

Enterprise Architect

Requirements Engineering

Version 1.0



Date:

2/05/2016

Author:

Sparx Systems & Stephen Maguire

Table of Contents

Requirements	4
Introduction	7
Meet the Requirement Tools	9
Specification Manager	10
Relationship Matrix	12
Requirement Properties	14
Requirements Diagram	16
Scenario Builder	18
Baseline Tool	20
Traceability Window	22
Dashboard Diagrams	24
The MDG Link for DOORS Add-In	26
Requirements Overview	27
What are Requirements	30
Levels and Types of Requirements	31
Characteristics of Good Requirements	35
Business Context for Requirements	40
Requirements Diagram	44
Creating and Viewing Requirements	48
Requirements Development	53
Elicitation	55
User Observations	56
Stakeholder Workshops	57
Creating Requirements	62
External and Internal Requirements	63
Using the Specification Manager	65
Requirement Properties	66
Color Coded Requirements Status	68
Displaying Properties on a Diagram	70
Import Requirements	73
Move Requirement External	78
Move Requirements Internal	79
Recording Requirement Types	80
Analysis	81
Models Used to Document Requirements	82
Requirements Naming and Numbering	84
Model Assumptions and Constraints	88
Create a Glossary	90
Create a Domain Model	91
Model the User Interface	93
Prioritize the Requirements	94
Specification	97
Specify Quality Attributes	98
Requirement Sources	100
Elaborate the Requirements	102
Validation	104
Derive Test Cases	105
Review Requirements	106
Requirements Management	109
Tracing Requirements	110

Tracking Requirements	113
Managing Changing Requirements	115
Impact Analysis of Changes	118
Requirement Volatility	120
Requirements Reuse	122
Requirements Documentation	124
Project Glossary	125
Software Requirement Specification	126
Use Case Report	127
Data Dictionary	129
Requirement Processes and Standards	130
Agile Requirements Processes	131
Business Analysis Body of Knowledge (BABOK)	133
UML Requirements	138
SysML Requirements	139
MDG Link for DOORS	140
Getting Started	142
Create a Link to a DOORS Module	143
Export Requirements to DOORS	147
Import Requirements from DOORS	152
Additional Requirements Tools	157
Auto Names and Counters	158
Import and Export Spreadsheets	160
Requirements Checklist	162
Documentation	164
Glossary	166
Auditing	168
Element Discussions	170
Maintenance Items	172
Team Reviews	173

Requirements

The field of requirements engineering is one of the most critical disciplines in the solution development lifecycle and has a documented impact on the success of projects.

Requirements Development

This comprises all the activities and tasks associated with discovering, evaluating, recording, documenting and validating the requirements for a particular project. Requirements are discovered, analyzed, specified and verified. Enterprise Architect has a wide range of tools and features to assist the analyst as they develop requirements. The center piece for requirements development is the **Specification Manager**, through which the requirements analyst can enter, view and manage requirements in textual form as if in a spread sheet. Requirement properties such as Status, Priority and Author can be edited in-line, and filters can be applied to restrict the display to particular requirements.

Item

1 REQ019 - Manage Inventory

The system **MUST** include a complete inventory management facility to store and track stock of books for the on-line bookstore.

1.1 REQ122 - Inventory Reports

Inventory reports are required that detail the available stock for each item including back orders. Future stock level reports should be able to predict the quantity of stock at a specified future date.

1.2 REQ023 - Store and Manage Books

A book storage and management facility will be required.

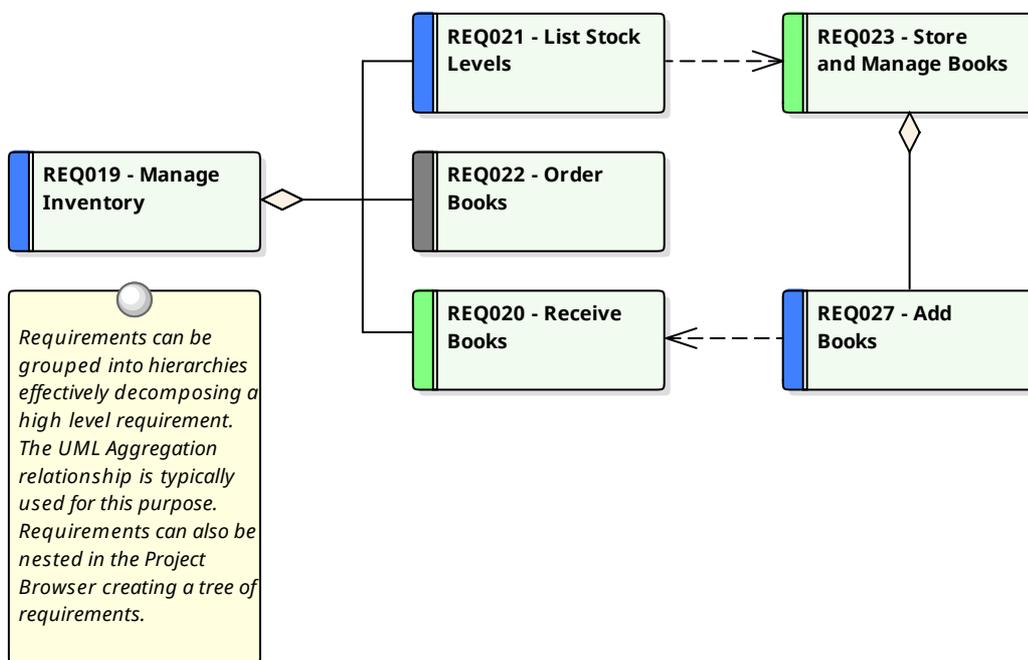
1.2.1 REQ022 - Order Books

A book order facility will be required to allow on-line ordering from major stockist's.

1.2.2 REQ021 - List Stock Levels

A facility will exist to list current stock levels and to manually update stock quantities if physical checking reveals inconsistencies.

The Specification Manager can be used in conjunction with a platform of other tools such diagrams, the Traceability window and the Element Discussions facility.



Requirements Management

This comprises the activities to maintain a set of requirements that represent an accord or agreement between the project team and the customer. It also has a focus on ensuring that the requirements are acceptable to the design and development teams, and that they are sufficiently specific to be implemented into working business, software or hardware systems. Enterprise Architect is a sophisticated platform for managing requirements, and regardless of the domain, the size of the project or the method being followed it provides tools that make it straight forward to manage even large repositories of requirements in complex projects.

Elements by Status



This diagram shows a Bar Chart element depicting element status for all the requirements in a selected package. It provides a useful summary for a requirements manager and is dynamically updated when the status changes and the diagram is reopened. There are a range of other pre-defined charts and user defined charts can also be added.

Requirement Documentation

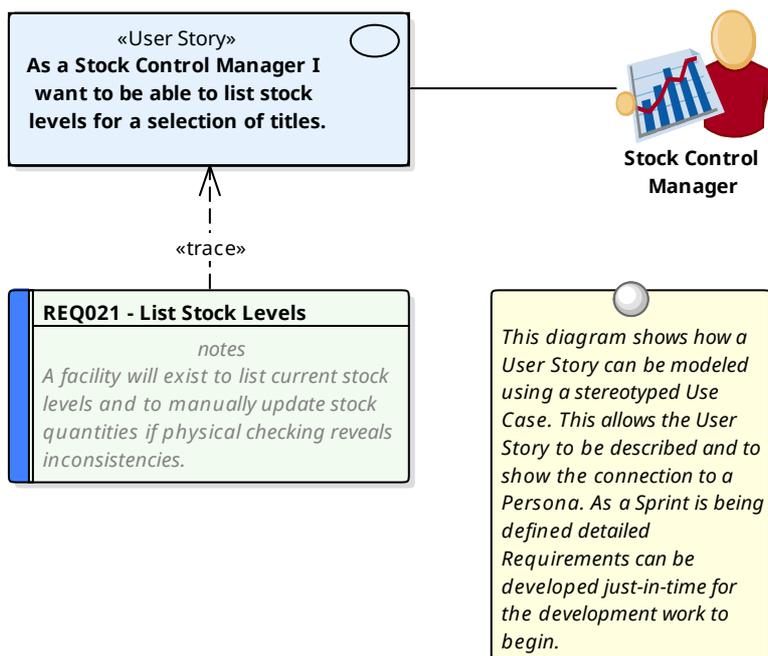
There are a number of documents that are commonly produced as part of the requirements engineering

discipline such as the Software (System) Requirements Specification and Use Case Reports and these can be generated automatically from a requirements model using built-in templates. In addition a wide range of other documents can be produced using built-in or customized templates.



Requirement Processes and Standards

Enterprise Architect is an open platform that supports any requirements engineering process. The tool has a rich feature set and is highly configurable, and its flexible design means that whatever method is being used the modeler will find features to help. So whether the team is using Formal requirements, Use Cases, User Stories or Story-Boards in any combination, Enterprise Architect can be used to develop, manage and document the requirements. The implementation of the UML extension mechanisms means that any type of requirement can be created and managed using built-in types or by using stereotyped elements and **Tagged Values**.



Introduction

Requirements engineering is one of the most important disciplines in the system lifecycle and when done well it will set the foundation for a successful project or program of work, ultimately ensuring that great value is delivered to the users and other stakeholders. Enterprise Architect is a sophisticated and intuitive platform for developing and managing requirements from modeling stakeholders and visions, business cases, business drivers and capabilities to detailed functional and non-functional requirements. Requirements can be prioritized, traced and tracked, and changes can be recorded, baselined, versioned and audited. Analysts can work together in a collaborative platform with role based Security, **Discussions**, Team Reviews, **Model Mail** and a range of other tools to encourage best practice and productivity.

How it will help you

Readers will typically come to the topic of requirements engineering with some existing knowledge or experience even if it is something that has been learnt in lectures or by on the job training, or perhaps by using a different tool. The readers will benefit by understanding the product features and the tools that are available to develop and manage requirements in Enterprise Architect and this will enable them to be more productive as an individual and also as a member of a team.

Who will benefit

Anyone involved in the development or management of requirements whether at a strategic level, a business value level or a system development level will benefit from reading this information. This includes a wide range of roles including Strategic Thinkers, Business and Requirements Analysts, Enterprise, Business, Technical and Solution Architects, Project and Program Managers, Developers, Test Desingers and User Experience Designers.

What you will learn

This topic will teach you how to use the powerful features of Enterprise Architect to develop and manage Requirements, to create documentation and to work collaboratively as a member of a team using a formal or informal system life cycle process or standard.

Overview of the Documentation

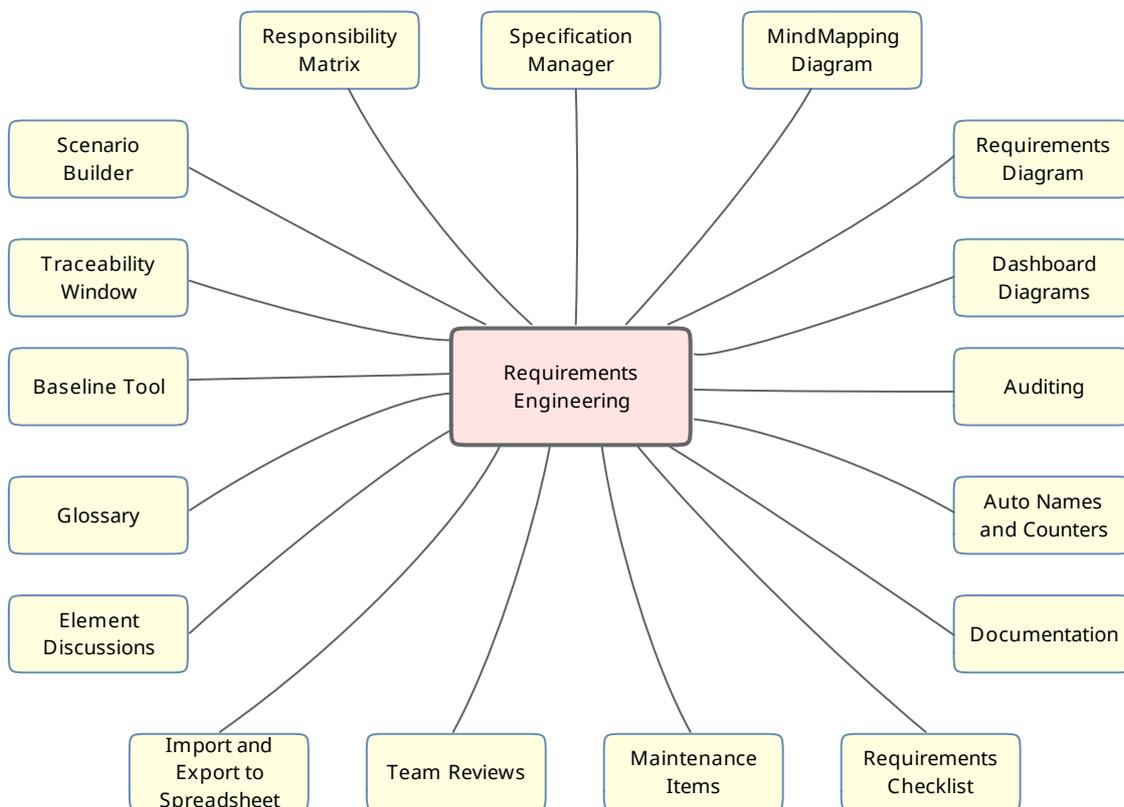
Meet the Requirements Tools	Lists the key tools that are used for developing and managing requirements, including a picture of the tool in action, where to find the tool, how to use it and how to become proficient in using the tool. There are a large number of additional useful tools that are described in the last topic, <i>Additional Requirements Tools</i> .
Requirements Overview	Puts requirements engineering in context by defining what requirements are, the different levels of requirements, characteristics of good requirements and the business context of requirements. The information also includes the concept of a Requirements diagram that readers coming from text based tools may not be familiar with, and how to create and view requirements in Enterprise Architect.
Requirements	Covers the activities and tasks associated with discovering, evaluating,

Development	recording, documenting and validating the requirements. The topic is conveniently divided into four sub-topics Elicitation, Analysis, Specification and Validation and covers a wide range of features that can be used from Mind Mapping diagrams for recording information in elicitation workshops to the Specification Manager for creating requirements to Test Cases for Validating them.
Requirements Management	Describes the activities needed to maintain a set of requirements that represent an accord or agreement between the project team and the customer. It includes composing hierarchies of requirements, tracing other elements back to requirements, tracking the properties of requirements including Status, Priority, and Difficulty. It also describes managing changing requirements, Volatility and assessing the impact of changing requirements.
Requirements Documentation	Describes how formal and informal Requirements Documentation can be generated directly from Enterprise Architect using a series of predefined and extensible templates. This includes Glossaries, Data Dictionaries, Use Case Reports and Documents such as a System Requirements Specification
Requirement Processes and Standards	Puts the usage of Enterprise Architects requirements tools in the context of system and requirements processes and standards. The topic describes how flexible the tools are and how they can be used with any process or standard.
Additional Requirements Tools	Lists a series of additional tools that can be used for requirements engineering, including a picture of the tool in action, where to find the tool, how to use it and how to become proficient in using the tool. There are a number of key tools that are the most important tools for requirement engineering that are described in the first topic entitled <i>Meet the Requirements Tools</i> .

Meet the Requirement Tools

Enterprise Architect is a sophisticated and flexible Requirements modeling tool that can be used across the requirements life cycle from planning through to product support. The tool can be used with any Requirements management process and there are a wide range of features that allow analysts to work using their preferred methods such as Word Processor views, Spreadsheet views, Diagrams, **Relationship Matrix** or a range of other core and extended features.

This Mind Map shows the landscape of the key Requirements tools that can be used to develop and manage requirements. While these are the primary tools there are a series of other tools described in the topic *Additional Requirements Tools*.



Specification Manager

Getting to Know the Specification Manager

Introducing the Specification Manager

The **Specification Manager** is the central tool for working with requirements and provides a Word Processor or Spreadsheet like interface for entering, maintaining and viewing Requirements. New Requirements can be created with names and detailed descriptions and properties such as Status and Priority can be added from drop-down lists. Existing Requirements can be viewed and managed in a convenient view and changing them in the Specification Manager will change them in all other places in the repository such as diagrams and windows. It is the perfect tool for those analysts more comfortable working with text rather than diagrams and who are accustomed to working in a Word Processor or Spreadsheet. It has the added advantage that the requirements are part of a model and can be traced to other elements including Business Drivers, Stakeholders and Solution Components.

Item

1 REQ019 - Manage Inventory

The system **MUST** include a complete inventory management facility to store and track stock of books for the on-line bookstore.

1.1 REQ122 - Inventory Reports

Inventory reports are required that detail the available stock for each item including back orders. Future stock level reports should be able to predict the quantity of stock at a specified future date.

1.2 REQ023 - Store and Manage Books

A book storage and management facility will be required.

1.2.1 REQ022 - Order Books

A book order facility will be required to allow on-line ordering from major stockist's.

1.2.2 REQ021 - List Stock Levels

A facility will exist to list current stock levels and to manually update stock quantities if physical checking reveals inconsistencies.

Where to find the Specification Manager

Main Menu: Tools | **Specification Manager**
 Project Browser Context Menu (Package): View as Requirements List

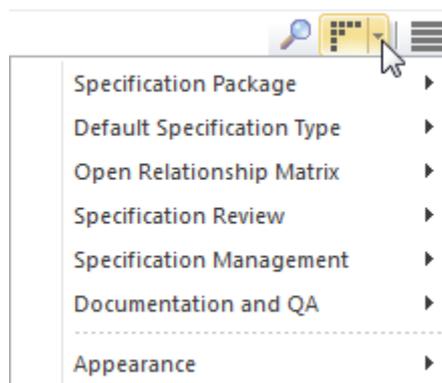
Usage of the Specification Manager

To create, view and maintain Requirements in a text based tool that resembles working in a word processor or spreadsheet. Details can be added to the requirements and requirement properties can be added from drop-down lists. When the requirements are changed in this specification manager the changes are conveniently reflected in the project browser and

all other windows.

Options for the Specification Manager

There are a wide range of options available from the options menu, to tailor the way you use the **Specification Manager**. These include Level (hierarchical) Numbering, Auto Naming, Spell Check, Documentation, Import and Export of Requirements, access to various related tools and more.



Learn more about the Specification Manager

[The Specification Manager](#)

Relationship Matrix

Getting to Know the Relationship Matrix

Introducing the Relationship Matrix

The **Relationship Matrix** is a tool for visualizing the way that requirements are related to each other and to other elements in the model in a visually compelling matrix or spreadsheet like view. It can be used to view the relationship between Stakeholders and their Requirements, how Use Cases are related to Business Requirements or Functional Requirements, how Capabilities are related to Business Drivers or which Components implement a set of requirements and more. Any number of matrices can be defined quickly and then saved to be viewed in workshops or included in documentation generated automatically from the model or exported to a spreadsheet file. When a matrix is created, connections can be viewed by placing the Requirements on one axis of the matrix and the connected elements on the other axis then the squares of the matrix will indicate the direction of the relationship.

