

An Introduction to Enterprise Architect

Requirements Management with Enterprise Architect

By Sparx Systems

All material © Sparx Systems 2007

www.sparxsystems.com

Trademarks

Object Management Group, OMG, Unified Modeling Language, UML, are registered trademarks or trademarks of the Object Management Group, Inc.

Microsoft, MS Word, Excel, Visual Studio, Visual Basic are trademarks or registered trademarks of the Microsoft Corporation.

All other product and / or company names mentioned within this document are used for identification purposes only, and may be trademarks or registered trademarks of their respective owners.

Table of Contents

INTRODUCTION TO REQUIREMENTS MANAGEMENT IN ENTERPRISE ARCHITECT	4
REQUIREMENTS MANAGEMENT WITH UML	5
GLOSSARY OF TERMS	5
GETTING STARTED WITH REQUIREMENTS MANAGEMENT	6
<i>Creating a New Project</i>	6
REQUIREMENTS MODELING	7
EXTERNAL REQUIREMENT ATTRIBUTES	8
<i>Useful Features for External Requirements</i>	8
Predefining Tagged Values Types for Requirements.....	10
<i>Different Views of Requirements Using the Report View</i>	11
TRACEABILITY AND RELATING EXTERNAL REQUIREMENTS	12
<i>Creating Relationships Using Diagrams</i>	12
Creating a Common Diagram	12
<i>Types of Relationships</i>	13
<i>The Relationship Matrix</i>	14
<i>Traceability Using the Hierarchy Window</i>	15
INTERNAL REQUIREMENTS	18
CREATING CUSTOMER QUALITY REQUIREMENTS DOCUMENTATION	20
EXTERNAL REQUIREMENTS REPORTS	20
INTERNAL REQUIREMENTS REPORTS	21
IMPLEMENTATION REPORT, DEPENDENCY REPORT, AND THE REPORT VIEW	21
<i>Implementation Report</i>	21
<i>Dependency Report</i>	22
<i>Report View</i>	22
ADDITIONAL REQUIREMENTS MANAGEMENT FEATURES	23
CREATING YOUR OWN REQUIREMENT TYPES	23
COLOR CODING REQUIREMENTS.....	23
AUTO ELEMENT NAMING	24
DRAG AND DROP REALIZATIONS	25
IMPORTING EXTERNAL REQUIREMENTS USING THE CSV IMPORT	25
<i>Import into EA</i>	26
AN INTRODUCTION TO USE CASES IN ENTERPRISE ARCHITECT	30
USE CASE DIAGRAMS	30
DEFINING SCENARIOS	31
LINKING WITH REQUIREMENTS	32
ADDITIONAL FEATURES OF ENTERPRISE ARCHITECT	33
SPELL CHECKING.....	33
FIND BROWSER	33
ATTACHING AND LAUNCHING EXTERNAL DOCUMENTS AND FILES	33
THE GLOSSARY FUNCTION	35
DEFINING REQUIREMENT ATTRIBUTES USING A PROFILE OR A TEMPLATE.....	37
<i>Defining Tagged Values</i>	37
<i>Defining Requirements in a Model Template</i>	39
<i>Defining Requirements Using a Profile</i>	41
GETTING INFORMATION IN AND OUT OF EA USING THE AUTOMATION INTERFACE	45

Introduction to Requirements Management in Enterprise Architect

Enterprise Architect (EA) is one of the few UML tools that integrates requirements management with other software development disciplines, by creating requirements directly in the model. Requirements management is built into the core product, solving many of the issues of traceability, interdisciplinary team divisions, integration with change and configuration management systems. Some of the features include:

- The ability to create and view requirements directly in the model.
- The ability to detail use cases directly in the model.
- The ability to enter attributes for each requirement such as difficulty, status, and type; you can also add your own attributes.
- The ability to trace requirements to business rules, test cases and analysis artifacts.
- A unique relationship to the traceability matrix – allowing you to quickly view the impact of changes in requirements.
- The ability to create customer quality MS Word and HTML documentation.

Requirements Management with UML

The management of requirements has traditionally been one of the more difficult and problematic disciplines in the software development industry. There are a number of reasons for this, but perhaps the most significant are the following:

- Diverse group input into the requirements
- Organizational boundary divisions
- Tool boundary divisions
- Volatility of requirements
- Imprecision and ambiguities of natural languages.

The UML and Enterprise Architect can be used to reduce (and in many circumstances remove) these problems. The UML has introduced a new way to describe functional requirements, the use case. While this is a welcome addition to the requirements analyst's toolbox, the lack of clear guidelines about their application has led to some misconceptions, and a myriad of different use case styles and interpretations. The introduction of the UML use case has also led to confusion as to where to place other requirements, such as security requirements, performance requirements and quality ranges. In this paper we will discuss many of these issues and how to use Enterprise Architect to create and manage requirements effectively.

Glossary of Terms

There are several terms used in this document which you may not be familiar with. The following is a list of terms, and how they relate to requirements management and Enterprise Architect.

- Element – A generic term referring to a singular object in a model. Some of the common elements you will come across include requirements, actors and systems.
- External requirement – A requirement that is modeled as its own element.
- Internal requirement – A requirement that is modeled as the “responsibility” of an existing element.
- Model – A representation of a particular system, such as a business process or a database.
- Diagram – A common way of representing the way in which models and elements interact. The currently open diagram is usually located in the centre of the Enterprise Architect interface.
- Attributes – Data fields containing information within requirement elements.

Getting Started with Requirements Management

Gathering requirements is typically the first step in developing a solution, be it for the development of a software application or the detailing of a business process. Requirements are essentially “what the system needs to do”. The requirements management built into EA can be used to define requirement elements, link requirements to model elements which implement that requirement, link requirements together into a hierarchy, report on requirements, and move requirements into and out of model element requirements.

Creating a New Project

Before we get started, you will need to start a project within which you can work. The following steps describe how to start a project which will include the models needed for everything we cover here.

1. Upon running Enterprise Architect, you will be presented with a start page in the centre of the interface.
2. Under **Manage Projects**, select **Create a New Project...**
3. Choose a place to store your project, give it a name and click **Save**
4. From the models in the list, check the boxes next to **Requirements** and **Use Case**, then click **OK**
5. The new project will be displayed in the upper right-hand area of the Enterprise Architect interface under the **Project View**

Next, we should open a diagram you can begin to experiment in.

1. In the **Project View**, left-click the **Plus** next to the **Requirements Model**
2. In the expanded view, double left-click on the newly displayed **Requirements Model diagram**.
3. You should see some example objects already displayed on the diagram.

Now you are ready to begin modeling your requirements in Enterprise Architect.

Requirements Modeling

In Enterprise Architect you can capture or view requirements directly in the model, using the Requirements element available from the custom palette. This is known as an external requirement.

- **The custom palette is located at the far left of the Enterprise Architect interface, in the UML toolbox.**

The use of a UML element allows the requirements to be traced to other UML elements such as other requirements, use cases (even at a flow level), test cases, and analysis or design elements. This element can be used to model or document any requirements ranging from formal business requirements through to performance or security requirements. These requirements are shown diagrammatically with their relationships (see Figure 1). The core information behind any one requirement is defined in the properties section (see Figure 2).

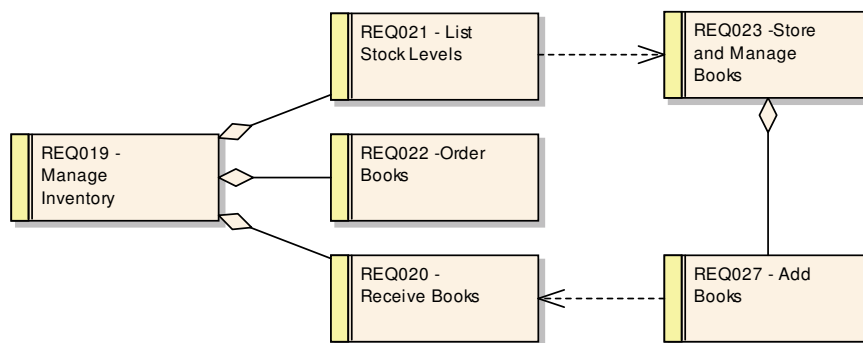


Figure 1: Custom diagram showing the requirement element

Creating an External Requirement

1. Left-click the **Custom Button** in the **UML Toolbox** to open the custom palette
2. Click-and-drag the **Requirement** element from the palette onto the diagram
3. Enterprise Architect allows you to specify several properties of the requirement
 - a. The **Short Description** field will be displayed on the diagram
 - b. See **External Requirement Attributes** below for more information
4. Click the **OK** button once finished
5. These properties can be edited again after creation by double-clicking the requirement element
6. The new requirement should now be visible in the **Project View**.

Common Uses

- Modeling business requirements
- Modeling stakeholders requests

Tips and Tricks

- The requirement elements name can be kept simply as text, or it can be manually numbered along with the text label. Enterprise Architect supports auto-numbering of requirements (see *Auto-Naming Elements* below).

External Requirement Attributes

Every element that is part of a model has properties or attributes, including requirements. The requirements attributes in Enterprise Architect are assigned in the properties sheet. (*Double-click on the Requirement*). Enterprise Architect has built-in requirements attributes such as status, difficulty, priority, and type. An example of the properties for a requirement is shown below.

The screenshot shows the 'Requirement' dialog box with the 'Properties' tab selected. The 'Short Description' field is populated with 'FR5 - 2000 hours mean time between failure.'. Below this, several attributes are listed with dropdown menus or text boxes: Status (Approved), Type (Performance), Difficulty (High), Phase (1.0), Priority (Medium), Last Update (09/04/03), Author (Chuck Wilson), Created (09/04/03), and Version (1.0). A 'Details' section at the bottom contains a text area with the following text: 'The Mean time between failure (MTBF) defines the failure tolerance of the system and the number of hours that it is operational before a failure occurs.' The dialog box has 'OK', 'Cancel', and 'Help' buttons at the bottom.

Figure 2: External requirements properties

Changing the Properties of an External Requirement

1. Double-click on the requirement element (either on the diagram or in the project view).
2. Alter the properties displayed in the properties window.
3. Click **OK** to apply the changes and exit the properties window.

Useful Features for External Requirements

When using external requirements, there are a number of ways of customizing the data and viewing the details. The following are some features, when used in conjunction with the external requirements, which enhance their use.

- Creating user-definable attributes using tagged values

- Viewing Requirements using the report view or diagram view
- Setting up relationships between requirements, as well as relationships to other UML elements such as use case, classes, test cases, etc.
- Tracing relationships between requirements and other elements.
- Creating a requirement hierarchy using child elements or packages

The following sections cover the details of these options.

Adding Additional Attributes to Requirements

It is common that there are a series of requirement attributes specific to any project. You can enter any number of additional attributes such as stability, cost, and lateness penalty through the use of tagged values.

Tagged values can be defined on a one-off basis for any element, or predefined to be included on creation of a new element.

Tagged value data for an element is available as a separate window, which is accessed using **Ctrl+Shift+6** (or from the main menu **View | Tagged Values**).

See Figure 3 for a diagram showing a one-off addition of a tagged value.

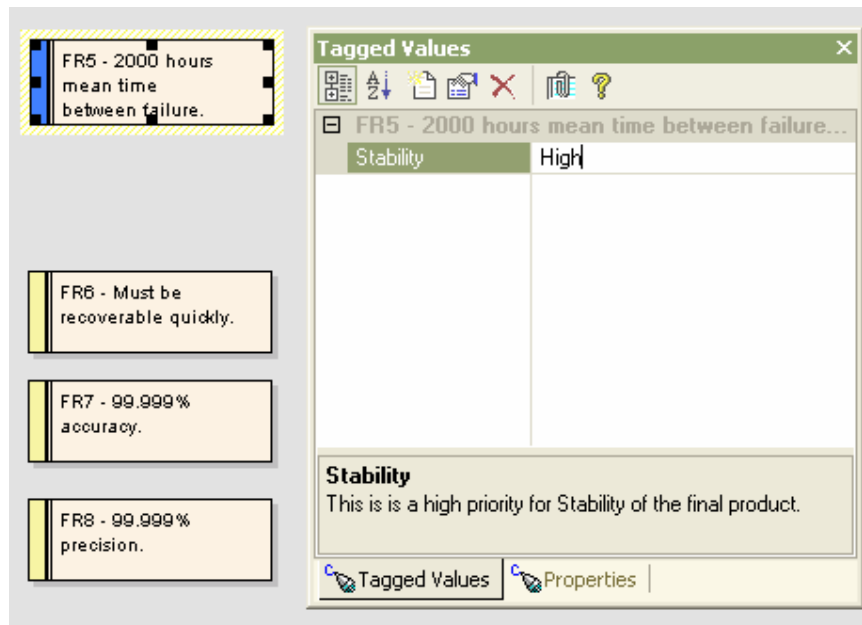


Figure 3: Requirements tagged value sheet allowing the assignment of attributes

How to Add One-Off Tagged Values to Requirements

1. Right-click the requirement
2. Select **Add | Add Tagged Value...**
3. Enter the name of the new attribute (eg. "Stability").
4. Enter the value for the new attribute (eg. "High").

