, there is no reliable way for the Architect/Requirements client to launch a specific version. Conflicts occur with the different versions of Office add-ins.

The Architect/Requirements client integrates with Office Excel, Word, and Visio. You must have the same version of Office Excel, Word, and Visio. You can have Office 2013 or Office 2016 installed. However, you cannot have a mixed installation of different versions of Office products. For example, an installation of Excel 2013 and Visio 2016 on the same computer is not supported. To install different versions of Office products, use a virtual machine.

Constraints for Supporting Microsoft Office Integration

Architect/Requirements provides integration with versions 2013 and 2016 of the Microsoft Office Professional editions subject to the following constraints:

- **Supported Microsoft Office Packages**

Product	Professional	Office 365
Office	32-bit Office Professional Plus	32-bit Office 365 ProPlus
Visio	32-bit Visio Professional	32-bit Visio Pro for Office 365

- **Limitations of Support for Microsoft Office Packages**

All Packages	Office 365 Packages
Only versions 2013 and 2016 are supported.	Only the PC device is supported.
64-bit formats are not supported.	Offline installation is supported for all Office 365 packages included above.
Home and student editions are not supported	Over-the-web installation is supported for Office 365 ProPlus version 2016, but not for version 2013.

- **Limitations of Support for Microsoft Office features**

Microsoft Office and Visio include features that are added, removed, and changed between different versions. While Architect/Requirements integration functions are certified to work with the packages listed above, it is possible that it may not work with all combinations of Office and/or Visio features. Siemens PLM Software recommends that you verify business solutions on a test environment before using them in a production environment.

Feature changes for Microsoft Office and Visio

Release	Office Professional Plus	Visio Professional
2013	* Changes in Office 2013 at https://technet.microsoft.com/en-us/library/cc178954(v=office.15).aspx	** What's New in Visio 2013 at https://support.office.com/en-us/article/What-s-new-in-Visio-11656535-2345-491d-bb9a-bef0141180f7
2016	* What's new and improved in Office 2016 at https://support.office.com/en-us/article/What-s-new-and-improved-in-Office-2016-29d7e38e-ef06-4d9c-a476-03d896928b2f	** What's New in Visio 2016 at https://support.office.com/en-us/article/What-s-new-in-Visio-2016-798f4f39-2833-486b-9ae9-55162672102e

* - Applies also to Office 365 ProPlus (Excel and Word)

** - Applies also to Visio Pro for Office 365

Windows Client Hardware

The minimum hardware requirement for running Office 2013 or Office 2016 is significantly greater than the minimum hardware requirement for Office 2010. If you are upgrading to Architect/Requirements 11.1 and also upgrading to Office 2013 or Office 2016, you may need additional hardware. If you plan to add Office 2013 or Office 2016 to your existing Windows 7 or Windows 10 system, ask your IT department to verify if the hardware is sufficient.

System Requirements references for Microsoft Office and Visio

Release	Office Professional Plus	Visio Professional
2013	* https://technet.microsoft.com/en-us/library/ee624351(v=office.15).aspx	** https://products.office.com/en-us/visio/microsoft-visio-faq-diagram-software?legRedirect=true&CorrelationId=d383e71d-bc7d-4ea1-af39-28bf8a2ecad3# , search for system requirements for Visio and click the link.
2016	* Open https://products.office.com/en-us/office-system-requirements , search for Office Professional Plus 2016 and click the link.	** Open https://products.office.com/en-us/office-system-requirements , search for Visio 2016 and click the link.

* - Applies also to Office 365 ProPlus (Excel and Word)

** - Applies also to Visio Pro for Office 365

New Visio 2013 and Visio 2016 Stencil and Diagram Formats Not Supported

Architect/Requirements 11.1 supports integration with Visio 2013 and Visio 2016, however, the new formats for diagrams (**VSDX**) and stencils (**VSSX**) are not supported.

- Offline copies of Visio diagrams integrated with Architect/Requirements must remain in **VSD** format. To avoid unnecessary re-work, use Visio's **Save As** feature to save offline copies of Visio diagrams in the **VSD** format only. There is currently no plan to provide support for Visio's **VSDX** format with Architect/Requirements.
- The Architect/Requirements stencil import capability continues to support the **VSS** format. You must not import stencils formatted in the new **VSSX** format into Architect/Requirements. To import a stencil formatted in the new **VSSX** format, save it as a **VSS** file.
- When you save a Visio diagram for offline use that you need to synchronize with Architect/Requirements later, ensure that the files are saved in the **VSD** format.

To save the diagram outside Architect/Requirements on a local or shared drive:

- . Select the diagram in the Architect/Requirements rich client.
- . Open the diagram double clicking it.
You can also click the right mouse button and select **Open** from the context menu.
- . Click **File** in Visio window.
- . Click **Save As** to display the folder selection dialog.
- . Select a folder or click **Browse**.
Visio displays the **Save As** dialog.
- . Type the file name in the **File name** box.
- . In the **Save As** dialog, ensure that **Visio 2003-2010 Drawing (*.vsd)** is selected in the **Save as** type drop down menu.
- . Click **Save**.
- . To open the diagram and synchronize it with Architect/Requirements, click **File** and select **Visio Live**→**Open** to display the **Open** dialog.
- When you select the Visio stencils to include a Live Visio diagram, you must select stencils stored in the **VSS** format only.

To add stencils to your Live Visio diagram:

- . When creating a Live Visio diagram, select any stencil included in the **Diagram inputs** dialog.
These stencils are always in the **VSS** format. These are stored in the Architect/Requirements database, and are maintained by your Architect/Requirements Project Administrator.
- . When the Live Visio diagram is open for editing, it is possible to add shapes from Visio stencils outside Architect/Requirements. These stencils must be in **VSS** format. For example, if you select **Open Stencil** from the **STENCIL** ribbon, you must select files with a ***.vss** extension only.

For more information on the use of Visio stencils and diagrams, see [Constructing System Views With Building Blocks and Diagrams](#).

Installing the Architect/Requirements Client with Office Integration

Before you install the Architect/Requirements client with Office integration, see the [Prerequisites for Installing the Architect/Requirements Client with Office Integration](#).



If you are installing the Architect/Requirements client with Office integration as a part of upgrade process from a previous version, you must uninstall the previous version of Architect/Requirements client installed on your computer.

For information on uninstalling the Architect/Requirements client, see [Uninstalling the Architect/Requirements Client](#).



Because your enterprise can customize Architect/Requirements, some steps in this procedure may vary from the steps for your particular site. If you have questions about installing the client, consult your system administrator.



- The 32-bit version of the Java Runtime Environment (JRE) must be installed. If necessary, you can install the required Java software in this procedure.

The **JRE.Version** configuration option's default value includes only the 1.8 version. However, sites upgrading from previous Architect/Requirements releases should uninstall all previously installed JREs from their client systems, install JRE 1.8, and edit their **JRE.Version** option to remove all 1.6 and 1.7 versions and add at least one 1.8 version.

For information about version of Java Runtime Environment (JRE) and Java Plug-in supported, see the Siemens PLM Software Certification Database:

http://www.plm.automation.siemens.com/en_us/support/gtac/certifications.shtml and see the **Teamcenter Systems Engineering Software Certifications** section.

- The Architect/Requirements live interface with the Microsoft Office is installed as part of the Architect/Requirements Client with Office Integration installation. For more information, see [Prerequisites for Using the Live Office Interface](#), later in this chapter.
1. Open the Architect/Requirements home page in a browser window and click the **Launch Teamcenter systems engineering** link.

For information on the supported browsers, open the following URL:

http://www.plm.automation.siemens.com/en_us/support/gtac/certifications.shtml and see the **Teamcenter Systems Engineering Software Certifications** section.



If you are installing Architect/Requirements on Windows 10, launch the browser as the administrator.

To launch the browser as administrator, right click the browser icon in the directory that the browser is installed and select **Run as administrator**.

The browser may prompt you to allow execution of the **Java™ Web Launcher** on your computer. You must allow this in order to proceed with the launch:

- Microsoft Internet Explorer will prompt with a dialog asking “Do you want to allow this website to open a program on your computer?”. The user should click the “Allow” button to proceed. un-checks the box labeled “Always ask before opening this type of address”.
- Mozilla Firefox will prompt with a “Launch Application” dialog. The user should click the “Open Link” button to proceed. The choice can be remembered if the user checks the box labelled “Remember my choice”.

Depending on whether Security Services is enabled on the Architect/Requirements server, one of two login pages is displayed:

- The Teamcenter Systems Engineering and Requirements Management login page is displayed if Security Services is not enabled. Enter your Architect/Requirements user name and password, select a language, and click **Log In**.
- The Teamcenter Login page is displayed if Security Services is enabled. Enter your Security Services user name and password, select a language, and click **Log In**.



If you do not have Power User or Administrator privileges, the client installation fails and displays a message to contact your local IT department. If you obtain the required privileges, the client installation procedure should be followed from the beginning.

Once logged in, Java may prompt you as to whether you want to run the **Teamcenter Systems Engineering** application. Select “Run” to proceed.

A confirmation message prompts you to install the client.

2. Click **Yes** to display the Proxy Server Settings dialog window, and then do one of the following:
 - To connect directly to the Architect/Requirements server, click **OK**.
 - To connect to the Architect/Requirements server through a proxy server, fill in the **Host address** and **Port** fields and click **OK**. If necessary, check the **Use proxy authorization** check box and enter your proxy server information.



If your browser settings specify an automatic configuration script, you must add configuration information to the Java control panel. This step is necessary if the **Use automatic configuration script** check box is checked in the Local Area Network (LAN) Settings dialog window.

The configuration information is in the file whose location is shown in the **Address** field. From this file, you need your proxy server's HTTP address and server port number.

Opening this file is an advanced operation. If you have questions about the file or the information, consult your system administrator.

After obtaining the information, do the following:

- In the Windows Control Panel, open the Java Control Panel, and then click the **Proxies** tab.
- Clear the **Use browser settings** check box, fill in the **Proxy Address** and **Port** fields for **HTTP**, and then click **Apply**.

3. In the InstallAnywhere wizard, click **Next** to display a window showing the installation directory path, and then do one of the following:
 - To change the installation drive or folder, do one of the following:
 - Enter the new path directly in the **Directory Name** field.
 - Click **Browse** to display the Select a Directory dialog window, navigate to and select the new installation drive or folder, and then click **Open** to enter the new path in the **Directory Name** field.
 - For installation in the default drive and folder, leave the **Directory Name** field unchanged.
4. Click **Next** to display a window with summary information for review.
5. Click **Next** to start the installation.

The wizard displays a progress indicator, followed by a window stating that the wizard has successfully installed the client.

6. Click **Finish** to exit the wizard.
7. The Architect/Requirements main window is displayed

For more information, see [Elements of the Architect/Requirements Main Window](#), in chapter *Using the Architect/Requirements Main Window*.



- If you logged in through Security Services (step 1), and if other Teamcenter applications are installed, you can launch those applications without logging in to each one individually.
- The wizard installs a Microsoft Word template for parsing requirements by keywords in an import document. For more information, see [Creating Requirements and Content by Import From Microsoft Office Word](#) in chapter 5, *Managing Requirements*.

Troubleshooting the ICE_JNIRegistry.DLL Java console error

If the **Web Application Configuration** variable **RegeditEnabled** is set to **false** and you install or launch the Architect/Requirements client, the following error is logged in the Java console.

```
ERROR You have not installed the DLL named 'ICE_JNIRegistry.DLL'.
no ICE_JNIRegistry in java.library.path
```

To resolve the error, add the *TCSE_CLIENT_LOC* environment variable to the computer running the Architect/Requirements client. Append the value of the *TCSE_CLIENT_LOC* to the *Path* variable.

To add the TCSE_CLIENT_LOC variable and edit the Path:

1. Open the Windows **Control Panel** and double click the **System** icon.
2. Click **Advanced system settings**.
3. On the **Advanced** tab, click **Environment Variables**.
4. In the **User variables** section, click **New**.
5. Type *TCSE_CLIENT_LOC* as the **Variable name**.

6. Type the Architect/Requirements client install location as the **Variable value**;
for example, **C:\Program Files\SiemensPLM\Teamcenter\SystemEngineering\Release_11.1**
7. Select the **Path** variable and click **Edit**.
You need local administrator privileges to edit the *Path* variable if it is present in the **System variables** section.
8. Add the following text at the end of the existing path:
;%TCSE_CLIENT_LOC%
9. Click **OK** three times to apply the changes.

Using the Live Office Interface

The live Office interface enables you to create, edit, view, and manipulate objects in the Architect/Requirements through certain Microsoft Office products. The interface consists of Architect/Requirements add-ins and templates for:

- Microsoft Office Excel. For more information, see [Using the Live Excel Interface](#) in the chapter 9, *Working With Object Properties*.
- Microsoft Office Visio. For more information, see [Live Visio Diagrams](#) in the chapter 6, *Constructing System Views With Building Blocks and Diagrams*.
- Microsoft Office Word.



The live Office interface is installed as part of the Architect/Requirements Client with Office Integration installation. For more information, see [Architect/Requirements Components](#) earlier in this chapter.

For more information about customizing the live interface with Microsoft Office, see the *Systems Architect/Requirements Management Project Administrator's Manual*.

For information on troubleshooting the live Office interface, see [Using the Live Office Diagnosis](#) and the appendix C, [Live Office Interface Frequently Asked Questions](#).

Prerequisites for Using the Live Office Interface

For the successful installation of the Architect/Requirements live Office interface, the following must be installed on your computer:

- Microsoft Internet Explorer, or Mozilla Firefox.

For information about version of Internet Explorer or Mozilla Firefox, see the Siemens PLM Software Certification Database:

http://www.plm.automation.siemens.com/en_us/support/gtac/certifications.shtml and see the **Teamcenter Systems Engineering Software Certifications** section.

- Microsoft Office.

For information about version of Microsoft Office, see the Siemens PLM Software Certification Database:

http://www.plm.automation.siemens.com/en_us/support/gtac/certifications.shtml and see the **Teamcenter Systems Engineering Software Certifications** section.



- o Only the **.xlsm** or **.xlsx** file type supports the full capability of live Excel. Siemens PLM Software recommends that you assign the **.xlsm** or **.xlsx** file type to all live Excel files that you save outside the Architect/Requirements database, for example, on a local drive.
 - o If you save live Excel files as the **.mhtml** file type, they become static and lose the live capability permanently.
- Microsoft Office Visio, if you intend to use the live Visio capability in the live Office interface.

For information about version of Microsoft Office Visio, see the Siemens PLM Software Certification Database:

http://www.plm.automation.siemens.com/en_us/support/gtac/certifications.shtml and see the **Teamcenter Systems Engineering Software Certifications** section.

- Microsoft .NET Framework.

For information about version of .NET Framework, see the Siemens PLM Software Certification Database:

http://www.plm.automation.siemens.com/en_us/support/gtac/certifications.shtml and see the **Teamcenter Systems Engineering Software Certifications** section.

64-bit .NET Framework is required for running Architect/Requirements Office Interface on a 64-bit platform.

-

To work with a non-US locale for regional settings with an English version of Microsoft Office, the Multilingual User Interface (MUI) Pack for that language or locale should be installed for Microsoft Office. The MUI Pack provides features that allow you to change the language of your Office user interface and online Help.

The MUI Pack is a prerequisite for the Microsoft Office APIs to work correctly with Architect/Requirements when the US version of Microsoft Office is used in a non-US locale. It should be installed regardless of whether the MUI Pack for another language for the user interface is used or not.

-

The MHTML libraries are required for the functioning of Architect/Requirements because the client interacts with Microsoft Office in MHTML format. The MHTML libraries are packaged with Outlook Express in Microsoft Windows, and with Windows Mail in Microsoft Windows Vista.

Without the MHTML libraries, the Architect/Requirements integration with Microsoft Office fails. The content in the preview pane displays MHTML markup tags.



When you open a locally saved copy of your Live Excel workbook or Live Visio diagram, Architect/Requirements asks you if you want to connect to the server and launches the default browser (Internet Explorer or Mozilla Firefox) from which you can log in to Architect/Requirements. However, if you close the browser window without logging in, in case of Internet Explorer, a message stating that you are not connected to the server is displayed immediately. In case of Mozilla Firefox, this message is displayed after a considerable delay giving the impression that the Live Excel or Live Visio application is hung.

Post Installation Procedures for Using the Live Office Interface

After completing the live Office installation:

- To use the live Visio capability of the live Office interface, you must enable .NET Framework programmability support for Microsoft Office Visio. For more information, see [Enabling .NET Framework Programmability Support for Microsoft Office Visio](#), later in this chapter.
- You must configure Microsoft Office to work with the live Office interface. For more information, see [Configuring Microsoft Office for the Live Office Interface](#), later in this chapter.

Enabling .NET Framework Programmability Support for Microsoft Office Visio



Microsoft .NET Framework and Microsoft Office Visio must be installed on your computer.

For information about version of Microsoft .NET Framework and Microsoft Office Visio, see the Siemens PLM Software Certification Database:

http://www.plm.automation.siemens.com/en_us/support/gtac/certifications.shtml and see the **Teamcenter Systems Engineering Software Certifications** section.

1. In the Windows Control Panel, double-click **Add/Remove Programs** to display the Add/Remove Programs dialog window.
2. In the **Currently installed programs** pane, select your Microsoft Office Visio installation, and click **Change** to run the Microsoft Office Visio Setup program.
3. In the Maintenance Mode Options window, select **Add or Remove Features**, and then click **Next** to display the Advanced Customization window.
4. With **Microsoft Office** expanded, click the plus sign (+) to the left of **Microsoft Office Visio**.
5. To the left of **.NET Programmability support**, click the down arrow, and then select **Run from my computer** from the pop-up list.



The **.NET Programmability support** option is available if you have installed Microsoft .NET Framework version 2.0 and 3.0 or 3.5.

6. Click **Update** to display a progress indicator, followed by a message stating that Microsoft Office Visio has been updated successfully.

If a message asks for the Microsoft Office Visio CD-ROM or network location, insert the CD-ROM or enter the location. Click **Next** to complete the setup.

Configuring Microsoft Office for the Live Office Interface

You must enable the live Office add-ins and templates to work with Microsoft Office. Also, you must set up .NET options.

Enabling Live Office Add-ins and Templates in Microsoft Office



In the Microsoft Office Trust Center, you can choose a higher security setting for the templates and add-ins.

1. In Microsoft Office Excel, Visio, or Word, click **File**→**Options**.
The Excel Options, Visio Options, or Word Options dialog window is displayed.
2. Select **Trust Center** in the left pane, and then select **Trust Center Settings** in the right pane.
The **Trust Center** dialog box is displayed.
3. Select **Add-ins** in the left pane, and then do one of the following in the right pane:
 - For standard security, clear the **Require Application Add-ins to be signed by Trusted Publisher** check box.
 - For higher security, check the **Require Application Add-ins to be signed by Trusted Publisher** check box.
4. Select **Macro Settings** in the left pane, and then select **Disable all macros except digitally signed macros** in the right pane. The unsigned macros would not work in this case.
5. Click **OK** twice to accept the changes.



When higher security is enabled and you open an existing file in Excel, Visio, or Word, the Message Bar displays a security warning stating that some active content has been disabled. To enable this content and add Siemens PLM Software to your list of trusted publishers, do the following:

1. In the Message Bar, click **Options**.

The Microsoft Office Security Options dialog window is displayed.

2. Under **Macros & ActiveX**, select **Enable this content**.
3. Under **Add-In**, select **Enable all code published by this publisher**.

When you click **OK**, the dialog window closes and the Message Bar is removed.

Setting Up .NET Options in Microsoft Office



Setting up .NET options is not applicable to Microsoft Office 365 ProPlus.

1. In the Windows Control Panel, double-click **Programs and Features** to display the Add/Remove Programs dialog window.
2. In the **Uninstall or change a program** pane, select your Microsoft Office installation, and click **Change** to run the Microsoft Office Setup program.
3. Select **Add or Remove Features**, and then click **Continue**.
4. On the **Installation Options** tab, do the following with **Microsoft Office** expanded:
 - Expand **Microsoft Excel**, and then select the **Run from my computer** option for **.NET Programmability support**.
 - Expand **Microsoft Word**, and then select the **Run from my computer** option for **.NET Programmability support**.

5. Click **Continue**.

The Configuration Progress window is displayed, followed by a message window stating that the configuration is complete.

6. Click **Close**.

If a message asks for the Microsoft Office CD-ROM or network location, insert the CD-ROM or enter the location. Click **Next** to complete the setup.

Using the Live Office Diagnosis

The Architect/Requirements diagnostic tool enables you to verify that your local machine is properly configured to run the Architect/Requirements interfaces with Microsoft Office.

To run the Architect/Requirements diagnostic tool, open the Architect/Requirements home page in Internet Explorer. Then, click the **Administrative Tools** link. On the Administrative Tools page, click the **Diagnostic Tools** link and on the Diagnostic Tools page, click the **Live Office Diagnosis** link.

To select the applications to be diagnosed, you can check the **Diagnose All** check box or the individual check boxes next to the applications. Then, click the **Diagnose** button to generate the diagnostic information for the selected applications. The diagnostic information contains a green checkmark on success or if no problem is found. If any error or problem is found, you can click the **Suggested Fix** button to display the suggested fix in a message box.

You can click the **Generate Report** button to display the diagnostic information in a browser or click the **Email Report** button to E-mail the diagnostic information as an attachment to the recipients.

You can click the question mark at the top right corner of the Architect/Requirements diagnostic tool window to display the related help information, such as the frequently asked questions.

Table 2-1 lists the applications and their corresponding diagnostic information displayed in the Architect/Requirements diagnostic tool.

Table 2-1. Live Office Diagnostic Information

Application	Diagnostic Information
System	<ul style="list-style-type: none">• Logged-in user name and the system name.• Installed operating system version.• System regional language and date format settings.• Installed and current .Net Framework versions.• Values of the TEMP and TMP environment variables.• .Net interoperability support for Microsoft Office.• Registration status of the IzmJniComAx library.• Installed live Office interface version.
Microsoft Office Excel	<ul style="list-style-type: none">• Installation status and version of Microsoft Office Excel.• Installation and registration status of .Net Interoperability support for Microsoft Office Excel.• Registration status and load behavior of the Architect/Requirements Excel add-in and import add-ins.• Security level of Microsoft Office Excel.• Status of the Architect/Requirements live Office interface files required for the working of the Excel add-in.

Table 2-1. Live Office Diagnostic Information

Application	Diagnostic Information
Microsoft Office Word	<ul style="list-style-type: none">● Installation status and version of Microsoft Office Word.● Installation and registration status of .Net Interoperability support for Microsoft Office Word.● Registration status and load behavior of the Architect/Requirements Word add-in.● Security level of Microsoft Office Word.● Status of the Architect/Requirements live Office interface files required for the working of the Word add-in.
Microsoft Office Visio	<ul style="list-style-type: none">● Installation status and version of Microsoft Office Visio.● Installation and registration status of .Net Interoperability support for Microsoft Office Visio.● Registration status and load behavior of the Architect/Requirements Visio add-in.● Security level of Microsoft Office Visio.● Status of Enable Automation for Microsoft Office Visio.● Status of COM add-ins for Microsoft Office Visio.● Status of the Architect/Requirements live Office interface files required for the working of the Microsoft Office Visio add-in. <p> The diagnostic tool does not flag the synchronization issue related to the local cache folders used for Microsoft Office Visio files. The issue is observed when there is a conflict with stencil downloads. As a workaround, delete the local cache folders and try the synchronization again. To remove the local cache, open the following folder in Windows Explorer and delete the contents of the folder:</p> <pre>%APPDATA%\TcRequirements\VisioStencils</pre>

Uninstalling the Architect/Requirements Client

In some situations, you may want to uninstall the Architect/Requirements client from your computer. You must uninstall Architect/Requirements client if you are upgrading to a new version of Architect/Requirements.



Elevated user privileges (power user or administrator) are required for uninstalling the Architect/Requirements client.



The Architect/Requirements client uninstaller can fail if there is a problem in locating a JRE. In such a case, insert the **LAX_VM** argument to explicitly specify the JRE used to run the uninstaller. For example, if your 32-bit JRE is installed in **C:\Program Files (x86)\Java\jre\bin**, and you are uninstalling the Architect/Requirements client version 10.0, the following command uninstalls the Architect/Requirements web application:

```
"Uninstall TcSE Client with Office Integration Release_10" LAX_VM "C:\Program Files (x86)\Java\jre\bin\java"
```

To uninstall Architect/Requirements client

1. Open the Windows **Control Panel**.
2. Open **Programs**→**Programs and Features**→**Uninstall or change a program**.
3. Select **TcSE Client with Office Integration**.



The exact name of the Architect/Requirements depends on the version of Architect/Requirements that you have installed.

4. Click **Uninstall/Change**.
5. Follow the instructions on the uninstall wizard to uninstall the client.

Chapter 3: Using the Architect/Requirements Main Window

This chapter describes the major elements of the Architect/Requirements main window and contains instructions for using those elements.

Starting a New Architect/Requirements Session

If the Architect/Requirements client is installed on your computer, you start a new session by logging in to the Architect/Requirements server through Microsoft Internet Explorer or Mozilla Firefox. For more information, see [Installing the Architect/Requirements Client with Office Integration](#).



Because your enterprise can customize Architect/Requirements, some steps in this procedure may vary from the steps for your particular site. If you have questions about starting a new session, consult your system administrator.



- You must know the URL of your site's home page for Architect/Requirements and your user name and password. If you have questions about logging in, consult your Architect/Requirements system administrator or project administrator.
- The Architect/Requirements client on your computer may be an earlier version. If a new version is available, you can upgrade your present version in this procedure.

1.

In Internet Explorer or Mozilla Firefox, open the home page for Architect/Requirements, and then click the **Launch Teamcenter systems engineering** link.

Depending on whether Security Services is enabled on the Architect/Requirements server, one of two login pages is displayed:

- The Teamcenter 11.1 Systems Engineering and Requirements Management login page is displayed if Security Services is not enabled.
- The Teamcenter Login page is displayed if Security Services is enabled.

2. Do one of the following:

-

On the Teamcenter 11.1 Systems Engineering and Requirements Management login page, enter your Architect/Requirements user name and password, select a language, and click **Log In**.



On the Teamcenter Login page, enter your Security Services user name and password, select a language, and click **Log In**.



If you log in through Security Services, and if other Teamcenter applications are installed, you can launch those applications without logging in to each one individually.



Siemens PLM Software recommends that users do not log in to multiple clients simultaneously using the same login credentials. In Architect/Requirements, a user must use only one client instance at a time.

If two Architect/Requirements clients are used simultaneously, locking errors are displayed on using user specific resources such as the **Recycle bin**. Changing user settings like effectivity affects both the clients.

If the latest version is installed, the Architect/Requirements main window is displayed (see [Elements of the Architect/Requirements Main Window](#)).

If a new version is available, an upgrade message is displayed.



Upgrades provide new and enhanced features and resolutions of previous problems. Siemens PLM Software recommends that you remove the earlier version and install the new one.



To remove the earlier version and install the new one, click **Yes**.

A progress indicator appears while files are downloaded. Then, the uninstaller wizard is displayed.

- Follow the subsequent instructions in the uninstaller, and click **Finish** when the removal is complete.

A progress indicator appears while files are downloaded. Then, the InstallAnywhere wizard for the client is displayed.

- Follow the subsequent instructions in the installer, and click **Finish** to display the Architect/Requirements main window (see [Elements of the Architect/Requirements Main Window](#)).
- To continue using the earlier version, click **No** to display the Architect/Requirements main window (see [Elements of the Architect/Requirements Main Window](#)).



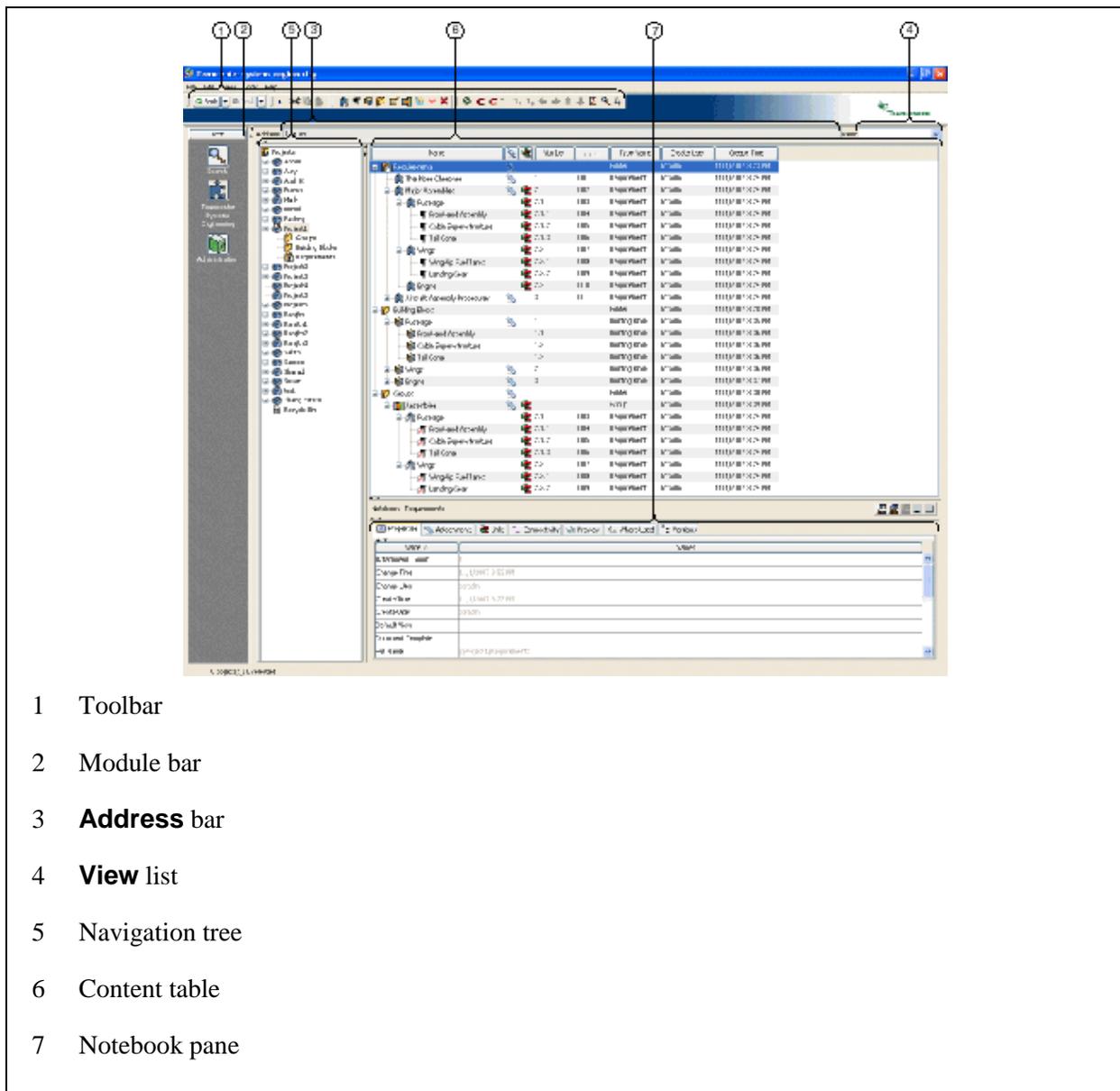
You can also start a new Architect/Requirements session through:

- A Microsoft Office Excel file containing previously exported object data. For more information, see [Exporting Objects to Microsoft Office Excel](#).

- An object URL copied to Microsoft Office Outlook, Microsoft Internet Explorer, or other Microsoft Windows programs that support hyperlinks. For more information, see [Copying Object URLs](#).
- A trace link between an object in another Teamcenter product and an object in Architect/Requirements. For more information, see [Linking to an Object in Another Teamcenter Product](#).
- A file from a previous live Excel session. For more information, see [Using the Live Excel Interface](#).

Elements of the Architect/Requirements Main Window

Figure 3-1 shows the elements of the Architect/Requirements main window.



- 1 Toolbar
- 2 Module bar
- 3 **Address** bar
- 4 **View** list
- 5 Navigation tree
- 6 Content table
- 7 Notebook pane

Figure 3-1. Elements of Architect/Requirements Main Window



While communications are in progress between the Architect/Requirements client and server, the status bar at the bottom of the main window displays:

- A network symbol in the left corner. Further actions are not processed until communications are complete and the network symbol is cleared.
- A progress indicator in the middle. The progress indicator is displayed until all server calls for the user action are complete.

Table 3-1 describes the elements of the Architect/Requirements main window.

Table 3-1. Elements of Architect/Requirements Main Window

Element	Description
Toolbar	Displayed below the pulldown menus at the top of the main window. Appearance and function are similar to a standard toolbar in a Microsoft Windows program. For more information, see Toolbar , later in this chapter.
Module bar	Displayed below the toolbar at the left border of the main window, extending to the bottom. Appearance and function are similar to the Outlook bar in Microsoft Office Outlook. For more information, see Module Bar , later in this chapter.
Address bar	Displayed to the right of the module bar and below the toolbar. Appearance and function are similar to the Address bar in Microsoft Internet Explorer and Microsoft Windows Explorer. For more information, see Address Bar , later in this chapter.
View list	Displayed to the right of the Address bar and below the toolbar. Appearance and function are similar to drop down lists in Microsoft Windows programs. For more information, see Applying a View in the Content Table in chapter 9, <i>Working With Object Properties</i> .
Navigation tree	Displayed to the right of the module bar and below the Address bar, extending to the bottom of the main window. Appearance and function are similar to the hierarchical Folders pane in Microsoft Windows Explorer. For more information, see Navigation Tree , later in this chapter.
Content table	Displayed to the right of the navigation tree in the upper pane. Appearance and function are similar to the columned pane in Microsoft Windows Explorer. For more information, see Content Table , later in this chapter.
Notebook pane	Displayed to the right of the navigation tree in the lower pane. Appearance and function are similar to a tabbed dialog window in a Microsoft Windows program. For more information, see Notebook Pane , later in this chapter.

Customizing the Main Window Display

Except for the toolbar, each major element can be disabled and enabled, by itself or in conjunction with one or more other elements. With this flexibility of display, the main window can be customized according to a specific purpose or your general preference.

For example:

- You can gain more space for the content table by disabling the navigation tree, the notebook pane, or both.

For more information, see [Navigation Tree](#), [Content Table](#), or [Notebook Pane](#), later in this chapter.

- If you work primarily in the Systems Engineering and Requirements Management module, you may want to keep the module bar disabled, occasionally enabling it for access to the Search module or Administration module.

For more information, see [Module Bar](#), later in this chapter.

Your last display settings are preserved in your next session. If an element is disabled when you exit Architect/Requirements, it is not displayed when you start a new session.

Toolbar

The Architect/Requirements main window displays the toolbar below the pulldown menus. Figure 3-2 shows the buttons on the Architect/Requirements toolbar.

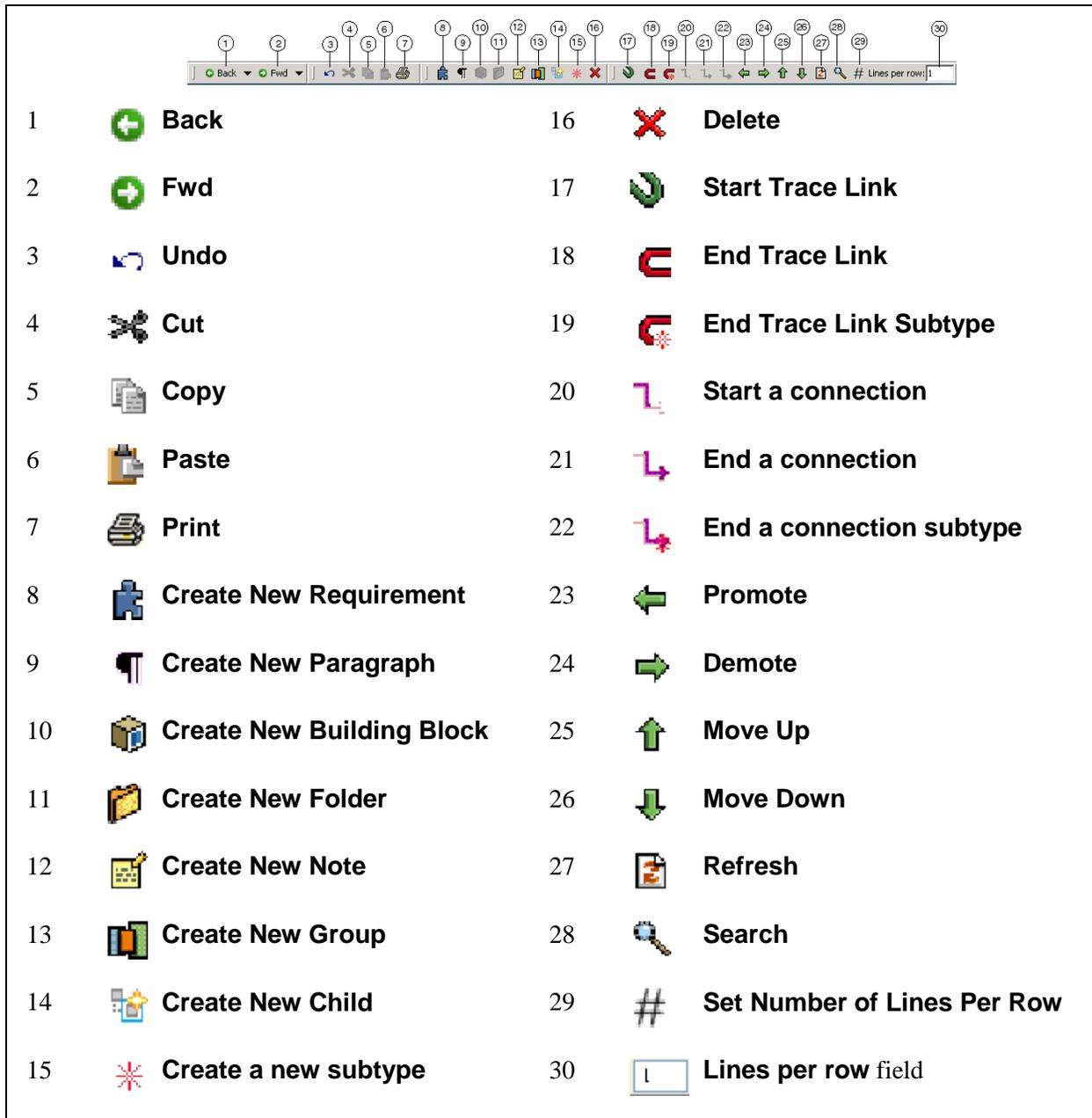


Figure 3-2. Buttons on Architect/Requirements Toolbar



- Toolbar buttons become available or dimmed according to the types of objects that you select.
- You can rest the pointer on a button to see its description in a tooltip.

Each toolbar button relates to an option on a pulldown menu. Table 3-2 describes the buttons on the Architect/Requirements toolbar.

Table 3-2. Buttons on Architect/Requirements Toolbar

Button	Related Menu Option	Action
Back	None	Navigates to the previous selection in the navigation tree. To move back more than one step at a time, click the down arrow to the right to display as many as ten previous navigation tree selections, and then select the project or folder to which you want to navigate. The currently selected project or folder is not displayed in the list.
Fwd	None	Navigates to the subsequent selection in the navigation tree. To move forward more than one step at a time, click the down arrow to the right to display as many as ten subsequent navigation tree selections, and then select the project or folder to which you want to navigate. The currently selected project or folder is not displayed in the list. This button is not enabled until the Back button is used to navigate to a previous selection.
Undo	Edit menu, Undo option	Reverses the previous action.
Cut	Edit menu, Cut option	Places the selected object or objects in the Clipboard.
Copy	Edit menu, Copy option	Places a duplicate of the selected object or objects in the Clipboard.
Paste	Edit menu, Paste option	Inserts the Clipboard contents in the selected location.
Print	File menu, Print option	Print the selected panel. For more information, see Printing a Selected Panel .
Create New Requirement	File menu, New → Requirement options	Creates a new requirement.
Create New Paragraph	File menu, New → Paragraph options	Creates a new paragraph, a system-defined requirement subtype.
Create New Building Block	File menu, New → Building Block options	Creates a new building block.

Table 3-2. Buttons on Architect/Requirements Toolbar

Button	Related Menu Option	Action
Create New Folder	File menu, New → Folder options	Creates a new folder.
Create New Note	File menu, New → Note options	Attaches a new note to the selected object.
Create New Group	File menu, New → Group options	Creates a new group.
Create New Child	File menu, New → Child options	Creates a new child of the selected folder, requirement, or building block.
Create a new subtype	File menu, New → Subtype options	Creates an object of the subtype that you choose in the Select Subtype dialog window, which is displayed when you click this button.
Delete	File menu, Delete option	Sends the selected object or objects to your Architect/Requirements Recycle Bin.
Start Trace Link	Edit menu, Links → Start Trace Link options	Starts a trace link from the selected object or objects.
End Trace Link	Edit menu, Links → End Trace Link options	Ends a trace link to the selected object or objects.
End Trace Link Subtype	Edit menu, Links → End Trace Link Subtype options	Ends a trace link subtype to the selected object or objects.
Start a connection	Edit menu, Connections → Start Connection options.	Starts a connection from the selected object or objects.
End a connection	Edit menu, Connections → End Connection options.	Ends a connection to the selected object or objects.
End a connection subtype	Edit menu, Connections → End Connection Subtype options.	Ends a connection subtype to the selected object or objects.
Promote	Edit menu, Promote option	Promotes the selected object to the next higher level in the hierarchy.
Demote	Edit menu, Demote option	Demotes the selected object to the next lower level in the hierarchy.

Table 3-2. Buttons on Architect/Requirements Toolbar

Button	Related Menu Option	Action
Move Up	Edit menu, Move Up option	Moves the selected object up by one position within its current level.
Move Down	Edit menu, Move Down option	Moves the selected object down by one position within its current level.
Refresh	View menu, Refresh option	Synchronizes the selected project or folder with the Architect/Requirements server.
Search	Tools menu, Search option	Activates the Search module.
Set the number of lines per row	View menu, Lines Per Row option	<p>Opens the Lines per row field, where you can enter the number of lines of text to display in the hierarchical content table. This feature is enabled when the Text property column is displayed, though the current setting is always effective.</p> <p>You can enter any number in the field. The default value is 1.</p> <p>In the View menu Lines Per Row option, you can choose 1 Line, 5 Lines, 10 Lines, or 20 Lines.</p>

To control the behavior and display of the toolbar, you can do the following:

- Rearrange button groups within the toolbar area.
- Display button groups in floating windows, independent of the toolbar area.

Rearranging Button Groups Within the Toolbar Area

Each group of buttons can be moved to different positions within the toolbar area.

To rearrange button groups within the toolbar area:

1. At the left of a button group, point to the vertical bar and hold down the left mouse button.
2. Drop the group in the new position within the toolbar area.

For each additional group that you want to move, repeat steps 1 and 2.

Displaying Button Groups in Floating Windows

Each group of buttons can be temporarily detached from the toolbar area and displayed in a separate floating window. These independent windows can be moved to any position on your screen, through the standard Microsoft Windows functions.

Also by standard functions, the windows can be closed to return them to the toolbar area. Floating toolbar windows cannot be resized, minimized, or maximized.

To display button groups in floating windows:

1. At the left of a button group, point to the vertical bar and hold down the left mouse button.
2. Move the pointer out of the toolbar area, and then release the mouse button.

For each additional group that you want to detach, repeat steps 1 and 2.

Module Bar

The module bar contains a button for each module of Architect/Requirements. By clicking a button, you gain access to one of the following modules:

- Search
- Systems Engineering and Requirements Management
- Administration

To control the display of the module bar, you can do the following:

- Disable or enable the module bar.
- Hide or show the module bar.
- Resize the module bar.

Disabling or Enabling the Module Bar

You can disable or enable the module bar at any time. Your last setting in the current session is preserved when you start a new session.



The module bar is enabled by default in the first session after the Architect/Requirements Client with Office Integration installation.

To disable or enable the module:

Pull down the **View** menu and choose **Module Bar**.

Hiding or Showing the Module Bar

You can hide or show the module bar at any time during the current session.

To hide the module bar:

Click the left arrow at the top of the vertical split bar to the right of the module bar.

To show the module bar:

Click the right arrow at the top of the leftmost vertical split bar.

Resizing the Module Bar

You can change the width, but not the height, of the module bar.

To resize the module bar:

1. At the right of the module bar, point to the vertical split bar until the pointer becomes a horizontal double arrow.
2. Hold down the left mouse button, move the split bar left or right, and then release the mouse button.

Address Bar

The **Address** bar shows the full path of the project or folder selected in the navigation tree. You cannot directly change the path in the **Address** bar. The path changes automatically with your navigation tree selection.



The **Address** bar does not show the full path of any object selected in the content table or in the notebook pane. That path is shown by the **Full Name** property for the object.

When versioning is enabled for a project, the **Address** bar also contains controls for choosing effectivity rules. An effectivity rule determines which version of a requirement or a building block is displayed in the main window. For more information, see chapter 10, [Working With Versions](#).



If you switch to another project that also has versions enabled and then switch back to the first project, the **Address** bar retains its previous size. If you switch to another project that does not have versions enabled and the **Effectivity** list is not displayed, the **Address** bar does not retain the previous size when you return to the original version-enabled project.

To control the display of the **Address** bar, you can do the following:

- Disable or enable the **Address** bar.
- Hide or show the **Address** bar.
- Resize the **Address** bar.

Disabling or Enabling the Address Bar

You can disable or enable the **Address** bar at any time. Your last setting in the current session is preserved when you start a new session.



- The **Address** bar is enabled by default in the first session after the Architect/Requirements Client with Office Integration installation.
- The controls for choosing effectivity rules are not accessible when the **Address** bar is disabled.

To disable or enable the Address bar:

Pull down the **View** menu and choose **Address Bar**.

Hiding or Showing the Address Bar

You can hide or show the **Address** bar at any time during the current session.

To hide the Address bar:

Click the up arrow on the left of the horizontal split bar below the **Address** bar.

You can simultaneously hide the **Address** bar, the navigation tree, the content table, and the notebook pane by clicking the right arrow at the top of the vertical split bar to the left of the **Address** bar.

To show the Address bar:

Click the down arrow on the left of the horizontal split bar above the navigation tree or the content table.

If the navigation tree or the content table is not displayed, click the down arrow on the left of the topmost horizontal split bar.

Resizing the Address Bar

You can change both the width and the height of the **Address** bar.

To change the width:

1. At the left of the **Address** bar, point to the vertical split bar until the pointer becomes a horizontal double arrow.
2. Hold down the left mouse button, move the split bar left or right, and then release the mouse button.

To change the height:

1. At the bottom of the **Address** bar, point to the horizontal split bar until the pointer becomes a vertical double arrow.
2. Hold down the left mouse button, move the split bar up or down, and then release the mouse button.

Navigation Tree

The navigation tree displays all projects to which you have access. You use the navigation tree to open and work with those projects and their folders. Also, you gain access to your Architect/Requirements Recycle Bin through the navigation tree.

Selecting a project, a folder, or the Recycle Bin in the navigation tree displays that object's contents in the content table. Selecting the root node clears the content table. For more information, see [Content Table](#), later in this chapter.

Each project node or folder can have a unique view of property columns in the hierarchical content table. These views let you easily work with the properties. For more information, see [Customizing Views of Property Columns](#) in chapter 9, *Working With Object Properties*.

To control the behavior and display of the navigation tree, you can do the following:

- Expand projects and folders.
- Disable or enable the navigation tree.
- Disable or enable the root node.
- Hide or show the navigation tree.
- Switch to the navigation tree.
- Resize the navigation tree.

Expanding a Project or a Folder

The navigation tree displays projects below the root node and folders below the project nodes. These folders may contain other folders, at lower levels in the navigation tree. You expand a project and its folders to view the hierarchy.

To expand a project or a folder:

Click the plus sign (+) to the left of the project or folder.

The plus sign becomes a minus sign (–), and the navigation tree displays the folders at the next lower level.



- To enable plus signs, pull down the **View** menu and choose **Root Node**.
- A plus sign is not displayed if the folder does not contain subfolders. The plus sign is removed if the last subfolder is moved or deleted from the folder.
- If you click a plus sign and the folder does not expand, it contains subfolders to which you do not have **Read** access. If you have questions about folder access, consult your project administrator.

You can also right-click the project or folder and choose **Expand** from the pop-up menu. You can collapse an expanded project or folder by clicking the minus sign, or by right-clicking the project or folder and choosing **Collapse** from the pop-up menu.

Disabling or Enabling the Navigation Tree

You can disable or enable the navigation tree at any time. Your last setting in the current session is preserved when you start a new session.



The navigation tree is enabled by default in the first session after the Architect/Requirements client installation.

To disable or enable the navigation tree, pull down the **View** menu and choose **Navigation Tree**.

Disabling or Enabling the Root Node

The root node occupies the highest level in the navigation tree. Directly below the root node, project nodes are displayed in alphabetical order. You can disable or enable the root node at any time. Your last setting in the current session is preserved when you start a new session.

With the root node enabled, the navigation tree displays plus signs (+) to the left of the projects, and you can click the plus signs to see the folders below the project nodes. With the root node disabled, the navigation tree displays projects without the root node and plus signs, and you have more space in the navigation tree.



The root node is enabled by default in the first session after the Architect/Requirements client installation.

To disable or enable the root node:

Pull down the **View** menu and choose **Root Node**.

Hiding or Showing the Navigation Tree

You can hide or show the navigation tree at any time during the current session.

To hide the navigation tree:

Click the left arrow at the top of the vertical split bar to the right of the tree.

To show the navigation tree:

Click the right arrow at the top of the rightmost vertical split bar.

Switching to the Navigation Tree

If you prefer to use the keyboard instead of the mouse, you can transfer the focus to the navigation tree for keyboard operations.

To switch to the navigation tree:

Pull down the **View** menu and choose the **Go To**→**Navigation Tree** options.

Resizing the Navigation Tree

You can change both the width and the height of the navigation tree.

To change the width:

1. At the right of the navigation tree, point to the vertical split bar until the pointer becomes a horizontal double arrow.
2. Hold down the left mouse button, move the split bar left or right, and then release the mouse button.

To change the height:

1. At the top of the navigation tree, point to the horizontal split bar until the pointer becomes a vertical double arrow.
2. Hold down the left mouse button, move the split bar up or down, and then release the mouse button.

Content Table

The content table displays the contents of the object selected in the navigation tree:

- When you select a project node, the content table displays the folders at the project's primary level, directly below the project node in the navigation tree.
- When you select a folder, the content table displays the folders, requirements, building blocks, and groups at the top level of the selected folder.
- When you select your Architect/Requirements Recycle Bin, the content table displays the objects that you deleted from all projects to which you have access.

Depending on your selection in the navigation tree, the content table displays the objects in one of two aspects:

- For a project or a folder, the objects are displayed hierarchically, according to superior and subordinate relationships among the objects.
- For the Recycle Bin, the objects are displayed sequentially, regardless of their levels and relationships in hierarchies from which they were deleted.

Both content table aspects display information in rows and columns, with the following similarities:

- Each row displays one object.
- Each column displays a property name and values for the objects to which the property applies.
-

Special graphics, called *indicators*, symbolize property names and values in some columns.

However, the functions of the hierarchical content table differ from those of the Recycle Bin. The two aspects differ also in the columns that they display.

To control the behavior and display of the content table, you can do the following:

- View objects in a hierarchy.
- Disable or enable the content table.
- Hide or show the content table.
- Switch to the content table.
- Resize the content table.

Hierarchical Content Table

The content table displays the hierarchical view when you select a project or a folder in the navigation tree. For a project, the initial view shows the folders at the primary level, directly below the project node. For a folder, the initial view shows the folders, requirements, building blocks, and groups at the selected folder's top level.

Below either level, the **Name** column shows the hierarchy of superior and subordinate objects. This column is permanently set and contains the following:



A plus sign (+) for each object that has subordinates, or *members*, at the next lower level. You can click a plus sign to see the members directly below the object. For more information, see [Viewing Objects in a Hierarchy](#), later in this chapter.

The **Member Count** property shows the number of members for a selected object. You can view this property in the **Properties** tab or floating window. For more information, see [Properties Tab](#) and [Using Tabs in Floating Windows](#), later in this chapter.



If you click a plus sign and the folder does not expand, it may contain subfolders to which you do not have **Read** access. If you have questions about folder access, consult your project administrator.



An object type indicator in each row. The indicator shows whether that object is a folder, a requirement, a building block, a group, a document (the system-defined folder subtype), or a paragraph (the system-defined requirement subtype). The indicator may show the symbol for a user-defined subtype.

For a requirement, a building block, or a paragraph, the indicator may also show both the symbol for a variant and the symbol for an object that is under static control, or *frozen*. For more information, see chapter 10, [Working With Versions](#).

- An object name in each row, to the right of the object type indicator.

The other columns represent certain system-defined properties that are displayed by default. Each column heading contains a property name, under which the cells contain values for the objects to which the property applies. For more information, see appendix [System-Defined Properties in the Systems Engineering and Requirements Management Module](#).



For a value that exceeds the cell size, you can rest the pointer on the cell to see the entire value in a tooltip. Multiple lines are used if necessary.

In some default columns, indicators symbolize the property names and values:



The **Attachment Count** property column displays an attachments indicator for each object to which notes or diagrams are attached. You can click an indicator to see the notes and diagrams in the **Attachments** tab or floating window. For more information, see [Attachments Tab](#), later in this chapter.



The **Trace Link Count** property column displays a links indicator for each object that is connected to other objects by a trace link. You can click an indicator to see the defining and complying objects in the **Links** tab or floating window. For more information, see [Links Tab](#), later in this chapter.

-

If the change management package is enabled for the project, the **Change Log** property column displays a links indicator for each object to which a change log is attached. You can click an indicator to see the change log in the **Attachments** tab or floating window. For more information, see [Attachments Tab](#), later in this chapter, and chapter 12, [Using the Change Management Package](#).

You can remove the default columns, and you can add columns for other properties. For example, you can add the **Version Count** property column to see a versions indicator for each object with versions or variants. When you click one of these indicators, that object's versions and variants are displayed in the **Versions** tab or floating window. For more information, see [Adding and Removing Columns](#) in chapter 9, *Working With Object Properties*, and [Versions Tab](#), later in this chapter.

In the hierarchical content table, you can also do the following:

- Add and remove columns for user-defined properties, rearrange and resize columns, and sort by any column. For more information, see [Rearranging Columns](#), [Resizing Columns](#), and [Setting the Sort Column](#), later in this chapter.
- Save the current view of column settings, and recall a saved view to display those column settings at any time. For more information, see [Customizing Views of Property Columns](#) in chapter 9, *Working With Object Properties*.
- Export the current view to Microsoft Excel. For more information, see [Exporting Objects to Microsoft Office Excel](#) in chapter 4, *Maintaining a Project*.
- Change editable property values for the objects. For more information, see [Editing the Properties of a Selected Object](#), [Editing Properties in Table View Cells](#), and [Using the Live Excel Interface](#) in chapter 9, *Working With Object Properties*.

Figure 3-3 shows the hierarchical content table.

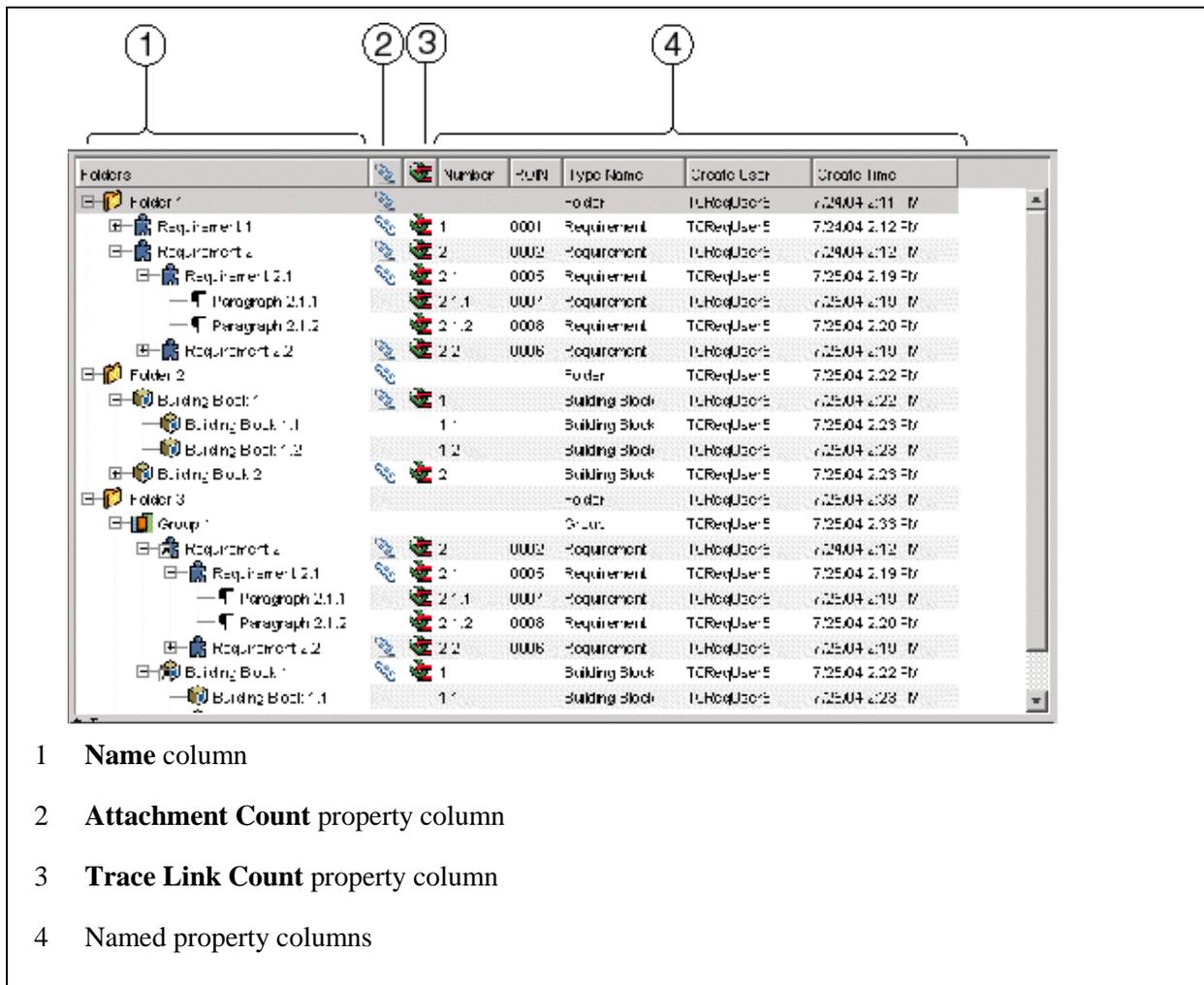


Figure 3-3. Hierarchical Content Table

Architect/Requirements Recycle Bin

When you select your Architect/Requirements Recycle Bin in the navigation tree, the content table displays the objects that you deleted from all projects to which you have access. This view displays the objects in the sequence of the current sort column, regardless of any relationships as parents, children, or siblings.

The columns in the Recycle Bin are permanently set and are displayed as follows:

- The leftmost column displays an object type indicator in each row, showing whether that object is a folder, requirement, building block, note, trace link, or group. The indicator may also show the symbol for a user-defined subtype and the symbol for a variant of a requirement or a building block.
- The attachments indicator column displays an indicator for each object to which one or more notes or diagrams are attached. You can click an indicator to see the notes and diagrams in the **Attachments** tab or floating window. For more information, see [Attachments Tab](#) and [Using Tabs in Floating Windows](#), later in this chapter; chapter 6, [Constructing System Views With Building Blocks and Diagrams](#); and chapter 8, [Recording Supplementary Information With Notes](#).
- The property columns display system-defined properties. Each column heading contains a property name, under which the cells contain values for the objects to which the property applies. For more information, see appendix [System-Defined Properties in the Systems Engineering and Requirements Management Module](#).

You can restore objects to the locations from which they were deleted. You can empty your Recycle Bin to remove the objects permanently from the database. For more information, see [Restoring Objects](#) and [Emptying Your Architect/Requirements Recycle Bin](#) in chapter 4, *Maintaining a Project*.



Your Recycle Bin can be emptied automatically after a number of days specified by your Architect/Requirements enterprise administrator. Also, your enterprise administrator can empty your Recycle Bin manually, for example, to manage storage space in the database. If you have questions about emptying your Recycle Bin, consult your enterprise administrator.

You cannot edit properties in the Recycle Bin. However, you can remove the default columns, and you can add columns for other system-defined properties. Also, you can rearrange and resize columns and sort by any column, and you can export the current view to Microsoft Excel. For more information, see [Adding and Removing Columns](#), [Rearranging Columns](#), [Resizing Columns](#), and [Setting the Sort Column](#) in chapter 9, *Working With Object Properties*, and [Exporting Objects to Microsoft Office Excel](#) in chapter 4, *Maintaining a Project*.

Figure 3-4 shows the Architect/Requirements Recycle Bin.

	Name ▲	RCIN	Original Location	Date Deleted	Type Name
				7/25/04 6:03 PM	Link
				7/25/04 6:06 PM	Link
	Building Block 4 -> Building Block 4.1			7/25/04 5:57 PM	Trace Link
	Building Block 4 -> Building Block 4.2			7/25/04 5:57 PM	Trace Link
	Experimental			7/25/04 6:04 PM	Diagram
	Folder 4		WExample Project	7/25/04 6:08 PM	Folder
	Group 4			7/25/04 6:04 PM	Group
	Note Folder 4			7/25/04 6:04 PM	Note
	Requirement 3	0011	WExample Project\Folder 1	7/25/04 2:39 PM	Requirement
	Requirement 1 -> Requirement 1.1			7/25/04 5:59 PM	Trace Link
	Requirement 4 -> Requirement 4.2			7/25/04 5:59 PM	Trace Link

13 object(s), 1 selected

- 1 Object type indicator column
- 2 Attachments indicator column
- 3 Property columns

Figure 3-4. Architect/Requirements Recycle Bin

Disabling or Enabling the Content Table

You can disable or enable the content table at any time. Your last setting in the current session is preserved when you start a new session.



The content table is enabled by default in the first session after the Architect/Requirements client installation.

To disable or enable the content table:

Pull down the **View** menu and choose **Content Table**.

Hiding or Showing the Content Table

You can hide or show the content table at any time during the current session.

To hide the content table:

Click the up arrow on the left of the horizontal split bar below the content table.

If the split bar does not display arrows, pull down the **View** menu and choose **Notebook Pane**.

You can hide the content table and the notebook pane simultaneously by clicking the right arrow at the top of the vertical split bar to the left.

To show the content table:

Do one of the following:

- If only the content table is hidden, click the down arrow on the left of the horizontal split bar above the notebook pane.
- If both the content table and the notebook pane are hidden, click the left arrow at the top of the rightmost vertical split bar.

If the content table is still hidden, click the down arrow on the left of the horizontal split bar above the notebook pane. If the split bar does not display arrows, pull down the **View** menu and choose **Notebook Pane**.

Switching to the Content Table

If you prefer to use the keyboard instead of the mouse, you can transfer the focus to the content table for keyboard operations.

To switch to the content table:

Pull down the **View** menu and choose the **Go To→Content Table** options.

Resizing the Content Table

You can change both the width and the height of the content table.

To change the width:

1. At the left of the content table, point to the vertical split bar until the pointer becomes a horizontal double arrow.
2. Hold down the left mouse button, move the split bar left or right, and then release the mouse button.

To change the height:

1. At the bottom of the content table, point to the horizontal split bar until the pointer becomes a vertical double arrow.
2. Hold down the left mouse button, move the split bar up or down, and then release the mouse button.

Notebook Pane

The notebook pane displays information for the object selected in the navigation tree or the content table. You work with that information through the following:

- The **Properties** tab displays the properties of the object selected in the navigation tree, the hierarchical content table, or your Architect/Requirements Recycle Bin. For more information, see [Properties Tab](#), later in this chapter.
- The **Attachments** tab displays notes, diagrams, spreadsheets, and change objects for the object selected in the navigation tree, content table, or Recycle Bin. For more information, see [Attachments Tab](#), later in this chapter.
- The **Links** tab displays each object that is connected by a trace link to the object selected in navigation tree or the hierarchical content table. For more information, see [Links Tab](#), later in this chapter.
- The **Connectivity** tab displays the ports and connections for the building block selected in the hierarchical content table. For more information, see [Connectivity Tab](#), later in this chapter.
- The **Preview** tab displays the content of the requirement selected in the hierarchical content table. For more information, see [Preview Tab](#), later in this chapter.
- The **Where Used** tab displays all groups, diagrams, and remote proxies that reference the object selected in the hierarchical content table. For more information, see [Where Used Tab](#), later in this chapter.
- The **Versions** tab displays all versions and variants of the object selected in the content table, or your Recycle Bin if versions are enabled for the project. For more information, see [Versions Tab](#), later in this chapter.

You can click the tabs to change from one to another, and work in the tab that is currently on top. In addition, you can open each tab in a separate floating window to display the selected object's information in multiple views. For more information, see [Using Tabs in Floating Windows](#), later in this chapter.

Floating tab windows also can display information for objects selected in the notebook pane:

- The **Properties** window displays the properties of the object selected in the **Attachments**, **Links**, **Connectivity**, **Where Used**, or **Versions** tab.
- The **Attachments** window displays notes, diagrams, spreadsheets, and change objects for the object selected in the **Links**, **Connectivity**, **Where Used**, or **Versions** tab.
- The **Links** window displays the trace links for the object selected in the **Connectivity**, **Where Used**, or **Versions** tab.
- The **Connectivity** tab displays the ports and connections for the building block selected in the **Links**, **Where Used**, or **Versions** tab.
- The **Preview** window displays the content of the note selected in the **Attachments** tab, or the content of the requirement selected in the **Links** tab.
- The **Where Used** window displays the groups, diagrams, and remote proxies for the object selected in the **Attachments**, **Links**, or **Connectivity** tab.
- The **Versions** window displays the versions and variants of the object selected in the **Links** or **Connectivity** tab.



Notebook pane tabs and windows show information for the most recently selected object.

To control the behavior and display of the notebook pane, you can do the following:

- Use tabs in floating windows.
- Disable or enable the notebook pane.
- Hide or show the notebook pane.
- Switch to the notebook pane.
- Resize the notebook pane.

Properties Tab

The **Properties** tab and floating window display all viewable properties that apply to the object selected in the navigation tree, the hierarchical content table, or your Architect/Requirements Recycle Bin. The **Properties** window displays the properties of the object selected in the **Attachments**, **Links**, **Connectivity**, **Where Used**, or **Versions** tab. For more information, see [Hierarchical Content Table](#) or [Architect/Requirements Recycle Bin](#), earlier in this chapter; and [Attachments Tab](#), [Links Tab](#), [Connectivity Tab](#), [Where Used Tab](#), [Versions Tab](#), or [Using Tabs in Floating Windows](#), later in this chapter.

This view contains a table that displays the properties sequentially. Each row displays one property, and the columns display the property names and values. You cannot add, remove, or rearrange columns in this table. However, you can resize both columns and sort by either column. For more information, see [Resizing Columns](#) and [Setting the Sort Column](#), later in this chapter.

Also, you can export the current view to Microsoft Excel, and you can change editable property values directly in the **Value** column. For more information, see [Exporting Objects to Microsoft Office Excel](#) in chapter 4, *Maintaining a Project*; [Editing the Properties of a Selected Object](#) in chapter 9, *Working With Object Properties*; and appendix [System-Defined Properties in the Systems Engineering and Requirements Management Module](#).

The **Properties** tab and window also contain controls that filter the properties displayed in the table. You can use these controls to view properties according to certain categories. For more information, see [Filtering Property Tables](#) in chapter 9, *Working With Object Properties*.

Figure 3-5 shows the **Properties** tab in the notebook pane.

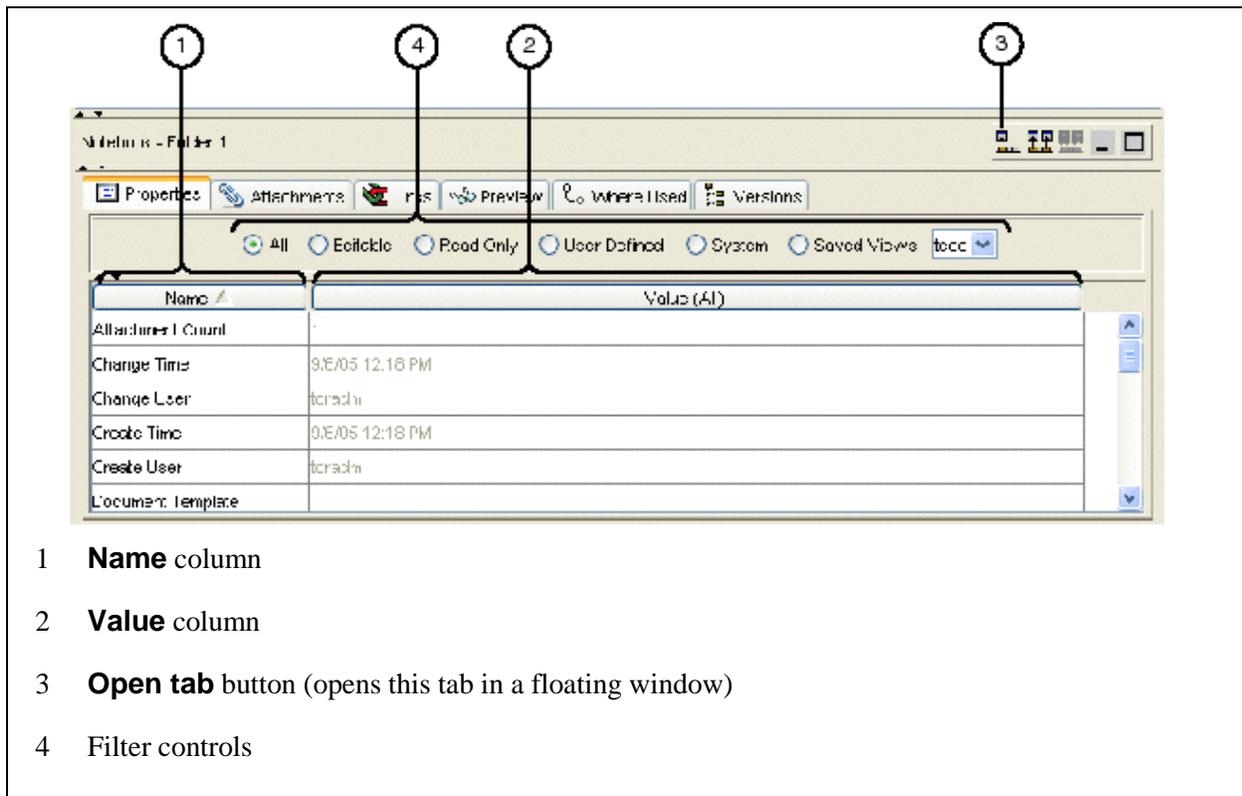


Figure 3-5. Properties Tab in Notebook Pane

Attachments Tab

The **Attachments** tab and floating window display the notes, diagrams, spreadsheets, change approval objects, and change logs attached to the object selected in the navigation tree, the hierarchical content table, or your Architect/Requirements Recycle Bin. The **Attachments** window displays the notes, diagrams, spreadsheets, change approval objects, and change logs for the object selected in the **Links**, **Connectivity**, **Where Used**, or **Versions** tab. For more information, see [Hierarchical Content Table](#) or [Architect/Requirements Recycle Bin](#), earlier in this chapter; and [Links Tab](#), [Connectivity Tab](#), [Where Used Tab](#), [Where Used Tab](#), [Versions Tab](#), or [Using Tabs in Floating Windows](#), later in this chapter.

This view contains a table that displays the objects hierarchically. The leftmost column, **Attachment**, contains the following:

- A plus sign (+) for each object, including notes, to which at least one note is attached. You can click the plus sign to see the notes at the next lower level. In turn, plus signs may be displayed for lower level notes to which other notes are attached.
- An object type indicator in each row. The indicator shows whether that object is a note, diagram, spreadsheet, a change approval object, or change log. The indicator may also show the symbol for a user-defined subtype.
- An object name in each row, to the right of the object type indicator.

The other columns display the default system-defined properties of the objects. In these columns, a heading contains the property name, under which the cells contain the values. For more information, see appendix [System-Defined Properties in the Systems Engineering and Requirements Management Module](#).



For a value that exceeds the cell size, you can rest the pointer on the cell to see the entire value in a tooltip. Multiple lines are used if necessary.

You can remove the default property columns, and you can add columns for more system-defined properties. Also, you can add and remove columns for user-defined properties, rearrange and resize columns, and sort by any column. For more information, see [Adding and Removing Columns](#), [Rearranging Columns](#), [Resizing Columns](#), and [Setting the Sort Column](#) in chapter 9, *Working With Object Properties*.

You can also do the following:

- Attach notes to objects in the **Attachments** tab and window, including other notes. For more information, see [Attaching a Note to an Object](#) in chapter 8, *Recording Supplementary Information With Notes*.
- Edit and view note content. For more information, see [Editing a Note](#) or [Viewing Note Content](#) in chapter 8, *Recording Supplementary Information With Notes*.
- Copy and move notes. For more information, see [Copying Objects](#) and [Moving Objects](#) in chapter 9, *Maintaining a Project*.
- Copy object URLs. For more information, see [Copying Object URLs](#) in chapter 9, *Maintaining a Project*.
- Edit and view diagrams. For more information, see [Editing a Diagram](#) in chapter 6, *Constructing System Views With Building Blocks and Diagrams*.
- Rename and delete objects. For more information, see [Renaming an Object](#) and [Deleting Objects](#) in chapter 4, *Maintaining a Project*.



When you delete a parent note, you also delete its child notes at all lower levels.

- Create reference links to notes, change logs, change approval objects, spreadsheets, and diagram images. For more information, see [Storing Property Values Through Reference Links](#) in chapter 9, *Working With Object Properties*, and [Creating a Diagram](#) in chapter 6, *Constructing System Views With Building Blocks and Diagrams*.
- Export objects to Microsoft Excel. For more information, see [Exporting Objects to Microsoft Office Excel](#) in chapter 4, *Maintaining a Project*.
- View and edit the properties of the objects, in this view or in the **Properties** floating window. For more information, see [Properties Tab](#), earlier in this chapter, and chapter 9, [Working With Object Properties](#).

Figure 3-6 shows the **Attachments** tab in the notebook pane.

Attachment	Text	Create User	Create Time
Component 1		tcradm	3/26/2007 1:39 ...
Diagram Image Note		tcradm	3/26/2007 1:42 ...
Design Note		tcradm	3/26/2007 1:48 ...
Testing Comments	Preliminary design testing con...	tcradm	3/26/2007 1:54 ...
Note	Some testing comments on th...	tcradm	3/26/2007 1:51 ...
SpreadSheet		tcradm	3/26/2007 1:43 ...
Functions	Excel functions include the fol...	tcradm	3/26/2007 1:45 ...

- 1 **Attachment** column
- 2 Property columns
- 3 **Open tab** button (opens this tab in a floating window)

Figure 3-6. Attachments Tab in Notebook Pane

Links Tab

The **Links** tab and floating window consist of the **Trace** and **Relations** subtabs. Each subtab displays specific information for the object selected in the navigation tree or the hierarchical content table.

The **Trace** subtab displays each object that defines and complies with the selected object. Defining objects may reside within the same Architect/Requirements project or in a different project. Complying objects may reside within any project and in other Teamcenter products.

The **Relations** subtab displays each trace link and connection for the selected object. In addition, the subtab displays the starting or ending object for each trace link and connection.

When the **Links** window is open, each subtab can display information for the object selected in the **Connectivity**, **Where Used**, and **Versions** tab. For more information, see [Hierarchical Content Table](#), earlier in this chapter; and [Connectivity Tab](#), [Where Used Tab](#), [Versions Tab](#), or [Using Tabs in Floating Windows](#), later in this chapter.

The following sections describe the **Trace** and **Relations** subtabs.

Trace Subtab

The **Trace** subtab consists of two panes, one for defining objects and one for complying objects. Each pane contains a table in which the leftmost column displays the linked objects in a hierarchy, according to their sequence in the defining or complying path:

- The **Defining Trace** column shows the defining objects that directly precede the selected object in the defining path. A plus sign (+) is shown for each defining object that complies with other objects, continuing the defining path upstream.
- The **Complying Trace** column shows the complying objects that directly follow the selected object in the complying path. A plus sign (+) is shown for each complying object that defines other objects, continuing the complying path downstream.

You can click a plus sign to see the next objects in a continuing path. For more information, see [Viewing Object Relationships](#) in chapter 7, *Showing Object Relationships With Trace Links*.

The hierarchical columns are permanently set, and each contains the following:

- An object type indicator in each row. The indicator shows whether that object is a folder, a requirement, a building block, a group, a document (the system-defined folder subtype), or a paragraph (the system-defined requirement subtype). The indicator may show the symbol for a user-defined subtype.

For a requirement, a building block, or a paragraph, the indicator may also show both the symbol for a variant and the symbol for an object that is under static control, or *frozen*. For more information, see chapter 10, [Working With Versions](#).
- An object name in each row, to the right of the object type indicator.

The other columns represent certain system-defined properties that are displayed by default. Each column heading contains a property name, under which the cells contain values for the objects to which the property applies. For more information, see appendix [System-Defined Properties in the Systems Engineering and Requirements Management Module](#).



For a value that exceeds the cell size, you can rest the pointer on the cell to see the entire value in a tooltip. Multiple lines are used if necessary.

You can remove the default property columns, and you can add columns for more system-defined properties. Also, you can add and remove columns for user-defined properties, rearrange and resize columns, and sort by any column. For more information, see [Adding and Removing Columns](#), [Rearranging Columns](#), [Resizing Columns](#), and [Setting the Sort Column](#) in chapter 9, *Working With Object Properties*.

In the **Trace** subtab, you can also do the following:

- Rename and delete linked objects, and export the current view to Microsoft Excel. For more information, see [Renaming an Object](#), [Deleting Objects](#), and [Exporting Objects to Microsoft Office Excel](#) in chapter 4, *Maintaining a Project*.
- Copy and move linked objects, through options on the **Edit** menu or by dropping selected objects on the destination. For more information, see [Copying Objects](#) and [Moving Objects](#) in chapter 4, *Maintaining a Project*.
- Open linked requirements to edit or view their content, and export linked requirements to Microsoft Word. For more information, see [Exporting Objects to Microsoft Office Word](#) in chapter 4, *Maintaining a Project*, and [Entering and Changing Requirement Content in Microsoft Office Word](#) and [Viewing Requirement Content](#) in chapter 5, *Managing Requirements*.
- Create and delete trace links, and navigate to linked objects. For more information, see [Creating Trace Links](#), [Linking to an Object in Another Teamcenter Product](#), [Deleting Trace Links for an Object](#), and [Navigating to a Linked Object](#) in chapter 7, *Showing Object Relationships With Trace Links*.
- Attach notes to defining and complying objects. For more information, see [Attaching a Note to an Object](#) in chapter 8, *Recording Supplementary Information With Notes*.
- Edit the properties of linked objects, in this view or in the **Properties** floating window. For more information, see [Properties Tab](#), earlier in this chapter; and [Editing the Properties of a Selected Object](#), [Editing Properties in Table View Cells](#), and [Using the Live Excel Interface](#) in chapter 9, *Working With Object Properties*.

Figure 3-7 shows the **Trace** subtab of the **Links** tab in the notebook pane.