Target +	REQ011 - Manage User Accounts	REQ012 - Provide Online Sales	REQ013 - Manage Deliveries	REQ014 - ShoppingBasket	REQ015 - Process Credit Card Payment	REQ016 - Add Users	REQ017 - Remove User	REQ018 - Report on User Account	REQ019 - Manage Inventory	REQ020 - Receive Books	REQ021 - List Stock Levels	REQ022 - Order Books
+ Source												
Add New Titles												
Add To Shopping Basket				↑								
Close Account							↑					
Create Account						↑						
Create Orders												↑
Delete User							↑					

Where to find the Relationship Matrix

Main Menu: Tools | **Relationship Matrix**

Project Browser Context Menu (Package): Relationship Matrix | As Source.../As Target.../As Both...

Resources Window: Matrix Profiles

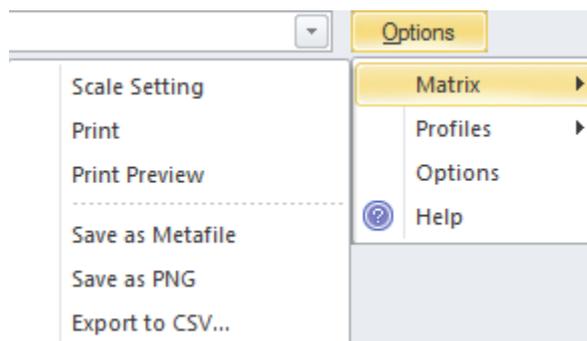
Usage of the Relationship Matrix

To display the relationships that exist between elements in two Packages in a visually compelling matrix such as which requirements are realized by which Use Cases. It is useful in analyzing missing elements or relationships for example to determine which Requirements are not realized by any Use Case or which Components do not have corresponding Requirements or Use Cases. It is particularly useful in workshops with Business Stakeholders who may not be familiar with

seeing requirements in Trace diagrams.

Options for the Relationship Matrix

There are a range of options that can be set for the matrix including saving it to the Resources window or to a CSV format for opening in a spreadsheet. The appearance of the matrix can also be altered by sorting the elements, showing an outline numbering view, and suppressing Package names. These items are available from the **Options button** on the **Relationship Matrix** window.



Learn more about the Relationship Matrix

[Relationship Matrix](#)

Requirement Properties

Getting to Know the Requirement Properties

Introducing Requirement Properties

Requirement Properties are used to define metadata about the requirement that is useful for the management of requirements for the purpose of prioritization and defining work packages for the implementation teams. All Enterprise Architect elements have standard properties such as Status, Author and Phase but the Requirement element has additional properties such as Difficulty and Priority. User defined properties can also be defined using **Tagged Values**.

Type:	Functional
Status:	Approved
Alias:	
Keywords:	
Author:	Paulene Dean
Difficulty:	Medium
Priority:	High
Version:	1.0
Phase:	1.0

Where to find Requirement Properties

Main Menu: Element | Properties...

Main Menu: Element | Properties Page | General, Requirements, Constraints, Scenarios, Files, Links, **Tagged Values**

Main Menu: View | Properties

or

Element Context Menu: Properties

or

Project Browser Context Menu | Properties...

Project Browser Context Menu | Properties Page | General, Requirements, Constraints, Scenarios, Files, Links, Tagged Values

Usage of the Requirement Properties

To define the important meta information about a requirement for the purposes of providing data to manage the requirements for prioritization, understanding which are the difficult requirements, and managing the lifecycle by using Status to determine requirements for implementation Packages.

Options for Requirement Properties

Enterprise Architect has a wide range of built in properties for all elements but it has a number of additional properties for requirements. If there are other properties that are needed by a modeler or team such as the volatility

(stability) of a requirement these can be added using the general purpose UML extension mechanism of **Tagged Values**.

REQ021 - List Stock Levels
<i>tags</i> Volatility = Medium
<i>notes</i> <i>A facility will exist to list current stock levels and to manually update stock quantities if physical checking reveals inconsistencies.</i>

**Learn more about
Requirement Properties**

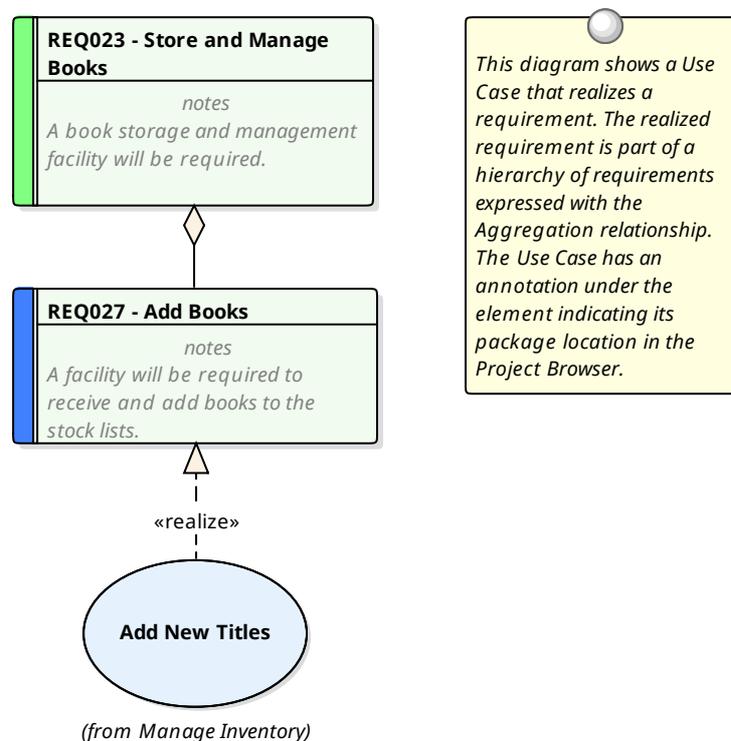
[Properties Dialog](#)

Requirements Diagram

Getting to Know the Requirements Diagram

Introducing the Requirements Diagram

The **Requirements Diagram** provides a visual representation of how Requirements are related to each other and to other elements in the model, including Business Drivers, Constraints, Business Rules, Use Cases, User Stories, design Components and more. The diagram is one of Enterprise Architect's extended diagram types and for analysts who are accustomed to working with requirements in a text based tool it will provide a welcomed and compelling graphical representation of the requirements.



Where to find the Requirements Diagram

Main Menu: Diagram | New... | Extended | Requirements

Project Browser Context Menu: Add Diagram... | Extended | Requirements

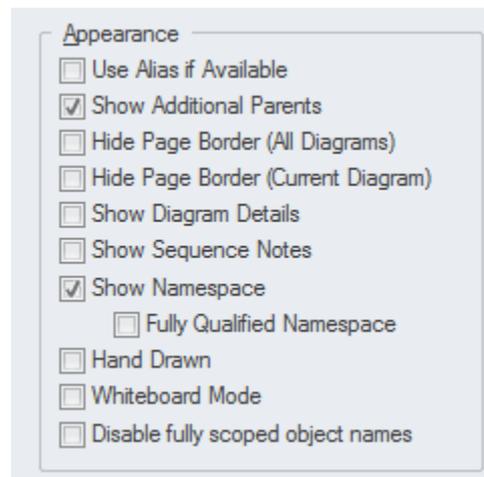
Usage of the Requirements Diagram

One usage is to show how Requirements are connected together in a hierarchy but a more compelling usage is to show how requirements are connected to other elements. The experienced modeler will define and manage the requirements in the **Specification Manager** and then use the requirements diagram to show how one or more requirements are related to up-process elements such as Business Drivers and down-process elements such as Use Cases, User Stories, User Experience designs and solution Components.

Options for the Requirements Diagram

The appearance of a diagram can be changed to suit the audience, and details can be included, suppressed or altered to ensure the diagram meets its main objective of communication. There is a wide range of options ranging from creating a Hand Drawn style of diagram to page

setup.



Learn more about the Requirements Diagram

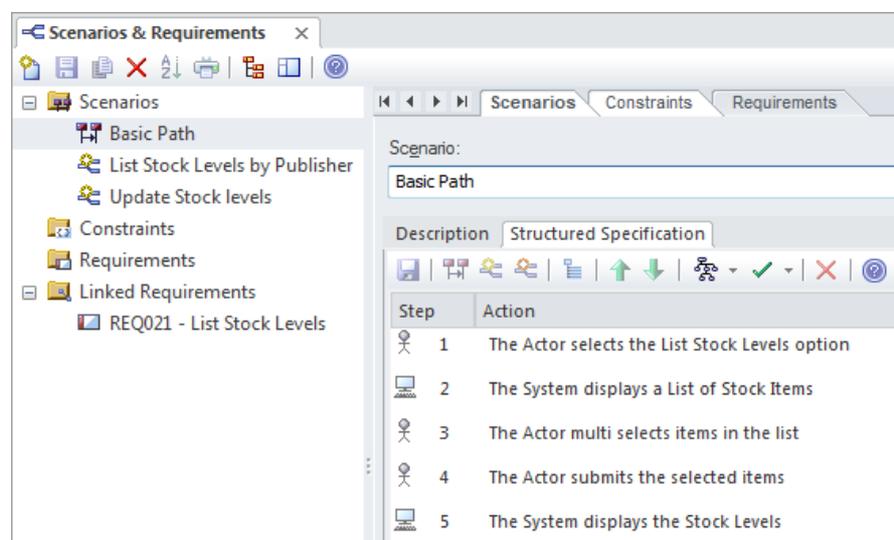
- [Working In Diagrams](#)

Scenario Builder

Getting to Know the Scenario Builder

Introducing the Scenario Builder

The **Scenario Builder** is used to define the details of a Use Case including defining detailed descriptions, creating one or more Scenarios and defining pre-conditions, post-conditions and other constraints. The detailed steps of a Use Case can be recorded and linked to other elements in the model and these can then be generated out as a diagram providing a visual representation of the Use Case and its Scenarios. The diagram and the text can be synchronized and individual steps can then be traced to other elements such as Components that will realize the Requirement specified in the Use Case.



Where to find the Scenario Builder

Main Menu: Tools | **Scenario Builder**

Main Menu: Elements | Properties Page | Scenarios

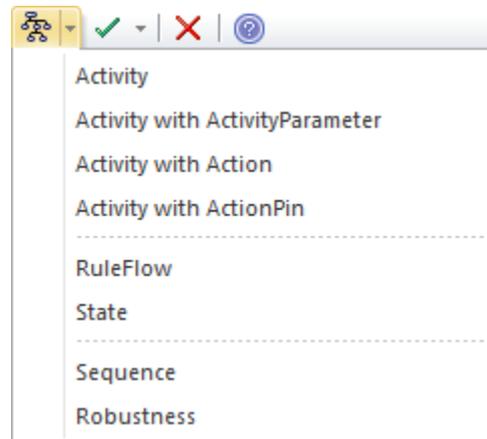
Element Context Menu: Properties | Scenarios

Usage of the Scenario Builder

To define the details of a Use Case and its scenarios and constraints which can be used to replace the traditional text-document based approach to defining Use Cases. This ensures that the Use Case diagram and the textual details of the Use Cases and its Scenarios and Constraints are all contained in the same model and can be traced. If the Use Cases are required in a document format for contractual or process reasons a Use Case Report can be generated automatically from the models using the in-built documentation engine.

Options for the Scenario Builder

The **Scenario Builder** can be viewed as a tabbed or a docked window or in an element's **Properties window**. The steps of a Use Case including its Scenarios can be automatically generated into a number of different diagram types available from the Generate Diagram toolbar icon.



**Learn more about the
Scenario Builder**

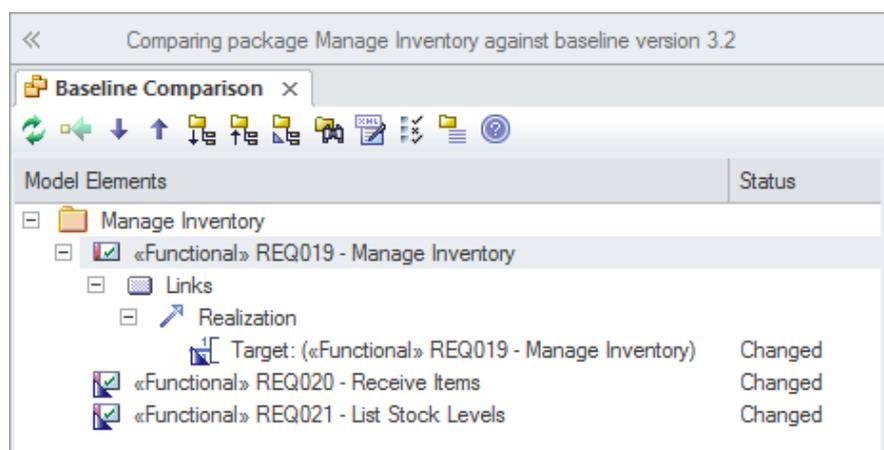
[Element Scenarios & Requirements](#)

Baseline Tool

Getting to Know the Baseline Tool

Introducing the Baseline Tool

The Baseline Tool can capture a snapshot of the Requirements at a point in time and then at a later time the repository can be compared to this (or another baseline) for the purpose of determining what has changed. Any number of baselines can be created and labeled and there is a baseline comparison tool which displays the differences between the baseline and the model and allows the modeler to revert a change in the model to a baseline at a granular level.



Where to find the Baseline Tool

Main Menu: Package | **Baselines...**

Project Browser Context Menu: **Package Control** | Package Baselines...

Usage of the Baseline Tool

Baselines are also useful when a formal requirements process is being followed or the Requirements form part of a contract, as the baseline can keep a snapshot of the requirements at important milestones such as contract signing or requirement phase sign off. This is also applicable to iterative and incremental processes such as Agile methods, as the requirements can be baselined before or even after a Sprint. When Requirements are still volatile and the Requirements' owners are still formulating their needs, a baseline can be created to take a snapshot at important points in the analysis phase, such as after an elicitation workshop.

Options for the Baseline Tool

There are several options that can be applied to configure the way the Baseline Compare tool presents information; these are available from the **Options button** on the **Baselines** window.

Always Expand to Differences

Show Elements that are:

- Changed
- In Baseline Only
- In Model Only
- Unchanged Items

Suppress these Changes:

- Suppress Diagrams
- Suppress Date Modified
- Suppress Date Created
- Suppress Children of Missing Items
- Suppress Advanced Properties

Baseline Diagram Compare Options

- Always open first parent with a Baseline
- When comparing from the Project Browser or a Diagram

**Learn more about the
Baseline Tool**

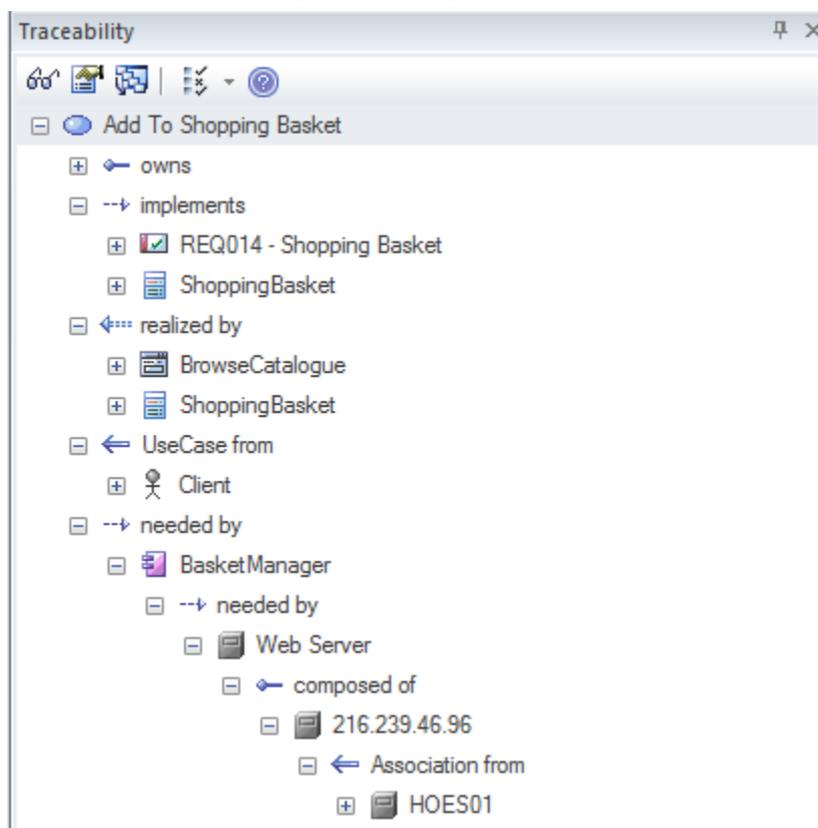
[Package Baselines](#)

Traceability Window

Getting to Know the Traceability Window

Introducing the Traceability Window

The **Traceability window** gives the modeler a hierarchical view of element connections, allowing traceability to be visualized and queried as elements are traversed in the model. This tool is particularly powerful because a modeler will often choose to hide diagram relationships but by selecting an element in the diagram and viewing its connections in the traceability window all its relationships will be displayed.



Where to find the Traceability Window

Main Menu: View | Traceability

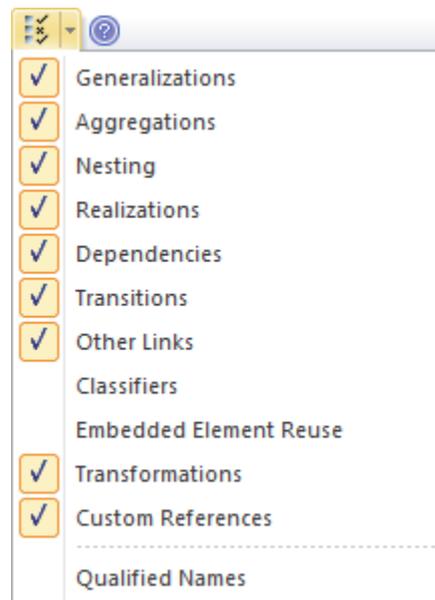
Usage of the Traceability Window

The Traceability Window can be used to view the way an element is connected to other elements in the repository in a hierarchy including the types of the relationships. This window gives a complete list of all relationships which cannot be seen by viewing elements in the **Project Browser** and may not appear in any diagrams. It is very useful for managing requirements and tracing how a requirement is related to up-process elements such as Business Drivers and down-process elements such as Components. It is a useful tool for newcomers to a model to gain a quick understanding of which are the important and well connected elements. It should be viewed before deleting an element in the model to ensure that the user understands the element's existing relationships.

Options for the Traceability Window

There are a series of options that restrict the traceability to specified connector types; these can be set to alter what is displayed in the window.

The options are available from the toolbar at the top of the window.