5. Click **OK** to set the attribute.

If you use tagged values often, consider leaving the window open and docked (see Figure 6).

Predefining Tagged Values Types for Requirements

Elements in EA, or more specifically requirement elements, can have an extended set of attributes defined for a project. The element attributes can be predefined using either a *UML Profile* or an *EA Template*. See Figure 4 for an example of an element using a predefined set of tagged values for a project's requirement elements.

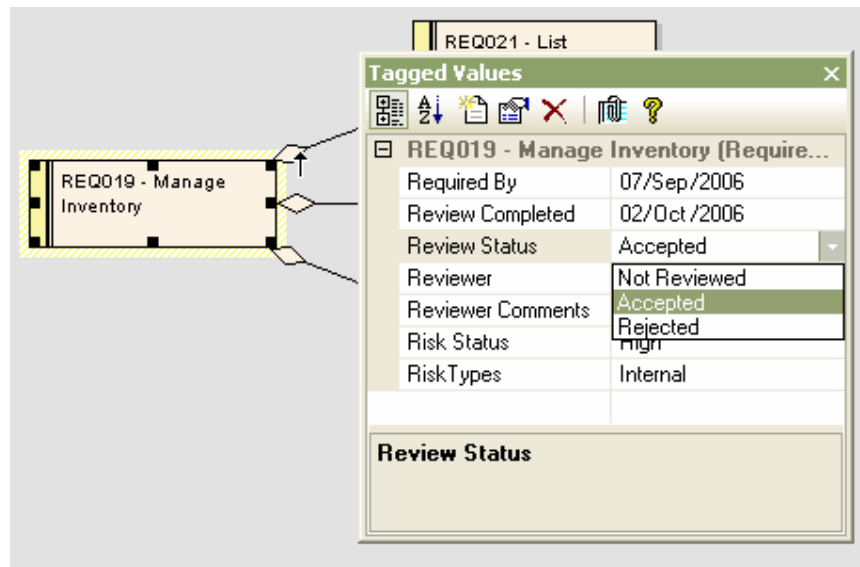


Figure 4: Using predefined tagged values

The predefined tagged values types can include a number of standard formats, such as date/time, calendar view, drop-down lists, etc.

These extended attributes can also be viewed directly on the element in the diagram.

To set this mode for a diagram, right-click on the diagram and in the context menu, select **Diagram Properties | Visible Compartments [✓] Tags**. Below is the same element in Figure 4 viewed in this mode.

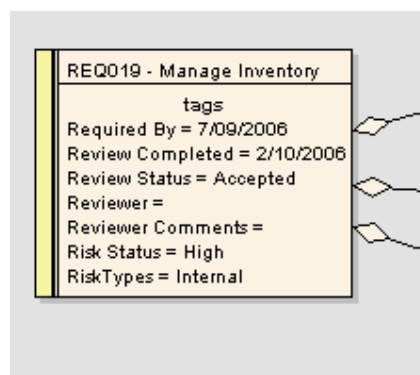


Figure 5: Tagged values visible on elements

For more information on extending requirement attributes using tagged values see:

- Appendix [Defining Requirement Attributes using a Profile or a Template](#)
- Figure 6 for a view of the docked tagged values window used with the report view.

Different Views of Requirements Using the Report View

It is not uncommon that the initial creation of requirements is carried out by users not so familiar with UML diagrams. In this case, there are some features that can be used to give a text-based view of the requirements using a hierarchical structure.

Some features to use are:

- The **Project View**. This can be docked on the left of the screen to display the hierarchy.
- The **Report View**. The diagram view can be set to report and search view mode.
 - To swap between the diagram view and the report view from the main menu, select **View | Report View**.
- The **Notes** and **Tagged Value** windows can be set to be viewed by default.
 - To view the notes window select **View | Notes**.
 - To view the tagged values window select **View | Tagged Values**.

The following is an example of viewing the requirements in this text-based mode.

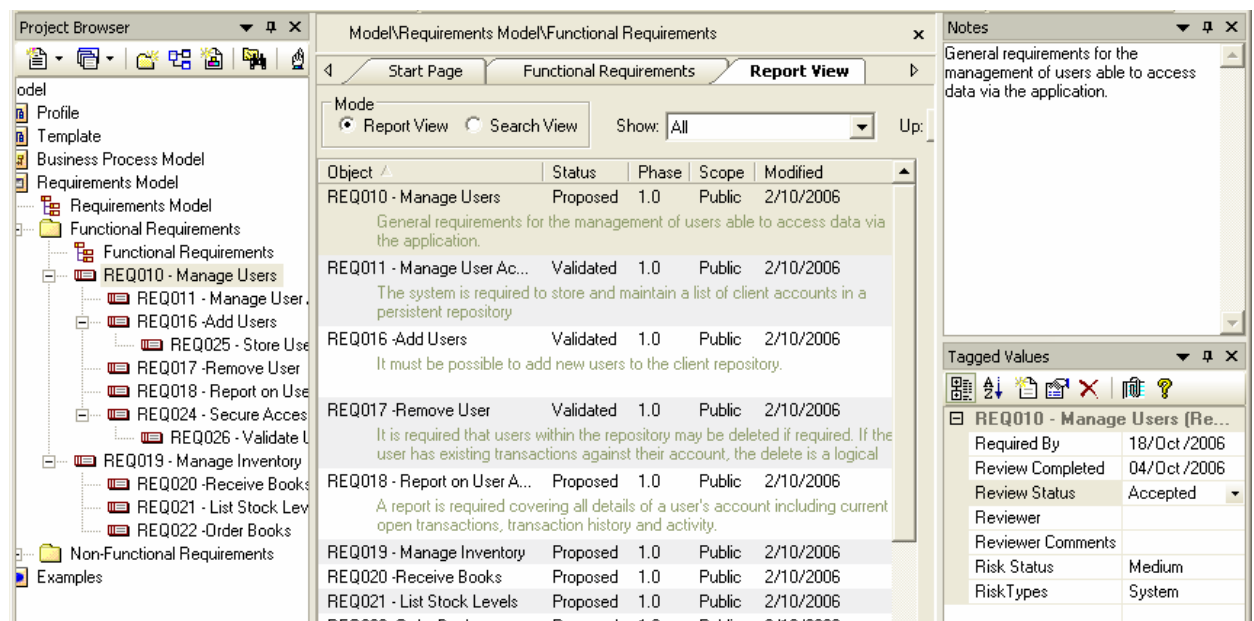


Figure 6: An alternate visual layout

Note: The elements above are structured using child elements. Using elements with child elements for the tree representation allows the report view to display all the entries in the one package.

Traceability and Relating External Requirements

In EA, there are three key methods used for tracking and forming relationships between external requirements and other elements. These relationships will define how those requirements are to be implemented within the system. The three key methods are as follows:

- **Creating and viewing relationships using diagrams**
Relationships between elements are easily created in a diagram using standard relationships defined in the toolbar.
- **Creating and viewing relationships using the Relationship Matrix**
The relationship matrix provides a process for viewing or creating links between elements in different packages, independent of them being defined in a diagram.
- **Tracing relationships using the Hierarchy View**
The hierarchy window provides a feature for tracing all the relationships of a selected element.

Creating Relationships Using Diagrams

Creating relationships between elements on a diagram is a simple process in Enterprise Architect. A click-and-drag method from source to target is used to define these relationships.

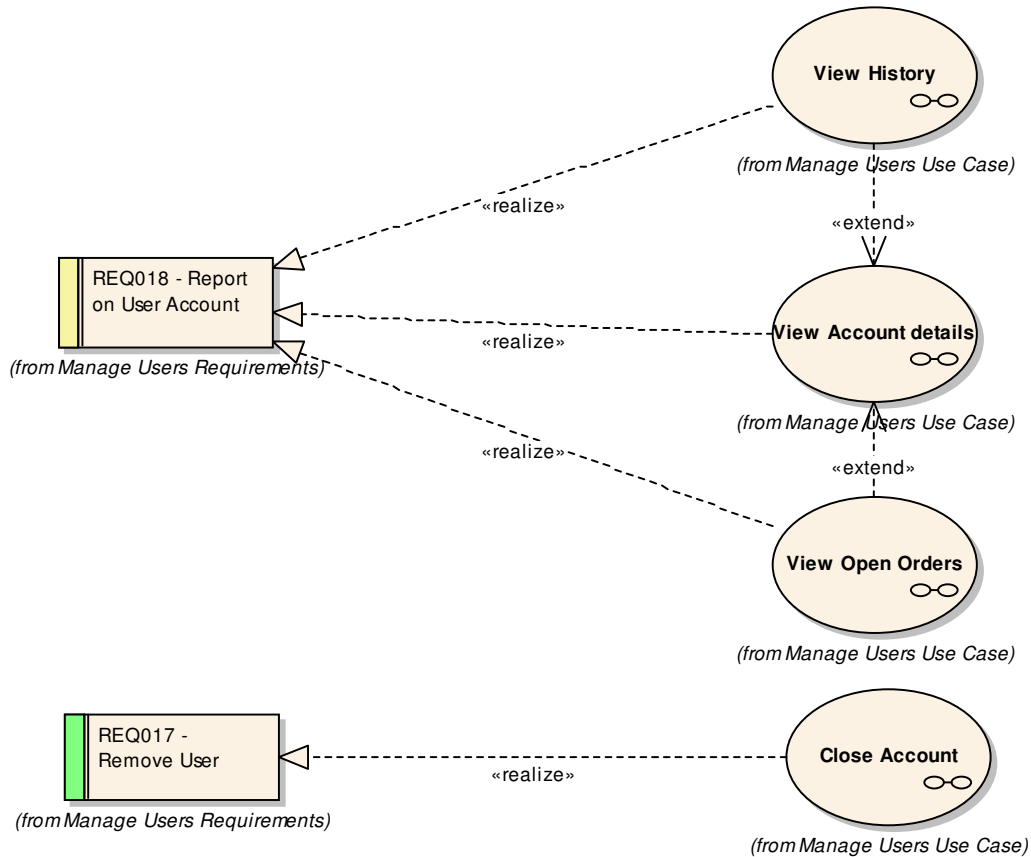
1. Add two requirements elements to your diagram.
2. Select the type of relationship you wish to create from the lower half of the custom palette (eg. **Aggregate**).
3. Click-and-hold the left mouse button on the source element for the relationship.
4. Drag the mouse to the target element, and then release the button.

Creating a Common Diagram

Creating links between objects in different packages can be a simple process using a common diagram. To do this, simply:

- Create a new diagram
- Drag onto the diagram, from the **Project View**, the elements in the different packages.

Below is an example of a diagram with elements from different packages that were linked via the relationship matrix.



Note: The properties of this diagram have been set to display the diagram source (using the **Diagram Properties: [✓] Highlight Foreign Objects**).

Types of Relationships

Aggregation

Requirements linked by aggregation relationships form a composition hierarchy. High level requirements may be composed of lower level requirements, which in turn are made up of finer and more specialized requirements. This hierarchical structure helps manage the complexity of large systems with thousands of requirements and many elements being employed to implement the requirements.