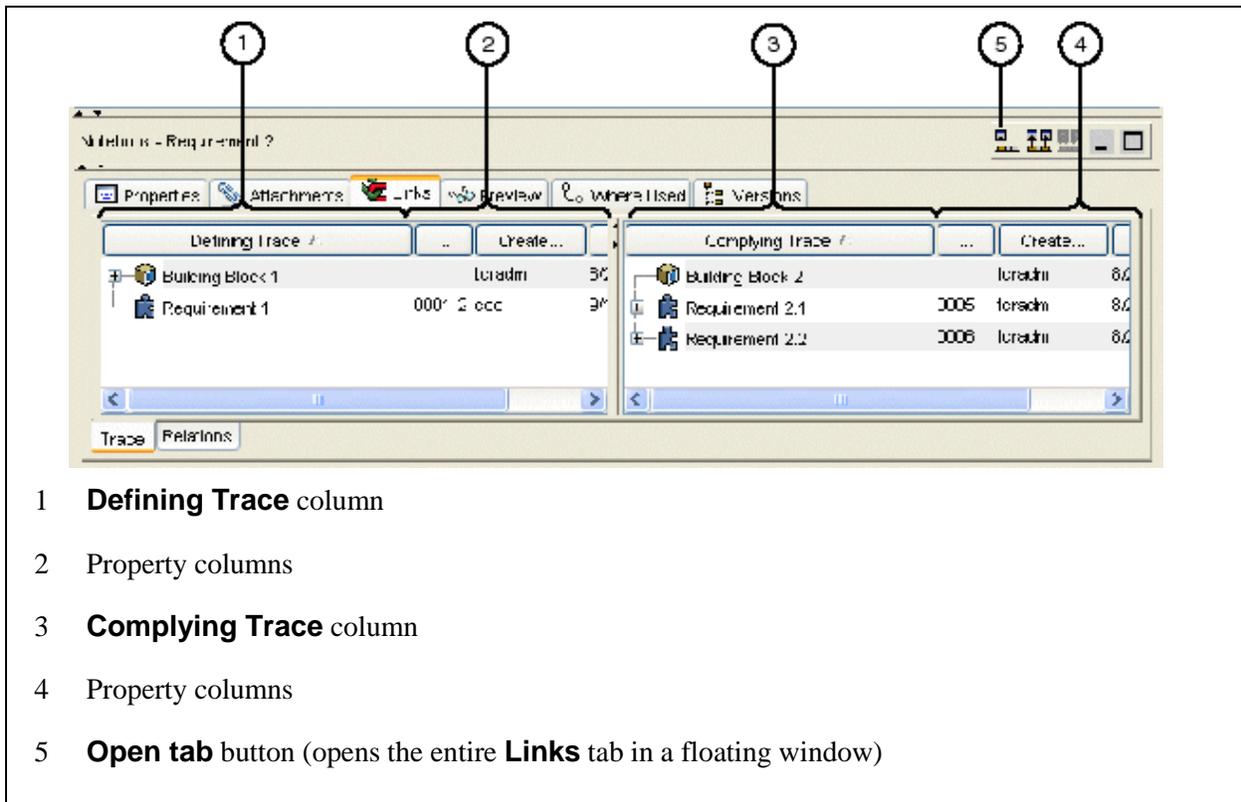


Figure 3-7. Trace Subtab of Links Tab in Notebook Pane

Relations Subtab

The **Relations** subtab displays trace links and connections as independent objects that you can view and modify. The subtab also displays the starting or ending object for each trace link and connection.

The **Relations** subtab consists of the **Start** pane and the **End** pane. Each pane contains two sequential tables, an *object table* at the left and a *link table* at the right:

- In the **Start** pane, the object table shows each object from which a trace link or a connection originates to the selected object. The actual trace link or connection is shown in the corresponding link table in the row that matches the starting object position in the object table.
- In the **End** pane, the object table shows each object at which a trace link or a connection originating from the selected object is completed. The actual trace link or connection is shown in the corresponding link table in the row that matches the ending object position in the object table.



In each pane, the rows in the object table and the link table remain synchronized at all times. For example, when you sort or scroll a link table, the corresponding object table is automatically sorted or scrolled to match the new view in the link table.

In each table in the **Relations** subtab, the leftmost column displays an object type indicator for each folder, document (the system-defined folder subtype), requirement, paragraph (the system-defined requirement subtype), building block, group, trace link, or connection. The indicator may show the symbol for a user-defined subtype.



For a requirement, a paragraph, or a building block, the indicator may also show both the symbol for a variant and the symbol for an object that is under static control, or *frozen*. For more information, see chapter 10, [Working With Versions](#).

The other columns display default system-defined properties for the objects in the table. In these columns, a heading contains the property name, under which the cells contain the values. You can remove the default property columns, and you can add columns for more system-defined properties. Also, you can add and remove columns for user-defined properties, rearrange and resize columns, and sort by any column. For more information, see appendix [System-Defined Properties in the Systems Engineering and Requirements Management Module](#); and [Adding and Removing Columns, Rearranging Columns, Resizing Columns](#), and [Setting the Sort Column](#) in chapter 9, *Working With Object Properties*.

In the **Relations** subtab, you can also do the following:

- Rename and delete linked objects, trace links, and connections. For more information, see [Renaming an Object](#) and [Deleting Objects](#) in chapter 4, *Maintaining a Project*.
- Export the trace links, connections, or starting and ending objects in a given table to Microsoft Excel or Microsoft Word. For more information, see [Exporting Objects to Microsoft Office Excel](#) and [Exporting Objects to Microsoft Office Word](#) in chapter 4, *Maintaining a Project*.
- Open starting and ending requirements to edit or view their content. For more information, see [Entering and Changing Requirement Content in Microsoft Office Word](#) and [Viewing Requirement Content](#) in chapter 5, *Managing Requirements*.
- Create and delete trace links and connections, and navigate to starting and ending objects. For more information, see [Creating Trace Links](#), [Deleting Trace Links for an Object](#), and [Navigating to a Linked Object](#) in chapter 7, *Showing Object Relationships With Trace Links*.
- Attach notes to trace links and connections. For more information, see [Attaching a Note to an Object](#) in chapter 8, *Recording Supplementary Information With Notes*.

- Edit the properties of trace links, connections, and their starting or ending objects. You can edit properties in this view or in the **Properties** floating window. For more information, see [Properties Tab](#), earlier in this chapter; and [Editing the Properties of a Selected Object](#), [Editing Properties in Table View Cells](#), and [Using the Live Excel Interface](#) in chapter 9, *Working With Object Properties*.

Figure 3-8 shows the **Relations** subtab of the **Links** tab in the notebook pane.

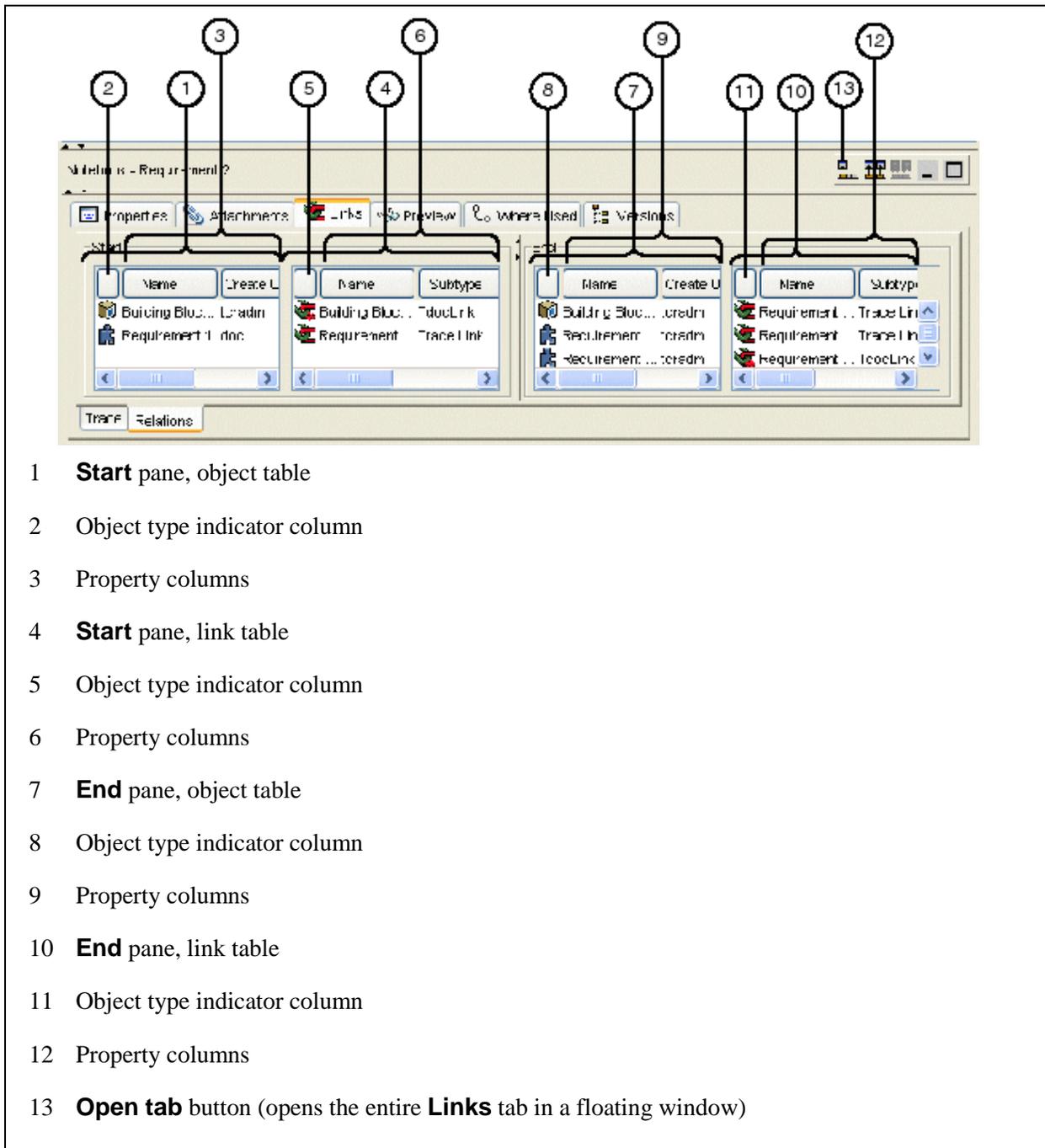


Figure 3-8. Relations Subtab of Links Tab in Notebook Pane

Connectivity Tab

The **Connectivity** tab and floating window display the ports and connections for the building block selected in the hierarchical content table. The **Connectivity** window displays the ports and connections for the building block selected in the **Links, Where Used**, or **Versions** tab.

This view contains a table that displays the objects hierarchically. The **Name** column contains the following:

- A plus sign (+) for each port that has at least one connection. You can click the port's plus sign to see the connections at the next lower level.
Below each connection, the port at the other end of the connection is displayed when you click the connection's plus sign. This hierarchy may extend to multiple levels below any given port.
- An object type indicator in each row. The indicator shows whether that object is a port or a connection. The indicator may also show the symbol for a user-defined subtype.
- An object name in each row, to the right of the object type indicator.

The other columns display the default system-defined properties of the ports and connections. In these columns, a heading contains the property name, under which the cells contain the values. For more information, see appendix [System-Defined Properties in the Systems Engineering and Requirements Management Module](#).



For a value that exceeds the cell size, you can rest the pointer on the cell to see the entire value in a tooltip. Multiple lines are used if necessary.

You can remove the default property columns, and you can add columns for more system-defined properties. Also, you can add and remove columns for user-defined properties, rearrange and resize columns, and sort by any column. For more information, see [Adding and Removing Columns](#), [Rearranging Columns](#), [Resizing Columns](#), and [Setting the Sort Column](#) in chapter 9, *Working With Object Properties*.

You can also do the following:

- Attach notes to ports and connections by using the **Attachments** window. For more information, see [Attaching a Note to an Object](#) in chapter 8, *Recording Supplementary Information With Notes*.
- Copy and move ports and connections. For more information, see [Copying Objects](#) and [Moving Objects](#) in chapter 9, *Maintaining a Project*.
- Copy the URLs of ports and connections. For more information, see [Copying Object URLs](#) in chapter 9, *Maintaining a Project*.
- Rename and delete ports and connections. For more information, see [Renaming an Object](#) and [Deleting Objects](#) in chapter 4, *Maintaining a Project*.

- Create reference links to ports and connections. For more information, see [Storing Property Values Through Reference Links](#) in chapter 9, *Working With Object Properties*, and [Creating a Diagram](#) in chapter 6, *Constructing System Views With Building Blocks and Diagrams*.
- Export ports and connections to Microsoft Excel. For more information, see [Exporting Objects to Microsoft Office Excel](#) in chapter 4, *Maintaining a Project*.
- View and edit the properties of the ports and connections, in this view or in the **Properties** floating window. For more information, see [Properties Tab](#), earlier in this chapter, and chapter 9, *Working With Object Properties*.

Figure 3-9 shows the **Connectivity** tab in the notebook pane.

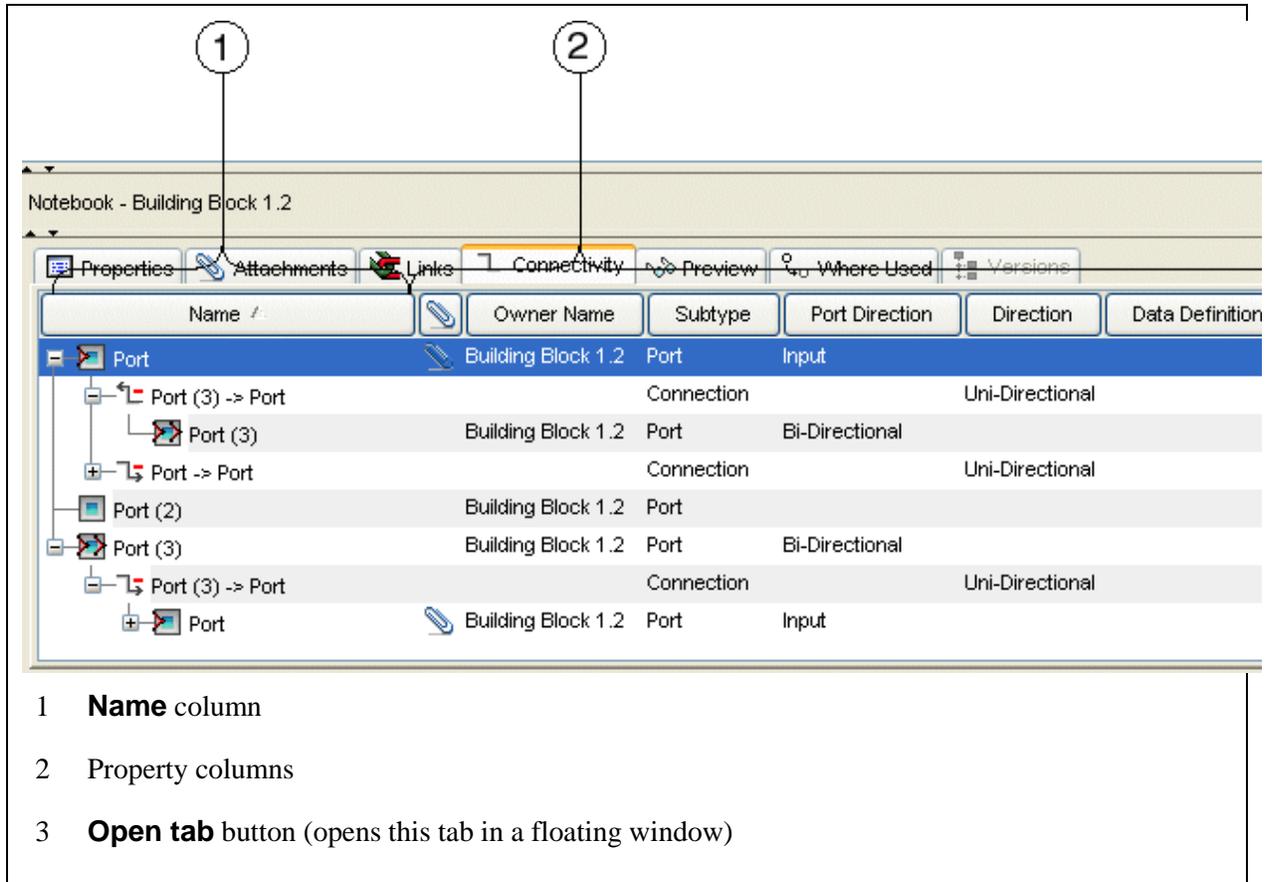


Figure 3-9. Connectivity Tab in Notebook Pane

Preview Tab

The **Preview** tab and floating window display the content of the requirement selected in the hierarchical content table. For more information, see [Hierarchical Content Table](#), earlier in this chapter.

The **Preview** window displays the content of the following:

- The note or diagram selected in the **Attachments** tab. For more information, see [Attachments Tab](#), earlier in this chapter.
- The requirement selected in the **Links** or **Versions** tab. For more information, see [Links Tab](#), earlier in this chapter, or [Versions Tab](#) and [Using Tabs in Floating Windows](#), later in this chapter.

You can see the entire content without opening the requirement, note, or diagram. You can print the content by right-clicking in content area and choosing **Print** from the pop-up menu. However, you cannot change the content from this view. For more information, see [Entering and Changing Requirement Content in Microsoft Office Word](#) in chapter 5, *Managing Requirements*; [Editing a Note](#) in chapter 8, *Recording Supplementary Information With Notes*; or [Editing a Diagram](#) in chapter 6, *Constructing System Views With Building Blocks and Diagrams*.



A multiple-line tooltip that you display in the content table may overlap the **Preview** tab, disabling the tab temporarily. The tab is enabled when the tooltip is removed.

Figure 3-10 shows the **Preview** tab in the notebook pane.

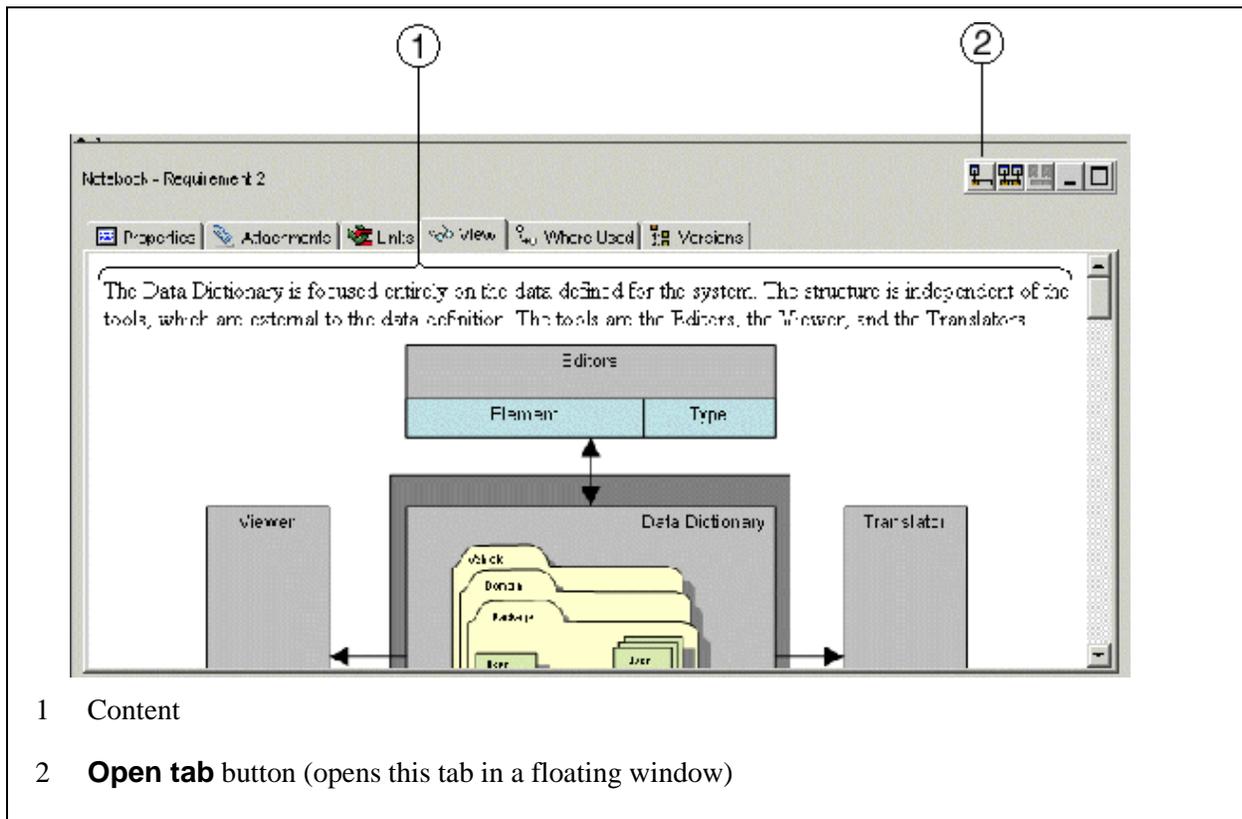


Figure 3-10. Preview Tab in Notebook Pane

Where Used Tab

The **Where Used** tab and floating window display the groups, diagrams, and remote proxies that reference the object selected in the hierarchical content table. The **Where Used** window displays such references for the object selected in the **Attachments**, **Links**, or **Connectivity** tab. For more information, see [Hierarchical Content Table](#), [Attachments Tab](#), [Links Tab](#), or [Connectivity Tab](#), [Where Used Tab](#), earlier in this chapter, or [Using Tabs in Floating Windows](#), later in this chapter.

This view contains a table that displays the groups, diagrams, and proxies sequentially. The leftmost column is permanently set and contains an object type indicator for each group, diagram, and proxy. The other columns display the default system-defined properties of those objects. In these columns, a heading contains the property name, under which the cells contain the values. For more information, see appendix [System-Defined Properties in the Systems Engineering and Requirements Management Module](#).

You can remove the default property columns, and you can add columns for more system-defined properties. Also, you can add and remove columns for user-defined properties, rearrange and resize columns, and sort by any column. For more information, see [Adding and Removing Columns](#), [Rearranging Columns](#), [Resizing Columns](#), and [Setting the Sort Column](#) in chapter 9, *Working With Object Properties*.

For the selected object in the hierarchical content table or the **Attachments** or **Links** tab, you can remove the object's reference from one or more groups displayed in the **Where Used** tab. For more information, see [Removing Group Members](#) in chapter 4, *Maintaining a Project*.

You can also do the following:

- Rename and delete groups, diagrams, and proxies, and export the table to Microsoft Excel. For more information, see [Renaming an Object](#), [Deleting Objects](#), and [Exporting Objects to Microsoft Office Excel](#) in chapter 4, *Maintaining a Project*.
- Open diagrams for editing. For more information, see [Editing a Diagram](#) in chapter 6, *Constructing System Views With Building Blocks and Diagrams*.
- View and edit properties for the objects, in this view or in the **Properties** window. For more information, see [Properties Tab](#), earlier in this chapter, and chapter 9, *Working With Object Properties*.

Figure 3-11 shows the **Where Used** tab in the notebook pane.

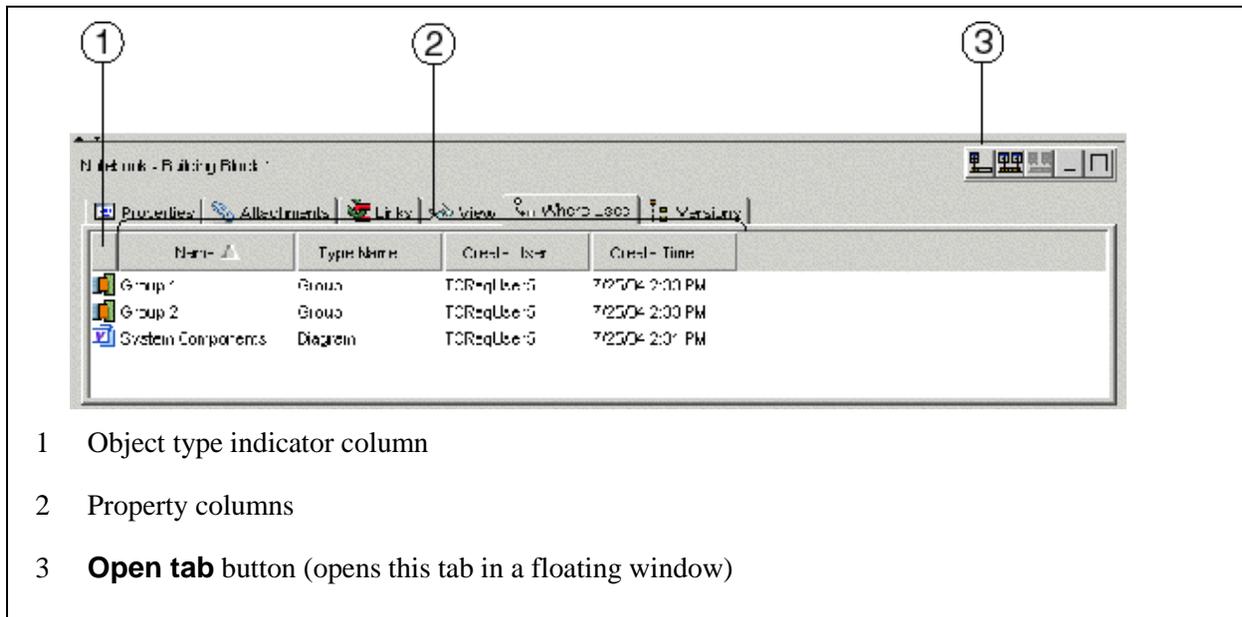


Figure 3-11. Where Used Tab in Notebook Pane

Versions Tab

The **Versions** tab and floating window display all versions and variants of the requirement or building block selected in the hierarchical content table, or your Architect/Requirements Recycle Bin if versions are enabled for the project. For more information, see [Enabling Versions for a Project](#) in chapter 10, *Working With Versions*.

The **Versions** window displays the versions and variants of the requirement or building block selected in the **Links** or **Connectivity** tab. For more information, see [Hierarchical Content Table](#), [Links Tab](#), and [Connectivity Tab](#), earlier in this chapter, and [Using Tabs in Floating Windows](#), later in this chapter.



The **Versions** tab is not available if versioning is disabled for the project.

This view contains a table that displays the versions and variants in a hierarchy, from the earliest to the latest. The hierarchical column, **Name**, is permanently set and contains the following:

- A plus sign (+) for each object that has later versions or variants. You can click a plus sign to see the later ones in separate rows below the earlier one. For more information, see [Viewing a Version Tree](#) in chapter 10, *Working With Versions*.
- An object type indicator in each row, showing whether the object is a requirement, a building block, or a paragraph (the system-defined requirement subtype). The indicator may also show the symbol for a user-defined subtype, the symbol for a variant, and the symbol for an object that is under static control, or *frozen*.
- An object name in each row, to the right of the object type indicator.

The other columns represent certain system-defined properties that are displayed by default. Each column heading contains a property name, under which the cells contain values for the objects to which the property applies. For more information, see appendix [System-Defined Properties in the Systems Engineering and Requirements Management Module](#).



For a value that exceeds the cell size, you can rest the pointer on the cell to see the entire value in a tooltip. Multiple lines are used if necessary.

You can remove the default property columns, and you can add columns for more system-defined properties. Also, you can add and remove columns for user-defined properties, rearrange and resize columns, and sort by any column. For more information, see [Adding and Removing Columns](#), [Rearranging Columns](#), [Resizing Columns](#), and [Setting the Sort Column](#) in chapter 9, *Working With Object Properties*.

You can also do the following:

- Rename versions and variants, and export the current view to Microsoft Excel. For more information, see [Renaming an Object](#) and [Exporting Objects to Microsoft Office Excel](#) in chapter 4, *Maintaining a Project*.
- Open requirement versions and variants in Microsoft Word to edit or view their content. For more information, see [Entering and Changing Requirement Content in Microsoft Office Word](#) or [Viewing Requirement Content](#) in chapter 5, *Managing Requirements*.
- Create defining and complying trace links, within the project, for versions and variants. For more information, see [Creating Trace Links](#) in chapter 7, *Showing Object Relationships With Trace Links*.

- Edit the properties of versions and variants, in this view or in the **Properties** floating window. For more information, see [Properties Tab](#), earlier in this chapter; and [Editing the Properties of a Selected Object](#) and [Editing Properties in Table View Cells](#) in chapter 9, *Working With Object Properties*.

Figure 3-12 shows the **Versions** tab in the notebook pane.

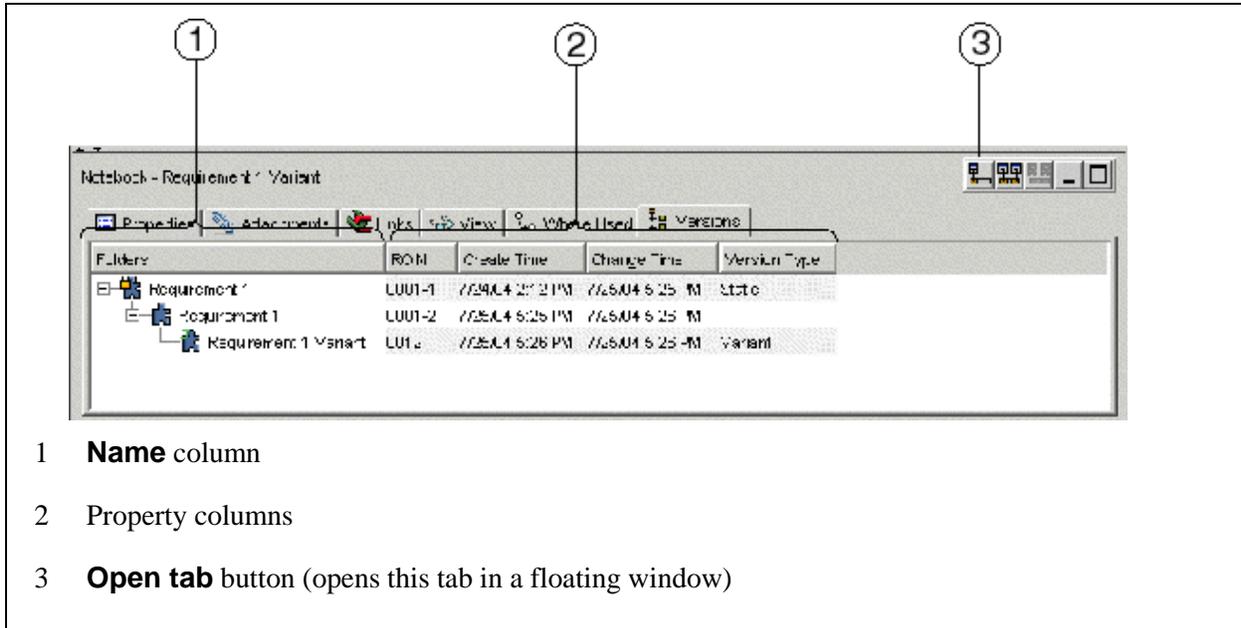


Figure 3-12. Versions Tab in Notebook Pane

Using Tabs in Floating Windows

For each tab in the notebook pane, you can temporarily transfer the information from the tab to a floating window. By opening a tab window, you can keep that information in view while you work on another tab directly in the notebook pane. Conversely, you can view a fixed tab in the notebook pane while working in a floating window.

Using the notebook pane's toolbar, you can open the tab that is currently on top, or you can open all tabs simultaneously, each tab in a separate window. Also with the toolbar, you can close all open tab windows in one action.

You close one window at a time through the standard Microsoft Windows functions. Also by standard functions, you can resize, minimize, and maximize the windows, and move them to any position on your screen.

In the same way as you work on the corresponding tab, you can use each floating window to work with information for the object selected in the content table. Moreover, tab windows extend your work to objects selected in the notebook pane:

- In the **Properties** window, you can work with the properties that apply to the object selected in the **Attachments**, **Links**, **Connectivity**, **Where Used**, or **Versions** tab. For more information, see [Properties Tab](#), earlier in this chapter.
- In the **Attachments** window, you can work with the notes, diagrams, spreadsheets, or change approval objects attached to the object selected in the **Links**, **Connectivity**, **Where Used**, or **Versions** tab. For more information, see [Attachments Tab](#), earlier in this chapter.
- In the **Links** window, you can work with the trace links and linked objects for the object selected in the **Connectivity**, **Where Used**, or **Versions** tab. For more information, see [Links Tab](#), earlier in this chapter.
- In the **Connectivity** window, you can work with the ports and connections for the building block selected in the **Links**, **Where Used**, or **Versions** tab. For more information, see [Connectivity Tab](#), earlier in this chapter.
- In the **Preview** window, you can see the content of the note or diagram selected in the **Attachments** tab and of the requirement selected in the **Links** tab. For more information, see [Preview Tab](#), earlier in this chapter.
- In the **Where Used** window, you can work with the groups, diagrams, and remote proxies for the object selected in the **Attachments**, **Links**, or **Connectivity** tab. For more information, see [Where Used Tab](#), earlier in this chapter.
- In the **Versions** window, you can work with the versions and variants of the requirement or building block selected in the **Links** tab. For more information, see [Versions Tab](#), earlier in this chapter.

Your column settings in the **Attachments**, **Links**, **Connectivity**, **Where Used**, and **Versions** tabs are maintained in the corresponding windows. Also, columns that you add and remove in the floating windows are maintained in the tabs when you close the windows. For more information, see [Adding and Removing Columns](#) in chapter 9, *Working With Object Properties*.

To enhance navigation among the views, you can add columns in which indicators symbolize the names and values of certain system-defined properties:

-

The **Attachment Count** property column contains an attachments indicator for each object to which one or more notes, diagrams, spreadsheets, or change approval objects are attached. You can click an indicator to see that object's attachments in the **Attachments** tab.

-

The **Trace Link Count** property column contains a links indicator for each object to which one or more objects are connected by a trace link. You can click an indicator to see that object's trace links in the **Links** tab.

-

The **Version Count** property column contains a versions indicator for each object that has one or more versions or variants. You can click an indicator to see that object's versions and variants in the **Versions** tab.

To open floating tab windows:

On the notebook pane's toolbar, do one of the following:

- To open the top tab, click the **Open tab** button.

The new window opens as the active window. The next unopened tab in the display order becomes the top tab. For example, if you open the **Links** window, that tab is dimmed in the notebook pane, and the **Preview** tab is on top.

- To open all tabs simultaneously, click the **Open all** button.

The notebook pane reduces to minimum height, and a new window opens for each tab. The windows are cascaded, with the **Preview** window active.

To close floating tab windows:

Do one of the following:

- To close all windows, click the **Close all** button on the notebook pane's toolbar.

The windows return to their fixed tab positions. The top tab depends on which windows were open, in conjunction with the tab display order from left to right. For example, if only the **Properties** and **Links** windows were open, the **Links** tab, the rightmost of the two in the display order, becomes the top tab.

- To close a single window, click the **Close** box in the upper right corner.

The window returns to its fixed tab position and becomes the top tab.

Figure 3-13 shows the toolbar in the notebook pane.

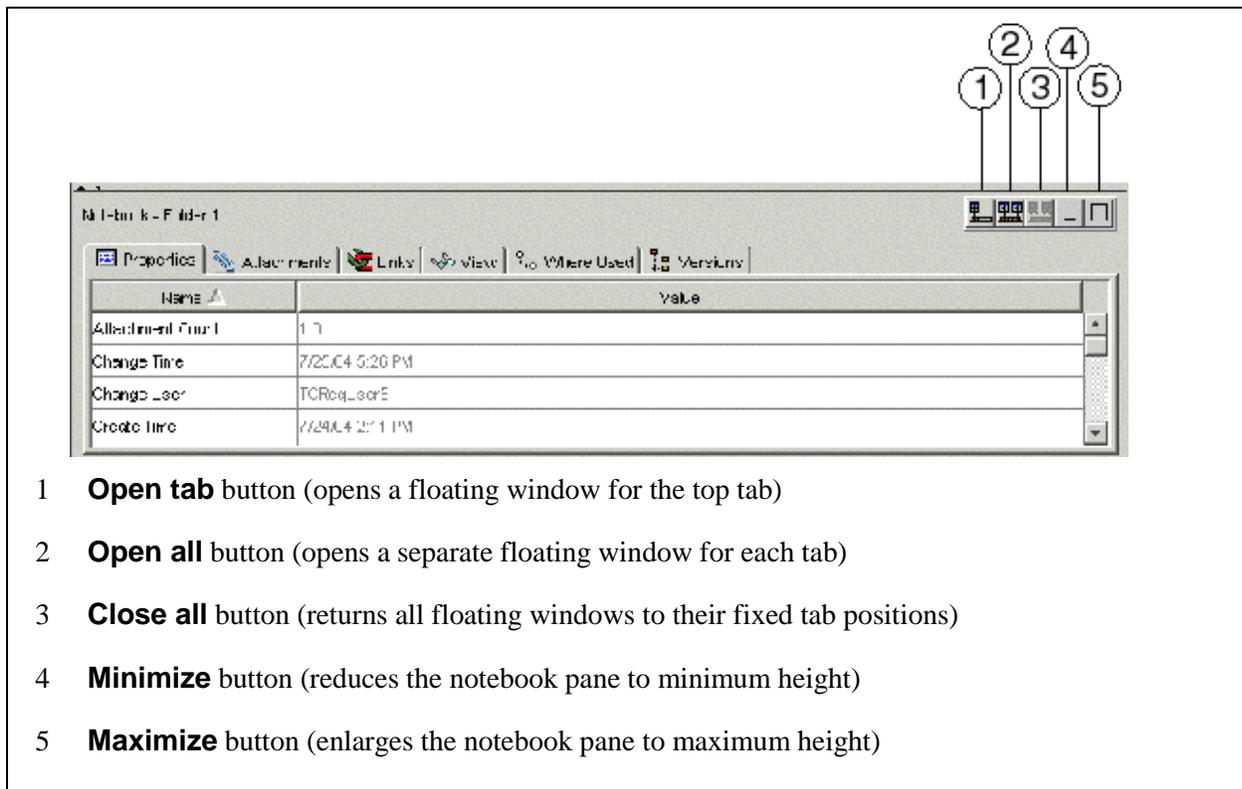


Figure 3-13. Toolbar in Notebook Pane

Disabling or Enabling the Notebook Pane

You can disable or enable the notebook pane at any time. Your last setting in the current session is preserved when you start a new session.



The notebook pane is enabled by default in the first session after the Architect/Requirements client installation.

To disable or enable the notebook pane:

Pull down the **View** menu and choose **Notebook Pane**.

Hiding or Showing the Notebook Pane

You can hide or show the notebook pane at any time during the current session.

To hide the notebook pane:

Click the down arrow on the left of the horizontal split bar above the notebook pane.

You can hide the notebook pane and the content table simultaneously by clicking the right arrow at the top of the vertical split bar to the left.



In the **Links** tab, you can hide a pane by clicking the left arrow or the right arrow on the vertical split bar between the panes.

To show the notebook pane:

Do one of the following:

- If only the notebook pane is hidden, click the up arrow on the left of the horizontal split bar below the content table.
- If both the notebook pane and the content table are hidden, click the left arrow at the top of the rightmost vertical split bar.

If the notebook pane is still hidden, click the up arrow on the left of the horizontal split bar below the content table.



In the **Links** tab, either pane may be hidden, with the tab displaying a vertical split bar to the left or the right of the pane that is shown. To show the **Defining Trace** pane, click the right arrow on the split bar. To show the **Complying Trace** pane, click the left arrow on the split bar.

Switching to the Notebook Pane

If you prefer to use the keyboard instead of the mouse, you can transfer the focus to the notebook pane for keyboard operations.

To switch to the notebook pane:

Pull down the **View** menu and choose the **Go To→Notebook Pane** options.

Resizing the Notebook Pane

You can manually change both the width and the height of the notebook pane. In addition, you can automatically reduce the notebook pane to its minimum height, enlarge it to its maximum height, and restore it to its default height.

When the notebook pane is reduced, the tabs are not visible, and the content table extends toward the bottom of the main window. When the notebook pane is enlarged, it extends toward the top of the main window, and the content table is not visible.

To manually change the width:

1. At the left of the notebook pane, point to the vertical split bar until the pointer becomes a horizontal double arrow.
2. Hold down the left mouse button, move the split bar left or right, and then release the mouse button.



In the **Links** tab, you can change the width of both panes by moving the vertical split bar between the panes.

To manually change the height:

1. At the top of the notebook pane, point to the horizontal split bar until the pointer becomes a vertical double arrow.
2. Hold down the left mouse button, move the split bar up or down, and then release the mouse button.

To automatically change the height:

On the notebook pane's toolbar (figure [Toolbar in Notebook Pane](#)), do one of the following:

- To reduce the notebook pane to its minimum height, click the **Minimize** button.

At the top of the notebook pane, its heading and toolbar remain visible below the horizontal split bar. On the toolbar, the **Minimize** button becomes the **Restore** button, and the **Maximize** button becomes unavailable.



While the notebook pane is reduced, you can use the other buttons to open the tabs in separate windows. For more information, see [Using Tabs in Floating Windows](#), earlier in this chapter.

- To enlarge the notebook pane to its maximum height, click the **Maximize** button.

On the toolbar, the **Maximize** button becomes the **Restore** button, and the **Minimize** button becomes unavailable.

- To restore the notebook pane to its default height, click the **Restore** button.



You can see a description of each button by resting the pointer on the button to display a tooltip.

Viewing Objects in a Hierarchy

A project defines a hierarchy in which objects reside at levels determined by the individual object types. Only folders can occupy the primary level, directly below the project node. Objects of other types must reside in folders.

The following views can contain such hierarchies:

- The hierarchical content table.
- The **Attachments, Links, and Versions** tabs and floating windows.
- The **Search Results** dialog window.
- The **Defining Object Traceability** and **Complying Object Traceability** windows.

In any folder's hierarchy, objects can be nested from the top level to progressively subordinate levels. The leftmost column displays a plus sign (+) for each object that has subordinates, or *members*, at the next lower level.

You can view only these members, which may themselves have members that extend the hierarchy to even lower levels. Or, you can view the entire hierarchy below the object, displaying its members and all lower level objects simultaneously.

For a folder or a group, members can be folders, requirements, building blocks, groups, and the system-defined and user-defined subtypes of those object types. For a requirement or a building block, only objects of that type are allowed as members, including system-defined and user-defined subtypes based on that object type.

The members of a folder, a requirement, or a building block reside in the same location as their superior. Therefore, these hierarchies show relationships of child and parent objects. Such relationships do not apply to groups and their members, because group members reside elsewhere in the project. Although group members occupy the next lower level, child and parent relationships shown for members reflect hierarchies in the actual locations of the members. For more information, see [Using Groups to Maintain Objects](#) in chapter 4, *Maintaining a Project*.

For a selected object of any of those types, the **Member Count** property shows the number of members at the next lower level. You can view this property in the **Properties** tab or floating window. For more information, see [Properties Tab](#) and [Using Tabs in Floating Windows](#), later in this chapter.

To view only the members of an object:

In the leftmost column, click the plus sign (+) to the left of the object.

The members are displayed directly below the object. The object's plus sign becomes a minus sign (-), which you can click to hide the members.



For performance reasons, plus signs are shown for some objects that may or may not have members when:

- A member has been deleted from the object and not yet emptied from the recycle bin (this affects the content pane but not the navigation pane).
- The object has members that do not match the current effectivity.
- The current user does not have **Read** access to the object's members. If you have questions about folder access, consult your project administrator.

To view the entire hierarchy below an object:

Select the object, pull down the **View** menu, and choose **Expand All**. Or, right-click the object and choose **Expand All** from the pop-up menu.

The object's members and all lower level objects are displayed. A minus sign (-) is shown for each object that has members. You can click a minus sign to hide all objects below that level.

Filtering Objects in a Hierarchy

Filtering objects in Architect/Requirements allows you to filter the displayed rows of data in a hierarchical view down to a smaller subset so that only objects whose property values meet the filter criteria that you specified are displayed. If you've ever worked with filters in a spreadsheet like Microsoft Excel, then you are already familiar with the concept of what data filtering does. It filters the data according to user specified filter criteria so that only the rows with objects that meet the filter rules displays. Because there is a filter available for each property column in the hierarchical view, the Architect/Requirements menu command refers to this as **column filters**. There is no ability to filter on the tree column of hierarchical view because the tree structure is needed to give objects their context.

You can filter objects in the following views:

- The hierarchical content table.
- The **Search Results** dialog window.
- The **Defining Object Traceability** and **Complying Object Traceability** windows.

One of the most important concepts to understand about data filtering in Architect/Requirements is that it is static filtering, not dynamic filtering. In static filtering, if the data changes to something that does not meet the filter criteria, it is still displayed until a user requests that the filter rules are applied again. When using static filtering and the new data appears in the table, it is not compared to the current filtered criteria. The new information is allowed to display even if it does not meet the filter rules because a user action caused it to appear. If it didn't work this way, then it would lead to all kinds of user interface problems. For example, if you attempt to insert an object while filters were on and dynamic filtering was being used, then the moment that you requested to insert a new object, it would probably not display because it did not meet one of the filter rules. Because it is a new object, you will not have an opportunity to enter any of its property values, including changing its name from the default value. If dynamic filtering were used, you almost always have to clear the filters before inserting an object or making other changes to property values that didn't match the current criteria.

Filtering Objects Versus Filtering Properties

In Architect/Requirements, there are two types of filtering: object filtering and property filtering. Although they both reduce the number of rows showing in a view, each one is unique in its use and purpose. It is important to understand the difference between the two so that there is no confusion about which menu choice to choose.

The other type of filtering in Architect/Requirements is called **Property Filters**. It is only available in the property tab (fixed or floating). It controls the number of properties and their values showing in the Properties view according to the group that a user selects. For example, there are choices to show all properties or only those that are editable, read only, user defined, system, or properties associated with a saved view. For more information about filtering properties in the properties tab, see [Filtering Property Tables](#).

Showing and Hiding Filters

Because filters can either be visible or invisible, the **View** menu has an option to **Toggle Column Filters** from their current state to the opposite state. If the filters are not showing, using **Toggle Column Filters** turns them on and if they are off, using the command a second time displays them.



Hiding filters does not clear the filters or the data. The previous filter values are still present and can be seen again if you show the filters.

This menu command attempts to work for the currently selected hierarchical view. It does not work on a standard, flat table view such as the Properties tab. If the currently selected view does not allow filtering, a warning appears. Filtering is probably most useful in the **Content view** and the **Search Results View**.



The **Navigation Tree** is neither a hierarchical nor a flat table; it is a tree structure. It is used to navigate a project's folder structure and controls what shows up in the content view. As a matter of convenience, if you use the navigation tree to navigate what you want to appear in the Content area and then choose **Toggle Column Filters** while the **Navigation Tree** is the currently selected view, the application toggles the column filters for you in the content area.



The Architect/Requirements varies the font used in the content table to indicate the state of the filtered data. Italics and color variations are used for this purpose.

When a filter is applied and an object matches the criteria, the entire parent hierarchy for that object is displayed even if all the parent objects do not match the criteria. The parent objects, which are displayed to provide a context to the matching objects, have the font in maroon. The maroon color indicates that the row does not match the filter criteria. However, it indicates that the row is a parent of an object that does match the filter criteria.

When using the filters in Architect/Requirements, italics are used to indicate that the data that is showing may not match the filter criteria. This could occur if data is added after the data was last filtered. It may also occur if data is filtered on a column and then the column is removed without first removing the filter. Rows in italics are in an undetermined state, meaning it is unknown whether they match the filter criteria. When the content table is filtered and any rows are added to the table, those rows show up with an italicized font to indicate that they were added since the data was last filtered. Because the filtering in Architect/Requirements is static, not dynamic, the new data may or may not match the specified filter criteria. It must be filtered again in order for its correct state to be determined. Adding a row to already filtered data can be accomplished in many ways, including, but not limited to, creating a new object, pasting in an object, or undoing a delete.

Filter Toolbar

Filters appear as another small table on top of the table in the currently selected hierarchical view. The first column on the left of the filter table is the toolbar that is used to execute filter commands. It contains five icons:



Filters the data according to the currently selected filter choices.



Unfilters the data. This option keeps any filter values and operators that you selected, but shows the data as though it was unfiltered. This is very useful when you want to momentarily see an object that was filtered out of the view and then quickly return to the filtered state.



Clears all of the filters and restores the data to an unfiltered state.



Shows the row of filter operators.



Hides the filters.

Filter Choices

To the right of the filter toolbar there is a filter for each column in the hierarchical view. Each filter is an editable dropdown selection box that allows you select from a list of values or type in a value.

The first four entries in each filter are always the same. These constants are shown in all capital letters and are described below.

ALL (Default value) If this value is selected, all of the rows are considered to have met the filter criteria. In other words, no data is hidden as a result of this filter value; any value (including blanks) is accepted.

BLANKS Only shows rows whose data for that column property is empty. In other words, it does not contain any characters.



Blanks do not equate to spaces. If you type only spaces in a field, the field contain characters—it is not empty—and it will not match to **BLANKS**.

NON-BLANKS Only shows rows whose data for that column property is not empty. In other words, if any characters are in the cell for that row, the object matches that filter.

----- (Separator). A dashed line serves as a separator in the list between the fixed, system-defined values and the dynamic values that appear based on the data that is available for that column.

The rest of the values, after the fixed system-defined ones, are dynamically derived based on the data. For all property types except for multichoice properties, each unique value within that column is listed.

However, multichoice properties are an exception. There are many different combinations of choices that could show up. For example, if the choices are “a”, “b” and “c”, the value of “a” could appear by itself or as “a, b”, “a, c” or even “a, b, c”. Instead, each choice from the property definition is listed only once.



The values that appear in a multichoice and a single-choice filter dropdown box are derived differently:

- Single-choice properties behave just like any other property; only the unique values that are displayed in a column for the currently showing rows are shown in the box. This may prevent you from seeing every possible value because filtering on values that are not present in the currently displayed data always results in all rows being excluded; hence, no data appear.
- Multichoice properties are unique. Because you can choose none, one, or more than one choice for each object, a variety of combinations can exist. Therefore, you need is a single entry for each possible choice, not the combination of choices appearing in the table cells.



As soon as a filter choice is changed, the system filters the data immediately. This saves you from having to make a filter choice and then click the **Filter** icon.

Filter Operators

The filter operators are located above the filter choices. To minimize the amount of space used by filters, and to maximize the amount of space available for data to show, these operators are hidden by default even when filters are first displayed. Also, the most commonly used values for each filter type are selected by default. As long as you want to match on a data value equaling a chosen filter value, the default operator of **Contains** for a multichoice field and **Equals (=)** for all other fields work without your having to display the filter operators. However, if you want to use advanced filtering operators such as **Not Equals (!=)**, **Starts With**, **Ends With**, and **Contains**, you can display the filter operators. As described in [Filter Toolbar](#), you can show the filter operators by clicking the fourth icon in the toolbar that contains the ellipses.



There is an exception to the general rule that the filter operators have several choices such as **Starts With**, **Ends With**, etc. In Architect/Requirements, there are certain system count properties that always display in a hierarchical table column with their corresponding icon. For example, a paper clip icon indicates that there are attachments for this object. And, an interlocking chain link icon indicates that there are traceability links for an object. In these cases, the only operator value that makes sense is **Equals (=)**. Therefore, it is the only available operator. Options like **Contains**, **Starts With** and **Ends With** do not make sense with a binary choice of a cell either having and or not having an icon.

In the overview of this section, it was mentioned that Architect/Requirements uses static filtering. This means that every user action does not automatically cause the data to be filtered. This is true with the operators. Selecting an operator does not filter the data. It selects the operator that will be used whenever a filter value is changed or the **Filter** icon is selected. Since there are always really two parts to filtering (operator and filter value) that must be specified, only one of these (changing the filter value) immediately begins the process of filtering the data.

Filtering Data

To filter the data, show the filters and optionally the filter operators. Select a filter operator if you do not want to use the default operator and then select a filter value. Only those rows that match the specified filter criteria are displayed.

Filter value dropdown boxes are editable. This means that instead of just selecting a value, you can type in a value. This is especially helpful when using operators such as **Contains**, **Starts With**, and **Ends With**. For example, a column filter may list choices of **somewhere**, **awesome**, **somehow**, and many others, but you would like to filter on all rows where the value for that column starts with **some**. To do this, first select the **Starts With** operator. Then double-click the filter dropdown box, type the word **some**, and press the enter key.

Saved Views

In Architect/Requirements, named views can be saved in the database and used to control which properties appear in the content area as well as the **Search Results** window and **Properties** tab. Filters and their state are also saved with the view information; for example, whether filters are showing, operators are showing, and the filter and operator values are all saved with the named view. These settings are restored when you choose the **Named View**, resulting in the data being filtered according to the saved filter values. This filtering makes the data conform more to the exact state that a **Named View** was saved with. For more information, see *Customizing Views of Property Columns*.

Similarly, filter settings are also saved with local view settings. For example, if you are in a project where you have filters visible and you switch to a project where filters were not displayed when you last visited that project, the filters are hidden for that project. When you switch back to the project where filters were showing, the filters return. Filters are not automatically applied to the view content; the settings of previous filter values correspond to the rules of when columns settings are restored.

When there are more than 28 saved views (in addition to **System Settings** and **Default View**) for the current user in the selected project, the user can select a view by choosing the **Default View** or the **Select a Saved View** drop down options. The user can also right-click a folder and choose **Select View** to access the options.

If the user chooses the **Default View** option, then the default view column settings are applied. If the user chooses **Select a Saved View**, then a dialog window is displayed that allows the user to choose a view from a drop down box of the saved views. The user can choose a saved view and the click **OK** to apply the settings of the selected view or click **Cancel** to close the dialog window without applying any settings.

Printing a Selected Panel

You can print a selected panel, such as the navigation tree, the content pane, or the notebook pane, in the Architect/Requirements. You can choose **File**→**Print** option or click the **Print** button on the Architect/Requirements toolbar.



- For the Search module, the print command prints the body of that module as it appears on the screen.
- For all other panels, including Search Results, the entire content of the active panel is printed, which may be more than the currently visible portion. The print command prints everything that the active panel's scroll bars would allow you to reach.
- Each printed panel is scaled to fit one page in width. However, the print command prints as many pages as necessary to show all the content vertically.
- If the active panel is wide, the one-page horizontal scaling may make the content too small to read. Try reducing the number of columns in the panel, or reduce the column widths. Choosing landscape mode gives a wider effective paper width and improves the scaling marginally. There is no option to print across multiple pages horizontally.
- When you are printing for the first time in a session, a message window may appear to indicate a delay in opening the Print dialog window. The Print dialog window opens quickly when you try to print again in the same session.
- The print commands in Architect/Requirements are built on Java libraries that are relatively new. Some options in the Print dialog window may not work as expected for

all printers. They include options such as color versus monochrome, print resolution, and single versus duplex printing. These libraries are expected to improve over time.

Viewing Architect/Requirements System Properties

Architect/Requirements Customer Support may request information about your system to answer your questions or resolve problems. Obtain this information from the System Information dialog window, which displays properties for:

- The Architect/Requirements software.
- The required Java software.
- The computer on which the client is installed.

You can rearrange and resize the columns in the table, and you can sort the columns in ascending or descending order. In addition, you can do the following:

- Choose a group of related properties to display.
- Clear one or more rows to reduce the number of properties.
- Save the system information in a text file.
- Export the system information to Microsoft Excel.

To view Architect/Requirements system properties:

Pull down the **Tools** menu and choose **System Information** to display the System Information dialog window, and then click the **System Properties** tab.

- To view only Java properties, click **Show Java**.
- To view only those properties that are not Java properties, click **Show non-Java**.
- To view all properties, click **Show All**.
- To clear one row, select the row, and then click **Hide Node(s)**.
- To clear a group of adjacent rows:
 - . Select the first row in the group, hold down the shift key or the control key, and then select the last row.
 - . Click **Hide Node(s)**.
- To save the currently displayed properties in a text file:

. Click **Save As** to display the Save Properties dialog window.

The **Save in** field displays the currently selected drive or folder. You can select another drive or folder by clicking the button to the right.

The **File name** field displays the default name of the text file. You can enter another file name in this field.

- . Click **Save**.

The Save Properties dialog window closes, and Architect/Requirements saves the file with the specified name in the selected location. You can print your system information from this file, and you can send the file to E-mail recipients as an attachment.

- To export the current table to Microsoft Excel, click **Export to Excel**.

Excel opens a read-only file (**.html**), in which the worksheet's columns, rows, and cell contents match those in the System Information dialog window.

You can use all of Excel's viewing, printing, and navigation features to analyze your system information. From within this file, you can send the table to E-mail recipients in the body of a message. This file is temporary and is deleted from your computer when you exit Architect/Requirements.

You can create a permanent file by pulling down the Excel **File** menu and choosing **Save As**, and then assigning the file name, file type, and location. This file remains on your computer when you exit Architect/Requirements.

Working With the Client Log File

In the System Information dialog window, the **Client Log** tab displays the contents of the client log file. This text file (.txt) is stored on your computer, in the location shown in the **Log Path** field. Architect/Requirements maintains a backup log file in the location shown in the **Backup Path** field.



Use this tab to provide information to Architect/Requirements Customer Support in resolving problems with the client. System performance is diminished if you use this tab during normal operations.



The **Client Log** tab shows the log file contents as of the time when you displayed the System Information dialog window. This data is not updated dynamically because of performance considerations. However, the data is updated when you do certain actions in the tab.

- To update the log file with the latest data, click **Refresh**.
- To copy the log path or the backup path to the Windows clipboard, click the **Copy Path** button to the right of the corresponding field.



You can use the path to locate the file in other applications. In Microsoft Outlook, for example, you can attach the file to a message by pasting the path into the **File name** field and clicking **Insert**.

- To open the log file in Microsoft Notepad, click **Open in Notepad**.

Architect/Requirements refreshes the log file and copies it to a temporary file in the same directory as the log file. Then, the temporary file opens in Notepad.



You can do the following through standard Notepad features:

- o Find and replace text.
- o Edit the log file.
- o Print the log file.
- o Save the log file with a different name or in a different location.

The **temp_** prefix indicates the temporary file. This file remains on your computer and is overwritten each time you click **Open in Notepad**.



Clicking **Open in Notepad** while the file is already open generates multiple Notepad windows. Each window shows the latest data as of the time the window was generated.

You can save or copy the temporary file to other locations. You can also delete this file from the log file directory.

- To copy any portion of the log file to the Windows clipboard, select the text in the scrolling pane and press control-C.
- To add a line of text, enter the text in the **Append Text Line** field and click the **Append Text** button.

The data is refreshed and the text is added as the last line. A date and time stamp precedes the text.



The text you enter in the field remains there until you delete it. Therefore, you can reuse text elements for additional lines. For example, you may enter leading characters to make your entries unique. Later, you can open the log file in Notepad and search for the characters to find the lines you added.

- To add a blank line, place the cursor in the **Append Text Line** field and click the **Append Text** button.

The log file is refreshed and the blank line is added at the bottom.

- To clear the current data and start a new log:



Before you clear the current data, you can keep a copy by renaming the log file, opening it Notepad and saving it with another name, or copying it to another location. The log file location is shown in the **Log Path** field.

- Click **Empty Log**.

A confirmation message asks if you are sure you want to empty the client log.

- To continue, click **Yes**.

Architect/Requirements copies the current data to the backup log file. In the **Client Log** tab and in the log file, all previous data is removed and new data is added.



For better focus on your client problem, empty the log file before you run your test case. By starting a new log file that contains only relevant data, you help Customer Support isolate the problem.

Setting Client Debugging Options

In the System Information dialog window, the **Logging Options** tab provides options for writing certain debugging information to the client log. Under the **Logging Option** heading, each successive option incorporates the preceding option and writes additional information.



Use this tab to provide information to Architect/Requirements Customer Support in resolving problems with the client. System performance is diminished if you use this tab during normal operations.



The System Information dialog window is not a modal dialog window, and you can leave it open while debugging a problem. This allows you to view the log and set debugging options without repeatedly opening and closing the dialog window.

1. Select one of the following options:

Option	Description
Off	Deactivates debug logging and returns to normal logging. For each server call, standard information is written to the log file without extra debugging information.
On	For development purposes. Allows developers to put in special code that is run only if the basic debug flag is turned on. Has no effect in a normal production environment.
Response Time	Adds server call response time. For each server call, writes the last method name, the response time, the calling command, and the first parameter.
Short	Adds all parameters. For each server call, writes the method name, the response time, the calling command, and all parameters.
Trace	Adds a full trace of method names. For each server call, writes one trace line and the information for the Short option.
Long	Adds formatting to separate each call. For each server call, writes a blank line, a header line, the information for the Trace option, and a footer line.

2. To activate the selected option, do one of the following:

- Click **OK** to activate the option and close the dialog window.
- Click **Apply** to activate the option and leave the dialog window open.



On the first server call after debug logging is activated, a warning symbol is displayed in the status bar at the bottom of the client window. To the right of the symbol, a message shows which option is activated.

The symbol and message cannot be disabled while debug logging is activated. They are removed when debug logging is deactivated.

You can also do the following in the **Logging Options** tab:

- To add a line of text, enter the text in the **Append Text Line** field and click the **Append Text** button.

The data is refreshed and the text is added as the last line. A date and time stamp precedes the text.



The text you enter in the field remains there until you delete it. Therefore, you can reuse text elements for additional lines. For example, you may enter leading characters to make your entries unique. Later, you can open the log file in Notepad and search for the characters to find the lines you added.



To keep a constant line available, you can use this **Append Text Line** field in combination with the same field in the **Client Log** tab. For example, you can use one field to add unique characters, such as a separator line containing dashes, and use the other field to repeatedly change the text you want to add to the log. When you are finished entering different text lines, return to the other tab where the separator line remains in that field ready to be added after the text.

- To add a blank line, place the cursor in the **Append Text Line** field and click the **Append Text** button.
- To clear the current data and start a new log:



Before you clear the current data, you can keep a copy by renaming the log file, opening it in Notepad and saving it with another name, or copying it to another location. The log file location is shown in the **Log Path** field in the **Client Log** tab.

- Click **Empty Log**.

A confirmation message asks if you are sure you want to empty the client log.

- To continue, click **Yes**.

Architect/Requirements copies the current data to the backup log file. In the **Client Log** tab and in the log file, all previous data is removed and new data is added.



For better focus on your client problem, empty the log file before you run your test case. By starting a new log file that contains only relevant data, you help Customer Support isolate the problem.

Changing Your Architect/Requirements Password

Your Architect/Requirements password can contain any combination of uppercase and lowercase letters, numerals, and symbols. There are no reserved characters, nor is there a maximum number of characters.



By default, there is no minimum number of characters, nor are any alphanumeric and numeric characters required. However, your Architect/Requirements system administrator may set restrictions for passwords at your site. If you have questions about changing your password, consult your Architect/Requirements system administrator.

To change your Architect/Requirements password:

1.

Pull down the **Tools** menu and choose **Change Password**.

The Change Password dialog window is displayed.

2. In the **Current Password** field, enter your current password.

3. In the **New Password** and **Verify New Password** fields, enter your new password.

4. Click **OK**.

The dialog window closes, and a confirmation message appears. Use the new password the next time you start Architect/Requirements.

Exiting Architect/Requirements

If you changed your Architect/Requirements password during the current session, your new password becomes effective when you exit Architect/Requirements.



Before you end the current session, close all temporary Microsoft Word or Excel files that you opened from Architect/Requirements. Otherwise, those files remain on your computer when you exit Architect/Requirements.

To exit Architect/Requirements:

1. Pull down the **File** menu and choose **Exit**.

A message appears, asking if you want to exit the application.

2. Click **Yes**.

The Architect/Requirements main window closes.

Procedure Notes

Step 1: You can also press control-Q or Alt-F4. Or, click the **Close** button in the upper right corner of the main window.

Chapter 4: Maintaining a Project

This chapter contains an overview of Architect/Requirements projects and their organization, including instructions for working with the objects in a project.

Overview of Projects

In Architect/Requirements, projects reside at the highest level of organization and are superior to all other objects. Each project defines a logical boundary for folders, requirements, building blocks, groups, notes, and trace links. These objects reside within a project at levels determined by the individual object types.

Directly below the project node, the primary level is reserved for folders, in which the other types of objects are organized into logical categories. Folders can be used, for example, to store requirements by source (such as customer or mil-spec), by related discipline (such as testability or reliability), or by technology (such as hardware or software).

As folders contain files and other folders in Microsoft Windows, folders contain requirements, building blocks, groups, and other folders in Architect/Requirements. A given folder can contain any number and combination of those objects. Consequently, requirements and building blocks can be stored in folders that are nested within other folders at progressively lower levels in the project.

A more document-centric method is to structure requirements and building blocks in a hierarchy of parent, child, and sibling relationships within a single folder. For each requirement and building block, the **Number** property value corresponds to that object's level in the hierarchy, similar to numbered paragraphs in Microsoft Word's outline view. This structured approach should be used if requirements and building blocks are to be exported to Word later in the process. However, folders containing those objects should not be nested within other folders.

Although both methods of organization can be used together, it is more effective to choose the method that is better suited to the particular requirements management process and to use that method exclusively. Consideration should be given also to the number of objects that exist at any level, at the top level in a folder, for example, or below parent objects in a hierarchy. Architect/Requirements imposes no limitations, but there is an impact on usability. If a folder contains several hundreds or thousands of objects, it is difficult for the user to locate the objects of immediate interest. There is also a performance consideration in populating a view with a large number of objects, which naturally takes longer than with a smaller number.

Objects can be copied from one location to another within a project and from one project to another. Trace links can be created between objects in different projects. However, objects can be moved only within the project where they reside.

Opening a Project

The navigation tree contains a node for each project to which you have access. Project nodes occupy the next level below the root node in the navigation tree. For more information, see [Navigation Tree](#) in chapter 3, *Using the Architect/Requirements Main Window*.

Only folders can occupy the primary level in a project hierarchy, the level directly below the project node. Those folders are displayed in the hierarchical content table when you open a project. For more information, see [Hierarchical Content Table](#) in chapter 3, *Using the Architect/Requirements Main Window*.

To open a project:

In the navigation tree, select the project node.

The content table displays the folders at the project's primary level.

You can display these folders in the navigation tree by double-clicking the project node, by clicking the plus sign (+) to the left, or by right-clicking the project node and choosing **Expand** from the pop-up menu.



If you click a plus sign and the folder does not expand, it may contain subfolders to which you do not have **Read** access. If you have questions about folder access, consult your project administrator.

Opening a Folder

A project can contain any number of folders, which can be nested to progressively subordinate levels. You can open a folder from the navigation tree or the hierarchical content table. For more information, see [Navigation Tree](#) and [Hierarchical Content Table](#) in chapter 3, *Using the Architect/Requirements Main Window*.

Each folder selected in the navigation tree can have a unique view of property columns in the content table. These views let you easily work with the properties. For more information, see [Customizing Views of Property Columns](#) in chapter 9, *Working With Object Properties*.

To open a folder from the navigation tree:

1.

Click the plus sign (+) to the left of the project node to display the folders at the project's primary level.

If necessary, click the plus signs for these folders to display lower level folders.



If you click a plus sign and the folder does not expand, it may contain subfolders to which you do not have **Read** access. If you have questions about folder access, consult your project administrator.

2. Select the folder to display the top level objects in the content table.

Procedure Notes

Step 1: You can also right-click the project node or folder and choose **Expand** from the pop-up menu. Or, double-click the project node or folder.

To open a folder from the hierarchical content table:

In the **Name** column, select the folder, and then do one of the following:

- To display only objects in the selected folder, pull down the **File** menu and choose **Open**, or right-click the folder and choose **Open** from the pop-up menu.

The content table displays the objects at the folder's top level. In the navigation tree, the folder is highlighted, and it is expanded if it contains lower level folders.

You can also display the top level objects by clicking the plus sign (+) to the left of the folder or by double-clicking the folder's object type indicator.

- To display objects in the selected folder and maintain the current view of other folders, do one of the following:
 - o For all objects in the folder, pull down the **View** menu and choose **Expand All**, or right-click the folder and choose **Expand All** from the pop-up menu.
The content table displays all objects at each level of the folder hierarchy. Any other folders remain as previously displayed.
 - o For only the top level objects in the folder, click the plus sign (+) to the left of the folder.
The content table displays the objects at the top level of the folder hierarchy. Any other folders remain as previously displayed.



If you click a plus sign and the folder does not expand, it may contain subfolders to which you do not have **Read** access. If you have questions about folder access, consult your project administrator.

Creating a Folder

You can create a folder at the primary level of a project, directly below the project node, or at any lower level as a child within an existing folder. Folders can also be created in live Excel. For more information, see [Creating Objects in Live Excel](#) in chapter 9, *Working With Object Properties*.

The new folder receives certain system-defined properties, including the **Subtype** property. You can assign the default subtype (**Folder**), a system-defined folder subtype (**Document**), or a user-defined subtype. The folder may also receive user-defined properties.

The subtype and all other editable properties can be changed after the folder's creation. In addition, you can copy the new folder to other locations, link it to other objects, and attach notes. For more information, see appendix [System-Defined Properties in the Systems Engineering and Requirements Management Module](#); chapter 9, *Working With Object Properties*; [Copying Objects](#), later in this chapter; chapter 7, [Showing Object Relationships With Trace Links](#); and [Attaching a Note to an Object](#) in chapter 8, *Recording Supplementary Information With Notes*.



You must have **Modify** permission for the project or the parent folder in which you intend to create the folder.

To create a folder at the primary level of a project:

1. Select the project node in the navigation tree, or select an existing primary folder in the hierarchical content table.
2. Assign the subtype by doing one of the following:
 - For the **Folder** subtype, pull down the **File** menu and choose the **New→Folder** options, or click the **Create New Folder** button on the toolbar.
 - For the **Document** subtype, pull down the **File** menu and choose the **New→Document** options.
 - For a user-defined subtype, pull down the **File** menu and choose the **New→Subtype** options to display the Select Subtype dialog window. To see the user-defined subtypes, click the plus sign (+) to the left of **Folder**, and click any lower level plus signs. Select a subtype and click **OK**.

The navigation tree displays the new folder in alphabetical order of the default name. The content table displays the folder in the last position at the primary level, with the default name in an open text field.

3. Enter the folder name in the text field, and then press the enter key.



Using the **View** field, you can set the folder to display a unique set of columns in the content table. For more information, see [Customizing Views of Property Columns](#) in chapter 9, *Working With Object Properties*.

Procedure Notes

Step 2: You can also right-click the project node and choose the options from the pop-up menu. Or, press control-L for the **Folder** subtype, control-D for the **Document** subtype, or control-U for a user-defined subtype.

Step 2: To reverse this action, you can pull down the **Edit** menu and choose **Undo New Folder**, **Undo New Document**, or **Undo Create Subtype**; click the **Undo** button on the toolbar; or press control-Z.

Step 3: You can reverse this action by pulling down the **Edit** menu and choosing **Undo Rename**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

To create a folder within a folder in the navigation tree:

1. Select the parent folder.

To see lower level folders, click the plus signs. Or, right-click a folder with a plus sign and choose **Expand** from the pop-up menu.

2. Assign the subtype to the child folder by pulling down the **File** menu and choosing one of the following:

- Choose the **New→Folder** or **New→Child** options to allow the child to inherit the parent subtype.

You can also right-click the parent and choose the **New→Folder** or **New→Child** options from the pop-up menu. Or, click the **Create New Folder** or **Create New Child** button on the toolbar. Or, press control-L or control-K.

- Choose the **New→Document** options to:
 - Assign the **Document** subtype when the parent subtype is **Folder** or a user-defined folder subtype.
 - Allow the child to inherit the parent subtype when the parent is **Document** or a user-defined document subtype.

You can also right-click the parent and choose the **New→Document** options from the pop-up menu. Or, press control-D.

- Choose the **New→Subtype** options to display the Select Subtype dialog window and assign a subtype to the new folder specifically:
 - Click the plus sign (+) to the left of **Folder** to display the **Document** subtype and user-defined folder subtypes.

Plus signs may be shown also for these subtypes. Click the plus signs to display additional subtypes at lower levels.
 - Select a subtype, and then click **OK** to close the dialog window.

To display the dialog window, you can also right-click the parent and choose the **New→Subtype** options from the pop-up menu. Or, press control-U.

In the content table, the new folder is displayed as the last object at the top level of the parent folder, with the default name in an open text field.

3. Enter the folder name in the text field, and then press the enter key.



Using the **View** field, you can set the folder to display a unique set of columns in the content table. For more information, see [Customizing Views of Property Columns](#) in chapter 9, *Working With Object Properties*.

Procedure Notes

Step 2: You can reverse this action by pulling down the **Edit** menu and choosing **Undo New Folder**, **Undo New Document**, **Undo New Subtype**, or **Undo New Child**. You can also click the **Undo** button on the toolbar or press control-Z.

Step 3: You can reverse this action by pulling down the **Edit** menu and choosing **Undo Rename**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

To create a folder within a folder in the hierarchical content table:

1. Do one of the following:

- Select the parent folder, pull down the **File** menu, and choose the **New→Child** options. Or, right-click the parent and choose the **New→Child** options from the pop-up menu. You can also click the **Create New Child** button on the toolbar, or press control-K.

The new folder inherits the subtype of the parent and is displayed as the last object at the next lower level, with the default name in an open text field.

- Within the parent folder, select an existing folder as a sibling of the new folder, and then pull down the **File** menu and choose one of the following:
 - Choose the **New→Folder** options to assign the subtype of the sibling to the new folder. Or, right-click the sibling and choose the **New→Folder** options from the pop-up menu. You can also click the **Create New Folder** button on the toolbar, or press control-L.
 - Choose the **New→Document** options to assign the **Document** subtype to the new folder when the sibling subtype is **Folder** or a user-defined folder subtype. When the sibling subtype is **Document** or a user-defined document subtype, that subtype is assigned to the new folder. You can also right-click the sibling and choose the **New→Document** options from the pop-up menu, or press control-D.
 -

Choose the **New→Subtype** options to display the Select Subtype dialog window and assign a subtype to the new folder specifically:

- Click the plus sign (+) to the left of **Folder** to display the **Document** subtype and user-defined folder subtypes.

You can click any additional plus signs to display more subtypes.

- Select a subtype, and then click **OK** to close the dialog window.

To display the dialog window, you can also right-click the sibling and choose the **New→Subtype** options from the pop-up menu. Or, press control-U.

The new folder is displayed as the last object at the level of the sibling, with the default name in an open text field.

2. Enter the folder name in the text field, and then press the enter key.



Using the **View** field, you can set the folder to display a unique set of columns in the content table. For more information, see [Customizing Views of Property Columns](#) in chapter 9, *Working With Object Properties*.

Procedure Notes

Step 1: You can reverse this action by pulling down the **Edit** menu and choosing **Undo New Folder**, **Undo New Document**, **Undo New Subtype**, or **Undo New Child**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Step 2: You can reverse this action by pulling down the **Edit** menu and choosing **Undo Rename**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Copying Objects

Folders, requirements, building blocks, groups, and notes can be duplicated as new objects in other locations. The destinations can be within the same Architect/Requirements project or in a different project, subject to the following conditions:

- Folders can be copied to the primary level and to the top levels of other folders. They cannot be copied to requirements, building blocks, notes, diagrams, or spreadsheets.
- Requirements can be copied to the top levels of folders and to other requirements as children. They cannot be copied to the project node or to building blocks, notes, diagrams, or spreadsheets. Each copied requirement receives a new **ROIN** property value.
- Building blocks can be copied to the top levels of folders and to other building blocks as children. They cannot be copied to the project node or to requirements, notes, diagrams, or spreadsheets.
- Groups can be copied to the top levels of folders. They cannot be copied to the project node or to requirements, building blocks, notes, diagrams, or spreadsheets.
- Notes can be copied to the project node, folders, requirements, building blocks, trace links, and to objects in the **Attachments** tab and window, including other notes.
- Diagrams can be copied to folders, requirements, and building blocks. They cannot be copied to the project node, to notes or spreadsheets, or to other diagrams.
- Spreadsheets can be copied to folders, requirements, and building blocks. They cannot be copied to the project node, to notes or diagrams, or to other spreadsheets.



Objects that are copied to groups are not duplicated as new objects. Instead, they are associated as group *members* through references to their present locations. For more information, see [Using Groups to Maintain Objects](#), later in this chapter.

When you copy certain object types from one project to another, the way you select the objects to copy from the originating project determines the behavior of the objects in the destination project:

- When the selection includes a shortcut and its referenced object, both objects are duplicated and the reference is within the destination project. When the selection does not include the referenced object, the copied shortcut in the destination refers to the object in the originating project. For more information, see [Working With Shortcuts](#), later in this chapter.
- When the selection includes a group and any of its members, each selected member is duplicated and becomes a member of the copied group in the destination project. When the selection does not include at least one member, the copied group is empty in the destination project.

- Trace links and connections between selected objects are preserved in the destination project. Trace links and connections to objects outside the selection are not copied with the selected objects.
- The **Version Type** and **Baseline** properties of copied versions are reset to their default values in the destination project. For versioned objects selected in the hierarchical content table, only the version specified in the effectivity field is copied to the destination project.
- Subtypes of selected objects are preserved in the destination project. A new subtype is automatically created in the destination if a copied subtype name does not exactly match an existing subtype name in the destination.
- Each selected spreadsheet is copied in its current state, for example, synchronized with the database through live Excel. However, the objects represented in the spreadsheet reside in the originating project.



Diagrams cannot be copied to other projects. If diagrams are included in the selection, they are ignored.

The hierarchical content table displays a plus sign (+) for each object that has one or more members at lower levels. In copying such objects, consider the following:

- When you copy a folder, you also copy all of the objects that it contains, including other folders. These objects may themselves have members, all of which are copied with the folder. The folder takes the last position at the next level below the destination object, maintaining its own hierarchy of objects. Its requirements and building blocks retain their **Number** property values.
- When you copy a requirement or a building block, you also copy all of its direct children and lower level descendants. The copied object takes the last position at the next level below the destination object, receiving a new **Number** property value for that level and position. The parent and child relationships are preserved, and each of these objects receives a new **Number** property value for the level and position below its parent.



- You must have **Modify** permission for each intended destination.
- To copy building blocks, TRAMs, and diagrams, and to copy a folder that contains those objects, you must have **Architect** privilege for the project.
- Objects cannot be copied from or to your Recycle Bin.

To copy objects:

1. Select each object that you want to copy.

In the navigation tree, you can select only one folder.

In the hierarchical content table or the **Attachments, Where Used**, or **Versions** tab or floating window, you can select nonadjacent objects by holding down the control key while you click the objects. To select adjoining objects, click the first object, hold down the shift key, and click the last object.

2. Pull down the **Edit** menu and choose **Copy**.

For a selection in the navigation tree, the content table, or the **Attachments**, **Links**, and **Versions** tabs or windows, you can also hold down the left mouse button on the selection, hold down the control key, and drop the objects on the destination. This action is subject to the conditions in step 3.

3. Set the destination by doing one of the following:

- To copy selected folders to the project's primary level, select the project node in the navigation tree.

The selection cannot include requirements, building blocks, groups, or notes.

- To copy the objects to a folder's top level, select the folder in the navigation tree, the hierarchical content table, or the **Links** tab or window.
- To copy the objects to a requirement or a building block, select the parent in the hierarchical content table or the **Links** or **Versions** tab or window.

For a requirement, the selection must consist solely of requirements. For a building block, the selection must consist solely of building blocks.

- To copy a selection that consists solely of notes:
 - If the destination is a folder, select it in the navigation tree, the hierarchical content table, or the **Links** tab or window.
 - If the destination is a requirement or a building block, select it in the hierarchical content table or the **Links** or **Versions** tab or window.

4. Pull down the **Edit** menu and choose **Paste**.

To duplicate these objects in additional destinations, repeat steps 3 and 4.

Procedure Notes

Step 2: You can also right-click the selection and choose **Copy** from the pop-up menu. Or, click the **Copy** button on the toolbar or press control-C.

Step 4: You can also right-click the destination and choose **Paste** from the pop-up menu. Or, click the **Paste** button on the toolbar or press control-V. To reverse this action, pull down the **Edit** menu and choose **Undo Copy**, click the **Undo** button on the toolbar, or press control-Z.

Copying Object URLs

Architect/Requirements assigns a Uniform Resource Locator (URL) to each folder, requirement, building block, group, note, and diagram. For one or more selected objects, including group members, object shortcuts, and linked objects, you can copy the URLs to the Clipboard. You can then insert the URLs in Windows programs that support hyperlinks, such as Microsoft Outlook and Microsoft Internet Explorer.

After insertion, URLs can be used to start a new Architect/Requirements session, and to navigate to objects in the main window. For example:

- In Outlook, you can paste the URLs into the body of a message and send it to recipients such as colleagues and customers. Recipients can start Architect/Requirements and navigate to an object by clicking its URL in the message.
- In Internet Explorer, you can paste a URL into the **Address** bar, and then click the **Go** button to navigate to the object in Architect/Requirements.

Also, E-mail recipients can copy a URL from a message body to the Internet Explorer **Address** bar, and then click the **Go** button to start Architect/Requirements and navigate to the object.



- The destination program must support hyperlinks.
- To support navigation, Architect/Requirements must be configured for object linking. If you have questions about navigating to objects, consult your Architect/Requirements system administrator.

To copy object URLs:

1. In the hierarchical content table, the **Attachments** tab or floating window, or the **Links** tab or window, select each object whose URL you want to copy.

You can select nonadjacent objects by holding down the control key while you click the objects. To select adjoining objects, click the first object, hold down the shift key, and click the last object.

If the destination program is Internet Explorer, select only one object.

2. Pull down the **Edit** menu and choose **Copy URL**→**Include Full Name** or **URL Only**. Choose **Include Full Name** to copy the URL link that includes the **Full Name** property, or choose **URL Only** to copy only the URL link.

For example:

Choosing the **Include Full Name** option for an object named **Requirement1** copies the following:

\ATK TBx\TBX Test Data\Folder\Requirement1 at

http://pnsvin07v01:7001/tcr/controller/ObjLauncher?wolf_objectid=1.0.24535

Choosing the **URL Only** option copies only the URL link as shown here:

http://pnsvin07v01:7001/tcr/controller/ObjLauncher?wolf_objectid=1.0.24535

The URLs are placed in the Clipboard for insertion in Windows programs.

Procedure Notes

Step 2: You can also right-click the selection and choose **Copy URL** from the pop-up menu.