**Learn more about the
Traceability Window**

[The Traceability Window](#)

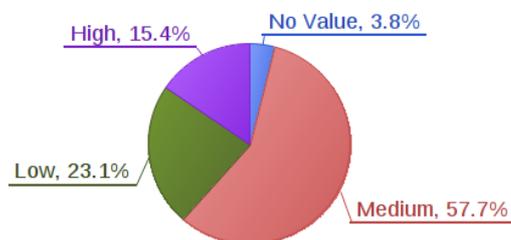
Dashboard Diagrams

Getting to Know the Dashboard Diagrams

Introducing Dashboard Diagrams

The Dashboard Diagrams allow high quality charts and graphs to be created to display repository information in a visually compelling way, such as the ratio of Requirement Priorities in a pie chart. There is a toolbox page of pre-configured charts and graphs, but the user is free to create and save any number of charts sourcing data from anywhere in the repository. The charts and graphs provide valuable summary information that assists in the management of requirements. High level reporting and project status can be easily tracked and documented using the numerous charts and report elements available that tightly link in with the model content and status.

Elements by Priority



This diagram shows a Pie Chart element depicting element priority for all the requirements in a selected package. It provides a useful summary for a requirements manager and is dynamically updated when the Priority changes and the diagram is reopened. There are a range of other pre-defined charts and user defined charts can also be added. A filter has been added to exclude all elements other than Requirements.

Where to find Dashboard Diagrams

Main Menu: Diagram | New... | Extended | Dashboard

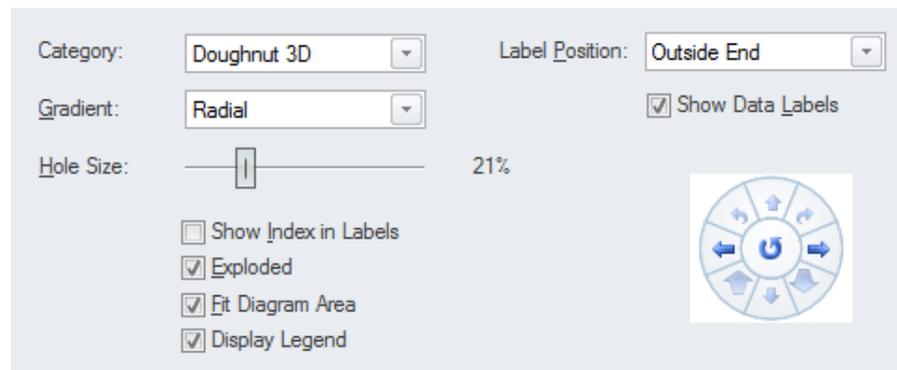
Project Browser Context Menu: Add Diagram... | Extended | Dashboard

Usage of Dashboard Diagrams

Dashboard Diagrams present compelling views of information such as the Status of Requirements in a particular release of the system that can be viewed inside the model or conveniently copied directly into management or project team presentations. They are useful for planning an iteration such as an Agile sprint to view how ready the requirements are for the implementation team; for example to view what percentage of the Requirements have been approved and are of high priority.

Options for Dashboard Diagrams

The Standard Charts and Graphs available from the toolbox can be configured in a number of ways, including changing the source, applying filters or modifying the appearance of the chart as indicated in this diagram, available from the chart's **Properties window** using the 'Appearance' section.



**Learn more about
Dashboard Diagrams**

[Charts](#)

The MDG Link for DOORS Add-In

The Model Driven Generator (MDG) Link™ for DOORS Add-In provides support for linking an Enterprise Architect model to an IBM® Rational® DOORS® requirements package. This tool is very useful when you need to perform your requirements management external to your Model Driven Development. Using this Add-in you can interchange requirements defined within IBM DOORS with the traceable Requirements used within Enterprise Architect's Requirements Management features.

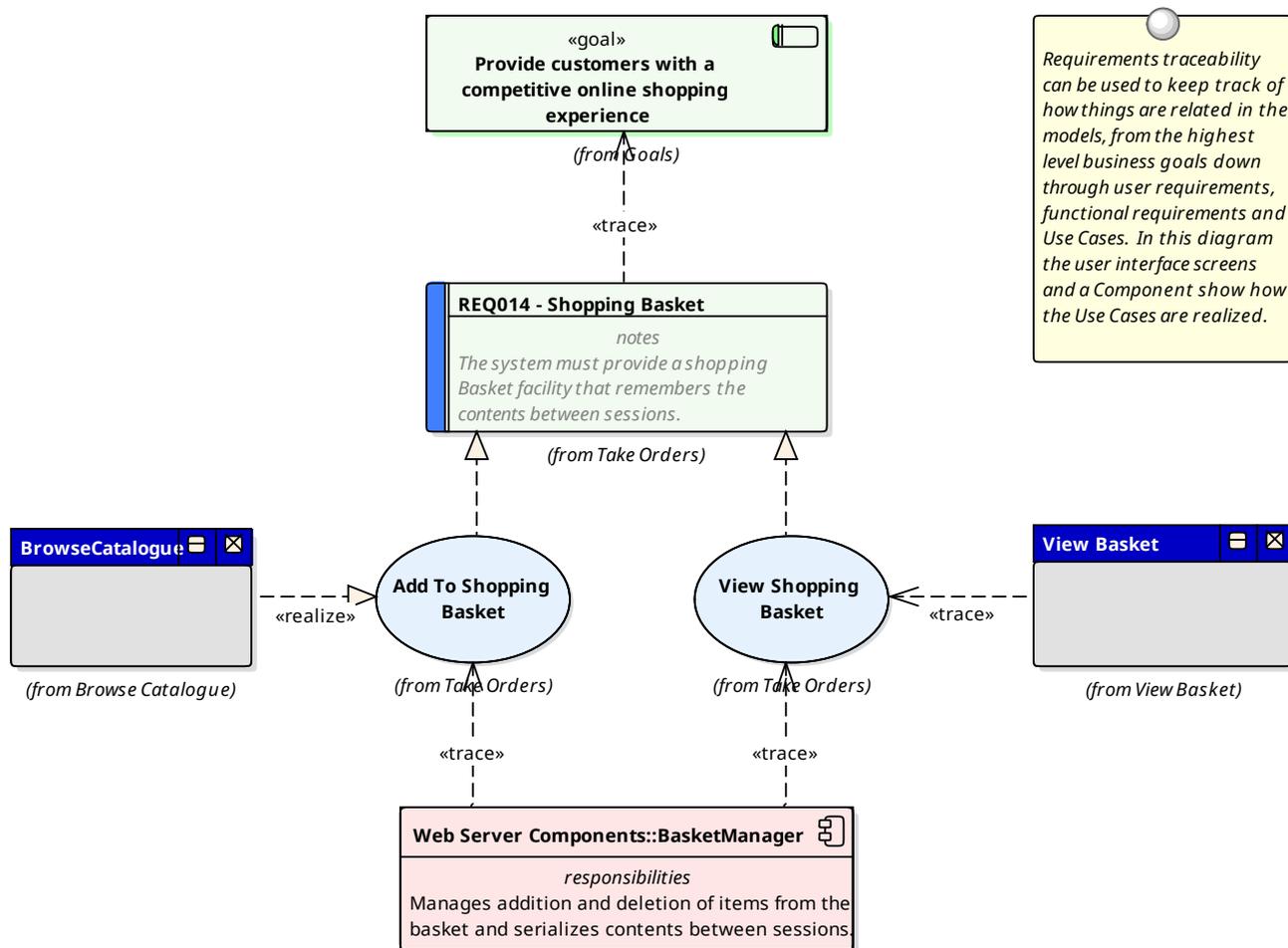
Requirements Overview

The discipline of requirements engineering is one of the most critical parts of any system development process. Whether it is an enterprise, business, engineering, real-time, software or hardware system the definition and management of requirements is critical to the success of any endeavor. Clear and unambiguous articulation of requirements will ensure that the implementation team has the problem defined and this gives the best chance of the correct solution being implemented. Enterprise Architect equips the requirements analyst and manager with a formidable set of tools to take on this important challenge.

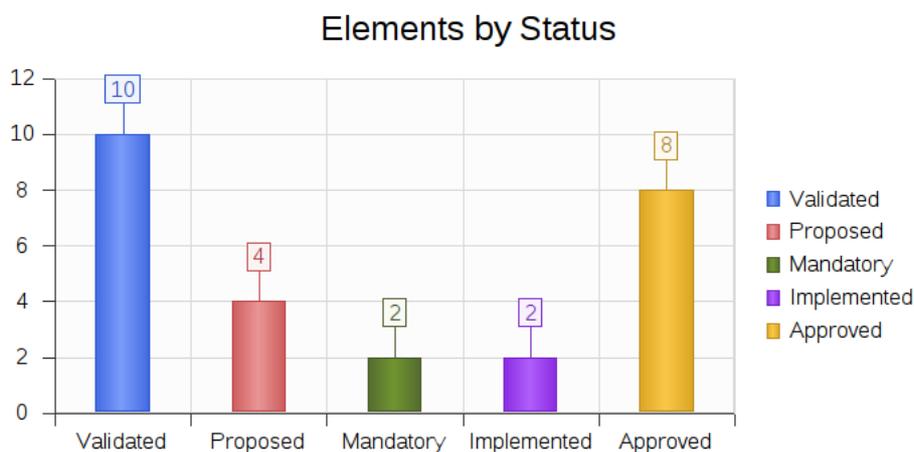
The powerful **Specification Manager** allows requirements to be entered, viewed and managed in a spread sheet format facilitating rapid input and editing of requirements. Requirement properties including **Tagged Values** can be edited in-line and values can be selected from drop-down lists.

Item	Priority	Status	Difficulty
1 REQ019 - Manage Inventory	Medium	Approved	Medium
The system MUST include a complete inventory management facility to store and track stock of books for the on-line bookstore.			
1.1 REQ122 - Inventory Reports	Medium	Proposed	Medium
Inventory reports are required that detail the available stock for each item including back orders. Future stock level reports should be able to predict the quantity of stock at a specified future date.			
1.2 REQ023 - Store and Manage Books	Low	Mandatory	
A book storage and management facility will be required.			
1.2.1 REQ022 - Order Books	Medium	Implemented	Medium
A book order facility will be required to allow on-line ordering from major stockist's.			
1.2.2 REQ021 - List Stock Levels	Medium	Approved	Medium
A facility will exist to list current stock levels and to manually update stock quantities if physical checking reveals inconsistencies.			

Requirements can be viewed in diagrams and related to other model elements creating compelling representations of traces between specifications and implementations.



Requirements can be managed in a variety of ways from the use of the **Project Gantt Chart** to the Dashboard diagrams that show properties like status for a set of requirements.

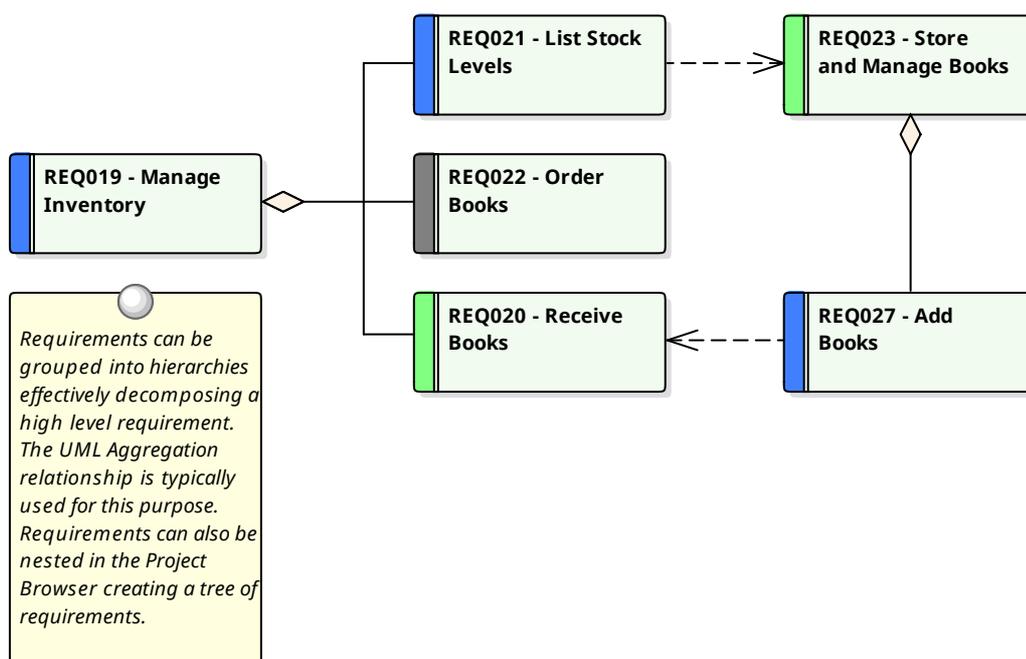


This diagram shows a Bar Chart element depicting element status for all the requirements in a selected package. It provides a useful summary for a requirements manager and is dynamically updated when the status changes and the diagram is reopened. There are a range of other pre-defined charts and user defined charts can also be added.

These are just a small number of the tools that are available to the requirements analyst and managers.

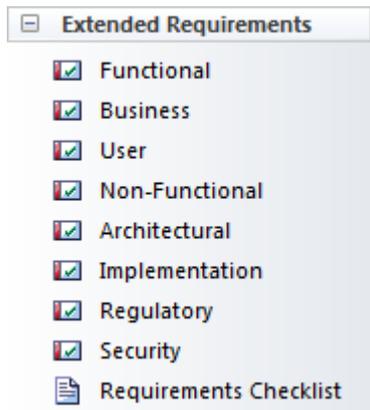
What are Requirements

A requirement is a capability or condition that must be met to ensure that a solution meets the needs of its stakeholders. There is often robust debate about exactly what constitutes a requirement and some proponents will include notions such as Business Drivers and Policies and Business Rules while others will have a much more restrictive view of the requirements. Also a number of requirement methods are Use Case centric and only articulate requirements at a business level while others augment the use cases with detailed functional requirements required by the developers. The highly iterative methods such as Agile typically use User Stories and Requirements together but defer the elaboration of requirements until an iteration (sprint) is being planned. Enterprise Architect provides generic tools to support any requirements method and any type of requirement can be created and managed using built-in types or by using stereotyped elements and **Tagged Values**.



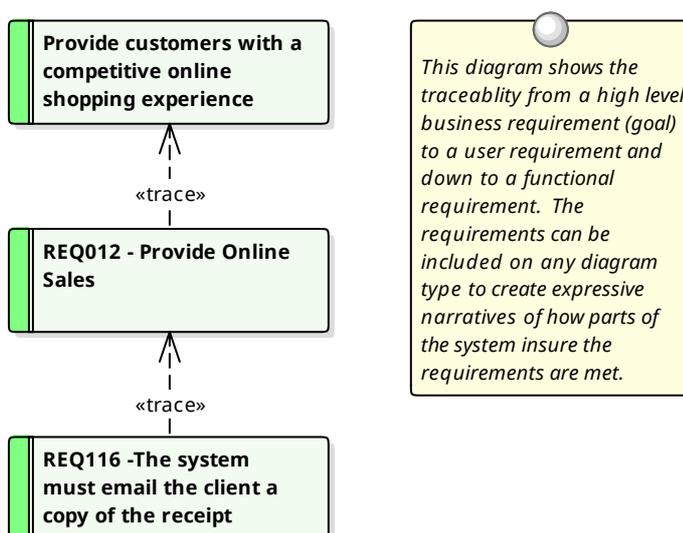
Levels and Types of Requirements

There are many different types of requirements ranging from high level business requirements down to detailed technical requirements that specify an intricate part of a computer algorithm or hardware device. There are also types based on the source of the requirement such as stakeholder requirements or the location in the process such as transition requirements. There is often confusion and debate about exactly what constitutes a Requirement so some teams will define Business Rules and Policies as Requirements and other will view them as business specifications. Regardless of the method or the process that is being followed Enterprise Architect allows the analyst to create sophisticated models of all requirement types.



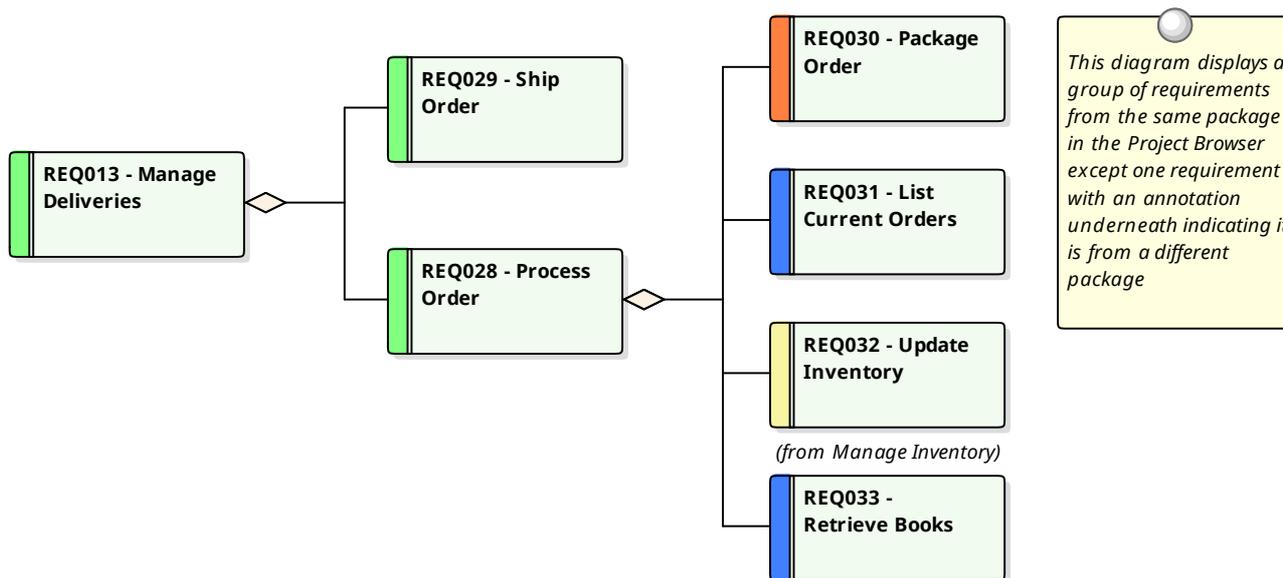
Business Requirements

Business Requirements are high-level requirements that express the objectives and desired outcomes of an organization. They are often disregarded as being 'fluffy' by engineers who cannot see how they would be implemented, but if they are articulated well they can be broken down to measurable statements. They are typically defined in a business case or other statements by the product owner or sponsor, the marketing department or the customer. They attempt to articulate why the organization is spending money and resources on the project. Enterprise architect has a Business Requirement element available from the requirements toolbox page for this purpose.