Realization

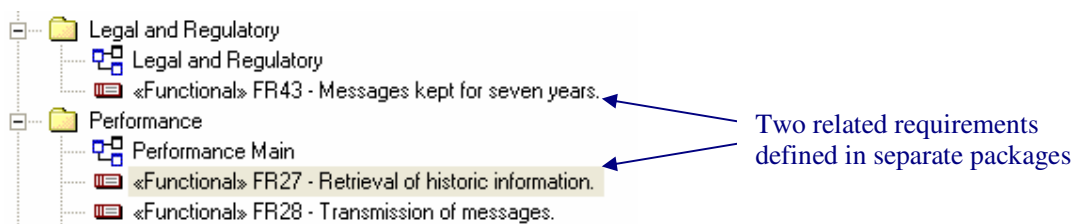
Requirements are implemented by model elements, such as use cases, classes, interfaces, components, etc. You may specify this relationship in EA using the realization link. A model element is marked as 'Realizing' a requirement. Once this link exists, EA will display the requirement in the element responsibilities tab, in the requirement hierarchy view, and in the dependency and implementation reports, as well as the standard RTF output (See [Requirements Documentation \(Reports\)](#) below for more information on reports).

The Relationship Matrix

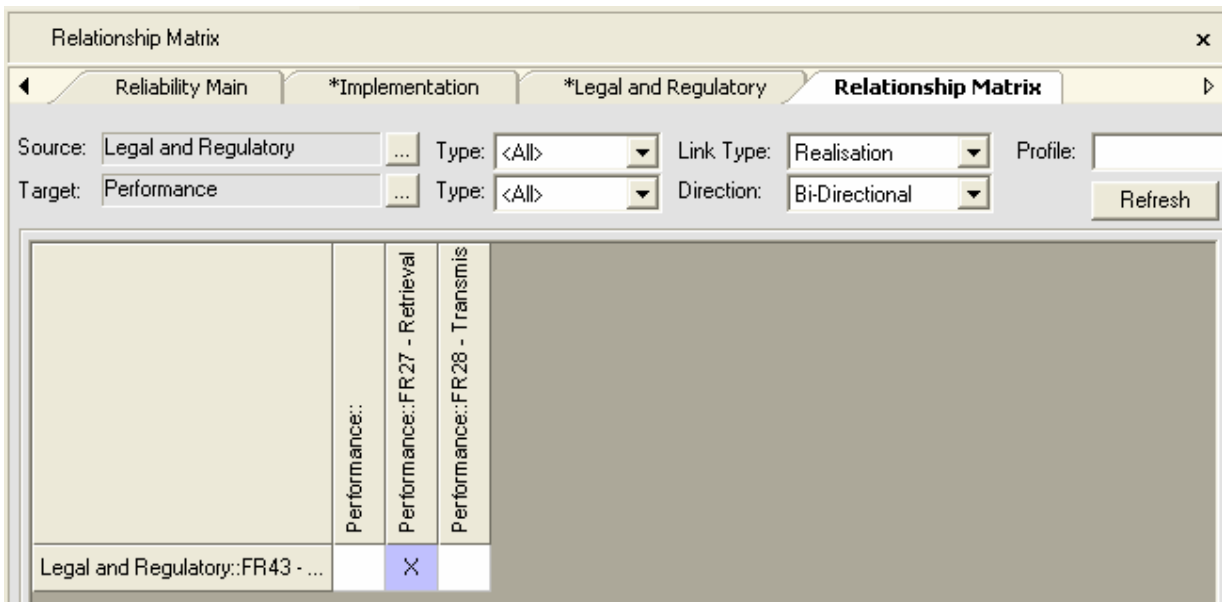
The relationship matrix allows you to create and view relationships, regardless of what diagram or packages the elements are placed in. It can be used with any UML element, but it is particularly useful with requirements management for two reasons:

- 1) With a large system definition, there will be requirements defined in different packages and diagrams that may have interdependent relationships. The relationship matrix can be used to set up these relationships that would otherwise need to be defined manually in a diagram.
- 2) As the development phase progresses, each element under the requirements, such as use case, classes, etc., needs to have its root defined by a specific requirement or group of requirements. It is this linking that is critical to backward traceability.

Below are two related requirements that are in separate packages.



To view the relationship matrix, change the view ([View | Relationship Matrix...](#)) from the menu at the top of the screen. The source and targets are set up to show the 'Legal and Regulatory' package as the source and the 'Performance' package as the target.



Adding a Relationship between Requirements Using the Relationship Matrix

1. Navigate to, and open (double left-click), the functional requirements diagram.
2. Add a requirement with the description "Font color must be green".
3. Navigate to, and open (double left-click), the non-functional requirements diagram.

4. Add a requirement with the description “Must match current look and feel”. Because these two requirements are related, the relationship should be defined.
5. Select the relationship you wish to create from the **Link Type** options (eg. Aggregate).
6. Right-click on the area where the two requirements overlap and select **Create new relationship...**
7. Right-click again for further relationship options.

Common Uses

- To view the relationships that exists between packages and elements.
- To create the relationships between requirements not located together in a diagram.
- To view the impact of a requirement change request on the system.

Tips and Tricks

- Use the relationships window to create, edit and delete relationships, rather than doing this graphically in the model diagrams.
- Save your favorite or commonly used matrix profiles. These will then appear in the resources view. This is very useful because it may often be necessary to look at the same kind of relationships.
- EA also supports an automatic process of creating a relationship using the drag-and-drop – see [Drag and Drop Realizations](#) in the **Additional Requirements Management Features** section.

Traceability Using the Hierarchy Window

The hierarchy window allows you to view the relationship between requirements. It can be used with other UML elements, and is particularly useful in requirements management.

To Use the Hierarchy Window for Viewing Relationships

- Open the hierarchy window (**View | Hierarchy** or **Ctrl+Shift+4**).
- Select an element to display its relationships.

The elements relationships are then displayed in the hierarchy window.

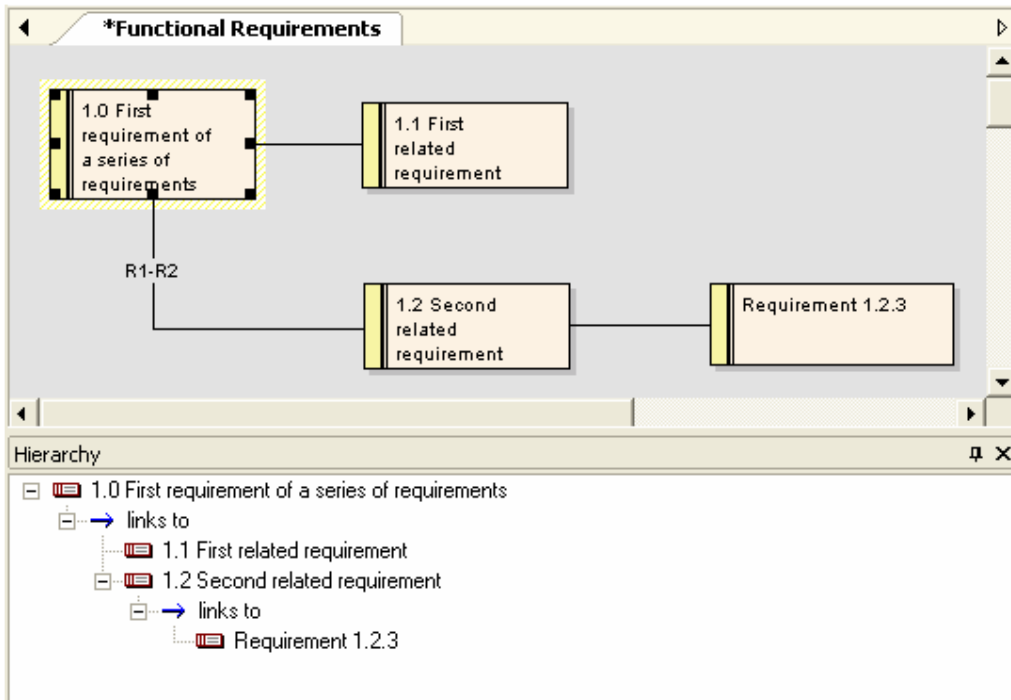


Figure 7: The hierarchy window showing the way requirement relationships are listed graphically

The hierarchy window is particularly useful to see the relationships from requirements to use cases, and down through the different levels of UML diagrams. Below is an example of a relationship between the legal requirement “Message kept for seven years”, and the use case for sending a message.

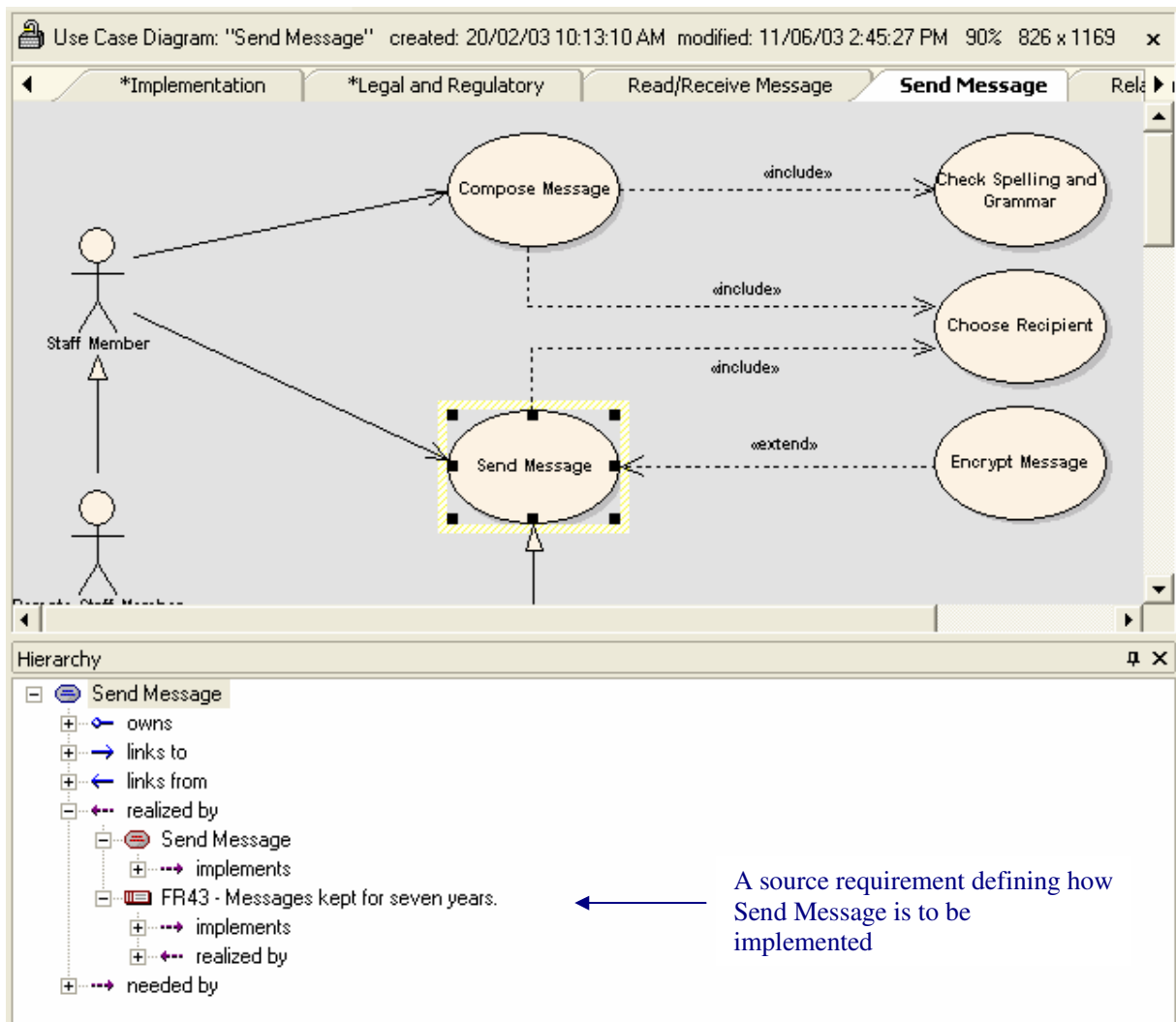


Figure 8: The hierarchy window showing the way use case relationships are related to requirements

Common Uses

- Often a diagram is deliberately drawn to show only one aspect, or part of, the underlying model. In this case, the hierarchy window is particularly useful to show the related elements that are not visible on the diagram.
- To get a quick snapshot of how a requirement (or any other element) relates to other elements in the model.