Moving Objects

Objects in a project can be moved from their current locations to other locations. Destinations must be within the project and are subject to the following conditions:

- Folders can be moved to the project node and to the top levels of other folders. They cannot be moved to requirements, building blocks, groups, or notes.
- Requirements can be moved to the top levels of folders and to other requirements as children. They cannot be moved to the project node or to building blocks, groups, or notes. Each moved requirement retains its **ROIN** property value.
- Building blocks can be moved to the top levels of folders and to other building blocks as children. They cannot be moved to the project node or to requirements, groups, or notes.
- Groups can be moved only to the top levels of folders. They cannot be moved to the project node or to requirements, building blocks, notes, or other groups.
- Notes can be moved to the project node, folders, requirements, building blocks, trace links, and to objects in the **Attachments** tab and window, including other notes.



An object cannot be moved in the following conditions:

- If its parent is frozen.
- If the object exists under multiple versions of the parent object and moving it results in the object getting displayed under different parents in the user interface.

This is also applicable for promote and demote operations.

For example, a requirement **A** has a child requirement **B**. The requirement **A** is frozen and a new version **A1** is created. Moving the child requirement **B** to a new owner is not allowed because moving **B** results in multiple owners for **B**. If you then create a new version of **B**, the new **B1** will have only one owner and you can move it.

The hierarchical content table displays a plus sign (+) for each object that has one or more members at lower levels. In moving such objects, consider the following:

- When you move a folder, you also move all of the objects that it contains, including other folders. These objects may themselves have members, all of which are moved with the folder. The folder takes the last position at the next level below the destination object, maintaining its own hierarchy of objects. Its requirements and building blocks retain their **Number** property values.
- When you move a requirement or a building block, you also move all of its direct children and lower level descendants. The moved object takes the last position at the next level below the destination object, receiving a new **Number** property value for that level and position. The parent and child relationships are preserved, and each of these objects receives a new **Number** property value for the level and position below its parent.



- You must have **Modify** permission for the objects and the destination.

- To move building blocks, TRAMs, and diagrams, and to move a folder that contains those objects, you must have **Architect** privilege for the project.

To move objects:

1. Select each object that you want to move.

In the navigation tree, you can select only one folder.

In the content table, your Architect/Requirements Recycle Bin, or the **Attachments**, **Links**, and **Versions** tabs or windows, you can select multiple nonadjacent and adjoining objects.

2. Pull down the **Edit** menu and choose **Cut**.

For a selection in the navigation tree, the content table, your Recycle Bin, or the **Attachments**, **Links**, and **Versions** tabs or windows, you can also hold down the left mouse button on the selection and drop the objects on the destination. This action is subject to the conditions in step 3.

3. Set the destination by doing one of the following:

- To move selected folders to the level below the project node, select the project node in the navigation tree.

If other object types are also selected, only the folders are moved.

- To move the objects to a folder's top level, select the folder in the navigation tree, the hierarchical content table, or the **Links** tab or window.

- To move the objects to a requirement or a building block, select the parent in the hierarchical content table or the **Links** or **Versions** tab or window.

If object types other than the parent's are also selected, only the objects of the parent's type are moved.

- To move a selection that consists solely of notes:

- If the destination is a folder, select it in the navigation tree, the hierarchical content table, or the **Links** tab or window.

- If the destination is a requirement or a building block, select it in the hierarchical content table or the **Links** or **Versions** tab or window.



In the content table, you can move a requirement or a building block within its level by selecting only that object, pulling down the **Edit** menu, and choosing **Move Up** or **Move Down**. Or, right-click the object and choose **Move Up** or **Move Down** from the pop-up menu. You can also select only that object and click the **Move Up** or **Move Down** button on the toolbar.

4. Pull down the **Edit** menu and choose **Paste**.



If you cut and paste between two separate instances of Architect/Requirements, the objects are copied to the destination. They are not moved from the source window.



In certain operations, such as moving objects from the content window to the navigation window, it is possible to see an **Object Not Found** error even if the operation is successful. Architect/Requirements uses a locking model that provides improved

performance but occasionally displays inconsistent data. This locking model does not compromise database integrity. If the operation is successful and no additional errors are reported, it is safe to ignore the error message.

Procedure Notes

Step 2: You can also right-click the selection and choose **Cut** from the pop-up menu. Or, click the **Cut** button on the toolbar or press control-X.

Step 4: You can also right-click the destination and choose **Paste** from the pop-up menu. Or, click the **Paste** button on the toolbar or press control-V. To reverse this action, pull down the **Edit** menu and choose **Undo Move**, click the **Undo** button on the toolbar, or press control-Z.

Moving frozen objects

If you freeze a parent object and its child objects, you cannot move the objects. However, if you need to move the child objects to a new parent object, you need to create a new version of the parent object and the child objects together. If you create a new version of the parent object and the child objects separately, the child objects ends up with two different parent objects. If you create a new version a child object with a frozen parent, the new version of the child object is also a child of the frozen parent. However, if you version the parent object and child objects together, the new version of the child object retains its link with the new version of the parent object only.



If the frozen parent object and child objects are multiselected and a new version created, a new version of the parent object is created first. Hence, the child object is movable.

For example, if you have the following requirement structure:

```
R1
    R1.1
R2
```

Where R1 and R1.1 are frozen.

Following are the steps to move R1.1 to R2.

1. Create a new version of R1 and R1.1
2. Move R1.1 to R2.

In the example given above, if only the child requirement R1.1 is frozen and the parent requirement R1 is not frozen, you need to create a new version of R1.1 and move it to R2.



You can also multiselect R1 and R1.1 and version them at the same time. If you version R1.1 before you create a new version of R1 and try to move R1.1, the move fails.

Renaming an Object

After you copy or move an object, you may want to assign a name that matches the object's new context within the project. Also, a new name may be appropriate to reflect a change in an object's purpose or content.

You can rename an object selected in the following views:

- The navigation tree and the hierarchical content table. For more information, see [Navigation Tree](#) and [Hierarchical Content Table](#) in chapter 3, *Using the Architect/Requirements Main Window*.
- The **Attachments**, **Links**, **Where Used**, and **Versions** tabs and floating windows. For more information, see [Attachments Tab](#), [Links Tab](#), [Where Used Tab](#), [Versions Tab](#), and [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.



- You must have **Modify** permission for the object.
- To rename a building block, a TRAM, or a diagram, you must have **Architect** privilege for the project.

To rename an object:

1. Select the object, and then pull down the **File** menu and choose **Rename**.
A text field opens around the object's current name.
2. Enter the new name, and then press the enter key.
The object is displayed with its new name.

Procedure Notes

Step 1: To see lower level objects in a hierarchical column, click the plus signs. Or, select an object with a plus sign, pull down the **View** menu, and choose **Expand All**. You can also right-click the object and choose **Expand All** from the pop-up menu.

Step 1: You can also right-click the object and choose **Rename** from the pop-up menu. Or, press the F2 key. You can cancel this action and close the text field by pressing the escape key.

Step 2: You can reverse this action by pulling down the **Edit** menu and choosing **Undo Rename**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Using Groups to Maintain Objects

A group consists of references to existing objects that reside elsewhere within the project. Each reference associates one object as a *member* of the group, and each referenced object remains in its present location.

By creating a group and adding members, you can define a collection of references to objects that reside in various locations. Then you can maintain the member objects directly from the group, without switching to other folders or views.

The member objects can be folders, requirements, building blocks, notes, diagrams, and other groups. A group can reference any number and any combination of these object types, as well as their system-defined and user-defined subtypes. Trace links and their subtypes cannot belong to groups.

Maintenance actions that change an object's location must be done by selecting the object where it resides, not in a group to which it belongs. Such actions include moving, promoting, demoting, and deleting an object.

Other actions can be done from a group as if the member objects were selected in their actual locations. Depending on the object type, you can do the following:

- Copy objects and their URLs, rename objects, create shortcuts to member objects, send object data by E-mail, and export the view of members to Microsoft Excel. For more information, see [Copying Objects](#), [Copying Object URLs](#), and [Renaming an Object](#), earlier in this chapter, and

[Working With Shortcuts](#), [Sending Object Data by E-Mail](#), and [Exporting Objects to Microsoft Office Excel](#), later in this chapter.

- Open requirements, paragraphs, and notes in Microsoft Word to edit or view their content, and export those objects to Word. For more information, see [Exporting Objects to Microsoft Office Word](#), later in this chapter; [Entering and Changing Requirement Content in Microsoft Office Word](#) and [Viewing Requirement Content](#) in chapter 5, *Managing Requirements*; and [Editing a Note](#) and [Viewing Note Content](#) in chapter 8, *Recording Supplementary Information With Notes*.
- Open diagrams in Microsoft Office Visio to edit or view their content. For more information, see [Editing a Diagram](#) in chapter 6, *Constructing System Views With Building Blocks and Diagrams*.
- Create, view, and delete trace links, and navigate to linked objects. For more information, see [Creating Trace Links](#), [Linking to an Object in Another Teamcenter Product](#), [Viewing Object Relationships](#), [Deleting Trace Links for an Object](#), and [Navigating to a Linked Object](#) in chapter 7, *Showing Object Relationships With Trace Links*.
- Change the values of editable properties. For more information, see [Properties Tab](#) in chapter 3, *Using the Architect/Requirements Main Window*, and [Editing the Properties of a Selected Object](#), [Editing Properties in Table View Cells](#), or [Using the Live Excel Interface](#) in chapter 9, *Working With Object Properties*.

After creating a new group, you specify its object references by adding members. For an existing group, you can add new members and remove current members at any time. For more information, see [Creating a Group](#), [Adding Group Members](#), and [Removing Group Members](#), later in this chapter.

Creating a Group

A group can be created only at the top level of a folder. The containing folder can reside directly below the project node or in another folder. The new group occupies the last position at the top level and cannot be promoted or demoted.

The new group receives certain system-defined properties, including the **Subtype** property. In this procedure, you can assign the default subtype, **Group**, or a user-defined subtype created by your project administrator. With either subtype, the group may also receive user-defined properties. The subtype can be changed after the group's creation. For more information, see chapter 9, [Working With Object Properties](#) and appendix [System-Defined Properties in the Systems Engineering and Requirements Management Module](#).

Before or after adding members, you can copy the group to other folders, link it to other objects, and attach notes. For more information, see [Adding Group Members](#), later in this chapter; [Copying Objects](#), earlier in this chapter; chapter 7, [Showing Object Relationships With Trace Links](#); and [Attaching a Note to an Object](#) in chapter 8, *Recording Supplementary Information With Notes*.

Groups can also be created in live Excel. For more information, see [Creating Objects in Live Excel](#) in chapter 9, *Working With Object Properties*.



You must have **Modify** permission for the folder in which you intend to create the group.

To create a group:

1. In the navigation tree or the hierarchical content table, select the folder.

To see lower level objects in the content table, click the plus signs. Or, select an object with a plus sign, pull down the **View** menu, and choose **Expand All**. You can also right-click the object and choose **Expand All** from the pop-up menu.

2. To assign the subtype, do one of the following:
 - For the **Group** subtype, pull down the **File** menu and choose the **New→Group** options, or click the **Create New Group** button on the toolbar.
 - For a user-defined subtype:
 - Pull down the **File** menu and choose the **New→Subtype** options to display the Select Subtype dialog window.
 - Click the plus sign to the left of the **Group** subtype, select a lower level subtype, and then click **OK**.
3. Enter the group name in the open text field, and then press the enter key.

Procedure Notes

Step 2: You can also right-click the folder and choose the options from the pop-up menu. Or, press control-G for the **Group** subtype or control-U for a user-defined subtype. You can also select the **Group** subtype in the Select Subtype dialog window. To reverse this action, pull down the **Edit** menu and choose **Undo New Group** or **Undo Create Subtype**, click the **Undo** button on the toolbar, or press control-Z.

Step 3: You can reverse this action by pulling down the **Edit** menu and choosing **Undo Rename**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Adding Group Members

For a new or existing group, you can add folders, requirements, building blocks, notes, diagrams, and other groups as members. These object types, and their system-defined and user-defined subtypes, can be added in any combination and in any number. You cannot add trace links or their subtypes.

Objects added as members are not moved to the group but remain in their present locations. Furthermore, parent and child relationships are retained in hierarchies where the objects actually reside. Therefore, a group is not the parent of its members, though they occupy the level below the group in the hierarchical content table.

Any lower levels below a member do not imply a hierarchy within the group itself. Below a folder, a requirement, or a building block, such lower levels show the member object's children, which do not belong to the group but are shown for additional context. Levels below a member group show objects that belong only to that group.

An object can belong to many groups concurrently and can be maintained from any one. Changes to a member affect the object in its actual location and are reflected in all other groups to which it belongs. Those actions can be done also for objects that do not belong to the group but are shown at levels below a member. For more information, see [Using Groups to Maintain Objects](#), earlier in this chapter.

You can add objects that you select in the navigation tree, the hierarchical content table, the **Attachments** tab or window, or the **Where Used** tab or window. You can remove members at any time. For more information, see [Navigation Tree](#), [Hierarchical Content Table](#), [Attachments Tab](#), [Where Used Tab](#), or [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*, and [Removing Group Members](#), later in this chapter.

In the **Where Used** tab or window, you can see each group to which an object belongs. To see the number of members for a group, you can view its **Member Count** property in the **Properties** tab or

window. For more information, see [Properties Tab](#) in chapter 3, *Using the Architect/Requirements Main Window*.



You must have **Modify** permission for the group to which you intend to add members.

To add group members:

1. Select the objects that you want to add as members.

In the navigation tree, you can select only one folder.

In the hierarchical content table or the **Attachments** or **Where Used** tab or window, you can select nonadjacent objects by holding down the control key while you click the objects. To select adjoining objects, click the first object, hold down the shift key, and click the last object.



If the selection is in the hierarchical content table and if the group is currently visible, you can drop the selection on the group to add the members and complete this procedure.

2. Pull down the **Edit** menu and choose **Copy**.
3. Select the group within its folder, pull down the **Edit** menu, and choose **Paste**.

The content table displays the members at the next level below the group.

Procedure Notes

Step 2: You can also right-click the selection and choose **Copy** from the pop-up menu. Or, click the **Copy** button on the toolbar or press control-C.

Step 3: You can also right-click the group and choose **Paste** from the pop-up menu. Or, click the **Paste** button on the toolbar or press control-V. To reverse this action, pull down the **Edit** menu and choose **Undo Copy**, click the **Undo** button on the toolbar, or press control-Z.

Removing Group Members

Objects belong to a group through references, with the member objects residing in locations other than the group's. When you remove members, you delete only the references that associate those objects with the group. The objects themselves are not deleted, nor are any other objects that are shown below the members for additional context. For more information, see [Adding Group Members](#), earlier in this chapter.

Because member objects remain in their present locations, any number of objects can belong to a particular group, and a given object can belong to any number of groups. Consequently, one or more members can be removed from a group at any time, and an object can be removed from one or more groups at the same time.

An object removed from one group remains a member of all others to which it belongs. It is removed from all groups when deleted from its actual location. You can view an existing object's groups in the **Where Used** tab or window. For more information, see [Where Used Tab](#) or [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*, or [Deleting Objects](#), later in this chapter.

In the **Properties** tab or window, you can view other relevant information for removing group members. For example, the **Full Name** property shows a selected member's actual location, and the **Member Count**

property shows the number of members for a selected group. For more information, see [Properties Tab](#) in chapter 3, *Using the Architect/Requirements Main Window*.

Deleted member references are moved to your Architect/Requirements Recycle Bin and can be restored before it is emptied. For more information, see [Restoring Objects](#) or [Emptying Your Architect/Requirements Recycle Bin](#), later in this chapter.



You must have **Modify** permission for the group from which you intend to remove members.

To remove one or more members from a group:

1. In the navigation tree or the hierarchical content table, open the group's containing folder, and then click the group's plus sign (+) to display the members.

You can see the objects below all members by selecting the group, pulling down the **View** menu, and choosing **Expand All**, or by right-clicking the group and choosing **Expand All** from the pop-up menu.

2. In the content table, select each member that you want to remove.

You can select nonadjacent members by holding down the control key while you click the members. To select adjoining members, click the first member, hold down the shift key, and click the last member.

3. Pull down the **File** menu and choose **Delete**, and then click **Yes** when a confirmation message is displayed.

The selected member references are moved to your Recycle Bin.

Procedure Notes

Step 1: For more information, see [Navigation Tree](#) or [Hierarchical Content Table](#) in chapter 3, *Using the Architect/Requirements Main Window*.

Step 3: You can also right-click the selection and choose **Delete** from the pop-up menu. Or, click the **Delete** button on the toolbar or press the delete key. To reverse this action, pull down the **Edit** menu and choose **Undo Delete**, click the **Undo** button on the toolbar, or press control-Z.

To remove an object from one or more groups:

1. Display the groups for the object by doing one of the following:
 - For an object in the hierarchical content table, select the object and click the **Where Used** tab. You can open the **Where Used** window for this object by clicking the **Open tab** button on the notebook pane's toolbar.
 - For an object in the **Attachments** or **Links** tab:
 - . With the **Where Used** tab on top in the notebook pane, click the **Open tab** button on the toolbar to open the **Where Used** window.
 - . Select the object in the **Attachments** or **Links** tab.

The **Where Used** tab or window displays the groups to which the object belongs.

2. In the tab or window, select each group from which you want to remove the object.

You can select nonadjacent groups by holding down the control key while you click the groups. To select adjoining groups, click the first group, hold down the shift key, and click the last group.
3. Pull down the **File** menu and choose **Delete**, and then click **Yes** when a confirmation message is displayed.

The object's reference in each selected group is moved to your Recycle Bin.

Procedure Notes

Step 3: You can also right-click the selection and choose **Delete** from the pop-up menu. Or, click the **Delete** button on the toolbar or press the delete key. To reverse this action, pull down the **Edit** menu and choose **Undo Delete**, click the **Undo** button on the toolbar, or press control-Z.

Working With Shortcuts

A shortcut is an object that represents another object by reference. A shortcut can represent a folder, a requirement, a building block, a data dictionary, or a data definition. The referenced object and the shortcut can reside in the same Architect/Requirements project. Also, a shortcut can be created in one project to reference an object in a different project.

Every shortcut must be created in a folder, including a shortcut to a folder. Shortcuts cannot be created at the primary level of a project, directly below the project node.

For the selected object, you can create a shortcut at the top level of the destination folder. Or, you can create the shortcut at a lower level, directly below another object of the selected object's type. You can select the object in the navigation tree, the hierarchical content table, or the **Links** tab or floating window.

For more information, see [Navigation Tree](#), [Hierarchical Content Table](#), [Links Tab](#), or [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.

The shortcut appears in the hierarchy with the same properties as the referenced object, including the **Number** property for a requirement or a building block. The shortcut also reflects the trace links and attachments for the referenced object. In addition, all lower level objects below the object appear below the shortcut, including all objects in a folder and all children of a requirement or a building block.

Without affecting the referenced object, you can copy, move, and delete a shortcut. For more information, see [Copying Objects](#) and [Moving Objects](#), earlier in this chapter, and [Deleting Objects](#), later in this chapter.

You can do the following on the shortcut as if you selected the referenced object itself:

- Rename the object and send data for the object by E-mail. For more information, see [Renaming an Object](#), earlier in this chapter, and [Sending Object Data by E-Mail](#), later in this chapter.
- Create a new object of the same type, as a sibling of the referenced object and in that object's location. For more information, see [Creating a Folder](#), earlier in this chapter; [Creating a Requirement Object](#) in chapter 5, *Managing Requirements*; or [Creating a Building Block](#), [Creating a Data Dictionary](#), or [Creating Data Definitions](#) in chapter 6, *Constructing System Views With Building Blocks and Diagrams*.
- Open requirements and paragraphs in Microsoft Word to edit or view their content, and export those objects to Word. For more information, see [Exporting Objects to Microsoft Office Word](#), later in this chapter, and [Entering and Changing Requirement Content in Microsoft Office Word](#) and [Viewing Requirement Content](#) in chapter 5, *Managing Requirements*.
- Create and edit diagrams in Microsoft Office Visio for building blocks. For more information, see [Live Visio Diagrams](#) and [Editing a Diagram](#) in chapter 6, *Constructing System Views With Building Blocks and Diagrams*.
- Create, view, and delete trace links, and navigate to linked objects. For more information, see [Creating Trace Links](#), [Linking to an Object in Another Teamcenter Product](#), [Viewing Object Relationships](#), [Deleting Trace Links for an Object](#), and [Navigating to a Linked Object](#) in chapter 7, *Showing Object Relationships With Trace Links*.

- Change the values of editable properties, if the shortcut and the referenced object reside in the same project. You cannot edit properties for a referenced object in a different project. For more information, see [Properties Tab](#) in chapter 3, *Using the Architect/Requirements Main Window*, and [Editing the Properties of a Selected Object, Editing Properties in Table View Cells](#), or [Using the Live Excel Interface](#) in chapter 9, *Working With Object Properties*.

Shortcuts cannot be created for groups, notes, trace links, or diagrams. However, those object types can be added to groups as members. For more information, see [Adding Group Members](#), earlier in this chapter.

Creating a Shortcut



- You must have **Modify** permission for the intended destination.
 - To create a shortcut to a building block, a data dictionary, a data definition, a TRAM, or a diagram, you must have **Architect** privilege for the project.
1. Select the master object, pull down the **Edit** menu, and choose **Copy**.
 2. In the navigation tree, the content table, or the **Links** tab or window, select the destination object for the shortcut.
 3. Do one of the following:
 - In the navigation tree or the **Links** tab or window, pull down the **Edit** menu and choose **Paste Shortcut**.
 - In the content table, pull down the **Edit** menu and then do one of the following:
 - o To display the master object's children below the new shortcut, choose the **Paste Shortcut**→**Show Children** options.
 - o To keep the master object's children undisplayed, choose the **Paste Shortcut**→**Hide Children** options.



The current display state of the children is effective when the shortcut is replaced and when it is exported to Microsoft Office Word, to Microsoft Office Excel, or to an XML data file. You can change the display. For more information, see [Hiding and Showing the Children of a Master Object](#), [Replacing Shortcuts With Copies of the Master Objects](#), [Exporting Objects to Microsoft Office Word](#), [Exporting Objects to Microsoft Office Excel](#), and [Exporting Objects to an Architect/Requirements XML File](#), later in this chapter.

Procedure Notes

Step 1: You can also right-click the master object and choose **Copy** from the pop-up menu. Or, click the **Copy** button on the toolbar or press control-C.

Step 3: You can also right-click the destination object and choose **Paste Shortcut** from the pop-up menu. Or, press control-S.

Step 3: You can reverse this action by pulling down the **Edit** menu and choosing **Undo**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Navigating From a Shortcut to the Master Object

1. Select the shortcut.
2. Pull down the **View** menu and choose the **Go To→Go To Object** options.

The master object is highlighted in the content table, with the **Address** bar showing the project where the object resides.

Procedure Notes

Step 2: You can also right-click the shortcut and choose the **Go To→Go To Object** options from the pop-up menu.

Hiding and Showing the Children of a Master Object



The children below a shortcut are also shortcut objects. The actual children reside with the master object.

1. In the content table, select the shortcut.
2. Do one of the following:
 - In the **Properties** tab or window or the Edit Properties dialog window, do the following:
 - . Double-click the **Shortcut Options** property value.
 - . In the Multi-Choice dialog window:
 - To display the master object's children below the shortcut, choose **ShowChildren**.
 - To hide the master object's children, choose **HideChildren**.
 - Pull down the **Edit** menu, and then do one of the following:
 - . To display the master object's children below the shortcut, choose the **Paste Shortcut→Show Children** options.
 - . To hide the master object's children, choose the **Paste Shortcut→Hide Children** options.



The current display state of the children is effective when the shortcut is replaced and when it is exported to Microsoft Office Word, to Microsoft Office Excel, or to an XML data file. For more information, see [Replacing Shortcuts With Copies of the Master Objects](#), [Exporting Objects to Microsoft Office Word](#), [Exporting Objects to Microsoft Office Excel](#), and [Exporting Objects to an Architect/Requirements XML File](#), later in this chapter.

Replacing Shortcuts With Copies of the Master Objects



Before you replace a shortcut, set the shortcut to hide or show the master object's children as you prefer. The master object copy retains the shortcut's last setting. For more information, see [Hiding and Showing the Children of a Master Object](#), earlier in this chapter.



- You must have **Architect** privilege to replace a shortcut to a building block, a data dictionary, a data definition, a TRAM, or a diagram.
 - To replace a shortcut to other object types, you must have **Read and Write** access privilege for the project that contains the master object.
 - The master objects can reside in this project or in other projects.
1. In the content table at the top level of the containing folder, select the shortcuts.



You cannot replace shortcuts at lower levels or children of any shortcuts.

You can select one shortcut or a group of adjacent or adjoining shortcuts.

2. Pull down the **Edit** menu and choose **Uncouple Shortcut**.

The shortcuts are deleted and copies of the master objects are created in the same location.

Step 2: You can reverse this action by pulling down the **Edit** menu and choosing **Undo Uncouple Shortcut**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Sending Object Data by E-Mail

Directly from the Architect/Requirements client, you can send E-mail containing data for the objects in any project to which you have access. For each object that you select, Architect/Requirements extracts data from the database to an E-mail message, independent of the E-mail program on your computer.

You can send the message to any number of recipients. For internal recipients, the users and user groups in the project, you can address the message automatically from a list maintained by your project administrator. As in the Microsoft Outlook Address Book, for example, you can select users individually, and you can select user groups for distribution to many users simultaneously. For external recipients, those without access to the project, you enter E-mail addresses manually.



Also from the Architect/Requirements client, you can send E-mail that does not contain data for selected objects. For example, you may want to communicate with external vendors who do not have project access. Or, you may want to notify an internal user group of a project meeting.

You can send data for folders, requirements, building blocks, groups, notes, and diagrams. These object types and their subtypes can be selected in any combination and in any number. In addition, you can select group members, object shortcuts, and linked objects. For more information, see [Using Groups to Maintain Objects](#) and [Working With Shortcuts](#), earlier in this chapter, and [Viewing Object Relationships](#) in chapter 7, *Showing Object Relationships With Trace Links*.

Architect/Requirements can generate the data in the MHTML format, the HTML format, or the plain text format, according to your choice. Or, you can send the URLs of the selected objects as hyperlinks, which internal recipients can click to start Architect/Requirements and to navigate to the objects.

You can select the objects in the hierarchical content table, the **Attachments** tab or floating window, the **Links** tab or window, the **Where Used** tab or window, or the **Versions** tab or window. For more information, see [Hierarchical Content Table](#), [Attachments Tab](#), [Links Tab](#), [Where Used Tab](#), [Versions Tab](#), or [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.



Data is sent only for the selected objects. The message does not include data for unselected objects at levels below a selected folder, requirement, building block, group, or shortcut.

To send object data by E-mail:

1. Select each object for which you want to send data.

You can select nonadjacent objects by holding down the control key while you click the objects. To select adjoining objects, click the first object, hold down the shift key, and click the last object.

- 2.

Pull down the **Tools** menu and choose **Send Email**.

The Send Email dialog window is displayed, with the name of each selected object in the **Object(s)** field.



In a tab window, right-click the selection and choose **Send Email** from the pop-up menu. The **Tools** menu is not available in tab windows.

- 3.

Address the recipients by doing one or both of the following:

- For project users and user groups:
 - . Click the **To** button to display the Select User dialog window.
 - . Check the check box for each recipient user and user group, and then click **OK** to close this dialog window and add the names in the **To** field.
- For external recipients, enter the E-mail addresses directly in the **To** field.

In the remaining fields, you can do any or all of the following:

- In the **Subject** field, enter a general description of the message.
- In the **Send As** field, select one of the following data formats:
 - . Select **Web Page including graphics (MHTML)** to send the data in an **.mhtml** file, as an attachment containing all data specified by the document template named in the **Format Using** field. In that field, you can select another document template to specify different data.

The document template also determines the style sheet that is applied to the data. For more information, see [Exporting Objects to Microsoft Office Word](#), later in this chapter. If you have questions about document templates, consult your project administrator.

- . Select **Plain Text** to send the data as unformatted text in the message body.
 - . Select **URL** to send each object's URL as a hyperlink in the message body.
 - . Select **Web Page without graphics (HTML) as attached file** to send the data in an **.html** file, as an attachment containing only text and tables.
 - . Select **Web Page without graphics (HTML) embedded in message** to send the data as HTML content in the message body, with text and tables only.
- In the **Message** field, enter the text of the message body.
4. Click **OK** to close the Send Email dialog window and send the message.

A confirmation message is displayed, stating that the message has been sent.

Procedure Notes

Step 1: To see lower level objects in a hierarchical view, click the plus signs. Or, select an object with a plus sign, pull down the **View** menu, and choose **Expand All**. You can also right-click the object and choose **Expand All** from the pop-up menu.

Step 3: You can also enter E-mail addresses for project users and user groups directly in the **To** field.

Exporting Objects

The following sections describe procedures for exporting objects to AP233 STP and XML files, to Architect/Requirements XML data file, and to Microsoft Excel.

Exporting Objects to an AP233 STP File

From any project where you have Read and Write access, you can export objects to an AP233 STP file generated by Architect/Requirements. The file is saved in the drive or folder that you specify. This data can be imported from the file to a new or existing project in the Systems Engineering and Requirements Management module, where Architect/Requirements uses the data to reproduce the objects. In addition, the file can be imported to update property values for existing objects. For more information, see [Importing Objects From an AP233 STP File](#), later in this chapter, and [Updating Properties From an AP 233 STP File](#) in chapter 9, *Working With Object Properties*.



The AP233 export feature is for exchanging requirements data with other products that support the AP233 standard. This feature is not suitable for archiving data or moving data from one Architect/Requirements project or installation to another. To archive or move data, export the objects using the Architect/Requirements XML format. For more information, see [Exporting Objects to an Architect/Requirements XML File](#), later in this chapter.

The export file carries the **.stp** file name extension. The data is exported in the format defined by AP233, the STEP² application protocol for systems engineering data representation. In Architect/Requirements, the data represents folders and requirements. You can export all such objects in a selected project or a folder. The objects are exported with complete data for the following:

- All applicable user-defined properties, and the following system-defined properties:
 - **Name**
 - **Text**
 - **ROIN**
 - **Number**
 - **Source Paragraph**
 - **Source Filename**
 - **Baseline**
 - **Version Type**
 - **Security Profile**
 - **Subtype**

- Parent, child, and sibling relationships among the objects.
- Notes attached to the objects.
- Trace links for requirements, and notes.

If a starting or ending object is not present in the exported set of objects, then the trace link is neither exported from nor imported back to Architect/Requirements.

- Versions and variants of requirements.

² Standard for the Exchange of Product Model Data (ISO 10303).



Exporting building blocks, groups, diagrams, graphics, and OLE objects to **.stp** file is not supported. However, these objects can be exported to XML data files.

To export objects to an AP233 STP file:

1. Do one of the following:
 - For all folders and requirements in the project, select the project node in the navigation tree.
 - For all folders and requirements in a folder, select the folder in the navigation tree or in the hierarchical content table.

2. Pull down the **File** menu and choose the **Export**→**AP233 (.stp)** options.

Architect/Requirements displays the Save dialog window.

3. In the **File name** field, enter the name of the export file.

The **Save in** field displays the current drive or folder. You can save the export file in this drive or folder, or you can use this field to change the file location.

4. Click **Save**, or press the enter key.

The dialog window closes, and a message states that the export is in progress.



Although the export file can be opened in a text editor such as Microsoft WordPad or Notepad, changing the data is an advanced operation. Siemens PLM Software recommends that you do not open the export file.

Procedure Notes

Step 1: To see lower level objects in the content table, click the plus signs. Or, select an object with a plus sign, pull down the **View** menu, and choose **Expand All**. You can also right-click an object and choose **Expand All** from the pop-up menu.

Step 2: You can also right-click the project node or folder and choose the **Export**→**AP233 (.stp)** options from the pop-up menu.

Step 3: Architect/Requirements automatically adds the **.stp** file name extension to the file name that you enter. You can also select an existing **.stp** file in the list to fill in the **File name** field automatically and overwrite that file.

Exporting Objects to an AP233 XML File

From any project where you have Read and Write access, you can export objects to an AP233 XML file generated by Architect/Requirements. The file is saved in the drive or folder that you specify. This data can be imported from the file to a new or existing project in the Systems Engineering and Requirements Management module, where Architect/Requirements uses the data to reproduce the objects. In addition, the file can be imported to update property values for existing objects.



When you export objects from a folder, and create new objects and relations, importing from the exported AP233 XML file to the same folder does not delete the objects or relations created after the export. The import operation only updates the existing objects and creates new objects if the importing objects do not exist in the selected folder.

When a set of exported objects in a folder contains both a requirement and its versioned object, the newly created version of the requirement is not updated upon importing the set of objects in to the same folder. However, a new requirement object is created.



The AP233 export feature is useful for exchanging requirements data with other products that support the AP233 Part 28 standard. This feature is not suitable for archiving data or moving data from one Architect/Requirements project or installation to another. To archive or move data, export the objects using the Architect/Requirements XML format. For more information, see [Exporting Objects to an Architect/Requirements XML File](#), later in this chapter.

In Architect/Requirements, the data represents folders and requirements. You can export all such objects in a selected project or a folder. The objects are exported with complete data for the following:

- All applicable user-defined properties, and the following system-defined properties:
 - **Name**
 - **Text**
 - **ROIN**
 - **Number**
 - **Subtype**

- Building blocks.
- Parent, child, and sibling relationships among the objects.
- Notes attached to requirements, building blocks, folders, and ports.
- Trace links for requirements and notes.

If a starting or ending object is not present in the exported set of objects, then the trace link is neither exported from nor imported back to Architect/Requirements.

- Graphics and OLE objects in requirements.

Some of the Architect/Requirements concepts are not defined in the AP233 standard. Therefore, the export functionality is limited as specified below:

- Groups, diagrams, spreadsheets, change logs, Matlab files, and trace links involving building blocks are not exported to or imported from AP233 XML files.
- Notes on trace links and connections are not exported.

- Proxy links (such as wolf links and advanced links between Architect/Requirements objects and Enterprise Knowledge Management objects) that are created for Architect/Requirements objects are not exported.
- Only the currently effective version of the objects are exported.
- The **Document Options** property of the requirement objects is not exported.

Exporting objects that have **Document Options** set and importing them back may have undesired behavior. For example, consider an object that has **Document Options** set. If it is exported and imported at the same location, the Number property may be changed incorrectly. If the same object is imported in a new folder, then the object is created without having **Document Option** set.

- Shortcut objects are exported as objects that are referenced by the respective shortcuts, and not as shortcuts.

When you export a shortcut and import it back in the same or a different folder, a new object is created. The new object's type and property values are the same as that of the object referenced by the exported shortcut.

- Objects in the Administration module and information related to baselines are not exported.
- The **Security profile** property of the objects is not exported.

To export objects to an AP233 XML file:

1. Do one of the following:
 - For all folders and requirements in the project, select the project node in the navigation tree.
 - For all folders and requirements in a folder, select the folder in the navigation tree or in the hierarchical content table.

2. Pull down the **File** menu and choose the **Export**→**AP233 (.xml)** option.

Architect/Requirements displays the Save dialog window.

3. In the **File name** field, enter the name of the export file.

The **Save in** field displays the current drive or folder. You can save the export file in this drive or folder, or you can use this field to change the file location.

4. Click **Save**, or press the enter key.

The dialog window closes, and a message states that the export is in progress.



Although the export file can be opened in a text editor such as Microsoft WordPad or Notepad, changing the data is an advanced operation. Siemens PLM Software recommends that you do not open the export file.

Procedure Notes

Step 1: To see lower level objects in the content table, click the plus signs. Or, select an object with a plus sign, pull down the **View** menu, and choose **Expand All**. You can also right-click an object and choose **Expand All** from the pop-up menu.

Step 2: You can also right-click the project node or folder and choose the **Export**→**AP233 (.xml)** options from the pop-up menu.

Step 3: Architect/Requirements automatically adds the **.xml** file name extension to the file name that you enter. You can also select an existing **.xml** file in the list to fill in the **File name** field automatically and overwrite that file.

Exporting Objects to an Architect/Requirements XML File

From any project where you have Read and Write access, you can export objects to an XML data file in a format defined specifically for Architect/Requirements. The content of this file is more complete than the AP233 XML file and is well suited for archiving Architect/Requirements projects or moving Architect/Requirements content from one project or installation to another.

The file is saved in the drive or folder that you specify. The data can be imported to a project node or folder in the Systems Engineering and Requirements Management module, where Architect/Requirements uses the data to reproduce the objects. Thus, objects can be shared among different projects, in the same Architect/Requirements installation and in installations at other sites. For more information, see [Importing Objects From an Architect/Requirements XML File](#), later in this chapter.

Data is exported for all of the objects in the folder selected in the navigation tree or the hierarchical content table. The data in an Architect/Requirements XML file may represent folders, requirements, building blocks, notes, diagrams, and versions and variants of requirements and building blocks. The objects are exported with complete data for system defined and user defined properties, and for parent, child, and sibling relationships among the objects.

Data is included for trace links between objects within the folder. Inter-folder and inter-project trace links are not exported.



- When the data is imported to a different project, some data may be ignored if the target project schema is not compatible with the originating project schema.
- Versions and variants are preserved while exporting or importing a project. However, when exporting at the folder level, only the currently effective versions are exported. Hence the version tree is not re-established during import. Similarly, the variants are marked as variants at import, but they are not associated with any version tree or master object, as that version may or may not be part of the export.

To export objects to Architect/Requirements XML data file:

1. Select the folder in the navigation tree or in the hierarchical content table.
2. Pull down the **File** menu and choose the **Export**→**XML** options.
Architect/Requirements displays the Save dialog window.
3. In the **File name** field, enter the name of the export file.

The **Save in** field displays the current drive or folder. You can save the export file in this drive or folder, or you can use this field to change the file location.

4. Click **Save**, or press the enter key.

The dialog window closes, and a message indicates that the export is in progress. When the export is complete, a confirmation message is displayed.



Although the export file can be opened in a text editor such as Microsoft WordPad or Notepad, changing the data is an advanced operation. Siemens PLM Software recommends that you do not open the export file.

Procedure Notes

Step 1: To see lower level objects in the content table, click the plus signs. Or, select an object with a plus sign, pull down the **View** menu, and choose **Expand All**. You can also right-click an object and choose **Expand All** from the pop-up menu.

Step 2: You can also right-click the folder and choose the **Export**→**XML** options from the pop-up menu.

Step 3: Architect/Requirements automatically adds the **.xml** file name extension to the file name that you enter. You can also select an existing **.xml** file in the list to fill in the **File name** field automatically and overwrite that file.

Exporting Objects to Microsoft Office Excel

You can export data to a Microsoft Office Excel workbook for objects in the following Architect/Requirements client views:

- The hierarchical content table and your Recycle Bin. For more information, see [Hierarchical Content Table](#) and [Architect/Requirements Recycle Bin](#) in chapter 3, *Using the Architect/Requirements Main Window*.
- The **Properties**, **Attachments**, **Links**, **Where Used**, and **Versions** tabs and floating windows. For more information, see [Properties Tab](#), [Attachments Tab](#), [Where Used Tab](#), [Links Tab](#), [Versions Tab](#), and [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.

From any of those client views, you specify the objects that you want to export. You can export only the currently selected objects, or you can export all displayed objects whether or not they are selected.

To specify the properties to export, you can use one of the following:

- The properties that are currently displayed in the client view.
- An Excel template, containing property columns, tags for property values, and other information. For more information about Excel templates, see the *Systems Architect/Requirements Management Project Administrator's Manual*. If you have questions about Excel templates, consult your project administrator.
- A saved view of property columns. For more information, see [Customizing Views of Property Columns](#) in chapter 9, *Working With Object Properties*. If you have questions about views, consult your project administrator.

In addition, you can choose the state in which you want to create the workbook. You can export the objects and properties to a workbook that is not connected to the Architect/Requirements database. Or, you can create a new Excel Live workbook, through which you can interactively edit the properties of exported objects and see your changes applied in the Architect/Requirements client. You can import the object data from either workbook to create new objects automatically. For more information, see [Using the Live Excel Interface](#) in chapter 9, *Working With Object Properties*, and [Importing Objects From Microsoft Office Excel](#), later in this chapter.

Architect/Requirements extracts the data to a read-only Excel workbook. Although this file is temporary, you can use all of Excel's viewing, printing, and navigation features. From within this file, you can send the data to E-mail recipients, such as colleagues, customers, and suppliers. In both the E-mail message and the file itself, you can click the object type indicators to navigate to those objects in the client.



If you sort the data in Excel, the object type indicators and the objects may be mismatched. This condition occurs in Excel when a cell contains a graphic that is taller than the row height. For a row height that accommodates the indicators, increase the size of Excel's standard font.

The effective size depends on the standard font that you use. With Arial, for example, a size of 11 points allows the indicators to be sorted with the correct objects. To increase the size, Click Excel's **Office Button**, choose **Excel Options**, and select the new size on the **Popular**→**When creating new workbooks** section.

If you want to make this change, do so before you export. Otherwise, you must exit and restart Excel for the change to take effect.

The temporary file is deleted from your computer when you exit Architect/Requirements. While this file is open, however, you can create a permanent version that is independent of Architect/Requirements.

You can store the permanent file for reference, as a record of the view at a certain time, for example, or as a comparison with the corresponding view in another project or folder. Because you assign the file name, file type, and location, you can easily retrieve the exported view. You can click an object type indicator in the file to start Architect/Requirements and automatically navigate to the object in the client.

The view in the content table is set by your selection in the navigation tree:

- With a project node selected, the content table shows the folders at the project's primary level.
- With a folder selected, the content table shows the folders, requirements, building blocks, and groups at the folder's top level.
- With the Recycle Bin selected, the content table shows the objects that you deleted from all projects to which you have access.

In the notebook pane, you set the views by selecting an object in the navigation tree or the content table and clicking the **Properties**, **Attachments**, **Links**, **Where Used**, or **Versions** tab. You can then open tab windows to set views for objects selected in the notebook pane.

While exporting the content to Excel using a view, Architect/Requirements reads the locale-specific styles for date from a CSS (for example, **DateStyle_en_US.css**) file. There should be a **DateStyle_languageCode_countryCode.css** file present for each locale. If the date styles for the client locale are not found, then the user is prompted to confirm before exporting the file using English (US) date styles.

Before exporting a table, consider the information that you want to see in Excel. For example, you can add columns to export other properties, and you can remove displayed columns to exclude those properties from the file. To adjust for more or fewer properties, you may want to rearrange or resize columns, or sort the information by a different column. For more information, see [Adding and Removing Columns](#), [Rearranging Columns](#), [Resizing Columns](#), and [Setting the Sort Column](#) in chapter 9, *Working With Object Properties*.



Microsoft Office Excel 2013 or Excel 2016 must be installed on your computer.

To export objects to Microsoft Office Excel:

1. To activate the view, do one of the following:
 - For the hierarchical content table, pull down the **View** menu and choose the **Go To**→**Content Table** options, or click inside the table.

- For the **Properties** tab, pull down the **View** menu and choose the **Go To→Notebook Pane** options, or click inside the table.
- For the Recycle Bin, the **Attachments** tab, the **Links** tab, the **Where Used** tab, or the **Versions** tab, click inside the table.

You can open the floating window for an activated tab by clicking the **Open tab** button on the notebook pane's toolbar.

2.

Pull down the **File** menu and choose the **Export→Excel Spreadsheet** options to display the Export To Excel dialog window.

You can also right-click inside the view and choose the **Export→Excel Spreadsheet** options from the pop-up menu.



In a tab window, right-click inside the table and choose the **Export→Excel Spreadsheet** options from the pop-up menu. The **File** menu is not available in tab windows.

3. Under **Object Selection**, do one of the following:

- To export only the selected objects, click **Export Selected Objects**.
- To export all objects in the view, click **Export All Objects in View**.

4. Under **Formatting**, do one of the following:

- To export the properties in the currently displayed columns, click **Use Current View Columns**.
- To export the properties specified in an Excel template or in a saved view of property columns:
 - Click **Use Excel Template or View**.

Below this button, the list becomes available. The list shows the Excel templates and the public views for the project, and includes the private and pending views for your user name.

- In the list, select the Excel template or the view.



- The Excel template may contain multiple worksheets, and each worksheet may specify different properties and rules. Therefore, data may be separated on worksheets in the export file.
- If the Excel template or the view contains properties that do not apply to an object type specified for export, the inapplicable values are indicated in the related cells.

5. Under **Output**, do one of the following:

- To create a workbook that is not connected to the database, click **Static Snapshot**.
- To create an interactive live Excel workbook that is connected to the database, click **Excel Live**.

6. Click **OK** to close the dialog window.

Excel opens a read-only file containing the specified rows and columns. This file is temporary and is deleted from your computer when you exit Architect/Requirements.

You can create a permanent file by pulling down Excel's **File** menu and choosing **Save As**. In the Save As dialog window, assign the file name, file type, and location outside the Architect/Requirements database, for example, on a local drive. This file remains on your computer when you exit Architect/Requirements.



- If you save a live Excel file, Siemens PLM Software recommends that you assign it the **.xlsm** file type. Only the **.xlsm** file type supports the full capability of live Excel.
- If you save a live Excel file as the **.mhtml** file type, it becomes static and loses the live capability permanently.

Exporting Objects to Microsoft Office Word

Through the Architect/Requirements interface with Microsoft Office Word, you can capture current information from the database for analysis and distribution. For each object that you specify, Architect/Requirements extracts data to a Word document, using a *document template* as a reference. In turn, the document template controls the export process through the following:



An *object template*, which determines the data that is exported. Tags in the object template represent certain object properties, whose values are extracted from the database to the export document for each specified object. By default, one of two object templates is assigned to each object type, and to the system-defined subtypes, **Document** for folders and **Paragraph** for requirements:

- o For requirements, paragraphs, notes, and diagrams, the default object template contains the **Name** property, in the template's heading, and the **HTML** property, in a paragraph below the heading. With this template, each object's name is extracted to a heading in the export document, and the full content, including tables and graphics, immediately follows in the body.

The **HTML** property is not displayed in the Systems Engineering and Requirements Management module because the value can exceed the effective size for table cells. However, you can see an object's full content in the **Preview** tab or floating window. For more information, see [Preview Tab](#) or [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.

- o For other object types, the default object template contains only the **Name** property in the heading. With this template, each object's name is extracted to a heading in the export document.
- A *style sheet*, which determines the Word formatting styles that are applied to the data in the export document.

The document template may be customized to export additional data and apply different styles. In the default object template, for example, your project administrator may add the **ROIN** property to export the Requirement Object Identification Number (ROIN) of each requirement and paragraph. Your project administrator may create style sheets to apply custom Word styles to the exported data for particular

object types. Furthermore, a unique object template may be assigned to each type definition, to export different data for that object type. For more information about templates and style sheets, see the *Systems Architect/Requirements Management Project Administrator's Manual*. If you have questions about templates or style sheets, consult your project administrator.

By the way in which you specify the objects, you set the context of the data in the export document. The context can range from broad to narrow. For example:

- To specify all requirements in a folder, select only the containing folder. The document includes data for all direct children and lower level descendants of each parent requirement, but not for the folder itself. In this context, you see the data according to the organization of requirements within the entire folder.

For each requirement at the top level, the data from the object template heading is formatted in Word's Heading 1 style. For each child and descendant, that data is formatted in a Word heading style that matches the object's level in the hierarchy. Headings appear in ascending order of the **Number** property values.



If the selected folder also contains groups or other folders, data is not exported for those objects. Data is not exported for requirements in folders within the selected folder.

- To specify one requirement, its direct children, and all lower level descendants, select only the parent requirement. In this context, you see the data according to the relationships in the entire hierarchy below the parent.

For the parent, the data from the object template heading is formatted in Word's Heading 1 style. For the direct children, that data is formatted in Word's Heading 2 style, with data for lower level descendants in heading styles that match those levels. Headings appear in ascending order of the **Number** property values.

- To specify all members of a group, select only that group. In this context, you see the data for all of the group's members, the objects shown at the level directly below the group, but not for the group itself.

For each member, the data from the object template heading is formatted in Word's Heading 1 style. Headings appear in the order in which the members are found in the database.



Data is not exported for objects that are shown at lower levels below the members. For more information, see [Adding Group Members](#) in chapter 4, *Maintaining a Project*.

- To specify one or more objects, of mixed types or of the same type, select the individual objects while holding down the control or shift key. In this context, you see data only for the selected objects, regardless of any hierarchical relationships to other objects.

For each object, the data from the object template heading is formatted in Word's Heading 1 style. Headings appear in the order in which the objects are displayed on the screen, regardless of the order in which the objects are selected.



When multiple objects are specified, data is not exported for unselected objects below a selected folder, requirement, building block, or group.



When objects with embedded OLE content are exported to Microsoft Office Word, none of the OLE content is present in the Word document. The OLE objects are exported with a preview

image representing the OLE content. Objects retain their OLE content when opened for edit individually.

The views where you select the objects depend on the objects that you want to specify:

- For all requirements in a folder, select the containing folder in the navigation tree, the hierarchical content table, or the **Links** tab or window.
- For individual objects, select them in the hierarchical content table, the **Links** tab or window, or the **Versions** tab or window. In each view, you can select one object or multiple nonadjacent and adjoining objects. In the hierarchical content table, you can select objects in different folders. In the **Links** tab or window, you can select objects in either pane but not in both panes at the same time.



A dash before a level in a **Number** value, for example, **1.-1**, indicates that the requirement is an unnumbered part of the parent paragraph at the next higher level. The requirement's **Document Options** property value is **Disable Numbering**. Such a requirement is not assigned a heading or an outline level in the export document.

The document template must now reside on the folder prior to export.

To add the document template to the folder:

1. In the **Notebook** pane, select **Properties**.
2. Scroll down to find **Document Template** in the list of properties.
3. Double-click the empty values cell next to **Document Template**.
The **Document Template** dialog box is displayed.
4. Select the document template that you want as the default template for the folder.
5. Click **OK**.

The following are guidelines for exporting Microsoft Word 2013 or Microsoft Word 2016 documents:

- Document templates must be used for successfully exporting documents from Microsoft Word. Users must create a custom document template, a custom style sheet, custom text object templates, and object templates. Assign this document template to a folder. To maintain consistency, users must import the styles into each associated component of the document template. Siemens PLM Software recommends using the Microsoft Word Document organizer to transfer the styles.
- The style sheet attached to the document template is applied when the document is exported and should not be changed at that time. Changing the style sheet at the time of export yields inconsistent results. For consistency in using the default styles, especially for the list styles such as bullets and numbers, you must allow the consistent formatting of the exported document. Custom styles work fine but you must create the styles properly using Microsoft Word.
- When exporting a document, the style sheet must be explicitly associated with the document template. If it is not, the style sheet is not applied successfully on exported document.
- You can also export macros when exporting objects to Word. However, you must add the macros to the Architect/Requirements Style Sheet that is used if you want the macros to be available after export. After the export, the macros can be triggered using VBA events.

- You cannot include the Diagram Images directly when exporting to Word. You must first symbolically reference it into a requirement or note. To include the diagram image in a Word export of the building block structure, create a note on the building block and reference Diagram Image in that note. For information on including notes in Word exports see *Object Templates* in the *Systems Architect/Requirements Management Project Administrator's Manual*.

To export objects to Microsoft Office Word:



Microsoft Office Word 2013 or Microsoft Office Word 2016 must be installed on your computer.

1.

Specify the objects as described above, and then pull down the **File** menu and choose the **Export**→**Word Document** options.

The Select Document Template dialog window is displayed. In the **Select a Document Template** field, you can change the selection for this document.

2. Do one of the following:

- To use the selected document template for all objects, click **OK** to close the Select Document Template dialog window and generate the document.
- To change the stylesheet for the document or the object template for one or more types, click **More** to display the Document Template dialog window.

. Do one or both of the following:

- o To change the Word styles that are applied to all data in the document, select the style sheet in the **Select Stylesheet** field.
- o To change the object template for a type in the **SubTypes** column:

a.

Double-click the cell in the **OverrideObjectTemplate** column to display the **Single-Choice** dialog window.

- a. Select the check box for the object template that you want to use, and then click **OK** to close this dialog window.

Repeat these steps for each additional type whose object template you want to change.



If you want to apply a numbered or bulleted list style, create an appropriate list style and apply the list style. You must also add the list style to the stylesheet.

There are out of the box Microsoft Word list styles can be added to the stylesheet such as List Bullet.

. Click **Export** to close the Document Template dialog window and generate the document.

A Word file opens (.mhtml), with data for the exported objects. You can view and print the document and send it by E-mail and fax.



This file is temporary and is deleted from your computer when you exit Architect/Requirements.

You can create a permanent document by clicking Word's **Office Button** and choosing **Save As→Word Document**. In the **Save As** dialog window, assign the file name, file type, and location. This document can serve as a record for comparison with future changes. Or, the document can be imported to create new requirements in the same project or another project, and content can be edited independently of Architect/Requirements.

Procedure Notes

Step 1: You can also right-click the selection and choose the **Export→Word Document** options from the pop-up menu.

OLE objects embedded in requirement content are not exported to Microsoft Office Word. This limitation includes the first OLE object encountered. However, the first object was exported in earlier Architect/Requirements versions.

The graphics for the OLE objects are extracted to the Word export document. However, double-clicking the graphic does not launch the OLE application.

This change was made because the embedded OLE objects sometimes became associated with the wrong graphic, which caused the graphics to behave incorrectly when double-clicked. Also, incorrect graphics appeared when the document was printed. Only the first OLE object could be exported, which led to inconsistent behavior.

To workaround or avoid:

You can use Tool Command Language (Tcl) to include the first OLE object in requirement content that is exported to Word. For example:

- To specify which document template to use in the export, enter:

```
set documentTemplate ""
```

Empty double quotation marks ("") specify the default document template. Or, enter the name of another document template in the quotation marks.

- To export the selected objects to a Word file on the server, including embedded OLE objects, enter:

```
set includeOle true
```

```
set file [exportDocument $selected MS_WORD $documentTemplate true $includeOle]
```

- To download the file to the client and open it in Word, enter:

```
createAction FileDownload [list $file MS_WORD]
```

Exporting Objects for importing into Teamcenter

Architect/Requirements allows you to export objects for importing into Teamcenter. The objects are exported in XML format.



You must have access to the Architect/Requirements server to access the exported file.

You can export the following objects:

- Requirements and paragraphs including their rich text
- Folders
- Document Folders

These map to requirement specifications in Teamcenter.

- Trace links
- Notes including rich text
- Groups



When exporting a portion of a project, references to project objects outside the exported region are not resolved. Objects referenced by a trace link, shortcut, or group membership are not exported. You must export at the project level to ensure that all intraproject relationships are preserved.

To export objects for Teamcenter:

1. Select the project or objects that you want to export.
2. Pull down the **File** menu and choose **Export**→**TC XML**.

Architect/Requirements displays a message confirming the export. It also displays the location of the exported file.

The exported objects are saved on the Architect/Requirements server in XML format. The naming convention that Architect/Requirements follows for the exported file is **TcSE_to_TC_Export-nnnnnnn.xml** where **nnnnnnn** is a number that increments with each export.

The schema for the XML file is written to **TcSE_to_TC_Schema-nnnnnnn.xsd** file. The schema includes user defined subtypes and properties. This file is used for mapping the Architect/Requirements type and property names to the corresponding Teamcenter names.

A folder named **TcSE_to_TC_Export-nnnnnnn.xml** is also created. It contains the **docm** files of the exported requirements and paragraphs.

You must process the exported XML file to convert it to TC XML format. The TC XML file can then be imported into Teamcenter.

The **TcSE_to_TC_Text-nnnnnnn.xml** file is used for importing rich text into Teamcenter.

For information about importing to Teamcenter, see the topic *Migrate from standalone Systems Engineering and Requirements Management* in Appendix A of the *Getting Started with Systems Engineering Guide* Teamcenter manual set.

For information on controlling what is exported, see *Customizing exports for migration to Teamcenter* in the *Systems Architect/Requirements Management API Reference*.

Importing Objects

The following sections describe procedures for importing objects from AP233 STP and XML files, from Architect/Requirements XML data file, and from Microsoft Excel.



If you encounter an out of memory error when importing a large document, you may need to increase the maximum memory available for the Architect/Requirements client.

For information on increasing the available memory, see *Appendix D - Changing the maximum memory available for rich client*.

Importing Objects From an AP233 STP File

To create objects automatically in the Systems Engineering module, you can import object data from files that carry the **.stp** file name extension. This data is in the format defined by AP233, the STEP³ application protocol for systems engineering data representation. In Architect/Requirements, the data represents folders and requirements that were previously exported to the file. Architect/Requirements uses the data in the file to reproduce the objects. You can also use the import file to update property values for existing objects. For more information, see [Exporting Objects to an AP233 STP File](#), earlier in this chapter, and [Updating Properties From an AP 233 STP File](#) in chapter 9, *Working With Object Properties*.



This function is for exchanging data with other products that support the AP233 standard. This procedure is not for archiving data or moving data between Architect/Requirements projects or installations. To archive or move data, import the objects using the Architect/Requirements XML format. For more information, see [Importing Objects From an Architect/Requirements XML File](#), later in this chapter.

Points to Note

- Before importing, ensure that you know the name and location of the **.stp** file containing the object data that you want.
- Although the import file can be opened in a text editor such as Microsoft WordPad or Notepad, changing the data is an advanced operation. Siemens PLM Software recommends that you do not open the import file.
- You must have **Modify** permission for the project or folder where you intend to import the objects.
- Importing building blocks, groups, diagrams, graphics, and OLE objects from an **.stp** file is not supported. However, these objects can be imported from XML data files.
- Date properties with blank values receive the value **TBD** when imported.

For a new or existing project, you can import the objects to the primary level, directly below the project node, or to the top level of any folder in the project. The imported objects are complete with the following:

- Applicable user-defined properties, and the following system-defined properties:
 - **Name**

³ Standard for the Exchange of Product Model Data (ISO 10303).

- o **Text**
 - o **ROIN**
 - o **Number**
 - o **Source Paragraph**
 - o **Source Filename**
 - o **Baseline**
 - o **Version Type**
 - o **Security Profile**
 - o **Subtype**
- Parent, child, and sibling relationships among imported objects.
 - Notes attached to the objects.
 - Trace links for requirements and notes.

If a starting or ending object is not present in the exported set of objects, then the trace link is neither exported from nor imported back to Architect/Requirements.

- Versions and variants of requirements.

Variants that are exported will be imported as new objects in Architect/Requirements.

To import objects from an AP233 STP file:

1. Do one of the following:
 - For a project's primary level, select the project node in the navigation tree.
 - For a folder's top level, select the folder in the navigation tree or in the hierarchical content table.



Siemens PLM Software recommends that you import the objects to an empty folder.

2. Pull down the **File** menu and choose the **Import**→**AP233 (.stp)** options.

Architect/Requirements displays the Open dialog window, which lists existing folders and files in the current drive or folder. The **Look in** field displays the current drive or folder. If the import file is not in the list, you can use this field to change the drive or folder.

- 3.

Select the import file in the list, and then click **Open**.

The Import AP233 dialog window is displayed.

4. Select **Create New Objects**, and then click **OK**. You can also select **Update Existing Objects** import property values from an AP 233 **.stp** file that contains records for objects in the database. For more information see chapter 9, [Updating Properties From an AP 233 STP File](#).



This action clears the queue for the **Undo** option and cannot be reversed.

The Import AP233 dialog window closes, and a message is displayed to indicate that the import is in progress. When the import is complete, a confirmation message is displayed.

Procedure Notes

Step 1: To see lower level objects in the content table, click the plus signs. Or, select an object with a plus sign, pull down the **View** menu, and choose **Expand All**. You can also right-click an object and choose **Expand All** from the pop-up menu.

Step 2: You can also right-click the import folder or the project node and choose the options from the pop-up menu.

Importing Objects From an AP233 XML File

When you import from an AP233 XML file, the import process updates objects if they already exist in Architect/Requirements. Otherwise, new objects are created. Architect/Requirements uses the data in the AP233 XML file to reproduce the objects.



When you export objects from a folder, and create new objects and relations, importing from the exported AP233 XML file to the same folder does not delete the objects or relations created after the export. The import operation only updates the existing objects and creates new objects if the importing objects do not exist in the selected folder.

When a set of exported objects in a folder contains both a requirement and its versioned object, the newly created version of the requirement is not updated upon importing the set of objects in to the same folder. However, a new requirement object is created.



The import operation from an AP233 XML file is for exchanging data with other products that support the AP233 Part 28 standard. The import operation is not for archiving data or moving data between Architect/Requirements projects or installations. To archive or move data, import the objects using the Architect/Requirements XML format. For more information, see [Importing Objects From an Architect/Requirements XML File](#), later in this chapter.

Points to Note

- Before importing, ensure that you know the name and location of the **.xml** file containing the object data that you want.
- Although the import file can be opened in a text editor such as Microsoft WordPad or Notepad, changing the data is an advanced operation. Siemens PLM Software recommends that you do not open the import file.
- You must have **Modify** permission for the project or folder where you intend to import the objects.
- The Architect/Requirements concepts of groups, diagrams, spreadsheets, change logs, Matlab files, and trace links involving building blocks are not defined in the AP233 standard. Therefore, those objects are not exported to or imported from AP233 XML files.

For a new or an existing project, you can import the objects to the primary level, directly below the project node, or to the top level of any folder in the project. The imported objects are complete with the following:

- Applicable user-defined properties, and the following system-defined properties:
 - **Name**
 - **Text**
 - **ROIN**
 - **Number**
 - **Subtype**

- Building blocks.
- Parent and child relationships among imported objects.
- Notes attached to requirements, building blocks, folders, and ports.
- Trace links for requirements and notes.

Behavior After Importing Objects

- When the subtype property of an object is changed after the object is exported to an AP233 XML file, importing it into the same folder ignores the changed subtype property of the object.

For example, export a folder that contains a requirement R1 with R12 and R13 as child objects to an AP233 XML file. Change the subtype property of R13 to **Paragraph**. Select the same folder and import it from the exported AP233 XML file. R13 is shown as a paragraph, rather than as a requirement.

- If Architect/Requirements does not find the exact subtype for an object, the object is created as its parent type.
- If a starting or ending object is not present in the exported set of objects, then the trace link is neither exported from nor imported back to Architect/Requirements.
- Date properties with blank values receive the default value when imported.
- Shortcuts of objects are imported as new objects.

When you export a shortcut and import it back in the same or a different folder, a new object is created. The new object's type and property values are the same as that of the object referenced by the exported shortcut.

- Variants are imported as normal objects.

To import objects from an AP233 XML file:

1. Do one of the following:
 - For a project's primary level, select the project node in the navigation tree.
 - For a folder's top level, select the folder in the navigation tree or in the hierarchical content table.

Siemens PLM Software recommends that you import the objects to an empty folder.

2. Pull down the **File** menu and choose the **Import→AP233 (.xml)** option.

Architect/Requirements displays the Open dialog window, which lists existing folders and files in the current drive or folder. The **Look in** field displays the current drive or folder. If the import file is not in the list, you can use this field to change the drive or the folder.

3. Select the import file in the list, and then click **Open**.

Procedure Notes

Step 1: To see lower-level objects in the table of contents, click the plus sign. Or, select an object with a plus sign, pull down the **View** menu, and choose **Expand All**. You can also right-click an object and choose **Expand All** from the pop-up menu.

Step 2: You can also right-click the import folder or the project node and choose the options from the pop-up menu.

Importing Objects From an Architect/Requirements XML File

From any project where you have Read and Write access, you can export objects to and import from an XML data file in a format defined specifically for Architect/Requirements. The content of this file is more complete than the AP233 XML file and is well suited for archiving Architect/Requirements projects or moving Architect/Requirements content from one project or installation to another.

To create or update objects automatically in the Systems Engineering and Requirements Management module, you can import object data from Architect/Requirements XML files. The data represents all objects in a folder that was previously exported using the Architect/Requirements XML format.

For a new or existing project, you can import the objects to the primary level, directly below the project node, or to the top level of any existing folder. The imported objects are complete with the following:

- Properties and values, both system-defined and user-defined.
 -  Date properties with blank values receive the value **TBD** when imported.
- Parent, child, and sibling relationships among imported objects.
- Trace links between objects represented in the file.
- Versions and variants of requirements and building blocks.



- You must have **Modify** permission for the project or folder where you intend to import the objects.
- To import building blocks, TRAMs, or diagrams to an existing project, you must have **Architect** privilege for the project.
- Imported diagrams are not connected directly to the Architect/Requirements database as are diagrams created through live Visio.
- If the import file contains data for properties that do not exist in the project, that data is ignored.
- If the import file contains data for trace links that cross project boundaries, that data is ignored.
- If the import file contains data for an object subtype that does not exist in the project, an object of the built-in base type is created.



- Before importing, ensure that you know the name and location of the Architect/Requirements XML file containing the object data that you want.
- A file that was exported from the Administration module cannot be imported to the Architect/Requirements module.
- The file being imported must have been exported from either the same Architect/Requirements version or the prior major version. Exporting from a newer version and importing into an older version is not supported because it may not be possible to map new data into an older database.

- Although the import file can be opened in a text editor such as Microsoft WordPad or Notepad, changing the data is an advanced operation. Siemens PLM Software recommends that you do not open the import file.
- Versions and variants are preserved while exporting or importing a project. However, when exporting at the folder level, only the currently effective versions are exported. Hence the version tree is not re-established during import. Similarly, the variants are marked as variants at import, but they are not associated with any version tree or master object, as that version may or may not be part of the export.

To import objects from Architect/Requirements XML data file:

1. Do one of the following:

- For a project's primary level, select the project node in the navigation tree.
- For a folder's top level, select the folder in the navigation tree or in the hierarchical content table.



To import building blocks and their subtypes to the selected folder, you must have **Architect** privilege for the project. Otherwise, building blocks and subtypes in the import file are ignored.

2. Pull down the **File** menu and choose the **Import**→**XML** options.

Architect/Requirements displays the Open dialog window, which lists existing folders and files in the current drive or folder.

3. Select the import file in the list, or enter the name in the **File name** field.

The **Look in** field displays the current drive or folder. If the import file is not in the list, you can use this field to change the drive or folder.

4. Click **Open**, or press the enter key.



This action clears the queue for the **Undo** option and cannot be reversed.

The dialog window closes, and a message is displayed to indicate that the import is in progress. When the import is complete, a confirmation message is displayed.

Procedure Notes

Step 1: To see lower level objects in the content table, click the plus signs. Or, select an object with a plus sign, pull down the **View** menu, and choose **Expand All**. You can also right-click an object and choose **Expand All** from the pop-up menu.

Step 2: You can also right-click the import folder or the project node and choose the **Import**→**XML** options from the pop-up menu.

If object names or text properties include certain control characters when exported to XML, then the file is not imported correctly. This problem occurs only in special cases when you try to copy a file from separate applications into the Architect/Requirements edit fields for object name and text properties. It does not occur when entering text with the keyboard.

These control characters are mapped to the HTML **&#nn;** notation in the XML exported file. One specific case is the Vertical Tab character, which appears as ****.

To work around or avoid:

Load the XML file into any XML or plain text editor, and replace the offending **&#nn;** patterns with a space. You may also remove them, depending on the control character and the desired outcome. You must replace or remove all the five characters, beginning with the ampersand and ending with the semicolon.

Importing Objects From Microsoft Office Excel

To create objects automatically in the Systems Engineering and Requirements Management module, you can import object data from any Microsoft Excel workbook that meets the following conditions:

- The import file can have any of the Excel file types including: **.xls, .xlsm, .xlsx**.
- Each column represents a single object property in Architect/Requirements, and each row below the column headings represents a single object.
- The column headings in the first row exactly match the names of object properties in Architect/Requirements. For any heading that does not match a property name, the import wizard has the ability to create new object properties or there is the option to ignore them.
- Objects that define a relationship between two other objects cannot be created by the **Excel Import Wizard**. This includes trace links and connections. The reason is difficulty in accurately identifying the two objects to be related.

For example, the conditions may be met by a file containing previously exported data. For more information, see [Exporting Objects to Microsoft Office Excel](#), earlier in this chapter.

Only the first worksheet in the file is imported. Cell formatting is ignored. In the selected import folder, one object is created for each populated row in the file.

Properties of imported objects are subject to the following conditions:

- Editable property values are applied to the new objects as entered in the worksheet.
- Read-only properties take system-defined values automatically, regardless of the values entered in the worksheet.
- Default values are assigned to editable text and numeric properties whose format is incorrect in the worksheet.
- Date properties with blank values receive the value **TBD** when imported.
- Markup (for example, boldface and italics), fonts, bullets, and other formatting are not preserved when importing text properties or the full text of requirements or notes. The imported text is plain.
- The **Number** property is applied to new requirements and building blocks. However, the rows representing these objects in the worksheet must be in correct logical order with no gaps in the numbering sequence.
- The **Type Name** and **Subtype** properties are assigned to each new object.



- You must have **Modify** permission for the folder to which you intend to import the objects.

- To import building blocks, TRAMs, or diagrams, you must have **Architect** privilege for the project.



If you want to perform imports that need to create new properties or new choices on existing properties, you must have the Project Administrator or Power User privilege. If you want a Power User to be able to import Excel spreadsheets with additional columns or changed information, the Power User must be assigned permission to modify definitions and type definitions. Power Users can make additions or changes to spreadsheets within Architect/Requirements and they can export and import such spreadsheets.

- If an error occurs during the import, all objects imported before the error are retained.
- Microsoft Office Excel 2013 or Excel 2016 must be installed on your computer.

To import objects from Microsoft Office Excel:

1.

Select the folder under which you want to import objects from Excel. To do this, pull down the **File** menu, and choose the **Import**→**Excel Spreadsheet** options. Or, right-click the folder and choose the **Import**→**Excel Spreadsheet** options from the pop-up menu.

Architect/Requirements displays the **Open** dialog window, which lists existing folders and files in the current drive or folder. If the import file is not in the list, you can use the **Look in** field to change the drive or folder.

2. Select the import file in the list, or enter the name in the **File name** field.
3. Click **Open** or press the enter key.



This action clears the queue for the **Undo** option and cannot be reversed.

4. Excel opens the file, places it in the background and the **TcSE Excel Import Wizard** window opens. There are four separate folder tabs for this wizard.

- **Start** - This area has a message that states “Welcome to the Excel Import Wizard for Teamcenter Systems Engineering. This wizard will allow you to import an Excel spreadsheet to a TcSE database.” Click **Next** to continue working with the wizard.
- **Default Type** - This is the area where you select how you want the information to import into Architect/Requirements from Excel. Your choices for the default type/subtype are: Requirement, Building Block, Data Definition, Data Dictionary, Document, Folder, Paragraph, TRAM and subtype of these objects. Once you select how to handle the new properties, select **Next**; the last tab is displayed.
- **Select Fields** - This area lists all of the column headings in the Excel spreadsheet that you have chosen to import and fields to select how the information should be handled during the import. The fields are listed as follows:
 - Excel Column: Lists all of the column headings in the spreadsheet.
 - TcSE Property: Allows you to indicate whether it is a new property to be created during the import. You also have the option to omit the property on import if you

choose not to add it to the Architect/Requirements database; however, this causes the column to be omitted upon import.

- o Type: Allows you to choose how the information is to be formatted within the project. You can choose; **Text**, **Date**, **Multi-Choice**, **Numeric**, or **Single Choice**.



The **Excel Import Wizard** associates existing properties (properties that already exist in the Architect/Requirements database) with new types (as happens with the new properties during the import process). For new properties, the **Excel Import** decides whether the property should be associated with types, based on which cells have a value for that property. For existing properties, if the property is already associated with the type, the values are imported. If the property is not already associated with the type, it associates with the type and the values are imported.

- **Choice Definitions** - This area shows only the choice properties that are either new or already existing in Architect/Requirements. For this area to work, you must have either one new or existing property to be imported as a **Single Choice** or **Multiple Choice** property; otherwise, this window is skipped. If there are new properties, the import wizard automatically detects them from the Excel sheet and displays them under the new choices column. If they are existing Architect/Requirements choice properties, they are shown in the **Existing** choices column. New choice properties are displayed in the **New Choices** column. This new column gives you the ability to keep the new choice or use the **Add New Choices** column and add additional choices. Select **Next** to continue.



The default type is associated with objects when the **Type Name** and **Subtype** columns either have no values, or have invalid values. This includes when they are non-existent in the Excel sheet. All the rows that do not have type or subtype information are imported as the default type selected in the **Default Type** window.



For performance reasons, the existing choices for dynamic picklist properties are not displayed.

The **Summary Window** displays all the new properties and the types they are associated with — including any new choices added to the Architect/Requirements database. Once you have finished reviewing the **Summary Window**, a **File Import** window displays the progress of the import. When complete, your file is in Architect/Requirements in the format you have chosen.

ASCII control characters are not allowed in Architect/Requirements text property values. If a control character is present in an imported Excel cell, the import fails. The error message displayed allows you to locate the problem cell. You can manually resolve the issue from the following information provided in the error message:

- The object number being processed (corresponds to the Excel row)
- The object name
- The property (column) name
- The property (cell) value

Deleting Objects

Objects that you delete from a project are moved to your Architect/Requirements Recycle Bin. These objects remain in the database, and you can restore them until you empty your Recycle Bin. For more information, see [Restoring Objects](#) and [Emptying Your Architect/Requirements Recycle Bin](#), later in this chapter.

You can delete objects from the navigation tree, the hierarchical content table, and the **Attachments** and **Links** tabs and windows. For more information, see [Navigation Tree](#), [Hierarchical Content Table](#), [Attachments Tab](#), [Links Tab](#), and [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.

When you delete a folder, you also delete all of the objects that it contains. When you delete a parent requirement, a parent building block, or a parent note, you also delete all of its children. You can delete children selectively, without deleting the parent.

Trace links for a deleted object are not moved to the Recycle Bin. They are disabled, not deleted, and are re-enabled if the object is restored. Individual links can be deleted without deleting the object. For more information, see [Deleting Trace Links for an Object](#) in chapter 7, *Showing Object Relationships With Trace Links*.



- To delete a folder, you must have **Full** permission for the folder and for all objects in the folder. If the folder contains building blocks, TRAMs, or diagrams, you must also have **Architect** privilege for the project.
- To delete a building block, a TRAM, or a diagram, you must have **Architect** privilege for the project. For a building block or a TRAM, you must also have **Full** permission for that object and for all of its descendants.
- To delete a requirement, you must have **Full** permission for that object and for all of its descendants.

To delete objects:

1. Select the objects that you want to delete.

In the navigation tree, you can select only one folder. In the hierarchical content table, the **Attachments** tab and window, and the **Links** tab and window, you can select one object or group of nonadjacent or adjoining objects.

To see lower level objects in a hierarchical column, click the plus signs. Or, select an object with a plus sign, pull down the **View** menu, and choose **Expand All**. You can also right-click the object and choose **Expand All** from the pop-up menu.

2. Pull down the **Edit** menu and choose **Delete**.

A message asks you to confirm this action.

3. Click **Yes** to move the objects to your Recycle Bin.

Procedure Notes

Step 2: You can also right-click the selection and choose **Delete** from the pop-up menu. Or, click the **Delete** button on the toolbar or press the delete key.

Step 3: You can reverse this action by pulling down the **Edit** menu and choosing **Undo Delete**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Shortcut Objects

- You cannot delete a shortcut object if its parent object is frozen.
- The delete operation does not delete the shortcut from all versions of the parent object. The shortcut is deleted only from the current version of the parent.

Deleting a Child Object

- Deleting a child object, which is also a child of the previous version of the parent, removes the child object from the current version of the parent. The child object is not deleted, and therefore it does not appear in the Recycle Bin. This behavior enables the previous version of the parent to display the child in the currently frozen effectivity.
- Deletion of a child object is not allowed when its parent is frozen. However, this restriction is not applicable if the previous version of the child also has the same parent as the current version child. In this case, deletion is allowed, which in effect removes the current version of the child from the parent.

Restoring Objects

Before you empty your Architect/Requirements Recycle Bin, you can restore objects to the paths from which they were deleted, shown by the **Original Location** property for each object. If a path no longer exists, you can recreate it and move the objects to the new path, or you can move the objects to an existing path.

Each object is restored also to its former state. For example, a folder is restored with all of the objects that it contained when it was deleted. These objects are not displayed in the Recycle Bin and cannot be restored individually. The **Member Count** property shows the number of top level objects, which may themselves have lower level members that are contained in and restored with the displayed folder.

Furthermore, a parent requirement or building block is restored with all of its descendants. These objects are not displayed in the Recycle Bin and cannot be restored individually. The **Member Count** property shows the number of direct children, which may themselves have children that descend from and are restored with the displayed parent.

All notes and trace links for a deleted object are restored with the object. The **Attachment Count** and **Trace Link Count** properties show the number of undisplayed notes and trace links that are restored with the displayed object.



You must have **Modify** permission for the objects and for the intended destinations.

To restore objects:

1. In the navigation tree, select the Recycle Bin.
The content table displays the objects that you deleted from all projects.
2. In the content table, select the objects, right-click the selection, and then choose **Restore From Recycle Bin** from the pop-up menu.
If the former path exists, the objects are returned to that path.



Descendants of a parent whose path has been changed are returned to the parent in the modified path. Descendants of a parent that no longer exists are restored at the next higher level than that from which they were deleted.

If the former path no longer exists, a warning message is displayed. To continue, recreate the path and move the objects to the new path, or move them to an existing path. For more information, see [Creating a Folder](#) or [Moving Objects](#), earlier in this chapter.

Procedure Notes

Step 2: You can select one object or a group of nonadjacent or adjoining objects.

Step 2: You can reverse this action by pulling down the **Edit** menu and choosing **Undo Restore Trash**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Emptying Your Architect/Requirements Recycle Bin

Objects that you deleted from all projects are moved to your Architect/Requirements Recycle Bin. The content table displays these objects sequentially when you select your Recycle Bin in the navigation tree. Although they are removed from their original locations, objects in the Recycle Bin remain in the database and can be restored. For more information, see [Restoring Objects](#), earlier in this chapter.

When you empty your Recycle Bin, Architect/Requirements permanently removes the objects from the database.



Objects cannot be restored after you empty your Recycle Bin.



Your Recycle Bin can be emptied automatically after a number of days specified by your Architect/Requirements enterprise administrator. Also, your enterprise administrator can empty your Recycle Bin manually, for example, to manage storage space in the database. If you have questions about emptying your Recycle Bin, consult your enterprise administrator.

Before emptying your Recycle Bin, you may want to view certain information for verification. For example, you can view the content of a requirement or note by selecting the object in the content table, pulling down the **File** menu, and choosing **Open Read-Only**. You can view a selected object's properties by pulling down the **File** menu and choosing **Properties** to display the Edit Properties dialog window, or by clicking the **Properties** tab.

Also for example, you can click the **Attachments** tab to display the notes for a selected object, or click the **Preview** tab to display the content of a selected requirement. In addition, you can open tabs in floating windows, and extend your verification to objects selected in the notebook pane.



Deleting a child object, which is also a child of the previous version of the parent, removed the child object from the current version of the parent. The child object is not deleted, and therefore it does not appear in the Recycle Bin. This behavior enables the previous version of the parent to display the child in the currently frozen effectivity.

Deletion of a child object is not allowed when its parent is frozen. However, this restriction is not applicable if the previous version of the child also has the same parent as the current version child. In this case, deletion is allowed, which in effect removes the current version of the child from the parent.

To empty your Architect/Requirements Recycle Bin:

1. In the navigation tree, select the Recycle Bin.
The content table displays the objects that you deleted from all projects.
2. Right-click the Recycle Bin and choose **Empty Recycle Bin** from the pop-up menu.
Architect/Requirements displays a message asking you to confirm this action.
3. Click **Yes**.



This action clears the queue for the **Undo** option and cannot be reversed.

Architect/Requirements removes the objects from the database.

Chapter 5: Managing Requirements

This chapter provides an overview of requirements and their management and contains instructions for creating requirements, editing and viewing requirement content, importing requirements from Microsoft Office Word, and promoting and demoting requirements within a folder hierarchy.

Overview of Requirements

The most important aspect of a requirement is its content. In Architect/Requirements, content is created and edited through Microsoft Office Word. Requirement content can consist of familiar elements such as text paragraphs and lists, hyperlinks, tables and graphics, and equations and other special characters. Paragraph formatting can be applied through Word styles and direct formatting options. Character formatting, such as boldface and italics, can be applied as well.

Every requirement resides in a folder. Within each folder, multiple levels of organization allow flexibility in the structure of requirements. For example, all requirements can occupy the folder's highest level. Or, requirements can be structured in a hierarchy, in which the levels convey relationships among the requirements in that folder.

Within a hierarchy, a requirement might occupy the highest level, as a sibling of all other requirements at level 1. Or, a requirement might occupy a lower level, as a child of the requirement at the next higher level. In turn, a child requirement might also be a parent of children at even lower levels.

Working with hierarchical requirements in Architect/Requirements is similar to working with numbered paragraphs in Word's outline view. Each requirement has a paragraph number that corresponds to the requirement's level in the hierarchy. The current level and paragraph number can be changed by promoting or demoting the requirement, or by manually changing the requirement's **Number** property.

To create requirements and content simultaneously, documents can be imported from Word. For analysis or distribution, requirements can be exported to Word, thus capturing both their content and their organization within a folder.