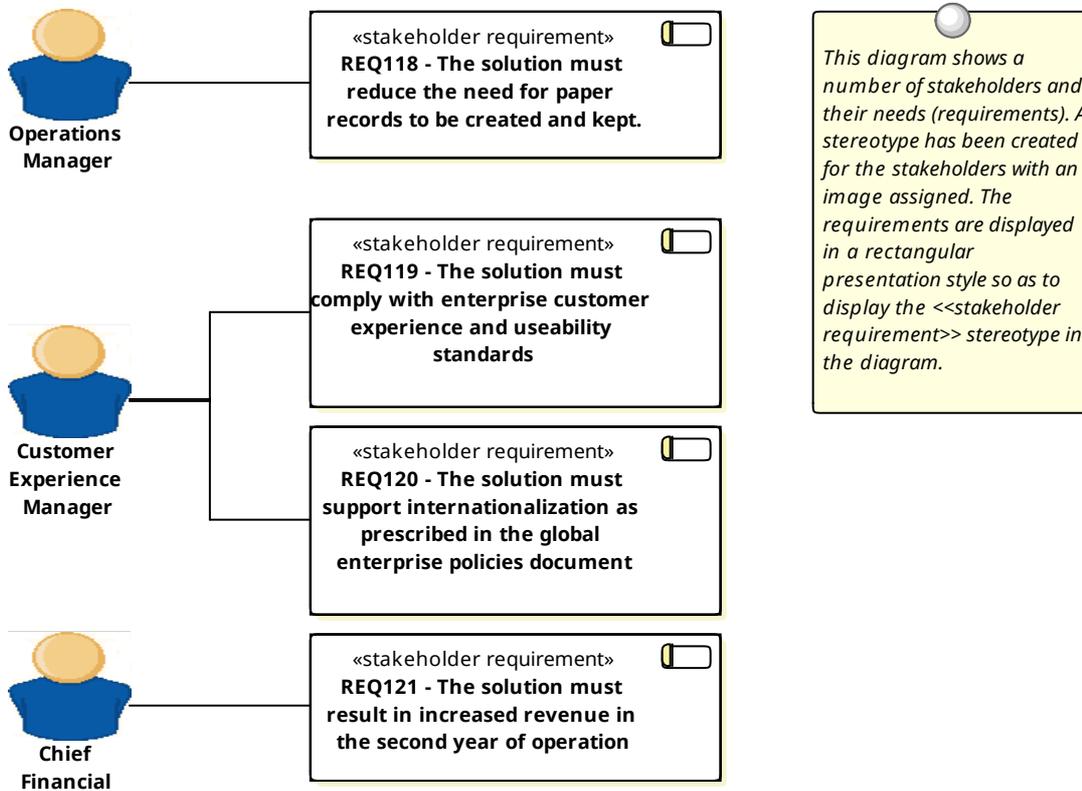
Functional Requirements

Functional Requirements are the bridge between the business and technical teams and provide the definition of what the system must do for its users that will in turn meet the business goals. Some methodologists believe that Functional Requirements can be described using only Use Cases or User Stories but this appears to be a purist view and in practice there seems to be a need for detailed textual Requirements that describe what the architect must design and the developer must implement. Enterprise architect has a Functional Requirement element available from the requirements toolbox page. There is also an Architectural Requirement available from the Extended Requirements page of the Requirements toolbox. In addition there is powerful support for modeling Use Cases and Scenarios using the ingenious **Scenario Builder**.



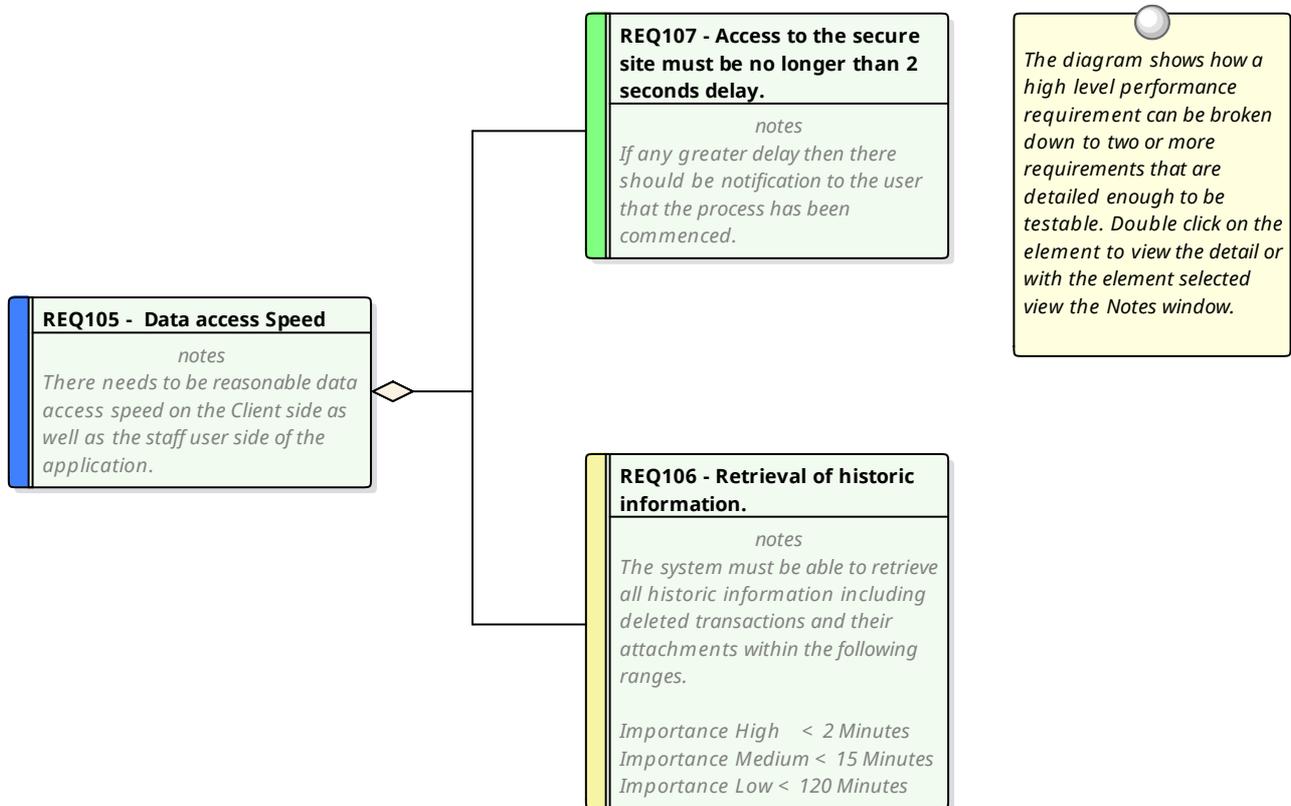
Stakeholder Requirements

Stakeholder Requirements are statements of the stakeholders' needs and expectations and describe the features that must be met if the business requirements are to be fulfilled. Analysts tend to focus on the functional aspects of the needs but stakeholders' expectations might include performance and reliability and a variety of other non-functional needs. Both are critical and act as precursors to the definition of the functional and non functional requirements that will be consumed by the designers and implementers to create solutions that meet the customer's expectations. Enterprise architect has a Requirement element that can be stereotyped to <<stakeholder requirement>> available from the requirements toolbox page for this purpose.



Non Functional Requirements

Non Functional Requirements and Quality Attributes describe how well a system will perform when it is operating. These typically define or constrain how the system should behave as a whole and include attributes like how well it performs, how secure it is, how many times it develops a fault and how easily it can be extended.

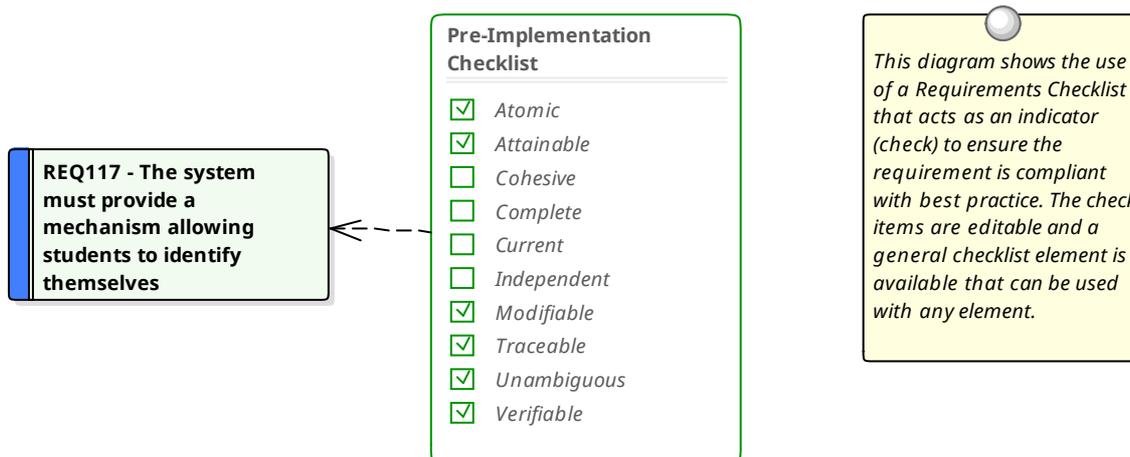


Transition Requirements

Transition Requirements define what is needed to transform the business and systems from the current state to the future state. They define a transitory situation and once the system has been fully implemented the requirements and their implementation will not be visible. They define things such as training, conversion and reformatting of data and parallel runs of business and technology systems

Characteristics of Good Requirements

More often than not errors and deficiencies in systems can be traced back to requirements engineering and much of the literature talks about the small cost of correcting a requirement compared to the large cost of correcting the system once it is built. Well articulated, managed and tested Requirements are therefore imperative to any system development process. Enterprise Architect has a convenient Requirements Checklist element available from the Extended Requirements Page of the Requirements Toolbox.



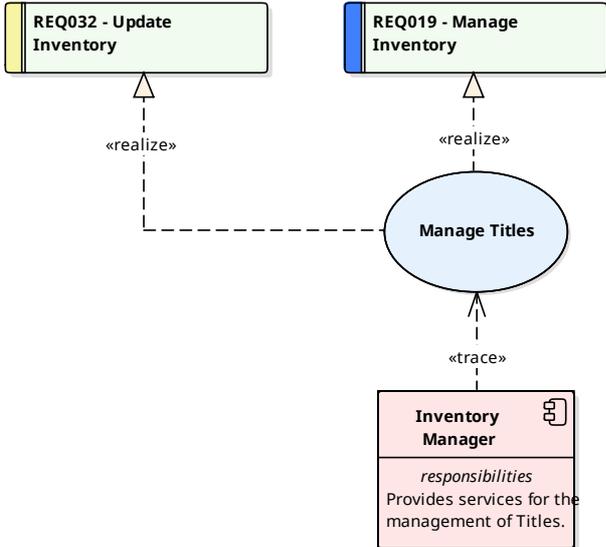
Checklist can be used to indicate if a Requirement is ready for implementation.

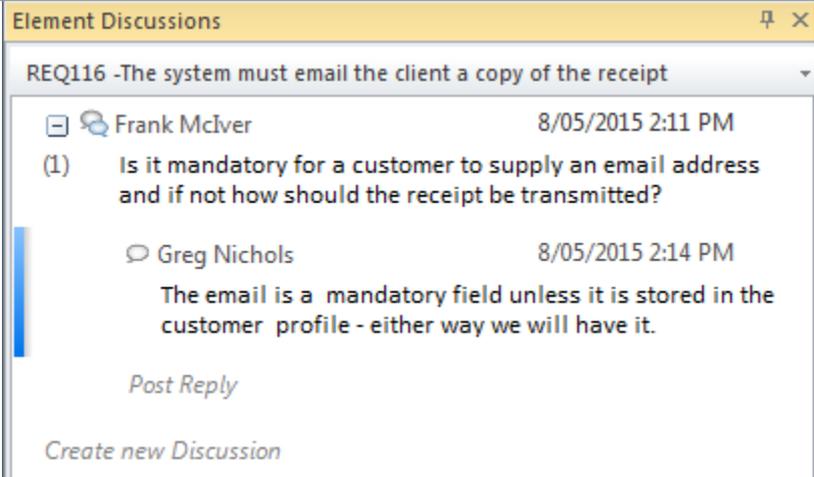
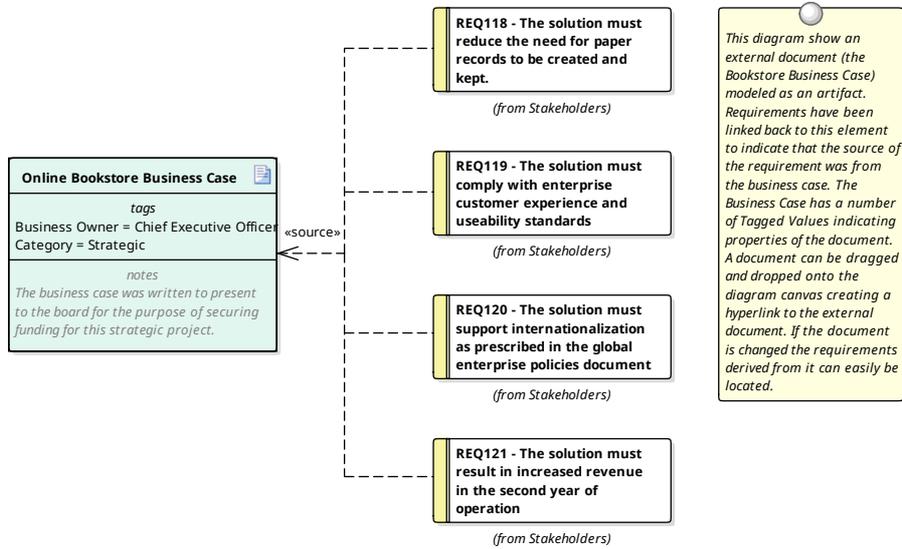
The

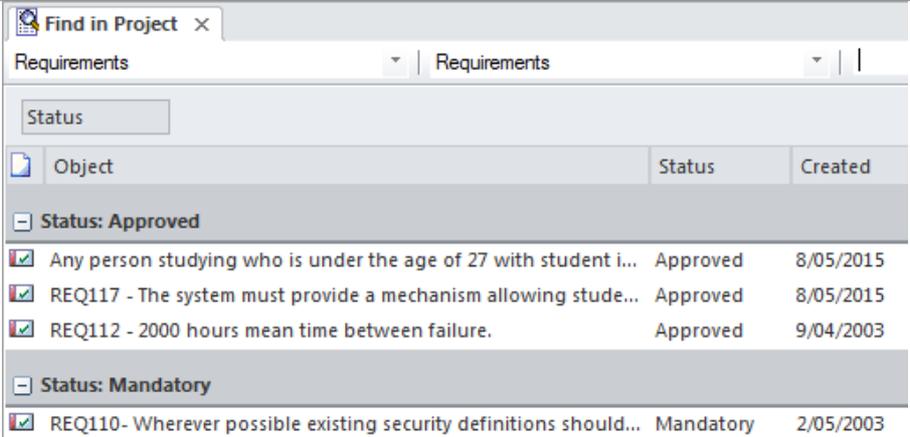
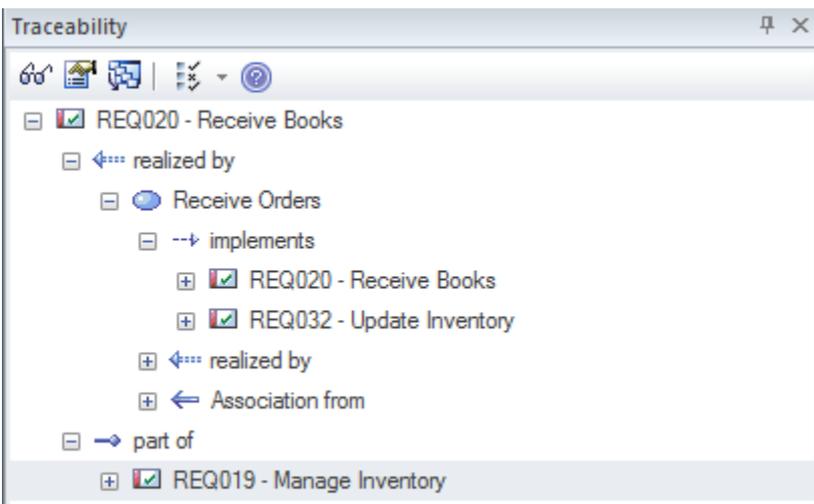
Qualities of Good Requirements

To be effective a set of Requirements must be complete and fully record the stakeholders' needs in a consistent, cohesive and unambiguous way. Enterprise Architect provides an extensive set of features and tools for helping the analyst produce sets of Requirements that are of high quality.