Tips and Tricks

- While the hierarchy window gives a view of the relationships for a single element you may consider using the relationship matrix as a way of viewing relationships at a higher level.
- Before you commence work on an element or group of elements, it is often useful to view how it is related to other elements. Open the hierarchy window (**View | Hierarchy**) and simply select the element in the project browser.

Internal Requirements

Enterprise Architect allows you to enter requirements at the level of an individual UML element. At this level these requirements can best be thought of as the ‘responsibility’ of the element.

Multiple internal requirements may be defined within any element from the properties window (**double-click the element**). The following diagram displays a single requirement defined within a use case element.

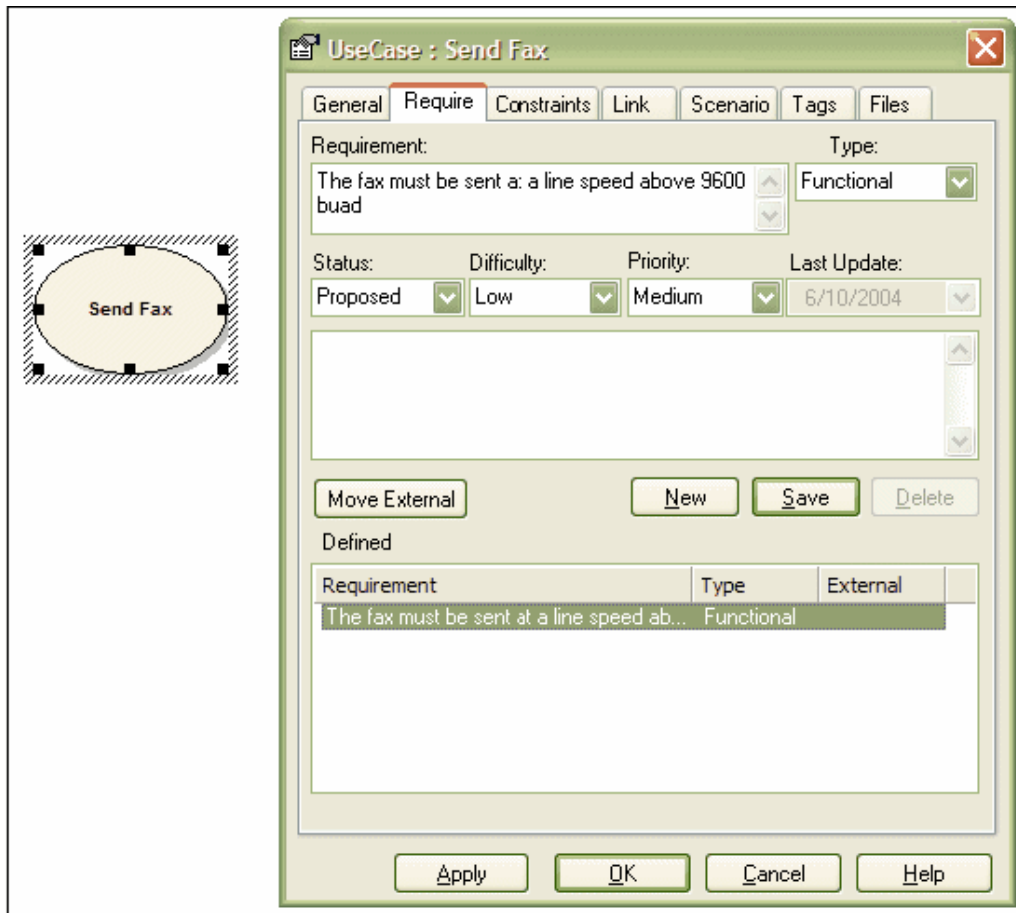


Figure 9: The property sheet for a UML element showing how to create internal requirements

Internal requirements can be externalized (see the [Move External] button above). This will create a new external requirement element with a realization relationship back to the original element (in this scenario – a use case element).

The definition of internal requirements within UML elements, such as use cases, gives a simple introduction into the more complex requirements definitions, using UML modeling in general.

Common Uses

- This can be the scenario used for setting requirements in general within EA. There are numerous reasons for defining requirements using the external requirement elements.
- To view the all the requirements or responsibilities of the element.

Tips and Tricks

- Even if the element doesn't have internal responsibilities, it will typically have external requirements. These will be displayed in the list, with the column 'External' displaying 'Yes'. Anyone working with the element will be able to view these and get an immediate view of what the element is required to do.
- As stated above, while working with an element you may define an important internal requirement. To ensure that this is captured and included, you can optionally move the requirement external (by using the move external button). This creates a custom requirement type and will request a package that the new requirement should be placed in.

Creating Customer Quality Requirements Documentation

The definition of requirements is often used as a contract either between different departments within an organization or between organizations. Therefore, it is often required that high quality documentation of this can be generated.

External Requirements Reports

Enterprise Architect’s RTF reports generator includes a report for external requirements. This can be easily copied and modified, using the report generator, to suit your reporting needs.

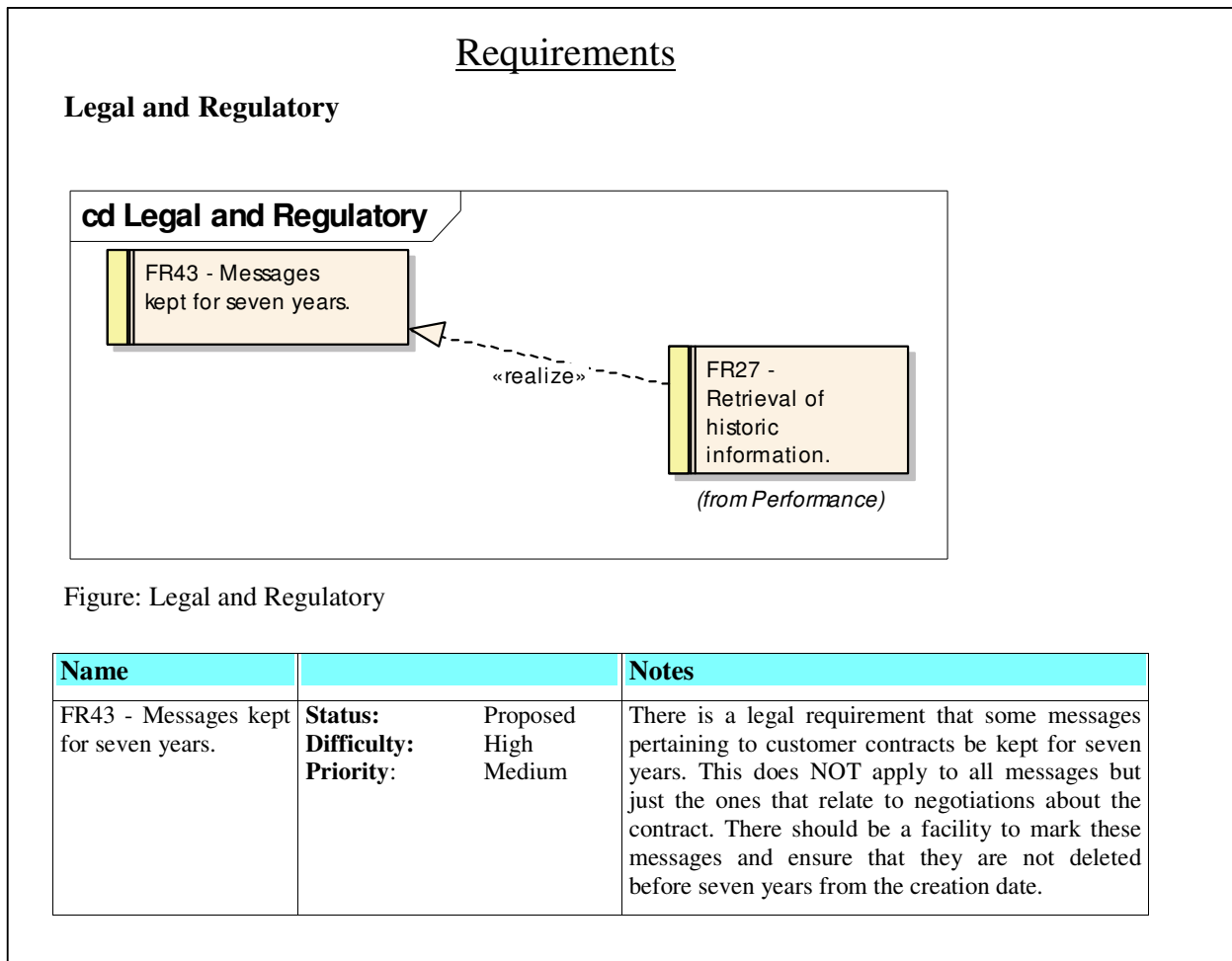


Figure 10: A short example of the RTF requirements report

To Copy an Existing RTF Document Template for Editing

1. From the main menu, select **Project | Documentation | Rich Text Format Report** (or press **F8**).
2. In the **Generate RTF Documentation** window, select **Manage Templates**.
3. In the **RTF Templates** window, select **New**.
4. In **New RTF Document Template** window, select from the **Copy Template** drop-down {Basic Template}.
5. In **New Template** type in a new name for your report.

This will open the editor ready for you to start editing the existing template to create a new report template in your own format.

Internal Requirements Reports

If you want a report on the internal requirements, the {basic template} report includes a section for the internal requirements. It is simple to copy this and remove the major detail around the internal requirements to give a report focused on these.

To modify this follow the procedure outlined above.

Common Uses

- To provide high quality customer documentation.

Tips and Tricks

- Consider making structural changes to these documents using the RTF report editor.
- Consider generating in your report definition a glossary, as an inclusion with the formal documentation.
- Consider encouraging customers to see the EA model as the documentation. (With the free EA viewer, you can allow customers to view the EA models.)

Implementation Report, Dependency Report, and the Report View

There are two additional reports, as well as the report view, that are very useful when managing requirements. They can be used in any discipline, but are particularly useful in the requirements discipline.

Implementation Report

The implementation report shows:

- Lists of elements that can be realized by other elements in the model; and
- Other model elements that realize them.

To access the implementation report, select from the main menu **Project | Documentation | Implementation Details**.

Common Uses

- To locate all elements that should have realizations.
- To locate all elements which implement a particular element.

Tips and Tricks

- Enterprise Architect, by default, only lists commonly realized elements such as use cases, requirements and components. By choosing **Set Target Types** you can tell EA to report almost any element.

Dependency Report

The dependency report lists the elements that have a dependency on another element. This is very useful for checking the dependencies placed on requirements. To access this, select from the main menu [Project | Documentation | Dependency Details](#).

Report View

The report view can be used to get a quick and simple, but clear, picture of the requirements and their detailed text. The report view swaps the diagram between two modes.

- The diagram mode which displays the elements in a graphical format.
- The report view mode which displays the textual content of the elements.

To swap between these two modes, select from the main menu [View | Report View](#).

This can be used simply, as displayed below.

There is also the option to create a fixed RTF report from this. To do this:

- [Right-click on the page](#).
- Select [Save as RTF report](#).

Object	Type	Stereotype	Scope	Status	Phase	Created
FR1 - Utilize existing hard... The system, where possible, must utilize existing hardware.	Requirement	Functional	Public	Approved	1.0	08/04/03
FR9 - Utilize existing soft... The organization has invested a significant amount of money purchasing, configuring and productionising software including s system need to utilize this investment.	Requirement	Functional	Public	Proposed	1.0	09/04/03
FR18 - Available anywhe...	Requirement	Functional	Public	Proposed	1.0	09/04/03
FR26 - Utilize the existin... The system needs to be able to send and receive postal mail. This feature needs to make use of the existing mail room facilitie there may need to be some additions or adjustment to the way the mail is handled but where ever possible the existing facilitie:	Requirement	Functional	Public	Proposed	1.0	09/04/03
FR32 - Ability to use a nu... The system must be able to be deployed onto a	Requirement	Functional	Public	Proposed	1.0	10/04/03

Figure 11: Screen view in report mode

Additional Requirements Management Features

Enterprise Architect provides a number of other features for requirements management that will be covered in the following section.