Though a traditional requirements specification can include hundreds or thousands of requirements, it is typically managed as a single document. In Architect/Requirements, however, each requirement is managed as a separate object, with specific properties and access control.

Creating a Requirement Object

Every requirement object must be created in a folder, which can reside directly below the project node or in another folder. Within a folder, requirements can be organized in a hierarchy, with multiple levels of parents, children, and siblings.

You can create a requirement at any level of a folder hierarchy:

- At the top level, in the last position following all other top level objects.
- As a sibling of an existing requirement that you select, directly following the selected requirement within the same level.
- As a child at the next level below an existing requirement that you select, following all other children at that level.

The new requirement object can be demoted to a lower level. It can be promoted if it does not already occupy the top level. For more information, see [Organizing Requirements in a Hierarchy](#), later in this chapter.

After creating a requirement, you can enter rich text content through the Architect/Requirements interface with Microsoft Office Word. Or, you can enter plain text content in Architect/Requirements property tables. You can import Word documents to create requirements and content simultaneously. For more information, see [Entering and Changing Requirement Content in Microsoft Office Word](#), [Editing Requirement Content in Text Property Cells](#), and [Creating Requirements and Content by Import From Microsoft Office Word](#), later in this chapter.

Security profiles can be automatically applied when an object is created. Applying security profile when an object is created prevents users from creating objects of a certain type or with a certain owner. If a new object inherits a security profile from its owner or gets the Instance Security Profile from its type definition, that security profile takes effect when the object is created. If the created users do not have at least write access in the profile, they are not allowed to create the object. If Creator is specified in the profile's Full Control or Modify And Read Access list, any user is able to create the object.



Requirements can be created also in live Excel. For more information, see [Creating Objects in Live Excel](#) in chapter 9, *Working With Object Properties*.

Requirement Subtype Assignment and Inheritance

The new requirement receives certain system-defined properties, including the **Subtype** property.

- For a new top level or a new sibling requirement, you can assign the default subtype, **Requirement**, a system-defined requirement subtype, **Paragraph**, or a user-defined subtype created by your project administrator.
- When you create a sibling of an existing requirement, you can allow it to inherit the subtype of the existing sibling.
- When you create a child of an existing requirement, the child inherits the subtype of the parent.

The new requirement may also receive user-defined properties. The subtype and all other editable properties can be changed after the requirement's creation. For more information, see chapter [Working With Object Properties](#) and appendix [System-Defined Properties in the Systems Engineering and Requirements Management Module](#).

Creating a Requirement at the Top Level of a Folder



You must have **Modify** permission for the folder in which you intend to create the requirement.

1. Select the folder in the navigation tree or in the content table.
2. Assign the subtype to the new requirement by doing one of the following:
 - For the **Requirement** subtype, pull down the **File** menu and choose the **New→Requirement** options, or click the **Create New Requirement** button on the toolbar.
 - For the **Paragraph** subtype, pull down the **File** menu and choose the **New→Paragraph** options, or click the **Create New Paragraph** button on the toolbar.
 - For a user-defined subtype, pull down the **File** menu and choose the **New→Subtype** options to display the Select Subtype dialog window. To see the user-defined subtypes, click the plus sign (+) to the left of **Requirement**, and click any lower level plus signs. Select a subtype and click **OK**.

The content table displays the new requirement as the last object at the top level, with the default name in an open text field.

3. Enter the requirement name, and then press the enter key.

You can reverse this action by pulling down the **Edit** menu and choosing **Undo Rename**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Procedure Notes

Step 2: You can also right-click the folder and choose the options from the pop-up menu. Or, press control-R for the **Requirement** subtype, control-H for the **Paragraph** subtype, or control-U for a user-defined subtype.

Step 2: To reverse this action, you can pull down the **Edit** menu and choose **Undo New Requirement**, **Undo New Paragraph**, or **Undo Create Subtype**. You can also click the **Undo** button on the toolbar, or press control-Z.

Creating a Sibling of an Existing Requirement



You must have **Modify** permission for the folder in which you intend to create the requirement.

1. In the content table, select the requirement that you want the new requirement to follow immediately as a sibling.
2. Assign the subtype to the new requirement by doing one of the following:
 - For the same subtype as the selected sibling, pull down the **File** menu and choose the **New→Requirement** options, or click the **Create New Requirement** button on the toolbar.
 - For the **Paragraph** subtype, pull down the **File** menu and choose the **New→Paragraph** options, or click the **Create New Paragraph** button on the toolbar.



For a user-defined subtype, pull down the **File** menu and choose the **New→Subtype** options to display the Select Subtype dialog window. To see the user-defined subtypes, click the plus sign (+) to the left of **Requirement**, and click any lower level plus signs. Select a subtype and click **OK**.

The content table displays the new requirement as the last object at the selected level, with the default name in an open text field.

3. Enter the requirement name, and then press the enter key.

To place the requirement in position, pull down the **View** menu and choose **Refresh**. Or, right-click the folder in the navigation tree and choose **Refresh** from the pop-up menu. You can also click the **Refresh** button on the toolbar.



If the **Number** property column is displayed, the values change according to the new relationships in the hierarchy.

Procedure Notes

Step 2: You can also right-click the folder or sibling and choose the options from the pop-up menu. Or, press control-R for the **Requirement** subtype, control-H for the **Paragraph** subtype, or control-U for a user-defined subtype.

Step 2: To reverse this action, you can pull down the **Edit** menu and choose **Undo New Requirement**, **Undo New Paragraph**, or **Undo Create Subtype**. You can also click the **Undo** button on the toolbar; or press control-Z.

Step 3: You can reverse this action by pulling down the **Edit** menu and choosing **Undo Rename**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Creating a Child of an Existing Requirement



You must have **Modify** permission for the folder in which you intend to create the requirement.

1. In the hierarchical content table, select the parent requirement.
2. Pull down the **File** menu and choose the **New→Child** options, or click the **Create New Child** button on the toolbar.

The content table displays the new requirement as the last child below the parent, with the default name in an open text field.

3. Enter the requirement name, and then press the enter key.

Procedure Notes

Step 2: You can also right-click the parent and choose the **New→Child** options from the pop-up menu. Or, press control-I. You can reverse this action by pulling down the **Edit** menu and choosing **Undo New Child**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Step 3: You can reverse this action by pulling down the **Edit** menu and choosing **Undo Rename**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Creating Requirements and Content by Import From Microsoft Office Word

Project-related documents, such as requests for proposal, process definitions, and specifications, often contain information that you want to reuse in requirements. By importing a Microsoft Office Word document, you extract its content and create new requirements simultaneously. You can import any file type associated with Word.

Architect/Requirements generates a separate requirement for each paragraph that is formatted in a Word style with a numbered outline level. You can import requirements by outline levels only, assigning the same subtype to all imported requirements. For further granularity, you can specify keywords that occur in the document.



If the document does not begin with a Word outline level style, a note named **Preamble** is generated for the import folder. This note receives all content from the beginning to the first outline level, or the entire content if there is no outline level. The note is in the **Attachments** tab and window.

Outline levels and keywords in the import document produce a hierarchy of requirements in the content table. The plus signs in the **Name** column and the values in the **Number** column show these parent, sibling, and child relationships.

Document Parsing by Outline Levels Only

Architect/Requirements uses Word's numbered outline levels to parse the import document into multiple requirements. For each paragraph that is formatted in an outline level style, level 1 through level 9, a separate requirement is created in the folder that you select.

- The paragraph text becomes the requirement name, for example, the text of a paragraph formatted in a Word heading style
- Any markups or graphics in the paragraph titles are not retained because the titles are imported as plain text. These imported titles become the value of the **Name** property. For example, a comment inserted in the title is not retained.
- All material under the paragraph becomes the requirement content. The paragraph itself does not appear in the content.
- The next outline level style starts another new requirement.

Document Parsing by Outline Levels and Keywords

Within an outline level in the import document, further granularity of requirements can be achieved through keywords. Starting with the first Word outline level style, the parser searches for the keyword or keywords that you specify.



When parsing a Word document into Architect/Requirements using keywords such as **shall**, **will**, special characters such as **&**, **\$**, **#**, ***** are not allowed.

- If no keyword is found before the next outline level, the requirement starts with the preceding outline level and contains all material under that paragraph. The paragraph text is the requirement

name, and the paragraph itself does not appear in the content. The **Paragraph** subtype is assigned by default.

- If only one keyword is found before the next outline level, the requirement starts after the nearest outline level preceding the keyword:
 - The text of the immediately preceding outline level paragraph becomes the requirement name.
 - All material under the outline level paragraph becomes the requirement content. The paragraph itself does not appear in the content.
 - The requirement has the subtype that you assign.
- If two or more keywords are found before the next outline level, the requirements are separated as follows:
 - A paragraph with no content is created. The text of the preceding heading level is used as the name of the paragraph.
 - A requirement or paragraph, depending on the selection, is created based on the sentence breaks. The text preceding the sentence containing the second keyword becomes the first requirement. The **Document Options** property value for the requirement is set to **Disable Numbering**.
 - The subsequent requirements contains the sentence containing the next instance of the keyword and the sentences following it till the next instance of the keyword is found.

These requirements also become the next child of the empty paragraph. Following is an example of the document text containing multiple keywords and the paragraph and requirements created after parsing it.

The headlamp assembly

The headlamp assembly is a composite part. The headlamp assembly *shall* include the lamp, the reflector, the socket, and the lens. It can also include multiple lamps in the same unit. The socket *shall* hold the lamp and connect to the battery. The reflector will position the lamp in the correct position. The lens *shall* focus and diffuse the light such that it provides a good view of the road while driving at night.

 The headlamp assembly	2.1.2	0320	Requirement
 The headlamp assembly	2.1.2.-1	0321	Requirement
 The headlamp assembly	2.1.2.-2	0322	Requirement
 The headlamp assembly	2.1.2.-3	0323	Requirement

Figure 5-1. Parsing a document based on multiple keywords

In the example in figure 5-1, the document is parsed and a paragraph with no content is created. It is identified as **2.1.2**. The following requirements are created:

Requirement number	Requirement text
2.1.2.-1	The headlamp assembly is a composite part. The headlamp assembly shall include the lamp, the reflector, the socket, and the lens. It can also include multiple lamps in the same unit.
2.1.2.-2	The socket shall hold the lamp and connect to the battery. The reflector will position the lamp in the correct position.
2.1.2.-3	The lens shall focus and diffuse the light such that it provides a good view of the road while driving at night.



If multiple keywords are found in the same sentence, they are treated as a single entry and only one requirement is created with the complete sentence.

The name of each requirement is the text of the immediately preceding outline-level paragraph. All requirements have the same subtype, which you assign.

Tables, irrespective of whether they contain the keyword, form a part of the requirement containing the preceding paragraph.

At the next outline level, the parser repeats the processing described above.



The **TcSEPreprocessor.dot** file is required for parsing by keywords. This file is installed in the Architect/Requirements client installation folder. A progress bar indicates the progress of the parsing process. Although the process is optimized for speed, it may take longer for large documents, and particularly for documents with large tables or several tables.

Style Sheet of the Import Folder

The import folder has its own style sheet, which is controlled by the document template that is applied to the folder. The styles in the Word document do not change the folder's style sheet, nor do they affect the imported requirements.

The folder's **Document Template** property shows the current document template. This value is blank if the default document template for the project is applied. You can view or edit that property in the **Properties** tab or floating window. For more information, see [Exporting Objects to Microsoft Office Word](#) in chapter 4, *Maintaining a Project*.

To import requirements from Microsoft Office Word:



- You must have **Modify** permission for the intended import folder.
 - Microsoft Office Word 2013 or Microsoft Office Word 2016 must be installed on your computer.
1. In the navigation tree or the content table, select the import folder, and then pull down the **File** menu and choose the **Import→Word Document** options.

A new Word session starts with the **Open** dialog window.

2.

Select the import document, and then click **Open**.

Word closes and Architect/Requirements displays the Document Import dialog window.

3. Under **Without keyword parsing, import all text as**, select the subtype for the requirements.

For further requirement granularity, you can specify one or more keywords:

- a. Under **Identify Requirement by**, click the **Keyword (eg: shall, will,...)** button.
- b. Enter the keyword or keywords in the text field, with a comma between two keywords.



To enable case sensitivity and wildcard characters, click the **Match Case and Allow Wildcards** button.

A requirement is created for each occurrence of a keyword. The selected subtype is assigned to each requirement.



For each outline level under which a keyword is not found, a requirement is created with the **Paragraph** subtype assigned by default.

4. Click **OK** to start the import.



This action clears the queue for the **Undo** option and cannot be reversed.

A message states that the document is being imported. You can close the message and perform other actions while the import is in progress. When the import is complete, a confirmation message is displayed.



If the **Number** value for an imported requirement contains a dash before a level, for example, **1.-1**, the requirement is an unnumbered part of the parent paragraph at the next higher level. The requirement's **Document Options** property value is **Disable Numbering**. When such a requirement is subsequently exported to Word, it is not assigned a heading or an outline level in the export document.

Procedure Notes

Step 1: You can also right-click the import folder and choose the **Import→Word Document** options from the pop-up menu.

Step 3: You can select the default subtype (**Requirement**), a system-defined requirement subtype (**Paragraph**), or a user-defined subtype. If you have questions about subtypes, consult your project administrator.

Entering and Changing Requirement Content in Microsoft Office Word

When you open a requirement, Architect/Requirements generates a temporary Microsoft Office Word file, which you use to edit the content in the database.



Microsoft Word must be installed on your computer.

For information about version of Microsoft Word, see the Siemens PLM Software Certification Database:

http://www.plm.automation.siemens.com/en_us/support/gtac/certifications.shtml and see the **Teamcenter Systems Engineering Software Certifications** section.



In Word, graphical shapes can be added to a document. These shapes can optionally be placed on a drawing canvas. When a requirement or note containing a canvas is edited with different versions of Word, the canvas may get corrupted. In such cases, the shapes are not displayed correctly.

To avoid this problem, do not use Word drawing canvases in Architect/Requirements requirement and note content. Add the shapes to the Word document directly.

Content Elements

As in a typical Word file, you can enter and change content elements such as:

- Headings, body paragraphs, lists, and hyperlinks.
- Comments and footnotes.
- Tables and graphics.
- Equations, symbols, and other special characters.

You can copy, move, and delete selected portions by the standard Word functions.

Content Formatting

You can apply manual formatting to content in the database through Word's **Format** menu and formatting toolbar.

The Word styles that you can apply are governed by the style sheet for the folder that contains the requirement. That style sheet is specified by the document template that is associated with the containing folder.

The document template also specifies the information that Architect/Requirements extracts from the database when you export data for objects in the folder to Word. For more information, see [Exporting Objects to Microsoft Office Word](#) in chapter 4, *Maintaining a Project*.

The current document template is shown by the folder's **Document Template** property, an editable system-defined property. If the property's value is blank, the default document template for the project is currently applied to the folder.



Before you open the requirement for editing, you can view the containing folder's **Document Template** property in the **Properties** tab or floating window. You can also change that property value to specify different styles for the Word file. For more information, see [Properties Tab](#) or [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*, or chapter 9, [Working With Object Properties](#). If you have questions about document templates or style sheets, consult your project administrator.

Changes to styles in the temporary file do not affect the content in the database. Although you can modify and create styles, and apply them with visible results in Word, such changes are not retained when you close the file. Nor is it effective to modify Word template settings or to attach other templates.

Formatting Lost in Microsoft Word

Your document is likely to lose the Bullets and Numbering formatting when a requirement is exported from Architect/Requirements to Microsoft Word. The problem occurs when the exported document does not contain the styles that are used in the document body. Siemens PLM Software recommends applying style-based formatting on the text to prevent losing the formatting.



Use the **List Number** style or **List Bullets** style for applying numbers or bullets to text instead of directly formatting the text using **Bullets** or **Numbers**.

Page Setup

Page setup modifications have no effect in the database. Changes to margins, paper, layout, or headers and footers are not retained beyond the temporary Word session. The next time you open that requirement, Architect/Requirements applies the initial settings.

Editing Requirement Content in Microsoft Office Word



You must have **Modify** permission for the requirement.

1.

Select the requirement in the content table, the **Links** tab or window, or the **Versions** tab or window.

2. Pull down the **File** menu and choose **Open**.

The requirement opens in an **.mhtml** Word file, in which you enter and change content as described above. This file is deleted from your computer when you exit Architect/Requirements.

3. To save the content in the database, pull down Word's **File** menu and choose **Save**, or click the **Save** button on Word's toolbar.



When the content is saved in the database, editing is disabled in the requirement's **Text** property cell. However, this cell can be reset for direct editing. For more information, see [Enabling Text Property Editing for a Requirement](#) and [Editing Requirement Content in Text Property Cells](#), later in this chapter.



If you choose **Save As**, the content is saved in the database but Word's Save As dialog window is not displayed. Therefore, the file name, file type, and location cannot be changed.

After you close this file, you can create a permanent file with the same content by exporting the requirement to Word. Or, you can open the requirement as a read-only file and choose **Save As** to display Word's Save As dialog window. For more information, see [Viewing Requirement Content](#), later in this chapter.

Procedure Notes

Step 2: You can also right-click the requirement and choose **Open** from the pop-up menu. Or, double-click the requirement's object type indicator.

Using OLE Objects in Word

Architect/Requirements supports Word text content that contains OLE embedded objects. Excel spreadsheets and Visio diagrams can be included in requirements and notes. Content with OLE objects can be created using Word document import or by inserting objects while editing a requirement or note.

When an embedded OLE object is used, the object content is stored in the database and is available to all. The OLE content is available when editing or viewing a single requirement or note. However, the OLE content is not included in document exports. Architect/Requirements does not merge OLE content from different sources into a single exported document. Preview images for the OLE objects are included in the export, however, they cannot be opened as OLE objects.

Microsoft Word's **Link to File** option is not a viable alternative to embedded OLE objects. When a file link is used to reference a file on a local disk drive, then the linked file will not be accessible when accessing the text on another machine. There are also issues even if the referenced file is on a shared drive accessible via the same path name on all user machines. Since Architect/Requirements objects are edited individually, Word generates duplicate identifiers for the file references. Those duplicates become a problem when multiple requirements or notes are later exported together as one Word document.

Referencing Information in Requirement Text Edited in Microsoft Word

There are two ways to reference information in requirement text edited in Microsoft Word.

- By referencing Architect/Requirements object text or properties, described in the [Storing Property Values Through Reference Links](#) section in this manual.
- Using the traditional Microsoft Word caption and bookmark features.

Microsoft Word's captions and bookmarks works correctly in documents where the caption or the definition of a bookmark is in the same document where it is referenced. Requirement text is a large document that is split into many small documents. Microsoft Word does not support references across different documents but you can use some special techniques to create the reference.

The difference between captions and bookmarks are that the caption name specific items such as tables and graphics in the document. A caption is a combination of text and fields. Cross references are created by making a link to the hidden bookmark on the caption. Hidden bookmarks are depicted by underscores preceding them.

The cross reference and caption issue is overcome by using user-defined bookmarks rather than the default hidden bookmarks in a document. The visible bookmarks are then referenced by Architect/Requirements. Following is an example of cross referencing.

Project **A** has folder **One**. The folder contains 3 requirement objects, **REQ-A**, **REQ-B**, and **REQ-C**. You need to create a cross reference in **REQ-C** that references a bookmark in **REQ-A**.

To create a cross reference in a document that references a bookmark in another document:

1. Open **REQ-A** in Microsoft Word and select the object or text being referenced.
2. Select **Insert** → **Bookmark** and enter a name for the bookmark.



Do not begin the bookmark name with an underscore (**_**) as bookmarks beginning with underscores are hidden.

3. Navigate to the end of the open requirement.

4. Select **Insert** → **Cross-Reference**.
5. Select **Reference Type: Bookmark** and **Insert Reference to: Bookmark Text**.
6. Select the required bookmark, click **Insert**, and then click **Close**.
7. Select the inserted cross-reference and select **Home** → **Cut**.
8. Open the document **REQ-C** in Microsoft Word.
9. Navigate to the location in the document where you want the cross-reference and paste the reference using **Home** → **Paste**.
10. Save both **REQ-A** and **REQ-C** and close them.



Opening **REQ-C** displays a reference error. This is due to the bookmark not being defined in the same document. Ignore the error. The error disappears when the complete document is later exported to Microsoft Word and the reference resolves correctly.

11. Right click the folder and generate the final document. The final document has the bookmark and the cross-reference.

Editing Requirement Content in Text Property Cells

Plain text content can be edited directly in the cells of the **Text** property. A column for this property can be added in the hierarchical content table, the **Trace** subtab of the **Links** tab and window, the **Versions** tab and window, and the Search Results dialog window. The **Text** property is displayed by default in the **Properties** tab and window and the Edit Properties dialog window. For more information, see [Adding and Removing Columns](#) in chapter 9, *Working With Object Properties*.

You can use any of these client views to edit the **Text** property for one requirement. For two or more requirements, you can use only the Edit Properties dialog window.



You must have **Modify** permission for each requirement that you intend to edit.



In the content table, you can set the number of lines in the cells. Pull down the **View** menu and choose **Lines Per Row** and the number. Or, click the **#** button on the toolbar and enter the number in the **Lines per row** field.

To edit a Text property cell for one requirement:

In the content table, the **Trace** subtab or window, the **Versions** tab or window, or the Search Results dialog window:

1. With the **Text** property added, double-click the cell that you want to edit.



A shaded background and blue text indicate that cell editing is disabled. The requirement's **Text Format** property value is **MHTML**, for content saved in Microsoft Office Word. Cell editing can be enabled by changing the **Text Format** value to **Text**. For more information, see [Enabling Text Property Editing for a Requirement](#), later in this chapter.

A text field opens in the cell. Scroll bars are provided to the right if the text length exceeds the width of the **Text** column.

2. Enter your changes, and then press the enter key or click outside the field.

In the **Properties** tab or window or the Edit Properties dialog window:



To see more text in the **Text** column, change the width before you begin. Scroll bars are not provided in the cell.

1. Double-click the **Text** property value to open a text field in the cell.



Dimmed text indicates that cell editing is disabled. The requirement's **Text Format** property value is **MHTML**, for content saved in Microsoft Word. Cell editing can be enabled by changing the **Text Format** value to **Text**. For more information, see [Enabling Text Property Editing for a Requirement](#), later in this chapter.

2. Enter your changes, and then press the enter key or click outside the field.

To display the **Properties** tab, select the requirement and click the tab. You can open the **Properties** window by clicking the **Open tab** button on the notebook pane toolbar. To display the Edit Properties dialog window, select the requirement, pull down the **File** menu, and choose **Properties**.

To edit Text property cells for two or more requirements:



The **Text** property cannot be displayed if any one selected requirement has a **Text Format** property value of **MHTML**, for content saved in Microsoft Word. Before you start this procedure, ensure that the **Text Format** value is **Text** for each requirement that you intend to edit. You can set this value if necessary. For more information, see [Enabling Text Property Editing for a Requirement](#), later in this chapter.

1. Select the requirements.

You can select nonadjacent and adjoining requirements of the same subtype or of mixed subtypes.

2. Pull down the **File** menu and choose **Properties**.

The Edit Properties dialog window is displayed.

3. In the **Value** column, double-click the **Text** property cell.

A text field opens in the cell.

4. Enter your changes, and then press the enter key or click outside the field.

5. Click **OK** to close the dialog window.

Procedure Notes

Step 1: You can select the requirements in the content table, the **Links** tab or window, the **Versions** tab or window, or the Search Results dialog window.

Step 2: You can also right-click the selection and choose **Properties** from the pop-up menu.

Enabling Text Property Editing for a Requirement

When editing is disabled in a **Text** property cell, the requirement has a **Text Format** property value of **MHTML**. This value indicates content saved in Microsoft Office Word.

Disabled cell editing is indicated also by certain visual cues in the **Text** property cell. Depending on the client view where the property is displayed, the cues are:

- A shaded background and blue text for disabled cells in the hierarchical content table, the **Trace** subtab of the **Links** tab and window, the **Versions** tab and window, and the Search Results dialog window.
- Dimmed text for disabled cells in the **Properties** tab or window and the Edit Properties dialog window.



Although you can edit disabled **Text** property cells in a live Excel file, edits are not saved in the Architect/Requirements database. An error message is displayed when you click outside the cell. For more information, see [Using the Live Excel Interface](#) in chapter 9, *Working With Object Properties*.

For a requirement with such a cue, you can enable direct editing in the **Text** property cell by changing the **Text Format** property value to **Text**. For more information, see [Editing Requirement Content in Text Property Cells](#), earlier in this chapter.



The next time the cell is edited, current headings, body text, lists, and table text are converted to plain text, and all text is concatenated into one paragraph. All formatting is removed, including graphics, table grids, and other content elements that may be saved in Word. For more information, see [Entering and Changing Requirement Content in Microsoft Office Word](#), earlier in this chapter.

To enable Text property editing for a requirement:

1. Select the requirement, and then do one of the following:
 - To use the **Properties** tab, click the tab. You can open this tab in a floating window by clicking the **Open tab** button on the notebook pane toolbar.
 - To use the Edit Properties dialog window, pull down the **File** menu and choose **Properties**. Or, right-click the requirement and choose **Properties** from the pop-up menu.
2. In the **Value** column, double-click the **Text Format** property value.
3. In the Single-Choice dialog window, check the **Text** check box and click **OK**.



This action changes the text format to plain text and removes all other content elements, effective the next time the cell is edited.

A warning message states that saving as plain text causes all formatting, pictures, and objects in the text to be lost on the next text edit.

4. To continue, click **OK** to close the Single-Choice dialog window.
5. Click **Close** to apply the change and close the Edit Properties dialog window.

Organizing Requirements in a Hierarchy

Requirements in a folder hierarchy can be organized in multiple levels of parents, children, and siblings. The content table shows this hierarchy in the **Name** column, with a plus sign for each requirement that has one or more children. In the **Number** property column, the values indicate each requirement's level in the hierarchy and its sequence within that level, according to the numbered outline style.

You can promote and demote requirements to higher and lower levels, thereby changing their relationships and **Number** property values. For example, you may want to promote or demote requirements after completing certain actions that affect the hierarchy. Such actions include creating and importing requirements, and copying, moving, deleting, and restoring objects. For more information, see [Creating a Requirement Object](#) and [Creating Requirements and Content by Import From Microsoft Office Word](#), earlier in this chapter, and chapter 4, [Maintaining a Project](#).

Positions of Promoted Requirements

A promoted requirement moves to the next higher level, as a sibling of all other requirements at the new level. In addition, all of the requirement's children move up by one level, including direct children and all lower level descendants. For example:

- A requirement previously occupying level 2, with number *n.2*, moves to level 1, with number *n*.
- Children previously occupying level 3, with numbers *n.2.1* and *n.2.2*, move to level 2, with numbers *n.1* and *n.2*.

Requirements cannot be promoted above the folder's top level, shown in the **Number** column by the highest level in the numbered outline style. A top level number has no following decimal point and consists of one or more digits, for example, *n* and *nn*.

Positions of Demoted Requirements

A demoted requirement moves to the next lower level, as a child of the immediately preceding sibling at the previous level. In addition, all of the requirement's children move down by one level, including direct children and all lower level descendants. For example:

- A requirement previously occupying level 2, with number *n.2*, moves to level 3, with number *n.1.1*.
- Children previously occupying level 3, with numbers *n.2.1* and *n.2.2*, move to level 4, with numbers *n.1.1.1* and *n.1.1.2*.

A requirement cannot be demoted if:

- Its **Number** value precedes that of any other requirement at the top level.
- It is a parent requirement's first direct child at any level, for example, numbers *n.1*, *n.n.1*, and *n.n.n.1*.
- Its **Number** value directly follows that of a building block within the current level.

Number Property Values of Promoted and Demoted Requirements

For both promoted and demoted requirements, **Number** property values are changed automatically for successive requirements at the new level and all lower levels. Also, values are changed for requirements that followed the promoted or demoted requirement at the previous level. Because requirements and building blocks can occupy the same level, numbers are changed also for building blocks that follow the promoted or demoted requirement within the new level.



When you change the **Number** property of a requirement from a positive value to a negative value (or vice versa), the requirements are reorganized. However, the **Document Options**→**Disable Numbering** property value does not change with any change in the **Number** value. The behavior of these requirements in export is based on the corresponding **Disable Numbering** values.

To change the **Number** property from a positive value to a negative value (or vice versa), Siemens PLM Software recommends that you use the **Document Options**→**Disable Numbering** property.

You can edit a requirement's **Number** property to promote or demote the requirement manually. For more information, see chapter 9, [Working With Object Properties](#).

Promoting and Demoting Requirements



- To promote a requirement to the folder's top level, you must have **Modify** permission for the requirement and for the folder. For any lower level, you must have **Modify** permission for the requirement and for the intended parent requirement.
 - To demote a requirement, you must have **Modify** permission for the requirement and for the intended parent requirement.
1. In the hierarchical content table, select each requirement that you want to promote or demote.
You can select nonadjacent and adjoining requirements.
 2. Pull down the **Edit** menu and choose **Promote** or **Demote**, or click the **Promote** or **Demote** button on the toolbar.

The content table displays each selected requirement at its new level. In the **Name** column, plus signs or minus signs are displayed or removed for affected parent, sibling, and child relationships. In the **Number** column, values are changed according to the new relationships.

Procedure Notes

Step 2: You can also right-click the selection and choose **Promote** or **Demote** from the pop-up menu. In addition, you can point to the selection, hold down the left mouse button, and drop the selection within the new level.

Step 2: You can reverse this action by pulling down the **Edit** menu and choosing **Undo Promote** or **Undo Demote**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Viewing Requirement Content

There are two ways to view the content of a requirement:

- In a read-only Microsoft Word file, you can view and print the content, and you can send it to E-mail and fax recipients. You cannot change the content in the database, and therefore you do not interfere with someone else's work.



Microsoft Office must be installed on your computer.

- In the **Preview** tab or window, you can view and print the content without opening the requirement in Word. You cannot change the content in the database. For more information, see [Preview Tab](#) or [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.

You can use Word to open requirements selected in the hierarchical content table, the **Links** tab or window, and the **Versions** tab or window. For requirements selected in the tabs and in your Architect/Requirements Recycle Bin, you can view content in the **Preview** tab or window. However, requirements in your Recycle Bin cannot be opened in Word. For more information, see [Hierarchical Content Table](#), [Architect/Requirements Recycle Bin](#), [Links Tab](#), and [Versions Tab](#) in chapter 3, *Using the Architect/Requirements Main Window*.

To view requirement content in Microsoft Word:

Select the requirement, pull down the **File** menu, and choose **Open Read-Only**. Or, right-click the requirement and choose **Open Read-Only** from the pop-up menu.

The requirement opens in an **.mhtml** Word file. This temporary file is deleted when you exit Architect/Requirements.

To view requirement content in the Preview tab or window:

Do one of the following:

- For a requirement in the hierarchical content table or the Recycle Bin, select the requirement and click the **Preview** tab. You can open the **Preview** window for this requirement by clicking the **Open tab** button on the notebook pane's toolbar.
- For a requirement in the **Links** or **Versions** tab:
 - With the **Preview** tab on top in the notebook pane, click the **Open tab** button on the toolbar to open the **Preview** window.
 - Select the requirement in the **Links** or **Versions** tab to display the content in the **Preview** window.

Comparing the Content of Different Requirements

Architect/Requirements works with the **Compare and Merge Documents** option in Microsoft Office Word to give you features for comparing requirement content. The comparison can be between two selected requirements, or between the requirements in two selected folders. The requirements or folders can reside in the same project or in different projects.

When you compare two requirements, you can choose to view only the content of a selected requirement. Or, you can choose to include the content of all direct children and lower-level descendants; this deep content is exported to a single Word document.

A comparison of two folders shows you the content of all requirements in each requirement hierarchy. The deep content of each folder's parent and child requirements is exported to a single Word document.

Setting the Comparison Mode

The default comparison mode uses Word's Legal blackline option. In this mode, the first requirement is opened in Word and the second requirement is merged into the first document. Then, the results are shown in a new, third Word document.



In the Legal blackline mode, only the first requirement's content can be edited. The second requirement is not opened in Word.

You can also set a comparison mode that uses Word's side by side option. In this mode, each document is opened in a separate Word window. By choosing **Compare Side by Side with** from Word's **Window** menu, you can tile the documents vertically for comparison. A common scroll bar synchronizes the content.



In the side by side mode, the content of each requirement can be edited if the document is generated from only one level of the hierarchy. Content from a deep comparison cannot be edited.

- To switch between modes, pull down the **Tools** menu and choose the **Compare**→**Compare Side by Side** options.

The side by side mode is in effect when a checkmark appears beside these options.

The Legal blackline mode is in effect when no checkmark appears.



When the side by side comparison is turned on, the difference in the documents is not visible.



The effective mode persists until you switch to the other mode or you exit Architect/Requirements. The Legal blackline mode is effective automatically when you start a new Architect/Requirements session.

For more information, see Microsoft Office Word Help.

Comparing Two Requirements

1. In the content table or the **Versions** tab or window, select the first requirement.
2. Pull down the **Tools** menu and choose one of the following options:
 - Choose **Compare**→**Start Compare 1st Level** to compare only the selected requirement.
 - Choose **Compare**→**Start Compare Deep** to compare the selected requirement and all of its direct children and lower-level descendants.
3. In the content table or the **Versions** tab or window, select the requirement that you want to compare with the first requirement.

You can select a requirement in the same project or in a different project.

4. Pull down the **Tools** menu and choose one of the following options:
 - Choose **Compare**→**End Compare 1st Level** to compare only the selected requirement.
 -

Choose **Compare**→**End Compare Deep** to compare the selected requirement and all of its direct children and lower level descendants.

The Word export process begins with the Select Document Template dialog window.

- To use the selected document template for all objects, click **OK**.
- To change the stylesheet for the document or the object template for one or more types, click **More**.

The Document Template dialog window is displayed.

- o Do one or both of the following:
 - . To change the Word styles that are applied to all data in the document, select the style sheet in the **Select Stylesheet** field.
 - . To change the object template for a type in the **SubTypes** column:
 - Double-click the cell in the **OverrideObjectTemplate** column to display the **Single-Choice** dialog window.
 - Check the check box for the object template that you want to use, and then click **OK** to close this dialog window.

Repeat these steps for each additional type whose object template you want to change.

- o Click **Export**.

The requirements open in the selected Word comparison mode. Use Word's comparison features to view the content. For more information, see the Microsoft Office Word Help.

Comparing the Requirements in Two Folders

1. In the content table, select the first folder.
2. Pull down the **Tools** menu and choose **Compare**→**Start Compare Deep**.

This option exports to Word the entire requirement hierarchy in the folder.

3. In the content table, select the folder containing the requirements that you want to compare with those in the first folder.

You can select a folder in the same project or in a different project.

4.

Pull down the **Tools** menu and choose **Compare**→**End Compare Deep**.

The Select Document Template dialog window is displayed. Here, you associate the template for exporting the first folder's requirements to Word.

- To use the selected document template for all objects, click **OK**.
- To change the stylesheet for the document or the object template for one or more types, click **More**.

The Document Template dialog window is displayed.

. Do one or both of the following:

- o To change the Word styles that are applied to all data in the document, select the style sheet in the **Select Stylesheet** field.
- o To change the object template for a type in the **SubTypes** column:

a.

Double-click the cell in the **OverrideObjectTemplate** column to display the **Single-Choice** dialog window.

- a. Check the check box for the object template that you want to use, and then click **OK** to close this dialog window.

Repeat these steps for each additional type whose object template you want to change.

. Click **Export**.

The Select Document Template dialog window is redisplayed. Here, you associate the template for exporting the second folder's requirements to Word.

- o Repeat the steps listed for the first folder above.

The exported requirements open in the selected Word comparison mode. Use Word's comparison features to view the content. For more information, see the Microsoft Office Word Help.

Chapter 6: Constructing System Views With Building Blocks and Diagrams

This chapter discusses methods of constructing views for system decomposition. Instructions are provided for using building blocks and live Visio, the Architect/Requirements interface with Microsoft Office Visio. **Architect** privilege is required for most of these procedures.

Building Blocks

With building blocks, you can construct hierarchies that decompose systems and illustrate relationships among system elements. A building block can represent any element in a hierarchy, such as a function or a component of a product, a task in a work breakdown structure, or a job function in an organizational chart.

Building blocks reside in folders, and a folder can contain any number of building blocks. Below each top level building block, subordinate building blocks can be organized in multiple levels of parents, children, and siblings.

For example, to construct a view of an entire system, create a single building block for the system at the top level. At the next lower level, create child building blocks for the major subsystems. Below those, continue to decompose the system into finer levels of detail.

In such a hierarchy, building blocks can be created at specific levels, promoted to higher levels, demoted to lower levels, and moved up or down within a level. For each building block, the **Number** property value shows the level in the hierarchy and the position within the level. As the structure changes, values are renumbered automatically for affected building blocks.

Trace links can be created to and from building blocks, to show defining and complying relationships with other objects. Such relationships may exist between building blocks that reside within the same hierarchy. Also, building blocks may have trace links to objects that reside elsewhere in the same Architect/Requirements project, in a different project, or in a different Teamcenter product.

A system-defined building block subtype, **TRAM**, can be used for *transitional mapping*, a method of interrelating system views for comparison and analysis. Through trace links, a TRAM can be associated with building block hierarchies to create a flow of information among source views and destination views, with the TRAM as the focal point.

Any number of source views can be linked to a TRAM, with the peak building blocks as defining objects. From a TRAM, peak building blocks in any number of destination views can be linked as complying objects. When the TRAM is selected, these defining and complying paths can be followed to evaluate relationships among the systems. For more information, see [Creating Trace Links](#) and [Viewing Objects in a Defining or Complying Path](#) in chapter 7, *Showing Object Relationships With Trace Links*.

Creating a Building Block

You can create a building block at any level of a folder hierarchy:

- At the top level, in the last position following all other top level objects.
- As a sibling of an existing building block that you select, directly following the selected building block within the same level.
- As a child at the next level below an existing building block that you select, following all other children at that level.

After its creation, the new building block can be demoted to a lower level, and it can be promoted if it does not already occupy the top level. For more information, see [Organizing Building Blocks in a Hierarchy](#), later in this chapter.

The new building block receives certain system-defined properties, including the **Subtype** property. When you create a child of an existing building block, the child inherits the subtype of the parent. When you create a sibling of an existing building block, you can allow it to inherit the subtype of the existing sibling, or you can assign the subtype as you do when you create a building block at the top level. For a new top level or a new sibling building block, you can assign the default subtype (**Building Block**), a system-defined building block subtype (**TRAM**), or a user-defined subtype created by your project administrator.

The new building block may also receive user-defined properties. The subtype and other editable properties can be changed after the building block's creation. For more information, see chapter 9, [Working With Object Properties](#) and appendix [System-Defined Properties in the Systems Engineering and Requirements Management Module](#).

After you create the building block, you can attach diagrams through live Visio, the Architect/Requirements interface with Microsoft Office Visio. For more information, see [Creating a Diagram](#), later in this chapter.

Before or after you attach the diagram, you can copy the new building block to other locations, link it to other objects, attach notes, and create versions and variants. For more information, see [Copying Objects](#) in chapter 4, *Maintaining a Project*; [Creating Trace Links](#) and [Linking to an Object in Another Teamcenter Product](#) in chapter 7, *Showing Object Relationships With Trace Links*; [Attaching a Note to an Object](#) in chapter 8, *Recording Supplementary Information With Notes*; and [Creating Versions](#) or [Creating Variants](#) in chapter 10, *Working With Versions*.

Building blocks can also be created in the live Excel interface. For more information, see [Creating Objects in Live Excel](#) in chapter 9, *Working With Object Properties*.



- You must have **Architect** privilege for the project.
- You must have **Modify** permission for the folder in which you intend to create the building block.

To create a building block at the top level of a folder:

1. Select the folder in the navigation tree or in the hierarchical content table.
2. Assign the subtype to the new building block by doing one of the following:
 - For the **Building Block** subtype, pull down the **File** menu and choose the **New→Building Block** options, or click the **Create New Building Block** button on the toolbar.
 - For the **TRAM** subtype, pull down the **File** menu and choose the **New→TRAM** options.
 - For a user-defined subtype, pull down the **File** menu and choose the **New→Subtype** options to display the Select Subtype dialog window. To see the user-defined subtypes, click the plus sign (+) to the left of **Building Block**, and click any lower level plus signs. Select a subtype and click **OK**.

The content table displays the new building block as the last object at the top level, with the default name in an open text field.

3. Enter the building block name, and then press the enter key.

Procedure Notes

Step 2: You can also right-click the folder or sibling and choose the options from the pop-up menu. Or, press control-B for the **Building Block** subtype or control-U for a user-defined subtype. To reverse this action, you can pull down the **Edit** menu and choose **Undo New Building Block**, **Undo New TRAM**, or **Undo Create Subtype**; click the **Undo** button on the toolbar; or press control-Z.

Step 3: You can reverse this action by pulling down the **Edit** menu and choosing **Undo Rename**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

To create a sibling of an existing building block:

1. In the hierarchical content table, select the building block that you want the new building block to follow immediately as a sibling.
2. Assign the subtype to the new building block by doing one of the following:
 - For the same subtype as the selected sibling, pull down the **File** menu and choose the **New→Building Block** options, or click the **Create Building Block** button on the toolbar.
 - For the **TRAM** subtype, pull down the **File** menu and choose the **New→TRAM** options.
 - For a user-defined subtype, pull down the **File** menu and choose the **New→Subtype** options to display the Select Subtype dialog window. To see the user-defined subtypes, click the plus sign (+) to the left of **Building Block**, and click any lower level plus signs. Select a subtype and click **OK**.

The content table displays the new building block as the last object at the selected level, with the default name in an open text field.

3. Enter the building block name, and then press the enter key.

To place the building block in position, pull down the **View** menu and choose **Refresh**. Or, right-click the folder in the navigation tree and choose **Refresh** from the pop-up menu. You can also click the **Refresh** button on the toolbar.

Procedure Notes

Step 2: You can also right-click the folder or sibling and choose the options from the pop-up menu. Or, press control-B for the **Building Block** subtype or control-U for a user-defined subtype. To reverse this action, you can pull down the **Edit** menu and choose **Undo New Building Block**, **Undo New TRAM**, or **Undo Create Subtype**; click the **Undo** button on the toolbar; or press control-Z.

Step 3: You can reverse this action by pulling down the **Edit** menu and choosing **Undo Rename**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

To create a child of an existing building block:

1. In the hierarchical content table, select the parent building block.
2. Pull down the **File** menu and choose the **New**→**Child** options, or click the **Create New Child** button on the toolbar.

The content table displays the new building block as the last child below the parent, with the default name in an open text field.

3. Enter the building block name in the text field, and then press the enter key.

To place the building block in position, pull down the **View** menu and choose **Refresh**. Or, right-click the folder in the navigation tree and choose **Refresh** from the pop-up menu. You can also click the **Refresh** button on the toolbar.

Procedure Notes

Step 2: You can also right-click the parent and choose the **New**→**Child** options from the pop-up menu. Or, press control-I. You can reverse this action by pulling down the **Edit** menu and choosing **Undo New Child**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Step 3: You can reverse this action by pulling down the **Edit** menu and choosing **Undo Rename**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Organizing Building Blocks in a Hierarchy

Within a folder, building blocks can be organized in multiple levels of parents, children, and siblings. The content table shows this hierarchy in the **Name** column, with a plus sign for each building block that has one or more members at lower levels. In the **Number** property column, the values indicate each building block's level in the hierarchy and its sequence within that level, according to the numbered outline style.

To reorganize the hierarchy, you can promote and demote building blocks to higher and lower levels, which changes their relationships and **Number** property values. For example, you may want to promote or demote building blocks after completing certain actions that affect the hierarchy, such as creating building blocks and copying, moving, deleting, and restoring objects. For more information, see [Creating a Building Block](#), earlier in this chapter, and chapter 4, [Maintaining a Project](#).

A promoted building block moves to the next higher level, as a sibling of all other building blocks at the new level. In addition, all of the building block's members move up by one level, including direct children and all lower level descendants. For example:

- A building block previously occupying level 2, with number $n.2$, moves to level 1, with number n .
- Children previously occupying level 3, with numbers $n.2.1$ and $n.2.2$, move to level 2, with numbers $n.1$ and $n.2$.

A demoted building block moves to the next lower level, as a child of the immediately preceding sibling at the previous level. In addition, all of the building block's members move down by one level, including direct children and all lower level descendants. For example:

- A building block previously occupying level 2, with number $n.2$, moves to level 3, with number $n.1.1$.
- Children previously occupying level 3, with numbers $n.2.1$ and $n.2.2$, move to level 4, with numbers $n.1.1.1$ and $n.1.1.2$.

In both cases, **Number** property values change for successive building blocks at the new level and all lower levels. Also, values change for building blocks that followed the promoted or demoted building block at the previous level. Because building blocks and requirements can occupy the same level, numbers change also for requirements that follow the building block within the new level.

Building blocks cannot be promoted above the folder's top level, shown in the **Number** column by the highest level in the numbered outline style. A top-level number has no following decimal point and consists of one or more digits, for example, n and nm .

A building block cannot be demoted if:

- Its **Number** value precedes that of any other building block at the top level.
- It is a parent building block's first direct child at any level, for example, numbers $n.1$, $n.n.1$, and $n.n.n.1$.
- Its **Number** value directly follows that of a requirement within the current level.

You can also promote or demote a building block by changing its value in the **Number** column. For more information, see [Editing the Properties of a Selected Object](#), [Editing Properties in Table View Cells](#), or [Using the Live Excel Interface](#) in chapter 9, *Working With Object Properties*.



- You must have **Architect** privilege for the project.
- To promote a building block to the folder's top level, you must have **Modify** permission for the building block and for the folder. For any lower level, you must have **Modify** permission for the building block and for the intended parent building block.
- To demote a building block, you must have **Modify** permission for the building block and for the intended parent building block.

To organize building blocks in a hierarchy:

1. In the hierarchical content table, select each building block that you want to promote or demote.
You can select nonadjacent building blocks by holding down the control key while you click the building blocks. To select adjoining building blocks, click the first building block, hold down the shift key, and click the last building block.
2. Pull down the **Edit** menu and choose **Promote** or **Demote**, or click the **Promote** or **Demote** button on the toolbar.

The content table displays each selected building block at its new level. In the **Name** column, plus signs or minus signs are displayed or removed for affected parent, sibling, and child relationships. In the **Number** column, values are changed according to the new relationships.

To reposition building blocks within their new levels, pull down the **View** menu and choose **Refresh**. Or, right-click the folder in the navigation tree and choose **Refresh** from the pop-up menu. You can also click the **Refresh** button on the toolbar.

Procedure Notes

Step 1: To see lower level objects, click the plus signs in the **Name** column. Or, select an object with a plus sign, pull down the **View** menu, and choose **Expand All**. You can also right-click an object and choose **Expand All** from the pop-up menu.

Step 2: You can also right-click the selection and choose **Promote** or **Demote** from the pop-up menu. In addition, you can point to the selection, hold down the left mouse button, and drop the selection within the new level.

Step 2: You can reverse this action by pulling down the **Edit** menu and choosing **Undo Promote** or **Undo Demote**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Live Visio Diagrams

Typically, diagrams are attached to building blocks, which can represent superior and subordinate elements of a system view. Diagrams can be attached also to folders, requirements, and groups.

Diagrams are created and edited in Architect/Requirements through the live Visio interface with Microsoft Office Visio. Each live Visio diagram is synchronized with the database and has a special stencil that contains shapes representing the object types. The live Visio stencil can be used to create, modify, and delete the following:

- Objects in the Architect/Requirements client. The stencil includes shapes for folders, requirements, building blocks, groups, notes, spreadsheets, and diagrams.
- Trace links between objects represented in the diagram. Defining and complying trace links can be managed from the diagram. Trace links can be viewed in the **Trace** subtab of the **Links** tab and window.
- Connections between objects represented in the diagram. Connections can show any type of relationship, such as a physical relationship, a functional relationship, a data flow, or a control flow. Connections can be viewed in the **Relations** subtab of the **Links** tab and window.



While copying from a source diagram, to preserve the layout and retain the members in a different diagram, you can create a new diagram and copy the source diagram to the new diagram using the **Copy** and **Paste from TcSE** commands.

Every member in a diagram is linked to the diagram. When you move a member out of the owner of the diagram (for example, to a folder), the link with the diagram is removed.

Similarly, if the diagram is moved to another owner (for example, to a new folder), the link with its owner is removed.

In the above cases, the Visio diagram deletes all its contents while synchronizing the diagram because the old links are not present. All the members of the new owner of the diagram (in this case the new folder) are created in the diagram while synchronizing the diagram.

Properties of Architect/Requirements objects are mapped to the **Text** properties of Visio shapes through the live Visio stencil. For more information about customizing a stencil, see the *Systems Architect/Requirements Management Project Administrator's Manual*.



- Microsoft Office Visio 2013 or Visio 2016 must be installed.

Microsoft .NET Framework 2.0 and 3.0 or 3.5 must be installed on your computer.

- To change editable property values for objects, you must have the **Modify** permission for the objects and the **Architect** privilege for the project.
- This section assumes that you have experience with Microsoft Office Visio.

User Actions in Live Visio Diagrams

You can create, view, and edit live Visio diagrams using the live Visio interface. This sections lists the various user actions available as pop-up menu commands in the live Visio diagram.

When a live Visio diagram is opened in the Microsoft Office Visio, right-click on the diagram and choose from any of the following pop-up menu commands:

- **Paste From TcSE** to paste the content that was copied from the Architect/Requirements client to the live Visio diagram.
- **Paste and Create In TcSE** to copy any existing object from the same or a different live Visio diagram and create a new object of the same type in the live Visio diagram as well as in the Architect/Requirements client.
- **Make Visio Diagram Non Live** to convert a live Visio diagram to a non-live Visio diagram. After making the diagram non-live, it would not connect with the Architect/Requirements client. You can use this command on a saved diagram on a local disk.
- **Get Connection From Visio Diagram** to copy the existing connection on a live Visio diagram and create a new connection on an existing or a new diagram. To use this command, the two ends to the connection must exist on the diagram.
- **Refresh Entire Visio Diagram** to synchronize the Visio diagram from the Architect/Requirements server. The connections that are not added from Visio are not synchronized.
- **Refresh With Connection** to synchronize the Visio diagram from the Architect/Requirements server with the connections that are not added from Visio.
- **Select Non TcSE Shapes** to select the non-Architect/Requirements shapes on a live Visio diagram.
- **Go To Parent Diagram** to open the parent diagram, if it exists.
- **Hide/Unhide Port** to hide or unhide the port shape on a live Visio diagram.
- **Hide/Unhide Port Name** to hide or unhide the port shape name on a live Visio diagram.
- **Paste as Tunnel Port** to create the port as a tunnel port on a diagram. However, only an external port shape can be pasted as a tunnel port. No new objects are added in the database.
- **Hide/Unhide Connection Name** to hide or unhide the connection shape name on a live Visio diagram.
- **Hide/Unhide Marked Connection Name** to hide or unhide the marked connection name of the shape on a live Visio diagram. Marked connections are the connections selected by using the **Hide/Unhide Connection Name** command.
- **Reset Marked Connection** to reset the marked state of connection to unmarked state.
- **Upgrade Diagram** to recreate the right-click menu options on each shape present on the Visio diagram. **Upgrade Diagram** should be used only if you are working on a Visio diagram that was created with Architect/Requirements 2007.0 or earlier. This action may take considerable time depending on size of the Visio Diagram that you are working with.

When a live Visio diagram is opened in the Microsoft Office Visio, right-click a shape (an object) and choose from any of the following pop-up menu commands.

- **TcSE Properties** to open a dialog window displaying all the properties of the object in Architect/Requirements.
- **Start Link** to start the trace link from the selected object.
- **End Link** to end the trace link at the selected object.
- **End Link Using Subtype** to end the trace link at the selected object as the chosen subtype.
- **Go To TcSE** to navigate to the corresponding object in the Architect/Requirements client.

- **Delete in TcSE** to delete the corresponding object in the Architect/Requirements database.

When a live Visio diagram is opened in Microsoft Office Visio, right-click a shape (an object) and choose from any of the following pop-up menu commands, depending on the object type of the shape.

- **Open Child Diagram** to open or create a live Visio diagram on the selected object. This command is available only on the building block shape.
- **Open Child Diagram w/Stencil** to open or create a live Visio diagram on the selected object using the selected stencil. This command is available only on the building block shape. If no stencil is selected, the stencil of the parent diagram is chosen.
- **Reverse Direction** to reverse the direction of a connection. This is available only on the connection shape.
- **Copy Connection** to copy the connection to the clipboard from a live Visio diagram. This command should be used along with the **Get Connection From Visio Diagram** command.
- **Set Data Definition** to associate a data definition with a connection or a port shape. This command is available only on a connection and a port shape.
- **Unassign Data Definition** to remove a data definition association with a connection or a port shape. This command is available only on a connection and a port shape.
- **Copy Port** to copy and create a tunnel port using the **Paste as Tunnel Port** command. The **Copy Port** command is available only on an external port shape.

To work with ports and connections in a live Visio diagram, you can do any of the following. For more information, see [Ports and Connections](#) later in this chapter.

- Create a port by selecting the port shape from the stencil and attaching it to a building block connection point. On successful attachment, a port object is created in the Architect/Requirements database and the building block's property is updated on the live Visio diagram.
- Move a port by selecting the port shape and move it on the same owner (building block). Moving a port to a different owner is not permitted. Also, if you have configured the port to be attached at any specific location on a building block, then the move honors the configuration on the property mapping XML file.

Moving a port having a connection attached to it moves the connection along with the port.

- Create a connection by selecting the connection shape from the stencil and attaching it on any two building blocks. It creates ports on the respective building blocks and creates connection between them. Ports and connection objects are created in the Architect/Requirements database.
- Create connection between ports by selecting the connection shape from the stencil and attaching it to any two ports. It creates the connection in the Architect/Requirements database and synchronizes the connection's properties in the Microsoft Office Visio.
- Move a connection by changing its two ends. You can detach one end of the connection and attach it to any other shape (building block). The connection between two ends is updated in the Architect/Requirements database.
- You cannot detach a port from a building block. If it is no longer required, you can delete it using the **Delete in TcSE** command.
- You cannot detach a connection from a building block or a port, and a warning message is displayed if you try. If it is no longer required, you can delete it using the **Delete in TcSE** command.

Stencil Diagnostic Utility

Stencil diagnostic utility is available as the following three commands when you right-click on a live Visio diagram:

- **Identify Subshape** to identify the subshapes in a group master shape. You can also identify the sequence in which you have added shapes to a group master shape.
- **Diagnose Connection Point** to diagnose the connection point section on a master shape.
- **Verify Prop Map XML And Stencil** to verify the property mapping XML file, the stencil file master shape mapping, and the availability of the mapped objects in Architect/Requirements. For more information about the property mapping file, see the *Systems Architect/Requirements Management Project Administrator's Manual*.

Creating a Diagram

Diagrams can be attached to building blocks, folders, requirements, and groups. For the object that you select, the live Visio interface generates a diagram containing a shape for each member of the object. Members of a folder, a requirement, or a building block are its direct children. Members of a group are objects that belong to the group while residing in their native locations. For an object of any of those types, the members are shown at the next lower level. For more information, see [Viewing Objects in a Hierarchy](#) in chapter 3, *Using the Architect/Requirements Main Window*.

The object to which you attach the diagram is the diagram owner. A shape for that object does not appear in the diagram. You can select the owner in the hierarchical content table or in either subtab of the **Links** tab or window. In the **Trace** subtab, you can select from either pane. In the **Relations** subtab, you can select from either object table but not from the links tables. For more information, see [Hierarchical Content Table](#), [Links Tab](#), [Trace Subtab](#), [Relations Subtab](#), or [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.

Existing connections between the members are included in the diagram. Connections are attached correctly on each member, and you can adjust them for the desired appearance. Connections can be viewed in the **Relations** subtab. For more information, see [Working With Connections in a Live Visio Diagram](#) and [Viewing the Connections for an Object](#), later in this chapter.

Existing trace links between the members are not included in the diagram. However, existing trace links can be added to diagrams, and new trace links can be created from Visio. For more information, see [Working With Trace Links in a Live Visio Diagram](#), later in this chapter.

The diagram object is added to the Architect/Requirements database and is displayed in the **Attachments** tab or window. For more information, see [Attachments Tab](#) in chapter 3, *Using the Architect/Requirements Main Window*.



You must have **Architect** privilege for the project.

To create a diagram:

1.

Select the diagram owner, and then pull down the **File** menu and choose the **Visio Live**→**Create Diagram** options.

The Diagram Inputs dialog window is displayed.

2. Do one or both of the following:

- Check the check box for each stencil whose shapes you want to include in the diagram. You can check all check boxes by clicking **Select All**.
- Clear the check box for each stencil whose shapes you want to exclude from the diagram. You can clear all check boxes by clicking **Unselect All**.



To use ports in the diagram, you must select a stencil that supports ports, such as the default **Port Type Stencil**.

You can also do the following:

- In the **Diagram Subtype** field, select a subtype to assign to the diagram.
- In the **Data Dictionary** field, select a data dictionary to attach to the diagram.

3. Click **OK**.

The diagram opens in a live Visio window, containing a shape for each member of the owner. The shapes are automatically arranged on the Visio page so that all are visible with no overlapping.



When you first save the diagram, if its **Copy Snapshot** property value is **Yes**, Architect/Requirements automatically attaches a note to the diagram. Named **Diagram Image** by default, this note contains an image (.gif) of the diagram. The diagram image note is updated each time changes to the diagram are saved. Through this note, the diagram's full content can be targeted by reference links.

Generating the note and updating the image can add time in saving a diagram. Siemens PLM Software recommends that you leave **Copy Snapshot** set to **No**, the default, until just before you create the first reference link to the diagram image note. For more information, see [Creating a Full Content Reference Link](#) in chapter 9, *Working With Object Properties*.

To disable updates to the image note, you can change **Copy Snapshot** from **Yes** to **No**. For more information, see [Editing the Properties of a Selected Object](#) in chapter 9, *Working With Object Properties*.

You can open the diagram image note in Word and resize the image. The new dimensions are applied to the image each time it is updated from Visio. You may want to resize the image periodically for consistency with the Visio diagram layout.

To save the diagram outside the Architect/Requirements database, click the Visio **File** menu and choose **Save As** to display the **Save As** dialog window, where you can store the diagram on a local or shared drive. Later, you can open the diagram and synchronize it with the database by pulling down the Architect/Requirements **File** menu and choosing the **Visio Live**→**Open** options to display the **Open** dialog window.



A live Visio diagram that is stored outside the database can be converted to a standard Visio file, one that is static and disconnected from the database. For more information, see [Disconnecting a Live Visio Diagram](#), later in this chapter.

Procedure Notes

Step 1: Except in the **Links** subtabs and windows, you can right-click the diagram owner and choose the **Visio Live**→**Create Diagram** options from the pop-up menu.

Creating a Child Diagram

A child diagram is a secondary part of a live Visio diagram. You can create additional diagrams using the stencils in the database to create a more defined structure or picture of the information needed in the diagram. If you want to use ports and paths, use a stencil that contains these objects.



By default, a child diagram inherits the stencils and data dictionary of the parent diagram.

To create a child diagram:

1. Select a shape in Visio that represents a building block that is a child of the diagram owner.
2. Right-click the object and select **Open Child Diagram** from the pop-up menu.

The object opens in a new drawing window. If the stencil set of the parent object is not visible, it is in the background of the child diagram and you must resize the new window to use the drawing tools of the parent. Any ports from the parent are visible in the child diagram when it is first opened.

Editing a Diagram

Live Visio diagrams are synchronized with the Architect/Requirements database. When members of a diagram owner are added, modified, or deleted in the Architect/Requirements client, the diagram is dynamically updated with those changes. Conversely, objects are automatically added, modified, or deleted in the client when those actions are done for corresponding shapes in the diagram.

You can use shapes from multiple stencils in the same diagram, although Architect/Requirements stencils contain the only shapes that apply to the database. If you add, modify, or delete shapes from other stencils, such as those provided with Microsoft Office Visio, the database is not affected. Shapes from non-Architect/Requirements stencils are saved within the diagram itself.

To display live Visio shapes, pull down the Visio **Window** menu and choose the Architect/Requirements stencils that you want to use.



- The arrangement of shapes in the diagram does not reflect the hierarchy or the display order of the objects in the Architect/Requirements client. Shapes can be placed anywhere on the Visio page without affecting the objects.
- For the objects represented by the shapes, you can create new diagrams in additional Visio windows. By right-clicking a shape and choosing **Create New Diagram** from the pop-up menu, you open a diagram containing a shape for each member of the object, which becomes the diagram owner. You can rename the diagram in the **Attachments** tab or window.

•

To navigate to an object in the Architect/Requirements client, you can right-click the corresponding shape in the diagram and choose **Go To TcSE** from the pop-up menu.

- A given Architect/Requirements object can appear any number of times in a live Visio diagram.

When the diagram owner is selected, its diagrams are displayed in the **Attachments** tab or window. For a selected member of the owner, the **Where Used** tab or window displays each diagram that contains a shape representing the member object. You can open a diagram for editing from either view. Also, you can open and edit a live Visio diagram that was saved locally on your computer or on a shared drive. For more information, see [Attachments Tab](#), [Where Used Tab](#), or [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.



You must have **Architect** privilege for the project.

To edit a diagram:

1. Open the diagram by doing one of the following:
 - For a diagram displayed in the **Attachments** or **Where Used** tab or window, select the diagram, and then pull down the **File** menu and choose **Open**.

Live Visio opens with a message stating that the diagram is being refreshed. When the message closes, the diagram is synchronized with the database.

If members of the owner were added, modified, or deleted since the diagram was last saved, the diagram is updated with the changes. By default, new members are cascaded in the lower left corner of the page. Go to step 2.

- For a diagram that is stored on a local drive:

. With any object selected in the content table or the **Links** tab or window, pull down the **File** menu and choose the **Visio Live**→**Open** options.

A new live Visio session starts with the Open dialog window, which lists existing folders and files in the current drive or folder. If the list does not display the diagram, you can use the **Look in** field to change the drive or folder, or use the **Files of type** field to see other types of files.

. Select the diagram, and then click **Open**.

A message states that the diagram is being refreshed. When the message closes, the diagram is synchronized with the database.

If members of the owner were added, modified, or deleted in the database since the diagram was last saved, the diagram is updated with the changes. By default, new members are cascaded in the lower left corner of the page.



If you open a live Visio diagram while the Architect/Requirements client is not running, a message asks if you want to connect directly to the server. If you click **Yes**, you are prompted to log in, and then to connect to the server. If you click **No**, the diagram is not refreshed, and changes to the diagram are not applied in the database.

2. In the diagram, do any or all of the following:

- To add a shape and a new member of the diagram owner:
 - . In an Architect/Requirements stencil, point to an appropriate shape for the object type and hold down the left mouse button.
 - . Move the pointer to the Visio page, and then release the mouse button.

The shape appears with selection handles and a rotation handle. In the database, a corresponding object is created as a direct child or member of the diagram owner.

- To add a shape for an object that is not a member of the diagram owner:
 - In the Architect/Requirements client, select the object, and then pull down the **Edit** menu and choose **Copy**.

You can also right-click the object and choose **Copy** from the pop-up menu, click the **Copy** button on the toolbar, or press control-C.

- In the diagram, right-click a blank area on the page, and then choose **Paste From TcSE** from the pop-up menu.

The shape appears on the top of the stack at the lower left corner of the page. However, an object is not created in the database.



A given Architect/Requirements object can appear any number of times in a live Visio diagram.

- To copy an existing shape and create a new member of the diagram owner:

- Select the shape, pull down the Visio **Edit** menu, and choose **Copy**.

You can also right-click the object and choose **Copy** from the pop-up menu, click the **Copy** button on the Visio toolbar, or press control-C.

- Right-click a blank area on the Visio page, and then choose **Paste And Create In TcSE** from the pop-up menu.

The shape appears with selection handles and a rotation handle. In the database, the object is created as a member of the diagram owner.



A given Architect/Requirements object can be copied any number of times in a live Visio diagram.

- To create a new diagram for an object represented by a shape, right-click the shape and choose **Create New Diagram** from the pop-up menu.

The diagram opens in a new Visio window, with a shape for each member of the object. That object is the owner of the diagram, which you can rename in the **Attachments** tab or window.

- To change the properties of an object in the Architect/Requirements client:

On the Visio page, right-click the shape for the object, and then choose **TcSE Properties** from the pop-up menu.

Live Visio displays the Teamcenter for systems engineering properties dialog window. Values that you cannot change are dimmed in the **Value** column.

- In the **Value** column, double-click a value that you want to change.

Depending on the property type, a text field or a list field opens.

- Enter the new value in the text field or select the new value in the list field, and then click outside the field.

For each additional value that you want to change, repeat steps b and c.

- . To apply the changes and close the dialog window, click **OK**.
- To rename an object in the Architect/Requirements client:
 - . Select the shape for the object, and then do one of the following:
 - If the object name is mapped to the shape itself, select the shape and press the **F2** key to open a text field.
 - If the object name is mapped to the subshape that contains the name, double-click the shape to open a text field.



The **Name** property of the object in the database is mapped to the **Text** property of the shape through the live Visio stencil. For more information about customizing a live Visio stencil, see the *Systems Architect/Requirements Management Project Administrator's Manual*. If you have questions about stencils, consult your project administrator.

- . Enter the new name in the text field, and then click outside the shape.
- To create a trace link between objects in the Architect/Requirements client:
 - . Right-click the shape that represents the starting object, and then choose **Start Link** from the pop-up menu.
 - . Right-click the shape that represents the ending object, and then do one of the following:
 - For the default subtype, choose **End Link** from the pop-up menu.
 -

For a user-defined subtype:

 - . Choose **End Link Using Sub-type** from the pop-up menu to display the Select Subtype dialog window.
 - . Select a trace link subtype, and then click **OK** to close the dialog window.

Live Visio displays a message confirming that the link is created. When the starting or ending object is selected in the Architect/Requirements client, the trace link is displayed in the **Links** tab or window. For more information, see [Links Tab](#) in chapter 3, *Using the Architect/Requirements Main Window*.

- To delete a shape and the corresponding object, right-click the shape and choose **Delete in TcSE** from the pop-up menu.



This option does not support the selection of multiple objects.

The shape is removed from the diagram, and the object is moved to your Architect/Requirements Recycle Bin. You can restore the object until your Recycle Bin is emptied. For more information, see [Restoring Objects](#) or [Emptying Your Architect/Requirements Recycle Bin](#) in chapter 4, *Maintaining a Project*.

- To delete a shape without deleting the corresponding object, select the shape, and then click the **Delete** button on the Visio toolbar or press the delete key.



If the object is a member of the diagram owner, live Visio displays a message stating that the diagram type will be changed to **Static** and asks if you want to continue. Click **Yes** to remove the shape.

The shape is removed from the diagram and the object remains in the client.

If the object is a member of the owner, the diagram's **Diagram Content** property value is changed from **Members** to **Static**. For more information, see [Changing a Diagram Type](#), later in this chapter.



The **Delete** button removes the trace link from the diagram. If you want to remove the trace link from Architect/Requirements, you must select the trace link and click **RMB→Delete In TcSE**.

3. To save your changes, pull down the Visio **File** menu and choose **Save**.

Also, you can save the diagram outside the database by pulling down Visio's **File** menu and choosing **Save As** to display the Save As dialog window. Later, you can open the diagram and synchronize it with the database by pulling down the **File** menu and choosing the **Visio Live→Open** options.

Procedure Notes

Step 1: For a diagram displayed in the **Attachments** or **Where Used** tab or window, you can also right-click the diagram and choose **Open** from the pop-up menu, or double-click the diagram. For a diagram that is stored on a local drive, you can also right-click anywhere in the content table or the **Links** tab or window, and choose **Visio Live→Open** from the pop-up menu.

Copying a Microsoft Office Visio Diagram to Live Visio

Elements of a Microsoft Office Visio diagram can be copied to a live Visio diagram. This process allows you to reuse Visio content to create objects in the Architect/Requirements database.

The source diagram is a standard Visio file that is stored outside the database, for example, on a local or shared drive. You use a special live Visio function to copy the source to the destination, either a new live Visio diagram or one stored in or outside the database. For more information, see [Creating a Diagram](#), earlier in this chapter.



Shapes in the source diagram must be represented in the mapping file for the live Visio stencil. A mapping file assigns shapes in the source to object types and properties in the database. For more information about configuring a mapping file, see the *Systems Architect/Requirements Management Project Administrator's Manual*. If you have questions about stencils or mapping files, consult your project administrator.



- You must have **Architect** privilege for the project.
- Microsoft Office Visio 2013 or Visio 2016 must be installed.

To copy a Microsoft Office Visio diagram to live Visio:

1. In the source file, select all or any portion of the diagram, and then pull down the Visio **Edit** menu and choose **Copy**.



To import a connection, the starting and ending objects must be selected with the connection shape. Tree connections are not supported.

2. In the live Visio file, right-click the page and choose **Paste And Create In TcSE** from the pop-up menu.



Use only this live Visio option to paste the selection. The standard Visio paste function may give unpredictable and undesired results.



This option is supported only on the first page of the live Visio diagram.

A progress indicator is displayed while shapes are pasted and objects are created. A second progress indicator shows that the server is synchronizing changes.



Do not perform any action in the live Visio file while either progress indicator is displayed. Otherwise, the file may be corrupted.

Live Visio displays the copied shapes initially with Visio selection indicators. For each shape represented in the mapping file, an object is created in the database and displayed in the content table.

Objects are not created for shapes that are not represented in the mapping file. These shapes are highlighted in red, and you can change the color through the standard Visio functions. You can select all of these shapes by right-clicking the live Visio page and choosing **Select Non TcSE Shapes** from the pop-up menu.

Changing a Diagram Type

For every live Visio diagram, the **Diagram Content** property controls the behavior of the shapes that represent the members of the diagram owner in the Architect/Requirements client. This property is displayed in the Edit Properties dialog window and in the **Properties** floating window. You can edit this property for a diagram selected in the **Attachments** or **Where Used** tab or window. For more information, see [Properties Tab](#), [Attachments Tab](#), [Where Used Tab](#), or [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.



You must have **Architect** privilege for the project.

To change a diagram type:

1. Do one of the following:

- To use the Edit Properties dialog window, select the diagram, and then pull down the **File** menu and choose **Properties**. Or, right-click the diagram and choose **Properties** from the pop-up menu.
- To use the **Properties** floating window:
 - With the **Properties** tab on top in the notebook pane, click the **Open tab** button on the toolbar to open the **Properties** window.
 - Select the diagram in the **Attachments** or **Where Used** tab.

2.

In the **Value** column, double-click the **Diagram Content** property value to display the Single-Choice dialog window.

3. Check the check box for one of the following diagram types:

- **Members** fully synchronizes the set of shapes with the owner's member set. As members are added to the owner and moved to other owners in the client, corresponding shapes are added and removed in the diagram.
- **Static** prevents the automatic addition and removal of shapes as the owner's member set changes. When members are added to the owner and moved to other owners in the client, the set of shapes is unchanged. However, the diagram remains connected to the database.
 - Shapes for new members can be manually added to the diagram, thus creating new objects in the client.
 - When object properties are changed in the client, corresponding shapes reflect the changes automatically.

Any shape can be manually removed from a **Static** diagram, but the corresponding object is not deleted in the client.

For both diagram types, shapes are automatically deleted when corresponding objects are deleted in the client.



A **Members** diagram is changed to **Static** when a shape is added for a non-member object or when a member shape is removed. For more information, see [Editing a Diagram](#), earlier in this chapter.

4. Click **OK** to close the dialog window.

Disconnecting a Live Visio Diagram

Live Visio diagrams can reside outside the Architect/Requirements database, for example, on a local or shared drive. Any such diagram can be converted to a standard Visio file, one that is permanently disconnected from the database.



A disconnected diagram can no longer be used interactively with the Architect/Requirements client.



Microsoft Office Visio 2013 or Visio 2016 must be installed on your computer.

To disconnect a live Visio diagram:

1. In the diagram, right-click the Visio page and choose **Make Visio Diagram Non Live** from the pop-up menu.



If the diagram is not stored outside the database, pull down the Visio **File** menu and choose **Save As**, and then use the Save As dialog window to specify the drive or folder.

A confirmation message states that the diagram cannot be converted to a live diagram again.

2. To continue, click **Yes**.



This action is permanent and cannot be reversed.

The diagram is disconnected. The current objects and properties remain in the file.

Viewing a Diagram if Microsoft Office Visio Is Not Installed

If Microsoft Office Visio is not installed on your computer, you can open a live Visio diagram through the Microsoft Office Visio Viewer. An ActiveX control, the Visio Viewer displays Visio diagrams in Microsoft Internet Explorer.

You can view and print the diagram, but you cannot edit the diagram in the Visio Viewer. Furthermore, the diagram is not synchronized with the Architect/Requirements database.



- Microsoft Internet Explorer version 7 or 8 (32 bit) must be installed on your computer.
- Microsoft Office Visio Viewer must be installed with Internet Explorer. If necessary, you can download the latest version of the Visio Viewer from the Microsoft Download Center at <http://office.microsoft.com/en-us/officeupdate/default.aspx>.

To view a diagram if Microsoft Office Visio is not installed:

Open the diagram by doing any of the following:

- In the Architect/Requirements client, select the diagram, and then pull down the **File** menu and choose **Open**. You can also right-click the diagram and choose **Open** from the pop-up menu. Or, double-click the diagram.
- In Internet Explorer, pull down the **File** menu and choose **Open**. Then, select the diagram in the Open dialog window.



You can also use the mouse to drop the diagram in the Internet Explorer window.

- In Windows Explorer, double-click the diagram. Or, select the diagram, pull down the **File** menu, and choose **Open**.
- In an E-mail message with the diagram attached, double-click the attachment. Or, right-click the attachment and choose **Open** from the pop-up menu.

The Visio Viewer displays the diagram in Internet Explorer.

- To pan and zoom in the drawing window, you can use toolbar buttons, keyboard shortcuts, or options in the pop-up menu.
- To view the properties of a shape, you can open the Properties and Settings dialog window and then select the shape.

Also in the Properties and Settings dialog window, you can do the following:

- Make rendering and display settings in the **Display Settings** tab.
- Set drawing layer visibility and colors in the **Layer Settings** tab.
- Set annotation visibility and colors in the **Markup Settings** tab.

Data Dictionaries and Data Definitions

Creating a Data Dictionary

A data dictionary is a special folder subtype for containing data definitions. You can create a data dictionary directly below the project node or in a folder at any lower level. Then, you can attach the data dictionary to a diagram in the **Attachments** tab or window. For more information, see [Attaching a Data Dictionary to a Diagram](#), later in this chapter.

To create a data dictionary:

1. In the navigation tree or the hierarchical content table, select the containing folder for the data dictionary.
2.
Pull down the **File** menu and choose the **New→Subtype** options.
The Select Subtype dialog window is displayed.
3. Under **Folder**, select the **Data Dictionary** subtype, and then click **OK**.
The content table displays the new data dictionary with a default name in an open text field.
4. Enter a meaningful name for the data dictionary, and then press the enter key.

Procedure Notes

Step 1: You can also right-click the containing folder and select **New→Subtype** from the pop-up menu. Or, click the **Create a new subtype** button on the toolbar or press control-U.

Step 3: To reverse this action, you can pull down the **Edit** menu and choose **Undo Create Subtype**, click the **Undo** button on the toolbar, or press control-Z.

Attaching a Data Dictionary to a Diagram

To assign data definitions to connections in a diagram, you must first attach a data dictionary to the diagram. If a diagram already has a data dictionary, you can attach a different one and assign those data definitions.

To attach a data dictionary to a diagram:

1. In the **Attachments** tab or window, select the diagram, and then pull down the **File** menu and choose **Properties**.
The Edit Properties dialog window is displayed.
2. In the **Value** column, double-click the **Data Dictionary** value.
The Single-Choice dialog window is displayed.
3. Check the check box for the data dictionary, and then click **OK**.
4. Click **Close** to close the Edit Properties dialog window.

Creating Data Definitions

A data definition is a building block subtype that describes a connection between other building blocks. When assigned to a connection in a live Visio diagram, the data definition's user-defined properties provide information about the connection.

Data definitions reside in data dictionaries and can be arranged in hierarchies, with multiple levels of parents, children, and siblings. You can create these hierarchies in the Architect/Requirements client and in a live Visio diagram.

In the Architect/Requirements client, you can create shortcuts to data definitions. These shortcuts allow you to reuse data definitions in a data dictionary. For more information, see [Working With Shortcuts](#) in chapter 4, *Maintaining a Project*.



- You must have **Architect** privilege for the project.
- You must have **Modify** permission for the data dictionary in which you intend to create the data definition.

Creating a Data Definition in the Architect/Requirements Client

As with a building block in a folder, you can create a data definition at any level in a data dictionary hierarchy.

1. To set the data definition's level in the hierarchy, do one of the following:
 - For the top level of a data dictionary, select the data dictionary in the navigation tree or the content table.
 - For the same level as another data definition, select the sibling in the content table.
 - For a child of another data definition, select the parent in the content table.
2. Pull down the **File** menu and choose the **New**→**Subtype** options.
The Select Subtype dialog window is displayed.
3. Under **Building Block**, select the **Data Definition** subtype, and then click **OK**.
The content table displays the new data definition with a default name in an open text field.
4. Enter a meaningful name for the data definition, and then press the enter key.

Creating a Data Definition in a Live Visio Diagram

While you build a diagram, you can create data definitions and attach them to connections. Through the Data Definitions dialog window in live Visio, you can create a data definition at any level of a data dictionary hierarchy. Each new data definition is automatically added to the Architect/Requirements client and database.

1. Right-click a connection and choose **Set Data Definition** from the pop-up menu.

The Data Definitions dialog window displays the current hierarchy for the data dictionary that is attached to the diagram. If the data dictionary contains data definitions, the tree is expanded with the data dictionary at the peak. If it does not contain data definitions, only the data dictionary is displayed.



If other users add or delete data definitions while the dialog window is open, the changes are not reflected in the tree. You can click **Refresh** to ensure that the tree shows the current hierarchy in the data dictionary.

Or, you can check the **Auto Refresh** check box to update the tree each time you create a data definition. However, each update involves a server call, which may affect performance in some cases, for example, a large number of data definitions in a data dictionary or a database on a remote server.

2. In the tree, select the node below which you want to create the data definition.

To navigate to a data definition in the client, you can right-click the data definition in the tree and choose **Go To TcSE** from the pop-up menu.



The tree may also display shortcuts to data definitions. You cannot create a data definition below a shortcut.

3. In the **Name** field, enter a meaningful name for the data definition.

In the **Subtype** field, you can select the subtype for the data definition.

4. Click **Add**.

The tree displays the new data definition at the specified level. In the client and in the database, the data definition is added to the data dictionary hierarchy.



The Data Definitions dialog window remains open. You can repeat steps 2 through 4 to create another data definition.

5. Do one of the following:

- To attach a data definition to the selected connection:

- . In the tree, select the data definition.
- . Click **Attach**.

The Data Definitions dialog window closes, and the data definition name appears on the connection.

- To close the Data Definitions dialog window without attaching the data definition to a connection, click **Exit**.

Ports and Connections

A port is a connection point through which data is communicated across a boundary of a physical system. Connections show the flow of data between two ports.

Creating Ports

In the Architect/Requirements client, you can create a port on any building block or its subtype, including a data definition. In a live Visio diagram, you can create a port on any shape that is mapped to a building block.



- You must have **Architect** privilege for the project.
- You must have **Modify** permission for the building block.

Create a port in the Architect/Requirements client

1. In the content table, select the building block on which you want to place the port.
2. Assign the object type by doing one of the following:
 - For the base type, **Port**, pull down the **File** menu and choose the **New→Port** options.
 - For a user-defined subtype:
 - . Pull down the **File** menu and choose the **New→Subtype** options to display the Select Subtype dialog window.
 - . Under the **Port** type, select the subtype and click **OK**.

If you have questions about subtypes, consult your Architect/Requirements project administrator.

The **Connectivity** tab or window displays the port with a default name in an open text field.

3. Enter a meaningful name, and then press the enter key.

The port is created also in diagrams where the building block is represented.

Procedure Notes

Step 2: You can also right-click the building block and choose the options from the pop-up menu.

Create a port in a live Visio diagram

1. Point to a port shape in the stencil pane.

The shape can represent the base type, **Port**, or a user-defined port subtype. If you have questions about port shapes or subtypes, consult your Architect/Requirements project administrator.

2. Drop the shape on a connection point on the building block shape.

The port is created also in the Architect/Requirements client.

Creating Connections in the Architect/Requirements Client



You must have **Architect** privilege for the project.

1. Select the starting object or objects, and then pull down the **Edit** menu and choose the **Connections**→**Start Connection** options.



- If you select only one starting object, you can select multiple ending objects. If you select multiple starting objects, you can select only one ending object.
- To connect objects in different projects, and if the target project is in a different Architect/Requirements client running in a separate window, you must use the mouse to create the connections. Hold down the left mouse button on the starting selection, move the pointer to the target project in its client window, and then release the mouse button to complete this procedure.