Quality	Description
Atomic	<p>A requirement should articulate a single stakeholder need or a quality attribute. When a requirement contains multiple needs it is not possible to reason about the needs independently. Enterprise Architect can assist by allowing modelers to create hierarchies of requirements in the Project Browser allowing them to be broken down to an atomic requirement.</p>
Attainable	<p>The need specified in the requirement must be able to be achieved. If a requirement is not attainable the system will not be able to deliver the business value required by the stakeholders. Enterprise Architect can assist</p>

	<p>by allowing each requirement to be traced to an implementation element such as a Use Case or a Component. A Relationship Matrix can be used to quickly identify those requirements that are not traced to a lower level element.</p>  <p><i>This diagram shows the expressive power of putting disparate elements onto a diagram to show the traceability between different layers of a system. The traceability can be from the Requirements to the Use Cases that Realize them to the logical Components that will deliver the required functionality and more.</i></p>
<p>Cohesive</p>	<p>The requirements as a set must be consistent and cohesive and express the behavior of the system; any gaps must be determined and overlap between requirements must be resolved. Following a requirements process will assist greatly and Enterprise Architect has a number of facilities that will make it easy to keep the requirements cohesive. Missing requirements can be identified using the Relationship Matrix where for example a matrix between stakeholders and their requirements would quickly identify stakeholders who didn't have requirements.</p>
<p>Complete</p>	<p>Each requirement must fully describe the necessary functionality or behavior that will result in the stakeholder's need being met. Enterprise Architect can help by team members using the Team Review facility or the Element Discussion window. Some analysts like to mark requirements as needing to be completed by appending the requirement with a tag such as 'TBC'. Enterprise Architect can assist by allowing the analyst to search across the requirements Packages for this tag and return a list of elements that require further work. A Model View could also be set up using this search to populate the view. The Element Discussion window is also helpful because the information added is not part of the requirement itself and does not contaminate the requirement's notes with text that isn't part of the Requirements definition.</p>

	
<p>Current</p>	<p>A Requirement must be up-to-date and reflect the current knowledge and project status. Enterprise Architect can assist the analyst by allowing the sources of requirements to be modeled and the requirements themselves can be traced back to these artifacts so when the source is changed all the affected elements could be located.</p> 
<p>Independent</p>	<p>The requirements should be independent of each other and not have overlapping statements that conflict with each other or restate the same need. A degree of analysis will be required as there will inevitably be some overlap but this can be kept to a minimum by creating requirements in hierarchies and working systematically. Enterprise Architect has a number of features that can assist with this including the Relationship Matrix which will help to identify overlap. The powerful and flexible search function could also be used to identify overlapping or conflicting statements.</p>

	
<p>Modifiable</p>	<p>This means that a requirement can be changed without there being the need to modify other related requirements. It also applies to a Software (System) Requirements Specification and requires that it can be changed easily. Enterprise Architect can assist with both these issues, the Requirements themselves can easily be located through the search facility and the text and properties changed easily. The System Requirements Specification is automatically generated from the model so by simply changing one or more requirements and regenerating the document it will be updated.</p>
<p>Traceable</p>	<p>A Requirement is a specification of a characteristic or behavior and does not exist in isolation but is typically related to up-process entities such as stakeholders and business drivers and goals and down-process entities such as use cases and components. Enterprise Architect allows elements to be traced in any direction and provides a number of powerful tools to visualize the traces including the Relationship Matrix, the Traceability Window and the Requirements diagram itself. The Insert Related Elements facility can be used to automatically construct a diagram of traces.</p> 
<p>Unambiguous</p>	<p>A requirement should only be able to be interpreted in one way. Requirements that are ambiguous can lead a project to be delayed, over budget or have the wrong functionality or behavior. Enterprise Architect can assist with ambiguity by allowing analysts to make comments about the requirements using the Element Discussion facility.</p>
	<p>A requirement is verifiable if the implemented system or product can be</p>

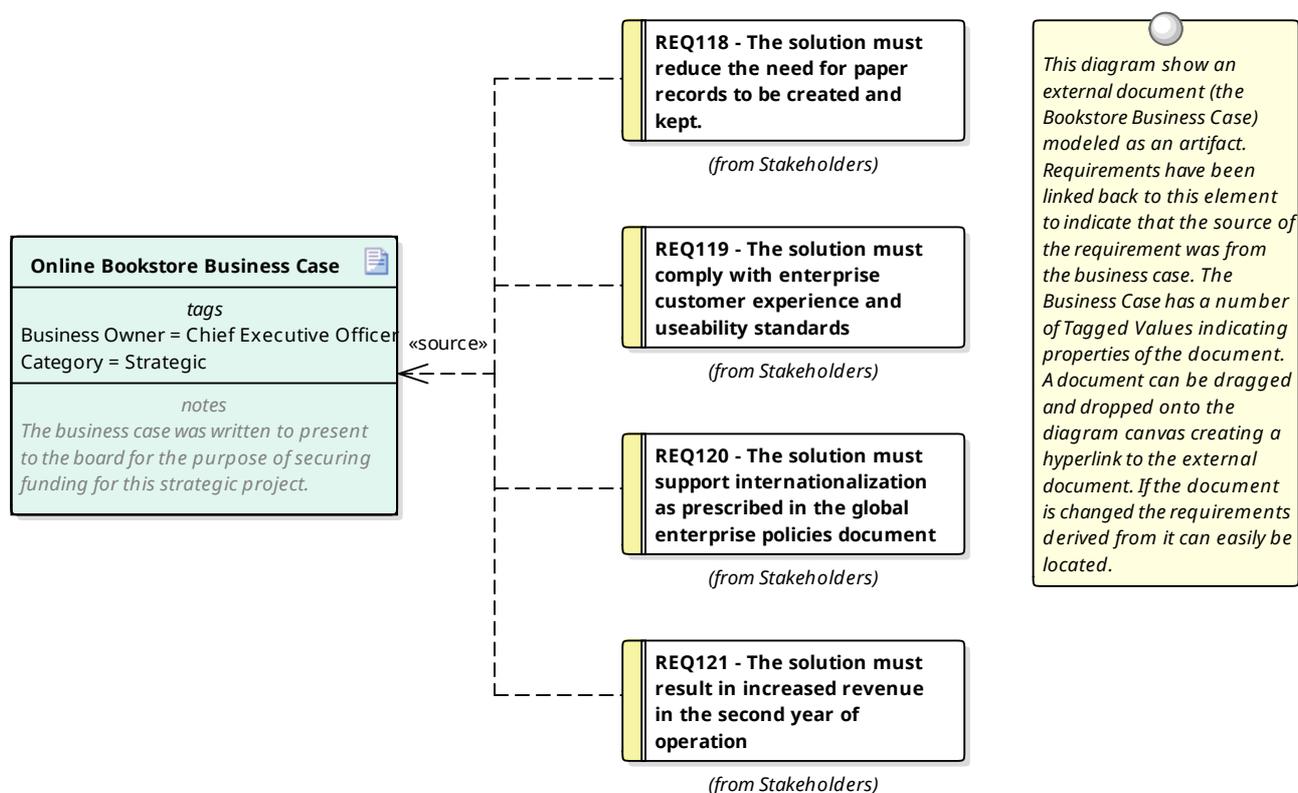
<p>Verifiable</p>	<p>tested to ascertain that the requirement has been met. Tantamount to being able to achieve this is knowing which test must be run to verify a particular requirement. Enterprise Architect can assist by allowing the modeler to trace test cases back to requirements and to visualize their relationship in a number of ways including the use of the Relationship Matrix. Test results can also be recorded directly inside Enterprise Architect.</p> <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <p style="text-align: center;">«User Story»</p> <p style="text-align: center;">As a Stock Control Manager I want to be able to list stock levels for a selection of titles.</p> <hr/> <p style="text-align: center;"><i>responsibilities</i></p> <p>A user must be warned when the report generation time is going to be more than one minute. Back ordered titles must only be included in total when the available stock is zero.</p> <hr/> <p style="text-align: center;"><i>constraints</i></p> <p>{The solution must use available Stored Procedures}</p> <hr/> <p style="text-align: center;"><i>tags</i></p> <p>Volatility = High</p> <hr/> <p style="text-align: center;"><i>test scripts</i></p> <p>System: : (Pass) Five non contiguous titles selected System: : (Fail) No titles selected System: : (Deferred) Two titles selected one with no stock</p> <hr/> <p style="text-align: center;"><i>maintenance</i></p> <p>Task: (New) Create application logic to display results in a table Task: (Verified) Create table index to ensure fast retrieval of stock levels Task: (New) Decide on Ux mechanism to multiselect a number of Titles Decision: (Verified) Display back order as negative stock</p> </div> <div style="text-align: right; margin-top: 10px;">  <p>Stakeholders:: Stock Control Manager</p> </div> <div style="border: 1px solid black; padding: 5px; margin-top: 10px; text-align: right;"> <p><i>This diagram shows an element with a variety of compartments displayed. This is useful when using diagrams to communicate other information stored in the repository. Enterprise Architect has a large set of extra information that can enrich the models and provide other disciplines such as project managers and testing analysts a place to store important project information.</i></p> </div>
<p>Necessary</p>	<p>Requirements should record a capability or behavior that is really needed or that specifies that the system or product should comply with constraints such as standards. Enterprise Architect can assist by allowing the modeler to relate each requirement back to its source and using the Relationship Matrix; requirements that have no source will be obviously identified as unnecessary or needing further investigation.</p>
<p>Feasible</p>	<p>A requirement that cannot be implemented will mean that the need of the stakeholder will not be met. It is best to identify these requirements as quickly as possible so as not to disappoint the owner of the requirement. Enterprise architect can assist by allowing analysts, architects, designers and developers to discuss the requirement and determine its feasibility using the Element Discussion window.</p>

Business Context for Requirements

Requirements don't appear in isolation but are usually defined or discovered in the context of a business problem or opportunity that has been defined in one or more business documents. These documents and the information they contain can be included in the models and provide an important anchor point for Requirements.

Business Case

The Business Case is a high level document or argument that attempts to articulate the reasons for initiating a project. It is an important artifact for the requirements analyst because it will typically contain information describing business value, drivers and business and technical risks. It places the endeavor in the context of other functions in the business and describes the solution options at a high level. It is an important source for requirements and should be included as an artifact in the model.



Drivers and Goals

Business Drivers and Goals are often documented by high level strategic thinkers such as business or enterprise architects. Drivers define resources, processes or constraints that are vital to the operation of the organization and Goals describe the position that the organization is wanting to attain. They are typically enterprise level concerns and so should be modeled above the level of individual projects. They often exist in high level documentation and even when they aren't clearly articulated at the organization level an analyst can mine them from previous project documentation such as a Vision document and model them in an enterprise package above the project packages in the repository.

Vision and Concept of Operation

While the Business Case describes the business reason for initiating the project the Vision typically elaborates the opportunity or problem in more detail describing the business context, the market position, key stakeholders and requirements, solution choices and constraints. The vision is more often than not created prior to the team being assembled and can be a great source of requirements information. The required system functionality is often expressed using Features.

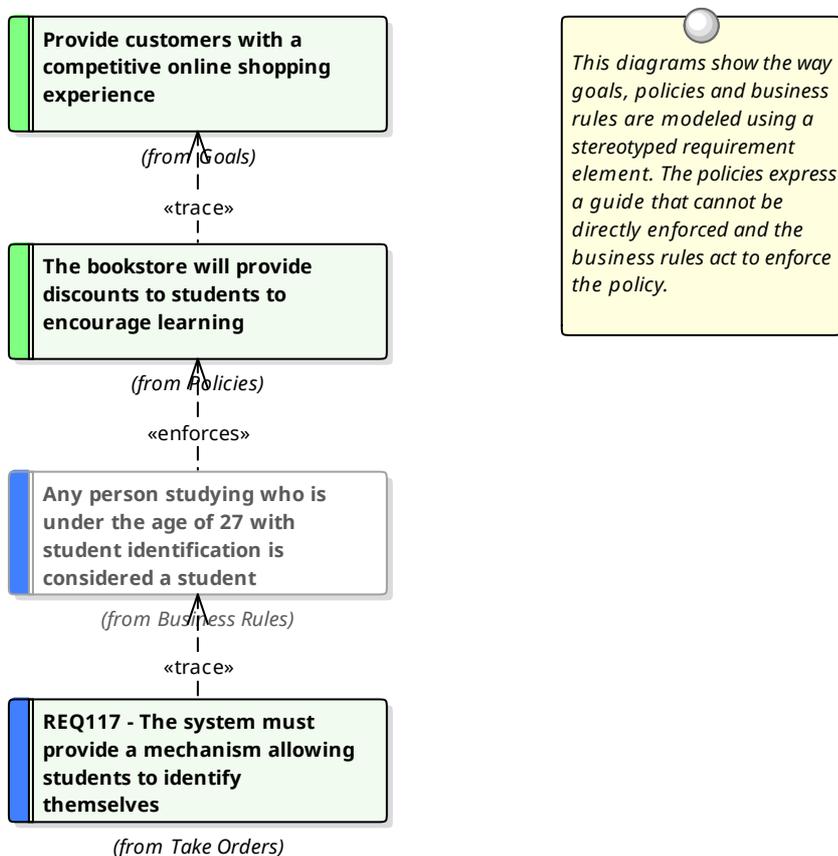


Enterprise Architect has a wide range of tools and element types that can be used to model the contents of the Vision Document including Users, Stakeholders, architecturally significant Use Cases and Requirements, Constraints and Deployment Environments.

Policies and Business Rules

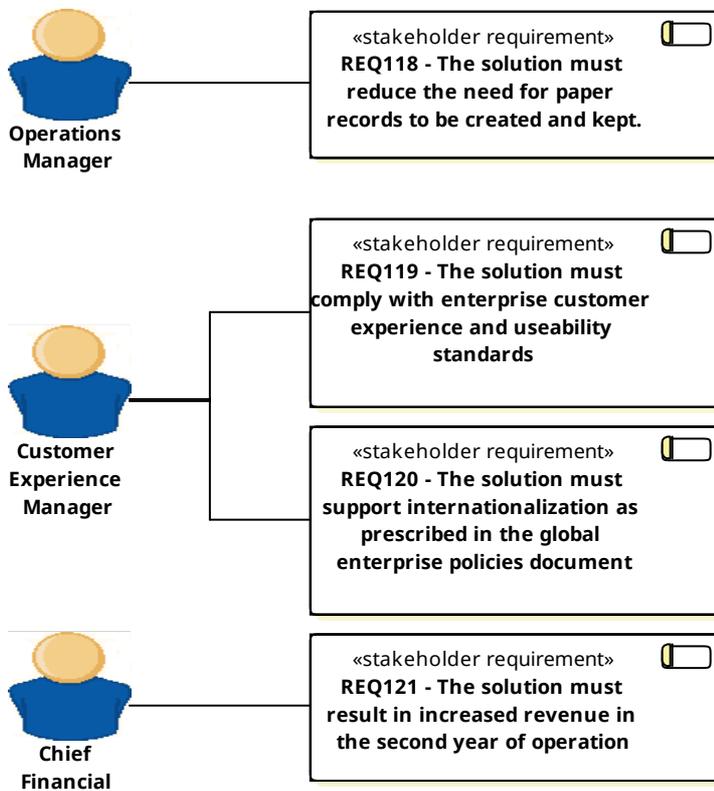
A policy is a high level principle or statement of intent typically defined and managed by a governance body and a Business Rule is an implementation of the Policy. They are not strictly requirements and are often defined at the enterprise level rather than the project level which facilitates their reuse across multiple projects. Policies and Business Rules can be modeled using stereotyped requirement elements and business and system requirements can be traced to them from individual projects. There is some overlap with regulatory and safety requirements which some methods consider to be types of Business Rules. Enterprise Architect supports the modeling of Policies and Business Rules by using stereotyped requirements but also has a powerful Business Rule Modeling capability that can create executable code for a variety of languages.

- The Business Rule Modeling is available in the Business and Software Engineering edition and the Ultimate edition of Enterprise Architect



Stakeholders and Their Concerns

Stakeholders typically have the same set of concerns regardless of whether projects are running or not. A Security Manager will for example be concerned about the vulnerability of sensitive organizational data, a Customer Experience Manager will be concerned about speed of access and a Chief Financial Officer will be interested in return on investment. These concerns can be modeled at the enterprise level as they are generic and independent of individual projects. They will provide a source of understanding for project level requirements and will help identify gaps in the requirements landscape. Enterprise Architect can be used to model Stakeholders using a stereotyped **UML Class** and these high level concerns can be modeled using a requirement stereotyped as a Stakeholder Concern.

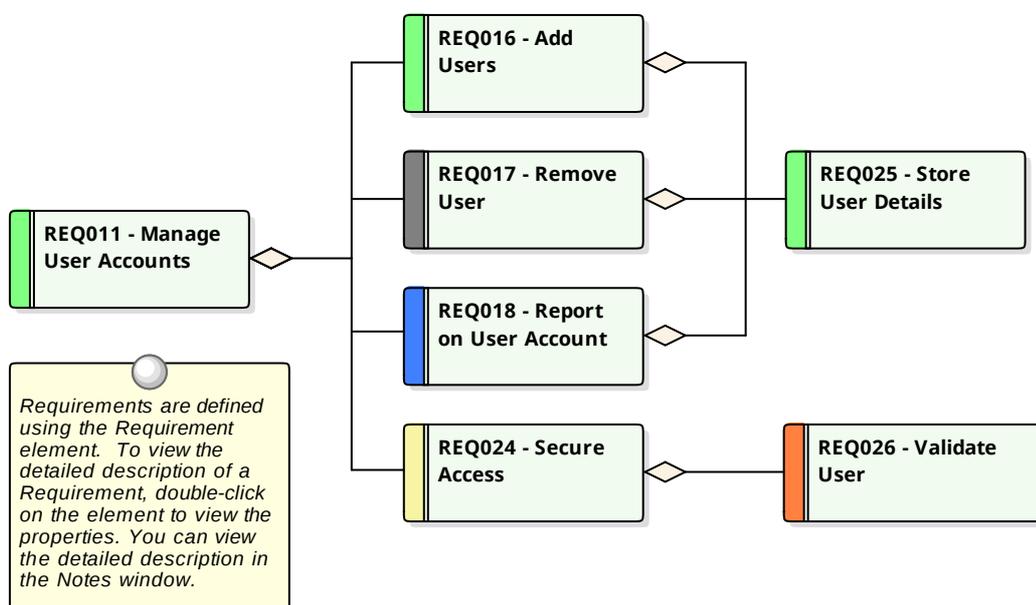


This diagram shows a number of stakeholders and their needs (requirements). A stereotype has been created for the stakeholders with an image assigned. The requirements are displayed in a rectangular presentation style so as to display the <<stakeholder requirement>> stereotype in the diagram.

Requirements Diagram

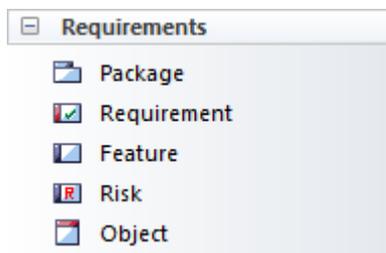
A requirements diagram is one of Enterprise Architect's extended diagram types and allows an analyst to model requirements visually, showing how requirements relate to each other and how they connect with other elements in the model including Business Drivers, Constraints, Business Rules, Use Cases and design Components. The requirement will be the main element seen on these diagrams and it has a name and a description and a series of properties (called attributes in some literature) such as status, complexity, difficulty and author. Enterprise Architect is designed to be a flexible tool and allows requirements to be created directly in the repository without the use of a diagram, but the diagram has proven to be a powerful tool to express the important role requirements play in the development process.

Example Diagram



Requirements Toolbox

You can create elements by dragging them from the 'Requirements' pages of the Toolbox onto the diagram canvas. Connectors can also be selected from the Toolbox and dragged between elements in the diagram or by using the Quick Linker. This table lists the elements available from the 'Requirements' toolbox but it is important to remember that other elements such as Use Cases and Components can be added to the diagram by opening other Toolbox pages using the 'More Tools...' option at the top of the Toolbox.

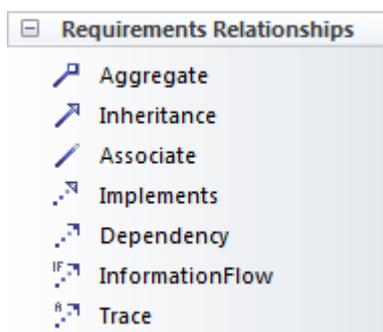


Requirements Toolbox Elements

Element	Usage
Package	Packages are container elements that can be used to group requirements and other elements into sets. They are not requirements themselves but act as a grouping mechanism; analysts should take care that the Package is not a high level requirement.
Requirement	The Requirement element is used for modeling a condition or capability that a system must have. The type of the requirement can be set but there are also a number of types of requirements such as Functional, User and Architectural requirements available from the Extended Requirements page of the toolbox. They are useful for modeling a range of other entities such as Rationales and Assumptions.
Feature	A Feature is a characteristic or property that the system must have to meet its business requirements. They are typically high level properties that represent a group of requirements.
Risk	A Risk is a condition that could cause the disruption, loss or compromise of a system. The element can be used to model both technical and business risks and can be connected to one or more elements.
Object	Objects are useful for modeling any of the entities that are discussed during a requirements elicitation workshop or while reading through project documentation. Formally they are Instances of Classes and when analysis is conducted a Domain Class can be derived from one or more objects.