Creating Your Own Requirement Types

Enterprise Architect provides you with a number of default requirement types. You are able to modify these, add your own, or even completely tailor the list to your own project or organization's needs. (This is accessible from the main menu: **Configuration | General Types | Requirements**). It is displayed as follows:

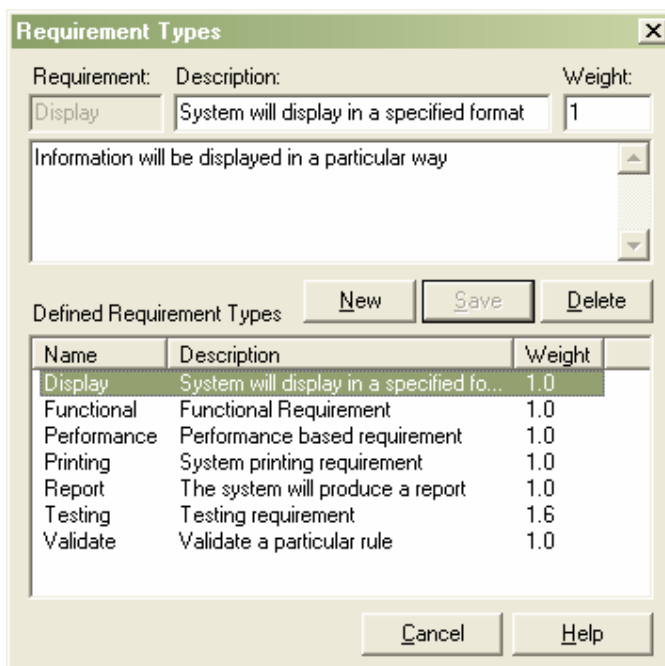


Figure 12: The requirements type definition window showing how to define your own requirement types

Common Uses

- Use this for complying with an industry, organization, project process, or standard that prescribes a list of requirement types. For example, the IEEE's Guide to Software Requirements Specifications.

Tips and Tricks

- Consider re-using the requirement types from a previous or related project.

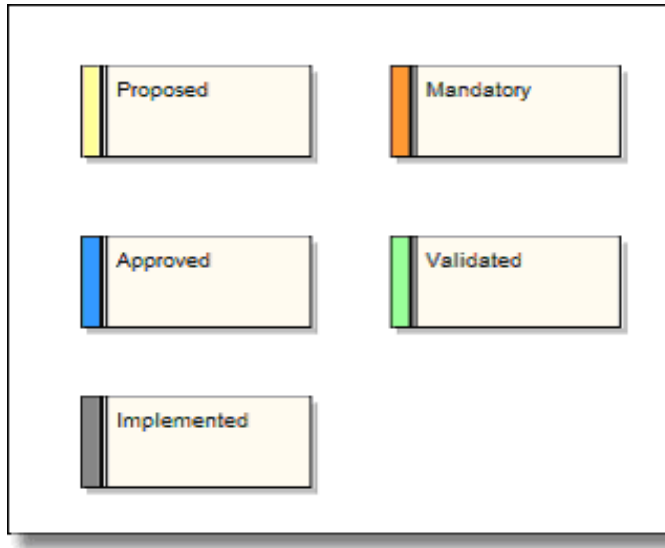
Color Coding Requirements

External requirements may be color coded to enable quick visual cues indicating the status of a requirement. To enable color coded external requirements take the following steps:

1. From the menu, select **Tools | Options** to open the local options window.
2. From the hierarchical tree select **Objects**.

3. Check the **Show Status colors on diagrams** checkbox.

Once color coding has been enabled, it will be applied as shown below.



Common Uses

- Gives a clear diagrammatic view of the status that each requirement has reached.

Auto Element Naming

This feature allows you to configure Enterprise Architect to automatically name and number requirements as they are created. (It can be used with almost any UML element such as use case and classes). It is particularly useful with requirements, since these are often required to have a unique reference for external checking.

To enable auto element naming, take the following steps:

1. Select **Settings | Auto Counters...** from the menu to open the **Auto Element Naming** window.
2. Change the **Type** to requirement.
3. Set the **Prefix** or **Suffix**. Both are optional (eg. "REQ").
4. Type in the value for your first requirement in the **Counter** box (eg. "0002" or "99").
5. Check the **Active** checkbox to enable the auto naming.
6. Click **save** to save your changes.
7. Click **Close** to exit the window.

The first requirement will take the value entered into the counter box and Enterprise Architect will then increase the value by one for every new requirement (eg. "0003" or "100").

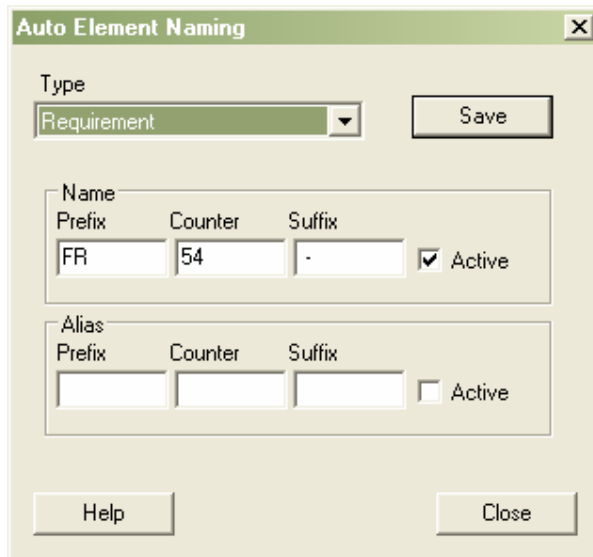


Figure 13: The auto element naming window showing how to set the counters for a requirement

Common Uses

- If your industry, organization, company, project, or team has naming standards, this mechanism can be used to ensure elements are named appropriately.

Tips and Tricks

- Auto element naming does not create Dewey-Decimal (xxx.yyy.zz) style numbering for aggregate associations. If you need this kind of numbering, consider using the automation interface. (See [Additional Features of Enterprise Architect](#)).
- Entering zeroes in front of the number will provide “padding” for the number and may make requirements easier to identify.

Drag and Drop Realizations

A fundamental aspect of the management of requirements is the ability to trace the parts of the system that implement, or realize, a particular requirement. A quick method of generating a realization link is to drag a requirement element from the project browser over an element in a diagram, which is to be the implementing element. EA will interpret this as a request to create the realization link and do so automatically.

Common Uses

- When a project member starts to create new elements in the analysis or design disciplines, it is useful to use this technique to ensure the new elements have a purpose in the model, and are being built because they realize some requirement.

Importing External Requirements Using the CSV Import

It is not uncommon for requirements to be initially entered into a document or a spreadsheet using some standard text formatting. EA provides a mechanism for importing text with a fixed structure. The simplest method is to import these text files into a spreadsheet and export this text as a CSV (“Comma Separated Values”), or tab delimited format file.

The following is a simple example of a spreadsheet containing a set of requirements to be imported into EA.

	A	B	C	D	E	F	G	H
	Name	Type	Notes	Phase	Version	Status	Priority	Difficulty
1	FR27 - Retrieval of histo	Requirement	The system must be able to	1	1	Approved	Medium	Medium
2	FR28 - Transmission of	Requirement		1	1	Mandatory	Medium	Medium
3	FR29 - The staff must n	Requirement	The system must use standard u	1	1	Proposed	Medium	Medium
4	FR30 - The interface mu	Requirement	Critical to the success of the app	1	1	Proposed	Medium	Medium
5	FR44 - The interface sh	Requirement	An important aspect of the propo	1	1	Proposed	Medium	Medium
6	FR5 - 2000 hours mean	Requirement	The Mean time between failure (t	1	1	Approved	Medium	High
7	FR6 - Must be recoverat	Requirement	In the event of software or hardwa	1	1	Proposed	Medium	Medium
8	FR7 - 99.999% accurac	Requirement	The system accuracy defines	1	1	Validated	High	High
9	FR8 - 99.999% precisio	Requirement	The precision	1	1	Implement	Medium	Medium
10	FR17 - All messages m	Requirement		1	1	Proposed	Medium	Medium
11	FR34 - Physical storage	Requirement		1	1	Proposed	Medium	Medium
12	FR35 - Wherever possib	Requirement	The organization has made signif	1	1	Proposed	High	High
13	FR31 - Other transmissi	Requirement	The system should be able to ad	1	1	Proposed	Medium	Medium
14	FR33 - Minimal or no de	Requirement		1	1	Proposed	Medium	Medium
15	FR43 - Messages kept t	Requirement	There is a legal requirement that	1	1	Proposed	Medium	High
16								
17								

Once completed, this spreadsheet is saved as a CSV format file. For example, if you are using Excel as the spreadsheet application, you would simply select **File | Save As**, in the field – *Save as Type*: Select *.CSV.

Import into EA

To import the file into EA, you need to create a CSV import structure in EA that corresponds to the columns in the CSV file. To do this, select from the main menu **Project | Import/Export | CSV Import/Export Specification**.

This will return the following window:

CSV Import/Export File Specification

Specification Name: Requirements Delimiter: ,

Notes:

Default Filename: C:\temp\requirements

Default Direction: Import

Default Types:

Available Fields

- Available Element Field
- GUID
- Stereotype
- Language
- Author
- Scope
- Alias

File Specification

- Select Element Field
- Name
- Type
- Notes
- Phase
- Version
- Priority
- Status

Buttons: Help, New, Save, Save As, Delete, Cancel

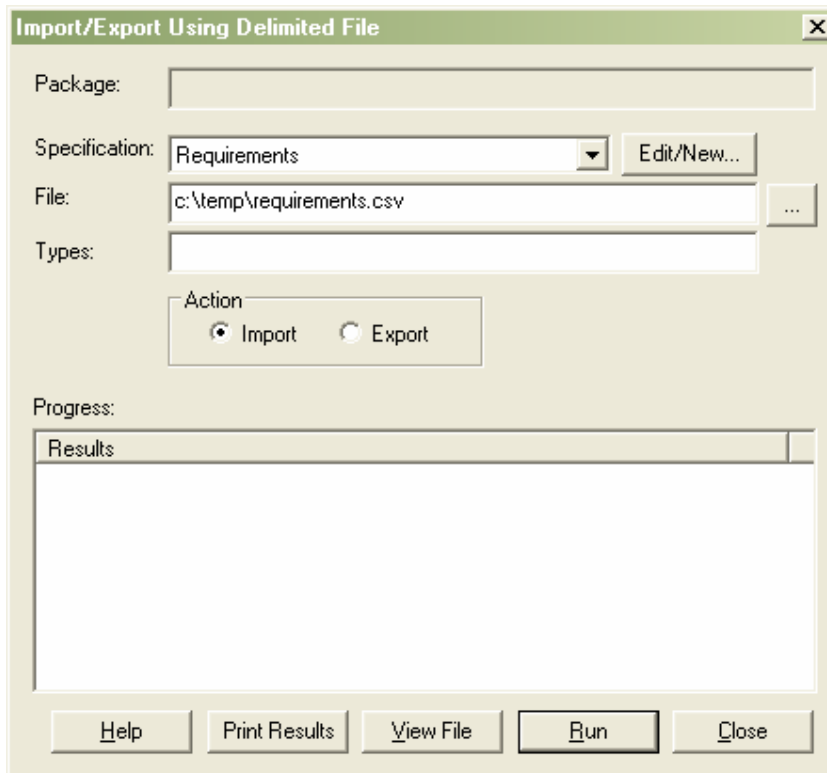
To set up a template:

- Give it a specification name
- Define the default filename the specification will use
- Set **Default Direction** to *Import*
- Select the key fields from **Available Fields** using **Add Fields** button to place them in the **File Specification** group.

Note: The order of the elements in the file specification must match the order of the columns in the spreadsheet.

Assuming the spreadsheet has been saved to a CSV format, you can now import it into EA. It is recommended at this stage to create a new package in EA that will contain the imported elements. These can then be dragged to different packages once imported.