You can also right-click the selection and choose the **Connections**→**Start Connection** options from the pop-up menu. Or, click the **Start a connection** button on the toolbar.

2. Select the ending object or objects.

If only one starting object is selected, you can select multiple ending objects. If multiple starting objects are selected, you can select only one ending object.



If you are connecting objects in different projects, navigate to the target project and select the ending object or objects. For connections in the same project, you can open a different folder and select the ending object or objects.

3. Assign the subtype by doing one of the following:

- To assign the default subtype (**Connection**), pull down the **Edit** menu and choose the **Connections**→**End Connection** options.

You can also right-click the selection and choose the **Connections**→**End Connection** options from the pop-up menu. Or, click the **End a connection** button on the toolbar.

- To assign a user-defined subtype:
 - Pull down the **File** menu and choose the **Connections**→**End Connection Subtype** options.



You can also right-click the selection and choose the **Connections**→**End Connection Subtype** options from the pop-up menu. Or, click the **End a connection subtype** button on the toolbar.

The Select Subtype dialog window displays the connection subtypes for the project.

- Select a subtype, and then click **OK** to close the dialog window.

For starting or ending objects that are ports, the connections are displayed in the **Connectivity** tab or floating window.

For ending objects other than ports, the **Start** pane in the **Relations** subtab displays starting objects in the left table and starting connections in the right table. If you linked to multiple starting objects, select one at a time to see this information.

For starting objects other than ports, the **End** pane in the **Relations** subtab displays ending objects in the left table and ending connections in the right table.

Procedure Notes

Step 3: You can reverse this action by pulling down the **Edit** menu and choosing **Undo End Connection**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.



The copy, cut, and paste operations are disabled for port and connection objects in the Architect/Requirements client. The corresponding commands in the **Connectivity** tab and the **Edit** menu are disabled.

However, you can copy a connection to a live Visio diagram. For more information, see [Copying a Connection to a Live Visio Diagram](#) later in this chapter.

Working With Connections in a Live Visio Diagram

In a new or existing live Visio diagram, you can create connections between shapes that represent requirements, building blocks, and other elements of the system design. For each connection shape that you create in a diagram, a corresponding connection object is created in the Architect/Requirements database.

In the Architect/Requirements client, connections between ports are displayed only in the **Connectivity** tab or floating window. For objects other than ports, connections can be viewed in the **Relations** subtab of the **Links** tab and window. In the **Where Used** window, you can see the diagram that references the connection object selected in the **Relations** tab or window. You can also search for specific connection objects in the Search module. For more information, see [Viewing the Connections for an Object](#), later in this chapter; [Relations Subtab](#) and [Where Used Tab](#) in chapter 3, *Using the Architect/Requirements Main Window*; and chapter 11, [Using the Search Module](#).



To navigate to an object in the client, you can right-click the corresponding shape in the diagram and choose **Go To TcSE** from the pop-up menu.

The connection shapes are from the Visio stencil selected when the diagram was created. The stencil consists of a **.vss** file, which determines the connection shapes themselves, and an **.xml** file that determines the information that is mapped to the shapes. Stencils for the project are maintained in the **Reports and Formatting** folder in the Administration module. If you have questions about stencils, consult your project administrator.



You must have **Architect** privilege for the project.

Creating a Connection in a Live Visio Diagram

1. Point to the connection shape in the stencil pane, hold down the left mouse button, and drop the shape on a connection point on the starting object.
2. Point to the end point (+) on the connection shape, hold down the left mouse button, and drop the end point on a connection point on the ending object.

Depending on where you place the connection shape in step 1, you may need to glue the begin point (x) to a starting connection point to create the connection object in the database.



- Multiple connections can be created between the same two objects. A name is automatically assigned to each connection.
- The begin point and the end point can be glued to the same object.
- You can also create a connection by copying and pasting an existing connection shape in the diagram, and then gluing it to the starting and ending objects.
- If the ending object is a building block, a port is automatically created on that object, and the connection end point is glued to the port.

Editing a Connection in a Live Visio Diagram

To reverse the direction between the same two objects, right-click the connection shape and choose **Reverse Direction** from the pop-up menu.

To change the starting or ending object, detach the begin point or the end point of the connection shape from the current connection point, and then glue the begin or end point to a connection point on another object.



If you drop the begin point or the end point anywhere other than on a connection point, live Visio displays a message stating that you are creating a dangling connection. If you click **Yes** to continue, the connection shape is removed from the diagram. The connection object remains in the Architect/Requirements client.

Copying a Connection to a Live Visio Diagram

You can copy a connection from the Architect/Requirements client to a live Visio diagram, and also from one diagram to another.

To copy a connection from the Architect/Requirements client to a diagram:

1.

In the **Relations** subtab, right-click the connection object and choose **Copy to Diagram** from the pop-up menu.

The connection object identifier is placed in the Windows clipboard.

2.

In the diagram, right-click the Visio page and choose **Paste From TcSE** from the pop-up menu.

The connection shape is added to the diagram and is attached to the correct points automatically.

To copy a connection from one diagram to another diagram:



The source diagram and the destination diagram need not be attached to the same owner. To get a connection, it is sufficient if two ends of the connection are present in the diagram.

1.

In the source diagram, right-click the connection shape and choose **Copy Connection** from the pop-up menu.

The connection object identifier is placed in the Windows clipboard.

2.

In the destination diagram, right-click the Visio page and choose **Get connection from Visio diagram** from the pop-up menu.

The connection shape is added to the diagram and is attached to the correct points automatically.

Deleting a Connection

You can delete connections from a live Visio diagram only, and also from both the diagram and the Architect/Requirements client at the same time.

To delete a connection from the diagram only:

Do one of the following:

- Select the connection shape, and then press the delete key or click the **Delete** button on the Visio toolbar.

The connection shape is removed from the diagram. The connection object remains in the Architect/Requirements client.

- Detach the begin point or the end point of the connection shape from the current connection point, and then drop the begin point or the end point anywhere other than on a connection point.

Live Visio displays a message stating that you are creating a dangling connection and asking if you want to continue. Click **Yes** to remove the connection shape. The connection object remains in the Architect/Requirements client.

To delete a connection from the diagram and the Architect/Requirements client:

Do one of the following:

- In the live Visio diagram, right-click the connection shape and choose **Delete in TcSE** from the pop-up menu.
- In the **Relations** subtab, select the connection object, and then pull down the **Edit** menu and choose the **Links**→**Delete Link** options. Or, right-click the connection and choose the **Links**→**Delete Link** or **Delete**→**Delete Link** options from the pop-up menu.

The connection shape is removed from the diagram and the connection object is moved to your Architect/Requirements Recycle Bin. The object remains in the database, and you can restore the connection until you empty your Recycle Bin. For more information, see [Restoring Objects](#) and [Emptying Your Architect/Requirements Recycle Bin](#) in chapter 4, *Maintaining a Project*.

Hiding and Showing Connection Names in a Live Visio Diagram

You can toggle connection names to make them visible or not visible.

- To hide a specific connection name, right-click the connection shape and choose **Hide/Unhide Connection Name** from the pop-up menu.



The first time a connection is hidden, it becomes a marked connection name. Marked names and unmarked names are toggled separately.

- To show a specific connection name, right-click the connection shape and choose **Hide/Unhide Connection Name** from the pop-up menu.
- To hide or show all unmarked connection names, right-click the Visio page and choose **Hide/Unhide Connection Names** from the pop-up menu.



Marked connection names remain in their current states.

- To hide or show all marked connection names, right-click the Visio page and choose **Hide/Unhide Marked Connection Names** from the pop-up menu.



Unmarked connection names remain in their current states.

- To unmark all marked connection names, right-click the Visio page and choose **Reset Marked Connection Names** from the pop-up menu.

The names are returned to the unmarked state and are shown on the connection shapes.

Attaching a Data Definition to a Connection

In a live Visio diagram, multiple connections can be glued to one building block or to one port on a building block. One data definition can be attached to each connection.

1. In the diagram, right-click the connection and choose **Set Data Definition** from the pop-up menu.

The Data Definitions dialog window displays the current hierarchy for the data dictionary that is attached to the diagram.



If other users add or delete data definitions while the dialog window is open, the changes are not reflected in the tree. You can click **Refresh** to ensure that the tree shows the current hierarchy in the data dictionary.

2. Do one of the following:

- To attach an existing data definition:

- . In the tree, select the data definition.

You can use the **Search** section to find the data definition in the tree:

- o

Enter the name in the **Search** field and click the **Search** button to highlight the data definition.

- o

Click **First** to move the selection to the top data definition.

- o

Click **Previous** to move the selection up by one data definition.

- o

Click **Next** to move the selection down by one data definition.

- o

Click **Last** to move the selection to the bottom data definition.

To navigate to a data definition in the client, you can right-click the data definition in the tree and choose **Go To TcSE** from the pop-up menu.

If the tree displays shortcuts to data definitions, you can select a shortcut and attach it to the connection.

- . Click **Attach**.

The Data Definitions dialog window closes, and the data definition name appears on the connection.

- To create and attach a new data definition:

- . Select the node below which you want to create the data definition.

- . Enter the name in the **Name** field.

You can also select a subtype in the **Subtype** field.

- . Click **Add**.

The tree displays the new data definition at the specified level.

- Click **Attach**.

The Data Definitions dialog window closes, and the data definition name appears on the connection.

Detaching a Data Definition From a Connection



You must have **Architect** privilege for the project.

In the **Connectivity** tab or window:

1. Select the connection to which the data definition is attached.
2. Pull down the **Edit** menu and choose the **Data Definition**→**Clear Data Definition Link** options.

You can also right-click the connection and choose **Data Definition**→**Clear Data Definition Link** from the pop-up menu.

In a live Visio diagram:

1. Right-click the connection to which the data definition is attached.
2. Choose **Un-Assign Data Definitiion** from the pop-up menu.

Viewing the Connections for an Object

For the object selected in the hierarchical content table, connections are displayed in the **Relations** subtab of the **Links** tab and window. The subtab shows connections to the object, connections from the object, and the starting or ending object for each connection.

In the **Links** window, the **Relations** subtab can show such information for the object selected in the **Where Used** or **Versions** tab. For more information, see [Hierarchical Content Table](#), [Relations Subtab](#), [Where Used Tab](#), [Versions Tab](#), or [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.



Connections between ports are displayed only in the **Connectivity** tab or floating window.

To view the connections for an object:

Do one of the following:

- For an object in the hierarchical content table, select the object and click the **Links** tab, and then click the **Relations** subtab.

You can open the **Links** window for this object by clicking the **Open tab** button on the notebook pane's toolbar.

- For an object in the **Where Used** or **Versions** tab:
 - With the **Links** tab on top in the notebook pane, click the **Open tab** button on the toolbar to open the **Links** window, and then click the **Relations** subtab.
 - Select the object in the **Where Used** or **Versions** tab.

In the **Start** pane, beginning connections to the selected object are displayed in the link table, at the right of the pane. At the left, the object table displays the object from which each beginning connection originates.

In the **End** pane, ending connections from the selected object are displayed in the link table, at the right of the pane. At the left, the object table displays the object at which each ending connection is completed.



In each pane, the rows in the object table and the link table remain synchronized at all times. For example, when you sort or scroll a link table, the corresponding object table is automatically sorted or scrolled to match the new view in the link table.

Procedure Notes

To see lower-level objects for selection in the content table, click the plus signs. Or, select an object with a plus sign, pull down the **View** menu, and choose **Expand All**. You can also right-click the object and choose **Expand All** from the pop-up menu.

To display the **Links** tab, you can also click the object's links indicator.

Working With Trace Links in a Live Visio Diagram

In a new or existing live Visio diagram, you can create trace links between shapes that represent requirements, building blocks, and other elements of the system design. For each trace link shape that you create in a diagram, a corresponding trace link object is created in the Architect/Requirements database.

In the Architect/Requirements client, trace links between objects can be viewed in the **Relations** subtab of the **Links** tab in the notebook pane. For more information, see [Overview of Trace Links](#) and [Creating Trace Links](#) in chapter 7, *Showing Object Relationships With Trace Links*.



You must have the **Architect** privilege for the project.

Creating a Trace Link in a Live Visio Diagram

1. Drag the trace link shape in the live Visio stencil pane to a connection point on the shape that represents the starting object.
2. Glue the trace link shape's end point to a connection point on the shape that represents the ending object.



The trace link is created in the database, but a trace link shape is not added to the diagram.

Copying a Trace Link to a Live Visio Diagram

1.
In the **Relations** subtab, right-click the trace link object and choose **Copy to Diagram** from the pop-up menu.
The trace link object identifier is placed in the Windows clipboard.
2.
In the diagram, right-click the Visio page and choose **Paste From TcSE** from the pop-up menu.
The connection shape is added to the diagram and is attached to the correct points automatically.

Chapter 7: Showing Object Relationships With Trace Links

This chapter discusses the role of trace links in joining objects in Architect/Requirements to one another within a single project, between two projects, and to objects in other Teamcenter products. Instructions are provided for creating, viewing, and deleting trace links, and for navigating to linked objects.

Overview of Trace Links

A trace link creates a directional relationship between two objects, a relationship conveyed by the terms *defining* and *complying*. A defining object specifies a condition that a product or a component must fulfill. A complying object partially or completely fulfills a condition specified by a defining object. Such a relationship establishes a path in which one object precedes the other.

For example, functional requirement *A*, for target tracking, defines hardware requirement *B*, for a CPU with a certain instruction rate. In this example:

- Requirement *A* precedes requirement *B*, with requirement *B* directly *downstream* in the complying path.
- Requirement *B* succeeds requirement *A*, with requirement *A* directly *upstream* in the defining path.

An object can assume both defining and complying relationships, continuing the path upstream and downstream. For example, a weight requirement may define a requirement to use aluminum. That complying requirement may in turn define temperature or environmental requirements consistent with the properties of aluminum. An object can have any number of defining and complying trace links.

Trace links can be created between objects within a single Architect/Requirements project, and also between objects in two different projects. In addition, trace links can be created from objects in Architect/Requirements to objects in other Teamcenter products, which are always complying objects. This two-way relationship allows for change analysis in both directions.

If the customer is considering a change to a product requirement, its trace links can be followed downstream, to all complying requirements that were derived from the product requirement, and eventually to all design elements that must comply. Conversely, an engineer working on a design element that must meet detailed, low level requirements can follow those trace links upstream, to the original customer requirements that define the design constraints.

Creating Trace Links

Trace links can be created from folders, requirements, building blocks, and groups to other objects of those types within Architect/Requirements. Also, objects in Architect/Requirements can be linked to objects in other Teamcenter products.

Linking Objects Within Architect/Requirements

You can link one defining object to one or more complying objects at the same time. Also by one action, you can link one or more defining objects to one complying object. However, multiple defining and complying objects cannot be linked simultaneously, and trace links cannot be linked to other trace links.

The defining and complying objects can reside together in the same Architect/Requirements project. Or, the complying objects can reside in a different project, in the same Architect/Requirements installation or in another installation. Defining and complying objects can be selected in the hierarchical content table and the **Links** and **Versions** tabs and floating windows. For more information, see [Hierarchical Content Table](#), [Links Tab](#), [Versions Tab](#), and [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.

In a given project, trace links within the project and interproject trace links are displayed together in the **Links** tab and floating window, the **Complying Object Traceability** view, and the Search Results dialog window. You can delete interproject trace links and navigate to linked objects in the same way as you do for those within a project. For more information, see [Viewing Object Relationships](#), [Deleting Trace Links for an Object](#), and [Navigating to a Linked Object](#), later in this chapter.

Each trace link receives certain system-defined properties, including the **Subtype** property. You can assign the default subtype, **Trace Link**, or a user-defined subtype created by your project administrator. Each trace link may also receive user-defined properties. The subtype and all other editable properties can be changed after the trace link's creation. For more information, see chapter 9, [Working With Object Properties](#) and appendix [System-Defined Properties in the Systems Engineering and Requirements Management Module](#).

You can create two or more trace links between the same objects if you assign a different subtype to each trace link. For example, defining requirement *A* is linked to complying requirement *B*, with the default subtype (**Trace Link**) assigned to the trace link. A second defining trace link, with user-defined subtype *SI*, can be created from requirement *A* to complying requirement *B*.

You cannot create and delete a trace link between two objects if they belong to any common baseline.

Subtypes also allow you to create trace links in both directions between the same objects. For example, defining requirement *A* is linked to complying requirement *B*, with subtype *SI* assigned to the trace link. By creating a trace link with user-defined subtype *S2*, you can link requirement *B*, as the defining object, back to requirement *A* as the complying object.



Only one trace link of a given subtype is allowed between the same two objects, regardless of the direction. However, you can create as many trace links between the objects as the number of available trace link subtypes.

Trace links can also be created in live Excel. For more information, see [Creating Trace Links in Live Excel](#) in chapter 9, *Working With Object Properties*.

To link objects in Architect/Requirements:

1. Select the defining object or objects, and then pull down the **Edit** menu and choose the **Links**→**Start Trace Link** options.



- If you select only one defining object, you can select multiple complying objects. If you select multiple defining objects, you can select only one complying object.
- If you are linking objects in different projects, and if the target project is in a different Architect/Requirements client running in a separate window, you must use the mouse to create the trace links. Hold down the left mouse button on the starting selection, move the pointer to the target project in its client window, and then release the mouse button to complete this procedure.

2. Select the complying object or objects.

If only one defining object is selected, you can select multiple complying objects. If multiple defining objects are selected, you can select only one complying object.



If you are linking objects in different projects, navigate to the target project and select the complying object or objects. For trace links within the same project, you can open a different folder and select the complying object or objects.

3. Assign the subtype by doing one of the following:

- To assign the default subtype (**Trace Link**), pull down the **Edit** menu and choose the **Links**→**End Trace Link** options, or click the **End Trace Link** button on the toolbar.



To assign a user-defined subtype:

- Display the Select Subtype dialog window by pulling down the **File** menu and choosing the **New**→**Subtype** options, by pressing control-U, or by clicking the **End Trace Link Subtype** button on the toolbar.

If you choose the **New**→**Subtype** options or press control-U, click the plus sign (+) to the left of **Trace Link** to see the user-defined subtypes. If you click the **End Trace Link Subtype** button, only trace link subtypes are shown, and user-defined subtypes are expanded automatically.

- Select a subtype, and then click **OK** to close the dialog window.

For each complying object, the objects in the defining path are shown in the **Defining Trace** column of the **Trace** subtab in the **Links** tab or window. In the **Relations** subtab, the **Start** pane shows defining objects in the left table and defining trace links in the right table. If you linked to multiple complying objects, select one at a time to see this information.

For a defining object, the objects in the complying path can be viewed in the **Complying Trace** column of the **Trace** subtab in the **Links** tab or window. In the **Relations** subtab, the **End** pane shows complying objects in the left table and complying trace links in the right table.

Procedure Notes

Step 1: You can also right-click the selection and choose the **Links**→**Start Trace Link** options from the pop-up menu. Or, click the **Start Trace Link** button on the toolbar.

Step 3: You can assign the default subtype also by right-clicking the selection and choosing the **Links**→**End Trace Link** options from the pop-up menu. To display the Select Subtype dialog window, you can also right-click the selection and choose **New**→**Subtype** from the pop-up menu.

Step 3: You can reverse this action by pulling down the **Edit** menu and choosing **Undo End Trace Link**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.



When a link is created, its name in the database is blank by default. On retrieving a link's name for display, if it is blank, a name is constructed from the names of the linked objects. In such case, changes to the names of the linked objects are reflected in the links name. You can also directly set a link's name. The links name is fixed after setting the links name and it does not use the linked object names. This behavior applies to all the user-visible link types, including trace links, connections, reference links, and generic links.

Linking to an Object in Another Teamcenter Product

You can create a trace link from a folder, a requirement, a building block, or a group in Architect/Requirements to an object in any of the following Teamcenter products:

- Engineering Process Management
- Teamcenter Enterprise
- Project

The object in Architect/Requirements is the defining object, and the trace link is bidirectional. In the other product's database, a complying trace link is created from the target object to the defining object.

You can select the defining object in the hierarchical content table or the **Links** tab or floating window. For more information, see [Hierarchical Content Table](#), [Links Tab](#), and [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.

To link to an object in another Teamcenter product:

1. Select the defining object, and then pull down the **Edit** menu and choose one of the following options:

-

Links→**Link To Application**

Architect/Requirements displays the **Select Object Chooser** dialog window. Select a product from the list, and then click **OK**.

- **Links**→**Link To TcEngineering**
- **Links**→**Link To TcEnterprise**
- **Links**→**Link To TcProject**

The product's object chooser is displayed. If necessary, enter your user name and password for the product.

2. In the chooser, navigate to and select the target object, and then click **Done**.



This action clears the queue for the **Undo** option and cannot be reversed.

A confirmation message is displayed, and then the chooser closes. The **Links** tab or window displays the object in the **Complying Trace** column.

Procedure Notes

Step 1: You can also right-click the defining object and choose the options from the pop-up menu.

Viewing Object Relationships

A given object may have both defining and complying relationships with other objects. Defining objects reside in an Architect/Requirements project. Complying objects may reside in a project and in other Teamcenter products. You can view the objects in a defining or complying path, from the next in the path to all objects upstream or downstream. You can also view the trace links themselves, as independent objects.

Viewing Objects in a Defining or Complying Path

For the object selected in the hierarchical content table, defining and complying objects are displayed in the **Trace** subtab of the **Links** tab and window. Each pane shows linked objects in the order of their precedence in the defining or complying path. In the **Links** window, the subtab can show such paths for the object selected in the **Where Used** or **Versions** tab. For more information, see [Hierarchical Content Table](#), [Trace Subtab](#), [Where Used Tab](#), [Versions Tab](#), or [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.

In addition, the **Defining Object Traceability** window displays defining objects, and the **Complying Object Traceability** window displays complying objects, for the object selected in the hierarchical content table. For a selected folder, each window shows all objects in the folder, whether or not the folder defines those objects. The objects are shown in the order of their precedence in the defining or complying path.

To view defining and complying objects in the Links tab or window:

Do one of the following:

- For an object in the hierarchical content table, select the object and click the **Links** tab, and then click the **Trace** subtab. You can open the **Links** window for this object by clicking the **Open tab** button on the notebook pane's toolbar.
- For an object in the **Where Used** or **Versions** tab:
 - . With the **Links** tab on top in the notebook pane, click the **Open tab** button on the toolbar to open the **Links** window, and then click the **Trace** subtab.
 - . Select the object in the **Where Used** or **Versions** tab.

The **Defining Trace** column displays the objects that define the selected object, its direct predecessors in the defining path. A plus sign (+) is shown for each defining object that complies with other objects, continuing the path upstream.

The **Complying Trace** column displays the objects that comply with the selected object, its direct successors in the complying path. A plus sign (+) is shown for each complying object that defines other objects, continuing the path downstream.

You can view the objects in a continuing path by doing the following:

- To view the direct predecessors or successors, click the plus sign for the object.
- To view all predecessors or successors simultaneously, right-click the object and choose **Expand All** from the pop-up menu. In the **Links** tab, you can also select the object, pull down the **View** menu, and choose **Expand All**.

To view defining objects in the Defining Object Traceability window:

Select the object in the hierarchical content table, and then pull down the **Tools** menu and choose the **Traceability**→**Defining** options.



You can also right-click the object and choose the **Traceability**→**Defining** options from the pop-up menu.

The **Defining Object Traceability** window displays the objects that define the selected object, its direct predecessors in the defining path. In the **Name** column, a plus sign (+) is shown for each defining object that complies with other objects, continuing the path upstream.

You can view the objects in a continuing path by doing the following:

- To view the direct predecessors, click the plus sign for the object.
- To view all predecessors simultaneously, right-click the object and choose **Expand All** from the pop-up menu.

The property columns provide additional information. You can add and remove columns, rearrange and resize columns, and sort by any column. For more information, see [Adding and Removing Columns](#), [Rearranging Columns](#), [Resizing Columns](#), and [Setting the Sort Column](#), in chapter 9, *Working With Object Properties*.



For a value that exceeds the cell size, you can rest the pointer on the cell to see the entire value in a tooltip. Multiple lines are used if necessary.

You can move, resize, minimize, and maximize the window through the standard Microsoft Windows functions. By right-clicking objects to display a pop-up menu, you can also do the following:

- Export objects to Microsoft Excel or Microsoft Word. For more information, see [Exporting Objects to Microsoft Office Excel](#) and [Exporting Objects to Microsoft Office Word](#) in chapter 4, *Maintaining a Project*.
- Open linked requirements in Microsoft Word to edit or view their content. For more information, see [Entering and Changing Requirement Content in Microsoft Office Word](#) and [Viewing Requirement Content](#) in chapter 5, *Managing Requirements*.
- Navigate to linked objects. For more information, see [Navigating to a Linked Object](#), later in this chapter.
- Open a live Excel file from a previous session. For more information, see [Using the Live Excel Interface](#) in chapter 9, *Working With Object Properties*.

To view complying objects in the Complying Object Traceability window:

Select the object in the hierarchical content table, and then pull down the **Tools** menu and choose the **Traceability**→**Complying** options.



You can also right-click the object and choose the **Traceability**→**Complying** options from the pop-up menu.

The **Complying Object Traceability** window displays the objects that comply with the selected object, its direct successors in the complying path. In the **Name** column, a plus sign (+) is shown for each complying object that defines other objects, continuing the path downstream.

You can view the objects in a continuing path by doing the following:

- To view the direct successors, click the plus sign for the object.
- To view all successors simultaneously, right-click the object and choose **Expand All** from the pop-up menu.

The property columns provide additional information. You can add and remove columns, rearrange and resize columns, and sort by any column. For more information, see [Adding and Removing Columns](#), [Rearranging Columns](#), [Resizing Columns](#), and [Setting the Sort Column](#) in chapter 9, *Working With Object Properties*.



For a value that exceeds the cell size, you can rest the pointer on the cell to see the entire value in a tooltip. Multiple lines are used if necessary.

You can move, resize, minimize, and maximize the window through the standard Microsoft Windows functions. By right-clicking objects to display a pop-up menu, you can also do the following:

- Export objects to Microsoft Excel or Microsoft Word. For more information, see [Exporting Objects to Microsoft Office Excel](#) and [Exporting Objects to Microsoft Office Word](#) in chapter 4, *Maintaining a Project*.
- Open linked requirements in Microsoft Word to edit or view their content. For more information, see [Entering and Changing Requirement Content in Microsoft Office Word](#) and [Viewing Requirement Content](#) in chapter 5, *Managing Requirements*.
- Navigate to linked objects. For more information, see [Navigating to a Linked Object](#), later in this chapter.
- Open a live Excel file from a previous session. For more information, see [Using the Live Excel Interface](#) in chapter 9, *Working With Object Properties*.

Viewing Trace Links for an Object

Trace links are displayed in the **Relations** subtab of the **Links** tab and window. For the object selected in the hierarchical content table, the subtab shows defining trace links, complying trace links, and the starting or ending object for each trace link.

In the **Links** window, the **Relations** subtab can show such information for the object selected in the **Where Used** or **Versions** tab. For more information, see [Hierarchical Content Table](#), [Relations Subtab](#), [Where Used Tab](#), [Versions Tab](#), or [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.

To view trace links for an object:

Do one of the following:

- For an object in the hierarchical content table, select the object and click the **Links** tab, and then click the **Relations** subtab.

You can open the **Links** window for this object by clicking the **Open tab** button on the notebook pane's toolbar.
- For an object in the **Where Used** or **Versions** tab:
 - With the **Links** tab on top in the notebook pane, click the **Open tab** button on the toolbar to open the **Links** window, and then click the **Relations** subtab.
 - Select the object in the **Where Used** or **Versions** tab.

In the **Start** pane, defining trace links to the selected object are displayed in the link table, at the right of the pane. At the left, the object table displays the object from which each defining trace link originates.

In the **End** pane, complying trace links from the selected object are displayed in the link table, at the right of the pane. At the left, the object table displays the object at which each complying trace link is completed.



In each pane, the rows in the object table and the link table remain synchronized at all times. For example, when you sort or scroll a link table, the corresponding object table is automatically sorted or scrolled to match the new view in the link table.

Navigating to a Linked Object

From any of the following trace link views, you can navigate to a linked object in its native location:

- The **Links** tab and floating window display defining objects, complying objects, and trace links for the object selected in the hierarchical content table. The **Links** window displays linked objects and trace links for the object selected in the **Where Used** or **Versions** tab.
- The **Defining Object Traceability** window displays defining objects, and the **Complying Object Traceability** window displays complying objects, for the object selected in the hierarchical content table. For a selected folder, each window also displays all objects in the folder, whether or not the folder complies with or defines those objects.

For more information, see [Viewing Object Relationships](#), earlier in this chapter; and [Hierarchical Content Table](#), [Links Tab](#), [Where Used Tab](#), [Versions Tab](#), or [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.

The linked object's native location may be within the same Architect/Requirements database. Or, the linked object may reside in one of the following Teamcenter products:

- Engineering Process Management (version 8.1.1 or later)
- Teamcenter Enterprise (version 3.1 or later)
- Project (version 3.2 or later)

To navigate to a linked object:

Do one of the following:

- In the **Links** tab or window, select the object, and then pull down the **View** menu and choose the **Go To**→**Go To Object** options. Or, right-click the object and choose the **Go To Object** option from the pop-up menu.
- In the **Defining Object Traceability** window or the **Complying Object Traceability** window, right-click the object and choose the **Go To Object** option from the pop-up menu.

If the object is within the same database, the hierarchical content table displays and highlights that object.

If the object is in another Teamcenter product, that product displays and highlights the object. If necessary, enter your user name and password for the product.



The **Go To Object** option is also available on the **View** menu and a pop-up menu in the Search Results dialog window. For more information, see [Basic Search View](#) or [Intermediate Search View](#) in chapter 11, *Using the Search Module*.

Deleting Trace Links for an Object

You can delete trace links for the object selected in the hierarchical content table or the **Where Used** or **Versions** tab or window. The trace links are displayed in the **Relations** subtab of the **Links** tab and window.

Deleted trace links are moved to your Architect/Requirements Recycle Bin and can be restored until your Recycle Bin is emptied. For more information, see [Restoring Objects](#) or [Emptying Your Architect/Requirements Recycle Bin](#) in chapter 4, *Maintaining a Project*.

To delete trace links for an object:

1. Do one of the following:
 - For an object in the content table, select the object and click the **Links** tab, and then click the **Relations** subtab.
You can open the **Links** window for this object by clicking the **Open tab** button on the notebook pane's toolbar.
 - For an object in the **Where Used** or **Versions** tab:
 - With the **Links** tab on top in the notebook pane, click the **Open tab** button on the toolbar to open the **Links** window, and then click the **Relations** subtab.
 - Select the object in the **Where Used** or **Versions** tab.

In the **Start** pane, defining trace links to the object are displayed in the link table, at the right of the pane. At the left, the object table displays the object from which each defining trace link originates.

In the **End** pane, complying trace links from the object are displayed in the link table, at the right of the pane. At the left, the object table displays the object at which each complying trace link is completed.

2. In either link table, select the trace links that you want to delete.
You can select one trace link or a group of nonadjacent or adjoining trace links. You cannot select from both link tables at the same time.
3. Pull down the **Edit** menu and choose **Delete**.



Delete is available also when objects are selected in the **Trace** subtab. However, trace links are identified explicitly only in the link tables of the **Relations** subtab. Siemens PLM Software recommends that you delete only from the link tables unless you are certain of the trace links between objects in the other subtab.



Delete is disabled if the selection is in either object table.

A message asks if you are sure you want to delete the selected objects.

4. Click **Yes** to move the selected trace links to your Recycle Bin.

Procedure Notes

Step 3: You can also right-click the selection and choose **Delete** from the pop-up menu. You can reverse this action by pulling down the **Edit** menu and choosing **Undo**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Generic Links

Generic link is a new database object type that can be sub-typed to support the application-specific linking needs. Specifically, the generic link supports the **DiagramElement** link for the Sparx interface.

Generic links and the linked objects appear on the **Relations** subtab of the **Links** tab. There is no visual indication of whether an object participates in the generic links.



- There are no user interface commands for creating generic links. The generic links can be created only through the API methods. However, once created, you can view and interact with generic links as described in this section.
- The standard object-level access mechanisms of the Architect/Requirements apply to the generic links.
- The generic link type definition is created while creating or upgrading the database. Application developers are responsible for the installation of the application-specific subtypes.
- The **Create Version** command creates a copy of a generic link to the new version. This means when an object with generic links is versioned, copies of the generic links exist to both the new and the old versions. For more information, see [Creating Versions](#) in chapter 10, *Working With Versions*.

Customizing Generic Links

The properties of a generic link can be viewed and modified in the same way as a trace link. The default behavior of a generic link is similar to that of a trace link for database operations such as import, export, and delete.

The behavior of a generic link can be customized using:

- Property settings on a generic link type (subtype) definition.
- Standard API methods, such as, **createLinks**, **deleteLinks**, **setValue**, **getValue**, and **getList**. For more information, see the *Systems Architect/Requirements Management API Reference* manual.
- The activators. For more information about the activators, see the *Systems Architect/Requirements Management API Reference* manual.

Chapter 8: Recording Supplementary Information With Notes

This chapter contains an overview of notes and instructions for attaching notes to objects and for editing and viewing notes.

Overview of Notes

Whereas requirements define the product that the customer wants, notes record supplementary information about the product development project. For example, notes can record phone conversations with the customer, minutes of project meetings, and new regulations that affect the project.

Depending on the importance of its information, a note may be either permanent or temporary. A note communicating a management decision could exist throughout the project's duration. In contrast, a note presenting a comment or a question to other team members may be deleted after it serves its purpose.

For informal purposes, or for recording information that applies only to certain objects, notes provide greater flexibility than do object properties. Each object type has a predefined set of properties, which apply to all objects of that type, and every object of a given type will always have those same properties. However, notes can be attached to a particular object at any time, in any number, and for any reason.

Although notes serve a relatively unstructured purpose, some formal usage conventions can be helpful. For example, note names based on purpose and content provide greater clarity for team members and for other users. Consistent naming conventions make notes easier to find, and make their interest to a user more apparent without the need for opening and reading the content. Names like **Decision Rationale**, **Design Note**, and **Testing Comments** are common in Architect/Requirements.

For entering note content, Architect/Requirements provides the following:

- A plain text editor.
- A rich text editor.
- An interface with Microsoft Office Word.

Attaching a Note to an Object

You can attach a note to an object selected in any of the following views:

- The navigation tree.
- The hierarchical content table.
- The **Attachments**, **Links**, and **Connectivity** tabs and windows.



In the **Attachments** tab and window, you can create a note on another note or on any other object. Notes are displayed hierarchically, and the new note becomes a child at the next lower level.

When you create a note, you choose the format in which you want to enter the initial content. After the note's creation, its **Text Format** property value shows the content format. You can choose one of the following formats:

- Plain text, shown by the **Text** value.
- Rich text, shown by the **HTML** value.
- Microsoft Office Word, shown by the **MHTML** value.

You also assign a subtype when you create the note. The subtype can be the default subtype, **Note**, or a user-defined subtype created by your project administrator. You can change the **Text Format** and **Subtype** properties after the note's creation. For more information, see [Editing the Properties of a Selected Object](#) in chapter 9, *Working With Object Properties*.

Creating a Plain Text Note

1.

Select the owning object for the note, pull down the **File** menu, and choose **New→Note Format→Plain Text Note**.

The **Create Note On** dialog window is displayed.



You can also display this dialog window by clicking the **Create New Note** button on the toolbar. However, this button invokes the editor for the last note you created, which may or may not be plain text.

2. In the **Name** field, enter a name that reflects the note's purpose.

3. In the **Type** field, select the subtype for the note.



When your project administrator adds or deletes note subtypes, the changes are not automatically reflected in the **Type** field. You can click **Refresh** to ensure that this field shows the current list of note subtypes.

4. Do one of the following:

- Enter initial plain text content in the text pane, and then click **OK**.



Instead of clicking **OK**, you can click **Use Rich Text** to transfer the plain text to the rich text editor. To transfer the plain text to Microsoft Office Word, you can click **Open in Word** instead of **OK**. For more information, see [Creating a Rich Text Note](#) and [Creating a Note in Microsoft Office Word](#), later in this chapter.



You cannot go back to rich text from Word.

- To create an empty note without initial content, click **OK**.

The **Attachments** tab or window displays the new note.



If the owning object is in that tab or window, you may need to click the owner's plus sign (+) to see the new note.

Procedure Notes

Step 1: You can also right-click the owning object and choose **New→Note Format→Plain Text Note** from the pop-up menu. Or, press control-N.

Step 4: You can reverse this action by pulling down the **Edit** menu and choosing **Undo Rename**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Creating a Rich Text Note

1.

Select the owning object for the note, pull down the **File** menu, and choose **New→Note Format→Rich Text Note**.

The Create Note On dialog window is displayed.



If the **Create Note On** dialog window is already displayed in the plain text mode, click **Use Rich Text** at the bottom of the dialog window.



You can also display this dialog window by clicking the **Create New Note** button on the toolbar. However, this button invokes the editor for the last note you created, which may or may not be rich text.

2. In the **Name** field, enter a name that reflects the note's purpose.

3. In the **Type** field, select the subtype for the note.



When your project administrator adds or deletes note subtypes, the changes are not automatically reflected in the **Type** field. You can click **Refresh** to ensure that this field shows the current list of note subtypes.

4. Do one of the following:

- Enter initial rich text content in the text pane, and then click **OK**.



Instead of clicking **OK**, you can click **Open in Word** to transfer the rich text to Microsoft Office Word. For more information, see [Creating a Note in Microsoft Office Word](#), later in this chapter.



You cannot go back to rich text from Word.

Use the buttons above the pane to apply rich text formatting. You can rest the pointer on each button to see its description in a tooltip.



You can create hyperlinks, but you cannot follow them from the rich text editor. To navigate to the target object of a hyperlink in the text pane, display the note content in the **Preview** tab or open the note in Word. Then, follow the hyperlink from that content.

- To create an empty note without initial content, click **OK**.

The **Attachments** tab or window displays the new note.



If the owning object is in that tab or window, you may need to click the owner's plus sign (+) to see the new note.

Procedure Notes

Step 1: You can also right-click the owning object and choose **New**→**Note Format**→**Rich Text Note** from the pop-up menu.

Step 4: You can reverse this action by pulling down the **Edit** menu and choosing **Undo New Note**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Creating a Note in Microsoft Office Word

Microsoft Office Word 2013 or Microsoft Office Word 2016 must be installed on your computer.

1.

Select the owning object for the note, pull down the **File** menu, and choose **New→Note Subtype→MS Word Note**.

You can also right-click the object and choose the options from the pop-up menu.

The Create Note On dialog window is displayed.



You can also display this dialog window by clicking the **Create New Note** button on the toolbar. However, this button invokes the editor for the last note you created, which may or may not be Microsoft Office Word.

2. In the **Name** field, enter a name that reflects the note's purpose.

3. In the **Type** field, select the subtype for the note.



When your project administrator adds or deletes note subtypes, the changes are not automatically reflected in the **Type** field. You can click **Refresh** to ensure that this field shows the current list of note subtypes.

4. Do one of the following:

- To enter initial content:

.

Click **Open in Word**.

The note is created in the **Attachments** tab or window while an **.mhtml** file opens in Word. You can reverse this action by immediately closing the file, and then pulling down the **Edit** menu and choosing **Undo New Note**, clicking the **Undo** button on the toolbar, or pressing control-Z.

.

Enter the content and save the Word file to commit the content to the database. For more information, see [Entering and Changing Requirement Content in Microsoft Office Word](#) in chapter 5, *Managing Requirements*.

- To create an empty note without initial content, click **OK**.

The **Attachments** tab or window displays the new note. You can reverse this action by pulling down the **Edit** menu and choosing **Undo New Note**, clicking the **Undo** button on the toolbar, or pressing control-Z.



If the owning object is in that tab or window, you may need to click the owner's plus sign (+) to see the new note.



You can also create an **MHTML** note through the Select Subtype dialog window. To display this dialog window, pull down the **File** menu or right-click the selected object, and then choose **New→Subtype**. Or, click the **Create a new subtype** button on the toolbar or press control-U.

When you select a note subtype and click **OK**, the **Attachments** tab or window displays the note with a default name in an open text field. Enter the name and press the enter key. Then, you can open the note in Word to enter content.

To reverse this action, you can pull down the **Edit** menu and choose **Undo Create Subtype**, click the **Undo** button on the toolbar, or press control-Z.

Editing a Note

Each note has a **Text Format** property, whose value shows the note's content format. Architect/Requirements provides a content editor for each of the following formats:

- Plain text, shown by the **Text** value.
- Rich text, shown by the **HTML** value.
- Microsoft Office Word, shown by the **MHTML** value.

You can add the **Text Format** column to the **Attachments** tab and window. Or, you can view this property for a selected note by pulling down the **File** menu and choosing **Properties** to display the Edit Properties dialog window. For more information, see [Adding and Removing Columns](#) in chapter 9, *Working With Object Properties*.



To edit an **MHTML** note, Microsoft Office Word 2003 or 2007 must be installed on your computer.

To edit a note:

1.

For the object to which the note is attached, display the notes by doing one of the following:

- For an object in the navigation tree, the hierarchical content table, or the Recycle Bin, select the object and click the **Attachments** tab.

You can open the **Attachments** window for this object by clicking the **Open tab** button on the notebook pane's toolbar.



Deleted notes in the Recycle Bin cannot be edited. However, notes attached to other deleted objects can be displayed in the **Attachments** tab or window, and can then be edited.

- For an object in the **Attachments** tab or window, including a top level note, click the plus sign (+) to display the notes at the next lower level.
- For an object in the **Links, Connectivity, Where Used**, or **Versions** tab:
 - With the **Attachments** tab on top in the notebook pane, click the **Open tab** button on the toolbar to open the **Attachments** window.
 - In the **Links, Connectivity, Where Used**, or **Versions** tab, select the object to display its notes in the **Attachments** window.

2. Select the note, and then pull down the **File** menu and choose **Open**.



If necessary, click the plus signs for child notes to display the note that you want to select.

For a plain text or rich text note, the Edit dialog window is displayed.

For a Microsoft Word note, Word opens an **.mhtml** file.

3. Do one of the following:

- In the Edit dialog window:

- . Enter content changes in the text pane.



In the rich text editor, use the buttons above the pane to apply rich text formatting. You can rest the pointer on each button to see its description in a tooltip.

You can also do the following:

- Change the name in the **Name** field.
- Change the subtype in the **Type** field.

When your project administrator adds or deletes note subtypes, the changes are not automatically reflected in the **Type** field. You can click **Refresh** to ensure that this field shows the current list of note subtypes.

- . Click **OK** to close the dialog window and save the changes in the database.



- Instead of clicking **OK**, you can click **Use Rich Text** to transfer plain text changes to the rich text editor.
- To switch from plain text or rich text to Microsoft Word, you can click **Open in Word** instead of **OK**.

-

In Microsoft Word:

- . Enter content changes as you do in a typical Word file.

For more information, see [Entering and Changing Requirement Content in Microsoft Office Word](#) in chapter 5, *Managing Requirements*.

- . To save the changes in the database, choose **Save** from Word's **File** menu.

Procedure Notes

Step 1: You can also right-click the note and choose **Open** from the pop-up menu. Or, double-click the note.

Enabling Text Property Editing

When editing is disabled in a **Text** property cell, the note has a **Text Format** property value of **MHTML**. This value indicates content saved in Microsoft Office Word.

Disabled cell editing is indicated also by certain visual cues in the **Text** property cell. Depending on the client view where the property is displayed, the cues are:

- A shaded background and blue text for disabled cells in the **Attachments** tab and window, the **Links** tab and window, and the Search Results dialog window.
- Dimmed text for disabled cells in the **Properties** floating window and the Edit Properties dialog window.



Although you can edit disabled **Text** property cells in a live Excel file, edits are not saved in the Architect/Requirements database. An error message is displayed when you click outside the cell. For more information, see [Using the Live Excel Interface](#) in chapter 9, *Working With Object Properties*.

For a note with such a cue, you can enable direct editing in the **Text** property cell by changing the **Text Format** property value to **Text**. For more information, see [Editing Text Property Cells](#), earlier in this chapter.



The next time the cell is edited, current headings, body text, lists, and table text are converted to plain text, and all text is concatenated into one paragraph. All formatting is removed, including graphics, table grids, and other content elements that may be saved in Word. For more information, see [Editing a Note](#), earlier in this chapter.

To enable Text property editing:

1. Select the note, and then do one of the following:
 - To use the **Properties** window, click the **Properties** tab, and then click the **Open tab** button on the notebook pane toolbar.
 - To use the **Edit Properties** dialog window, pull down the **File** menu and choose **Properties**. Or, right-click the note and choose **Properties** from the pop-up menu.

2.

In the **Values** column, double-click the **Text Format** property value.

The Single-Choice dialog window is displayed.

3. Select the **Text** option button, and then click **OK**.



This action changes the text format to plain text and removes all other content elements, effective the next time the cell is edited.

A warning message states that saving as plain text causes all formatting, pictures, and objects in the text to be lost on the next text edit.

4. To continue, click **OK** to close the Single-Choice dialog window.
5. Click **Close** to apply the change and close the Edit Properties dialog window.

Editing Text Property Cells

Plain text content can be edited directly in the cells of the **Text** property column. You can add this column in the **Attachments** tab and window, where notes are displayed for the object selected in the navigation tree or the content table. Also, notes can be output to the Search Results dialog window, where the **Text** property can be added. The **Text** property is displayed by default in the **Properties** floating window and the Edit Properties dialog window. For more information, see [Adding and Removing Columns](#) in chapter 9, *Working With Object Properties*.

You can use any of these client views to edit the **Text** property for one note. For two or more notes, you can use only the Edit Properties dialog window.

To edit a Text property cell for one note:

In the **Attachments** tab or window, the **Links** tab or window, or the Search Results dialog window:

1. With the **Text** property added, double-click the cell that you want to edit.



A shaded background and blue text indicate that cell editing is disabled. The note's **Text Format** property value is **MHTML**, for content saved in Microsoft Office Word. Cell editing can be enabled by changing the **Text Format** value to **Text**. For more information, see [Enabling Text Property Editing](#), later in this chapter.

A text field opens in the cell. Scroll bars are provided to the right if the text length exceeds the width of the **Text** column.

2. Enter your changes, and then press the enter key or click outside the field.

In the **Properties** floating window or the Edit Properties dialog window:



To see more text in the **Text** column, change the width before you begin. Scroll bars are not provided in the cell.

1. Double-click the **Text** property value to open a text field in the cell.



Dimmed text indicates that cell editing is disabled. The note's **Text Format** property value is **MHTML**, for content saved in Microsoft Office Word. Cell editing can be enabled by changing the **Text Format** value to **Text**. For more information, see [Enabling Text Property Editing](#), later in this chapter.

2. Enter your changes, and then press the enter key or click outside the field.

To display the **Properties** window, select the object to which the note is attached, click the **Properties** tab, and then click the **Open tab** button on the notebook pane toolbar.

To display the Edit Properties dialog window, select the note, pull down the **File** menu, and choose **Properties**.

To edit Text property cells for two or more notes:



The **Text** property cannot be displayed if any one selected note has a **Text Format** property value of **MHTML**, for content saved in Microsoft Office Word. Before you start this procedure, ensure that the **Text Format** value is **Text** for each note that you intend to edit. You can set this value if necessary. For more information, see [Enabling Text Property Editing](#), later in this chapter.

1. Select the notes in the **Attachments** tab or window, the **Links** tab or window, or the Search Results dialog window.

You can select nonadjacent and adjoining notes of the same subtype or of mixed subtypes.

2. Pull down the **File** menu and choose **Properties**.

The Edit Properties dialog window is displayed.

3. In the **Value** column, double-click the **Text** property cell.

A text field opens in the cell.

4. Enter your changes, and then press the enter key or click outside the field.
5. Click **OK** to close the dialog window.

Procedure Notes

Step 2: You can also right-click the selection and choose **Properties** from the pop-up menu.

Viewing Note Content

The **Attachments** tab and floating window display the notes for the object selected in the navigation tree, the hierarchical content table, or your Recycle Bin. For the object selected in the **Links, Connectivity, Where Used, or Versions** tab, notes are displayed in the **Attachments** window.

Each note has a **Text Format** property, whose value shows the note's content format. A read-only mode is provided in the content editor for each of the following formats:

- Plain text, shown by the **Text** value.
- Rich text, shown by the **HTML** value.
- Microsoft Word, shown by the **MHTML** value.

Also, the **Preview** window can display read-only content in all formats. For more information, see [Preview Tab](#) in chapter 3, *Using the Architect/Requirements Main Window*.

You can add the **Text Format** column to the **Attachments** tab and window. Or, you can view this property for a selected note by pulling down the **File** menu and choosing **Properties** to display the Edit Properties dialog window. For more information, see [Adding and Removing Columns](#) in chapter 9, *Working With Object Properties*.



To view an **MHTML** note in Word, Microsoft Word 2013 or Microsoft Word 2016 must be installed on your computer.

To view note content:

1. For the object to which the note is attached, display the notes by doing one of the following:
 - For an object in the navigation tree, the hierarchical content table, or the Recycle Bin, select the object and click the **Attachments** tab.

You can open the **Attachments** window for this object by clicking the **Open tab** button on the notebook pane's toolbar.



Deleted notes in the Recycle Bin cannot be opened for viewing. However, notes attached to other deleted objects can be displayed in the **Attachments** tab or window, and can then be opened.

- For an object in the **Attachments** tab or window, including a top level note, click the plus sign (+) to display the notes at the next lower level.
- For an object in the **Links, Connectivity, Where Used, or Versions** tab:
 - With the **Attachments** tab on top in the notebook pane, click the **Open tab** button on the toolbar to open the **Attachments** window.
 - In the **Links, Connectivity, Where Used, or Versions** tab, select the object to display its notes in the **Attachments** window.



In the **Attachments** tab or window, you can rest the pointer on the note's **Text** property cell to see the plain text content in a tooltip.

2. Select the note, and then do one of the following:



If necessary, click the plus signs for child notes to display the note that you want to select.

- Pull down the **File** menu and choose **Open Read-Only**.

For a plain text or rich text note, the Edit dialog window displays the content in read-only mode.

For a Microsoft Word note, the content is displayed in a read-only **.mhtml** file. You can also print the content and send it by E-mail and fax.

- Use the **Preview** floating window by doing one of the following:

- . For an object in the navigation tree, the hierarchical content table, or the Recycle Bin:

- Select the object, and then click the **Attachments** tab to display the object's notes.

You can open the **Attachments** window for this object by clicking the **Open tab** button on the notebook pane's toolbar.

- With the **Preview** tab on top in the notebook pane, click the **Open tab** button on the toolbar to open the **Preview** window.
- In the **Attachments** tab or window, select the note to display its content in the **Preview** window.

- . For an object in the **Links, Connectivity, Where Used, or Versions** tab:

- With the **Attachments** tab on top in the notebook pane, click the **Open tab** button on the toolbar to open the **Attachments** window.
- In the **Links, Connectivity, Where Used, or Versions** tab, select the object to display its notes in the **Attachments** window.
- With the **Preview** tab on top, click the **Open tab** button to open the **Preview** window.
- In the **Attachments** window, select the note to display its content in the **Preview** window.

To print the content, you can right-click in the content area and choose **Print** from the pop-up menu.

Procedure Notes

Step 2: You can also right-click the note and choose **Open Read-Only** from the pop-up menu.

Chapter 9: Working With Object Properties

This chapter provides an overview of property types and values and contains instructions for editing properties in the Architect/Requirements client and with live Excel, the Architect/Requirements interface with Microsoft Excel.

Overview of Object Properties

Properties are named values that determine the nature and behavior of all objects of a given object type. Every property belongs to one of two property types, either *system-defined* or *user-defined*.

System-Defined Properties

Architect/Requirements assigns system-defined properties to each built-in object type. The system-defined properties of an object type are inherited by all subtypes based on that object type.

Some system-defined properties record general information, such as an object's creation time, that applies to all object types. Other system-defined properties have a more specific purpose that applies to only one or two object types, for example, the paragraph number of a requirement, or the text of a requirement or a note.

Each system-defined property falls into one of two categories:

- A *read-only* property has a value that can be changed only by the system itself. The user cannot change such a value, except as an indirect result of certain other actions on an object. For example, an object's change time is a read-only property.
- An *editable* property has a default value that can be changed directly by the user. For example, an object's name is an editable property.

The **Properties** tab and floating window display all viewable system-defined properties for the object selected in the content table. Other views in the Systems Engineering and Requirements Management module display certain system-defined properties by default. In these views, default properties can be removed and other system-defined properties can be added. For more information, see [Properties Tab](#) and [Using Tabs in Floating Windows](#) in chapter *Using the Architect/Requirements Main Window*; [Adding and Removing Columns](#), later in this chapter; and appendix [System-Defined Properties in the Systems Engineering and Requirements Management Module](#).

User-Defined Properties

User-defined properties are created locally by an Architect/Requirements project administrator. By assigning these properties to type definitions in the Administration module, the project administrator can extend the built-in object types for specific purposes. Subject to user permissions, all user-defined properties are editable.

Property Values

Whether system-defined or user-defined, each property value is one of the following:

-

Choice

The value is selected from a predefined list of choices. A choice list can be defined as either of the following:

- o *Single-choice*

From a single-choice list, you can select only one choice for the value.

- o *Multiple-choice*

From a multiple-choice list, you can select any combination of choices, and that combination becomes the value.

-

Date

The value is a calendar date, and can be defined to include the time of day. User inputs are checked, and only valid calendar dates are accepted. Sorting is by the date's chronological value, regardless of its alphanumeric representation.

-

Numeric

The value is a number, and can be defined to allow a variety of floating-point formats. User inputs are checked, and only valid numeric values are accepted. Sorting is by the number's binary value, regardless of its string representation.

-

Text

The value is any string of plain text, including a blank string. Sorting is alphanumeric.

Releasing Reservation of Objects

When you open an object, the Architect/Requirements places a reservation or lock on that object that prevents other users from modifying it simultaneously. You can release this reservation by choosing the **Tools**→**Release Reservation** command. You can only release your own reservations, however, administrators can also release reservations owned by other users.

The **Release Reservation** command should not be used when you are editing the contents of an object. However, it is helpful to use this command if there was a system problem that caused the reservation to be held even after closing the object after modifications.

One of the common uses of the **Release Reservation** command is when the activators are edited in multi-pane editors. You must use this command when other users want to edit the same activator. This command is not needed if you want to edit the activator again during the same session.

Customizing Views of Property Columns

Custom column settings let you easily see and work with a set of properties. For example, in a given folder, you can group objects by a common purpose, and apply a view with property columns related to the purpose.

Custom views are available in:

- The hierarchical content table.
- The **Attachments** tab or window.
- The **Trace** subtab of the **Links** tab or window.
- The **Connectivity** tab or window.
- The **Versions** tab or window.
- The **Defining Object Traceability** and **Complying Object Traceability** windows.
- The Search Results dialog window.



Custom views are available also through the **Saved Views** filtering option in the **Properties** tab and window. For more information, see [Filtering Property Tables](#), later in this chapter.

Views are project specific, residing in the **Reports and Formatting** folder in the Administration module. If you have questions about custom property views, consult your project administrator.



- Views are available also in the Basic pane of the Search module. You can specify the property columns for each search by selecting a view in the field at the bottom of the **Output Options** section. The columns in the selected view are output to the Search Results dialog window.
- Views are not enabled when the root node or your Architect/Requirements Recycle Bin is selected in the navigation tree.
- When a child folder is created in the content table of the same type as the parent, the child inherits the default view named in the parent's **Default View** property value if that value has been set. If the child folder is of a different subtype, it does not inherit the parent's **Default View** property value. However, if the parent's view is modified locally, the child inherits the column settings in the database, not the local modifications. For more information, see [Modifying a View Locally](#), later in this chapter.

If no default view is set for the parent, the child's view is determined by the conditions described in table [Conditions Determining View in Content Table](#), later in this chapter.

- When a folder's subtype is changed, the folder's **Default View** property value does not change.

If the folder is a parent of other folders, the subtype change does not extend to the children.

Conditions That Determine the Column View in the Content Table

Each folder selected in the navigation tree can display a unique set of columns in the content table. Architect/Requirements evaluates certain conditions to determine the default view for the content table. Table 9-1 describes these conditions and views, from the first condition evaluated to the last.

Table 9-1. Conditions Determining View in Content Table

Condition	View
Local modifications to column settings are recorded on your computer.	The columns that you set only by adding, removing, rearranging, resizing, or sorting columns. These settings are not saved in the database. For more information, see Modifying a View Locally , later in this chapter.
Local modifications are not recorded or are overridden by the <Default View> option in the View field.	The view named in the Default View property value for the folder. For more information, see Setting the Default View for a Folder , later in this chapter.
No default view is set for the selected folder.	The default view for the folder type definition in the Administration module. If you have questions about type definitions, consult your project administrator.  For the selected folder, the Default View property value is blank when the type definition's default view is used.
No default view is set for the folder type definition.	The default view for your user name. For more information, see Setting the Default View for Your User Name , later in this chapter.
No default view is set for your user name.	The system default view. In this view, the property columns are Attachment Count, Trace Link Count, Number, ROIN, Create Time, Create User, and Type Name .  The View list displays *System Settings* when the system default view is used.

Applying a View in the Content Table

For each folder that you select in the navigation tree, you can apply a unique view of property columns to display in the content table. Using the **View** list, you can apply *public* views, which are visible to all users in the project. Also, you can apply non-public views that you create and save in the database. For more information, see [Creating a Column View](#), later in this chapter.

The **View** list is located to the right of the **Address** bar above the content table. Your selection is recorded only on your computer, not in the database, and persists each time you select the folder.

To apply a view in the content table:

- Do either of the following:
 - Click the **View** list and select the name of the view.
 - Right-click inside the content table, choose **Select View** from the pop-up menu, and choose the name of the view.

The content table displays the columns that are saved in the database for the view. The **View** list shows the view name, and also your user name if you created the view and it is not yet public.

The view persists when you refresh the data by pulling down the **View** menu and choosing **Refresh**, or by clicking the **Refresh** button on the toolbar.



- The **<Default View>** option overrides any local modifications by applying the default view for the folder. If you select **<Default View>** in the **View** list, the list displays a blank value instead of the actual name of the default view. The default view is determined by the conditions described in table [Conditions Determining View in Content Table](#), earlier in this chapter. For more information, see [Modifying a View Locally](#), later in this chapter.

- The ***System Settings*** option applies the Architect/Requirements default columns. The default columns are the following:

- **Attachment Count**
- **Trace Link Count**
- **Number**
- **ROIN**
- **Create Time**
- **Create User**
- **Type Name**

Applying a View in the Notebook Pane

You can apply a custom view in the **Attachments**, **Connectivity**, and **Versions** tabs and windows. You can also apply a view in the **Trace** subtab of the **Links** tab and window, and in the **Defining Object Traceability** and **Complying Object Traceability** windows.

To apply a view in the notebook pane:

- Right-click inside the tab or window, choose **Select View** from the pop-up menu, and choose the name of the view from the submenu.



You can apply the Architect/Requirements default columns for the tab or window by choosing ***System Settings***.

The tab or window displays the columns that are saved in the database for the view. The view name is checked on the pop-up submenu.

The view persists when you refresh the data by pulling down the **View** menu and choosing **Refresh**, or by clicking the **Refresh** button on the toolbar.



In the **Properties** tab and window, you can use the **Saved Views** filtering option to apply a custom view. For more information, see [Filtering Property Tables](#), later in this chapter.

Creating a Column View

Using the current column settings, you can create a new view in the database. You can then set unique columns for the new view.

You can create a custom view in:

- The hierarchical content table.
- The **Attachments**, **Connectivity**, and **Versions** tabs or windows, and the **Trace** subtab of the **Links** tab or window.
- The **Defining Object Traceability** and **Complying Object Traceability** windows.
- The Search Results dialog window.



If you change column settings before step 1, you modify the current view on your computer. The changes are recorded locally, not in the database, and they persist until you reapply that view. For more information, see [Modifying a View Locally](#) and [Applying a View in the Content Table](#), later in this chapter.

To create a column view:

1.

Pull down the **View** menu and choose **Save As** to display the Save View dialog window.

You can also right-click inside the content table or the tab or window, and then choose **Save As** from the pop-up menu.

2. Enter a unique name in the **Enter new view name** field, and then click **OK**.



If you enter a view name that already exists in the project, an error message is displayed and you must repeat steps 1 and 2.

In the content table, the **View** field displays the view name and your user name. In the tab or window, the view name is checked on the **Select View** pop-up submenu.

To customize the view, you can choose unique column settings, and then pull down the **View** menu and choose **Save**. For more information, see [Adding and Removing Columns](#), [Rearranging Columns](#), [Resizing Columns](#), and [Setting the Sort Column](#), later in this chapter.



As the creator, you can mark the view to be made visible to all users in the project. First, switch to the Administration module and select the view in the **Reports and Formatting** folder. Next, in the **Properties** tab or window, double-click the **Shared State** property value to display the Single-Choice dialog window. Then, check the **Pending** check box and click **OK**. A project administrator can set the value to **Public**.

Setting the Default View for a Folder

For the folder selected in the navigation tree, the default view in the content table is determined by the conditions described in table [Conditions Determining View in Content Table](#), earlier in this chapter.



You cannot set the default view in the notebook pane.

Architect/Requirements references the default view for the folder when one of the following conditions exists:

- Column settings are not changed locally. For more information, see [Modifying a View Locally](#), later in this chapter.
- Local modifications are overridden by the **<Default View>** option in the **View** field. For more information, see [Applying a View in the Content Table](#), earlier in this chapter.

A default view may be set for the folder type definition in the Administration module.

Architect/Requirements references that default view if one is not set for the individual folder in the Systems Engineering and Requirements Management module.

By setting the default view for the folder, you override the type definition's default view with one that you specify. Also, you can change the folder's default view if one already exists.



This procedure requires one of the following:

- You must have **Project Administrator** privilege for the project.
- Your user name must be included in the **Full Control** property value of the security profile that is applied to the folder. If you have questions about security profiles, consult your project administrator.

To set the default view for a folder:



This procedure does not change the current view and column settings.

1. In the navigation tree or the hierarchical content table, select the folder.
The **Properties** tab displays the folder's properties.
2. In the **Value** column, double-click the **Default View** value.
3. In the Single-Choice dialog window, check the check box for the default view and click **OK**.

Procedure Notes

Step 1: Instead of the **Properties** tab, you can use the **Properties** floating window by clicking the **Open tab** button on the notebook pane toolbar. Or, pull down the **File** menu and choose **Properties** to use the Edit Properties dialog window.

Setting the Default View for Your User Name

To determine the view for a selected folder, Architect/Requirements references the default view for your user name when both of the following conditions exist:

- Column settings are not changed locally, or local modifications are overridden by the **<Default View>** option in the **View** field. For more information, see [Modifying a View Locally](#) or [Applying a View in the Content Table](#), earlier in this chapter.
- A default view is not set for the folder, or for the folder type definition in the Administration module. For more information, see [Setting the Default View for a Folder](#), earlier in this chapter. If you have questions about type definitions, consult your project administrator.

Your default view applies to each folder in the navigation tree for which all conditions exist. For more information, see table [Conditions Determining View in Content Table](#), earlier in this chapter.

The project administrator may set a default view on your user object in the Administration module. You can change that setting in the Systems Engineering and Requirements Management module, or you can set your own default view if one does not exist. If you have questions about user objects, consult your project administrator.



You cannot set the default view in the notebook pane.

To set the default view for your user name:

- With the view applied, pull down the **View** menu and choose **Set View As Default**.

The view name becomes the value of your user object's **Default View** property.

Modifying a View Locally

Local modifications to column settings override the view that is currently applied. These column settings are saved only on your computer.

The column view can be modified in the following:

- The hierarchical content table.
- The **Attachments**, **Connectivity**, and **Versions** tabs or windows, and the **Trace** subtab of the **Links** tab or window.
- The **Defining Object Traceability** and **Complying Object Traceability** windows.
- The Search Results dialog window.

You can add and remove columns, rearrange columns, resize columns, and sort by any column. For more information, see [Adding and Removing Columns](#), [Rearranging Columns](#), [Resizing Columns](#), and [Setting the Sort Column](#), later in this chapter.



- In the content table, local modifications apply only to the folder selected in the navigation tree. The settings revert to those in the database if the modified view is the folder's default view and you select **<Default View>** in the **View** list. For more information, see [Applying a View in the Content Table](#), earlier in this chapter.
- You do not need special privileges or permissions to modify any view locally. Your changes do not affect the view object in the database.
- Local modifications are not retained when you upgrade to a newer version of Architect/Requirements.

Adding and Removing Columns

You can add and remove property columns in the hierarchical content table; the **Attachments**, **Links**, **Where Used**, and **Versions** tabs and windows; and the Search Results dialog window.

In each table, you can add and remove columns for all property types, for one type, or for mixed types. You can choose a different set of columns in each view, and in each project to which you have access. Architect/Requirements saves your last settings from the current session, and displays those columns when you start a new session.



- You cannot add or remove columns in the **Properties** tab and window.
- You cannot remove the **Name**, **Defining Trace**, or **Complying Trace** column, nor can you remove the object type indicator column in the **Attachments** and **Where Used** tabs and windows.

To add and remove columns:

1. Do one of the following:
 - With the content table or a fixed tab activated, pull down the **View** menu and choose **Format Columns**.
 - In a floating tab window, right-click any column heading.

Architect/Requirements displays the Column Settings dialog window, which lists the names of all property columns for the view. The list shows whether each property is system-defined, editable, and displayed by default.

2. Do one or more of the following:
 - To add all columns for all property types, click **All** in the **Select by Group** list.
 - To remove all property columns, click **None** in the **Select by Group** list.
 - To display only the default columns, click **Default** in the **Select by Group** list.
 - To add all columns for only one property type, do one of the following in the **Select by Group** list:
 - Select **System** for properties that Architect/Requirements assigns automatically to built-in object types. For more information, see appendix [System-Defined Properties in the Systems Engineering and Requirements Management Module](#).
 - Select **User-Defined** for properties that your project administrator assigns to built-in object types and custom subtypes.
 - Select **Editable** for properties whose values you can change.
 - To add or remove columns of mixed property types:
 - To add an undisplayed column, do one of the following:
 - Check the check box to the left of the property name in the table.
 - Select the property name in the table, and then click **Show**.
 - Double-click the property name in the table.
 - To remove a displayed column, do one of the following:
 - Clear the check box to the left of the property name in the list.
 - Select the property name in the list, and then click **Hide**.
 - Double-click the property name in the list.
3. Click **OK**.

The Column Settings dialog window closes, and the table displays your choices.

Rearranging Columns

You can change the position of the property columns in the hierarchical content table; the **Attachments**, **Links**, **Where Used**, and **Versions** tabs and windows; and the Search Results dialog window. For more information, see [Hierarchical Content Table](#), [Attachments Tab](#), [Links Tab](#), [Where Used Tab](#), [Versions Tab](#), and [Using Tabs in Floating Windows](#), earlier in this chapter; and chapter 11, [Using the Search Module](#).

In each table, you can use the Column Settings dialog window to move multiple columns at the same time. Also, you can use the mouse to move one column at a time. You can set different column positions in each table.



- You cannot rearrange columns in the **Properties** tab and window.
- You cannot move the leftmost column in a table. The leftmost column is the **Name**, **Defining Trace**, or **Complying Trace** column in the related view, or the object type indicator column in the **Attachments** and **Where Used** tabs and windows.

To rearrange columns with the Column Settings dialog window:

1. To display the Column Settings dialog window, do one of the following:
 - With the content table or a fixed tab activated, pull down the **View** menu and choose **Format Columns**.
 - In a floating tab window, right-click any column heading.
2. Select a property name in the dialog window, and then do one of the following:
 - To move the column to the first position after the leftmost column in the table, click **Move to Top**.
 - To move the column one position to the left in the table, click **Move Up**. You can click **Move Up** until the name reaches the intended position.
 - To move the column one position to the right in the table, click **Move Down**. You can click **Move Down** until the name reaches the intended position.

For each additional column that you want to move, repeat step 2.

3. Click **OK** to close the dialog window and move the columns to their new positions.

To rearrange columns with the mouse:

- For each column that you want to move, point to the heading, hold down the left mouse button, and drop the heading in the new position.

Resizing Columns

You can change the width of every column in all tables.

To resize columns:

1. Point to the right border of a column heading until the pointer becomes a horizontal double arrow.
2. Hold down the left mouse button, move the border left or right, and then release the mouse button.

For each additional column that you want to resize, repeat steps 1 and 2.



You can resize most columns automatically by double-clicking the right border of the column heading.

Setting the Sort Column

You can sort information by any column in each of the following:

- The hierarchical content table and your Architect/Requirements Recycle Bin (a sequential table).
- The **Properties**, **Attachments**, and **Where Used** tabs and windows (sequential tables), and the **Links** and **Versions** tabs and windows (hierarchical tables).
- The Search Results dialog window, either a hierarchical or a sequential table depending on the criteria specified in the Search module.

For example, you can sort by the **Name** column to see the objects in alphabetical order of their names. Or, you can sort by the **Create Time** column to arrange the information by the dates and times the objects were created. Every column can be sorted in ascending or descending order, including columns that contain indicators.

Although the sort column applies to all displayed rows in the table, hierarchical and sequential tables differ in the way the rows are sorted:

- In a hierarchical table, the objects remain at their present levels in the hierarchy, shown in the leftmost column. The rows are sorted within each level according to the sort column.
- In a sequential table, all rows are sorted in a continuous series according to the sort column. Objects in the Recycle Bin are displayed without regard to their levels in the hierarchies from which they were deleted.



This procedure changes only the order of display in the view. It does not affect the order of the objects or the properties in the database.

To set the sort column:

Point to the heading of the intended sort column, and then do one of the following:

- For ascending order, click the heading until it displays an up arrow.
- For descending order, click the heading until it displays a down arrow.

Modifying a View in the Database

When you change the column settings of an applied view, the modifications are recorded only on your computer. You can use the local column settings to overwrite those for the named view in the database. Then, the view can be applied consistently. For more information, see [Modifying a View Locally](#), [Applying a View in the Content Table](#), and [Applying a View in the Notebook Pane](#), earlier in this chapter.

The column view can be modified in the following:

- The hierarchical content table.
- The **Attachments**, **Connectivity**, and **Versions** tabs or windows, and the **Trace** subtab of the **Links** tab or window.
- The **Defining Object Traceability** and **Complying Object Traceability** windows.
- The Search Results dialog window.



To overwrite a public view, you must meet one of the following conditions:

- You must have **Project Administrator** privilege for the project.
- Your user name must be included in the **Full Control** property value of the security profile that is applied to the view. If you have questions about security profiles, consult your project administrator.

To modify a view in the database:

1. With the view applied, choose the new column settings.

The modifications are recorded locally on your computer.

2. Pull down the **View** menu and choose **Save**.

The modifications are recorded in the view object, which is located in the **Reports and Formatting** folder in the Administration module.

Procedure Notes

Step 1: You can add and remove columns, rearrange and resize columns, and set the sort column. For more information, see [Adding and Removing Columns](#), [Rearranging Columns](#), [Resizing Columns](#), and [Setting the Sort Column](#), earlier in this chapter.

Filtering Property Tables

In the **Properties** tab and floating window and the Edit Properties dialog window, you can set the properties that you want to see in the property tables. By default, each of these tables displays all viewable properties for the selected object or objects. For more information, see [Properties Tab](#) in chapter 3, *Using the Architect/Requirements Main Window*.

You can limit the properties to a specific subset through filtering options on the **View** and pop-up menus and through filter buttons in each table. For example, when you export objects to Microsoft Excel, you can use a filter to see only those properties in the table and in the Excel file. For more information, see [System-Defined Properties](#) and [User-Defined Properties](#), earlier in this chapter; appendix [System-Defined Properties in the Systems Engineering and Requirements Management Module](#); and [Exporting Objects to Microsoft Office Excel](#).



- The filter buttons in the **Properties** tab are hidden each time you start a new Architect/Requirements session. In the **Properties** tab window and the Edit Properties dialog window, the buttons are hidden each time you open the window.
- If you choose a filter from a menu, all tables display the same properties. To display different properties in a table, use the filter buttons in that table.

To show or hide the filter buttons:

Do one of the following:

- Pull down the **View** menu and choose **Filter Properties**→**Toggle Controls**.
- Right-click the table and choose **Filter**→**Toggle Controls** from the pop-up menu.
- Click the down or up arrow on the horizontal split bar above the table.



To show the buttons, you can also point to the split bar until the pointer becomes a double vertical arrow. Then, click the split bar to size the filter bar automatically, or drag the split bar to the size that you want.

To display only properties whose values you can change:

Do one of the following:

- Pull down the **View** menu and choose **Filter Properties**→**Editable**.
- Right-click the table and choose **Filter**→**Editable** from the pop-up menu.
- Click the **Editable** filter button.

To display only properties whose values cannot be changed:

Do one of the following:

- Pull down the **View** menu and choose **Filter Properties**→**Read-only**.
- Right-click the table and choose **Filter**→**Read-only** from the pop-up menu.
- Click the **Read Only** filter button.

To display only the custom properties for the project:

Do one of the following:

- Pull down the **View** menu and choose **Filter Properties**→**User Defined**.
- Right-click the table and choose **Filter**→**User Defined** from the pop-up menu.
- Click the **User Defined** filter button.

To display only system-defined properties:

Do one of the following:

- Pull down the **View** menu and choose **Filter Properties**→**System**.
- Right-click the table and choose **Filter**→**System** from the pop-up menu.
- Click the **System** filter button.

To display the properties in a saved view:

Do one of the following:

- Pull down the **View** menu and choose **Filter Properties**→**View Specific**.
- Right-click the table and choose **Filter**→**View Specific** from the pop-up menu.
- Click the **Saved Views** filter button, and then select the view from the list to the right.

If you choose a menu option and if the filter buttons are hidden, you cannot select a specific view. The properties are displayed from the saved view previously selected as the filter.

The saved view filters are unavailable if there are no public views for the project and no private views for your user name. For more information, see [Customizing Views of Property Columns](#).

To display all viewable properties:

Do one of the following:

- Pull down the **View** menu and choose **Filter Properties**→**All**.
- Right-click the table and choose **Filter**→**All** from the pop-up menu.
- Click the **All** filter button.

Editing the Properties of a Selected Object

For a selected object, you can change the values of editable properties in the **Properties** tab or floating window. You can also use the Edit Properties dialog window to change editable values while displaying another tab or window.

You can select the object in the hierarchical content table; the **Attachments, Links, Where Used, and Versions** tabs and floating windows; the **Complying Object Traceability** view; and the Search Results dialog window.



- You must have **Modify** permission for the object.
- To edit properties for a building block, a TRAM, or a diagram, you must have **Architect** privilege for the project.

To edit the properties of a selected object:

1. Select the object type indicator for the object, and then do one of the following:
 - To display the **Properties** tab, click the tab.
 - To display the Edit Properties dialog window, pull down the **File** menu and choose **Properties**.

Values that you cannot change are dimmed in the **Value** column.

2.

In the **Value** column, double-click the value that you want to change.

If the property has a choice value, one of the following dialog windows appears:

- In the Single-Choice dialog window, you can select only one choice.
- In the Multi-Choice dialog window, you can select any combination of choices.



The choice list may be set dynamically, and you may see different choices from time to time. If you have questions about the choice list, consult your project administrator.

A text field opens if the property has a date, numeric, or text value. For a date value, a calendar icon is displayed to the right of the text field.



- For date and numeric values, you can rest the pointer on the text field to see the valid format in a tooltip.
- For the system-defined property named **Text**, dimmed text in a cell indicates that editing is disabled because MHTML content is saved in Microsoft Word. You can enable this cell by changing the requirement's **Text Format** property value to **Text**, and then edit the **Text** property. For more information, see [Enabling Text Property Editing for a Requirement](#) in chapter 5, *Managing Requirements*.

3. Do one of the following:

- In the Single-Choice dialog window, click the button for the new value, and then click **OK** or press the enter key.

You can click **Set to Default** to select the button for the default value. If the default value is blank, no button is selected. If you have questions about default values, consult your project administrator.

- In the Multi-Choice dialog window:
 - To select the choices for the value, do one or both of the following:
 - Check the check box for each choice that you want to add, or click **Select All** to check all check boxes simultaneously.
 - Clear the check box for each choice that you want to remove, or click **Unselect All** to clear all check boxes simultaneously.

You can click **Set to Default** to check the check box for the default choice and clear any other checked choices. If the property has no default value, all check boxes are cleared. If you have questions about default values, consult your project administrator.

- Click **OK** or press the enter key.
- In the text field, enter the new value, and then press the enter key.

For a date or numeric property, the value must match the valid format shown in the tooltip.

For a text value, all keyboard characters are valid.



For a date property, you can click the calendar icon to display a calendar in a dialog window. To enter the new value from the calendar, do one of the following:

- Select the month, year, and day to enter that date.
-

Click **Today** to enter today's date automatically.

-

Click **TBD** to enter the value **TBD**, which marks the value for entry at a later date.

Click **OK** to close the calendar and apply the value.

To change an additional value, repeat steps 2 and 3.



When filtering is on and a particular filter value for a column is selected, the data are filtered according to the selected choice. Only objects matching the filter value are shown in the columns.

If another filter value is selected after the first filter is implemented, only the objects in the window are filtered with the new filter value. If the objects in the current view do not match the filter chosen, the row is not filtered and the objects could be removed from view until a new filter is used.

You cannot use more than one filter at one time. Filters can be applied as layers to further define the view. Restoring the objects to view removes all applied filters.

Procedure Notes

Step 1: You can also right-click the object and choose **Properties** from the pop-up menu. For a note or a diagram, you can display the dialog window also from the **Attachments** tab.

Step 3: To reverse this action, you can pull down the **Edit** menu and choose **Undo Change Properties**, click the **Undo** button on the toolbar, or press control-Z.

Property Editing for Multiple Object Selections

In the Edit Properties dialog window, you can change the values of editable properties for two or more selected objects. You can select the objects in the following hierarchical tables:

- The content table.
- The **Attachments, Links, Where Used, or Versions** tab or floating window.
- The **Complying Object Traceability** view.
- The Search Results dialog window.

The Edit Properties dialog window displays the editable properties that are currently shown in the table. Before you open the dialog window, you may want to set the columns for the properties that you want to see in the dialog window. For more information, see [Adding and Removing Columns](#), [Rearranging Columns](#), [Resizing Columns](#), and [Setting the Sort Column](#), earlier in this chapter.

Conditions of Property Display for Multiple Selections

When two or more objects are selected in a hierarchical table, certain properties that may be shown are subject to the following conditions:

- A property that has different values among the selected objects, such as the **Name** property, has a blank value in the dialog window.
- A property whose value is the same for all selected objects has that value in the dialog window.
- The **Subtype** property is displayed in the dialog window only if the selected objects are of the same base type.
- The **Number** property is not displayed in the dialog window.
- The **Text** property cannot be displayed if any one selected requirement has a **Text Format** property value of **MHTML**. This value indicates that the content is saved in Microsoft Word.



Ensure that the **Text Format** value is **Text** for each requirement whose **Text** property you intend to edit. You can set this value if necessary. For more information, see [Enabling Text Property Editing for a Requirement](#), in chapter 5, *Managing Requirements*.

Editing Properties for Multiple Objects



- You must have **Modify** permission for the objects.
- To edit properties for building blocks, TRAMs, or diagrams, you must have **Architect** privilege for the project.
- Your changes are applied to the current value of each selected object.
- If a change fails for any value, no values are changed in the database.

1.

Select the objects, pull down the **File** menu, and choose **Properties**.



You can select nonadjacent and adjoining objects of the same type or of mixed types.

Architect/Requirements displays the Edit Properties dialog window. Values that you cannot change are dimmed in the **Value** column.



The Edit Properties dialog window shows only those properties that apply to all selected object types.

2. In the **Value** column, double-click the value that you want to change.

If the property has a choice value, one of the following dialog windows is displayed:



In the Single-Choice dialog window, you can select only one choice.



In the Multi-Choice dialog window, you can select any combination of choices.



Each choice list may be set dynamically, and you may see different choices from time to time. If you have questions about a choice list, consult your project administrator.

A text field opens if the property has a date, numeric, or text value. For a date value, a calendar icon is displayed to the right of the text field.



- For date and numeric values, you can rest the pointer on the text field to see the valid format in a tooltip.
- For the system-defined property named **Text**, dimmed text in a cell indicates that editing is disabled because MHTML content is saved in Microsoft Word. You can enable this cell by changing the requirement's **Text Format** property value to **Text**, and then edit the **Text** property. For more information, see [Enabling Text Property Editing for a Requirement](#) in chapter 5, *Managing Requirements*.

3. Do one of the following:

- In the Single-Choice dialog window, select the new value in the value pane, and then click **OK** or press the enter key.

You can click **Set to Default** to select the default value. This button also selects the **Set** button in the **Action** group.



If the property has no default value, all values are cleared. If you have questions about default values, consult your project administrator.