Requirements Relationship Toolbox

You can create relationships by selecting the corresponding icon in the Toolbox and dragging between any two elements in the diagram canvas, or by using the Quick Linker. This table lists the relationships available from the 'Requirements Relationship' page of the Toolbox but it is important to remember that other relationships such as Composite can be added to the diagram by opening other Toolbox pages.



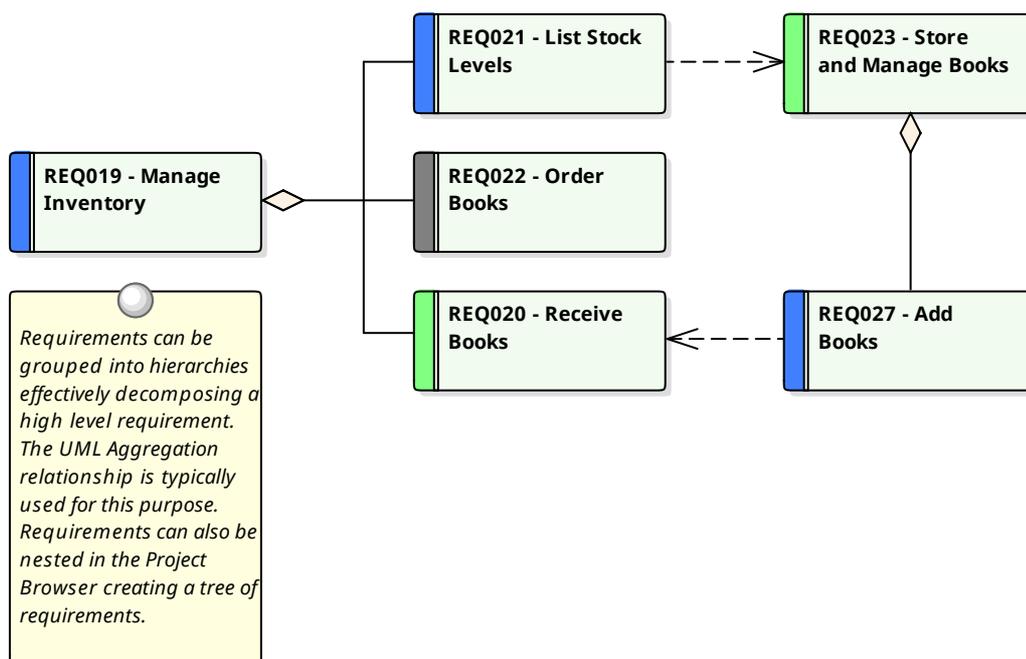
Requirements Relationships Toolbox

Connector	Use
Aggregate	Used to show that a requirement (diamond end) is made up of a another requirement (tail end). This allows hierarchies of requirements to be

	created.
Inheritance	Used to show that an element (triangle end) is a more generalized version of another element (tail end). The relationship is used between Classifiers such as Use Cases, Classes, Artifacts and Components.
Associate	Used to show a semantic or structural relationship between two elements.
Implements	Used to show that a model element implements a Requirement. Typically it would be used by an architect or designer to indicate that the need expressed in the Requirement would be met by a particular module, Use Case or Component in the system.
Dependency	Used to show that a Requirement (tail end) relies upon another element (arrow end).
Information Flow	Used to show that data flows between two elements in a Repository. The type of data can be represented as Information Items that can be selected from any part of the model. They could be used to show the Requirement that information flows between the proposed system and a supplier's system or to represent a Constraint that two Components must communicate via a certain protocol.
Trace	Used to show that an element (tail end) is more elaborated in the model than the element at the arrow end. So a User Requirement could be traced to a Stakeholder Requirement or to a Business Goal,

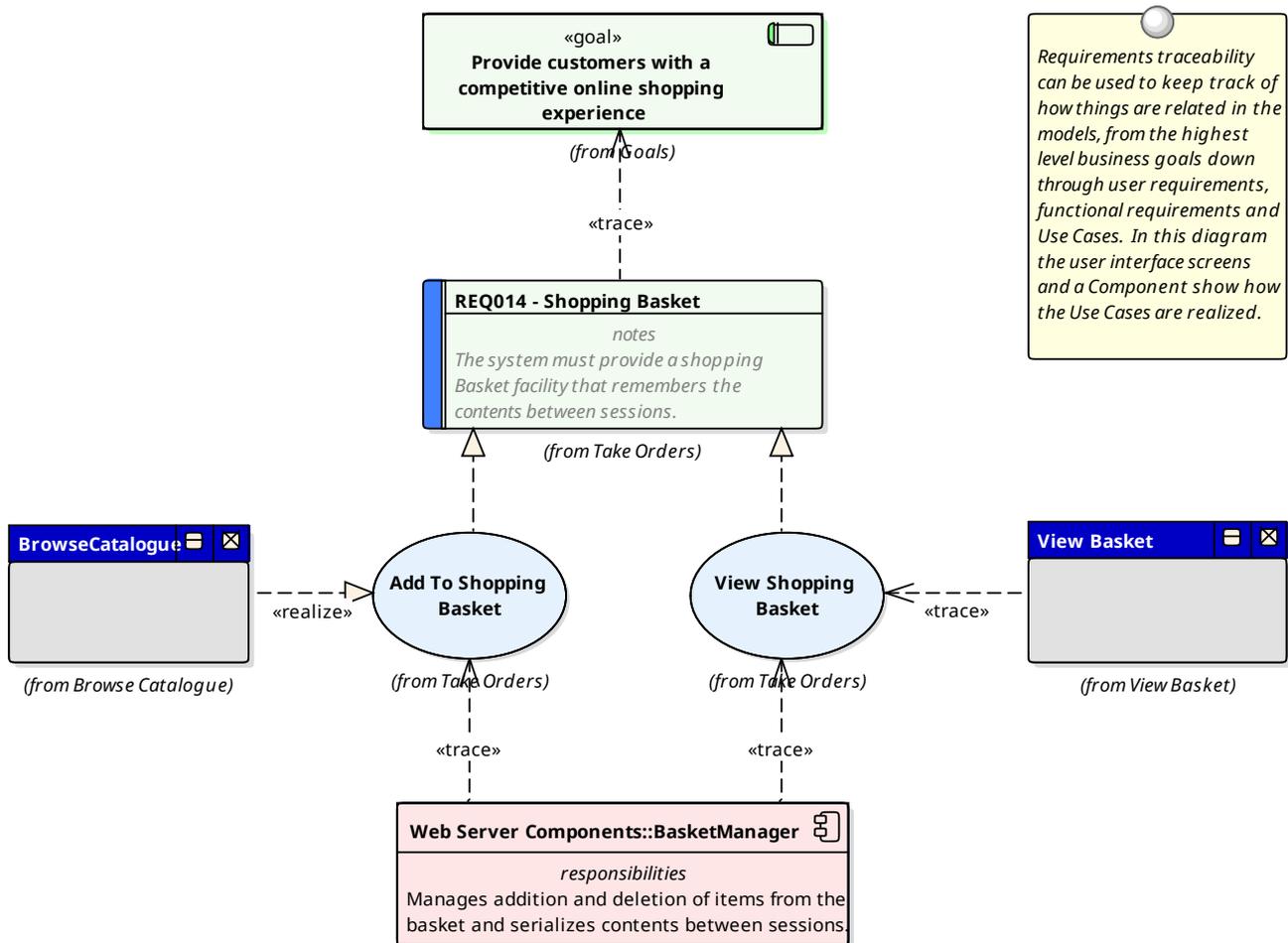
Example Diagram - Hierarchies

This diagram shows how requirements can be connected into hierarchies thus allowing high level requirements to be broken down to verifiable requirements.



Example Diagram - Traces

This diagram shows how Requirements can be connected to other elements in the model thus displaying traceability.

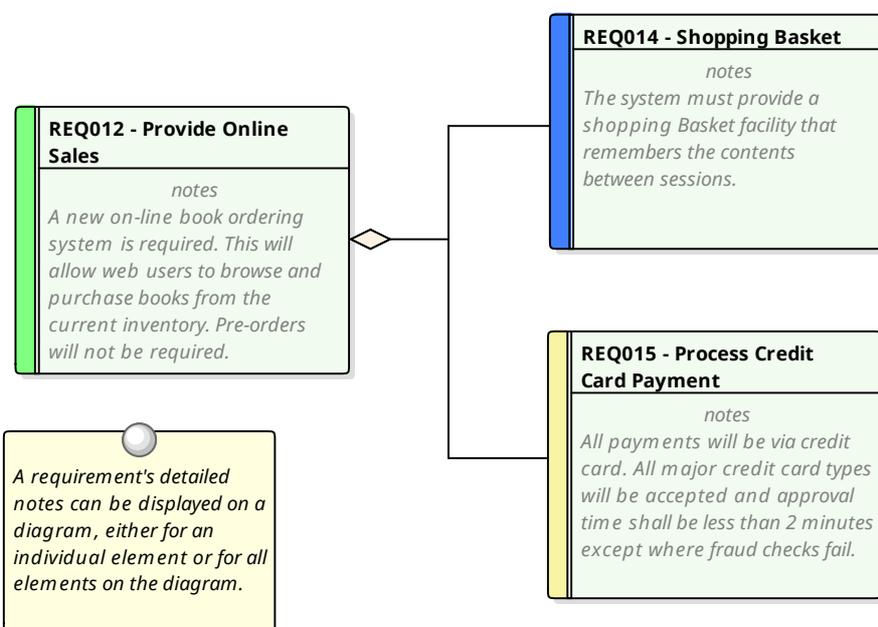


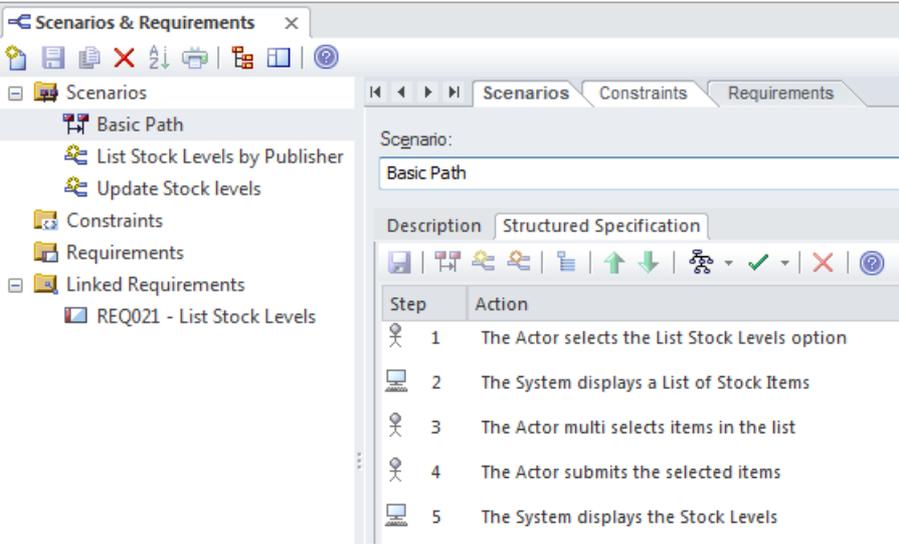
Creating and Viewing Requirements

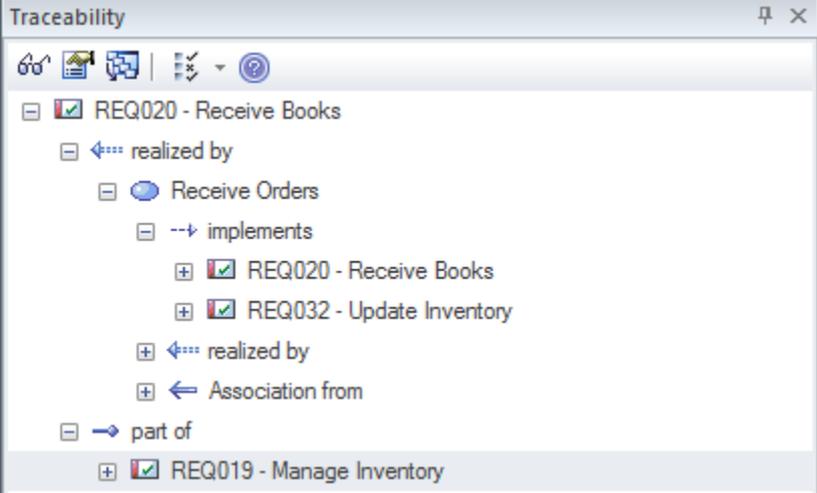
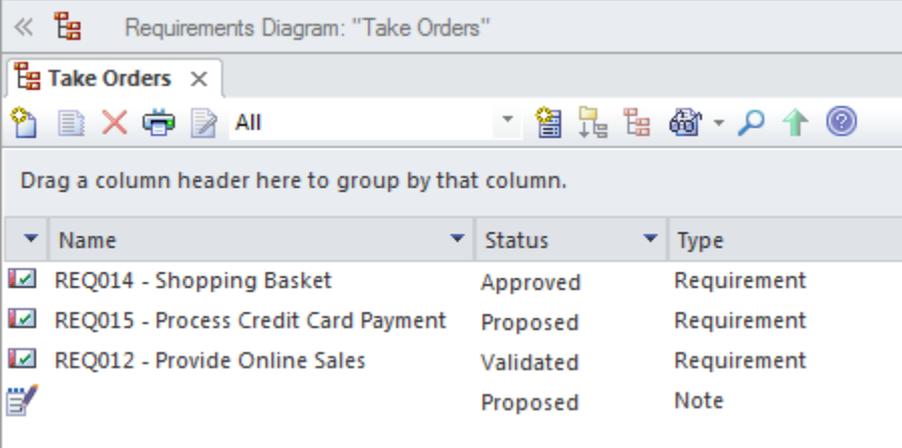
Enterprise Architect is a sophisticated and flexible modeling platform and the tool offers a rich set of features for both the development and the management of requirements for any size project in any domain and using a variety of processes. Requirements can be created in a variety of ways and then visualized in a series of windows and dialogs that make it easy to develop and manage them and to communicate within the team and to the business customers.

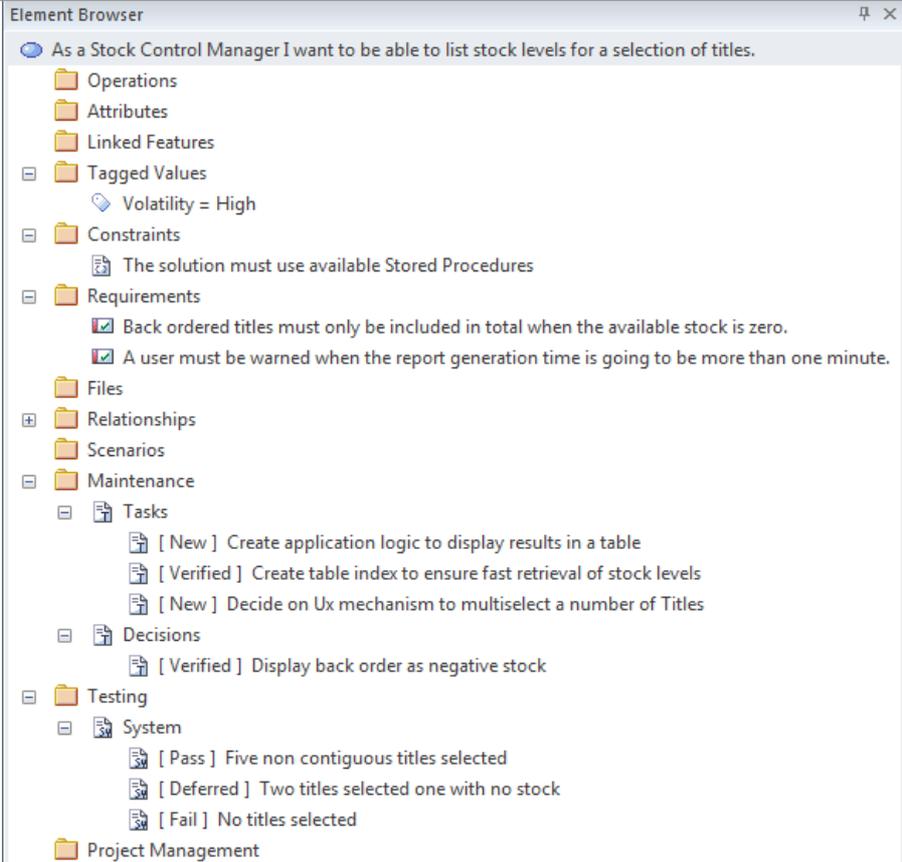
Viewing Requirements

Topic	Detail
Specification Manager	<p>Shows Requirements (and other element types) in a simple text format, and helps the modeler to create, edit and manage these elements. The Specification Manager will be the preferred tool for many requirement analysts as it allows the modeler to work in a familiar spread sheet like interface and edit requirement properties such as Status, Priority and Difficulty using drop-down lists and to add notes to the Requirements without needing to draw diagrams.</p> <div data-bbox="520 904 1422 958" style="background-color: #ffff00; padding: 2px;">Item</div> <p>1 REQ019 - Manage Inventory</p> <p>The system MUST include a complete inventory management facility to store and track stock of books for the on-line bookstore.</p> <p>1.1 REQ122 - Inventory Reports</p> <p>Inventory reports are required that detail the available stock for each item including back orders. Future stock level reports should be able to predict the quantity of stock at a specified future date.</p> <p>1.2 REQ023 - Store and Manage Books</p> <p>A book storage and management facility will be required.</p> <p>1.2.1 REQ022 - Order Books</p> <p>A book order facility will be required to allow on-line ordering from major stockist's.</p> <p>1.2.2 REQ021 - List Stock Levels</p> <p>A facility will exist to list current stock levels and to manually update stock quantities if physical checking reveals inconsistencies.</p>
Project Browser	Shows the content and structure of the repository and allows requirements to be viewed in a hierarchy of Packages, elements and diagrams.