To access the import process in EA, select from the main menu **Project | Import/Export | CSV Import/Export**. This will bring up the following window ready for you to enter information needed to perform the import.

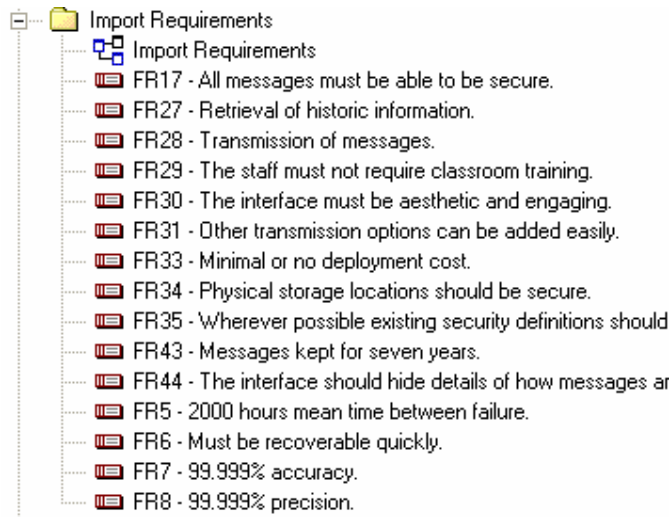


To start the import, you need to fill in the following fields:

- **Specification:** The CSV format that was defined above should now be selectable from this drop-down field.
- **File:** Insert the file location of the CSV file created from the spreadsheet.
- **Action:** Set the action to *import*.

Select **Run** to start the import process.

The data imported will be placed in the currently selected package. Below is a view of the data imported (via CSV) from the spreadsheet above.



Once imported, the requirements can then be placed in different packages by dragging the elements in the project view to their correct package.

An Introduction to Use Cases in Enterprise Architect

Use cases are used to model single tasks a user of the system may perform. They give a slightly more complex definition of the process involved in a system that conforms to the requirements laid down.

Enterprise Architect allows you to draw use case diagrams, and to specify the use case in a number of different ways. In addition to the introduction here in this section, Enterprise Architect contains use case related features such as activity, sequence and state diagrams.

Use Case Diagrams

Use case diagrams describe how a user of the proposed system will interact with this system to perform a discrete unit of work. It describes a single interaction over time that has meaning for the end user.

Use cases typically have requirement and constraint definitions associated with them. These describe the essential features and rules under which the use case will operate. Below is a simple example of a use case for an email-based contact and address book.

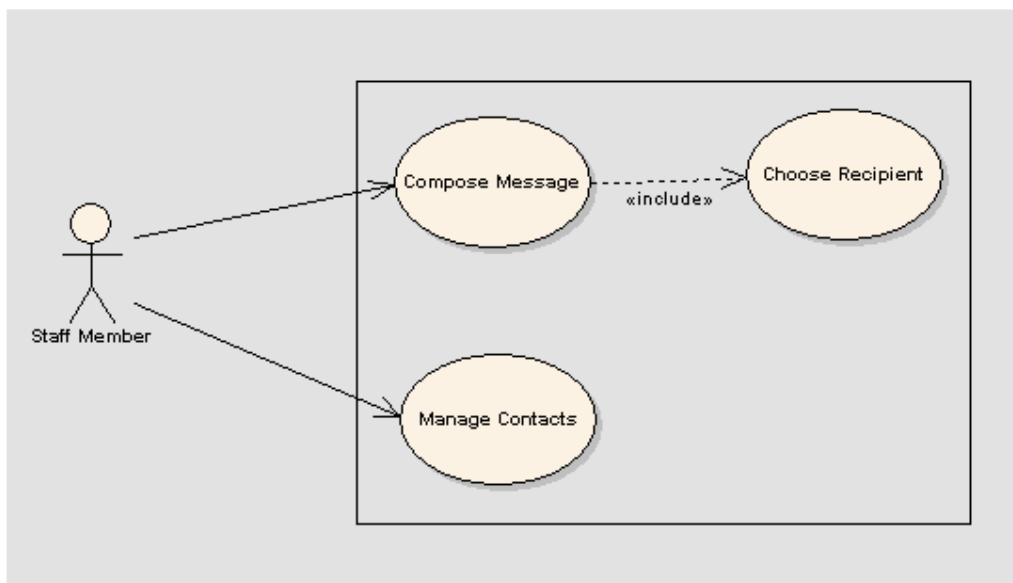


Figure 14: Use case diagram showing actors and use cases.

How to Create a Use Case Diagram

Enterprise Architect provides a use case model you can use. You can either include it when creating a new project, or **right-click the model in the Project View** and select **Add Model from Pattern**. This will provide you with further information on use cases.

Once you have added the use case model to your project, navigate to the use case model diagram and double left-click to open it.

Open the **Use Case palette** from the **Toolbox** on the left of the Enterprise Architect interface. The elements listed here include actor and use case. These elements can be dragged onto the diagram in the same way as requirement elements. Relationships can also be defined in all the same way as between requirements.

Common Uses

- To define the scope of the system;
- To define the people and other systems that will use the system;
- To document the way the business process is performed; and
- To provide the basis for the user documentation, help system or manuals.

Tips and Tricks

- **Consider using a classifier style representation of a use case for all system actors.** Newcomers are often confused (not unreasonably) by a stick person representing something non-human, like another system.
- **Consider putting the actors (primary or active) that initiate a use case on one side of the diagram,** and the actors (secondary or passive) that don't initiate the use case, but play a part in the use case on the other side.

Defining Scenarios

The use case property sheet allows you to specify attributes that apply to the use case as well as detailing the scenarios. Double left-clicking on a use case element allows you to define the specifics of the use case.

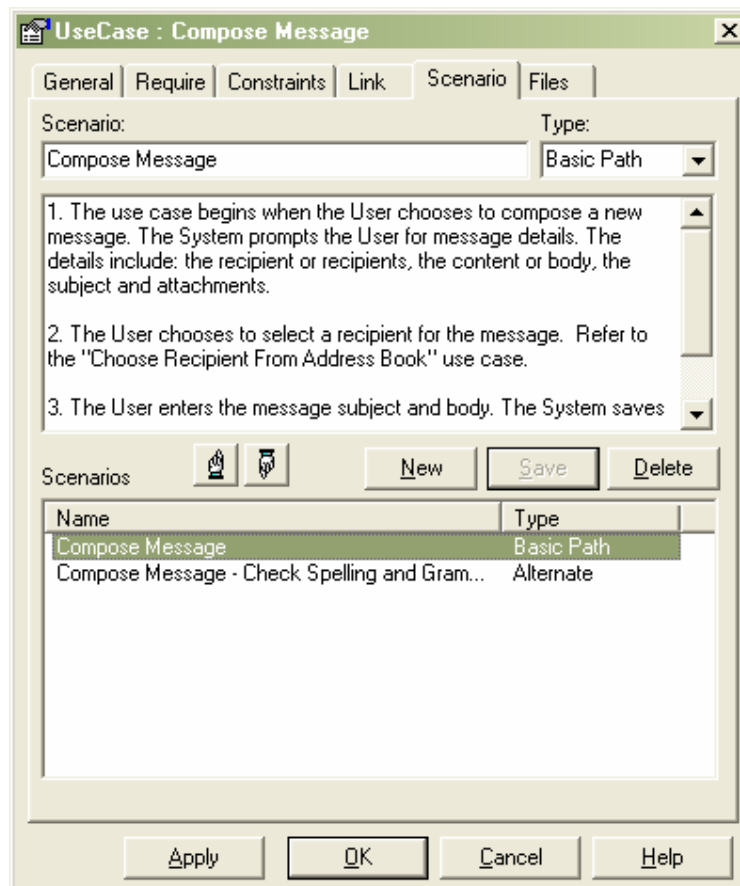


Figure 15: The property sheet for a use case showing the scenarios tab

Linking with Requirements

Using the realize relationship, you can define which use cases are implementing the requirements. See [Tracking and Relating Requirements](#) for more information on creating these links.

Additional Features of Enterprise Architect

Several of the features Enterprise Architect provides are useful across any modeling you may decide to undertake.

Spell Checking

Enterprise Architect comes with a built-in spell checking tool. This is particularly important with the requirements discipline. The requirements model is often destined to travel across organizational boundaries and form part of a contract for services internal or external to an organization.

Find Browser

Enterprise Architect comes with a very useful find window that allows you to search across the model, locate a requirement or a fragment of text, and to link straight to the object itself. The find function is accessible using either **Ctrl-F** or from the main menu using **Edit | Find**.

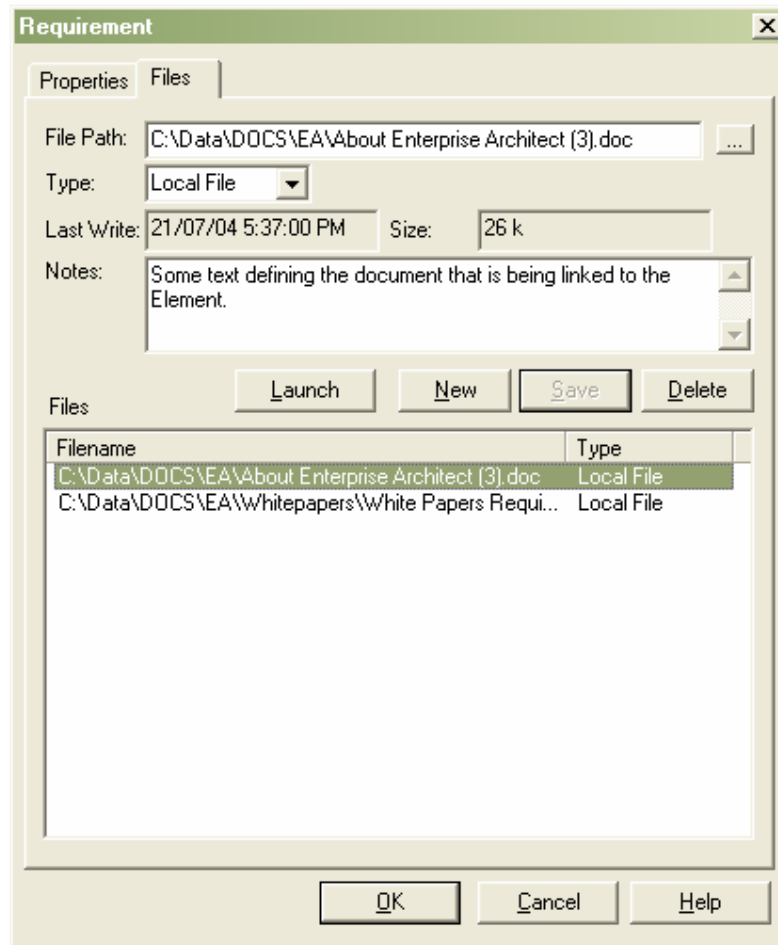
Common Uses

- To find any orphaned requirements;
- To search a word or phrase you know is part of a requirement;
- To find objects which have a particular status, such as *proposed* or *approved*; and
- To find those elements which have been modified during a certain period, for example, since the last project phase, or since a requirements baseline.

Attaching and Launching External Documents and Files

The UML specification allows for textual descriptions for each element, and Enterprise Architect provides this facility. If you have documents that need to be referenced, these can be linked to the element (a requirement, a use case, etc.), using options that allow for external documents to be linked or internal RTF documents created and stored internally. The following covers these options.

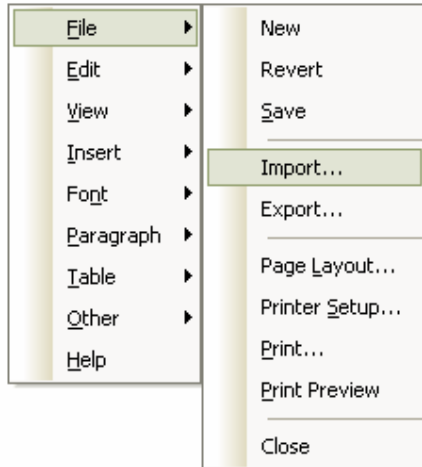
- 1) Elements and the **Files** tab.
External files can be linked to an element using the **Files** tab in the element properties window. Below is the properties dialog for a requirement with the files entries and a listing of linked documents.