- In the Multi-Choice dialog window:

- Do one of the following:

- To add choices to the current values, click **Add**, and then check the check box for each choice.

All previous choices are retained.

- To change the current values to different choices, click **Set**, and then check the check box for each new choice.

All previous choices are cleared. The checked choices become the new value for each object.

- To remove choices from the current values, click **Remove**, and then check the check box for each choice.

All other previous choices are retained.

You can use the buttons in the **Selection** group as follows:

- Click **Select All** to check all check boxes.
- Click **Unselect All** to clear all check boxes.
- Click **Set to Default** to check the check box for the default choice and clear any other checked choices. This button also selects the **Set** button in the **Action** group.



If the property has no default value, all check boxes are cleared. If you have questions about default values, consult your project administrator.

-

Click **OK** or press the enter key.

The **Action** column in the Edit Properties dialog window displays your action from the Multi-Choice dialog window. The displayed action is retained if you change the value back to the original one in the database.

- In the text field, enter the new value, and then press the enter key.

For a date or numeric property, the value must match the valid format shown in the tooltip.

For a text value, all keyboard characters are valid.



For a date property, you can click the calendar icon to display a calendar in a dialog window. To enter the new value from the calendar, do one of the following:

- Select the month, year, and day to enter that date.



Click **Today** to enter today's date automatically.



Click **TBD** to enter the value **TBD**, which marks the value for entry at a later date.

Click **OK** to close the calendar and apply the value.

To change an additional value, repeat steps 2 and 3.

4. To close the Edit Properties dialog window and apply your changes in the database, click **OK** or press the enter key.

You can reverse this action by pulling down the **Edit** menu and choosing **Undo Change Properties**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Editing Properties in Table View Cells

In the following table views, you can change the values of editable properties directly in the cells:

- The hierarchical content table.
- The **Attachments** tab and window.
- The **Trace** subtab of the **Links** tab and window.
- The **Versions** tab and window.
- The Search Results dialog window.



For a value that exceeds the cell size, you can rest the pointer on the cell to see the entire value in a tooltip. Multiple lines are used if necessary.

You may want to add, remove, rearrange, resize, or sort columns. For more information, see [Adding and Removing Columns](#), [Rearranging Columns](#), [Resizing Columns](#), and [Setting the Sort Column](#), earlier in this chapter.



- You must have **Modify** permission for the objects.
- To edit properties for building blocks, TRAMs, or diagrams, you must have **Architect** privilege for the project.

To edit property values in table view cells:

1.

Click a cell containing a value that you want to change.

If the property has a choice value, one of the following dialog windows appears:

- In the Single-Choice dialog window, you can select only one choice.
- In the Multi-Choice dialog window, you can select any combination of choices.



The choice list may be set dynamically, and you may see different choices from time to time. If you have questions about the choice list, consult your project administrator.

A text field opens if the property has a date, numeric, or text value. For a date value, a calendar icon is displayed to the right of the text field.



- For date and numeric values, you can rest the pointer on the text field to see the valid format in a tooltip.
- For the system-defined property named **Text**, dimmed text in a cell indicates that editing is disabled because MHTML content is saved in Microsoft Word. You can enable this cell by changing the requirement's **Text Format** property value to **Text**, and then edit the **Text** property. For more information, see [Enabling Text Property Editing for a Requirement](#) in chapter 5, *Managing Requirements*.

2. Do one of the following:

- In the Single-Choice dialog window, click the button for the new value, and then click **OK** or press the enter key.

You can click **Set to Default** to select the button for the default value. If the default value is blank, no button is selected. If you have questions about default values, consult your project administrator.

- In the Multi-Choice dialog window:
 - To select the choices for the value, do one or both of the following:
 - Check the check box for each choice that you want to add, or click **Select All** to check all check boxes simultaneously.
 - Clear the check box for each choice that you want to remove, or click **Unselect All** to clear all check boxes simultaneously.

You can click **Set to Default** to check the check box for the default choice and clear any other checked choices. If the property has no default value, all check boxes are cleared. If you have questions about default values, consult your project administrator.

- Click **OK** or press the enter key.
- In the text field, enter the new value, and then press the enter key.

For a date or numeric property, the value must match the valid format shown in the tooltip.

For a text value, all keyboard characters are valid.



For a date property, you can click the calendar icon to display a calendar in a dialog window. To enter the new value from the calendar, do one of the following:

- Select the month, year, and day to enter that date.

-

Click **Today** to enter today's date automatically.

-

Click **TBD** to enter the value **TBD**, which marks the value for entry at a later date.

Click **OK** to close the calendar and apply the value.

To change an additional value, repeat steps 1 and 2.

You can reverse this action by pulling down the **Edit** menu and choosing **Undo Change Properties**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Calculating Numeric Property Values With Formulas

Your project administrator can define numeric properties to use formulas that compute values automatically. When such numeric properties are assigned to folders, requirements, building blocks, or groups, you can use the formulas to update the values for selected parent objects.

In a given numeric property, the formula may be **Average**, **Maximum**, **Minimum**, **Multiply**, or **Sum**. The formula operates on that property's value for each direct child of the selected parent, and the corresponding cell for the parent is updated with the calculated value.

For example, assume that a numeric property named **Cost** uses the **Sum** formula and applies to requirements. Also, requirement *A* has two direct children: requirement *B*, whose **Cost** value is *10*; and requirement *C*, whose **Cost** value is *20*. When the **Cost** value is calculated for requirement *A*, that cell displays *30*, the total cost of the two direct children.



Calculations apply only to objects at the next level below a selected object. If a selected object has no children, its values are not updated.

You can select the parent objects in the hierarchical content table or the **Links** or **Versions** tabs or window. You may want to add, remove, rearrange, resize, or sort columns. For more information, see [Adding and Removing Columns](#), [Rearranging Columns](#), [Resizing Columns](#), and [Setting the Sort Column](#), earlier in this chapter. For information about modifying a numeric property definition to use a formula, see the *Systems Architect/Requirements Management Project Administrator's Manual*. If you have questions about numeric properties, consult your project administrator.



- You must have **Modify** permission for the objects.
- To calculate properties for building blocks or TRAMs, you must have **Architect** privilege for the project.

To calculate numeric property values with formulas:

1. Select the parent objects, pull down the **Tools** menu, and choose **Calculate Properties**.



A shaded cell with the number in blue indicates that the value is not a formula result but was entered manually. If you select the corresponding object, the formula does not operate on the children and no calculation is made. The cell must be blank before the value can be calculated.

The Calculate Properties dialog window displays each numeric property that contains a formula and applies to a selected object type.

2. Check the check box for each property that you want to update, and then click **Calculate**.

In the column for each property, the values for all direct children are computed according to the formula. The result is shown in the corresponding cell for each selected parent.

Procedure Notes

Step 1: You can select nonadjacent and adjoining objects of the same type or of mixed types.

Step 2: You can reverse this action by pulling down the **Edit** menu and choosing **Undo Change Properties**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Updating Properties From an AP 233 STP File

To edit properties automatically, you can import property values from an AP 233 data file (.stp) that contains records for objects in the database. A .stp file contains data in the format defined by AP 233, the STEP⁴ application protocol for systems engineering data representation. In Architect/Requirements, the data represents folders and requirements, including properties and values, that were previously exported to the file. Any property that can be exported to an AP 233 file can be updated from the same file. For more information, see [Exporting Objects to an AP233 STP File](#) in chapter 4, *Maintaining a Project*.

You match file records with database objects by specifying a property named in the file as a unique identifier, a key that Architect/Requirements uses to locate the objects in the database. For all other properties in the file except the key, values are overwritten in the database for each object in the file. Updates are added to each object's change log. For more information, see [Viewing a Change Log](#) in chapter 12, *Using the Change Management Package*.



- You must have **Modify** permission for the objects.
- To update properties for building blocks, TRAMs, or diagrams, you must have **Architect** privilege for the project.

To update properties from an AP 233 STP file:

1. Do one of the following:
 - For a project's primary level, select the project node in the navigation tree.
 - For a folder's top level, select the folder in the navigation tree or in the hierarchical content table.
2.
Pull down the **File** menu and choose **Import**→**AP233 (.stp)**.
The Open dialog window is displayed.
3.
Select the data file and click **Open**.
The Import AP233 dialog window is displayed.
4. Select **Update Existing Objects**, and then enter the key in the **Index Property** field.



- The property value must be unique for each object. If duplicate values are found, processing stops and no properties are updated.
- The property name is case sensitive and the property must be recorded in the file.

⁴ Standard for the Exchange of Product Model Data (ISO 10303).

In the **Index Property** field, you can enter any user-defined property name or one of the following system-defined property names:

- **Name**
- **Text**
- **ROIN**
- **Number**
- **Source Paragraph**
- **Source Filename**
- **Baseline**
- **Version Type**
- **Security Profile**
- **Subtype**

5. Click **OK**, or press the enter key.



This action clears the queue for the **Undo** option and cannot be reversed.

The Import AP233 dialog window closes, and a message is displayed to indicate that the import is in progress. When the import is complete, a confirmation message shows the number of objects for which the property is updated.

Using the Live Excel Interface

Through the Architect/Requirements live Excel interface, you can use Microsoft Excel to edit properties for selected objects. Live Excel generates an Excel file in which you can change editable property values for objects in the following views:

- The hierarchical content table.
- The **Links** and **Versions** tabs and floating windows.
- The **Defining Object Traceability** and **Complying Object Traceability** windows.
- The Search Results dialog window.



- You must have **Modify** permission for the objects.
- To edit properties for building blocks, TRAMs, or diagrams, you must have **Architect** privilege for the project.
- Microsoft Office Excel 2013 or Excel 2016 and the Microsoft .NET Framework 4.0 CLR must be installed on your computer.

Types of Live Excel Sessions

You can use live Excel while the Architect/Requirements client is running and connected to the database. You can also open an existing live Excel file in an independent live Excel session, without running the client but with the option to connect to the database.

Live Excel Sessions With the Architect/Requirements Client

With the Architect/Requirements client running, you can create a new live Excel file that contains a row and column for each object and property that you export from the active view. Or, you can open an existing live Excel file from a previous session.

Preparing Architect/Requirements Property Columns for Live Excel Export

Before starting live Excel, you may want to display or hide the children of parent objects, add columns to display other properties, or remove displayed property columns. For more information, see [Viewing Objects in a Hierarchy](#) in chapter 3, *Using the Architect/Requirements Main Window*, and [Adding and Removing Columns](#), earlier in this chapter.

Object and Property Selection for Live Excel Export

For a new file, you specify the objects and properties that you want to export. You can export only the currently selected objects, or you can export all displayed objects whether or not they are selected.

In addition, you can use an Excel template to specify the properties and other information to export, or you can export only the properties that are currently displayed in the view. For more information about Excel templates, see the *Systems Architect/Requirements Management Project Administrator's Manual*. If you have questions about Excel templates, consult your project administrator.

Live Excel Synchronization

During a live Excel session, the worksheet is synchronized with the Architect/Requirements database and client. Editable values that you change in the worksheet are immediately applied to the database, and then are displayed in the client. Values that you or other users change in the client are updated automatically in the worksheet.

Independent Live Excel Sessions

You can use an existing live Excel file to change editable property values without running the Architect/Requirements client. You can also start the client from an existing live Excel file.

Database Connection Options for Independent Live Excel Sessions

When you open an existing live Excel file independent of the Architect/Requirements client, you have the following options:

- Connect to the Architect/Requirements server. Changes in the live Excel file are immediately applied to the corresponding properties in the Architect/Requirements database.
- Remain disconnected from the Architect/Requirements server. Changes in the live Excel file are accumulated until pending changes are applied to the corresponding properties in the Architect/Requirements database.

Starting Architect/Requirements From an Independent Live Excel Session

You can start Architect/Requirements from a live Excel file by doing the following:

1. In the open file, select an object type indicator.
A login page is displayed.
2. Enter your user name and password.
The main window displays the object highlighted in the content table.

For more information, see [Starting a New Architect/Requirements Session](#) in chapter 3, *Using the Architect/Requirements Main Window*.

User Actions in Live Excel Files

You can copy or move a value to any other cell, and you can create Excel formulas to edit values automatically. With Excel's AutoFilter feature, you can hide and show rows according to criteria for any property column. And you can navigate to objects in the view by selecting object type indicators in the file.



- Some actions clear the queue for Excel's **Undo** option and cannot be reversed. Subject to those conditions, you can reverse an action by clicking **Undo** button on Excel's title bar, or by pressing control-Z.
- Some changes that you can make in the file do not affect the database or the view. For example, inserting or deleting a row in a worksheet does not create or delete an object in the database. Nor does inserting, deleting, rearranging, or resizing columns in a worksheet change the property columns displayed in the view.

Considerations for Sorting Data

If you sort the data in a live Excel file, the object type indicators and the objects may be mismatched. This condition occurs in Microsoft Office Excel when a cell contains a graphic that is taller than the row height. For a height that accommodates the indicators, increase the size of Excel's standard font.

The effective size depends on the standard font that you use. With Arial, for example, a size of 11 points allows the indicators to be sorted with the correct objects. To increase the size, click Excel's **Office Button**, click **Excel Options**, and select the new size in the **Popular**→**When creating new workbooks** section.



You must exit and restart Excel for this change to take effect. If you want to change the font size, do so before creating or opening a live Excel file.

Microsoft Office Excel Comment Indicators

If Excel's comment indicators are enabled when you select a cell, an indicator marks the cell if it contains a property value. The comment shows the property name and whether the property is editable. If the property has a choice value, the comment also shows the choices.



Values that you cannot change are dimmed in the cells.

- To enable or disable indicators, click Excel's **Office Button**, click **Advanced**, and select the related option from the **Display** section in the right pane.
- To view a comment, rest the pointer on the cell.

Editing Properties in a Live Excel Session With the Architect/Requirements Client

1. With the view activated, do one of the following:



To create a new file:

.

Pull down the **File** menu and choose the **Export**→**Excel Spreadsheet** options to display the Export To Excel dialog window.

You can also right-click inside the view and choose the **Export**→**Excel Spreadsheet** options from the pop-up menu.

.

Under **Object Selection**, do one of the following:

- o To export only the selected objects, click **Export Selected Objects**.
- o To export all objects in the view, click **Export All Objects in View**.

.

Under **Formatting**, do one of the following:

- o To export the properties in the currently displayed columns, click **Use Current View Columns**.
- o To export the properties specified in an Excel template, click **Use Excel Template**.

The template selection field becomes available. You can accept the default Excel template or select a different template from the list.



- o A template may contain multiple worksheets, each worksheet with different properties. Therefore, data may be separated on worksheets in the live Excel file.
- o If properties in the template do not apply to an object type specified for export, the inapplicable property values are indicated in the related cells.

For more information about Excel templates, see the *Systems Architect/Requirements Management Project Administrator's Manual*. If you have questions about templates, consult your project administrator.

.

Under **Output**, click **Excel Live**, and then click **OK**.



If your live Office and Architect/Requirements versions do not match, a message asks you to reinstall live Office. Click **OK** to start the uninstaller and follow the instructions. Click **Finish** to start the installer, follow the instructions, and click **Finish**. A message may ask you to restart your computer to complete the installation. After installation, repeat steps a. through d.

Live Excel opens a read-only **.html** file. Values that you cannot change in the database are dimmed in the worksheets. Go to step [2](#).

- To open an existing file, pull down the **File** menu and choose **Excel Live Open**.

You can also right-click inside the view and choose the **Excel Live**→**Open** options from the pop-up menu.

Live Excel starts with Excel's Open dialog window. When you select the file and click **Open**, one of the following is displayed:

- . A message stating that there are no pending changes in the file.

This file does not contain a new value for any property in the database. However, the database may contain new values that are not reflected in the file. When you click **OK**, all worksheets in the file are updated with changes to the database since the file was last saved. Go to step 2.

- . A message asking if you want to check in pending changes before refreshing the file from the Architect/Requirements server.

This file contains new values that were entered without a connection to the server, and these changes have not been applied to the properties in the database. For other properties, the database may contain new values that are not reflected on any worksheet in the file.

- o To apply the pending changes to the database, click **Yes**.

First, pending changes are applied to the database and displayed in the main window. Second, all worksheets in the file are updated with other changes to the database since the file was last saved.



During this process, other users may edit the same properties for which the file contains pending changes. Therefore, those values also may be updated in the file when the process is complete.

- o To discard the pending changes, click **No**.

All worksheets in the file are updated with changes to the database since the file was last saved.

2. In the live Excel file, do any or all of the following:



Values that you cannot change in the database are dimmed in the worksheets.



To change a value:

- Double-click the cell that contains the value.



Do not change the value in the Architect/Requirements client while the cell is in edit mode. Otherwise, live Excel displays an error message stating that an edit is in progress.

A pop-up list appears if the property has a choice value.



The choice list may be set dynamically, and you may see different choices from time to time. If you have questions about the choice list, consult your project administrator.

A text field opens if the property has a date, numeric, or text value.



For the system-defined property named **Text**, the value cannot be changed in the Architect/Requirements database for a requirement with MHTML content saved in Microsoft Word. Although you can edit the cell in the live Excel file, when you click outside the cell, an error message states that you cannot overwrite rich text.

You can enable this cell by changing the requirement's **Text Format** property to **Text**, and then edit the **Text** property. For more information, see [Enabling Text Property Editing for a Requirement](#) in chapter 5, *Managing Requirements*.

- Do one of the following:

- o In the pop-up list, select the choice or choices for the new value, and then click another cell to close the list.

Buttons indicate a single-choice list. You can select only one choice.

Checkboxes indicate a multiple-choice list, in which you can select any combination of choices. Selecting an unchecked choice adds it to the value, and selecting a checked choice removes it from the value.

- o In the text field, enter the new value, and then press the enter key.

For a date or numeric property, the value must match the valid format for the property.

For a text value, all keyboard characters are valid.

You can also select the cell and enter the new value directly.



If you create an Excel formula, you can automatically update values in all dependent cells when you change the value in a precedent cell. For more information, see [Microsoft Office Excel Help](#).

- To copy or move a value:
 - . Select the source cell, and choose **Copy** or **Cut** on Excel's **Home** ribbon.
 - . Select the destination cell, and then click **Paste** on Excel's **Home** ribbon.



If you cut the value and paste it into a different row, the value remains with the source object in the database. Therefore, the value is not moved in the client view.

- To hide or show rows, click the button in the heading of a property column, and then select a filtering criterion from the pop-up list.



To open a requirement in Word for editing or viewing:

- . Select any data cell except **Home** in the row for the requirement.
- . Do one of the following:
 - In Excel, on the **Add-Ins** ribbon:

- For editing, click the **Open** button.

When you save the Word file, your edits are applied to the requirement's **Text** property value.



If the live Excel file contains the **Text** property, click **Refresh** on the **Add-Ins** ribbon to update the value. This action also updates the value in Architect/Requirements. Conversely, clicking **Refresh** on the Architect/Requirements toolbar also updates the value in live Excel.

- For viewing, click the **Open Read-only** button.

Although you can edit this file, the changes cannot be saved in live Excel or in Architect/Requirements.



To navigate to an object in the main window view, do one of the following in the **Home** column of the worksheet:

- . Click the blank space in the cell containing the object type indicator.
- . Use the arrow keys to select the cell containing the object type indicator.

The object is displayed and highlighted in the view.



You can update the file with the latest data from the Architect/Requirements database by clicking the **Refresh** button on the **Add-Ins** ribbon.

3. To end the live Excel session, click the Excel's **Office Button** and choose **Exit Excel**.

Live Excel displays a message asking if you want to save the changes.

- If you click **No**, your changes are applied in the database but a permanent file is not created.

The temporary file remains on your computer until you exit Architect/Requirements. Then the file is deleted from your computer.

- If you click **Yes**, Excel displays the Save As dialog window.

For this permanent file, assign the file name, file type, and location outside the Architect/Requirements database, for example, on a local drive.



- . For a live Excel file, Siemens PLM Software recommends that you assign the **.xlsm** file type. Only the **.xlsm** file type supports the full capability of live Excel.
- . If you save a live Excel file as the **.mhtml** file type, it becomes static and loses the live capability permanently.

Editing Properties in an Independent Live Excel Session

1. With the live Excel file open, initiate the live Excel session by selecting any cell on any worksheet.



If your live Office version does not match your Architect/Requirements version, a message asks if you want to reinstall live Office. Click **OK** to start the uninstaller and follow the instructions. Click **Finish** to start the installer, follow the instructions, and click **Finish**. A message may ask you to restart your computer to complete the installation. After installation, repeat this step.

A message asks if you want to connect to the Architect/Requirements server.

2. Do one of the following:

- To log in to the server for immediate database changes, click **Yes**.

Microsoft Internet Explorer displays the Teamcenter systems engineering login page. Enter your Architect/Requirements user name and password in the **User Name** and **Password** fields, and then click **Log In**.

Live Excel displays one of the following:

- o A message stating that there are no pending changes in the workbook.
This worksheet does not contain a new value for any property in the database. However, the database may contain new values that are not reflected in the worksheet.
When you click **OK**, a progress indicator appears, and the worksheet is updated with changes to the database since the file was last saved.
- o A message asking if you want to check in pending changes before refreshing the worksheet from the Architect/Requirements server.

This worksheet contains new values that were entered without a connection to the server, and these changes have not been applied to the corresponding properties in the database. For other properties, the database may contain new values that are not reflected in the worksheet.

- To apply the pending changes to the database, click **Yes**.

A progress indicator appears during this process, which has two parts. First, pending changes are applied to the database. Second, the worksheet is updated with other changes to the database since the file was last saved.



During this process, other users may edit the same property values for which the worksheet contains pending changes. Therefore, those values also may be updated in the worksheet when the process is complete.

- To discard the pending changes, click **No**.

A progress indicator appears, and the worksheet is updated with changes to the database since the file was last saved.

- To remain disconnected from the server and accumulate changes, click **No**.

Live Excel displays a message stating that you are working offline. Changes are accumulated and can be applied to the database by logging in later. Values that you cannot change in the database are dimmed in the worksheet.

3. In the live Excel file, do any or all of the following:



Values that you cannot change in the database are dimmed in the worksheets.



To change a value:

- Double-click the cell that contains the value.

A pop-up list appears if the property has a choice value.



The choice list may be set dynamically, and you may see different choices from time to time. If you have questions about the choice list, consult your project administrator.

A text field opens if the property has a date, numeric, or text value.



For the system-defined property named **Text**, the value cannot be changed in the Architect/Requirements database for a requirement with MHTML content saved in Microsoft Word. Although you can edit the cell in the live Excel file, when you click outside the cell, an error message states that you cannot overwrite rich text.

You can enable this cell in this live Excel session if the file contains the **Text Format** property and is connected to the Architect/Requirements server. For more information, see [Enabling Text Property Editing for a Requirement](#) in chapter 5, *Managing Requirements*.

If the file is disconnected from the Architect/Requirements server, cell editing cannot be enabled in this live Excel session.

- Do one of the following:

- In the pop-up list, select the choice or choices for the new value, and then click another cell to close the list.

Buttons indicate a single-choice list, in which you can select only one choice.

Checkboxes indicate a multiple-choice list, in which you can select any combination of choices. Selecting an unchecked choice adds it to the value, and selecting a checked choice removes it from the value.

- In the text field, enter the new value, and then press the enter key.

For a date or numeric property, the value must match the valid format for the property.

For a text value, all keyboard characters are valid.

You can also select the cell and enter the new value directly.



If you create an Excel formula, you can automatically update values in all dependent cells when you change the value in a precedent cell. For more information, see Microsoft Excel Help.

- To copy or move a value:
 - Select the source cell, and choose **Copy** or **Cut** on Excel's **Home** ribbon.
 - Select the destination cell, and then click **Paste** on Excel's **Home** ribbon.



If you cut the value and paste it into a different row, the value remains with the source object in the database. Therefore, the value is not moved in the view, although the value is moved to the destination cell in Excel.

- To hide or show rows, click the button in the heading of a property column, and then select a filtering criterion from the pop-up list.
4. To end the live Excel session, click the Excel's **Office Button** and choose **Exit Excel**.

Live Excel displays a message asking if you want to save the changes.

- Click **Yes** to display Excel's **Save As** dialog window.
- Assign the file name, file type, and location outside the Architect/Requirements database, for example, on a local drive.
- Click **Save** to close the file.



- For a live Excel file, Siemens PLM Software recommends that you assign the **.xlsm** file type. Only the **.xlsm** file type supports the full capability of live Excel.
- If you save a live Excel file as the **.mhtml** file type, it becomes static and loses the live capability permanently.

Creating Objects in Live Excel

In addition to editing properties, you can use live Excel to add new folders, requirements, building blocks, and groups to the Architect/Requirements database. For an existing object in a live Excel file, you create a sibling object that has the same object type and subtype.



- The live Excel file must be connected to the Architect/Requirements server. For more information, see [Types of Live Excel Sessions](#), earlier in this chapter.
 - To create building blocks, you must have **Architect** privilege for the project.
 - You cannot create notes, diagrams, or spreadsheet objects in live Excel.
 - Microsoft Excel 2013 or Excel 2016 must be installed. The Microsoft .NET Framework 4.0 CLR must also be installed. For more information, see [Prerequisites for Using the Live Office Interface](#) in chapter 2, *Installing the Architect/Requirements Client with Office Integration*.
- Right-click a cell for the existing sibling, and then choose the **Teamcenter for systems engineering**→**New Object** options from the pop-up menu.



- o Ensure that the entire row is not selected. Otherwise, a different pop-up menu is displayed.
- o You can right-click any cell in the row within the table of objects and properties. You cannot select the options if you right-click a cell outside that table.

In the live Excel file, the new object is added on a new row directly below the sibling. In the client, the new object is placed at the level of the sibling according to the current sort sequence.



If the sibling has a property set by an activator, the new object's value is blank in the corresponding cell. If you have questions about activators, consult your project administrator. For more information about activators, see the *Systems Architect/Requirements Management API Reference* manual.

Repeat this procedure for each additional object that you want to create.

Creating Trace Links in Live Excel

You can use live Excel to create trace links between objects in Architect/Requirements. For more information, see chapter 7, [Showing Object Relationships With Trace Links](#).

For defining and complying objects, you can do the following:

- Start the trace links in the live Excel file and end the trace links in the Architect/Requirements client.
- Start the trace links in the Architect/Requirements client and end the trace links in the live Excel file.
- Start and end the trace links in the live Excel file.

The live Excel file must be connected to the Architect/Requirements server. For more information, see [Types of Live Excel Sessions](#), earlier in this chapter.



Microsoft Excel 2013 or Excel 2016 must be installed. The Microsoft .NET Framework 4.0 CLR must also be installed. For more information, see [Prerequisites for Using the Live Office Interface](#) in chapter 2, *Installing the Architect/Requirements Client with Office Integration*.

1. Select the defining object or objects, right-click the selection, and choose the **Teamcenter for systems engineering**→**Start Link** options from the pop-up menu.



- Ensure that the entire row is not selected. Otherwise, a different pop-up menu is displayed.
- If you select only one defining object, you can select multiple complying objects. If you select multiple defining objects, you can select only one complying object.

2. Select the complying object or objects, and then right-click the selection and do one of the following:
 - For the default trace link subtype, choose the **Teamcenter for systems engineering**→**End Link** options from the pop-up menu.
 - For a user-defined trace link subtype:
 - Choose the **Teamcenter for systems engineering**→**End Link as Subtype** options from the pop-up menu to display the Select Subtype dialog window.
 - Select a trace link subtype, and then click **OK** to close the dialog window.

Disconnecting a Live Excel Spreadsheet

Any live Excel spreadsheet can be permanently disconnected from the Architect/Requirements database. When disconnected, the spreadsheet is made static and cannot be used to edit object properties interactively.



Microsoft Excel 2013 or Excel 2016 must be installed.

The Microsoft .NET Framework 4.0 CLR must also be installed on your computer.

1. Do one of the following:

- In Microsoft Office Excel, click the **Non-Live** button on the **Add-Ins** ribbon.

A confirmation message states that the spreadsheet cannot be converted to a live spreadsheet again, and asks if you want to continue.

2. Click **Yes**.



This action is permanent and cannot be reversed.

The spreadsheet is disconnected, and the current objects and properties remain in the static file.

Attaching a Spreadsheet to an Object

To store equations in the Architect/Requirements database, you can create spreadsheet objects containing worksheets from existing Microsoft Excel files. Each spreadsheet can store equations so that they are not overwritten by updated property values when the spreadsheet is opened.

For example, equations may be entered in property cells in a file containing object data previously exported from Architect/Requirements. If a spreadsheet is created from a live Excel file, equations entered in property columns are automatically preserved when the spreadsheet is opened and the properties are refreshed. A spreadsheet can also be created from an Excel file that does not contain equations or object data. For more information, see [Using the Live Excel Interface](#), earlier in this chapter, and [Exporting Objects to Microsoft Office Excel](#) in chapter 4, *Maintaining a Project*.

You select the spreadsheet owner in the hierarchical content table. For that object, the new spreadsheet is displayed in the **Attachments** tab or window. The spreadsheet remains in the database and the equations can be edited in Excel until the spreadsheet object or its owner is deleted. In addition, the spreadsheet object can be copied or moved to a different owner. For more information, see [Attachments Tab](#) or [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*; and [Copying Objects](#), [Moving Objects](#), or [Deleting Objects](#) in chapter 4, *Maintaining a Project*.

To attach a spreadsheet to an object:

1. Select the object, and then pull down the **File** menu and choose the **New→Spreadsheet** options.

Architect/Requirements displays the Open dialog window, which lists existing folders and files in the current drive or folder. If the list does not display the spreadsheet, you can use the **Look in** field to change the drive or folder, or use the **Files of type** field to see other types of files.
2. In the list, select the spreadsheet that you want to use for the new spreadsheet object, and then click **Open**.

A message is displayed while the spreadsheet is created. Then the **Attachments** tab or window displays the new spreadsheet object, with a default name in an open text field.
3. Enter the spreadsheet name in the text field, and then press the enter key.

Procedure Notes

Step 1: You can also right-click the object and choose the **New→Spreadsheet** options from the pop-up menu.

Storing Property Values Through Reference Links

A reference link is a placeholder for the current value of a property. You can create reference links in the Microsoft Word content of requirements, notes, change logs, and change approval objects. Each time a containing object is opened in Word, reference link values are updated automatically. Thus, a single property of a target object can be referenced by many other objects.



Changes to either a containing object or the target object do not trigger change events on the other end of the reference link.

Reference Link Concepts

This section provides conceptual information about reference links.

Reference Link Types

You can create the following types of reference links:

- Plain text, consisting solely of a text string.

This type of reference link targets a plain text property. Plain text properties apply to all object types and can be displayed in all views in the Systems Engineering and Requirements Management module.

- Full content, which can consist of HTML and embedded elements such as tables, graphics, and OLE objects.

This type of reference link targets the full content of a requirement, a note, a change log, a change approval object, or a live Visio diagram. Only these object types can be targeted by full content reference links.



Right-click a live Visio diagram and select **Copy Snapshot**. This copies the diagram's snapshot image, which can then be pasted into a requirement in Microsoft Word.

The full content is referenced through a note that is automatically attached to the diagram object when it is saved. This note contains an image (.gif) of the diagram.

The diagram image note must be selected as the target, not the diagram object itself.

For more information, see [Creating a Diagram](#) in chapter 6, *Constructing System Views With Building Blocks and Diagrams*.

Reference Link Target Objects

The target object can be selected in the following:

- The hierarchical content table.
- The **Attachments, Links, Connectivity, Where Used, or Versions** tab or window.
- The Search Results dialog window.
- The **Complying Object Traceability** window or the **Defining Object Traceability** window.

If a shortcut is selected as the target object, the reference link relation is created to the selected shortcut and not to its primary object. If the shortcut has its own unique value for the referenced property, the value of the shortcut appears when the reference is resolved into the referencing object's text. However, if the shortcut does not have its own unique value, the information from the primary object is used when the reference is resolved. In either case, the reference relation appears in the **Where Used** and **Links/Relations** tabs for the shortcut object and not for the primary object. The link is broken if the shortcut is deleted, even if the primary object exists.

Behavior of Reference Links

In Architect/Requirements, objects that contain rich text, requirements, and notes can dynamically include information from other objects. The referenced information can be either a property value or the text content of a requirement or note. When a reference targets a versionable object, Architect/Requirements determines which version of the information to use. This section describes the behavior of these reference links.

Fixed and Dynamic Version References

A reference to a versionable object can be either fixed or dynamic. Fixed references always get the information from the same version of the object. Dynamic references get the information from the currently effective version of the object.

There are two ways to specify how dynamic links behave. The **Project Settings** property of a project controls the default behavior for reference links in that project. Selecting the **Promote References** choice indicates dynamic references. By default, a project uses fixed references.

The behavior for a specific link can be controlled with the **Reference Settings** property (which can be accessed by right-clicking a reference link). Double-clicking the **Reference Settings** property allows you to select or deselect the **Promote References** value. The **Reference Settings** property allows a specific reference link to override the default behavior.



Although the **Reference Settings** property is a multi-select property, the **Promote References** and **Fixed References** options are mutually exclusive. If you select both the options, **Promote References** takes precedence.

The use of fixed or dynamic references does not affect how the references are stored in the database. The setting for a project can be changed at any time to change the behavior of the existing references.

The **Reference Settings** property requires **Read & Write** privileges to the objects for which the reference link is created. When nothing is selected in **Reference Settings**, the project's default setting is used. Choosing **Promote References** or **Fixed References** overrides the project setting for the reference link.



The **Promote References** behavior is supported for references to versionable objects, requirements, buildings blocks, and shortcuts to versionable objects. References to notes do not support the **Promote References** behavior.

Reference Promotion and Demotion

When fixed references are used and a new version of a referenced object is created, Architect/Requirements decides which version to use for the reference based on the state of the referencing object. If the referencing object is baselined or frozen, then the reference remains

with the old version. This behavior prevents modifications to the new version from indirectly modifying the content of a frozen or baselined object.

If the referencing object is not frozen, then the reference is promoted to the new version. This behavior ensures that the non-frozen objects reference the latest information.

When the current non-frozen version of an object is deleted, any references to it are shifted back to the prior frozen version. When dynamic references are used, the reference shows information from the currently effective version of the referenced objects. The promotion and demotion of the reference is dynamic. Even if the referencing object is frozen or baselined, the content can appear to change. This does not compromise the content of a baseline. Baselined content is preserved but the baseline effectivity must be set to display it.

Indirect Changes to Text Content

Modifying a referenced object indirectly causes a change to the text content of the referencing object because reference links dynamically access information from the referenced object. This behavior may not be desirable if the referencing object is frozen or baselined.

Architect/Requirements allows references from frozen to non-frozen objects to exist. A warning is displayed when an object is frozen that has a reference to a non-frozen object. You can take necessary steps to avoid this situation. If needed, activators can be used to prevent indirect changes to text content.

Broken References

In cases where the referenced information cannot be retrieved, the **[Broken Link: ...]** string is inserted in the text in place of the reference and the previously referenced information is included as plain text. References are broken when the referenced object is deleted and there is no prior version to use instead. With dynamic references, a reference appears broken when there is no version of the referenced object that appears in the current effectivity. In this case, a different broken link indicator is used, such as **[no effective version for <name of referenced object>: ...]**.

Illustration

Here is an illustration of how a reference behaves in a particular scenario. For the illustration:

- . Create two requirements, Req1 and Req2-A.
- . Edit the text of Req1 and include a reference to the name of Req2-A.
- . Freeze Req2-A, create a new version from Req2-A and name it as Req2-B.
- . Create a baseline that includes Req1 and Req2-B.
- . Create a new version of Req2-B named Req2-C.
- . Freeze Req2-C and create a version named Req2-D.

If fixed references are used, the content of Req1 displays the name of the baselined version of Req2 and Req2-B regardless of the effectivity setting. The reference was promoted to Req2-B when Req2-A was versioned because Req1 was not frozen at that time. After Req1 was baselined, references are no longer promoted; the reference remains at Req2-B.

If dynamic references are used, then the content of Req1 changes based on effectivity. Req2-B is displayed in the baseline effectivity, Req2-C in current frozen version, and Req2-D in current version. Req2-A could be displayed in a frozen as of date effectivity.

When the **Promote References** option is not enabled, effectivity has no impact on reference links. References are to a specific version. Effectivity will control which objects you will see in the user

interface: A, A-1, B, B-1, B-2, or C. However, for any object that appears, its content that is referenced from other objects comes from a specific version of that object, regardless of the effectivity.

For example, frozen A-1 references frozen B-1, and C references the non-frozen B-2:

- A-1's content will always show information pulled from B-1, even if **Effectivity** is **Current Version** and it is B-2 that currently appears in the client.
- Likewise, C's content will always show information pulled from B-2, even if **Effectivity** is **Current Frozen Version** and it is B-1 that currently appears in the client.

When the **Promote References** property setting is enabled, reference links are sensitive to effectivity. For example, consider a frozen object A referencing a non-frozen object B. B1 is created as the new version of B. Enabling the **Promote References** option resolves the reference link from A to B in **Current Frozen Version** effectivity and to B1 in **Current Version** effectivity.

The above example (with objects A, B, and B1) is used to illustrate the similarities and differences in the resolution of reference links with and without enabling the **Promote References** option. The behavior can be seen in the **Links→Relations** and the **Where Used** tabs in the notebook pane.

Table 9-2 lists the resolution of the reference links with the **Promote References** option disabled. Table 9-3 lists the resolution of the reference links with the **Promote References** option enabled.

Table 9-2. Reference Links with Promote References Disabled

Tab in the Notebook pane	Effectivity: Current Version	Effectivity: Current Frozen Version (same behavior)
Links→Relations	Selecting A displays B as the target. Selecting B1 displays no reference.	Selecting A displays B as the target. Selecting B displays A as the source.
Where Used	Selecting B1 displays no reference.	Selecting B displays A.

Table 9-3. Reference Links with Promote References Enabled

Tab in the Notebook pane	Effectivity: Current Version	Effectivity: Current Frozen Version (same behavior)
Links→Relations	Selecting A displays B1 as the target. Selecting B1 displays A as the source.	Selecting A displays B as the target. Selecting B displays A as the source.
Where Used	Selecting B1 displays no reference.	Selecting B displays A.

Copying Objects That Contain Reference Links

When copying objects that contain reference links:

- If the reference link is not located within the copy region (but is located within the same project), the new reference link points to the same target it originally pointed to.
- If the reference link is located within the copy region, the new reference link points to the new object created inside the copy region.
- If the reference link is located in another project and is not within the copy region, the new reference link is disabled and is represented by the label [**Broken Link: contents**].

The *contents* variable represents the last actual value of the property before the reference link was disabled.

Modifying Objects That Contain Reference Links

If object *A* contains reference links to object *B*:

- Reference links between *A* and *B* cause no special restriction on modifying or deleting either one.

If more access control is desired, security profiles can be applied to the objects.

For more information about access control and security profiles, see the *Systems Architect/Requirements Management Project Administrator's Manual*.

- If referenced properties of *B* are changed, the reference link values in *A* are indirectly changed, even without the **Modify** permission for *A*.

The reference link values are updated automatically the next time *A* is opened in Microsoft Word. If *A* is currently open, the reference link values are not updated until the Word file is closed and then reopened.

All the above is true even if *A* is frozen.

Deleting Objects That Contain Reference Links

If object *A* contains reference links to object *B*:

- If *A* is deleted, the reference links to *B* are also deleted. The **Modify** permission for *B* is not necessary.
- If *B* is deleted, the reference link values in *A* are preceded by the label **[Broken Link]**, and the values are resolved as follows:
 - Plain text property references include the old value. For example:

[Broken Link: 1234-5]



If object *A* contains a reference link to a plain text property *X* of object *B*, the above behavior also applies if the type definition for *B* is changed so that property *X* is no longer used for that object type.

For information about modifying a type definition, see the *Systems Architect/Requirements Management Project Administrator's Manual*.

- For full content reference links, the old value may consist of complex elements such as tables, graphics, and OLE objects. This value may exceed the effective size for display. Therefore, the value is a URL that is an artifact of how the body text references are handled as subdocuments in Word. For example:

[Broken Link: url]

- If the deleted *B* is restored from Recycle Bin, the references in *A* are fully restored.



If you save a Word document containing broken links, the reference link values are not restored even if you restore the deleted object from your Recycle Bin.

- Once *B* is discarded from Recycle Bin, references are lost and only the **[Broken Link]** labels remain.

All the above is true even if *A* is frozen.

For more information, see [Deleting Reference Links](#), later in this chapter.

Freezing Objects That Contain Reference Links

If object A references information from object B, and neither is frozen:

- When you freeze A making it A-1, a warning message displays **References content from unfrozen object**.
- Freezing A does not impose any special access control or limitations on B.
- If B remains unfrozen, then A-1's content will change whenever the referenced content in B changes.
- It is up to the user whether and when to freeze B.

If objects A-1 and C reference information from object B-1, and A-1 and B-1 are both frozen, but C is not:

- When a new version B-2 is created for B, it continues to refer to the frozen B-1 because A-1 is frozen.
- When a new version B-2 is created for B, it now refers to the new version B-2 because C is not frozen.

Restating the above examples as a rule:

- References from a non-frozen object to other objects are always promoted to reference the latest versions of those other objects.
- References from a frozen object to other objects are never promoted to reference new versions of those other objects.

Exporting and Importing Objects That Contain Reference Links

This section provides information on importing and exporting objects that contain reference links to XML data files, to AP 233 data files, and to Microsoft Word.

XML Export/Import

If the reference link is located within the export region, the new reference link points to the new object created inside the export region.

If the reference link is located in another project and is not within the export region, the new reference link is disabled and is represented by the label **[Broken Link: contents]**. The *contents* variable represents the last actual value of the property before the reference link was disabled.

- For a plain text property, the value is the entire value.
- For a full content property, the value is only the plain text of the content and does not include elements such as tables, graphics, or OLE objects.

If both ends of the reference (A & B) are internal to the export region, a new reference is created at import between the newly created A' and B'.

If A is exported, but the referenced object (B) is external to the export region:

- Plain text property references appear in the imported A' as local text.
- Full content references will appear in the imported A' as an invalid URL artifact.

AP 233 Export/Import

- The AP-233 standard has no concept of reference links.
- References in object A's text are fully expanded when exported.
- References to full content are expanded using only the plain text property value. Therefore, markup, tables, graphics and OLE objects are lost from referenced content.
- At import, all of A's text is local, with no references. Therefore, all reference links are lost in an AP-233 export and import round trip.

Microsoft Word Export

- All referenced content is exported with fresh, current values.
-

For the folder containing the export documents, the document template has a new property named **Document Template Rules**.

- When this property value includes **Preserve References**, reference links to full content are exported as hyperlink URLs.
- Otherwise, reference links are expanded inline for a clean document.
- Reference links are never created during document import.

Access Control for Objects That Contain Reference Links

Create a reference in object *A* to content of *B*:

- You must have Modify permission for *A* where the reference is being created.
- Read access to *B* is adequate to create a reference to its content.

View object *A* that includes content from *B*:

- You must have Read rights to *A* to see that object at all.
- If you can view *A*, but do not have Read rights to *B*, then the token [No Access] appears in *A*'s text where the reference occurs.
- If you edit *A* when the [No Access] token is present:
 - If you leave the token in place when you save, the reference remains intact and other users with Read rights to both *A* and *B* still see the referenced information.
 - If you remove the token, the reference is deleted.

For more information about access control and security profiles, see the *Systems Architect/Requirements Management Project Administrator's Manual*.

Dynamic vs. Static Behavior

If object *A* references information from object *B*:

- When values in *B* change, there is no immediate update to the database content of *A*. The change log for *A* never reflects changes due to the referenced content from *B*.
- When *A* is opened for edit in Word, it always includes fresh content of references from *B*. When the edit is saved, the current values from *B* are included within *A*'s saved text.
- When you search, string matching against *A* will see those embedded *B* values last saved from editing *A*. Those values may or may not be the true current values in *B*.
- When *A*'s text is displayed in the Preview tab, it always includes fresh content for all references. This does not refresh the database content for *A*.
- For export to Word, fresh content is exported, and there is no database change.
- For **getValue** (Text, HTML, MHTML) calls, fresh content is exported, and there is no database change: A **getValue/setValue** round trip on MHTML refreshes the database content. This can be an expensive process when graphics and OLE objects are involved.

Creating a Plain Text Reference Link

1. Select the requirement, note, change log, or change approval object in which you want to create the reference link.
2. Select **File**→**Open** to open an **.mhtml** Word file.
3. Display the target object.
4. In the row for the target object, right-click the plain text property value and choose **Copy Reference Link** from the pop-up menu.



For a requirement, a note, a change log, or a change approval object, right-clicking in the leftmost column copies the object's full content rather than a plain text value. For an object of any other type, right-clicking in the leftmost column copies the object name.

You can also right-click the plain text property value in the **Properties** tab or window or the Edit Properties dialog window.

5. In the Word file, paste the reference link at the desired position in the content.



For plain text properties, you can paste reference links that point to the same object that is open in Word. For example, within the Word file for a given object, you can paste a reference link to the name of that same object.

The value is inserted as a hyperlink, which you can click to navigate to the object.

6. Click Microsoft Word's **Office Button** →**Save**.

If the property value is changed after you close the file, the reference link value remains the same until you reopen the object. At that time, the reference link value is updated automatically to match the property value.

Creating a Full Content Reference Link

1. Select the object to contain the link, pull down the **File** menu, and choose **Open**.
2. Display the target object.



For a target diagram, open and save the diagram if the following are true:

- It is from an Architect/Requirements version earlier than 2007.1.
- This is the first full content reference link created to the diagram.

Saving the diagram automatically generates a note containing the diagram image (.gif), through which the diagram's full content is referenced.

3. In the row for the target object, right-click the leftmost column, and then choose **Copy Reference Link** from the pop-up menu.

Depending on the view, the leftmost column is:

- The **Name** column in the content table, the **Versions** tab or window, the **Complying Object Traceability** window, the **Defining Object Traceability** window, and the Search Results dialog window.
- The **Defining Trace** or **Complying Trace** column in the **Trace** subtab.
- The object type indicator column in any other view.



- If you right-click in a property column, you copy a plain text value rather than the object's full content.
- For a live Visio diagram, select the diagram image note as the target, not the diagram object itself.

For more information, see [Creating a Diagram](#) in chapter 6, *Constructing System Views With Building Blocks and Diagrams*

You cannot copy full content from the **Properties** tab or window or the Edit Properties dialog window.

4. In the Word file, paste the reference link at the desired position in the content.



You cannot paste a reference link that points to the full content of the same object that is open in Word.

The current value is inserted. Initially, the reference link value displays only the plain text in the content and the URL of the object, with the entire value formatted as a hyperlink. The reference link resolves to the full content after you store it in the database, close this file, and reopen the object in Word.

5. Click Microsoft Word's **Office Button** → **Save**.

After you close this file and reopen the object:

- The reference link value is preceded by a down arrow and a right arrow. You can click the down arrow to open the target object for editing. You can click the right arrow to navigate to the target object.
- The reference link is displayed as a subdocument in Word's outline view.

If the content is changed after you close the file, the reference link value is updated automatically the next time you open the object.

Viewing Reference Links

In the **Preview** tab and window, you can view reference link values within the content of the requirement selected in the hierarchical content table. For requirements, notes, change logs, and change approval objects, the **Preview** window displays reference link values within the content of the object selected in the **Attachments**, **Links**, **Where Used**, or **Versions** tab. For more information, see [Hierarchical Content Table](#), [Attachments Tab](#), [Relations Subtab](#), [Where Used Tab](#), [Versions Tab](#), or [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.



- Content formatting is determined by the document template for the containing folder. To apply other formatting, you can change the **Document Template** property value for the folder.

For more information, see [Editing the Properties of a Selected Object](#), earlier in this chapter.



To view full content reference link values as hyperlink URLs, the document template's **Document Template Rules** property must include the **Preserve Links** value.

For information about modifying a document template, see the *Systems Architect/Requirements Management Project Administrator's Manual*.

- The reference link value displays the text **No Access** if you do not have at least **Read Only** access privilege to a referenced object.

In the **Relations** subtab of the **Links** tab or window, you can view the reference link objects:

- For the object selected in the hierarchical content table, the subtab shows reference links to the object, reference links from the object, and the starting or ending object for each reference link.
- In the **Links** window, the **Relations** subtab can show such information for the object selected in the **Attachments**, **Where Used**, or **Versions** tab.

To view reference link values in the Preview tab or window:

Do one of the following:

- For an object in the hierarchical content table, select the object and click the **Preview** tab. You can open the **Preview** window for this object by clicking the **Open tab** button on the notebook pane's toolbar.
- For an object in the **Attachments**, **Links**, **Where Used**, or **Versions** tab:
 - With the **Preview** tab visible in the **Notebook** pane, click the **Open tab** button on the toolbar to open the **Preview** window.
 - Select the object in the **Attachments**, **Links**, **Where Used**, or **Versions** tab to display the content in the **Preview** window.

To view reference link objects in the Relations subtab of the Links tab or window:

Do one of the following:

- For an object in the hierarchical content table, select the object and click the **Links** tab, and then click the **Relations** subtab.

You can open the **Links** window for this object by clicking the **Open tab** button on the notebook pane's toolbar.

- For an object in the **Attachments**, **Where Used**, or **Versions** tab:
 - With the **Links** tab visible in the **Notebook** pane, click the **Open tab** button on the toolbar to open the **Links** window, and then click the **Relations** subtab.
 - Select the object in the **Attachments**, **Where Used**, or **Versions** tab.



For a live Visio diagram, select the diagram image note as the target, not the diagram object itself.

For more information, see [Creating a Diagram](#) in chapter 6, *Constructing System Views With Building Blocks and Diagrams*.

In the **Start** pane, reference links that target the selected object are displayed in the link table, at the right of the pane. At the left, the object table displays the object that contains each reference link.

In the **End** pane, reference links contained in the selected object are displayed in the link table, at the right of the pane. At the left, the object table displays the target object for each reference link.



In each pane, the object table and the link table remain synchronized at all times. For example, when you sort or scroll a link table, the corresponding object table is automatically sorted or scrolled to match the new view in the link table.

Deleting Reference Links

References can be deleted while editing text in Microsoft Word and from the Architect/Requirements client. When editing in Word, the delete actions are different for references to full text and references to property values.

- To delete the reference links between two objects:
 - . Select one of the objects in the hierarchical content table.
 - . Go to the **Links**→**Relations** tab.
 - . The **Relations** tab displays the reference links in which the object is participating. For each of the reference links to be deleted, right-click the reference link and click **Delete**.

The [**Broken Link xxx**] label appears in the objects that were using the deleted reference links, where *xxx* is the referenced text as it last appeared.

- When editing in Word, you can delete references to simple properties by selecting the range of text that is highlighted as a link and deleting those characters. When the edited text is saved, the reference link relation is deleted.
- To remove a reference to the full text of another object while editing in Word, switch to **View**→**Outline**. In the outlining toolbar, click **Remove Subdocument**. This command converts the embedded full text reference into local text. The local text can be selected and deleted, or blended into the remainder of the object's content. In either case, when the current text is saved, the reference link relation is deleted.

Chapter 10: Working With Versions

This chapter contains an overview of object versioning and instructions for working with versions and variants.

Overview of Versions and Variants

The majority of product development involves changing and improving an existing product, rather than starting from scratch. As a result, the product structures, requirements, and functions remain essentially the same as in the original design, but change slightly as the product matures. To accommodate this evolutionary process, Architect/Requirements employs versions and variants for requirements and building blocks.

Each version is an independent object, with its own relationships to other objects, including child objects and defining and complying objects. A version has all of the same child objects as the object from which it was created. Also, a version can be created from a prior version. The relationships among multiple versions of an object, from the original object, through prior versions, to the current *working* version, are illustrated by a *version tree*.

For objects with multiple versions, *effectivity* determines which version is displayed in the Architect/Requirements main window. The **Effectivity** fields provide controls for filtering versions according to *effectivity rules*, each of which specifies a category for display.

Like a version, a variant is an independent object, although it differs in its relationships. A variant is a copy without any connections to the original parent. The ROIN numbering is an additional count to the existing hierarchy. The variant has a different icon which is a requirement with a green circle on the left hand side of the object. You can have multiple variants at the same time. It's a way to create a new object that is related to an existing object without retaining the structure. A variant occupies a branch in the tree structure, with no relation to the rest of the tree.

For example, consider a requirement for miles per gallon for a compact car.

The requirement is refined to get better mileage per year. The requirement is the same but has new revisions each year. However, if you need a requirement for a midsize car, it is a different requirement and needs to be revised on its own each year. Such a requirement can be a variant of the requirement of the original compact car.

Deleting a variant does not affect the overall tree structure.

Architect/Requirements does not limit the number of versions and variants in a project. However, usability should be considered. Users may become confused if many different versions and variants of the same requirement or building block are in effect for a single product release. If a user does not see a

particular version, the cause may be that the effectivity is set so that the version is not in effect, and thus not displayed.

Enabling Versions for a Project

In the Architect/Requirements Administration module, the value for a project's **Packages** property determines whether versions are enabled. When **Version** is added to the value, users can do the following in the Systems Engineering and Requirements Management module:

- Create versions of requirements, building blocks, and their subtypes, and also create versions of prior versions of those objects. For more information, see [Creating Versions](#), later in this chapter.
- Choose effectivity rules through controls displayed in the **Address** bar. For more information, see [Filtering Versions by Effectivity Rule](#), later in this chapter.
- View a version tree in the **Versions** tab and window in the notebook pane. For more information, see [Viewing a Version Tree](#), later in this chapter, and [Versions Tab](#) and [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.
- Create baselines of requirements, building blocks, and their subtypes. For more information, see [Creating a Baseline](#), later in this chapter.
- View a baseline in the hierarchical content table. For more information, see [Viewing a Baseline](#), later in this chapter.



You must have **Project Administrator** privilege for the project.

To enable versions for a project:

1. On the module bar, click the **Administration** button to gain access to the Architect/Requirements Administration module.
2. In the navigation tree, select the **Projects** node.
The content table displays all projects to which you have access.
3. Select the project in the content table.
The **Properties** tab or window displays all viewable properties for the project.
4.
In the **Value** column, double-click the value for the **Packages** property.
The Multi-Choice dialog window is displayed.
5. Check the **Version** check box, and then click **OK** to close the dialog window.
A confirmation message states that the **Version** package cannot be removed and asks if you want to continue.
6. Click **Yes**.



This action cannot be reversed. The **Version** package cannot be removed from the project.

The **Version** value is added to the **Packages** property, indicating that versions are enabled for the project.

Filtering Versions by Effectivity Rule

For objects with multiple versions, effectivity determines which version is displayed in the hierarchical content table. You choose an effectivity rule to filter the display according to a version category.

To filter versions by effectivity rule:

In the **Effectivity** list, do one of the following:

- Select **Current Frozen Version** to display the most recently frozen versions, together with non-frozen objects.
- Select **Current Version** to display the most recently created versions, whether frozen or non-frozen, together with non-frozen objects.
- Select **Frozen As Of Date** to display frozen versions from a past date to the present, together with non-frozen objects.
- Select **Version As Of Date** to display both frozen and non-frozen versions from a past date to the present, together with non-frozen objects.
- Select **Baseline** to display only the objects in a particular baseline.
- Select **Baseline with in-work** to display objects in a baseline together with non-frozen objects.

The content table displays the versions that match the selected rule. In the notebook pane, the matching version is highlighted in the **Versions** tab or window.



The **Number** property for a requirement or a building block is displayed as a - (hyphen character) if the object is not effective. The non-effective objects can be seen only in the **Versions** tab in the **Notebook** pane.

If you add the **Number** property in the **Versions** tab, the **Number** property of an effective object is displayed in a position relative to its parent. However, a non-effective object has - in its **Number** property.

Freezing Objects

To create a version of a requirement or a building block, that object must first be made static, or frozen. You can freeze one object at a time, or you can freeze multiple objects at the same time. If you select only one object, you can also freeze all of its lower level descendants in a single action. For a selected folder, you can freeze all requirements and building blocks at all levels in the folder.



- For an object to which a security profile is applied, one of the following conditions must be true:
 - o Your user name must be included in either the **Full Control** or **Modify and Read Access** property value of the security profile.
 - o You must have **Project Administrator** privilege for the project.
- To freeze building blocks or TRAMs, you must have **Architect** privilege for the project.

To freeze objects:

1. In the hierarchical content table, select each object that you want to freeze.

You can select nonadjacent objects by holding down the control key while you click the objects. To select adjoining objects, click the first object, hold down the shift key, and click the last object.

2. Pull down the **Tools** menu, and then do one of the following:
 - To freeze only the objects in the selection, choose the **Versions**→**Freeze**→**Selected Objects** options.
 - To freeze one selected object and all of its descendants, choose the **Versions**→**Freeze**→**Deep** options.

A message asks you to confirm this action. Click **Yes** to continue.

A lock symbol appears on the object type indicator for each specified object, showing that the object is now static. The object cannot be modified, although versions of the object can be created and modified.

Procedure notes

Step 1: To see lower-level objects, click the plus signs. Or, select an object with a plus sign, pull down the **View** menu, and choose **Expand All**. You can also right-click an object and choose **Expand All** from the pop-up menu.

Step 2: You can reverse this action by pulling down the **Edit** menu and choosing **Undo**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Unfreezing Objects

A static requirement or building block can be unfrozen to return the object to its editable state. You can unfreeze one object at a time, or you can unfreeze multiple objects at the same time. If you select only one frozen object, you can also unfreeze all of its lower level descendants in a single action. For a selected folder, you can unfreeze all requirements and building blocks at all levels in the folder.



- For an object to which a security profile is applied, one of the following conditions must be true:
 - Your user name must be included in the **Full Control** property value of the security profile.
 - You must have **Project Administrator** privilege for the project.
- To unfreeze building blocks or TRAMs, you must have **Architect** privilege for the project.
- You cannot unfreeze objects that are baselined. For more information, see [Creating a Baseline](#), later in this chapter.

To unfreeze objects:

1. In the hierarchical content table, select each object that you want to unfreeze.

You can select nonadjacent objects by holding down the control key while you click the objects. To select adjoining objects, click the first object, hold down the shift key, and click the last object.

2. Pull down the **Tools** menu, and then do one of the following:
 - To unfreeze only the objects in the selection, choose the **Versions**→**UnFreeze**→**Selected Objects** options.
 - To unfreeze one selected object and all of its descendants, choose the **Versions**→**UnFreeze**→**Deep** options.

A message asks you to confirm this action. Click **Yes** to continue.

The lock symbol is removed from the object type indicator for each specified object, showing that the object is now unfrozen.

Procedure notes

Step 1: To see lower-level objects, click the plus signs. Or, select an object with a plus sign, pull down the **View** menu, and choose **Expand All**. You can also right-click an object and choose **Expand All** from the pop-up menu.

Step 2: You can reverse this action by pulling down the **Edit** menu and choosing **Undo**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Creating Versions

You can create versions of requirements and building blocks. The new version has the same properties and values, child objects, defining and complying objects, and notes as the object from which it was created. A version of a requirement has the same text as the prior requirement.

A version's ROIN is signified by the **-2** attached to the ROIN.

Versions play a part with effectivity. Only one version is viewable at a time. Versions keep a history of the object through the life cycle of the project.



- You can only create a new version from the current version. You cannot create versions from prior versions of requirements or building blocks.
- The object must be frozen before you can create the version. For more information, see [Freezing Objects](#), earlier in this chapter.
- You must have the **Modify** permission for the object from which you intend to create the version.
- To create versions of building blocks or TRAMs, you must have the **Architect** privilege for the project.

To create versions:

1. In the hierarchical content table, select each object for which you want to create a version.

You can select nonadjacent objects by holding down the control key while you click the objects. To select adjoining objects, click the first object, hold down the shift key, and click the last object.

2. Pull down the **Tools** menu and choose the **Versions**→**Create Version** options.

The content table displays the new version or versions:

- If you created one version, it is displayed with an open text field around the default name. Enter the version name in the text field, and then press the enter key.
- If you created multiple versions, you can rename each one separately. For more information, see [Renaming an Object](#) in chapter 4, *Maintaining a Project*.

Each new version is also added to the appropriate version tree in the **Versions** tab and window. For more information, see [Viewing a Version Tree](#), later in this chapter, and [Versions Tab and Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.

Procedure notes

Step 1: To see lower-level objects, click the plus signs. Or, select an object with a plus sign, pull down the **View** menu, and choose **Expand All**. You can also right-click an object and choose **Expand All** from the pop-up menu.

Step 2: You can also press control-E. You can reverse this action by pulling down the **Edit** menu and choosing **Undo New Version**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Creating Variants

You can create variants of requirements and building blocks, and of prior versions and variants of those objects. The new variant has the same properties and values, defining and complying objects, and notes as the object from which it was created. A variant of a requirement has the same text as the prior requirement. However, the new variant does not retain the base object's relationships to child objects.



- You must have **Modify** permission for the object from which you intend to create the variant.
- To create variants of building blocks or TRAMs, you must have **Architect** privilege for the project.

To create variants:

1. In the hierarchical content table, select each object for which you want to create a variant.

You can select nonadjacent objects by holding down the control key while you click the objects. To select adjoining objects, click the first object, hold down the shift key, and click the last object.

2. Pull down the **Tools** menu and choose the **Versions**→**Create Variant** options.

The content table displays the new variant or variants. You can rename each one separately. For more information, see [Renaming an Object](#) in chapter 4, *Maintaining a Project*.

Each new variant is also added to the appropriate version tree in the **Versions** tab and window. For more information, see [Viewing a Version Tree](#), later in this chapter, and [Versions Tab](#) and [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.

Procedure notes

Step 1: To see lower-level objects, click the plus signs. Or, select an object with a plus sign, pull down the **View** menu, and choose **Expand All**. You can also right-click an object and choose **Expand All** from the pop-up menu.

Step 2: You can also press control-I. You can reverse this action by pulling down the **Edit** menu and choosing **Undo New Variant**, by clicking the **Undo** button on the toolbar, or by pressing control-Z.

Viewing a Version Tree

A version tree shows the relationships among multiple versions of a requirement or a building block. You view a version tree in the **Versions** tab or window. For more information, see [Versions Tab](#) and [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*.

To view a version tree:

Do one of the following:

- For an object in the content table, select the object and click the **Versions** tab, or click the object's versions indicator.

You can open the **Versions** window for this object by clicking the **Open tab** button on the notebook pane's toolbar.
- For an object in the **Links** tab:
 - With the **Versions** tab on top in the notebook pane, click the **Open tab** button on the toolbar to open the **Versions** window.
 - Select the object in the **Links** tab.

The **Name** column displays the versions and variants in a hierarchy, from the earliest to the latest. A plus sign is shown for each version or variant that has later versions or variants, extending the hierarchy to lower levels.

You can view the successive versions or variants by doing the following:

- To view the direct successors, click the plus sign for the object.
- To view all successors, right-click the object, and then choose **Expand All** from the pop-up menu.

In the **Versions** tab, you can also select the object, pull down the **View** menu, and then choose **Expand All**.

Procedure notes

To see lower-level objects in the content table, click the plus signs. Or, select an object with a plus sign, pull down the **View** menu, and choose **Expand All**. You can also right-click an object and choose **Expand All** from the pop-up menu.

Deleting a Version or a Variant

Deleting a version has a large impact on the overall structure of the version tree. A variant occupies a branch in the tree structure, with no relation to the rest of the tree. Therefore, deleting a variant does not affect the overall tree structure, but merely removes it from the tree.



- You cannot delete a frozen object or a baselined object. For more information, see [Freezing Objects](#), earlier in this chapter, and [Creating a Baseline](#), later in this chapter.
- To delete a version or variant of a building block or a TRAM, you must have **Architect** privilege for the project.

To delete a version or a variant:

Select the version or variant in the content table, and then pull down the **File** menu and choose **Delete**.

The version or variant is moved to your Architect/Requirements Recycle Bin.

Procedure notes

To see lower-level objects, click the plus signs in the **Name** column. Or, select an object with a plus sign, pull down the **View** menu, and choose **Expand All**. You can also right-click an object and choose **Expand All** from the pop-up menu.

You can also right-click the version or variant and choose **Delete** from the pop-up menu. Or, click the **Delete** button on the toolbar.

Creating a Baseline

A baseline allows you to fix versionable objects permanently in an unalterable state. Versionable objects are requirements, building blocks, and their system-defined and user-defined subtypes. You can baseline these objects to record their history at a given time, for example, after a milestone such as requirements approval.

The baseline process freezes and labels all versionable objects at all levels in a selected container. You can select a project to baseline all of its versionable objects simultaneously. Or, you can select one or more folders to baseline only those objects.

First in this process, unfrozen versionable objects are frozen, while objects that are already frozen are skipped. Then, each frozen object is labeled with the baseline name that you assign. This label is shown by the **Baseline** property value for each baselined object. Shortcuts and groups are ignored for the baseline.

A baseline cannot be deleted, nor can a baselined object be unfrozen. However, new versions and variants can be created from a baselined object.



Do not create a baseline if you intend for the frozen state of the objects to be reversible. Instead, you can freeze the objects without assigning them a baseline name. If the **Baseline** property value is blank, a frozen object can be unfrozen to return it to an editable state. For more information, see [Freezing Objects](#) and [Unfreezing Objects](#), earlier in this chapter.

Baselines can be renamed through the **Baseline** property definition for the project. This multiple choice property definition resides in the **Property Definitions** folder in the Administration module. For more information about modifying a choice definition, see the *Systems Architect/Requirements Management Project Administrator's Manual*.



- To create a baseline of a building block or TRAM, you must have **Architect** privilege for the project.
- When you assign the baseline name, that value is added to the choice list for the project's **Baseline** property in the Administration module. Therefore, you must have **Project Administrator** privilege for the project in which you want to create the baseline in the Systems Engineering and Requirements Management module.
- The **Effectivity** field must first be set to **Current Version**. For more information, see [Filtering Versions by Effectivity Rule](#), earlier in this chapter.

To create a baseline:

1. Select the container whose objects you want to baseline.

In the navigation tree, you can select only one project or folder.

In the hierarchical content table, you can select one or more folders. To select nonadjacent folders, hold down the control key while you click the folders. To select adjoining folders, click the first folder, hold down the shift key, and click the last folder.

2.

Pull down the **Tools** menu and choose the **Versions**→**Create Baseline** options.

The Baseline Name dialog window is displayed.

3. Enter the baseline name in the text field, and then click **OK** or press the enter key.

A confirmation message states that this irreversible operation freezes and labels all versionable descendants of the selected containers.

4. To continue, click **Yes** or press the enter key.



- This action cannot be reversed.
- Baselined objects cannot be unfrozen.
- The baseline cannot be deleted.

In the content table, a lock symbol is shown on the object type indicator of each versionable object in the container. When one of these objects is selected, the baseline name is shown by the **Baseline** property value in the **Properties** tab or window.

When the **Effectivity** field is set to **Baseline**, the baseline name can be used to filter the view of objects in the hierarchical content table. For more information, see [Viewing a Baseline](#), later in this chapter.

Procedure notes

Step 1: To see lower-level objects, click the plus signs. Or, select an object with a plus sign, pull down the **View** menu, and choose **Expand All**. You can also right-click an object and choose **Expand All** from the pop-up menu.

Viewing a Baseline

Above the hierarchical content table, the **Effectivity** fields provide controls for filtering displayed objects according to a baseline. For the project or folder selected in the navigation tree, you can display only the objects in a baseline, or you can display baselined objects together with non-frozen objects. For more information, see [Creating a Baseline](#) and [Filtering Versions by Effectivity Rule](#), earlier in this chapter.

To view a baseline:

1.

In the **Effectivity** list, do one of the following:

- To display only objects in a particular baseline, select the **Baseline** effectivity rule.
- To display objects in a baseline and non-frozen objects, select the **Baseline with in-work** effectivity rule.

The Select a Baseline dialog window is displayed.

2. Check the check box for the baseline, and then click **OK** or press the enter key.

The content table displays each object whose **Baseline** property value includes the baseline name.

- If you selected the **Baseline** effectivity rule, all other objects are filtered from the content table.
- If you selected the **Baseline with in-work** effectivity rule, the content table also displays any non-frozen objects in the folder. These are the most recent versions.



The baseline name is displayed in the read-only field to the right of the **Effectivity** list. To view another baseline by the selected effectivity rule, you can display the **Select a Baseline** dialog window by clicking the button to the right of the read-only field.



If the referenced object is not in the same baseline as the referencing object, the reference is not effective when the effectivity is set to **Baseline**. The reference is effective only if the effectivity is set to the version of the baseline date.

Procedure Notes

Step 2: If necessary, click the plus sign (+) for the folder that contains the baselined objects. Or, select the folder, pull down the **View** menu, and choose **Expand All**. You can also right-click the folder and choose **Expand All** from the pop-up menu.

Chapter 11: Using the Search Module

This chapter contains an overview of the Teamcenter Systems Engineering and Requirements Management Search module and instructions for performing basic, intermediate, and advanced searches.

Overview of the Search Module

Through the Architect/Requirements Search module, you can quickly find objects within a project. In its concepts, functions, and appearance, the Search module is similar to the familiar Search feature of Microsoft Windows Explorer.

A search is always contained within a specified starting container, such as a project or a folder, or a hierarchical requirement or building block. Also, searches are always within a project, not across projects. The starting point for the search is determined by the way in which you activate the Search module.

You can perform basic, intermediate, and advanced searches. From any view in the Search module, you can save a search and recall it later. However, certain features, such as sorting the search results, are available only in the Advanced search view.

The Architect/Requirements Search module consists of the following views:

- The Basic search view, in which you can perform searches based on simple queries as in Microsoft Windows Explorer.
- The Intermediate search view, in which you can perform more complicated searches based on detailed criteria relating to object properties.
- The Advanced search view, in which you can perform complex searches based on queries that you construct, using statements similar to a structured query language.

In all views, the toolbar includes the **New**, **Open**, and **Save** buttons.

Activating the Search Module

The starting point for the search is determined by the way in which you activate the Search module:

- From the Systems Engineering and Requirements Management module or the Administration module, click the **Search** button on the module bar.



When the user changes projects, the **Search** button on the module bar is momentarily disabled until the project change initialization is completed.

The starting point is the object selected in the navigation tree in the Systems Engineering and Requirements Management module. However, this is not the case if the user previously activated the search module, specified a different starting point (for example, the object selected in the content table), and has not made a selection in the navigation tree since.

- From the Systems Engineering and Requirements Management module, click the **Search** button on the main window's toolbar.
 - If the navigation tree is active, the starting point is the project node or folder selected in the navigation tree.
 - If the content table is active, the starting point is the object selected in the content table.
- From the Systems Engineering and Requirements Management module, with an object selected in either the navigation tree or the content table, pull down the **Tools** menu and choose the **Search** option.
 - If the navigation tree is active, the starting point is the project node or folder selected in the navigation tree.
 - If the content table is active, the starting point is the object selected in the content table.
- From the Systems Engineering and Requirements Management module, with an object selected in either the navigation tree or the content table, right-click inside the pane and choose **Search** from the pop-up menu.

The starting point is the object selected in the right-clicked pane.

The **Look in** field displays the starting point for all methods of Search module activation.



When the starting point is the object selected in the content table, it does not change if the user activates the search module via the **Search** button on the module bar, unless the user selects an object in the navigation tree again.

Search Results



If you encounter an out of memory error when the search result is large, you may need to increase the maximum memory available for the Architect/Requirements client.

For information on increasing the available memory, see [Appendix D - Changing the maximum memory available for rich client](#).

Each view in the Search module contains the Basic pane. In this pane, you enter simple criteria such as an object name, a string of text, a ROIN of a requirement, an object type or subtype, or a specific location within the project.

Searching subordinates is incorporated in the search choices that are found in the dropdown menu. Select **Search subordinates: via Query** to perform search using the database query. Select **Search subordinates: via Relations** to perform search by traversing over the descendants of the selected object (specified in the **Look in** field). Select **Search members only** to limit search on the objects owned by the selected object.

You can also choose from two toggle choices, **Include Subtypes** and **Exclude Shortcuts**.

Additionally, the **Case Sensitive** check box is available to non-English locales, and defaults to case-insensitive search.

Also in the Basic pane, you use the **Output Options** section to choose the format in which the search results are displayed. You can choose from the following options:



For the Search Results dialog window:

- o Choose **Reuse Results Window** to output each search to the same instance of the Search Results dialog window. The window is cleared for each new search and refreshed with those results.
- o Choose **New Results Window** to output each search to a separate instance of the Search Results dialog window.

The property column settings in the Search Results dialog window are determined by the saved view that you select in the field at the bottom of the **Output Options** section. For more information, see [Search Results Dialog Window](#), later in this chapter, and [Customizing Views of Property Columns](#) in chapter 9, *Working With Object Properties*.



For a value that exceeds the cell size, you can rest the pointer on the cell to see the entire value in a tooltip. Multiple lines are used if necessary.



Choose **Microsoft Word** to output the search results to a Word document. The document contains data for each object that matches your search criteria. The data in the document, and the Word formatting styles applied to the data, are determined by the document template that you select in the field at the bottom of the **Output Options** section. For more information, see [Exporting a Report to Microsoft Office Word](#), later in this chapter.



Microsoft Office Word 2013 or Word 2016 must be installed on your computer.

- Choose **Microsoft Excel** to output the search results to an Excel spreadsheet. The spreadsheet contains data for each object that matches your search criteria. The data in the spreadsheet is determined by the Excel template that you select in the field at the bottom of the **Output Options** section. For more information, see [Exporting a Report to Microsoft Office Excel](#), later in this chapter.

 Microsoft Office Excel 2013 or Excel 2016 must be installed on your computer.
- Choose **Microsoft Visio** to output the search results to a Visio diagram. The diagram contains a shape for each object that matches your search criteria. The shapes in the diagram are determined by the Visio stencil that you select in the field at the bottom of the **Output Options** section. For more information, see [Exporting a Report to Microsoft Office Visio](#), later in this chapter.

 Microsoft Office Visio 2013 or Visio 2016 must be installed on your computer.
- Check the **Keep Previous Settings** check box to save the search criteria for using it with a different project or a folder.

Search Results Dialog Window

When you choose the Search Results dialog window, each object that matches your search criteria is displayed in a table. For an object selected in the Search Results dialog window, you can do the following:

- Navigate to the object in the content table by pulling down the **View** menu and choosing **Go To Object**, or by right-clicking the object and choosing **Go To Object** from the pop-up menu.
- Edit object property values directly in the table cells. You can also edit properties in the Edit Properties dialog window, which you display by pulling down the **File** menu and choosing **Properties**, or by right-clicking the object and choosing **Properties** from the pop-up menu. For more information, see chapter 9, [Working With Object Properties](#).

 For a value that exceeds the cell size, you can rest the pointer on the cell to see the entire value in a tooltip. Multiple lines are used if necessary.
- Copy or move an object to the Clipboard by pulling down the **Edit** menu and choosing **Copy** or **Cut**, or by right-clicking the object and choosing **Copy** or **Cut** from the pop-up menu. You can then paste the object in another location. For more information, see [Copying Objects](#) or [Moving Objects](#) in chapter 4, *Maintaining a Project*.
- Rename an object by pulling down the **File** menu and choosing **Rename**, or by right-clicking the object and choosing **Rename** from the pop-up menu. For more information, see [Renaming an Object](#) in chapter 4, *Maintaining a Project*.

- Delete an object by pulling down the **File** menu and choosing **Delete**, or by right-clicking the object and choosing **Delete** from the pop-up menu. For more information, see [Deleting Objects](#) in chapter 4, *Maintaining a Project*.

-

Export the Search Results to Microsoft Excel by pulling down the **File** menu and choosing the **Export**→**Excel Spreadsheet** options. You can also right-click in the table and choose the **Export**→**Excel Spreadsheet** options from the pop-up menu. For more information, see [Exporting Objects to Microsoft Office Excel](#) in chapter 4, *Maintaining a Project*.

- Export the Search Results to Microsoft Word by pulling down the **File** menu and choosing the **Export**→**Word Document** options. You can also right-click in the table and choose the **Export**→**Word Document** options from the pop-up menu. For more information, see [Exporting Objects to Microsoft Office Word](#) in chapter 4, *Maintaining a Project*.
- Add and remove property columns, rearrange and resize columns, and sort by any column. For more information, see [Adding and Removing Columns](#), [Rearranging Columns](#), [Resizing Columns](#), and [Setting the Sort Column](#) in chapter 9, *Working With Object Properties*.
- Copy an object's URL to the Clipboard by pulling down the **Edit** menu and choosing **Copy URL**→**Include Full Name** or **URL Only**, or by right-clicking the object and choosing **Copy URL**→**Include Full Name** or **URL Only** from the pop-up menu. You can then paste the URL into Windows programs that support hyperlinks. For more information, see [Copying Object URLs](#) in chapter 4, *Maintaining a Project*.
- Open a folder, or open a requirement or a note for editing, by pulling down the **File** menu and choosing **Open**, or by right-clicking the object and choosing **Open** from the pop-up menu. For more information, see [Entering and Changing Requirement Content in Microsoft Office Word](#) in chapter 5, *Managing Requirements*, or [Editing a Note](#) in chapter 8, *Recording Supplementary Information With Notes*.
- Open a requirement or a note for viewing by pulling down the **File** menu and choosing **Open Read-Only**, or by right-clicking the object and choosing **Open Read-Only** from the pop-up menu. For more information, see [Viewing Requirement Content](#) in chapter 5, *Managing Requirements*, or [Viewing Note Content](#) in chapter 8, *Recording Supplementary Information With Notes*.
- Create trace links by pulling down the **Edit** menu and choosing **Links**→ options, or by right-clicking the object and choosing **Links**→ options from the pop-up menu. For more information, see [Creating Trace Links](#) or [Linking to an Object in Another Teamcenter Product](#) in chapter 7, *Showing Object Relationships With Trace Links*.

Saving, Recalling, and Running Searches and Reports

From the Basic, Intermediate, or Advanced view in the Search module, you can save a search and recall it later. For example, you can save a search that you frequently perform, and recall it to use the same criteria for convenience. Or, you can open a saved search and modify its criteria, and then output those results or save your changes as a different search. A search that you recall is automatically displayed in the same view from which it was saved.

Saved search reports are specific to a project. Searches are stored as schema objects in the project's **Reports and Formatting** folder in the Administration module. If you have questions about search reports, consult your project administrator.



You can copy the URLs of these objects to run reports independently of the Architect/Requirements client. In the content table, right-click a search object and choose **Copy URL to Run** from the pop-up menu.

Then, paste the URL into a Windows program that supports hyperlinks, such as Microsoft Outlook. By clicking the URL hyperlink in an E-mail message, for example, a recipient can run the report without running the client. The user must log in with an Architect/Requirements user name and password.

The **Shared State** property is assigned to each new search. This property value determines the report's availability among the users in the project:

- With the **Private** value, the report can be recalled only by its creator, and only that user can change the report. **Private** is the default value for each new report.
- With the **Public** value, the report can be recalled by all users in the project. The report can be changed only by a user with **Project Administrator** privilege for the project. A project administrator can also delete a public report. In addition, a project administrator can assign the **Public** value to a private report.
- With the **Pending** value, a private report is marked as a request for the project administrator to make the report public. The report's creator can assign the **Pending** value to mark the request. Then, the project administrator can assign the **Public** value to make the report available to the project's other users.

Private, public, and pending reports can be exported from the Administration module as can other schema objects. However, only public reports can be imported. For more information about exporting and importing project data, see the *Systems Architect/Requirements Management Project Administrator's Manual*.

To save search criteria:

1.

On the Search module toolbar, click the **Save** button to display the Saved Report Information dialog window.

2. Enter a unique name in the **Report Name** field.

You can also do the following:

- Enter additional text in the **Brief Report Description** field.
- Check the **Use same starting point every time** check box to begin at the currently selected point each time you recall the search.

3. Click **OK** or press the enter key.

To recall search criteria:

1.

On the Search module toolbar, click the **Open** button to display the Select Saved Report dialog window.

2. Check the check box for the search, and then click **OK** or press the enter key.

The Basic, Intermediate, or Advanced view displays the search criteria.

To run a report:

1.

In the Search module or the Systems Engineering and Requirements Management module, pull down the **Tools** menu and choose **Run Report** to display the Select Report dialog window.

2. Check the check box for the saved search criteria to use for the report, and then click **OK** or press the enter key.



By default, the starting point for the report is shown in the **Look in** field. However, a different starting point may be specified in the saved search.

A message states that the request is submitted. Depending on server response time, a progress indicator is displayed. Above the indicator, the message changes to show the current stage of the process.

You can continue working in the Architect/Requirements client during this process. When the message closes, the objects that match the search criteria are displayed in the specified output format.



- If you click **Cancel**, the process runs until the server reaches the next checkpoint.
- Your Architect/Requirements system administrator can set the server polling interval through the Web Application Configuration administrative tool. For more information about administrative tools, see the *Systems Architect/Requirements Management System Administrator's Manual*.

Exporting a Report to Microsoft Office Excel

For criteria that you enter in any Search module view, you can export search results directly to a Microsoft Office Excel workbook. As with workbooks exported from other views of the Architect/Requirements client, the report contains property values for objects that meet your search criteria. For more information, see [Exporting Objects to Microsoft Office Excel](#) in chapter 4, *Maintaining a Project*.