	<ul style="list-style-type: none"> ⊕ Manage Users ⊖ Manage Inventory <ul style="list-style-type: none"> ⊖ 1: «Functional» REQ019 - Manage Inventory <ul style="list-style-type: none"> ⊖ 1.1: «Functional» REQ023 - Store and Manage Books <ul style="list-style-type: none"> ☑ 1.1.1: «Functional» REQ022 - Order Books ☑ 1.1.2: «Functional» REQ021 - List Stock Levels ⊖ 1.1.3: «Functional» REQ020 - Receive Books <ul style="list-style-type: none"> ☑ 1.1.3.1: «Functional» REQ027 - Add Books ☑ 1.1.4: REQ032 - Update Inventory ☑ 1.2: «FunctionalRequirement» REQ122 - Inventory Reports
<p>Requirements Diagram</p>	<p>Shows the arrangement of a group of Requirements and other elements, and can show whether the elements are in the same Package or different Packages. It is a powerful way of presenting requirements information because connectors can be created to show how the a Requirement relates to other elements in the Repository including other Requirements.</p>  <p>REQ012 - Provide Online Sales <i>notes</i> A new on-line book ordering system is required. This will allow web users to browse and purchase books from the current inventory. Pre-orders will not be required.</p> <p>REQ014 - Shopping Basket <i>notes</i> The system must provide a shopping Basket facility that remembers the contents between sessions.</p> <p>REQ015 - Process Credit Card Payment <i>notes</i> All payments will be via credit card. All major credit card types will be accepted and approval time shall be less than 2 minutes except where fraud checks fail.</p> <p><i>A requirement's detailed notes can be displayed on a diagram, either for an individual element or for all elements on the diagram.</i></p>
<p>Relationship Matrix</p>	<p>Shows one group of elements on the horizontal axis and another group of elements on the vertical axis with markers indicating if they have a relationship and an arrow showing the direction of the connector. Relationships can be created directly in the matrix and these will be displayed on diagrams containing the source and target elements. The Relationship Matrix is a powerful tool to quickly visualize gaps or missing requirements and is particularly useful for project managers and business stakeholders who may be less familiar with diagrammatic representations of Requirements and formal languages such as the UML.</p>

													
<p>Scenario Builder</p>	<p>Used to create Scenarios, Constraints such as pre-conditions and post-conditions and to detail the steps of Use Case Scenarios. It can also be used to create behavior diagrams that allow the steps in a Scenario to be visualized and used as a target or source for connectors.</p>  <table border="1" data-bbox="901 1355 1422 1608"> <thead> <tr> <th>Step</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>The Actor selects the List Stock Levels option</td> </tr> <tr> <td>2</td> <td>The System displays a List of Stock Items</td> </tr> <tr> <td>3</td> <td>The Actor multi selects items in the list</td> </tr> <tr> <td>4</td> <td>The Actor submits the selected items</td> </tr> <tr> <td>5</td> <td>The System displays the Stock Levels</td> </tr> </tbody> </table>	Step	Action	1	The Actor selects the List Stock Levels option	2	The System displays a List of Stock Items	3	The Actor multi selects items in the list	4	The Actor submits the selected items	5	The System displays the Stock Levels
Step	Action												
1	The Actor selects the List Stock Levels option												
2	The System displays a List of Stock Items												
3	The Actor multi selects items in the list												
4	The Actor submits the selected items												
5	The System displays the Stock Levels												
<p>Traceability Window</p>	<p>Used to display the hierarchy of elements based on their relationship to other elements. While the Project Browser will give a structural view based on containment in a name space the Traceability window displays related elements regardless of their location in the repository.</p>												

																
Diagram List	<p>Lists the elements in a diagram, filtered and sorted according to the settings you define; shows all or selected default properties of each element. The properties can be edited in-line for each Requirement and new Requirements can be created in the diagram list.</p>  <table border="1" data-bbox="520 1077 1422 1312"> <thead> <tr> <th>Name</th> <th>Status</th> <th>Type</th> </tr> </thead> <tbody> <tr> <td>REQ014 - Shopping Basket</td> <td>Approved</td> <td>Requirement</td> </tr> <tr> <td>REQ015 - Process Credit Card Payment</td> <td>Proposed</td> <td>Requirement</td> </tr> <tr> <td>REQ012 - Provide Online Sales</td> <td>Validated</td> <td>Requirement</td> </tr> <tr> <td></td> <td>Proposed</td> <td>Note</td> </tr> </tbody> </table>	Name	Status	Type	REQ014 - Shopping Basket	Approved	Requirement	REQ015 - Process Credit Card Payment	Proposed	Requirement	REQ012 - Provide Online Sales	Validated	Requirement		Proposed	Note
Name	Status	Type														
REQ014 - Shopping Basket	Approved	Requirement														
REQ015 - Process Credit Card Payment	Proposed	Requirement														
REQ012 - Provide Online Sales	Validated	Requirement														
	Proposed	Note														
Package Browser	<p>Lists the elements in a Package, filtered and sorted according to the settings you define; shows all or selected default properties of each element.</p>															
Element Browser	<p>Displays a selected element's Attributes, Tagged Values, Constraints Internal Requirements, Relationships, Maintenance Items, Testing, Project Management items, Files, and more. It is a powerful way of displaying this information in one place without the need to open up other windows.</p>															

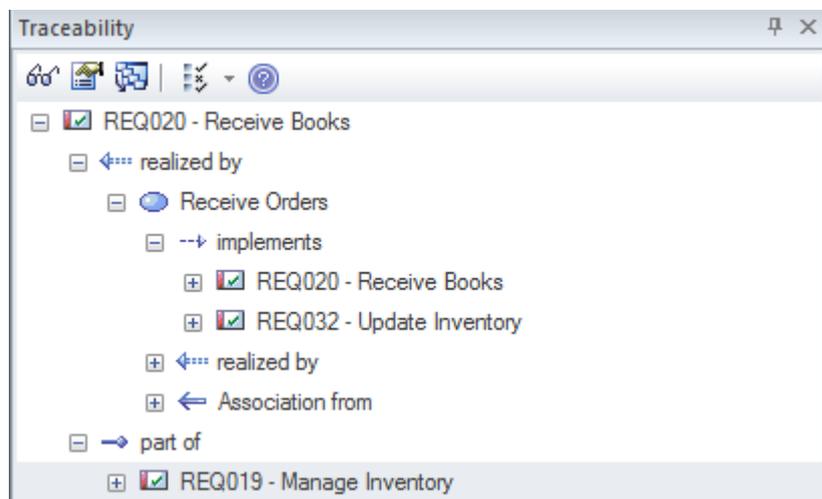
	
Model Search	Enables you to locate Requirements in general in the model, or specific Requirement elements, according to the search criteria you use.
Model Views	Enables you to maintain links to commonly-used elements, and to rapidly show developments and changes in (Requirement) Package contents through either reports or slide shows of selected diagrams.

Requirements Development

This comprises all the activities and tasks associated with discovering, evaluating, recording, documenting and validating the requirements for a particular project or program of work. Requirements are discovered, analyzed, specified and verified and Enterprise Architect has a wide range of tools and features to assist the analyst as they develop requirements. The center piece for requirements development is the **Specification Manager** allowing the requirements analyst to enter, view and manage requirements in textual form in a spread sheet like view.

Item	Priority	Status	Difficulty
1 REQ019 - Manage Inventory	Medium	Approved	Medium
The system MUST include a complete inventory management facility to store and track stock of books for the on-line bookstore.			
1.1 REQ122 - Inventory Reports	Medium	Proposed	Medium
Inventory reports are required that detail the available stock for each item including back orders. Future stock level reports should be able to predict the quantity of stock at a specified future date.			
1.2 REQ023 - Store and Manage Books	Low	Mandatory	
A book storage and management facility will be required.			
1.2.1 REQ022 - Order Books	Medium	Implemented	Medium
A book order facility will be required to allow on-line ordering from major stockist's.			
1.2.2 REQ021 - List Stock Levels	Medium	Approved	Medium
A facility will exist to list current stock levels and to manually update stock quantities if physical checking reveals inconsistencies.			

The Specification Manager can be used in conjunction with a platform of other tools such as diagrams, the **Traceability window** and the **Element Discussions** facility. These windows provide other views of the requirements giving the modeler and the viewer a deep understanding of how a requirement relates to other parts of the repository and providing detail not visible through the Specification Manager.



Elicitation

Elicitation is the process of information discovery and this information will form the precursors to requirements. The information will typically be raw and often heterogeneous and it will not be until the requirements analysis phase is performed that true requirements will be able to be derived from this information. Elicitation will take many forms and all of the skills of the analyst will be needed to determine which documents, tools, people and processes to examine to discover the information. Some of these locations are listed in this table.

Location of requirements information

Requirements are not handed to the analyst on a plate but rather will be discovered in a variety of places including: by observing stakeholders performing their work, in business documents and a variety of other locations.

Source	Description
Observing Users	Observing users perform their work is a powerful way of gaining information about Requirements and often reveals details that would not be discovered by user interviews.
Business Documents	A number of business documents such as the Business Case, Vision or Concept of Operation will provide a source for Business Requirements and should be discovered and included as sources of Requirements.
Stakeholder Workshops	Getting all the important stakeholders into a workshop is a powerful and productive way to get information that will help with the Requirements definition. Typically there are fertile and robust discussions that provide the basis for deriving Requirements.
Current System Issues	There will often be documented information recording errors, faults and issues with the incumbent system that will provide the basis for Requirements for the replacement system. Care needs to be taken that any Requirements derived from these list are owned by a stakeholder and that there is a business need to include them in the new system.

User Observations

Observing users perform their work is a powerful and unobtrusive way of gaining an understanding of the tasks they carry out and how they use information and other software and hardware devices to achieve an outcome. Even if the processes that support the planned system will be different the observations of the current processes will provide a useful context for discussions. It will also help the analyst empathize with the user and may result in a deeper understanding of the issues they face and provide the basis for the discovery of potential solutions. An analyst will often discover unmentioned documents, checklists and clue cards that can help illuminate the process. Equipped with a mobile phone or camera, it is also useful for the analyst to take photographs of the user working which will help in the requirements analysis phase.

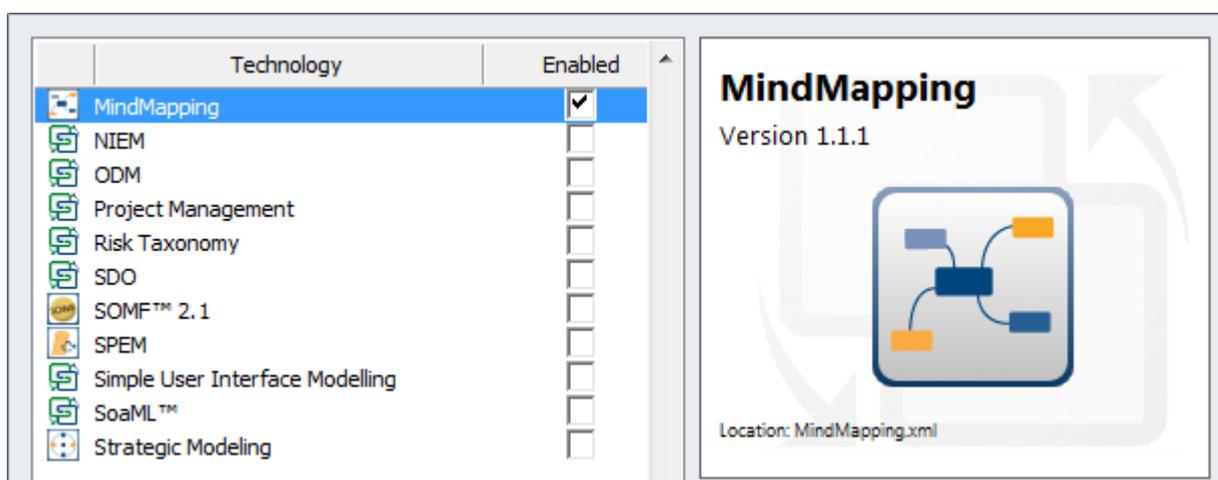
Enterprise Architect supports the modeler in representing files such as photos and scanned documents directly in the model, creating a rich and expressive representation of the user at work. There is the option to represent these as an Artifact (which, with a single key stroke (**F12**), will launch the file) or to use a hyperlink or even to include the image itself in a diagram.

Stakeholder Workshops

The requirements analyst or business analysis is charged with the difficult task of eliciting requirements, this necessitates excellent communication with the stakeholders including the customer and the analysis team. One very successful way of facilitating the elicitation of the stakeholders needs is to run a workshop with all the key stakeholders present. The analyst's skills as a communicator, diplomat and mediator are important to create a collaborative and respectful environment conducive to the exploration of the stakeholders needs and concerns. It is imperative that the analyst uses terminology that the stakeholders understand and displays an understanding or a willingness to learn the elements that make up the domain.

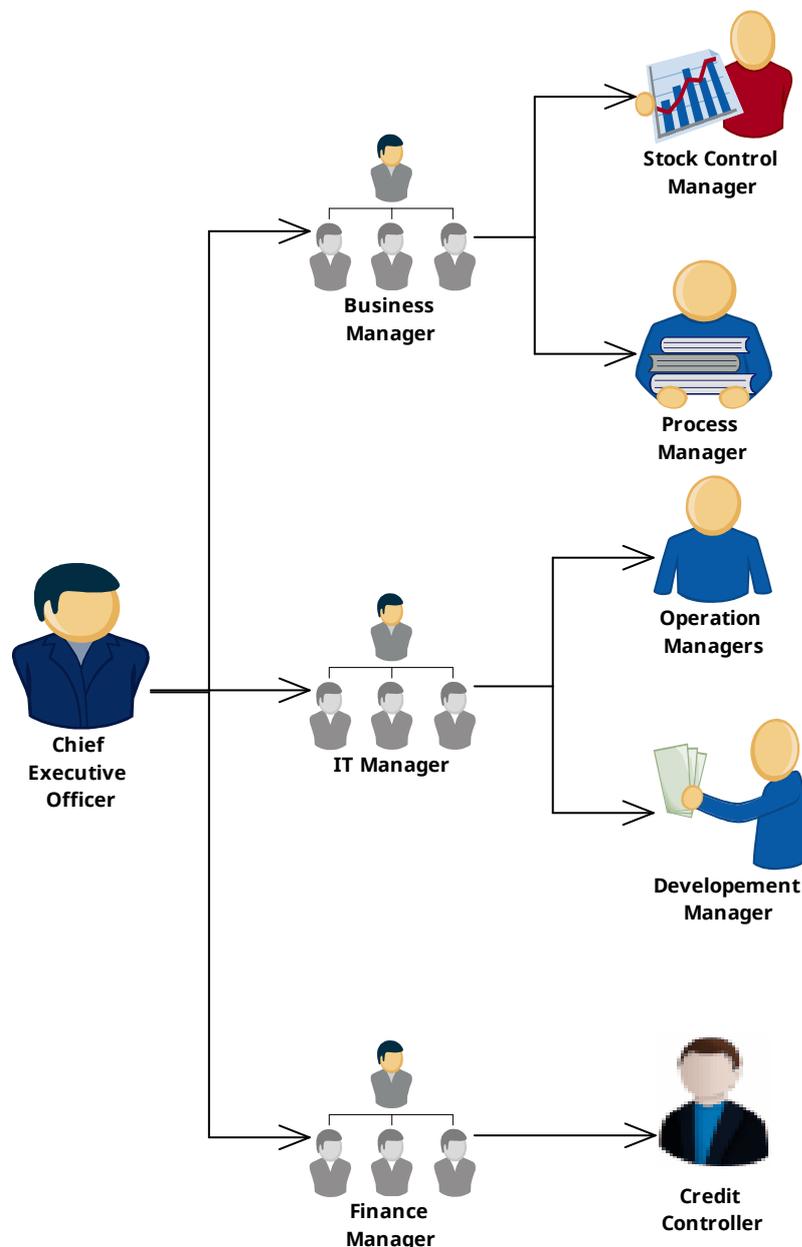
There is sometimes a misconception that what will be articulated is a set of clearly defined requirements that can be entered into the tool as Stakeholder Requirements; this is far from the reality of what happens. Stakeholders will typically articulate a wide range of ideas including Policies, Business Rules, Data definitions, Project Management Constraints, Functional Requirements, Business Requirements, existing system problems and even suggested solutions. Even when an external consultant is used to run these meetings the analyst will not have time to categorize all of these statements in the meetings. What is needed is a way for the scribe who is tasked with documenting the statements to get them into the tool without any concern for what type of information is being recorded. Having them recorded in the tool rather than scribbled in the analyst's notebook is best practice because it allows them to be displayed during the meeting and for stakeholders to see each others comments.

Enterprise Architect has a number of facilities that can help with these workshop. One method that is very effective is to use the MindMapping diagram to record the stakeholders statements which is very effective because it is a well known method and doesn't introduce any of the formality that comes with modeling languages such as UML.



As important terms are uncovered they could be entered into the **Project Glossary** and even if there is not time to discuss and debate the agreed meaning the words will act as a initial list of important entities in the domain. Alternatively the terms could be created in a Domain Model and related to each other with connectors that describe the important relationships between the terms.

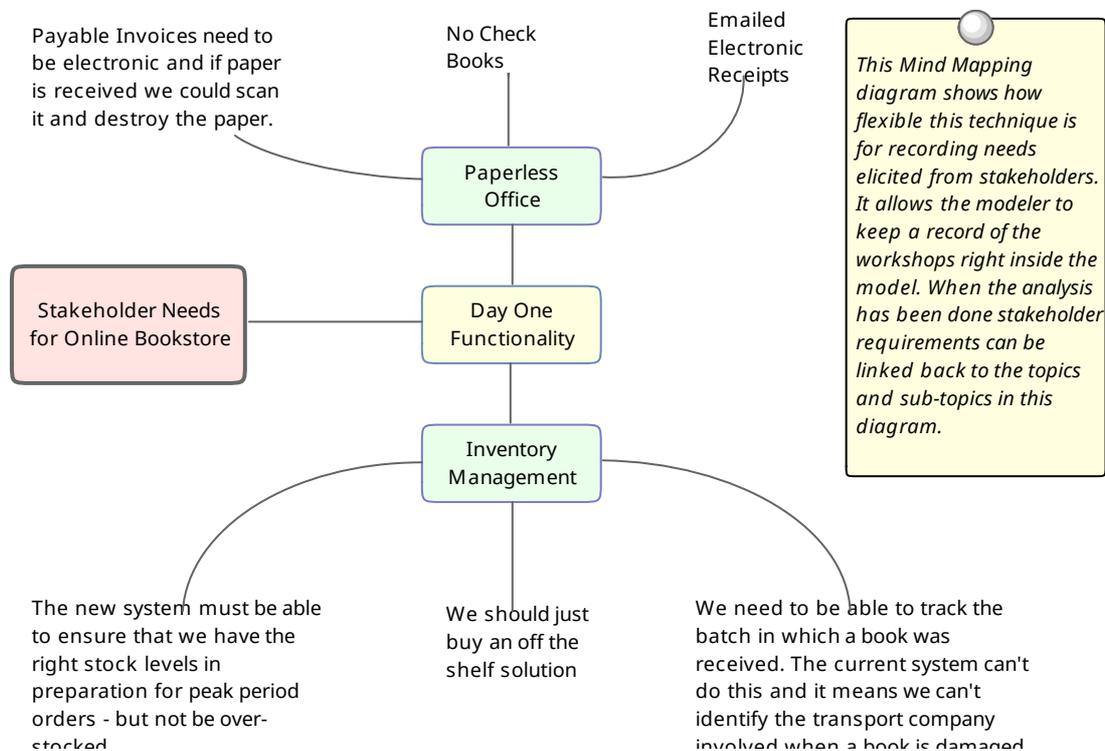
The stakeholders can also be modeled and their organizational relationship to each other can be described in a diagram. This is a useful technique that allows key stakeholders to locate themselves in the models which creates buy-in.