- 2) **Linked Documents.**
Each element can have an internal RTF document linked to it. This is accessible by selecting the element, then right-clicking, and from the context menu selecting the **Linked Document** option. This will open the RTF editor for editing.
- 3) **Using a Document Artifact Element.**
The document artifact element is available from the **Toolbox** under **Deployment**. After creating and naming this, subsequent double-click selection of the element will open the RTF editor for word processing the internal document.

Note: Linked documents and document artifact documents can be referenced in RTF report templates using the Sections: `Element::Model Document` and `Package::Model Document`.

Options two and three above allow for external documents to be imported using the RTF editor menu. This menu is accessible by using a right-click on the body of the document. The following shows the import and export options selection.



Common Uses

- To attach a textual document that describes the requirement or use case. Organizations often require use cases to be described using text. In these situations it is beneficial to make the document available by attaching it from within Enterprise Architect.
- Formal business specifications including regulatory constraints and legal requirements may be attached as files making them available for all project members to view.

Tips and Tricks

- Consider attaching the relevant pages of your organization's standards either as a document or as a web page on your organization's Intranet Site.

The Glossary Function

Enterprise Architect has a glossary built into the core product. This allows you to enter terms and their definitions or descriptions directly into the model. Having a common and shared description of a term is important when relating new concepts to other parties involved in the system development process.

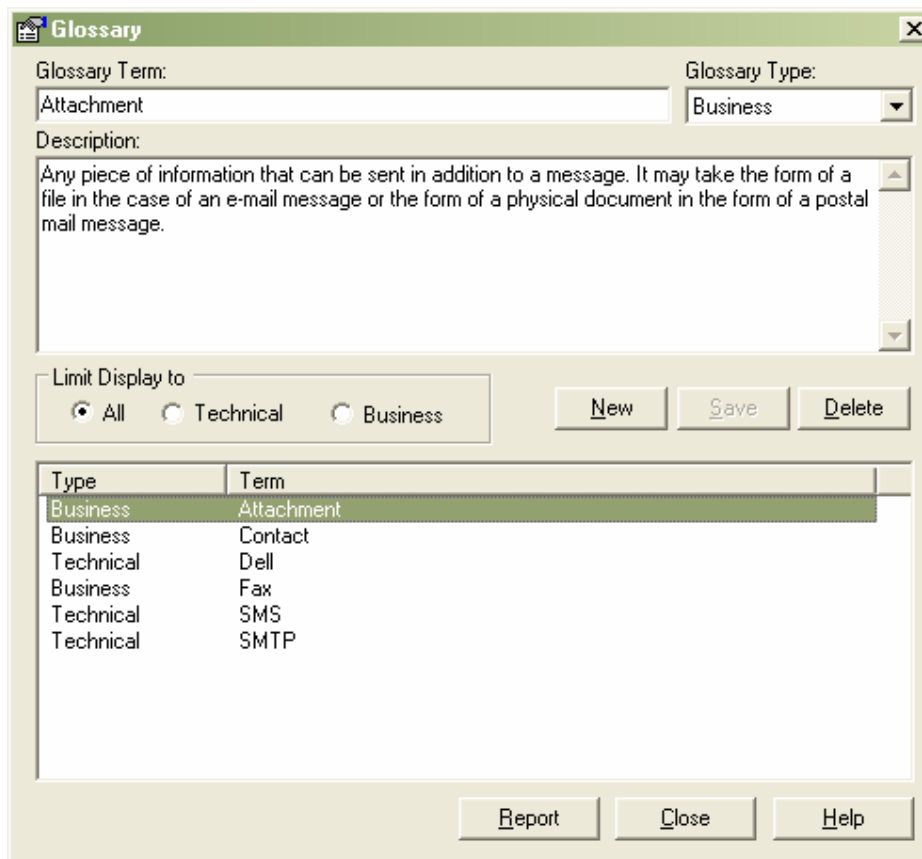


Figure 16: Glossary window showing business and technical terms and descriptions

Common Uses

- Provide business terms and their descriptions for use by the technical project members to understand important business concepts.
- Provide technical terms and their descriptions for use by the business project members.
- Provide definition of process-related terms, such as the definition of a formal requirement or a process worker.

Tips and Tricks

- Consider reusing the glossary from a previous or related project. The common terms that relate to your domain can be included in a base project (this can be exported from one repository and imported into another repository using the option on the main menu **Tools | Export Reference data** and **Tools | Import Reference data**).
- Consider generating a report (available from the glossary window) from the glossary as an attachment or inclusion with formal documentation.
- When creating new reports using the RTF generator consider adding the glossary as an appendage to some of the key documents to be generated.

Defining Requirement Attributes Using a Profile or a Template

As stated above, the requirement element can be pre-defined to include a set of user-defined attributes. These are used to document user specific qualities. The additional attributes can be defined using either a *Model Template* or a *Profile Definition*.

A short comparison of these two options is as follows:

1. The template definition is the simplest to set up, however:
 - a. It alters the default requirements element; and
 - b. It is restricted to one set of modifications to the requirement definition.
2. The profile definition is more complex to set up, but allows multiple requirement definitions and retains the default requirement element.

Both of these require attributes to be defined using the tagged values definition.

Note: If the project contains a number of requirement types that contain different attributes, then the profile approach, although more complex to set up, is better to use.

Defining Tagged Values

With tagged values, the user can define any number of fields with a wide variety of predefined or user-defined data types.

To set up a standard tagged value, select from the main menu [Settings | UML | Tagged Value Types](#). This will bring up the tagged values definition screen as shown below.

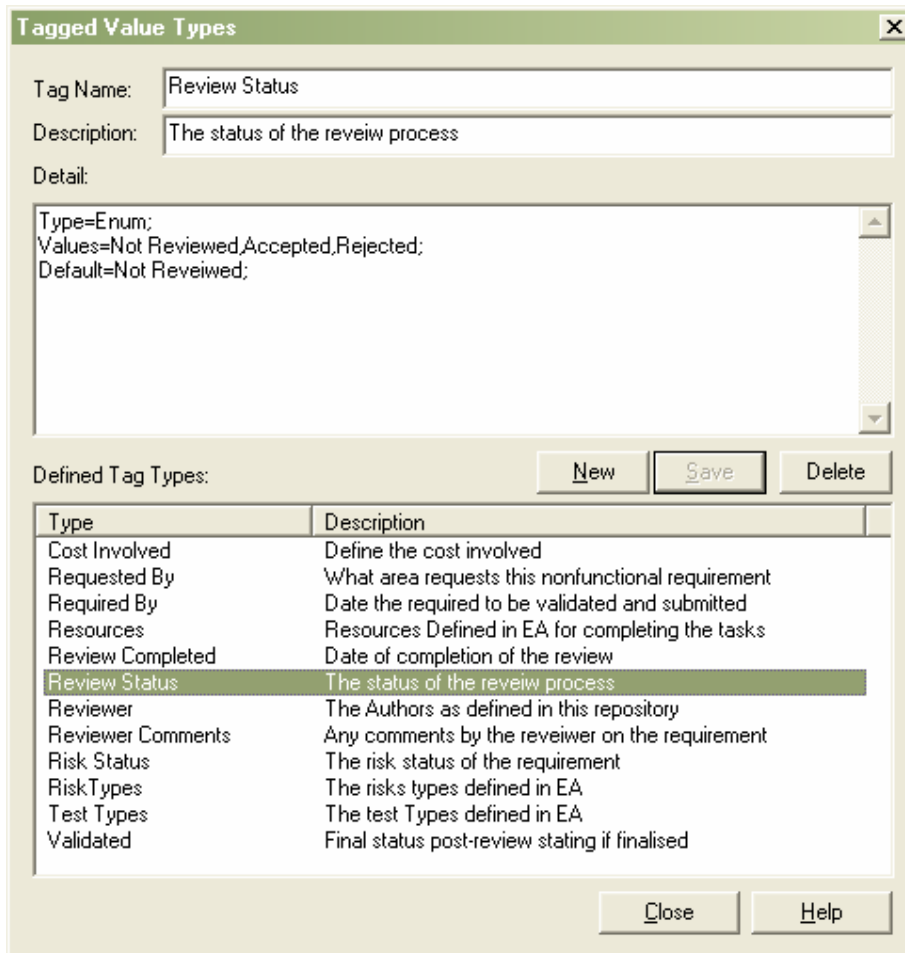


Figure 17: Tagged values definition window

In the example above, the tagged value selected, called “Review Status”, uses a predefined type to display a drop-down list of selectable options. In the detail area it contains:

Type=Enum;

Values=Not Reviewed,Accepted,Rejected;

Default=Not Reviewed;

When viewed in the tagged values window, this is presented as a drop-down option box.

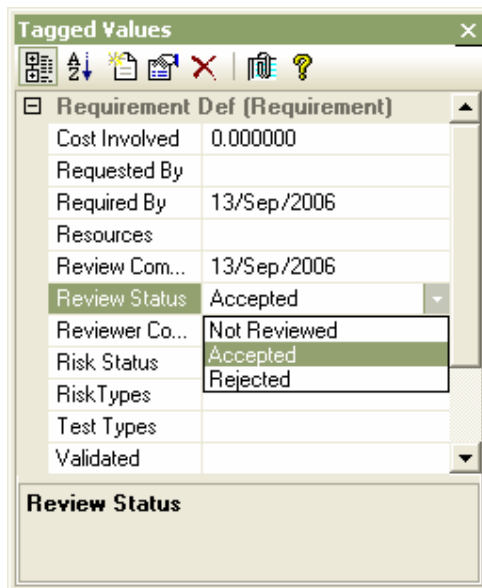


Figure 18: An element's tagged values

There are numerous standard types available, such as numeric and string types; Enumerated lists (see above), Date-Time, Boolean, Memo, etc.

For more information on setting up the standard types, and a list of types available in EA's help: use the [Index](#) tab to locate 'Predefined Tagged Value Types'

Defining Requirements in a Model Template

EA supports the creation of a template package. This can be used to define specific settings for any element type (i.e. a requirement element). An element of a specific type can be placed in the template. Any feature settings defined on this element will dictate the feature settings whenever a new element of this type is created. This is commonly used for re-defining the properties, such as default color, default stereotype, etc.

When dealing with additional attributes for requirement elements, the template can be used to define a set of tagged values that are assigned to any new requirement element.

To create a template for the model:

1. Set up a specific package that will contain the definition element(s)
2. Create a diagram for this package
3. From the main menu, select [Settings | Template Package](#).

This prompts the user to select the package to be used for defining templates.

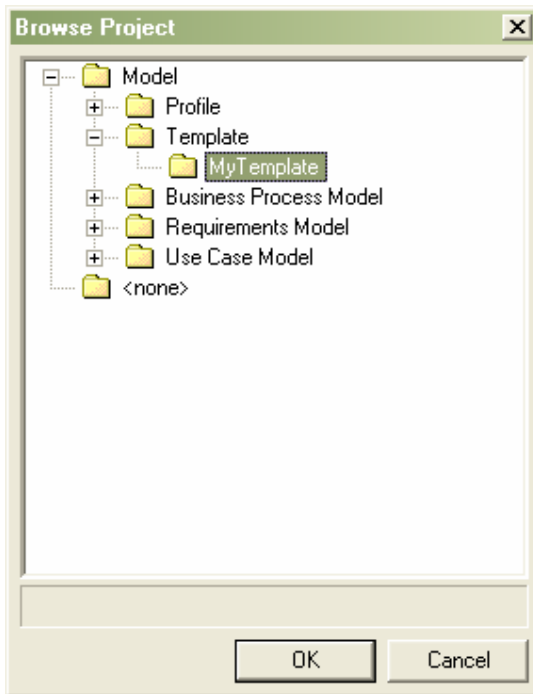


Figure 19: Setting the package for a template definition

On selecting the package and pressing **OK**, the package is now the default package for defining templates.

To pre-define new requirements tagged values:

1. Select the template package diagram.
2. From the **Toolbar**, drag a requirement element onto the diagram.
3. Select the requirement element.
4. Open the tagged values window: **View | Tagged Values (Ctrl+Shift+6)**.
5. Add any number of preset tagged values by selecting the **New Tag** button. This will display a list of the tagged values available.
6. Save the template diagram.