You can export the search results to a workbook that is not connected to the Architect/Requirements database. Or, you can use the report to create a new live Excel workbook for editing property values interactively. For more information, see [Using the Live Excel Interface](#) in chapter 9, *Working With Object Properties*.

To specify the properties, you can base the report on an Excel template, which contains property columns, tags for property values, and other information. Or, you can base the report on a saved view of property columns. For more information about saved views, see [Customizing Views of Property Columns](#) in chapter 9, *Working With Object Properties*. For more information about Excel templates, see the *Systems Architect/Requirements Management Project Administrator's Manual*. If you have questions about Excel templates or views, consult your project administrator.



Microsoft Office Excel 2013 or Excel 2016 must be installed on your computer.

To export a report to Microsoft Office Excel:

1. Enter your complete search criteria in the Basic Search view, Intermediate Search view, or the Advanced Search view.

For more information, see [Basic Search View](#), [Intermediate Search View](#), or [Advanced Search View](#), later in this chapter.

2. Under **Output Options**, select **Microsoft Excel**.



You can check the **Live?** check box to export the report to a live Excel workbook.

At the bottom of **Output Options**, the list shows the Excel templates and the public views for the project, including the private and pending views for your user name.

3. In the list, select the Excel template or the saved view.



- The Excel template may contain multiple worksheets, and each worksheet may specify different properties and rules. Therefore, data may be separated on worksheets in the export file.
- If the Excel template or the view contains properties that do not apply to an object type specified for export, the inapplicable values are indicated in the related cells.



For a public report, Siemens PLM Software recommends that you select a public view. If the report is imported later with other project data, pending and private views will not be available because they cannot be imported. For more information, see [Customizing Views of Property Columns](#) in chapter 9, *Working With Object Properties*. If you have

questions about reports, views, or importing project data, consult your system administrator.

4. Click the **Search** button.

A message states that the request is submitted. Depending on server response time, a progress indicator is displayed. Above the indicator, the message changes to show the current stage of the process. You can continue working in the Architect/Requirements client during this process.



- If you click **Cancel**, the process runs until the server reaches the next checkpoint.
- Your Architect/Requirements system administrator can set the server polling interval through the Web Application Configuration administrative tool. For more information about administrative tools, see the *Systems Architect/Requirements Management System Administrator's Manual*.

When the message closes, the report opens in a read-only Excel workbook (.xlsm). In this temporary file, you can view and print the data and send it by E-mail and fax. This file is deleted from your computer when you exit Architect/Requirements.

You can create a permanent workbook from this report by clicking Excel's **Office Button** and choosing **Save As** to display the **Save As** dialog window. A permanent workbook can serve as record for comparison with future changes, or as a means of editing properties through live Excel. Also, the data in the workbook can be imported to create new objects in any Architect/Requirements project. For more information, see [Importing Objects From Microsoft Office Excel](#) in chapter 4, *Maintaining a Project*.

Procedure Notes

Step 4: If you checked the **Live?** check box in step 2, and if the live Office interface is not installed, a message asks if you want to install live Office. Click **OK** to start the installation. When the installation is complete, repeat steps 2 through 4.

Exporting a Report to Microsoft Office Word

For criteria that you enter in any Search module view, you can export search results directly to a Microsoft Office Word document. As with Word documents generated from other views of the Architect/Requirements client, the report contains data such as property values for objects that meet your search criteria. For more information, see [Exporting Objects to Microsoft Office Word](#) in chapter 4, *Maintaining a Project*.

You can also choose the document template on which you want to base the report. For more information about document templates, see the *Systems Architect/Requirements Management Project Administrator's Manual*. If you have questions about document templates, consult your project administrator.



Microsoft Office Word 2013 or Microsoft Office Word 2016 must be installed on your computer.

To export a report to Microsoft Office Word:

1. Enter your complete search criteria in the Basic Search view, Intermediate Search view, or the Advanced Search view.

For more information, see [Basic Search View](#), [Intermediate Search View](#), or [Advanced Search View](#), later in this chapter.

2. Under **Output Options**, select **Microsoft Word**.

The field at the bottom of the **Output Options** section displays the default document template for this report. You can choose a different document template in this field.

3. Click the **Search** button.

A message states that the request is submitted. Depending on server response time, a progress indicator is displayed. Above the indicator, the message changes to show the current stage of the process. You can continue working in the Architect/Requirements client during this process.



- If you click **Cancel**, the process runs until the server reaches the next checkpoint.
- Your Architect/Requirements system administrator can set the server polling interval through the Web Application Configuration administrative tool. For more information about administrative tools, see the *Systems Architect/Requirements Management System Administrator's Manual*.

When the message closes, the report opens in a read-only Word document (**.html**). In this temporary file, you can view and print the data and send it by E-mail and fax. This file is deleted from your computer when you exit Architect/Requirements.

You can create a permanent document from this report by clicking Word's **Office Button** and choosing **Save As** to display the **Save As** dialog window. A permanent document can serve as record for comparison with future changes. Also, the data in the document can be imported to create new requirements in any Architect/Requirements project. For more information, see [Creating Requirements and Content by Import From Microsoft Office Word](#) in chapter 5, *Managing Requirements*.

Exporting a Report to Microsoft Office Visio

You can export search results to a Microsoft Office Visio diagram from any Search module view. The diagram contains shapes and other data for objects that meet your search criteria. You can choose the Visio stencil on which you want to base the report. For more information, see [Basic Search View](#), [Intermediate Search View](#), or [Advanced Search View](#), later in this chapter. For more information about Visio stencils, see the *Systems Architect/Requirements Management Project Administrator's Manual*. If you have questions about stencils, consult your project administrator.



- Microsoft Office 2013 or a later version and Office Live must be installed. The Office Live version must match your Architect/Requirements version. For more information, see [Using the Live Office Interface](#) in chapter 3, *Using the Architect/Requirements Main Window*.
- The report is exported to a static diagram, not to a live Visio diagram.

To export a report to Microsoft Office Visio:

1. Enter your search criteria.



- If the criteria include qualifying relationships, for example, with **ADD** statements in the Advanced Search view, the diagram displays results in a multiple level tree. With this structure, you can easily see how system elements work together. For more information, see [FOR EACH and ADD Statements](#), later in this chapter.
- If you enter criteria in the Basic Search view or the Intermediate Search view, all results are displayed at the same level.

2. Under **Output Options**, select **Microsoft Visio**.
3. In the field at the bottom of **Output Options**, select **Default Static Tree Stencil** or a custom static tree stencil for your project.
4. Click the **Search** button.

A message states that the request is submitted. Depending on server response time, a progress indicator is displayed. Above the indicator, the message changes to show the current stage of the process. You can continue working in the Architect/Requirements client during this process.



- If you click **Cancel**, the process runs until the server reaches the next checkpoint.
- Your Architect/Requirements system administrator can set the server polling interval through the Web Application Configuration administrative tool. For more information about administrative tools, see the *Systems Architect/Requirements Management System Administrator's Manual*.

When the message closes, Visio opens a read-only diagram, which you can view, print, and send by E-mail. This file is deleted from your computer when you exit Architect/Requirements.

In a diagram that contains a tree, the superior level shows the objects that match the search criteria, and subordinate levels show the objects that fulfill the qualifying relationships, which are indicated by colored lines. Initially, the tree is displayed vertically with the superior level at the top.



The color for each relationship category is set in the mapping file (.xml) associated with the stencil.

- To change the vertical or horizontal orientation of the tree, right-click a blank area on the Visio page and choose **Display Left to Right / Top to Bottom** from the pop-up menu.
- To hide or show one or more relationships in the entire diagram:



Right-click a blank area on the Visio page and choose **Toggle Relationship** from the pop-up menu.

The Toggle Relationship dialog window is displayed. Initially, the checked relationships are those that are specified in the search criteria.

- Do one or both of the following:
 - Clear the check box for each relationship that you want to hide.
 - Check the check box for each relationship that you want to show.



Relationships not specified in the search criteria are not shown even if you check the check boxes.

By checking **Redraw Tree in Visio**, you can reset the layout of the shapes when you apply your changes. The current layout is retained when this check box is cleared.

- Click **Apply** to see your changes and keep the dialog window open, or click **OK** to apply your changes and close the dialog window.



The Toggle Relationships dialog window does not apply to a diagram that displays all search results at the same level.

- To hide or show all relationships for one object, right-click the shape and choose **Show / Hide Subordinates** from the pop-up menu.



This option is available only in a diagram that displays search results in a tree.

- To view an object's properties, right-click the shape and choose **Teamcenter for systems engineering properties** from the pop-up menu.

- To navigate to an object in the Architect/Requirements client:
 - Right-click the shape and choose **Go To Teamcenter for systems engineering** from the pop-up menu to display the login page.
 - Enter your user name and password, and then click **Login** to see the object highlighted in the client.

You can save the diagram locally with Visio's **Save As** dialog window. When you reopen the file, however, you cannot show or hide relationships or subordinates, nor can you change the orientation.

Procedure Notes

Step 4: If the Office Live interface is not installed, a message asks if you want to install Office Live. Click **OK** to start the installation. When the installation is complete, repeat steps 2 through 4.

Order of Search Results

The search functionality is enhanced for improving the query performance. As a consequence, the order of search results in some cases may be different from the order in the previous Architect/Requirements releases.

Subtype Order: When the search includes multiple subtypes of a base type, the search results include a mix of objects of all subtypes. The mixed order is because the search performs a single query for better performance.

To get the results clustered by subtype, choose **Advanced** search type and add a **SORT** with Subtype clause to the search query. If the search includes only one subtype or when the search is on a base type, the subtypes are automatically clustered in the search results.

View Sort Order: When search results are displayed in a View, the view sort order is used to order the search results if the search query does not contain the **SORT** clause. By default, the search results are ordered by the view's sort column, except when the search query includes a **SORT** clause.



- For any search that does not include a **SORT** clause, the view's sort order may be different than the natural search order. For example, the natural order of the result with a **FOR EACH** clause is the **Number** order. This may be different from the sort order specified by the selected view. In this case, the view sort order is used to order the search results.
- In case of output to Word or output to Excel, the search results are ordered by the natural search order. For example, the natural order of the result with a **FOR EACH** clause is the **Number** order.

Basic Search View

When you first activate the Search module, the Basic search view is displayed by default. This view contains only the Basic search pane. You use this pane to search for objects that meet basic criteria, such as an object name, a string of text, a ROIN of a requirement, an object type or subtype, or a specific location within the project.

The search starts at the object that is selected in the Systems Engineering and Requirements Management module when the Search module is activated. The target and search criteria that you specify determine the search results that you obtain. Selecting an appropriate starting point and the specific object types of interest yields search results more quickly than does a broad search for all object types across an entire project.

At the top of the Basic search view, the **Look in** field displays the container that is the starting point for the search. You can change the starting point by selecting a different container in this field. For more information, see [Activating the Search Module](#), earlier in this chapter.

To the right of the **Look in** field, the **Keep Same Location** check box is cleared by default, allowing you to automatically set the starting location to your last selection in the Systems Engineering and Requirements Management module. You can check the **Keep Same Location** check box to switch to the Systems Engineering and Requirements Management module while maintaining the current starting point in the Search module.



Selecting your Architect/Requirements Recycle Bin in the Systems Engineering and Requirements Management module does not change the starting point in the **Look in** field.

The Basic search view also contains the following:

- **Look in** field
- **Search For Name** field
- **Containing Text** field
- **Matching ROIN** field
- **Types/SubTypes** tree
- **Search** button
- **Search Type** section
- **Search Options** section
- **Output Options** section
- **Keep Same Location** check box

Figure 11-1 shows the Basic search view in the Search module.

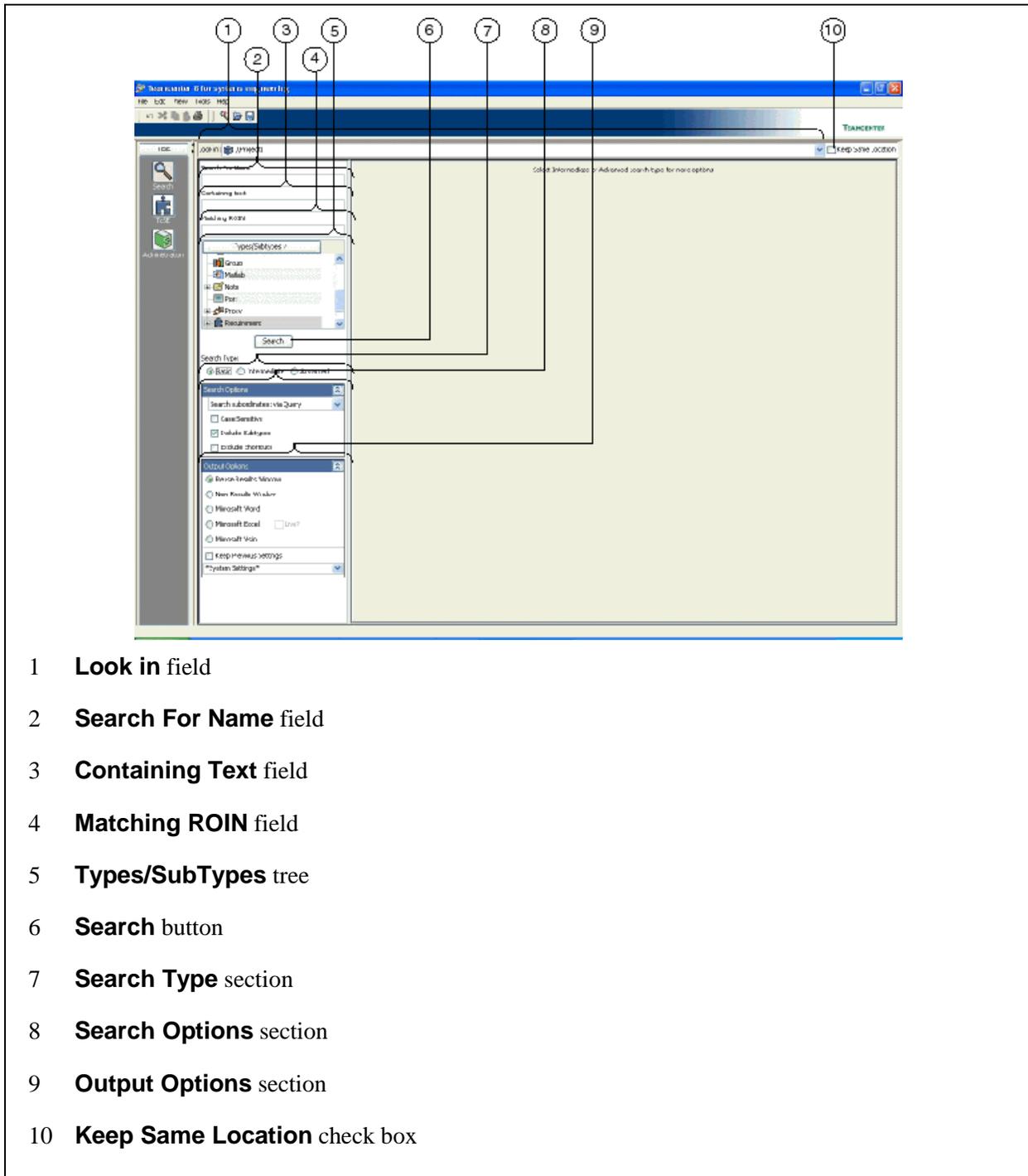


Figure 11-1. Basic Search View in Architect/Requirements Search Module

To use the Basic search view:

1. To specify the search criteria, do one or both of the following:
 - To limit the search to objects that have a specific name, contain specific text, or ROIN, do one or both of the following:

o

In the **Search For Name** field, enter any portion of the name that you want to find.



If you leave this field empty, the search extends to all objects specified in the **Types/SubTypes** tree.

o

In the **Containing text** field, enter any portion of the text that you want to find.

o

In the **Matching ROIN** field, enter any portion of the ROIN of the requirement that you want to find.

These fields are not case sensitive. In all the three fields, you can use wildcard characters to substitute for unknown characters or to cover several possible conditions. Wildcard characters are the asterisk (*) and the question mark (?). For example:

- o The characters **a*b** locate the strings **ab**, **axb**, and **axyzb**.
- o The characters **a?b** locate the strings **axb**, **ayb**, and **azb**.
- o The characters **00*** locate the ROINs **001**, **0002**, and **00345**.



o

In the **Search For Name** field and the **Containing text** field, you do not need to enter wildcard characters for a relative search, which is performed automatically.

o

In the **Matching ROIN** field, enter wildcard characters for a relative search.

If you want to search for text that actually contains an asterisk (*), a question mark (?), an open bracket ([) or closed bracket (]), or a backslash (\), you must precede such characters with a backslash (\).



You can start the search at this point by placing the cursor in either field and pressing the enter key.

●

To extend or limit the search to specific object types or subtypes, do one or both of the following in the **Types/SubTypes** tree:

- o To include an object type or subtype in the search, select the type or subtype in the tree.

You can select multiple types or subtypes by holding down the control key while you click the items. To select adjoining items, click the first item, hold down the shift key, and click the last item.

- o To exclude a currently selected object type or subtype from the search, clear the type or subtype from the tree.

You can clear multiple types or subtypes by holding down the control key while you click the items.



In the **Search Options** section, do one or more of the following:

- o

Select **Search subordinates: via Query** from the dropdown box to perform search using the database query.

- o

Select **Search subordinates: via Relations** from the dropdown box to perform search by traversing over the descendents of the selected object (specified in the **Look in** field).

- o

Select **Search members only** from the dropdown box to limit search on the objects owned by the selected object.

- o

Check the **Case Sensitive** check box to limit the search to the case entered in the text fields.

- o

Clear the **Include Subtypes** check box to exclude subtypes of types selected in the **Types/SubTypes** tree.

- o

Check the **Exclude Shortcuts** check box to exclude shortcuts in the search result.



In advanced searches, the **Exclude Shortcuts** toggle only applies to the initial **SELECT** statement. To remove shortcuts from a **FOREACH ADD**, insert a where clause, toggle **Tcl Script**, and choose the **Not Shortcut** activator.



If shortcuts are included, there is a difference in the behavior between searches depending on relations and searches depending on query. When the search is done depending on relations, both direct shortcuts as displayed with black shortcut icon overlay and shortcut descendents as displayed with red shortcut icon overlay are included in the result. When search is done depending on query, only direct shortcuts are included in the result.



The **Search subordinates: via Query** and **Search subordinates: via Relations** selections do not affect what is included in the search result. They only affect the search performance.

When a project or a high level folder (with a large number of descendents) is selected, the search performance is better with the **Search subordinates: via Query** selection. When a lower level folder (with a relatively small number of descendents) is selected, the search performance is better with the **Search subordinates: via Relations** selection.

For related information on improving the search performance using database indexes, see the *Systems Architect/Requirements Management Server Installation Manual*.



In the **Output Options** section, select the format for the search results. For more information, see [Search Results](#), earlier in this chapter.

2. Click the **Search** button to display the search results in the format selected in the **Output Options** section.

You can clear the current criteria and start a new search by clicking the **New** button.

Intermediate Search View

The Intermediate search view allows you to select properties for specified object types and refine the search by specifying criteria for any of the properties. After activating the Search module, you display the Intermediate search view by clicking the **Intermediate** button in the **Search Type** section in the Basic search pane. For more information, see [Activating the Search Module](#) and [Basic Search View](#), earlier in this chapter.

At the top of the Intermediate search view, the **Look in** field displays the container that is the starting point for the search. You can change the starting point by selecting a different container in this field. For more information, see [Activating the Search Module](#), earlier in this chapter.

To the right of the **Look in** field, the **Keep Same Location** check box is cleared by default, allowing you to automatically set the starting location to your last selection in the Systems Engineering and Requirements Management module. You can check the **Keep Same Location** check box to switch to the Systems Engineering and Requirements Management module while maintaining the current starting point in the Search module.



Selecting your Architect/Requirements Recycle Bin in the Systems Engineering and Requirements Management module does not change the starting point in the **Look in** field.

The Intermediate search view also contains the following:

- The **Select Properties** pane, which lists all available properties for the selected object types in the basic **Search** pane.
- The **Specify Criteria** pane, in which you can specify the selection criteria for an object property. Initially, this pane displays criteria based on the property selected in the **Select Properties** pane. The criteria change according to the type of property that you select. For example, the pane displays different criteria for numeric, date, text, and choice properties.
The **Specify Criteria** pane also displays criteria for the row that you select in the **Edit Search Criteria** pane. You can remove specified criteria by clicking **Delete Row**.
- The **Edit Search Criteria** pane displays each set of criteria specified in the **Specify Criteria** pane.
You can select a row in the **Edit Search Criteria** pane to change that criteria in the **Specify Criteria** pane.

To use the Intermediate search view:

1. In the **Search Type** section of the Basic pane, click the **Intermediate** button.

The **Select Properties** pane displays all available properties for the object types selected in the **Types/SubTypes** tree in the Basic pane.

2.

Select a property in the **Select Properties** pane to display the criteria for the property type in the **Specify Criteria** pane.



The selected property does not remain highlighted, allowing you to select that same property a second time. There are times when you may want to add the same property to a search more than once. For example, you may want to search for all change users whose ID starts with *t* but is not equal to *test*. You can accomplish this by including the **Change User** property twice, once for each rule.

3.

Click any field in the **Specify Criteria** pane to automatically add criteria to the **Edit Search Criteria** pane.

You can remove criteria from the **Edit Search Criteria** pane by selecting the criteria and pressing the **Delete Row** button.

Depending on the type of property selected in the **Select Properties** pane, one of the following is displayed in the **Specify Criteria** pane:

- The text criteria panel allows you to specify search criteria for text properties. You can specify text that exactly matches the string you enter, and you can specify partial text matches. Also, you can specify text that does not equal or does not contain certain criteria.

You can choose either the **Case Insensitive** or **Case Sensitive** option for any text criteria. Also, you can enter regular expressions directly in a text field.

- The date criteria panel allows you to specify search criteria for date properties. It also shows the format of that property. You can search for dates that occurred in the last or next number of days, weeks, months, or years. Or, you can indicate a specific date and search for items whose property matches that date, that does not equal that date, occurs before that date, or occurs after that date. The final option allows you to specify a date range.

This panel includes a calendar widget. Clicking the calendar icon displays the widget, in which you can select a date in a calendar dialog window. Clicking the **OK** button accepts the selected date. Clicking **Cancel** or pressing the escape key closes the dialog window without changing the selected date in the text field.



You can search for objects that have a property whose value is **TBD**, which marks the value for entry at a later date. You can also exclude these objects from the search.

- o To search for objects with a **TBD** property value, select **TBD** in the **Specify Criteria** pane and choose **is** from the list to the right.

- o To exclude objects with a **TBD** property value, select **TBD** in the **Specify Criteria** pane and choose **is not** from the list to the right.

- The numeric criteria panel allows you to specify search criteria for numeric properties. It also shows the format of that property. You can search for numbers that using any one of several available operands, including *equals*, *not equals*, *less than*, *less than or equal to*, *greater than*, and *greater than or equal to*. You can also indicate a range that the number must fall within.
- The choice criteria panel allows you to specify search criteria for properties with choices. A list displays the choices that you can select. The operands you can use are *equals*, *any*, *not equals*, and *not any*.

The *equals* operand is inclusive and means that you will only find items that match each of the choices you selected. The *any* operand is exclusive and means that you will find items that match any of the choices you selected.

4. To change specified criteria, select the appropriate row in the **Edit Search Criteria** table, and then change the criteria in the **Specify Criteria** pane.
5. To begin the search, click the **Search** button.



- To begin the search without pressing the **Search** button, press the enter key when the cursor is inside a text field.



When you select a property in the **Select Properties** pane, it does not stay selected. Therefore, you can immediately select that same property again and specify additional criteria using the same field.

- Criteria are automatically added to the search when you click any field in a criteria pane.
- To remove criteria from the search, click the **Delete Row** button while editing that criteria in the **Specify Criteria** pane.



Switching from Basic or Advanced search to Intermediate search does not carry over the search criteria. You must enter the criteria again to replicate the search.

Advanced Search View

Although the Advanced search view shares some components with the other views in the Search module, the Advanced view has a fundamental difference. In the Basic and Intermediate views, you can search for objects using limited criteria, and it is not necessary to understand the concept of a query object that is constructed behind the scenes. In the Advanced search view, however, you see and directly construct a query. If you have used a structured query language to query a database, then you are already familiar with this concept.

You can create complex queries through the Advanced search view because it includes additional statements and properties not available in the other two search modes. The specific statements that are available will be discussed later. The Basic and Intermediate views limit the number of properties to a standard, predefined set of the most commonly used system properties, and those user interfaces are less complicated. In the Advanced search view, all properties for the selected object types are available.

At the top of the Advanced search view, the **Look in** field displays the container that is the starting point for the search. You can change the starting point by selecting a different container in this field. For more information, see [Activating the Search Module](#), earlier in this chapter.

To the right of the **Look in** field, the **Keep Same Location** check box is cleared by default, allowing you to automatically set the starting location to your last selection in the Systems Engineering and Requirements Management module. You can check the **Keep Same Location** check box to switch to the Systems Engineering and Requirements Management module while maintaining the current starting point in the Search module.



Selecting your Architect/Requirements Recycle Bin in the Systems Engineering and Requirements Management module does not change the starting point in the **Look in** field.

The Advanced search view also contains the following:

- Basic pane
- **Query View** pane
- **Modify** pane
- **Query Edit** pane
- **Properties** pane
- **Specify Criteria** pane

Figure 11-3 shows the Advanced search view in the Search module.

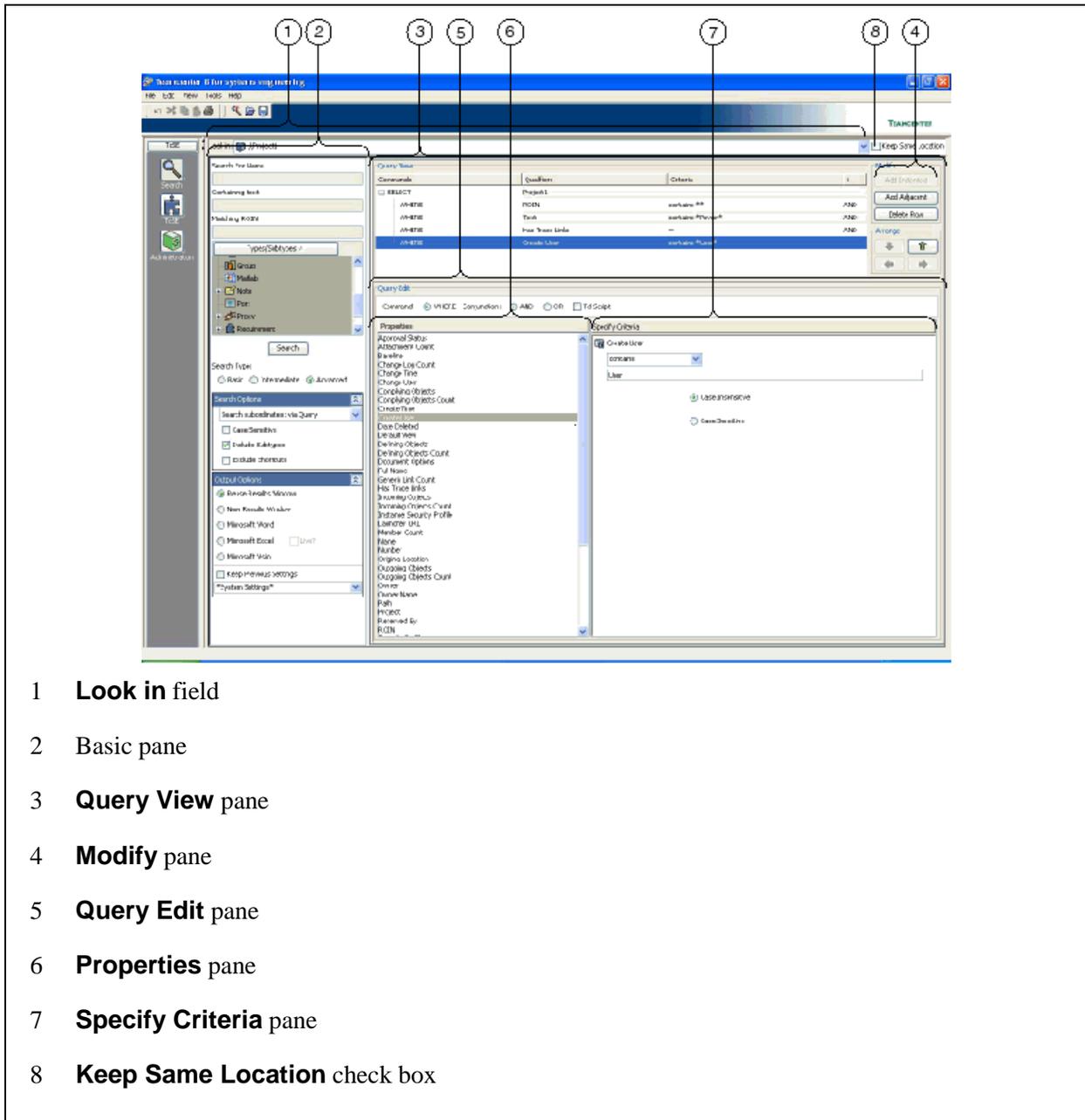


Figure 11-3. Advanced Search View in Architect/Requirements Search Module

Switching to the Advanced Search View

When you first start Architect/Requirements and switch to the Search module, it defaults to the Basic search. You can change the search type to the Intermediate search, which extends the Basic view. Regardless of which of the two search modes you are in, switching to the Advanced Query mode always takes any search criteria you already entered to build the initial query.



Because switching to the advanced mode always builds a new query from whichever search mode you were already in, you should be very careful about switching from the Advanced Query mode down to a lower level search if you want to keep your query information. If you construct a query in the Advanced Query mode, switch to the Basic or Intermediate search, and then switch back to Advanced, your previous advanced query will be replaced by the new one that is built from the criteria you had entered in the lower level search. If you want to keep your advanced query but switch to a lower level query for a new search, it is best to first save your query using the **Save** button.

Basic Pane

The Basic pane of the Advanced search view is the same pane used in the Basic and Intermediate views, with two exceptions:



The **Search for Name** and **Containing text** fields are disabled because any values entered there are displayed as **WHERE** clauses in the query tree view. Although they are disabled, they remain visible so that the other components in the Basic search pane do not move when you switch to the Advanced view.



The **Types/Subtypes** tree is automatically enabled and disabled as appropriate for the type of statement you are editing in the query. Query statements are qualified by either object types or by properties. If a statement uses object types, then the **Types/Subtypes** tree is enabled while you edit that statement. If a statement use properties, then the **Types/Subtypes** tree is disabled, and the **Properties** pane is enabled.

Query View Pane

The **Query View** pane displays the query as it is being constructed. However, this pane is more than a read-only view of the query. It is a hierarchical tree table where rows can be selected, added, updated, and deleted. The query tree table view is the major component that drives the creation and updating of a query in the Advanced search view.

The **Query View** pane contains three columns, **Commands**, **Qualifiers**, and **Criteria**. Depending on the row that is selected, what goes in the columns can vary slightly, but basically you can read the query top-down, left to right.

Modify Pane

The **Modify** pane contains buttons through which you modify the statements contained in the **Query View** pane. Each button is used to change specific rows in the query and is automatically enabled and disabled as appropriate to the selected row. To see a description of a button in a tooltip, you can rest the pointer on the button.

Because a query is a hierarchy, indention is important to understanding the query and its execution. The **Add Indented** button adds a subordinate statement (a child node) indented under the current row. For example, because a **SELECT** statement is the root node for a query, the **Add Indented** button is the only button that is enabled when you select the first row of the query tree. When you click this button, you see a list of the statement types that are valid as child nodes of the selected statement. You use the **Query Edit** pane to specify what statement you want to add. The **Add Indented** button is disabled when you click on a statement that cannot have child nodes, such as a **WHERE** clause.

The **Add Adjacent** button adds a statement, or sibling, after the current row.

The **Delete Row** button deletes the current row and any of its child nodes. You can select multiple rows in the query tree table by clicking a row and then either using shift-click to select a range of rows or control-click to select multiple, nonadjacent rows. If you select multiple rows, the button name changes to **Delete Rows**, and all selected rows and their children are deleted.

You can use the arrow buttons under **Arrange** to change the structure of the query without changing individual statements. All button actions move the selected row and any of its child nodes along with it.

- The down arrow button moves the current row and any of its children down below its next sibling.
- The up arrow button moves the current row and any of its children above its previous sibling.
- The left arrow button promotes a statement up one level to make it a sibling of its current parent.
- The right arrow button demotes a statement down one level to make it a child node of its previous sibling.



Each button is enabled or disabled as is generally appropriate but not every possible combination is covered. It is possible to construct a query that makes no sense or does not execute. If the query does not return results, verify that the query is properly constructed. If an error message is returned, a specific problem may be indicated.

Query Edit Pane

The **Query Edit** pane displays the statements that relate to the row that you select in the **Query View** pane. When you add a new row, the valid types of commands that can be added for the selected row in the query tree table are shown to allow you to choose which type of command you want.

When you choose a statement type, the user interface changes to present you with the options that are valid for that type of statement. Selecting a row in the query tree table allows you to edit an existing statement. When you do this, the command area shows what type of command it is, but it cannot be changed. For more information, see [Statement Types](#), later in this chapter.

The **Properties** pane is for specifying properties for those statements that use properties, such as **WHERE** and **SORT**. Because this is the Advanced Query, all possible properties for the selected object types are listed, unlike the Basic and Intermediate views that have a more limited set of properties.

The **Specify Criteria** pane is for specifying criteria for a given property. When you select an item from the **Properties** pane, the **Specify Criteria** pane displays the appropriate controls for that property type. For example, if you select a date property such as **Create Time**, the **Specify Criteria** pane displays controls that allow you to specify criteria such as a specific date or a date range.

Constructing Advanced Queries

Although the Advanced search view limits your choices to those options that may be valid at any given time, it does not prevent you from creating a query that is illogical or cannot be executed. Queries allow greater flexibility, but a thorough understanding of their construction is required. Therefore, consider the following rules in constructing queries.

- When you switch to the Advanced view from the Basic or Intermediate view, any search information that is already entered from the previous view is used to build an initial query.
- A query is represented by a hierarchical tree structure.
- Each query starts with a **SELECT** statement.
- There are two kinds of statements, those that are qualified by selecting object types, and those that are qualified by specifying property criteria.
- Once a statement is added, you cannot change its statement type.
- The user interface does not require that siblings be in a certain order but the server will process them in a specific order. For more information, see [Sibling Order](#), later in this chapter.
- All **WHERE** clauses are implicitly joined together with an **AND** conjunction. You can specify an explicit **OR** condition between multiple **WHERE** clauses.

Hierarchical Structure

The first thing to consider about a query is that it is hierarchical. The first statement is always a **SELECT** statement, and every other statement below it is subordinate to it. The user interface indicates these levels of a hierarchy by using the indentation of a standard tree table component. This is the same type of tree interface used in many other views in Architect/Requirements, such as the content table of the Systems Engineering and Requirements Management module.

Each item in the tree is referred to as a node. The top level node is called the root node. Nodes are containers that can have ownership of other nodes. A node that contains other nodes is called a parent node. Parent nodes can be collapsed and expanded. A node that is contained within a higher level node is called a child node. Nodes at the same level and with the same parent are called siblings. A node that does not have any children is called a leaf node.



A query is represented by a hierarchical tree table. In order to understand how to properly construct a query and use the interface, it is essential that you understand the concepts of an outline tree structure and how indentation affects the scope of a statement.

Because the query is a hierarchy, it should come as no surprise that operations for a row affect its children. For example, deleting a parent row will also delete all its child nodes. As you would expect, moving a parent row will move its children along with it. And, promote and demote (moving a row left and right in the hierarchy) adjusts the hierarchical level one position for the parent and all its children.

Statement Qualifiers

Each line of a query is a statement that is used to define what you are searching for. Statements have qualifiers and criteria that narrow their scope. Statements are qualified using either an object type or by specifying property criteria. For example, in a **SELECT** statement, you can specify which type of objects you want to iterate over. And, in a **WHERE** clause you can narrow down those objects even further by specifying criteria for one or more properties.

It is not necessary to know whether a statement requires object types or properties because the user interface enables the appropriate panel and disables the inappropriate one for that type of statement.

Statement Types

When you select a line in the query tree table and press a button to add a statement to a query, the user interface gives you a choice of valid statements for that point in the query. Once you choose what type of statement you want, for example, **WHERE**, **SORT**, or **REMOVE**, you cannot change it. You can change the qualifiers and criteria for that statement, but you cannot change its statement type.

The reason for this is because each statement has certain rules about where they can be used (what type of clause is their parent) and what must be specified as qualifiers, criteria, and valid subordinate clauses. Changing a type could invalidate the whole query and cause an error. These types of problems are avoided and the user interface is simplified by not allowing statement types to be changed.



Statement types cannot be changed because it could invalidate the query structure. Deleting a query row and adding a new one accomplishes the same thing and allows the user interface to verify that the new statement conforms to the query language syntax.

Sibling Order

In an Architect/Requirements query, the order of siblings is usually important to its functionality. Although the user interface does not require you to put siblings in a specific order, it helps to understand the order that is used to process sibling statements. The server processes query statements in the following order:

1. **SELECT** statement
2. **WHERE** clauses
3. **SORT** clauses
4. All other subcommands

Sometimes the order of siblings is unimportant. For example, if a **SELECT** statement contains multiple **WHERE** clauses and a **SORT** clause, it does not matter which of the three clauses come first because the server still processes them in the order stated above.

In an Architect/Requirements query, the order of siblings is usually unimportant. For example, the following two queries are identical in their results because the two **WHERE** clauses and the one **FOR EACH** clause are all at the same level.

```
SELECT Requirements
  WHERE Name like "Require"
  WHERE Text like "shall"
  FOR EACH Paragraph
    Add Notes
  SORT by NAME

SELECT Requirements
  FOR EACH Paragraph
    Add Notes
  SORT by NAME
  WHERE Text like "shall"
  WHERE Name like "Require"
```

But, most of the time the sibling order does matter. If you have multiple **SORT** clauses at the same level, then the first one indicates the primary sort, the second one indicates the secondary sort, and so on. It does not matter whether the **SORT** clause appears before or after any **WHERE** clauses that are its siblings. For more information, see [SORT Clause](#), later in this chapter.

Also, a **FOR EACH** statement requires that the first child under it be an **ADD** clause or a **WHERE** clause. So, in this case, the order of siblings is important because the **ADD** and the **WHERE** clauses must appear before any other clauses under a **FOR EACH** statement. It may help if you think of the **FOR EACH** and the **ADD** clauses as two parts of one statement.

Additionally, if you have multiple **FOR EACH** statements at the same level, the server will process the first one and its result will be processed by any subsequent **FOR EACH** clauses. Likewise, if you have multiple **REMOVE** clauses that are siblings, the server processes them in the order that you've indicated. Some objects may be removed from the first **REMOVE** statement before it gets to any subsequent statements.

Advanced Query Statements

This section discusses the query statements used in the Advanced search view.

SELECT Statement

Each query starts with a **SELECT** statement and there is only one **SELECT** statement in a query. You cannot create and join multiple **SELECT** statements together. When you enter the Advanced view from either the Basic or Intermediate view, the initial **SELECT** statement is built based on the object types that are currently selected in the **Types/SubTypes** tree of the Basic pane.



You can use **WHERE** clauses to narrow the scope of any **SELECT** statement. For more information, see [WHERE Clause](#), later in this chapter.

A **SELECT** statement has an optional parameter which is indicated in the **Query Edit** pane with a **Use Starting Object** check box. If you check this check box, the object identified in the **Look in** field is used as a starting location. For example, instead of selecting all requirements, the first statement could select a specific requirement as a starting point. Then all subsequent statements, such as a **FOR EACH** and an **ADD** statement, would operate on that object.



The **Look in** field at the top of all the search panels may change depending on how the search module is invoked.

When **Use Starting Object** is selected, the qualifier of the **SELECT** statement in the query tree table changes from one or more objects types to the name of the specific object that you've chosen as the starting point. The only time this might be confusing is if the object name you're starting with is similar to an object type. For example, if you have a requirement object named **Requirement**, then it wouldn't be clear by just looking at the **SELECT** statement row whether you were searching for requirement object types or the specific object named **Requirement**.

Using meaningful names for objects such as folders and requirements is a good practice. It also avoids confusion when searching for object types versus searching with a given object as the starting point.

WHERE Clause

A **WHERE** clause narrows a search and is subordinate to another statement, such as a **SELECT** or a **FOR EACH** statement. In multiple **WHERE** clauses, the conjunctions **AND** and **OR** are explicit and are used in the **Query View** pane as follows:

- *Conjunction statements* are used to group a set of **WHERE** clauses for evaluation, specifying their precedence in the same way as parentheses specify the order of evaluation for mathematical equations. You can specify **AND** and **OR** conjunction statements as qualifying statements. Each conjunction statement must follow a **WHERE** clause and must have at least one subordinate **WHERE** clause. To specify a conjunction statement:
 - For a child of another statement, select the parent, click **Add Indented** in the **Modify** pane, and then click **AND** or **OR** in the **Query Edit** pane.
 - For a sibling of another statement, select the sibling, click **Add Adjacent** in the **Modify** pane, and then click **AND** or **OR** in the **Query Edit** pane.

A subordinate **WHERE** clause is automatically added below the new conjunction statement. You then complete the blank **WHERE** clause. For example:

```
SELECT Requirement
  WHERE Status = "closed"
  AND
    WHERE Version = "5.1.1"
    WHERE Version = "5.1.2"
  OR
    WHERE Create Date > 1/1/04
```

- *Ending conjunctions* are used in **WHERE** clauses, to control how an individual clause is evaluated with the result of its next sibling **WHERE** clause:
 - An **AND** conjunction specifies a condition in which both that **WHERE** clause *and* its next sibling must be true.
 - An **OR** conjunction specifies a condition in which either that **WHERE** clause *or* its next sibling may be true.

The **AND** conjunction ends all **WHERE** clauses by default. Therefore, the **AND** button is selected automatically in the **Query Edit** pane when you add a new **WHERE** clause. To specify an **OR** conjunction, select the **WHERE** clause and click the **OR** button. For example:

```
SELECT Requirement
  WHERE Assigned To = "Frank" AND
  WHERE Status = "closed" AND
  WHERE Version = "5.1.1" OR
  WHERE Version = "5.1.2"
```

For multiple sibling **WHERE** clauses with mixed conjunctions, **AND** conditions are evaluated first. Then **OR** conditions are evaluated. For the last clause in a series, an ending conjunction has no effect and is hidden in the **Query View** pane.

Multichoice fields allow for **OR** conditions because you can specify the operators **ANY** or **NOT ANY**. Using the **ANY** operator, for example, you can specify the following:

```
WHERE Subtype is Any Document or Folder
```

FOR EACH and ADD Statements

A **FOR EACH** statement iterates over the result of the parent statement that it is subordinate to in order to add more objects to the result set. As stated earlier, the first child under a **FOR EACH** statement should be an **ADD** clause.



Because the query language structure requires that a **FOR EACH** statement have an **ADD** clause, it may help to think of the **FOR EACH** statement and the **ADD** clause under it as just one statement written on two lines.

When you create a **FOR EACH** statement, you can choose an object type as a qualifier, but it is not necessary to do so. For example, you may want to iterate over all requirement objects within a folder and add notes for all paragraph objects. Because paragraphs are subtypes of requirements, the start of your query might look something like this:

```
SELECT Requirement
  FOR EACH Paragraph
    Add Notes
```



You can use **WHERE** clauses to narrow the scope of any **FOR EACH** or **ADD** statement. For more information, see [WHERE Clause](#), earlier in this chapter.

Unlike most statements, the qualifier is optional in a **FOR EACH** statement. Omitting a qualifier is equivalent to specifying all of the object types that you have selected in the result set up to that point. If no qualifier is specified, the **FOR EACH** statement iterates over those object types.

Each **ADD** clause must specify a relationship. When you create a new **ADD** clause or select an **ADD** row in the query edit table, the **Query Edit** pane in the middle of the right panel contains a box to specify the relationship between the objects that result from the **FOR EACH** statement and what kind of objects you want to add to that result. For example, you can add all complying objects, defining objects, diagrams, and notes.

An **ADD** clause also contains an optional parameter to specify whether you want its scope to be deep or not. This box is also in the **Query Edit** pane. If the deep box is not checked, only objects at the first level under the **FOR EACH** result set are added to the result. If the deep option is used, objects are added from each level below the starting result set.

```
SELECT Requirement Specification
  FOR EACH
    ADD DEEP Members
      WHERE Release = "2.0"
```

You can use multiple **ADD** clauses with a single **FOR EACH** statement. This allows showing more than one relationship at a particular level in the result. When multiple **ADD** clauses include objects at the same level in the hierarchy, indented under the same parent object, the results are displayed in the same order as the **ADD** clauses. You can rearrange adjacent **ADD** clauses using the up and down arrows to get the desired order in the result. Non-adjacent **ADD** clauses can also include objects at the same hierarchy level.

You can not rearrange the non-adjacent **ADD** clauses. If you need to rearrange non-adjacent **ADD** clauses, use the **Move To Top** check box, to control the result order. Selecting **Move To Top** for an **ADD** clause causes the result objects to be inserted before previous objects at the same hierarchy level.



TOP is appended to the **ADD** command in the query view when **Move To Top** is selected.

Table 11-1 describes the options in the **Relationship** field in the **Query Edit** pane.

Table 11-1. Options in Relationship Field for ADD Statements

Option	Description
Child Diagrams	Specifies all child diagram objects in a parent diagram to which the superior FOR EACH statement applies.
Complying Link List	Specifies complying trace links for the objects to which the superior FOR EACH statement applies.
Complying Objects	Specifies objects that comply with the objects to which the superior FOR EACH statement applies.
Connection List	Specifies objects that are connected to the objects to which the superior FOR EACH statement applies. Both starting and ending connections may be output.
Data Definitions	Specifies all data definitions in the data dictionary to which the superior FOR EACH statement applies.
Defining Link List	Specifies defining trace links for the objects to which the superior FOR EACH statement applies.
Defining Objects	Specifies objects that define the objects to which the superior FOR EACH statement applies.
Diagrams	Specifies diagrams attached to the objects to which the superior FOR EACH statement applies.
From Connections	Specifies connections that start at the objects to which the superior FOR EACH statement applies.
Incoming Link List	Specifies incoming generic links for the objects to which the superior FOR EACH statement applies.
Incoming Objects	Specifies objects that have incoming generic links to which the superior FOR EACH statement applies.
Next Version	Returns the next version plus any variants to which the superior FOR EACH statement applies. Use the deep option to get all later versions.
MatLAB Objects	Specifies all MatLAB objects to which the superior FOR EACH statement applies.
Members	Specifies objects owned by the objects to which the superior FOR EACH statement applies.

Table 11-1. Options in Relationship Field for ADD Statements

Option	Description
Notes	Specifies notes attached to the objects to which the superior FOR EACH statement applies.
Outgoing Link List	Specifies outgoing generic links for the objects to which the superior FOR EACH statement applies.
Outgoing Objects	Specifies objects that have outgoing generic links to which the superior FOR EACH statement applies.
Owner	Specifies the object that owns each object to which the superior FOR EACH statement applies.
Ports	Specifies all port objects that are included in the Visio diagram to which the superior FOR EACH statement applies.
Prior Version	Returns the immediately prior version to which the superior FOR EACH statement applies. Use the deep option to get all prior versions.
Referencing	Specifies objects that are pointed to by reference links in the objects to which the superior FOR EACH statement applies.
Spreadsheet List	Specifies spreadsheets attached to the objects to which the superior FOR EACH statement applies.
To Connections	Specifies connections that end at the objects to which the superior FOR EACH statement applies.
Uses	Specifies objects referenced by the objects to which the superior FOR EACH statement applies.
Version List	Returns all versions and variants for an object (including itself) to which the superior FOR EACH statement applies.
Where Referenced	Specifies objects that contain reference links pointing to the objects to which the superior FOR EACH statement applies.
Where Used	Specifies the groups, diagrams, and remote proxies that reference the objects to which the superior FOR EACH statement applies.

Examples of FOR EACH and ADD Statements

Following are examples for show trace relationships.

The **Defining Objects** and **Complying Objects** keywords are used to follow trace link relationships. The deep option allows you to recursively follow those relationships.

```
SELECT Requirement
  FOR EACH
    ADD DEEP Complying Objects
```

To include the trace link objects in the result, use the **Defining Link List** and **Complying Link List** relationships . Following example shows the defining object and the connecting trace link.

```
SELECT Requirement
  FOR EACH
    ADD Defining Link List
  FOR EACH
    ADD Defining Objects
```

You can use multiple **ADD** statements with a **FOR EACH** to show multiple relationships. For example, to see all the trace relationships for a requirement, use the following statement:

```
SELECT Requirement
  FOR EACH
    ADD Complying Objects
    ADD Defining Objects
```

Different icons are used in the search result window to distinguish the relationship represented. Multiple non-adjacent **ADD** statements can include results at same hierarchy level. By default, the **Members** are displayed before sibling **Defining Objects** because the **ADD Members** clause comes first. Selecting **Move To Top** in the **ADD Defining Objects** clause reverses the order and puts **Defining Objects** before the sibling members. For example:

```
SELECT MyDocument
  FOR EACH
    ADD DEEP Members
  FOR EACH
    ADD TOP Defining Objects
```

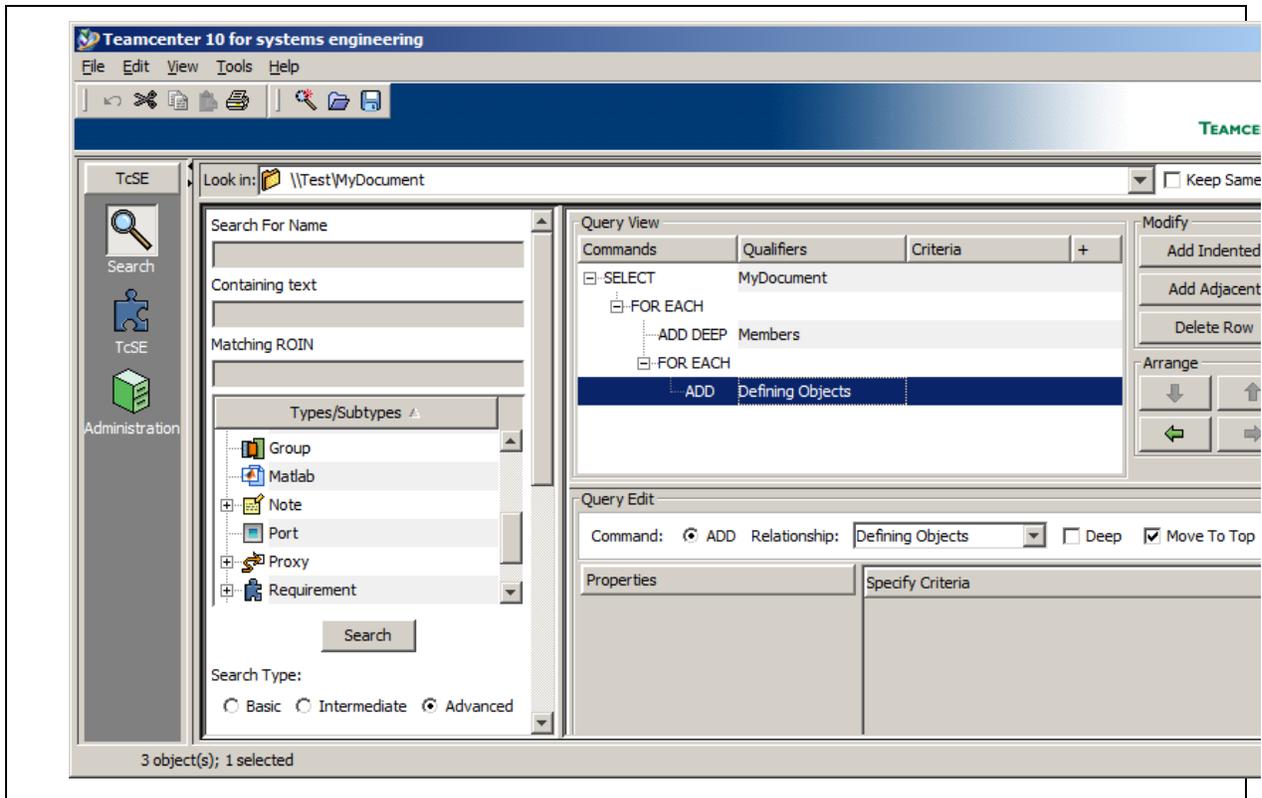


Figure 11-4. Example of the Advanced Search with Move to Top selected

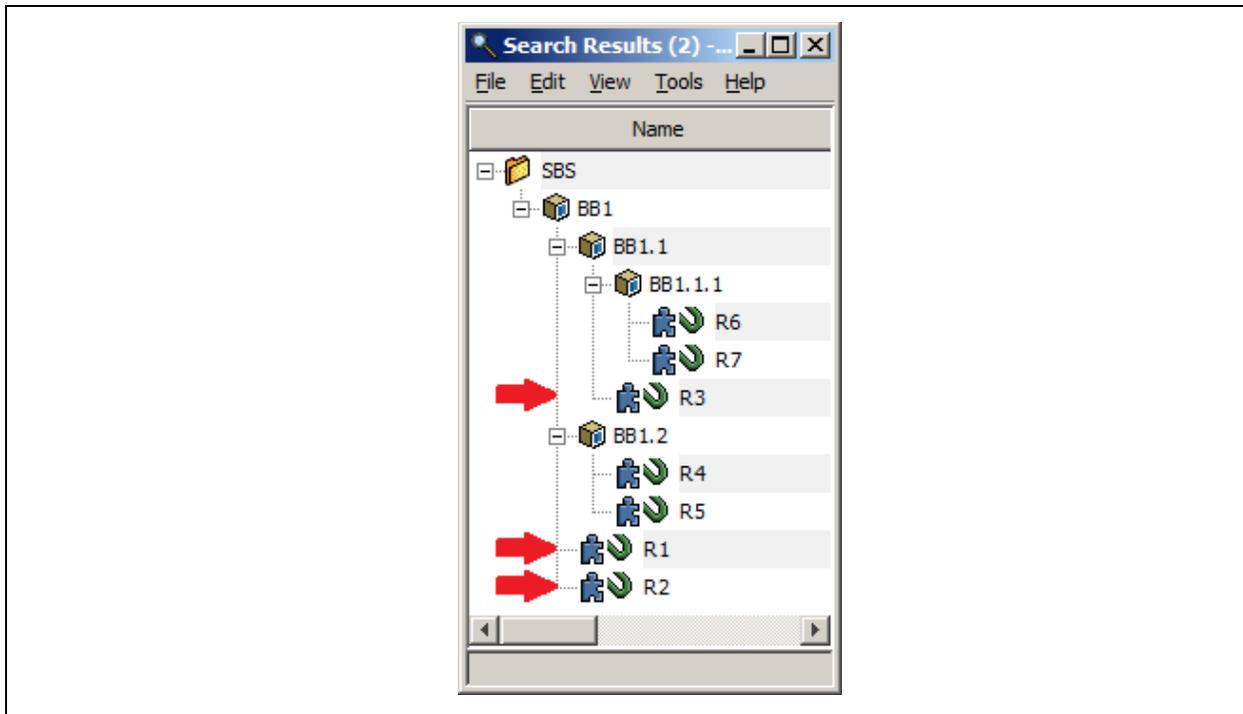


Figure 11-5. Output of the Search when Move To Top is not selected

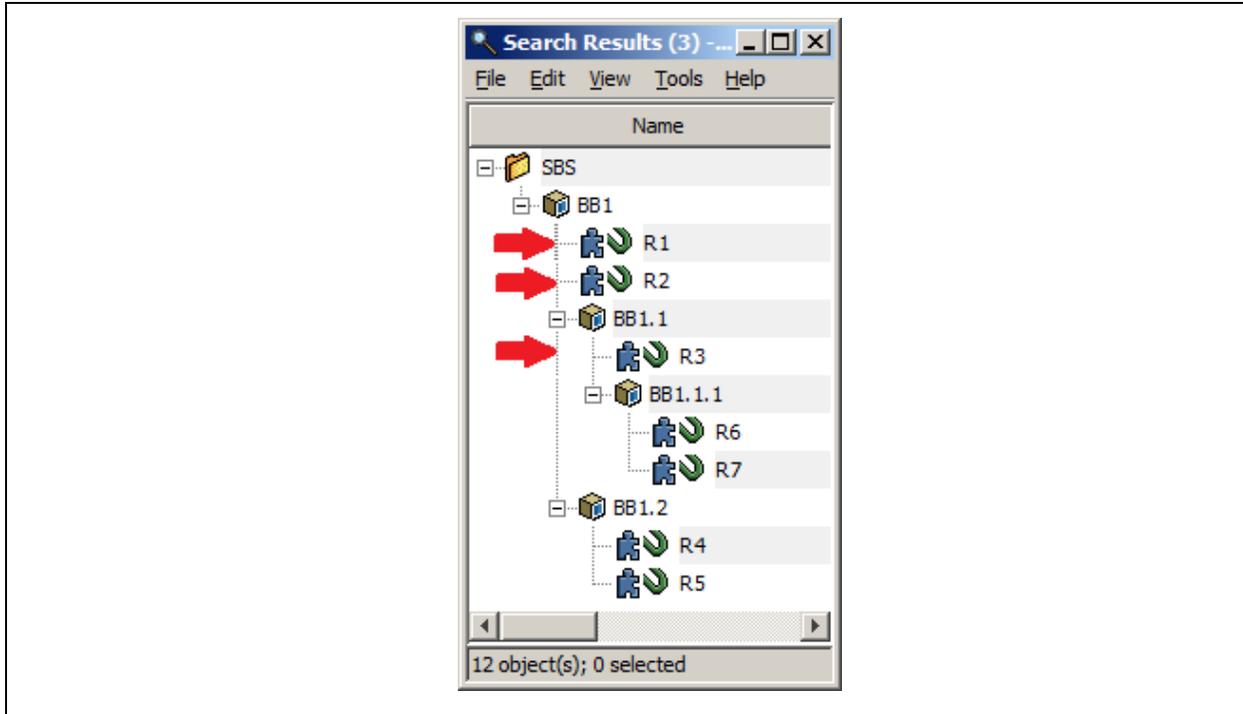


Figure 11-6. Output of the Search after Move To Top is selected

You can search for Trace Links and show the related objects using the following search statement:

```
SELECT Trace Link
FOR EACH
    ADD Complying Objects
    ADD Defining Objects
```

REMOVE Clause

A **REMOVE** clause is used to remove items from the query result. In some queries, you may have to temporarily add objects to the result to get to other objects. **REMOVE** can then get rid of the temporary objects. For example you may want to find all the design elements that may be affected by a requirement change. If the requirement is not directly linked to the design elements, but is indirectly linked through derived requirements, then the derived requirements would have to be added to the result in order to get to the design elements. The derived requirements could then be removed to simplify the result.

```
SELECT Customer Requirement
  FOR EACH
    ADD ComplyingDerived Requirement
  FOR EACH
    ADD Building Block
  REMOVE Derived Requirement
```

A **REMOVE** clause can only be used directly under a **SELECT** or **FOR EACH** statement (and after the **ADD** statement that must be the first child node under a **FOR EACH**). A **REMOVE** statement is qualified by which type of objects you want to remove and can be further defined by adding **WHERE** clauses subordinate to the **REMOVE** clause. In fact, a **WHERE** clause is the only type of child node allowed under a **REMOVE** clause.

SORT Clause

A **SORT** clause is used to arrange the query results in a specific order. Unlike some statements, the order of **SORT** clauses is important. The first **SORT** clause specifies the primary sort, and each sibling **SORT** clause specifies subsequent sort criteria. For example, if the first **SORT** clause is to sort by **Create User** and the next adjacent **SORT** clause is **Create Time**, you see the results sorted primarily by the user that created the object, and within objects created by the same user the results are sorted by create date and time.

Each **SORT** clause is qualified by the property you want to sort on and whether you want to sort ascending or descending. The default is ascending.



The **SORT** clause applies to the **SELECT** and **FOR EACH**, **ADD** commands that add objects to the search result. Further sorting of objects that are already in the result, such as after a **REMOVE** command, is not supported.

Whenever the Architect/Requirements server sorts the data due to a **SORT** clause in the query, the Architect/Requirements client removes the sorted column indicator (up or down arrow on the sorted column's header). The absence of the sorted column indicator shows the data was sorted on the server and was not altered by the client, regardless of which **Named View** was selected. This feature allows you to use a view to specify the properties you want to display and override the saved sort order (saved view) with the advanced query statements. This is important because an advanced query allows for multiple **SORT** clauses so that there can be a primary sort order followed by subsequent secondary sorts. A **Named View** does not allow for secondary sorts.

Chapter 12: Using the Change Management Package

This chapter provides an overview of the Architect/Requirements change management package and contains instructions for submitting, approving, rejecting, and viewing changes.

Overview of Change Management

Requirements and building blocks can be submitted through the Architect/Requirements change management package for change approval. A submitted object may be either of the following:

- An object with no prior versions is a new submission. When the changes are submitted, the object is frozen for the first time.
- An object with prior versions is a revision to previously submitted content. When the changes are submitted, this revision is frozen.



To use the change management package, the **Version** package must be enabled for the project. For more information, see [Enabling Versions for a Project](#) in chapter 10, *Working With Versions*.

The change management package tracks the following events:

- - A **Submit** event occurs when a requirement or a building block is submitted for approval. The package sends E-mail to recipients in the following categories:
 - *Approvers* have the responsibility to approve or reject the changes. Each of these users receives an E-mail message containing two hyperlinks:
 - A hyperlink to the object in the Architect/Requirements client. The user can click this hyperlink to navigate to the object and review the changes.
 - A hyperlink to the change management package. The user can click this hyperlink to navigate to the approval and rejection page in Microsoft Internet Explorer.

o

Notifiers do not approve or reject but are notified of submitted changes for informational purposes. Each of these users receives an E-mail message containing a hyperlink to the object in the Teamcenter Systems Engineering and Requirements Management client. The user can click the hyperlink to navigate to the object and review the changes.

Each message also contains the submitter's comments, if any. In addition, each approver and notifier receives the object's content in **.mhtml** format as an E-mail attachment. For an object that has prior versions, the previously submitted content is included.

The **Submit** event automatically creates a *change approval object* for the submitted object. Displayed in the **Attachments** tab or window, the change approval object contains a table of information related to the approval process. The initial table row is added for the **Submit** event.

•

A **Response** event occurs each time an approval or a rejection is received.

For each response, a row is added to the change approval object that is attached to the submitted object. The change management package also tracks delinquent responses.

•

An **Approved** event occurs if the changes are approved by all approvers. The approved object remains frozen. An E-mail message is sent to each approver and to the submitter, stating that the changes are approved.

The **Approved** event concludes the approval process, and a final row is added to the change approval object.

•

A **Rejected** event occurs each time the changes are rejected by one approver. When the first **Rejected** event occurs, the rejected object is unfrozen and remains in that state.

For each **Rejected** event:

- o An E-mail message is sent to the submitter only, stating that the changes are rejected by the individual approver.
- o A row is added to the change approval object for this event.

If other **Response** events are received after the first rejection, each later event is added to the change approval object.



All aspects of the change management package can be customized through the Teamcenter Systems Engineering and Requirements Management application programming interface (API). For more information, see the *Systems Architect/Requirements Management API Reference*.

Submitting Changes for Approval

When you submit changes for a requirement or a building block, the change management package sends an E-mail message to the approvers and notifiers. These users must be identified in a security profile, and that security profile must be applied to the requirement or building block. If you have questions about

security profiles, consult your project administrator. For more information about security profiles, see the *Systems Architect/Requirements Management Project Administrator's Manual*.



Each approver automatically has full access to the changes until that user approves or rejects the change request.

You can apply the security profile through the object's **Security Profile** property in the **Properties** tab or window.



- You must have **Modify** permission for the object.
- To submit a building block or a TRAM, you must have **Architect** privilege for the project.

To submit changes for approval:

1. In the hierarchical content table, select the requirement or building block that contains the changes.
- 2.

Pull down the **Tools** menu and choose the **Versions**→**Submit For Approval** options.

The Submit For Approval dialog window is displayed. The **Approvals**, **Notify**, and **Object(s)** fields are read-only, displaying information from the applicable security profile. The following fields are optional:

- In the **Subject** field, you can overwrite the default text with a description of your particular request.
 - In the **Message** field, you can enter plain text for the body of the E-mail message.
3. To close the dialog window and send the request, click **OK**.

A confirmation message states that the request for approval is sent, and the E-mail message is sent to the approvers and notifiers.

If the object is not frozen, a lock symbol appears on the object type indicator in the hierarchical content table. If the changes are approved, the object remains frozen. If the changes are rejected by one approver, the object is unfrozen.

In the **Attachments** tab or window, a change approval object is automatically created for the submitted object. The change approval object contains a table in which the columns display information related to the approval process. Initially, the table contains only a row for your submission. A new row is added for each **Response** event that occurs. If the changes are approved, a final row is added for that event. For more information, see [Overview of Change Management](#), earlier in this chapter, and [Modifying a Change Approval Object](#) and [Viewing a Change Approval Object](#), later in this chapter.

Approving or Rejecting a Change

If you are listed as a change approver for a requirement or a building block, you receive an E-mail message requesting your approval when a change to the object is submitted. You gain access to the change through Microsoft Internet Explorer. You have that access until you approve or reject the request. For more information, see [Submitting Changes for Approval](#), earlier in this chapter.

In the browser, you can review the content of the proposed change with or without running the Architect/Requirements client. Also, you can view the current information for the change approval object that is attached to the object submitted for approval. For more information, see [Viewing a Change Approval Object](#), later in this chapter.

To approve or reject a change:

1.

To gain access to the change in Internet Explorer, do either of the following:

- In the browser, do the following:

- . Open the Architect/Requirements home page, and then click the **Change Approval** hyperlink.



If the Architect/Requirements client is not running, either the Architect/Requirements log in page or the Teamcenter Login page is displayed. Enter your Architect/Requirements or Security Services user name and password, select a language, and click **Log in**.

The browser displays the Changes Waiting for Approval page for your user name.

- . In the **Select from available projects** field, select the project that contains the change.

The page displays a table of changes waiting for your approval.

- . In the row that contains the change, click the **Change Status** button.

The browser displays the Approve / Reject page for the change.

- In the E-mail change approval request, click the hyperlink indicated to approve or reject and provide comments.

The browser displays the Approve / Reject page for the change.

On the Approve / Reject page, you can do the following:

- Click the **Proposed Change** button to open the change content in an **.mhtml** file for review.
- Click the **View Comments** button to open a new browser window containing the current information in the change approval object.
- Enter plain text comments in the **Fill in your comments to approve/reject the change** field.

2. Do one of the following:

- To approve the change, click the **Approved** button.

The browser displays your Changes Waiting for Approval page with the change removed from the table. In the client, a **Response** event indicating your approval is added to the change approval object, which can be viewed in the **Attachments** tab or window.



If yours is the last outstanding approval, an **Approved** event is added to the change approval object. The change approval process is concluded, and the approved object remains frozen.

An E-mail message is sent to each approver and to the submitter, stating that the change is approved by all approvers. Two **.mhtml** files are attached to this message. One attachment contains the approved content, and the other contains the table from the change approval object. The recipient can click the link in the message to navigate to the approved object in the client.

- To reject the change, click the **Rejected** button.

The browser displays your Changes Waiting for Approval page with the change removed from the table. In the client, a **Response** event indicating your rejection is added to the change approval object, which can be viewed in the **Attachments** tab or window.

An E-mail message is sent to the submitter, stating that the change is rejected by your user name. Two **.mhtml** files are attached to this message. One attachment contains the rejected content, and the other contains the table from the change approval object. The submitter can click the link in the message to navigate to the rejected object in the client.



If yours is the first rejection, the rejected object is unfrozen and remains in that state.

Modifying a Change Approval Object

In the **Attachments** tab, a change approval object is automatically created for a requirement or a building block that is submitted for approval. The change approval object contains a table in which the rows show all responses that are currently received. Each row shows the time of response, the responding user name, the user's comments, and the response type.

If you are a change approver for a submitted requirement or building block, you can rename the change approval object, and you can edit the object's content in Microsoft Word as you edit requirement content.



To edit the content of the change approval object, Microsoft Office Word 2013 or Microsoft Office Word 2016 must be installed on your computer.

To modify a change approval object:

1. Do one of the following:

- For a requirement or a building block in the hierarchical content table, select the object, and then click the **Attachments** tab to display the change approval object. You can

open the **Attachments** window for this requirement or building block by clicking the **Open tab** button on the notebook pane's toolbar.

- For a requirement or a building block in the **Links** or **Versions** tab:
 - With the **Attachments** tab on top in the notebook pane, click the **Open tab** button on the toolbar to open the **Attachments** window.
 - In the **Links** or **Versions** tab, select the requirement or building block to display its change approval object in the **Attachments** window.
- 2. In the **Attachments** tab or window, select the change approval object, and then do one or both of the following:
 - To rename the object, pull down the **File** menu and choose **Rename**, enter the new name in the open text field, and press the enter key.

You can also right-click the object and choose **Rename** from the pop-up menu, or press the F2 key.
 - To edit the content:
 - Pull down the **File** menu and choose **Open**.

You can also right-click the object and choose **Open** from the pop-up menu, or double-click the object.

The change approval object opens in Word as an **.mhtml** file, in which you can enter and change content within the table of responses and anywhere outside the table. This file is temporary and is deleted from your computer when you exit Architect/Requirements.
 - To save the content in the database, click Word's **Office Button** and select **Save**.

Viewing a Change Approval Object

The **Attachments** tab and floating window display the change approval object for the requirement or building block selected in the hierarchical content table. For the requirement or building block selected in the **Links** or **Versions** tab, the change approval object is displayed in the **Attachments** window.

For the change approval object selected in the **Attachments** tab or window, you can view the content in two ways:

- In a read-only Microsoft Word file, where you can view and print the content and send it to E-mail and fax recipients. You must have **Project Administrator** privilege to change the content in the database.



Microsoft Office Word 2013 or Microsoft Office Word 2016 must be installed on your computer.

- In the **Preview** window, where you can view and print the content without opening the change approval object in Word. You cannot change the content in the database. For more information, see [Preview Tab](#) in chapter 3, *Using the Architect/Requirements Main Window*.

To view a change approval object in Microsoft Word:

1. Do one of the following:
 - For a requirement or a building block in the hierarchical content table, select the object, and then click the **Attachments** tab to display the change approval object for the selected requirement or building block. You can open the **Attachments** window for this requirement or building block by clicking the **Open tab** button on the notebook pane's toolbar.
 - For a requirement or a building block in the **Links** or **Versions** tab:
 - . With the **Attachments** tab on top in the notebook pane, click the **Open tab** button on the toolbar to open the **Attachments** window.
 - . In the **Links** or **Versions** tab, select the requirement or building block to display its change approval object in the **Attachments** window.
2. In the **Attachments** tab or window, select the change approval object, and then pull down the **File** menu and choose **Open Read-Only**.

You can also right-click the change approval object and choose **Open Read-Only** from the pop-up menu.

The change approval object opens in Word as an **.mhtml** file. This file is deleted from your computer when you exit Architect/Requirements.

To view a change approval object in the Preview window:

For a requirement or a building block in the hierarchical content table:

1. Select the requirement or building block, and then click the **Attachments** tab to display the change approval object.
You can open the **Attachments** window for this requirement or building block by clicking the **Open tab** button on the notebook pane's toolbar.
2. With the **Preview** tab on top in the notebook pane, click the **Open tab** button on the toolbar to open the **Preview** window.
3. In the **Attachments** tab or window, select the change approval object to display its content in the **Preview** window.

For a requirement or a building block in the **Links** or **Versions** tab:

1. With the **Attachments** tab on top in the notebook pane, click the **Open tab** button on the toolbar to open the **Attachments** window.
2. In the **Links** or **Versions** tab, select the requirement or building block to display its change approval object in the **Attachments** window.
3. With the **Preview** tab on top, click the **Open tab** button to open the **Preview** window.
4. In the **Attachments** window, select the change approval object to display its content in the **Preview** window.

Viewing a Change Log

A change log captures the history of modifications to an object of a given type definition. An object's change log contains a table in which the rows show all change events that are currently recorded. Each row shows the date and time of change, the user who made the change, and the type of change. For changes to object properties, the log also shows the previous and new values.



The previous and new values of requirement and note text are not shown because their text size and rich content can exceed the constraints of the tabular layout of the change log.

You can also compare the content of the requirement. For more information, see [Comparing the Content of Different Requirements](#).

Change logs may apply to folders, requirements, building blocks, groups, notes, trace links, and connections. For each type definition, your project administrator can specify change events that are tracked by a change log. The change log is automatically created for an object of that type in the Systems Engineering and Requirements Management module when a specified change event occurs. A row is added to the change log for each later change event for that object.

The **Attachments** tab and floating window display the change log for the object selected in the hierarchical content table. For the object selected in the **Links** or **Versions** tab, the change log is displayed in the **Attachments** window.

For the change log selected in the **Attachments** tab or window, you can view the content in two ways:

- In a read-only Microsoft Word file, where you can view and print the content and send it to E-mail and fax recipients. You must have **Project Administrator** privilege to change the content in the database.



Microsoft Office Word 2013 or Microsoft Office Word 2016 must be installed on your computer.

- In the **Preview** window, where you can view and print the content without opening the change log in Word. You cannot change the content in the database. For more information, see [Preview Tab](#) in chapter 3, *Using the Architect/Requirements Main Window*.

To view a change log in Microsoft Word:

1. Do one of the following:
 - For a requirement or a building block in the hierarchical content table, select the object, and then click the **Attachments** tab to display the change log for the selected requirement or building block. You can open the **Attachments** window for this requirement or building block by clicking the **Open tab** button on the notebook pane's toolbar.
 - For a requirement or a building block in the **Links** or **Versions** tab:
 - . With the **Attachments** tab on top in the notebook pane, click the **Open tab** button on the toolbar to open the **Attachments** window.
 - . In the **Links** or **Versions** tab, select the requirement or building block to display its change log in the **Attachments** window.

2. In the **Attachments** tab or window, select the change log, and then pull down the **File** menu and choose **Open Read-Only**.

You can also right-click the change log and choose **Open Read-Only** from the pop-up menu.

The change log opens in Word as an **.mhtml** file. This file is deleted from your computer when you exit Architect/Requirements.

To view a change log in the Preview window:

For a requirement or a building block in the hierarchical content table:

1. Select the requirement or building block, and then click the **Attachments** tab to display the change log.
You can open the **Attachments** window for this requirement or building block by clicking the **Open tab** button on the notebook pane's toolbar.
2. With the **Preview** tab on top in the notebook pane, click the **Open tab** button on the toolbar to open the **Preview** window.
3. In the **Attachments** tab or window, select the change log to display its content in the **Preview** window.

For a requirement or a building block in the **Links** or **Versions** tab:

1. With the **Attachments** tab on top in the notebook pane, click the **Open tab** button on the toolbar to open the **Attachments** window.
2. In the **Links** or **Versions** tab, select the requirement or building block to display its change log in the **Attachments** window.
3. With the **Preview** tab on top, click the **Open tab** button to open the **Preview** window.
4. In the **Attachments** window, select the change log to display its content in the **Preview** window.

Appendix A: Glossary

This appendix defines Architect/Requirements terms.

B

Building Block

Object that can represent any element of a hierarchy, such as a function or a component of a product, a task in a work breakdown structure, or a job function in an organizational chart. Building blocks allow you to construct hierarchies that decompose systems and illustrate relationships among system elements.

Below each top level building block, subordinate building blocks can be organized in multiple levels of parents, children, and siblings. A system-defined building block subtype, **TRAM**, can be used for transitional mapping, a method of interrelating system views for comparison and analysis. To illustrate these relationships graphically, diagrams can be used in conjunction with building blocks. See also *Diagram*, *Complying Object*, *Defining Object*, *Full Block Number*, and *TRAM*.

C

Child

Object that is subordinate to a higher level object in a folder hierarchy. Compare with *Parent* and *Sibling*. See also *Direct Child* and *Member*.

Circular Reference

Defining trace link from a complying object back to its defining object. Circular references are allowed only if the defining and complying trace links are of different subtypes. See also *Complying Object*, *Defining Object*, and *Trace Link*.

Complying Object

Object that partially or completely fulfills a condition specified by a defining object. Trace links from complying objects can be traced up through the hierarchy to one or more originating sources. Compare with *Defining Object*.

D

Defining Object

Object that specifies a condition that a product or component must fulfill. Trace links from defining objects can be traced down through the hierarchy to the lowest level requirement that meets the original condition. Compare with *Complying Object*.

Derived Requirement

Requirement that has been translated into parameters suitable for lower level analysis and design. The inputs to the derivation process are higher level requirements. The outputs are new requirements.

Diagram

Object that graphically illustrates relationships for the object to which the diagram is attached. Typically, diagrams are attached to building blocks, which can represent superior and subordinate elements of a system view. Diagrams also can be attached to folders, requirements, and groups.

Diagrams are created and edited through live Visio, the Architect/Requirements interface with Microsoft Office Visio. Each live Visio diagram is interactively synchronized with the Architect/Requirements database, and is associated with a special stencil that contains shapes representing the Architect/Requirements object types. Objects can be created, modified, and deleted in the database by adding, modifying, and deleting the corresponding shapes in the diagram. The diagram is updated dynamically when members of the diagram owner are created, modified, and deleted in the database. See also *Building Block*. Compare with *Note* and *Spreadsheet*.

Direct Child

Child that occupies the level directly below the parent in a folder hierarchy. Compare with *Parent* and *Sibling*. See also *Child* and *Member*.

E

Editable Properties

System-defined or user-defined properties whose values can be changed directly by the user, subject to the permissions of individual users. Compare with *Read-Only Properties*.

F

Folder

Object that occupies the top level of organization within a project hierarchy. As folders contain files and other folders in Microsoft Windows, folders contain requirements, building blocks, groups, and other folders in Architect/Requirements.

Full Block Number

System-defined property that Architect/Requirements assigns to each building block. The full block number indicates the building block's level within a hierarchy of building blocks in a folder. The current level and full block number can be changed by promoting or demoting the building block, or by manually changing its **Number** property.

G

Group

Collection of references to existing objects that reside elsewhere in an Architect/Requirements project. Each reference associates one object as a *member* of the group. The referenced objects remain in their present locations, and can be maintained directly from the group without the need to switch to other folders or views. See also *Member*.

I

Indicator

Special graphic that symbolizes certain information about an object. There are four types of indicators: object type indicators, attachments indicators, links indicators, and versions indicators.

M

Member

Subordinate object occupying the next level below the immediate superior. Members of a folder, a requirement, or a building block reside in the superior object's location as direct children. Members of a group reside elsewhere in an Architect/Requirements project, retaining existing relationships to parents in those locations without becoming children of the group. The **Member Count** property shows the number of members for a selected object. See also *Direct Child* and *Parent*.

N

Note

Object that can be attached to folders, requirements, building blocks, and groups. Notes are useful for recording information that relates only to certain objects, rather than to all objects of a type. The content is created and edited in Microsoft Word. Compare with *Diagram* and *Spreadsheet*.

O

Object

Data element in an Architect/Requirements project. The objects types in the Systems Engineering and Requirements Management module are folders, requirements, building blocks, groups, notes, diagrams, and trace links.

P

Paragraph Number

System-defined property that Architect/Requirements assigns to each requirement. The paragraph number indicates the requirement's level within a hierarchy of requirements in a folder. The current level and paragraph number can be changed by promoting or demoting the requirement, or by manually changing its **Number** property.

Parent

Object to which one or more other objects are subordinate at lower levels of a folder hierarchy. Compare with *Child* and *Sibling*. See also *Direct Child* and *Member*.

Project

Object at the highest level of organization in Architect/Requirements, superior to all other objects. Projects define boundaries for user access, user customization of object types and properties, and organization of the other objects.

R

Read-Only Properties

System-defined properties whose values cannot be changed directly by the user. Compare with *Editable Properties*.

Requirement

Object whose content specifies one or more conditions that must be fulfilled by a system or a component. Requirement content also specifies the method by which the requirement is fulfilled. The content is created and edited in Microsoft Word. See also *Paragraph Number*, *Complying Object* and *Defining Object*.

Root Node

Topmost node in the navigation tree. Whether the root node is enabled or disabled, it always occupies the highest level in the tree. Below the root node, the navigation tree contains a node for each project to which you have access.

S

Sibling

Object that is subordinate to the same parent, and occupies the same level, as one or more other objects in a folder hierarchy. Compare with *Child* and *Parent*. See also *Direct Child* and *Member*.

Spreadsheet

Object that stores equations in the Architect/Requirements database. Spreadsheet objects can be created from worksheets in existing Microsoft Excel files. Equations in worksheet cells are preserved when the spreadsheet is opened, rather than being overwritten by updated property values from the database. Spreadsheets can be attached to folders, requirements, building blocks, and groups. Equations and other content can be edited in Excel. Compare with *Diagram* and *Note*.

Style Sheet

Formatting information specified when objects in an Architect/Requirements project are exported to a Microsoft Word document. For each exported object type, Architect/Requirements references the specified style sheet and applies that formatting in the export document.

System-Defined Properties

Properties that Architect/Requirements assigns automatically to built-in object types. Some system-defined properties receive a value that cannot be changed directly by the user, such as the object's creation date and time. Other system-defined properties, such as the object's name, receive a default value, which can be changed directly by the user. Compare with *User-Defined Properties*.

T

Trace Link

Object that establishes a directional relationship between two other objects, and indicates which object precedes, or defines, the other in the relationship. Defining and complying trace links can be created between objects that reside within the same Architect/Requirements project, and also can be created between objects that reside in different projects. Defining trace links can be created from objects in Architect/Requirements to complying objects in other Teamcenter products.

Trace links cannot be created from one trace link to another. See also *Complying Object* and *Defining Object*.

TRAM

System-defined building block subtype that can be used for transitional mapping, a method of interrelating system views for comparison and analysis. Through trace links, a TRAM can be associated with building block hierarchies to create a flow of information among source views and destination views, with the TRAM as the focal point. See also *Building Block* and *Trace Link*.

U

Uniform Resource Locator (URL)

Address that locates a specific resource on the Internet or on an intranet.

User-Defined Properties

Properties that an Architect/Requirements project administrator assigns to built-in object types and to custom subtypes. All user-defined properties are editable properties. That is, their values can be changed directly by the user, subject to the permissions of individual users. Compare with *System-Defined Properties*.

Appendix B: System-Defined Properties in the Systems Engineering and Requirements Management Module

This chapter describes the system-defined properties of the object types used in the Systems Engineering and Requirements Management module.

Overview of System-Defined Properties

Architect/Requirements automatically assigns system-defined properties to all built-in object types used in the Systems Engineering and Requirements Management module:

- *Folders*
- *Requirements*
- *Building blocks*
- *Notes*
- *Trace links*
- *Groups*

Each system-defined property falls into one of two categories:

- A *read-only* property has a value that cannot be changed, except as an indirect result of certain user actions on an object.
- An *editable* property has a default value that can be changed directly by the user.

For more information, see [Working With Object Properties](#).

Table of System-Defined Properties

The **Properties** tab and floating window display all viewable system-defined properties for the object selected in the content table. Other views in the Systems Engineering and Requirements Management module display certain system-defined properties by default. In these views, default properties can be removed and other system-defined properties can be added. For more information, see [Properties Tab](#) and [Using Tabs in Floating Windows](#) in chapter 3, *Using the Architect/Requirements Main Window*, and [Adding and Removing Columns](#) in chapter 9, *Working With Object Properties*.

Table B-1 lists each system-defined property, including the object types to which it applies and its description.

Table B-1. System-Defined Properties in Systems Engineering and Requirements Management Module

Property	Applies To	Description
Approval Status	Requirement	<p>Current stage of the change approval process for this requirement. This property is read-only. Values are:</p> <ul style="list-style-type: none"> • Change Approved - the changes are approved by all approvers. • Change Rejected - the changes are rejected by at least one approver. • None - no changes are submitted. • Pending - changes are submitted and at least one approver response is outstanding.
Attachment Count	All object types	<p>In the Properties tab and window and the Edit Properties dialog window, the number of notes, diagrams, spreadsheets, and change approval objects attached to the selected object. This property is read-only.</p> <p>In other client views, contains a graphic indicator if the object has one or more notes, diagrams, spreadsheets, or change approval objects. To see the number of attachments, rest the pointer on the indicator to display a tooltip. You can click an indicator to see these objects in the Attachments tab or window.</p>
Baseline	Building block, Requirement	Names of the baselines associated with the object. This property is read-only.
Change Approval Pending	Change Approval	Indicates if a change approval is pending for the object to which the change approval object is attached. This property is read-only.
Change Approvers	Change Approval	List of users who have the responsibility to approve or reject the changes. This property is read-only.

Table B-1. System-Defined Properties in Systems Engineering and Requirements Management Module

Property	Applies To	Description
Change Notifiers	Change Approval	List of users who are notified of submitted changes for informational purposes. Change notifiers do not approve or reject the changes. This property is read-only.
Change Rejected	Change Approval	Indicates if a change is rejected for the object to which the change approval object is attached. This property is read-only.
Change Time	All object types	<p>Date and time when the object was last modified. This property is read-only. For any object type, the property is updated when:</p> <ul style="list-style-type: none"> ● A defining or complying trace link is created on the object or is deleted from the object. ● The object is deleted, thus moving to the Recycle Bin of the deleting user. ● The object is restored from the Recycle Bin. <p>Additional change events depend on the object type:</p> <ul style="list-style-type: none"> ● For a requirement, the property is updated when the requirement content is edited, when the requirement is frozen, and when it is unfrozen. ● For a note, the property is updated when the note content is edited. ● For a building block, the property is updated when the building block is frozen and when it is unfrozen. ● For a folder, the property is updated when a member is added or deleted. <p>For additional details on when the Change Time property is updated, see the <i>Updates to Change Time and Change User Properties</i> topic in chapter 2 of the <i>Systems Architect/Requirements Management Project Administrator's Manual</i>.</p>
Change User	All object types	Login name of the user who last modified the object. Change events for this property are the same as those described above for the Change Time property. This property is read-only.
Complying Objects	Defining objects of all types	ROIN of each requirement, name of each building block, and OID of each object in another Teamcenter product that complies with the selected defining object. This property is read-only.

Table B-1. System-Defined Properties in Systems Engineering and Requirements Management Module

Property	Applies To	Description
Copy Snapshot	Diagram	<p>Controls the behavior of the diagram image. Values are:</p> <ul style="list-style-type: none"> ● No - The GIF image in the diagram image note is not updated even if the diagram is changed and saved. This is the default value. ● Yes - The GIF image in the diagram image note is updated when the diagram is saved. <p>This property is editable.</p>
Create Time	All object types	Date and time when the object was created. This property is read-only.
Create User	All object types	Login name of the user who created the object. This property is read-only.
Date Deleted	Deleted objects of all types in Recycle Bin	Date and time when the object was deleted. This property is read-only.
Defining Objects	Complying objects of all types	ROIN of each requirement, and name of each object of all other types, that defines the selected complying object. This property is read-only.
Diagram Content	Diagram	<p>Controls the behavior of the shapes that represent the members of the live Visio diagram owner. This property is editable. Values are the following:</p> <ul style="list-style-type: none"> ● Members fully synchronizes the set of shapes with the member set. As members are added to the owner and moved to other owners, shapes are added and removed. <p>A Members diagram is changed to Static when a shape is added for a non-member object or when a member shape is removed.</p> <ul style="list-style-type: none"> ● Static prevents the automatic addition and removal of shapes as the member set changes. When members are added to the owner and moved to other owners, the set of shapes is unchanged. <p>The diagram remains connected to the database. Shapes for new members can be manually added to the diagram. Shapes automatically reflect changes to corresponding object properties.</p>

Table B-1. System-Defined Properties in Systems Engineering and Requirements Management Module

Property	Applies To	Description
		Any shape can be manually removed. The corresponding object is not deleted. For both values, shapes are automatically deleted when corresponding objects are deleted in the client.
Direction	Connection	Direction of the connection. Values are Non-Directional , Uni-Directional , and Bi-Directional . This property is editable.
Document Template	Folder	Document template used when objects in the folder are exported to Microsoft Word. The document template controls the export process through object templates, which determine the data exported for the objects, and a style sheet, which determines the Word formatting applied to the data. This property is editable.
Expiry Date	Change Approval	Date after which the changes are automatically approved if one or more change approvers have not responded. This property is editable.
Full Name	All object types	Complete path to the object, beginning with the project name. This property is read-only.
GUID	Folder, Requirement, Building block, Group	Global Unique Identifier of the Teamcenter product containing an object that is linked to the selected object. This property is read-only.
Image Height	Note (Image Note subtype)	Height, in points, of the GIF image in the diagram image note. This property is editable.
Image Scale	Note (Image Note subtype)	Indicates whether the GIF image in the diagram image note is scaled from the original size. Values are No , the default, and Yes . If Yes , the Image Height and Image Width properties show the dimensions. This property is editable.
Image Width	Note (Image Note subtype)	Width, in points, of the GIF image in the diagram image note. This property is editable.
Member Count	Folder, Requirement, Building block, Group	Number of subordinate objects occupying the next lower level below the selected object. This property is read-only.
Name	All object types	Name of the object. This property is editable.

Table B-1. System-Defined Properties in Systems Engineering and Requirements Management Module

Property	Applies To	Description
Number	Requirement, Building block	Current paragraph number of the requirement, or current full block number of the building block. Conforming to the numbered outline style, the value indicates the object's level in the hierarchy and its relationships to parent, sibling, and child objects. This property is editable.
OID	Folder, Requirement, Building block, Group	Object Identifier of an object (in another Teamcenter product) that is linked to the selected object. This property is read-only.
Original Location	Deleted objects of all types in Recycle Bin	Complete path from which the object was deleted. This property is read-only.
Project	All object types	Name of the project that contains the object. This property is read-only.
Reserved By	All object types	Login name of the user who is currently modifying the object. No other user can modify the object until the current change user releases the reservation by completing modifications. This property is read-only.
ROIN	Requirement	Requirement Object Identification Number of the requirement. Each ROIN is unique within the project This property is read-only.
Security Profile	All object types	Access rules that specify which users can view, modify, and delete the object. This property is editable.
Source Filename	Requirement	Name of the imported document from which the requirement was generated. This property is editable.
Source Paragraph	Requirement	In the imported document, the number of the paragraph that generated the requirement. This property is read-only.
Static Date	Building Block, Requirement	Date and time when the object was frozen. This property is read-only.
Stencils	Diagram	Live Visio stencils associated with the diagram. This property is editable.
Subtype	All object types	User-defined subtype assigned to the object. If no subtype is assigned, this property has the same value as

Table B-1. System-Defined Properties in Systems Engineering and Requirements Management Module

Property	Applies To	Description
		the Type property for the object. This property is editable.
Synergy Check-In Comment	Requirement	Plain text information about a requirement that is checked in to IBM Synergy. The default comment is Check In from TeSE . This property is editable.
SynergyID	Requirement	Full path to a requirement in a IBM Synergy project. This property is read-only.
SynergyTask	Requirement	IBM Synergy task to which a checked out requirement is assigned. This property is editable.
Text	Requirement, Note	Plain text contained in the requirement or note. This property is editable.
Text Format	Requirement, Note	Format of the requirement or note content. This property is editable. Values are the following: <ul style="list-style-type: none"> ● Text, for plain text content. ● HTML, for rich text content. ● MHTML, for content entered in Microsoft Word.
Trace Link Count	Folder, Requirement, Building block, Group	In the Properties tab and window and the Edit Properties dialog window, the number of objects connected to the selected object by a trace link. This property is read-only. In other client views, contains a graphic indicator if the object has one or more trace links to other objects. To see the number of trace links, rest the pointer on the indicator to display a tooltip. You can click an indicator to see the linked objects and trace links in the Links tab or window.
Type Name	All object types	Built-in object type. This property is read-only.
Version Count	Building block, Requirement	In the Properties tab and window and the Edit Properties dialog window, the number of versions of the selected object. This property is read-only. In other client views, contains a graphic indicator if the object has one or more versions or variants. To see the number of versions or variants, rest the pointer on the indicator to display a tooltip. You can click an indicator

Table B-1. System-Defined Properties in Systems Engineering and Requirements Management Module

Property	Applies To	Description
Version Number	Building block, Requirement	to see the versions and variants in the Versions tab or window. Version number of this object. This property is read-only.
Version Type	Building block, Requirement	Indicates whether the object is frozen or a variant. Values are Static , for a frozen object, and Variant . An object can have both values. This property is read-only.
WhereUsed Object Count	All	<p>In the Properties tab and window and the Edit Properties dialog window, the number of times the object is referenced by another object. This property is read-only.</p> <p> -1 is displayed for certain objects, where calculating the correct value would require too much time. However, the Where Used tab or window always displays the correct information for these objects.</p> <p>In other client views, contains a graphic indicator if the object is referenced by one or more other objects. To see the number of referencing objects, rest the pointer on the indicator to display a tooltip. You can click an indicator to see the referencing objects in the Where Used tab or window.</p>

Appendix C: Live Office Interface

Frequently Asked Questions

This chapter contains a list of Live Office Interface FAQs.

General

What are the prerequisites to use the Live Office Interface?

- Microsoft Office Office 2013 or Office 2016.
- Microsoft Office Visio 2013 or Visio 2016.
- The Microsoft .NET Framework versions 4.0 CLR.

Mixed installation of the Microsoft Excel 2013 and Microsoft Excel 2016 is not supported. When more than one version is installed, there is no reliable way for the Architect/Requirements client to launch one specific version. If you need to install Office 2016, you must install Excel 2013 on a different computer. You can also use a virtual machine for installing other versions of Microsoft Office.

For more information, see [Prerequisites for Using the Live Office Interface](#) in the chapter 2, *Installing the Architect/Requirements Client with Office Integration*.

Can I install Microsoft Office 2013, and Office 2016 on the same machine?

Yes. However, the live Office interface may become unstable. Microsoft recommends that you should install only one version of Office on your machine.

Mixed installation of the Microsoft Office Excel 2013 and Microsoft Office Excel 2016 is not supported. When more than one version is installed, there is no reliable way for the Architect/Requirements client to launch a specific version. If you need to install Microsoft Office 2016, then Excel 2013 must be available on a separate computer, or you can use the virtual machine for the installations.

What are the Architect/Requirements default addins?

Following addins are installed when you install Architect/Requirements.

- **TcROffice2K3Addin.Connect Friendly Name**
- **TcRExcelImport2K3Addin.Connect Friendly Name**

Active only in Microsoft Excel.

- **TcRVisioAddin.Connect Friendly Name**

Active only in Microsoft Visio.



Errors have been observed when these addins conflict with other custom addins. You must disable these addins to resolve the errors.

How do I disable Architect/Requirements addins in Microsoft Office application?

1. Launch the Microsoft Office application for which you want to disable the addin. For example, Microsoft Word, Microsoft Excel, or Microsoft Visio.
2. Click **File**.
3. Click **Options**.
4. In the **Options** dialog, click **Add-Ins** in the left pane.
5. Select **COM Add-ins** from the **Manage** drop down list and click **Go**.
6. In the **COM Add-Ins** dialog, clear the check box for the Architect/Requirements addin.
For information on the default Architect/Requirements addins, see [What are the Architect/Requirements default addins?](#)
7. Click **OK**.

What are the registry entries created by Architect/Requirements for the Live Office Interface?

- Entry for the Microsoft Office Excel on either 32-bit or 64-bit Windows:

```
HKEY_CURRENT_USER\Software\Microsoft\Office\Excel\Addins\  
TcROffice2K3Addin.Connect  
LoadBehavior = 3
```

```
HKEY_CURRENT_USER\Software\Microsoft\Office\Excel\Addins\  
TcRExcelImport2K3Addin.Connect  
LoadBehavior =3
```

- Entry for the Microsoft Office Word on either 32-bit or 64-bit Windows:

```
HKEY_CURRENT_USER\Software\Microsoft\Office\Word\Addins\  
TcROffice2K3Addin.Connect  
LoadBehavior =3
```

- Entry for the Microsoft Office Visio on 32-bit Windows:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Visio\Addins\  
TcRVisioAddin.Connect  
LoadBehavior =3
```

- Entry for the Microsoft Office Visio on 64-bit Windows:

```
HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\Microsoft\Visio\Addins\  
TcRVisioAddin.Connect  
LoadBehavior =3
```



The value of the **LoadBehavior** registry entry must be set to 3 for the live office interface to work properly.

What should I do if the Architect/Requirements Live Office Interface does not work on my machine after I install the latest version of the Architect/Requirements client?

OR

How to fix the communication error message when I export to live Excel from Architect/Requirements?

Run the Architect/Requirements **Live Office Interface Diagnostic Tool** to diagnose any issues.

For more information, see [Using the Live Office Diagnosis](#) in the chapter 2, *Installing the Architect/Requirements Client with Office Integration*.

If the **Live Office Interface Diagnostic Tool** reports that the Office add-ins are not registered on your system, then go to your client installation folder and run the **register_office.bat** and **register_visio.bat** batch files.



To run the **register_office.bat** and **register_visio.bat** batch files, the supported version of Microsoft Office and Microsoft Office Visio must be installed on your system.

What should be the security settings in the Microsoft Office Excel, Microsoft Office Word, and Microsoft Office Visio in order to use the Architect/Requirements Live Office Interface?

- Configure Microsoft Word to use the following settings:

Macro Settings: Choose the **Disable all macros except digitally signed macros** option.

Add-ins: Check the **Require Application Add-ins to be signed by Trusted Publisher** check box. The unsigned add-ins present on the system will not work in this case.

For more information, see [Configuring Microsoft Office for the Live Office Interface](#) in the chapter 2, *Installing the Architect/Requirements Client with Office Integration*.

Where can I find the log file for Live Office Interface?

The log file for the live Office interface captures errors encountered while using any of the live Office interface features. It helps in identifying and troubleshooting any problems.

The log file can also capture trace, which enables logging every Architect/Requirements communication in addition to errors. You can enable logging by setting **Trace = True** in the **Trace.txt** file available at *C:\Documents and Settings\%User Name%\Application Data\TcRequirements*.

- The live Office interface log file is located at:

C:\Users\%User Name%\AppData\Roaming\ TcRequirements\OfficeInterface_log.txt.

Microsoft Office Word Troubleshooting

What do I do if the formatting is lost while exporting requirements to Microsoft Word?

Bullets and Numbering formatting is lost when a requirement is exported from Architect/Requirements to Microsoft Word. The problem occurs when you format the text directly instead of formatting the text using the styles from the style sheet attached to the document. The direct formatting is also lost when you apply the **AutoNumbering** and **AutoBullet** styles or on saving the objects as Microsoft Word tags the bullets and the numbers as same. To preserve the bullets and numbering, apply styles containing the **Bullet** or **Number** format. This allows preserving the formatting in the style sheet.

Bullets format applied to a document are converted to numbers if both of the following conditions are true:

- The text of the requirement includes automatic bullets.
- The style sheet uses automatic numbering for the headings.



Note that the bullets are displayed correctly in the **Preview** tab in the **Notebook** pane when the folder's document template is set to the one used for export.

How to fix problems Importing Microsoft Word Document?

If you have moved from Microsoft Office 2010 to Microsoft Office 2013 or a later version and are getting errors in importing Word documents using keyword parsing, you must clear the Microsoft Form cache by running the diagnostic tool for Word.

To clear the Microsoft Form cache

1. Open the Architect/Requirements application URL.
2. Click the **Administrative Tools** link.
3. Click the **Diagnostic Tools** link.
4. Click the **Live Office Diagnosis** link.
5. In the **TcSE Office Interface Diagnostic Tool**, unselect all check boxes and select **Word** only.
6. Click **Diagnose**.
7. Click **No** to prevent launching Word.

The **Word Diagnostic Information** in the **TcSE Office Interface Diagnostic Tool** window displays a log of the diagnosis performed.

8. Click the **Clear cache** link.
9. Click **Close**.

Microsoft Office Excel Troubleshooting

What should I do if the live Excel functionality does not work on my system? The diagnostic tool does not suggest any errors. The security and registry settings are correct.

Verify that the supported versions of Microsoft Office and Microsoft Office Visio are installed on your system.

Mixed installation of the Microsoft Office Excel 2013 and Excel 2016 is not supported. When more than one version is installed, there is no reliable way for the Architect/Requirements client to launch one specific version. If you are required to install the Microsoft Office 2016, then Excel 2013 should be available on a separate system. You can also use a virtual machine for the other versions.

How to fix the missing Architect/Requirements toolbar in Microsoft Office Excel?

OR

How to view the Architect/Requirements menu when I open a saved live Excel workbook?

OR

What should I do if the Microsoft Office Excel Add-ins do not load after I open a saved live Excel workbook?

To fix any of the above issues, follow the steps below:

1. Launch Microsoft Excel and open a live Excel file.
2. Check the security settings of Microsoft Office Excel.

Configure Microsoft Excel to use the following settings:

Macro Settings: Choose the **Disable all macros except digitally signed macros** option.

Add-ins: Check the **Require Application Add-ins to be signed by Trusted Publisher** check box. The unsigned add-ins present on the system will not work in this case.

For more information, see [Configuring Microsoft Office for the Live Office Interface](#) in the chapter 2, *Installing the Architect/Requirements Client with Office Integration*.

3. Kill any Microsoft Office Excel processes running in the background using the **Task Manager**. Reopen the work book as mentioned in step 1.

To diagnose any further issues, you may run the Architect/Requirements **Live Office Interface Diagnostic Tool**.

For more information, see [Using the Live Office Diagnosis](#) in the chapter 2, *Installing the Architect/Requirements Client with Office Integration*.

I made changes in the saved independent live Excel workbook after I connected to the Architect/Requirements server. I cannot see these changes in the Architect/Requirements client. What could be the reason?

Refresh the Architect/Requirements client to synchronize and view the changes from the Microsoft Office Excel workbook.

In what format should I save my live Excel workbook to retain the live behavior after I open the saved live Excel sheet?

- Save in the **XLSM** format.

I edited a requirement (a Microsoft Office Word document) from a live Excel workbook using the Architect/Requirements Open menu in the live Excel sheet. However, the changes are not reflecting in the Architect/Requirements client. What could be the issue?

Run the Architect/Requirements **Live Office Interface Diagnostic Tool** to diagnose the live Word interface. Ensure you are connected to Architect/Requirements.

For more information, see [Using the Live Office Diagnosis](#) in the chapter 2, *Installing the Architect/Requirements Client with Office Integration*.

Microsoft Office Visio Troubleshooting

I open a saved live Visio diagram from my local disk, connected to Architect/Requirements and changed the property of object, but I cannot see the property value updated in Architect/Requirements client. What could be the issue?

Refresh the Architect/Requirements client to synchronize and view the changes.

What could be the issue if adding a master shape in live Visio diagram does not create type or subtype in Architect/Requirements?

- Check the mapping file for the type or subtype mapping.
For more information on customizing the interface with Microsoft Office, see the *Systems Architect/Requirements Management Project Administrator's Manual*.
- Run the **Stencil Diagnostic Utility** available as a command when you right-click on a live Visio diagram. You can identify the subshape, diagnose the connection point, verify the property mapping file, and verify the stencil.

For more information, see [Stencil Diagnostic Utility](#) in the chapter 6, *Constructing System Views With Building Blocks and Diagrams*.

What should I do if I cannot open at least one stencil object when I try to create a new live Visio diagram?

You must register the type library on your system again:

1. Locate the **regtlib.exe** file on your machine. Usually, it is located at `C:\WINDOWS\system32\URTTemp`.
2. Locate the **vislib.dll** file on your system.
The **vislib.dll** file is usually located at `C:\Program Files\Microsoft Office\Office12(OR 11)\vislib.dll`.
3. Open a command prompt and change the directory to the path of the **regtlib.exe** file.
4. Run the **regtlib C:\Program Files\Microsoft Office\Office12(OR 11)\vislib.dll** command. Change the path of the file if it is different.

How can I map more than one Architect/Requirements property to a Microsoft Office Visio master shape?

You must create a group master shape that has as many shapes as the number of properties you want to map. You must know the sequence in which you have added the shapes to the group shape. In property mapping file, you can map various properties with each of the sub shape of the group shape.



You can use the **Identify Subshape** command to identify the subshape sequence.

For more information, see [Stencil Diagnostic Utility](#) in the chapter 6, *Constructing System Views With Building Blocks and Diagrams*.

How can I identify the sequence of the sub shapes in a group shape?

On a live Visio diagram, right-click and run the **Identify Subshape** command to identify the sequence of the subshapes in a group shape.

For more information, see [Stencil Diagnostic Utility](#) in the chapter 6, *Constructing System Views With Building Blocks and Diagrams*.

While creating or opening Visio diagram, some of the shapes appear outside the Visio page. What should I do?

1. Click on the Visio page and choose the **File**→**Page Setup** command.
2. Click on **Page Size** tab.
3. Select the **Same as printer paper size** option, and then the **Size to fit drawing contents** option.
4. Click the **Apply** or the **OK** button.

Troubleshooting for the Architect/Requirements Client

The client installation of Systems Architect/Requirements Management is failing. What could be the possible cause?

The installation of Systems Architect/Requirements Management fails.

The error occurs when you click **Launch Teamcenter systems engineering** after opening the Systems Architect/Requirements Management page. On a client machine that does not have Systems Architect/Requirements Management installed, clicking the **Launch Teamcenter systems engineering** link installs the Systems Architect/Requirements Management client.

To resolve the error, check and modify the following Web application configuration:

1. On the Systems Architect/Requirements Management home page, click **Administrative Tools**.
2. On the **Administrative Tools** page, click **Web Application Configuration**.
3. Provide the logon credentials and click **Log In**.
4. Locate the **WOLF.AppIP** parameter. If it is set to **localhost**, change it to the machine name on which Systems Architect/Requirements Management is installed.
5. Click **Update** to save the changes.

How do I unregister and register the Architect/Requirements Client and Microsoft Office?

For the Architect/Requirements Microsoft Office interface functionality, the required libraries must be registered. The Microsoft Office interface libraries get registered each time client is launched. When the Microsoft Office interface is invoked, Architect/Requirements checks the registry for the installed version of Microsoft Office. It compares the version with the version of the client. If the versions do not match, Architect/Requirements attempts to unregister the old version and register the version installed on the computer. If the process of unregistering and registering the client, users need to manually unregister and register the Architect/Requirements client.

You require Power User or Administrator privileges for registering or unregistering the Architect/Requirements client, Microsoft Office, Microsoft Visio.

To unregister the Architect/Requirements client, run the following batch files from the client installation directory:

unregister_client.bat

To register the Architect/Requirements Client, run the following batch files from the client installation directory:

register_client.bat

To unregister Microsoft Office, run the following batch files from the client installation directory:

unregister_office.bat

To register Microsoft Office, run the following batch files from the client installation directory:

register_office.bat

To unregister Microsoft Visio, run the following batch files from the client installation directory:

unregister_visio.bat

To register Microsoft Visio, run the following batch files from the client installation directory:

register_visio.bat

What should I do if I cannot create a live Visio diagram from the Architect/Requirements client and the Visio Live—>Create Diagram menu is disabled?

You need the **Architect** privilege to create a Visio diagram. Please contact your project administrator for setting up the privilege.

I am not able to open a Microsoft Office Word document from the Architect/Requirements client. I get several errors related to com.inzoom.comjni.ComJniException.eComError. How I can get rid of those?

Un-register the **jacozoom** library by:

```
$Directory$ regsvr32 IzmJniComAx.dll /u
```

Re-register the **jacozoom** library by:

```
$Directory$ regsvr32 IzmJniComAx.dll /s
```

In the above commands, *\$Directory\$* is the folder where **IzmJniComAx.dll** file is located. For Architect/Requirements 2007.x release, the file location is the **System32** folder on your machine. For Architect/Requirements 11.1 release, the file location is the Architect/Requirements client installation folder.

If the above actions do not resolve the issue, verify if you have more than one version of the Architect/Requirements client installed on the same machine. The versions could be the Architect/Requirements 7.1.4 or earlier and the Architect/Requirements 2007.2 or later. Remove all of them and reinstall the version you intend to work with.

How to fix the several pop up messages I get related to vtable call when I launch the Architect/Requirements client?

One of the error messages you get is **Something has gone wrong with vtable call. Stack is corrupted. Please report to infoZoom.** If you have installed the Architect/Requirements 2007.1.4 and the Architect/Requirements TcSE 2007.2 or later on the same machine, you might get these error messages. The reason is the **jacozoom** DLL files are not backward compatible.

To avoid getting these errors do following:

1. Close the Architect/Requirements clients and go to the installation folder of the Architect/Requirements 2007.2 or later version.
2. Copy the **izmcomjni.jar**, **izmcomtlb.jar**, and **izmjnicom.dll** files to the Architect/Requirements 2007.1.4 client installation folder.

When you launch the Architect/Requirements client, you should not get this error.

Where can I find the log file for the Architect/Requirements client?

The log file for the Architect/Requirements client captures errors encountered while using any of the features. It helps in identifying and troubleshooting any problems.

- The Architect/Requirements client log file is located at:

C:\Users\%User Name%\AppData\Roaming\TcRequirements\tcr_log.txt.

What are the log files that I can use for troubleshooting?

You can use Systems Architect/Requirements Management log files for debugging purposes.

To view the location of the client log file:

1. Select **Tools**→**System Information**.
2. Click **Client Log**.

The client log tab displays the path for the log file. You can navigate to the log path from Windows Explorer. The following log files are present in the log path location:

- **tcr_log.txt**

This is the only file with up to date client log information and the only file that you must use for debugging.

- **tcr_log_old.txt**

This is a backup of the log file. You can use this file for debugging a problem that happened earlier than the first item shown in **tcr_log.txt**.

- **temp_tcr_log.txt**

This is a snapshot of the log file created when you click the **Open In Notepad** button. This file is not kept up to date and you must not use it for collecting client log information.

Appendix D: Changing the maximum memory available for rich client

The Architect/Requirements rich client is a 32-bit Java application. By default, the maximum memory available to the rich client is set to 1 GB. This is adequate most of the time, but you can change it using the procedure below if it is insufficient. One of the conditions where you need to increase the maximum memory is when you get an out of memory condition from a large search result or import of a large document.



The client systems running the rich client must have sufficient physical memory installed to accommodate the maximum memory setting that you choose. Additionally, your client systems may need memory for other applications, so verify the feasibility of the maximum memory setting with your IT department before making a change.

To modify the client memory allocation:

1. Create a temporary folder and extract the **launch.jsp** file from the **tcr.war** file.

Open the command prompt, change to the directory location of the **tcr.war** file, and run the following command:

```
jar xvf tcr.war ugs/tc/req/launch.jsp
```



You must have the JDK installed to run the **jar** command.

You can download JDK from

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>.

2. Open the **ugs/tc/req/launch.jsp** file in a text editor like **Textpad** or **Notepad++**.
3. Search for the following string:

```
<PARAM name="vm_args" value="-Xms64m -Xmx1024m"> <!-- vm args for launching client -->
```

Xmx1024m in the string denotes the maximum heap space. In this case, the heap space is set to **1024 MB** or **1 GB**. Change it to the desired value.

For information about the optimum heap space, see the Java documentation at <http://docs.oracle.com/>.

4. Add the updated **launch.jsp** file to the **tcr.war** file. Run the following command at the command prompt:

```
jar uvf tcr.war ugs/tc/req/launch.jsp
```

5. Deploy the updated **tcr.war** file on your Web server.

Index

- .gif images, diagrams202
- .html file name extension..... See Excel or Word
- .mhtml file name extension.....See Word
- .stp file name extension..... 138
 - Importing objects 152
 - Updating properties.....290
- .xml file name extension 142, 156
- Action column, Edit Properties dialog window
.....284
- Activating the Systems Architect/Requirements
Management Search module.....340
- ADD statement, queries372
- Adding
 - Columns272
 - Group members.....128
- Address bar
 - Disabling or enabling58
 - Hiding or showing.....59
 - Overview.....58
 - Resizing59
- Advanced search view362
- AP 233 data format
 - Updating property values290
- AP233 data format
 - Exporting project data 138
 - Importing project data 152
- AP233 STEP File
 - Exporting project data 138
- AP233 STP File
 - Exporting project data 138
- AP233 XML file
 - Importing objects 154
- AP233 XML File
 - Exporting project data 140
- Applying a column view
 - Content table267
 - Notebook pane268
- Applying views to folders265
- Approval Status property396
- Approved, change management event380
- Approver, change management user379, 381
- Approving a change..... 381
- Attaching
 - Data definitions to connections 228
 - Data dictionaries to diagrams 216
 - Diagrams..... 201
 - Notes 246
 - Spreadsheets 308
- Attachment Count property 64, 91, 396
- Attachments indicator 64, 68, 91
- Attachments tab
 - Columns
 - Adding and removing 272
 - Rearranging 274
 - Resizing 275
 - Sorting information..... 275
 - Description..... 75
 - Exporting to Excel 143
 - Using in floating window 90
- Auto Refresh check box, Data Definitions dialog
window 218
- Baseline Name dialog window 336
- Baseline property 396
- Baseline with in-work, effectivity rule .. 328, 337
- Baseline, effectivity rule 328, 337
- Baselines
 - Creating 335
 - Viewing 337
- Basic search view 353
- Behavior
 - Reference links 310
- Browser
 - Installing Systems Architect/Requirements
Management client..... 33
 - Starting Systems Architect/Requirements
Management
 - Home page 45
 - Object URL..... 122
- Browser and dialog window examples 17
- Building blocks
 - Adding to groups 128
 - Baseline

Creating.....	335	Search results.....	341
Viewing.....	337	Shortcuts	
Connections		Creating	133
Creating, Architect/Requirements client	221	Hiding and showing children of master	
Creating, editing, and deleting	223	objects	134
Viewing.....	231	Navigating to master objects	134
Copying.....	119	Overview	131
Creating.....	192	Replacing with copies of master objects	135
In live Excel	305	Spreadsheets, attaching.....	308
Defining and complying objects, viewing .	238	Subtypes.....	192
Deleting.....	162	Trace links	
Demoting	195	Creating	234, 237
Diagrams		Deleting	243
Attaching.....	201	Navigating to linked objects	242
Editing.....	204	Viewing	241
Viewing, if Visio not installed	214	Trace Links	
E-mail, sending object data.....	135	Creating, Copy to Diagram, and Paste to	
Exporting		TcSE	231
AP233 data file	138	TRAM subtype	
Word	146	Adding to groups	128
XML data file.....	142	Creating	192
Freezing	329	Overview	191
Importing		Removing from groups	129
AP233 STP File	152	Unfreezing	330
AP233 XML File	154	URLs, copying.....	122
Excel	158	Versions and variants	
XML data file.....	156	Creating a variant.....	332
Linking		Creating a version	331
To other Teamcenter products	237	Deleting	334
Within Systems Architect/Requirements		Viewing	333
Management.....	234	Buttons, Systems Architect/Requirements	
Members, filtering.....	97	Management toolbar	51
Members, viewing.....	95	Calculating numeric property values with	
Moving.....	123	formulas	288
Notes, attaching.....	246	Calendar, date property value	280, 285, 287
Number property.....	193, 194, 195	Case Sensitive check box, Search module.....	356
Overview.....	191	Change approval object	
Promoting.....	195	Modifying	383
Properties		Viewing	384
Calculating numeric values with formulas		Change Approval Pending property	396
.....	288	Change Approvers property.....	396
Editing.....	279, 283, 286, 292	Change log indicator.....	66
System-defined, descriptions	396	Change Log property	66
Updating, from AP 233 data file.....	290	Change log, viewing	386
Removing from groups	129	Change management	
Renaming	125	Approving a change.....	381
Restoring.....	163	Change approval object	
Search module		Modifying	383
Advanced search view	362	Viewing	384
Basic search view.....	353	Change log, viewing	386
Intermediate search view	358	Events	

Approved.....	380	Columns	
Rejected.....	380	Adding and removing	272
Response	380	Overview	272
Submit	379	Rearranging	274
Overview	379	Resizing	275
Rejecting a change	381	Sorting information.....	275
Submitting changes for approval	380	System default view.....	266, 267
Users		Views, applying to folders.....	265
Approver	379, 381	Comparing	
Changes waiting for approval, gaining		Requirements in two folders	189
access	382	Two requirements	188
Notifier.....	380	Comparing the content of different requirements	
Change Notifiers property.....	397	187
Change Password dialog window	110	Comparison mode, requirements	187
Change Rejected property	397	Complying Object Traceability window	
Change Time property	397	Navigating to linked objects	242
Change User property	397	Viewing complying objects	238
Changing		Complying objects.....	233
Diagram type, live Visio	211	Complying Objects property.....	397
Levels, object hierarchy	184, 195	Complying Trace column, description	77
Password, Systems Architect/Requirements		Connections	
Management.....	110	Connectivity tab.....	83
Property values		Creating	
AP 233 data file	290	Architect/Requirements client	221
Edit Properties dialog window	279	Creating, editing, and deleting.....	223
live Excel	292	Data definitions	
Multiple object selections	283	Attaching	228
Properties tab	279, 283	Deleting	226
Table view cells	286	Names, hiding and showing.....	227
Saved views	272	Viewing	231
Saved views, hierarchical content table	276	Connectivity tab	
Child diagrams	203	Description.....	83
Child Object, Deleting	163	Constructing Advanced Queries	367
Child objects		Containing text field, Search module	355, 364
Creating		Containing text field, wildcard characters	355
Building blocks	194	Content elements, requirements.....	175
Folders.....	116	Content table	
Requirements	170	Column view, conditions	266
Filtering.....	97	Columns	
Viewing.....	95	Adding and removing	272
Choice property		Hierarchical view.....	64
Definition	263	Rearranging	274
Value, changing		Resizing	275
Edit Properties dialog window	279	Sorting information.....	275
Live Excel interface	298, 303	Systems Architect/Requirements	
Properties tab	279	Management Recycle Bin.....	68
Table view cells	286	Default columns.....	267
Choosing an effectivity rule.....	328	Disabling or enabling.....	70
Column Settings dialog window		Exporting to Excel	143
Adding and removing columns	272	Filtering objects in a hierarchy	97
Rearranging columns	274	Hiding or showing	70

Hierarchical.....	64	Rich text format	248
Overview.....	63	Word.....	250
Plus signs	64, 95, 113	Ports.....	220
Resizing	71	Reference links	
Switching to.....	70	Full content.....	319
Systems Architect/Requirements Management		Plain text.....	318
Recycle Bin.....	68	Requirements	
Viewing objects in a hierarchy.....	95	By importing from Word.....	171
Views, applying to folders	265	Child of existing requirement	170
Conventions		In Architect/Requirements client views.....	168
Browser and dialog window examples	17	Sibling of existing requirement	169
Names	18	Subtypes.....	168
Revisions.....	17	Top level of folder	169
Values	18	Saved views, content table.....	269
Copy Connection, live Visio menu option.....	199	Shortcuts	133
Live Visio diagram	225	Spreadsheets	308
Copy Port, live Visio menu option.....	199	Trace links	
Copy SnapShot menu.....	309	To other Teamcenter products	237
Copy Snapshot property.....	202, 398	Within Systems Architect/Requirements	
Copy to Diagram		Management	234
Trace Links		Trace Links	
Live Visio diagram	232	Live Visio diagram	232
Copy to Diagram, live Visio menu option		Variants.....	332
Relations subtab	225, 232	Versions.....	331
Copying		Views, folder	269
Copy SnapShot to requirement	309	Current Frozen Version, effectivity rule.....	328
Objects	119	Current Version, effectivity rule.....	328
Reference links, in copied objects.....	313	Data definitions	
URLs of objects	122	Attaching to connections	228
Visio diagram to live Visio.....	209	Creating	
Create Note On dialog window		In Architect/Requirements client.....	217
MHTML note, Word.....	250	In diagram.....	218
Plain text note	247	Shortcuts	
Rich text note	248	Creating	133
Create Time property	398	Hiding and showing children of master	
Create User property	398	objects.....	134
Creating		Navigating to master objects	134
Baselines	335	Overview	131
Building blocks	192	Replacing with copies of master objects.....	135
Connections		Data Definitions dialog window.....	218, 228
Architect/Requirements client.....	221	Data dictionaries	
Live Visio diagram	224	Attaching to diagrams.....	216
Data definitions		Creating	215
In Architect/Requirements client	217	Shortcuts	
In diagram	218	Creating	133
Data dictionaries	215	Hiding and showing children of master	
Diagrams.....	201	objects.....	134
Folders	115	Navigating to master objects	134
Groups.....	127	Overview	131
Notes	246	Replacing with copies of master objects.....	135
Plain text format.....	247	Date Deleted property.....	398

Date property	
Definition	263
Value, changing	
Edit Properties dialog window	279
Live Excel interface	298, 303
Properties tab	279
Table view cells	286
Debugging options, Architect/Requirements	
client	107
Default	
Property columns, content table	266, 267
View	
Setting for folder	270
Setting for user name	271
Default View option, View list	267
Defining Object Traceability window	
Navigating to linked objects	242
Viewing defining objects	238
Defining objects	233
Defining Objects property	398
Defining Trace column, description	77
Delete in TcSE, live Visio menu option	199
Delete in Teamcenter for systems engineering,	
live Visio menu option	226
Deleting	
Connections	226
Folders, requirements, building blocks,	
groups, notes	162
Reference links	314, 323
Trace links	243
Versions, variants	334
Demoting	
Building blocks	195
Requirements	185
Diagnose Connection Point, live Visio menu	
option	200
Diagram Content property	208, 211, 398
Diagram Image note	202
Diagram Inputs dialog window	201
Diagrams	
Adding to groups	128
Connections	
Creating, editing, and deleting	223
Viewing	231
Copy Snapshot property	202
Copying from Visio to live Visio	209
Creating	201
Data definitions	
Creating	218
Data dictionaries	
Attaching	216
Creating	215
Disconnecting from Systems	
Architect/Requirements Management	
database	213
Editing	204
E-mail, sending object data	135
Embedding, Word	319
Exporting	
AP233 data file	138
Word	146
XML data file	142
Images (.gif)	202
Importing	
AP233 STP File	152
AP233 XML File	154
Excel	158
XML data file	156
Overview	191
Ports, creating	220
Removing from groups	129
Shapes, default arrangement	202
Trace Links	
Creating, Copy to Diagram, and Paste to	
TcSE	231
Types, changing	211
URLs, copying	122
Viewing, if Visio not installed	214
Dialog Window examples	17
Dialog windows	
Baseline Name	336
Change Password	110
Column Settings	
Adding and removing columns	272
Rearranging columns	274
Create Note On	
MHTML note, Word	250
Plain text note	247
Rich text note	248
Data Definitions	218, 228
Diagram Inputs	201
Document Import	174
Edit	253
Edit Properties	
Data dictionaries, attaching to diagrams	216
Diagram type, changing	211
Multiple object selections	283
Properties, editing	279
Properties, filtering	277
Text Format property	182, 255
Export To Excel	145, 295
Import AP233	

Importing objects	153	Document Import dialog window	174
Updating properties	290	Document Options property	
Multi-Choice	279, 283, 286, 326	Exporting objects to Word	148
Open	153, 155, 157, 159, 205, 290, 308	Importing requirements from Word	174
Print	102	Document template	146
Proxy Server Settings		Document Template property	173, 399
Starting new Architect/Requirements		Document Template Rules property	316, 321
session	46	Document, folder subtype	
Save	139, 141, 142	Adding to groups	128
Save Properties	103	Applying views	265
Save View	269	Creating	115
Saved Report Information	345	Removing from groups	129
Search Results	341, 342	Edit dialog window	253
Select a Baseline	337	Edit Properties dialog window	
Select Document Template	149	Data dictionaries, attaching to diagrams	216
Comparing requirements in two folders	189	Diagram type, changing	211
Comparing two requirements	188	Properties	
Select Object Chooser	237	Editing	279
Select Report	345	Filtering, property tables	277
Select Saved Report	345	Multiple object selections	283
Select Subtype	207	Text Format property	182, 255
Building blocks	193	Edit Search Criteria pane, Search module	358, 360
Data definitions	217	Editable properties	
Data dictionaries	215	Definition	261
Folders	115, 116, 117	System-defined, descriptions	396
Groups	128	Values, changing	279, 283, 286, 292
Notes	250	Values, numeric, calculating with formulas	
Requirements	169, 170	288
Trace links	235	Values, updating from AP 233 data file	290
Select User	136	Viewing	272
Send Email	136	Editing	
Single-Choice	149, 279, 283, 286	Change approval object	383
Comparing requirements in two folders	189	Connections	
Comparing two requirements	188	Live Visio diagram	224
Data dictionaries, attaching to diagrams	216	Diagram type, live Visio	211
Diagram Content property	211	Diagrams	204
Text Format property	182, 255	Groups	128, 129
Submit For Approval	381	Notes	
System Information	103	Plain text format	252
Client Log tab	105	Rich text format	252
Logging Options tab	107	Text property	257
Teamcenter for systems engineering		Word	252
properties	206	Properties	
Toggle Relationship	351	Calculating numeric values with formulas	
Direction property	399	288
Disable Numbering value, Document Options		Edit Properties dialog window	279
property		From AP 233 data file	290
Exporting objects to Word	148	Live Excel	292
Importing requirements from Word	174	Multiple object selections	283
Disconnecting a live Excel spreadsheet	307	Properties tab	279, 283
Disconnecting a live Visio diagram	213		

Table view cells	286	PLM XML	150
Requirement content		Word	146
Content elements.....	175	XML data file	142
Formatting.....	175	Reference links, in exported objects	315
Page setup	177	Reports, Search module	
Text property cells	179	Excel	347
Word	177	Visio.....	350
Requirements		Word.....	349
Word	175	System information to Excel	103
Effectivity		Filter operators.....	101
Definition	325	Filter toolbar	98
Rule, filtering versions.....	328	Filtering data.....	101
Elements of Systems Architect/Requirements		Filtering objects	97
Management main window.....	48	Filtering objects in a hierarchy	97
E-mail, sending from Systems		Filtering properties	97
Architect/Requirements Management client		Filtering property tables.....	277
.....	135	Filtering versions by effectivity rule.....	328
Embedding, diagrams, Word	319	First, button, Data Definitions dialog window	
Emptying Systems Architect/Requirements		228
Management Recycle Bin.....	165	Floating windows	
Enabling Text property editing		Notebook pane tabs	90
Notes	255	Toolbar buttons.....	55
Requirements	182	Folders	
Enabling versions for a project	326	Adding to groups	128
End a connection subtype, toolbar button 54, 221		Applying views.....	265
End a connection, toolbar button	54, 221	Baseline	
End Link Using Subtype, live Visio menu option		Creating	335
.....	198	Viewing	337
End Link, live Visio menu option.....	198	Column Settings	272
Equations, using in Excel spreadsheet	308	Comparing requirements	189
Excel		Connections	
Editing properties, live Excel.....	292	Creating, editing, and deleting.....	223
Exporting objects	143	Viewing	231
Exporting reports, Search module.....	347	Copying	119
Exporting system information.....	103	Creating	115
Exporting table views	343	In live Excel.....	305
Importing objects	158	Default view, setting for folder.....	270
Live interface, Excel template	347	Defining and complying objects, viewing .	238
Search results output option.....	342	Deleting	162
Spreadsheets, attaching to objects.....	308	Diagrams	
Template	143, 293, 347	Attaching	201
Exclude Shortcuts check box, Search module	356	Editing	204
Exiting Systems Architect/Requirements		Viewing, if Visio not installed.....	214
Management.....	110	Document subtype	115
Expanding projects and folders.....	60	Adding to groups	128
Expiry Date property.....	399	Removing from groups.....	129
Export To Excel dialog window	145, 295	Document Template property	146, 175
Exporting		Document Template Rules property ..	316, 321
Objects		E-mail, sending object data.....	135
AP233 data file	138	Expanding.....	60
Excel	143	Exporting	

AP233 data file	138	Views	
PLM XML	150	Creating	269
Word	146	FOR EACH statement, queries.....	372
XML data file.....	142	Formatting, requirement content	175
Importing		Formulas, calculating numeric property values	
AP233 STP File	152	288
AP233 XML File	154	Freezing objects.....	329
Excel	158	Frozen As Of Date, effectivity rule	328
XML data file.....	156	Full Name property.....	129, 399
Linking		Generic links.....	244
To other Teamcenter products	237	Behavior.....	244
Within Systems Architect/Requirements		Creating, Modifying	244
Management.....	234	Get connection from Visio diagram, live Visio	
Members, filtering.....	97	menu option	225
Members, viewing.....	95	Get Connection From Visio Diagram, live Visio	
Moving.....	123	menu option	198
Notes, attaching.....	246	Go To Parent Diagram, live Visio menu option	
Opening.....	113	198
Properties		Go To TcSE, live Visio menu option ...	198, 204, 223
Calculating numeric values with formulas		Groups	
.....	288	Adding members.....	128
Editing.....	279, 283, 286, 292	Connections	
System-defined, descriptions	396	Creating, editing, and deleting.....	223
Updating, from AP 233 data file.....	290	Viewing	231
Removing from groups	129	Copying	119
Renaming	125	Creating	127
Restoring.....	163	In live Excel.....	305
Search module		Defining and complying objects, viewing .	238
Advanced search view	362	Deleting	162
Basic search view.....	353	Diagrams	
Intermediate search view	358	Attaching	201
Search results	341	Editing	204
Shortcuts		Viewing, if Visio not installed.....	214
Creating.....	133	E-mail, sending object data.....	135
Hiding and showing children of master		Exporting to Word.....	146
objects	134	Importing from Excel	158
Navigating to master objects.....	134	Linking	
Overview.....	131	To other Teamcenter products	237
Replacing with copies of master objects	135	Within Systems Architect/Requirements	
Spreadsheets, creating.....	308	Management	234
Style Sheet Source property.....	173	Members	
Subtypes.....	115	Filtering	97
Trace links		Viewing	95
Creating.....	234, 237	Moving.....	123
Deleting.....	243	Notes, attaching	246
Navigating to linked objects	242	Overview	126
Viewing.....	241	Properties	
Trace Links		Calculating numeric values with formulas	
Creating, Copy to Diagram, and Paste to		288
TcSE.....	231	Editing	279, 283, 286, 292
URLs, copying	122		

System-defined, descriptions	396
Updating, from AP 233 data file	290
Removing members	129
Renaming	125
Restoring	163
Search module	
Advanced search view	362
Basic search view	353
Intermediate search view	358
Search results	341
Spreadsheets, attaching	308
Subtypes	127
Trace links	
Creating	234, 237
Deleting	243
Navigating to linked objects	242
Viewing	241
Trace Links	
Creating, Copy to Diagram, and Paste to TcSE	231
URLs, copying	122
GUID property	399
Hide/Unhide Connection Name, live Visio option	198
Hide/Unhide Connection Name, live Visio option	227
Hide/Unhide Connection Names, live Visio option	227
Hide/Unhide Marked Connection Name, live Visio menu option	198
Hide/Unhide Marked Connection Names, live Visio option	227
Hide/Unhide Port Name, live Visio menu option	198
Hide/Unhide Port, live Visio menu option	198
Hiding	
Shortcuts, children of master objects	134
Hierarchical content table	64
Home page, Systems Architect/Requirements Management	33, 45
HTML note	
Creating	248
Editing	252
HTML, value of Text Format property	246
Identify Subshape, live Visio menu option	200
Image Height property	399
Image Scale property	399
Image Width property	399
Images, diagrams	202
Import AP233 dialog window	
Importing objects	153
Updating properties	290
Importing	
Keywords, import document, Word	171
Objects	
AP233 STP File	152
AP233 XML File	154
Excel	158
XML data file	156
Outline levels, import document, Word	171
Reference links, in imported objects	315
Requirements from Word	173
Include Subtypes check box, Search module.	356
Indicators	63, 90
Installing	
Systems Architect/Requirements Management client	33
Systems Architect/Requirements Management MATLAB interface	43
Installing Architect/Requirements Client with Office Integration	
Components	28
Introduction	27
Prerequisites	28
Privileges	29
Registry	29
Intermediate button, Search module	358
Intermediate search view	358
Internet Explorer	
Installing Systems Architect/Requirements Management client	33
Starting Systems Architect/Requirements Management	45
ISO 10303	138, 152, 290
Java, properties	103
Keep Previous Settings	
Search results output option	342
Keep Same Location check box, Search module	353, 358, 362
Keyboard operations	
Switching to content table	70
Switching to navigation tree	61
Switching to notebook pane	93
Keywords, import document, Word	171
Last, button, Data Definitions dialog window	228
Levels, object hierarchy	
Changing	184, 195
Creating	171
Building blocks	193
Requirements	169
Filtering	97
Viewing	95

Links	See Trace links
Links indicator	65, 91
Links tab	
Columns	
Adding and removing	272
Rearranging	274
Resizing.....	275
Sorting information.....	275
Description.....	77
Exporting to Excel	143
Hiding or showing panes	93
Navigating to linked objects	242
Relations subtab.....	80
Resizing panes	94
Trace subtab.....	77
Using in floating window.....	90
Viewing	
Defining and complying objects	238
Trace links.....	241
Live Excel interface	
Creating	
Objects	305
Trace links.....	306
Editing properties, independent session.....	293
Editing properties, session with Systems	
Architect/Requirements Management client	
.....	292
Excel template.....	143, 293
Overview.....	292
Spreadsheet	
Creating.....	295
Disconnecting from Systems	
Architect/Requirements Management	
database.....	307
Live Office Diagnosis	
Diagnose	42
Diagnostic Information	42
Email Report	42
Generate Report	42
Help.....	42
Live Office Interface Diagnostic Tool	42
Run.....	42
Suggested Fix.....	42
Live Office Interface Diagnostic Tool	42
Diagnose	42
Diagnostic Information	42
Email Report	42
Generate Report	42
Help.....	42
Run.....	42
Suggested Fix.....	42
Live Visio	
Hide/Unhide Connection Name option	227
Hide/Unhide Connection Names option.....	227
Hide/Unhide Marked Connection Names	
option.....	227
Reset Marked Connection Names option ..	227
Live Visio diagram	
Connectors	
Deleting	226
Live Visio interface	
Diagram type, changing.....	211
Diagrams	
Copying from Visio	209
Creating	201
Creating Trace Links, Copy to Diagram,	
and Paste to TcSE.....	231
Creating, editing, and deleting connections	
.....	223
Disconnecting from Systems	
Architect/Requirements Management	
database	213
Editing	204
Ports, creating	220
Viewing connections	231
Viewing, if Visio not installed.....	214
Overview	197
Stencil Diagnostic Utility	200
Log file, Architect/Requirements client	105
Logging in, Systems Architect/Requirements	
Management	33, 45, 292
Long, client debugging option.....	107
Look in field, Search module.....	353, 358, 362
Main window, Systems Architect/Requirements	
Management	48
Make Visio Diagram Non Live, live Visio menu	
option.....	198
Matching ROIN field, Search module	355
Matching ROIN field, wildcard characters....	355
MATLAB interface, installing.....	43
Member Count property . 95, 128, 129, 205, 206,	
399	
Members	
Adding to groups	128
E-mail, sending object data.....	135
Filtering in a hierarchy	97
Removing from groups.....	129
Restoring.....	163
Viewing in a hierarchy	95
Members diagram type, live Visio.....	211
MHTML Library:	38
MHTML note, Word	

Creating.....	250	Networking symbol	48
Editing.....	252	Next, button, Data Definitions dialog window	228
MHTML, value of Text Format property	246	Non-Live button, disconnecting live Excel file	307
Microsoft Internet ExplorerSee Internet Explorer		Notebook pane	
Microsoft Office Excel	See Excel	Applying a column view.....	268
Microsoft Office Visio.....	See Visio	Attachments tab	75
Microsoft Office Word	See Word	Columns	
Microsoft Word		Adding and removing	272
Styles		Rearranging	274
Direct formatting.....	176, 406	Resizing	275
Modify pane, Search module	365	Sorting information.....	275
Modifying		Connectivity tab.....	83
Change approval object.....	383	Disabling or enabling.....	93
Module bar		Hiding or showing	93
Disabling or enabling	57	Links tab	
Hiding or showing.....	57	Description.....	77
Overview.....	57	Relations subtab.....	80
Resizing	57	Trace subtab.....	77
Moving objects.....	123	Overview	72
MUI Pack for Office	38	Preview tab	85
Multi-Choice dialog window .	279, 283, 286, 326	Properties tab	74
Multilingual User Interface (MUI) Pack.....	38	Resizing	94
Name column		Switching to.....	93
Content table	64	System Settings option	268
Versions tab	88	Using tabs in floating windows	90
Name conventions.....	18	Versions tab	88
Name property	399	Where Used tab	86
Navigating		Notes	
Linked objects		Adding to groups	128
In other Teamcenter products	242	Attaching notes to notes	246
In Systems Architect/Requirements		Attaching to objects	246
Management project.....	242	Copying	119
Modules, Systems Architect/Requirements		Creating	
Management.....	57	Plain text format	247
Objects, Architect/Requirements		Rich text format	248
Live Visio interface.....	204	Word.....	250
Objects, Systems Architect/Requirements		Deleting	162
Management		Diagram images (.gif).....	202
Excel	144, 300	Editing	
Live Visio interface.....	223	Plain text format	252
Search module.....	342	Rich text format	252
Shortcuts, master objects	134	Text property	257
Navigation tree		Word.....	252
Disabling or enabling	61	E-mail, sending object data.....	135
Expanding projects and folders.....	60	Enabling Text property editing.....	255
Hiding or showing.....	61	Exporting	
Overview.....	60	AP233 data file	138
Plus signs	60, 113	Word.....	146
Resizing	62	XML data file	142
Root node, disabling or enabling	61		
Switching to	61		

Importing		Ports	220
AP233 STP File	152	Requirements, by importing from Word	171
AP233 XML File	154	Requirements, in Architect/Requirements	
Excel	158	client views	168
XML data file	156	Requirements, subtypes	168
Overview	245	Requirements, top level of folder	169
Printing content	85, 254, 260	Shortcuts	133
Properties		Sibling of existing requirement	169
Editing	279, 283	Spreadsheets	308
System-defined, descriptions	396	Trace links	234, 237
Removing from groups	129	Data definitions	
Renaming	125	Attaching to connections	228
Restoring	163	Creating, in Architect/Requirements client	
Search module		217
Advanced search view	362	Creating, in diagram	218
Basic search view	353	Data dictionaries	
Intermediate search view	358	Attaching to diagrams	216
Search results	341	Creating	215
Subtypes	246	Defining	233
URLs, copying	122	Deleting	
Viewing content	259	Folders, requirements, building blocks,	
Notifier, change management user	380	groups, notes	162
Number property	400	Trace links	243
Building blocks	193, 194, 195	Diagrams	
Requirements	169, 170, 171, 185	Copying from Visio to live Visio	209
Numeric property		Creating	201
Definition	263	Editing	204
Value, changing		Editing	
Edit Properties dialog window	279	Notes, plain text format	252
Live Excel interface	298, 303	Notes, rich text format	252
Properties tab	279	Notes, Word	252
Table view cells	286	Exporting	150
Object template	146	AP233 data file	138
Object type indicator	64, 68, 75, 77, 86, 88	Excel	143
Objects		Word	146
Adding to groups	128	XML data file	142
Complying	233	Importing	
Connections		AP233 STP File	152
Creating, Architect/Requirements client	221	AP233 XML File	154
Creating, editing, and deleting	223	Excel	158
Viewing	231	XML data file	156
Copying	119	Linking	
Creating		To other Teamcenter products	237
Building blocks	192	With Generic links	244
Child of existing requirement	170	Within Systems Architect/Requirements	
Folders	115	Management	234
Groups	127	Moving	123
In live Excel	305	Properties	
Notes, plain text format	247	Calculating numeric values with formulas	
Notes, rich text format	248	288
Notes, Word	250	Definition	261

Editable	261	Open in Word button, Create Note On dialog	
Editing	279, 283, 286, 292	window	250
Read-only	261	Open Read-only button, live Excel interface.	300
Reference links.....	309	Opening	
Reference links, target objects	309	Diagrams, if Visio not installed.....	214
Reference links, types	309	Folders	113
System-defined, descriptions	396	Notebook pane tabs in floating windows.....	90
Types.....	261	Notes	252, 259
Updating, from AP 233 data file.....	290	Projects	112
Viewing.....	272	Requirements	177, 186
Release reservation	264	Order	
Removing from groups	129	Search Results.....	352
Renaming	125	Organizing requirements in a hierarchy	184
Restoring.....	163	Original Location property	400
Search module		Outline levels, import document, Word	171
Advanced search view	362	Output Options section, Search module	357
Basic search view.....	353	Output options, Search module.....	341
Intermediate search view	358	Page setup, requirement content.....	177
Search results	341	Paragraph, requirement subtype	
Sending data by E-mail.....	135	Adding to groups	128
Shortcuts		Creating	168, 174
Creating.....	133	Removing from groups.....	129
Hiding and showing children of master		Parsing, import document, Word	
objects	134	By outline levels and keywords.....	171
Navigating to master objects.....	134	By outline levels only	171
Overview.....	131	Password	
Replacing with copies of master objects	135	Security Services	34, 46
Shortcuts, replacing with copies of master		Systems Architect/Requirements Management	
objects	135	Changing.....	110
Subtypes.....	174	Logging in.....	34, 46
Building blocks	192	Paste and Create In TcSE, live Visio menu	
Folders.....	115	option.....	198
Generic links	244	Paste as Tunnel Port, live Visio menu option	198
Groups.....	127	Paste From TcSE, live Visio menu option.....	198
Notes	246	Paste From Teamcenter for systems engineering,	
Requirements	168	live Visio menu option	225, 232
Trace links.....	234	Paste to TcSE	
Trace Links		Trace Links	
Creating, Copy to Diagram, and Paste to		Live Visio diagram	232
TcSE.....	231	Pending, Shared State property value	
Types.....	23	Reports, Search module	344
Off, client debugging option	107	Plain text note	
OID property	400	Creating	247
OLE content, exported requirements	147	Editing	252
On, client debugging option.....	107	PLM XML	
Open button, live Excel interface.....	300	Exporting objects.....	150
Open Child Diagram w/Stencil, live Visio menu		Plus signs	
option	199	Content table.....	64, 95, 113
Open Child Diagram, live Visio menu option	199	Navigation tree	60, 113
Open dialog window	153, 155, 157, 159, 205,	Ports	
290, 308		Connectivity tab.....	83

Creating.....	220	Read-only	
preamble, note, document import	171	Definition.....	261
Prerequisites for Using the Live Office Interface		System-defined, descriptions.....	396
.....	37	Reference links	
MHTML Library.....	38	Copying	313
Multilingual User Interface (MUI) Pack.....	38	Creating, full content	319
Preview tab		Creating, plain text.....	318
Description.....	85	Deleting	314, 323
Notes, viewing content.....	259	Exporting	315
Reference links, deleting.....	323	Importing	315
Reference links, viewing.....	321	Overview	309
Requirements, viewing content.....	186	Removing referenced property from type	
Using in floating window.....	90	definition.....	314
Previous, button, Data Definitions dialog		Target objects	309
window.....	228	Types	309
Printing		Versioning	315
Note content	85, 254, 260	Viewing	321
Requirement content	85, 149, 177, 186	System-defined	
Selected Panel	102	Definition.....	261
Private, Shared State property value		Descriptions	396
Reports, Search module	344	Systems Architect/Requirements Management	
Project property.....	400	system.....	103
Projects		Types	261
Expanding	60	User-defined, definition.....	263
Opening.....	112	Viewing	272
Overview.....	111	Properties tab	
Promoting		Columns	
Building blocks	195	Resizing	275
Requirements	185	Sorting information.....	275
Properties		Description.....	74
Columns		Exporting to Excel	143
Adding and removing	272	Filtering properties	277
Rearranging.....	274	Using in floating window	90
Resizing.....	275	Property columns	
Sorting.....	275	Views	
System default view	266, 267	Applying, content table.....	267
Views, applying to folders	265	Applying, notebook pane.....	268
Definition	261	Content table, conditions	266
Editable		Property editing, multiple object selections... 282	
Definition	261	Proxy Server Settings dialog window	
System-defined, descriptions	396	Starting new Architect/Requirements session	
Editing		46
Calculating numeric values with formulas		Public, Shared State property value	
.....	288	Reports, Search module.....	344
Edit Properties dialog window.....	279	Queries, Advanced search view.....	367
From AP 233 data file.....	290	Queries, statements.....	370
live Excel	292	Query Edit pane, Search module	366
Multiple object selections	283	Query View pane, Search module	364
Properties tab	279, 283	Read-only properties	
Table view cells	286	Definition.....	261
Filtering.....	277	System-defined, descriptions.....	396

Rearranging	
Columns	274
Toolbar buttons	55
Recalling	
Saved reports, Search module	345
Saved views	
Content table	267
Notebook pane	268
Recycle Bin	
Description	68
Emptying	165
Exporting to Excel	143
Restoring objects	163
Reference links	
Behavior	310
Copying	313
Creating	
Full content	319
Plain text	318
Deleting	314, 323
Exporting	315
Importing	315
Overview	309
Promote References	310
Reference Settings	310
Removing referenced property from type definition	314
Target objects	309
Types	309
Versioning	315
Viewing	321
Refresh button, Data Definitions dialog window	218, 228
Refresh button, live Excel interface	300
Refresh Entire Visio Diagram, live Visio menu option	198
Refresh With Connection, live Visio menu option	198
Registering	
Architect/Requirements Client and Microsoft Office	410
Rejected, change management event	380
Rejecting a change	381
Relations subtab, Links tab	
Deleting	
Reference links	323
Overview	80
Viewing	
Generic links	244
Reference links	321
Trace links	241
Relative searches, wildcard characters	355
REMOVE clause, queries	378
Removing	
Columns	272
Group members	129
Renaming objects	125
Replacing a shortcut with a copy of the master object	135
Report generation	See Advanced search view
Reports, Search module	
Exporting	
Excel	347
Visio	350
Word	349
Progress messages	345, 348, 349, 350
Running from Search module	345
Running from URL	344
Saving and recalling	344
Requirements	
Adding to groups	128
Baseline	
Creating	335
Viewing	337
Comparing content	
Comparison mode	187
Overview	187
Requirements in two folders	189
Two requirements	188
Connections	
Creating, editing, and deleting	223
Viewing	231
Copying	119
Creating	
By importing from Word	171
Child of existing requirement	170
In Architect/Requirements client views	168
In live Excel	305
Sibling of existing requirement	169
Subtypes	168
Top level of folder	169
Defining and complying objects, viewing	238
Deleting	162
Demoted, positions	184
Demoting	185
Diagrams	
Attaching	201
Editing	204
Viewing, if Visio not installed	214
Editing	
Word	175
Editing content	

Content elements.....	175	Creating	133
Formatting.....	175	Hiding and showing children of master	
Page setup	177	objects.....	134
Text property cells	179	Navigating to master objects	134
Word	177	Overview	131
E-mail, sending object data.....	135	Replacing with copies of master objects	135
Enabling Text property editing	182	Spreadsheets, attaching.....	308
Exporting		Subtypes.....	168, 174
AP233 data file	138	Trace links	
PLM XML	150	Creating	234, 237
Word	146	Deleting	243
XML data file.....	142	Navigating to linked objects	242
Freezing	329	Viewing	241
Importing		Trace Links	
AP233 STP File	152	Creating, Copy to Diagram, and Paste to	
AP233 XML File	154	TcSE	231
Excel	158	Unfreezing	330
Word	171	URLs, copying.....	122
XML data file.....	156	Versions and variants	
Keywords, import document, Word.....	171	Creating a variant.....	332
Linking		Creating a version	331
To other Teamcenter products	237	Deleting	334
Within Systems Architect/Requirements		Viewing	333
Management.....	234	Viewing content.....	186
Members, filtering.....	97	Requirements, ROIN	
Members, viewing.....	95	Search module	
Moving.....	123	Basic search view	353
Notes, attaching.....	246	Reserved By property	400
Number property.....	169, 170, 171, 185	Reset Marked Connection Names, live Visio	
Outline levels, import document, Word.....	171	option	227
Overview	167	Reset Marked Connection, live Visio menu	
Paragraph subtype	168, 174	option	198
Adding to groups.....	128	Resizing	
Removing from groups	129	Address bar	59
Printing content.....	85, 149, 177, 186	Columns.....	275
Promoted, positions.....	184	Content table.....	71
Promoting.....	185	Module bar.....	57
Properties		Navigation tree	62
Calculating numeric values with formulas		Notebook pane	94
.....	288	Response Time, client debugging option.....	107
Editing.....	279, 283, 286, 292	Response, change management event.....	380
System-defined, descriptions	396	Restoring objects	163
Updating, from AP 233 data file.....	290	Reverse Direction, live Visio menu option... 199,	
Removing from groups	129	224	
Renaming	125	Revision conventions.....	17
Restoring.....	163	Rich text note	
Search module		Creating	248
Advanced search view	362	Editing	252
Intermediate search view	358	ROIN property	400
Search results	341	Running	
Shortcuts		Saved reports, Search module.....	345

Search report, from URL	344	Activating	340
Save dialog window	139, 141, 142	Advanced search view	362
Save Properties dialog window	103	Basic search view	353
Save View dialog window	269	Intermediate search view	358
Saved Report Information dialog window	345	Order of Search Results	352
Saved views	102	Output Options	341
Applying		Overview	339
Content table	267	Progress messages	345, 348, 349, 350
Notebook pane	268	Reports	
Content table, conditions	266	Exporting to Excel	347
Modifying	272	Exporting to Visio	350
Saved views, content table		Exporting to Word	349
Creating	269	Running from Search module	345
Saved views, hierarchical content table		Running from URL	344
Modifying	276	Saving and recalling reports	344
Search results output option	341	Search results	341
Saving and recalling		Search Options section, Search module	356
Content table views	269	Search Results	
Search module reports	344	Order	352
Search feature, Data Definitions dialog window		Search Results dialog window	341, 342
.....	228	Search subordinates via query, Search module	
Search for Name field, Search module	364	356
Search For Name field, Search module	355	Search subordinates via relations, Search	
Search For Name field, wildcard characters	355	module	356
Search members only, Search module	356	Security Profile property	400
Search module		Security profile, applying to object	381
Case Sensitive check box	356	Security Services ...34, 45, See Security Services	
Containing text field	355, 364	Select a Baseline dialog window	337
Edit Search Criteria pane	358, 360	Select Document Template dialog window ...	149
Exclude Shortcuts check box	356	Comparing requirements in two folders ...	189
Include Subtypes check box	356	Comparing two requirements	188
Intermediate button	358	Select Non TcSE Shapes, live Visio menu	
Keep Same Location check box	353, 358, 362	option	198
Look in field	353, 358, 362	Select Object Chooser dialog window	237
Matching RION field	355	Select Properties pane, Search module	358, 360,
Modify pane	365	361	
Output Options section	357	Select Report dialog window	345
Query Edit pane	366	Select Saved Report dialog window	345
Query View pane	364	SELECT statement, queries	370
Search for Name field	364	Select Subtype dialog window	207
Search For Name field	355	Building blocks	193
Search members only	356	Data definitions	217
Search Options section	356	Data dictionaries	215
Search subordinates via query	356	Folders	115, 116, 117
Search subordinates via relations	356	Groups	128
Select Properties pane	358, 360, 361	Notes	250
Specify Criteria pane	358, 360	Requirements	169, 170
Types/Subtypes tree	364	Trace links	235
Types/SubTypes tree	355	Select User dialog window	136
Search module, Systems Architect/Requirements		Send Email dialog window	136
Management			

Sending E-mail from Systems	
Architect/Requirements Management client	
.....	135
Set Data Definition, live Visio menu option..	199
Set View As Default, View menu option.....	271
Setting	
Default view	
For folder	270
For user name.....	271
Sort column.....	275
Shapes, default arrangement in live Visio	
diagrams.....	202
Shapes, live Visio diagrams, adding and deleting	
.....	204
Shared State property	
Reports, Search module	344
Short, client debugging option	107
Shortcut Objects.....	163
Shortcuts	
Creating.....	133
Hiding and showing children of master	
objects	134
Navigating to master objects.....	134
Overview.....	131
Replacing	135
Showing	
Shortcuts, children of master objects	134
Showing and hiding filters	97
Single Sign-On (SSO)..... See Security Services	
Single-Choice dialog window 149, 279, 283, 286	
Comparing requirements in two folders.....	189
Comparing two requirements.....	188
Data dictionaries, attaching to diagrams ...	216
Diagram Content property	211
Text Format property	182, 255
SORT clause, queries.....	378
Sorting, property columns.....	275
Source Filename property	400
Source Paragraph property.....	400
Specify Criteria pane, Search module....	358, 360
Spreadsheets	
Attaching to objects	308
Creating.....	295
Disconnecting from Systems	
Architect/Requirements Management	
database.....	307
Start a connection, toolbar button	54, 221
Start Link, live Visio menu option.....	198
Starting Systems Architect/Requirements	
Management	
Excel	
Exported table view	144
Live interface.....	292
Home page	45
Microsoft Outlook	122
Object URL.....	122
Startup folder, Word	35
Static Date property	400
Static diagram type, live Visio.....	211
Stencil Diagnostic Utility	200
Stencils property	400
STEP (ISO 10303).....	138, 152, 290
Style sheet.....	146
Styles Vs Direct formatting	176, 406
Styles, Word	146
Imported requirements.....	173
Requirement content.....	175
Submit For Approval dialog window	381
Submit, change management event	379
Submitting changes for approval.....	380
Subtype property.....	400
Subtypes	
Building blocks.....	192
Folders	115
Groups	127
Notes.....	246
Requirements	168, 174
Search module	
Advanced search view	362
Basic search view	353
Intermediate search view	358
Search results	341
Trace links	234
Synergy Check-In Comment property	401
SynergyID property	401
SynergyTask property.....	401
System Information dialog window.....	103
Client Log tab	105
Logging Options tab	107
System Settings option, notebook pane	268
System Settings option, View list.....	267
System-defined properties	
Definition.....	261
Descriptions	396
Values, changing	279, 283, 286, 292
Values, updating from AP 233 data file ...	290
Viewing	272
Systems Architect/Requirements Management	
E-mail, sending from client	135
Exiting	110
Home page	33, 45
Installing client component.....	33

Live Excel interface	
Editing properties, independent session	293
Editing properties, session with Systems Architect/Requirements Management client	292
Objects, creating	305
Overview	292
Trace links, creating	306
Logging in	33, 45, 292
Login page	34, 45
Main window	48
MATLAB interface, installing	43
Password, changing	110
Recycle Bin	
Description	68
Emptying	165
Exporting to Excel	143
Restoring objects	163
Search module	
Activating	340
Advanced search view	362
Basic search view	353
Intermediate search view	358
Order of Search Results	352
Output Options	341
Overview	339
Reports, exporting to Excel	347
Reports, exporting to Visio	350
Reports, exporting to Word	349
Saving and recalling reports	344
Search results	341
Starting	
Excel, exported table view	144
Excel, live interface	292
Home page	45
Microsoft Outlook	122
Object URL	122
System information	103
Toolbar	
Displaying button groups in floating windows	55
Overview	51
Rearranging button groups	55
Tabs, notebook pane	
Attachments tab	75
Connectivity tab	83
Links tab	
Description	77
Relations subtab	80
Trace subtab	77
Preview tab	85
Properties tab	74
Using tabs in floating windows	90
Versions tab	88
Where Used tab	86
TBD button, date property calendar	280, 285, 287
TBD search criteria, Intermediate search view	360
TcSE Properties, live Visio menu option	198
TcSEPreprocessor.DOT file	173
TCSEPreprocessor.dot file	35
Teamcenter for systems engineering properties, dialog window	206
Template	
Document	146
Excel	143, 293, 347
Object	146
Text Format property	246, 401
Text property	401
Definition	263
Notes, editing	257
Notes, enabling for cell editing	255
Requirements	
Editing content	179
Requirements, enabling for cell editing	182
Value, changing	
Edit Properties dialog window	279
Live Excel interface	298, 303
Properties tab	279
Table view cells	286
Text, value of Text Format property	246
Today button, date property calendar	280, 285, 287
Toggle Column Filters, View menu option	97
Toggle Relationship dialog window	351
Toolbar buttons	
End a connection	54, 221
End a connection subtype	54, 221
Start a connection	54, 221
Toolbar, Systems Architect/Requirements Management	
Displaying button groups in floating windows	55
Overview	51
Rearranging button groups	55
Tools	
Release reservation	264
Tooltips	
Attachments tab	75
Complying Object Traceability window	240
Connectivity tab	83

Defining Object Traceability window.....	239	Search reports, running.....	344
Hierarchical content table	64	Sending by E-mail	136
Overlapping Preview tab.....	85	Usage	
Search Results dialog window	341, 342	Child Object.....	163
Trace subtab, Links tab	78	Shortcut Objects	163
Versions tab	88	Use Rich Text button, Create Note On dialog	
Trace Link Count property.....	65, 91, 401	window	248
Trace links		User name	
Creating		Default view, setting.....	271
In live Excel	306	Security Services	34, 46
To other Teamcenter products	237	Systems Architect/Requirements Management	
Within Systems Architect/Requirements		34, 46
Management.....	234	User-defined properties	
Deleting.....	243	Definition.....	263
Navigating to linked objects	242	Values, changing	279, 283, 286, 292
Overview.....	233	Values, numeric, calculating with formulas	
Properties		288
System-defined, descriptions	396	Values, updating from AP 233 data file	290
Restoring.....	163	Viewing	272
Subtypes.....	234	Using groups to maintain objects	126
Trace Links		Using OLE Objects in Word.....	178
Creating, Copy to Diagram, and Paste to		Using tabs in floating windows	90
TcSE.....	231	Using the live Visio interface	197
Trace subtab, Links tab		Value conventions	18
Overview.....	77	Values, object properties	
Viewing		Calculating numeric values with formulas	288
Defining and complying objects	238	Changing.....	279, 283, 286, 292
Trace, client debugging option.....	107	System-defined, descriptions.....	396
TRAM, building block subtype		Updating from AP 233 data file.....	290
Adding to groups.....	128	Variants	
Creating.....	192	Baseline	
Overview.....	191	Creating	335
Removing from groups	129	Viewing	337
Transitional mapping See TRAM, building block		Creating	332
subtype		Deleting	334
Type Name property	401	Renaming.....	125
Types		Viewing	333
Object.....	23	Verify Prop Map XML And Stencil, live Visio	
Property.....	261	menu option	200
Types/Subtypes tree, Search module	364	Version As Of Date, effectivity rule.....	328
Types/SubTypes tree, Search module.....	355	Version Count property	66, 91, 401
Unassign Data Definition, live Visio menu		Version Number property	402
option	199	Version Type property	402
Unfreezing objects	330	Versions	
Unregistering		Baseline	
Architect/Requirements Client and Microsoft		Creating	335
Office	410	Viewing	337
Updating properties from an AP 233 data file	290	Choosing an effectivity rule.....	328
Upgrade Diagram, live Visio menu option	198	Creating	331
URLs of objects		Deleting	334
Copying.....	122	Enabling for project	326

Freezing requirements, building blocks	329	Search results output option.....	341
Overview	325	Visio	
Reference links in	315	Copying diagrams to live Visio	209
Renaming	125	Diagrams, live Visio	
Unfreezing requirements, building blocks	330	Connections, viewing	231
Viewing.....	333	Diagram type, changing.....	211
Versions indicator	66, 91	Editing	204
Versions tab		Viewing, if Visio not installed.....	214
Columns		Diagrams, live Visio interface	
Adding and removing	272	Connections, creating, editing, and deleting	
Rearranging.....	274	223
Resizing.....	275	Creating	201
Sorting information.....	275	Shapes, default arrangement.....	202
Description.....	88	Trace Links, Creating, Copy to Diagram,	
Exporting to Excel	143	and Paste to TcSE.....	231
Using in floating window.....	90	Exporting reports, Search module	350
View list		Search results output option.....	342
Applying a column view	267	Using the live Visio interface	197
Default View option.....	267	Web browser	
System Settings option.....	267	Installing Systems Architect/Requirements	
Viewing		Management client.....	33
Baselines	337	Starting Systems Architect/Requirements	
Change approval object.....	384	Management	
Change log	386	Home page	45
Connections.....	231	Object URL.....	122
Defining and complying objects	238	WHERE clause, queries.....	371
Diagrams, if Visio not installed	214	Where Used tab	
Notes, content	259	Columns	
Objects in a hierarchy	95	Adding and removing	272
Properties	272	Rearranging	274
Reference links.....	321	Resizing	275
Requirement content	186	Sorting information.....	275
System information, Systems		Description.....	86
Architect/Requirements Management....	103	Exporting to Excel	143
Trace links.....	241	Using in floating window	90
Variants	333	WhereUsed Object Count property	402
Versions	333	Wildcard characters, relative searches.....	355
Views		Word	
Applying		Exporting objects	146
Content table	267	Exporting reports, Search module	349
Notebook pane	268	Importing requirements	173
Content table, conditions	266	Keywords, import document	171
Creating.....	269	Notes	
Default		Creating	250
Setting for folder	270	Editing	252
Setting for user name	271	OLE content, exported requirements	147
Saved.....	102	Outline levels, import document.....	171
Views, applying to folders	265	Requirements	
Views, content table		Content elements	175
Saving and recalling.....	269	Creating	171
Views, hierarchical content table		Editing	175

Editing content.....	177	Requirement data.....	146
Formatting.....	175	TCSEPreprocessor.dot file	35
Page setup	177	Viewing	
Using OLE Objects in Word.....	178	Notes, content	259
Search results output option.....	341	Requirement content.....	186
Startup folder	35	XML data format	
Styles		Exporting project data	142
Imported requirements	173	Importing project data	156
Requirement content	175		