New requirements, created in other diagrams, will now include these tagged values – ready to have data entered.

For more information on this topic, in EA's help, perform an **Index** search using the term **Templates Package**.

Below is a view of a new requirement containing tagged values defined in a template.

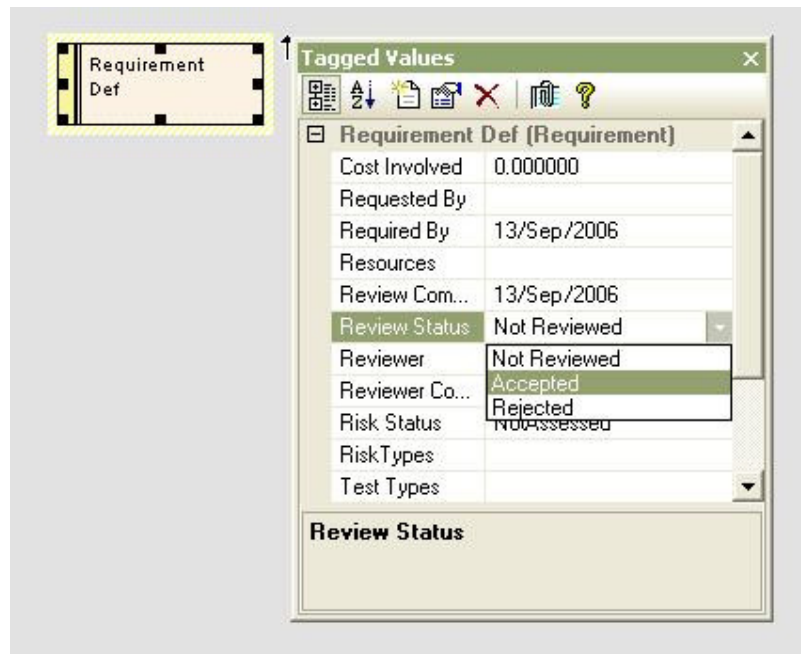


Figure 20: Requirement element definition with associated tagged values

Defining Requirements Using a Profile

EA supports the creation of profiles. Profiles allow the user to define a set of extensions to standard EA elements using tagged values.

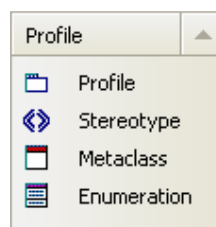
In simple terms, this is similar to the template definition above, however it also allows for:

- Any number of new element types to be defined for one standard EA element type (i.e. multiple and different requirement element types);
- The standard element type to remain unchanged; and
- A toolbox group to be defined to make the new elements user-accessible.

As is the case with a template definition, it is necessary to define a set of tagged values that will be used in the profile definition. See [Defining Tagged Values](#) above.

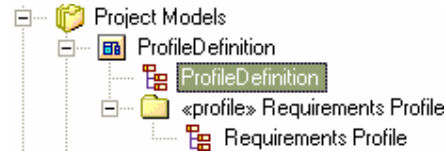
To create a profile:

1. In the project browser, set up a specific package that will contain the profile.
2. Create a diagram for the profile under this package.
3. In the **UML toolbox** open the **Profile** section.



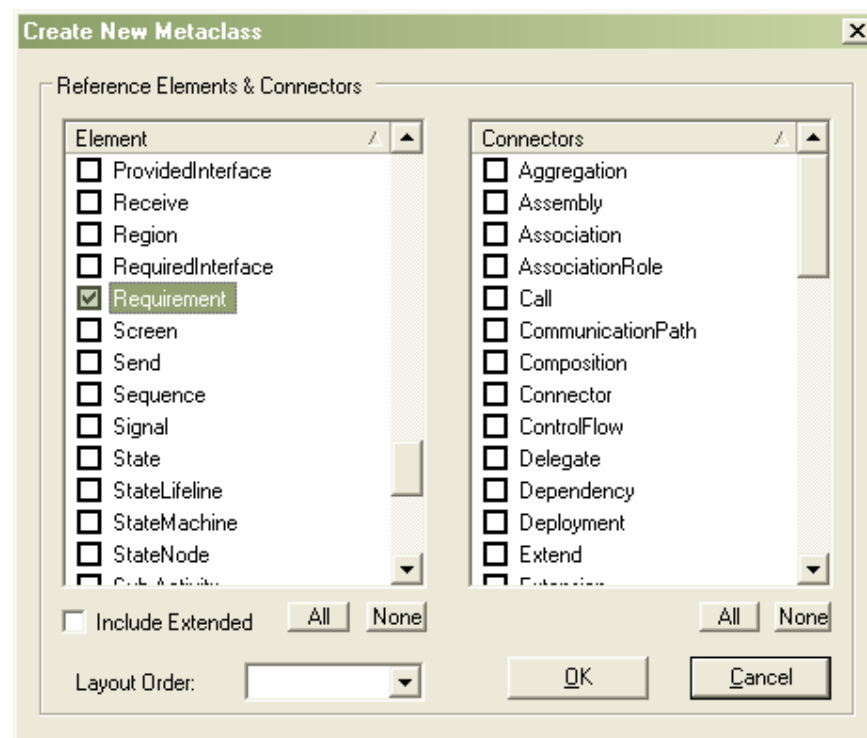
5. Drag the **Profile** element onto the diagram.

Once you have entered a name for the profile, a package will be created with the «profile» stereotype and a child diagram beneath it. This child diagram (eg. 'Requirements Profile' in the image below) will be used to add stereotypes to the profile.

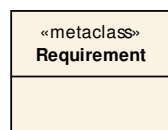


To add a new requirement definition:

1. Open the system created diagram contained in the «profile» package.
2. Create a **Metaclass** element by dragging the **Metaclass** tool from the **Profile** Toolbox. This will bring up a dialog box to select the type of element that is to be created.
3. For a requirement select the **Requirement** element type.
4. Press **OK**.



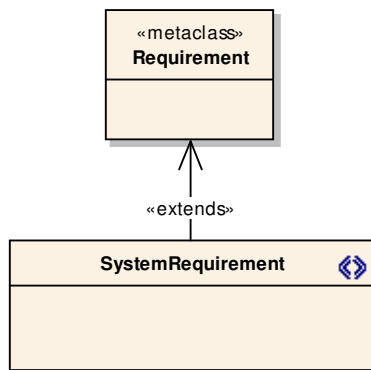
The metaclass will appear as shown below.



To define where the tagged values will be included in the profile:

1. Drag the **Stereotype** tool, from the **Profiles** toolbox, onto the diagram.
2. In the **Properties** dialog box, select the **Name** field. Type in your preferred name for the element in the new toolbox, i.e. 'SystemRequirement'.
3. Press **OK** in the **Properties** dialog.
4. From the **Profile** toolbox click on the **Extension** connector, select the new stereotype element 'SystemRequirement' and drag the mouse to the requirement metaclass. This should create an extension connector between these two elements.

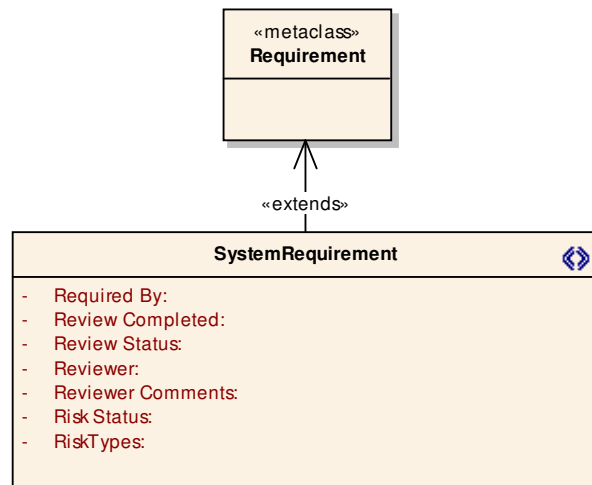
Below is a view of the elements and connection created.



To add the tagged values:

1. Select the stereotype (i.e. 'SystemRequirement')
2. From the context menu, select **Attributes**
3. In the **Attributes** dialog box, add as a new entry in the **Name** field for each of the tagged values created above, you want included in this requirement element.

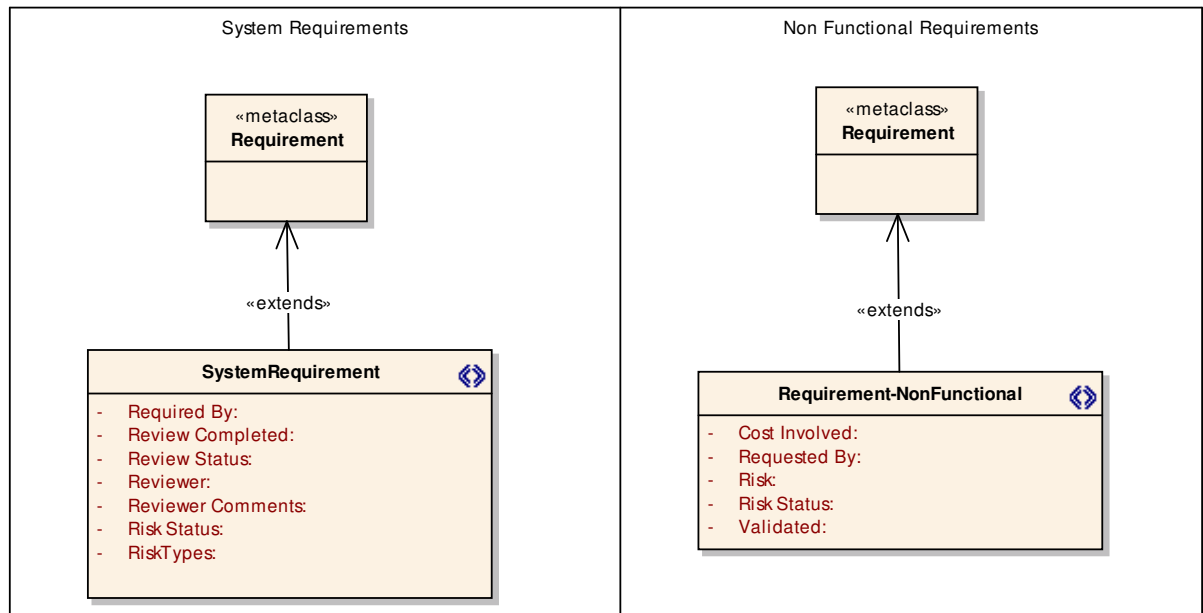
Below is an example of the above 'SystemRequirement' populated with some of the tagged values defined in Figure 17: Tagged values definition window.



Note: In the case of creating simple string or Boolean tagged values, these do not need to be defined in the tagged values definition, but can be entered directly in the attributes of the 'SystemRequirement' element.

Any number of these definitions can be set up in one profile. Each stereotype element would need to have a unique name.

Below is an example of the definition of two different requirements.



To set up the new requirements to be viewed in the Toolbox:

1. Select the «profile» package.
2. Right-click, and from the context menu, select **Save Package to UML Profile**.
3. Set the filename to save the XMI file.
4. Select **Save**.
5. Open the **Resources** view.
6. From the resources tree, select **UML Profiles**.
7. Right-click and from the context menu, select **Import Profile**.

Once this has been imported, it can be included on the toolbox using the following:

1. Select the newly imported profile;
2. Right-click on the new profile, and from the context menu, select **Show Profile in UML Toolbox**.

A new toolbox with the name of your profile package will be added to the toolbox. Now you can create requirement elements in any package by dragging these from the toolbox.

Getting Information In and Out of EA Using the Automation Interface

Enterprise Architect is an open modeling tool allowing you to freely exchange data using external application software. EA includes an automation interface which provides a way for other applications to access the information in an EA model using Windows OLE Automation (ActiveX). Typically this will involve scripting clients such as MS Word, Visual Basic, or Visual Studio.

As this is a fully programmable interface for reading from, and writing information to, the repository, it can be used to create fully customized interfaces to external applications.

For more information on the Automation Interface see the help section of EA: *Automation and Scripting*. For some examples of code for add-ins using the automation interface see http://sparxsystems.com.au/resources/developers/autint_vb.html.