This diagram shows the organizational structure of stakeholders using an alternate image. Using graphics on diagrams tends to make the diagrams more appealing, particularly to high-level business stakeholders. An Alternate Image can be selected by choosing this option from the Appearance sub menu available from the element Context Menu.

Mind Mapping Diagram

A Mind Mapping diagram can be used to record the stakeholder's statements during an elicitation workshop. The statements are not categorized but simply recorded and later during the analysis phase of Requirement's development they can be converted to the appropriate elements or retained and the Requirements can be traced back to the topics effectively creating a record of how the Requirement was derived. This is a powerful technique that shields the stakeholders from needing to know the modeling languages and allows them to concentrate on articulating their needs, it also frees the analyst up from concerns about which element to use to model the statements. This step is usually performed in the analysis phase of the Requirement's Development process.



Glossary

Prior to a workshop an analyst can populate the **Project Glossary** with the existing terms and their meanings that have been gleaned from reading project documentation such as a Business Case or Vision Document. During the workshops, as new terms are uncovered they can be added to the Glossary and their definitions can be discussed and entered or deferred till later in the analysis phase.

Glossary Item Details

Term: Type:

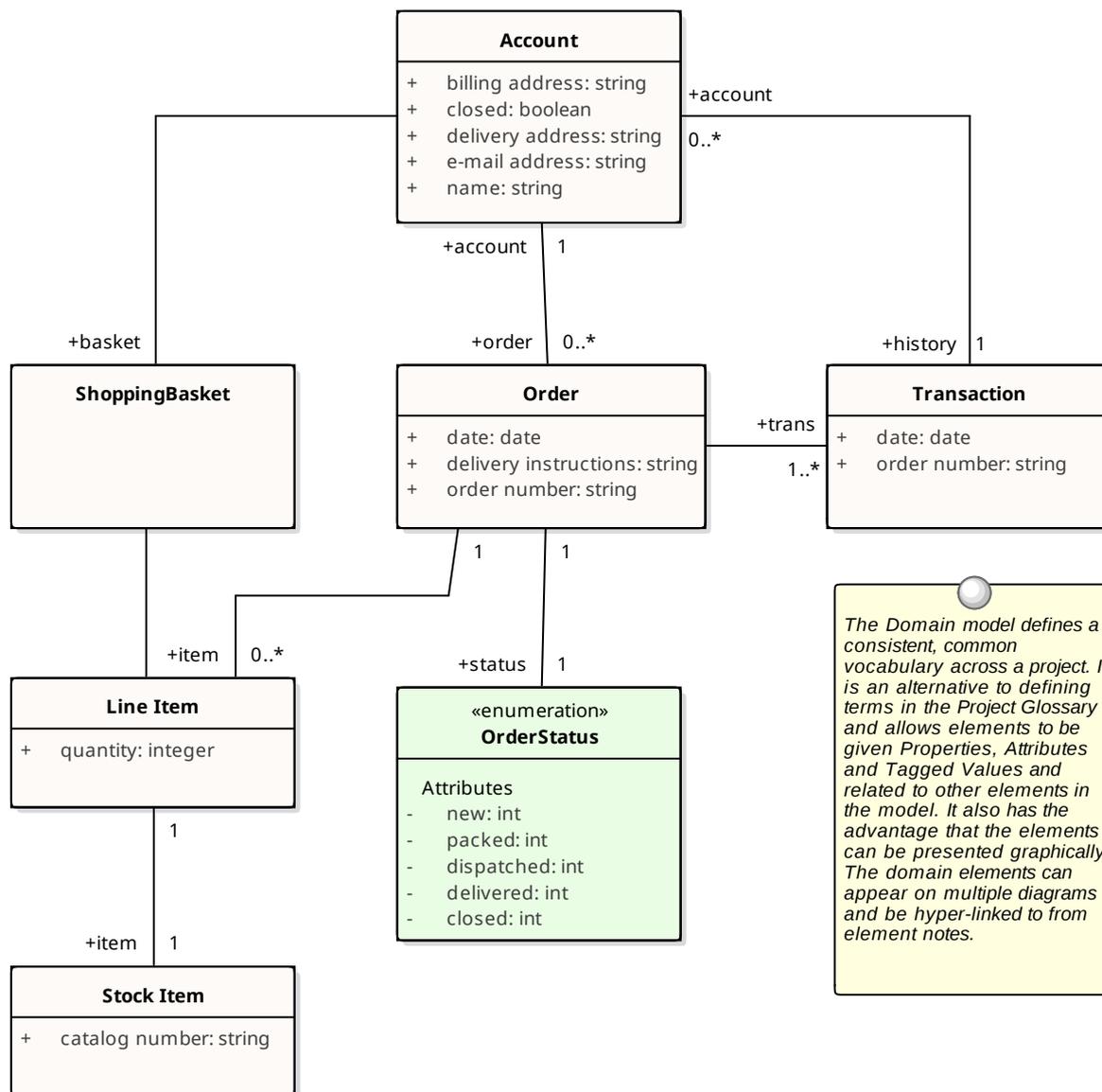
Meaning:

B *I* U **A** | | x^2 x_2

The Stock Item defines the items (books) that are stocked in the warehouse for on-line purchase.

Domain Model

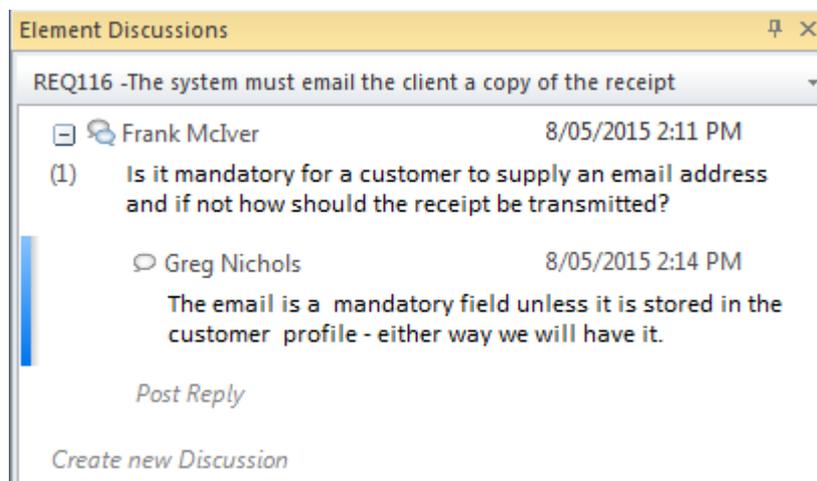
A domain model will act as a guiding model for discussions with many stakeholders and ideally a skeleton model should be created prior to the commencement of any workshops. The Domain Model should be kept simple and domain elements should be given a name and a description or a responsibility and initially only important connections should be made between elements. As the workshop progresses new elements will be uncovered and can be added directly to the model giving the stakeholders confidence that their needs and concerns are being addressed and managed well. Enterprise Architect allows domain models to be created using the **UML Class** diagram.



The Domain model defines a consistent, common vocabulary across a project. It is an alternative to defining terms in the Project Glossary and allows elements to be given Properties, Attributes and Tagged Values and related to other elements in the model. It also has the advantage that the elements can be presented graphically. The domain elements can appear on multiple diagrams and be hyper-linked to from element notes.

Element Discussions

The **Element Discussion window** is a convenient facility that allows commentary to be made on elements without contaminating the notes with discussions that ultimately don't contribute to the integrity of the model. Modelers often place notes on diagrams or write questions in the element notes fields and these are distracting and must be removed when formal documentation is generated from the model. The Element Discussion window allows a modeler to initiate a discussion and for others to reply. It is a perfect way for discussing requirements.



A **Discussions** summary window conveniently displays the Discussions for all elements in the repository.

Creating Requirements

Enterprise Architect has extensive support for developing Requirements and provides a number of specialized tools for this purpose. As with all model content a modeler is encouraged to check whether the Requirements have been entered into the repository by someone else before embarking on the task of creating new Requirements. It is also possible that the Requirements have been defined in another tool such as a spread sheet and could be imported into Enterprise Architect without the need to create each Requirement manually. Enterprise Architect has two locations for Requirements: they can be created in the model as an element that will appear in the **Project Browser** or they can be created inside another element as an Internal Requirement.

External and Internal Requirements

Enterprise Architect can support any type of requirement process and allows requirements to be defined as elements in the model - these are called External Requirements but the tool also allows requirements to be defined for a specific element which are called Internal Requirements. An analyst who wants to define a user requirement such as the 'The system must allow bus schedules to be updated' would use an external requirement. A modeler wanting to describe how a component should behave would use an internal requirement for the Component such as 'The editor must support Unicode'. There is often contention between analysts and developers as to whether a requirement should be internal or external and Enterprise Architect provides a facility to move internal requirements to be external to the element and analogously external requirements can be moved to be internal. When they are moved they are still retained in the original element but a copy of them is made as a requirements element. The internal and external requirements are synchronized and changing one will change the other.

Creating External Requirements

Requirement analysts typically come from varied backgrounds and often have predilections about how they want to work so Enterprise Architect provides a wide range of ways of creating and managing requirements. External Requirements are Requirements that will appear in the **Project Browser** and can be added to diagrams and viewed as separate elements with their own properties.

Methods for creating external requirements

Method	Description
Using the Specification Manager	Using the context menu and selecting 'Add New Element' will result in a new element being created in the grid ready for details to be added
Dragging a Toolbox item onto the current diagram	Dragging and dropping an item from a displayed toolbox page onto the current diagram will result in the element being added to the diagram.
Directly in the Project Browser	In the Project Browser , choose the location for the element to be inserted and select the 'New Element' toolbar option or 'Add Element' from the context menu.
Directly in a Package List	Display the context menu by right-clicking in the body of the window and select 'New Element'; the element will be added to the list.
Directly in a Diagram List	Display the context menu by right-clicking in the body of the widow and select 'New Element'; the element will be added to the list.
Importing from a variety of sources	Enterprise Architect supports a wide range of ways of importing requirements from external sources.
Moving an internal requirement external	Requirements that are defined inside an element can be moved external to a location specified by modeler thus creating a new element.

Notes

An Internal Requirement will not be displayed on a diagram by default, to ensure it is displayed you must set the compartment as visible either for the individual element or for all elements on the diagram.

Creating Internal Requirements

Internal Requirements can be created in one of two ways, by creating the Requirement from an element's property sheet or by moving an External Requirement inside an element. This section describes how to create a Requirement directly inside an element.

Creating internal requirements

Step	Action
1	Select the element you wish to house the Internal Requirement and open the Property window for the element.
2	Select the Requirement section, the Requirement panel will be displayed.
3	Enter the name of the Requirement in the 'Requirement' field and any details into the notes field. Other properties such as Status and Priority can be added.
4	Select Save to save the Requirement. The Requirement will be added to the element and will be displayed in the bottom section of the Window. Repeat the steps above to enter another Requirement.

Using the Specification Manager

Enterprise Architect has a powerful and fully featured tool called the **Specification Manager** for creating, visualizing and managing specifications which is particularly useful for working with requirements. Requirements analysts come from varied backgrounds and often have strong predilections about the way they work and the tools they like to use, so Enterprise Architect has functionality to suit a wide range of work styles. Some analysts like to work with elements and can use the diagramming interface, others will like to work with hierarchies so they can work in the **Project Browser** but for those who like to work with text and prefer an interface like tables and spread sheets the specification manager will be their tool of choice.

Item	Priority	Status	Difficulty
1 REQ019 - Manage Inventory The system MUST include a complete inventory management facility to store and track stock of books for the on-line bookstore.	Medium	Approved	Medium
1.1 REQ122 - Inventory Reports Inventory reports are required that detail the available stock for each item including back orders. Future stock level reports should be able to predict the quantity of stock at a specified future date.	Medium	<div style="border: 1px solid black; padding: 2px;"> Proposed ▼ Approved Implemented Mandatory Proposed Validated Mandatory </div>	Medium
1.2 REQ023 - Store and Manage Books A book storage and management facility will be required.	Low	Mandatory	
1.2.1 REQ022 - Order Books A book order facility will be required to allow on-line ordering from major stockist's.	Medium	Implemented	Medium
1.2.2 REQ021 - List Stock Levels A facility will exist to list current stock levels and to manually update stock quantities if physical checking reveals inconsistencies.	Medium	Approved	Medium

When you create delete or update elements in the Specification Manger they are automatically updated in the Project Browser and any diagrams the element appears in.

Requirement Properties

Requirements development and management is critical to the success of any project and the properties of the requirements are important to the prioritization, and the way they will be elaborated and used within an implementation or development team. All Enterprise Architect elements have standard properties such as Status, Author and Phase but the requirement element has additional properties such as Difficulty and Priority. Some requirements processes will specify specific properties such as Owner and Volatility (Stability) and these can be configured by using **Tagged Values** that can be applied to each requirement. The 'Notes' field for a requirement has special significance as it often contains a formal and contractual description of how the system must behave or perform.

Access

Right-click on Requirement element | Properties

Use to

- Document requirements
- Set requirement properties such as Type, Difficulty and Priority
- Set other element properties common to both Requirements and other model elements

Reference

Field	Action
Short Description	The name of the Requirement, which could include numbering, a natural language description or some other formal specification. If you have setup Auto Names and Counters for Requirements and they are active, and you already have some text in this field, it would be over-written by the auto-counter text.
Alias	An alternate name (alias) to be used for this requirement. If you have set Alias autonaming and autonumbering, and you have already have some text in this field, it is over-written by the auto-counter text.
Status	The current status of this requirement.
Difficulty	An estimate of the difficulty in meeting this requirement; select from: <ul style="list-style-type: none"> • Low • Medium • High
Priority	The relative importance of meeting this requirement compared to other requirements; select from: <ul style="list-style-type: none"> • Low • Medium

	<ul style="list-style-type: none"> • High
Author	The modeler who created this requirement.
Key Words	A set of user-defined words that could be used to index or define the subject of this requirement.
Type	<p>The type of this Requirement, typically used as a category for the Requirement. Possible values are defined on the 'Requirements' tab of the General Types window.</p> <p>This field displays a single value. You can click on the drop-down arrow and select a different value if necessary.</p> <p>However, be aware that you can define <i>multiple</i> values for this field, including stereotypes that you create or that are used in integrated or imported MDG technologies. You assign these multiple values using the Properties window for the selected Requirement element, in the 'Stereotype' field.</p> <p>This has two impacts on the 'Type' field:</p> <ul style="list-style-type: none"> • The value displayed in the field might have been set on the Properties window, and might not be shown in the drop-down list (if it is a stereotype and not a General Type) • If you select a different value from the 'Type' drop-down list, you change only the first of the multiple values (the one displayed in the field); you do not change any of the other multiple values, which remain set
Phase	The project phase of this requirement.
Version	The version of this requirement.
Last Update	Read-only field specifying when this Requirement was last changed.
Created	Read-only field specifying when this Requirement was first created.
Notes	The description of this requirement, typically providing a more detailed explanation of the requirement. Some requirement processes prescribe that only a statement of the requirement be provided, and the 'Notes' field in these cases would remain blank. Novice modelers sometimes make the mistake of using this field for analyst discussions about the requirement; this commentary is best entered in the purpose-built Element Discussions window .

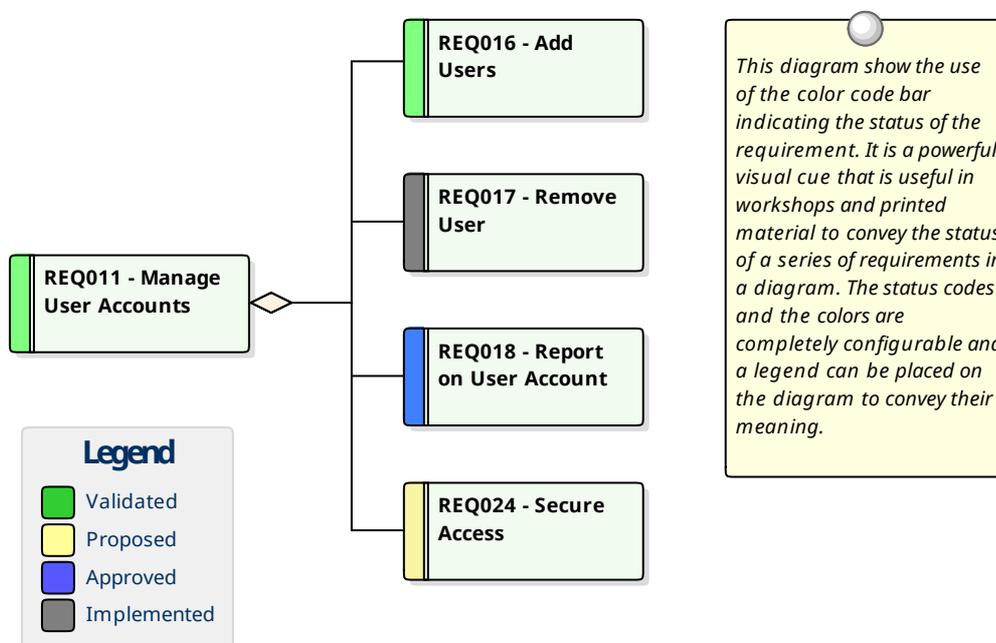
Notes

- In Requirement Management tools and texts, the characteristics of a requirement are commonly called Attributes; however, in UML the term Attribute refers to a different type of feature, and the Requirement characteristics are defined as properties - in this Enterprise Architect documentation, the term properties is used.
- In a project, it might be necessary to define more information in a Requirement than is provided by the standard properties and **Tagged Values** can be used for this purpose.

Color Coded Requirements Status

The status of a requirement is an important property for project managers and other team members. Enterprise Architect provides a way to display the status of each requirement as a color code in a diagram. The color codes act as a compelling visual cue, allowing team members and other stakeholders to get a quick view of the status of a set of requirements. The color codes are pre-configured in Enterprise Architect but can be customized to suit any team, including deleting or adding additional codes and changing the color of existing ones. The default color codes are:

- Yellow for Proposed
- Blue for Approved
- Green for Validated
- Orange for Mandatory
- Black for Implemented



A modeler can choose whether to display the color codes for diagrams in a repository, by changing this setting in the Options window.

Access

Tools | Options | Objects | Show status colors on diagrams

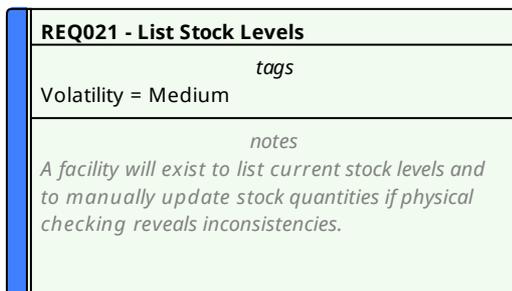
Enable color coded requirements

Step	Action
1	Select the 'Show status colors on diagrams' checkbox to enable the status of requirements to be represented by color coding in a diagram.

Displaying Properties on a Diagram

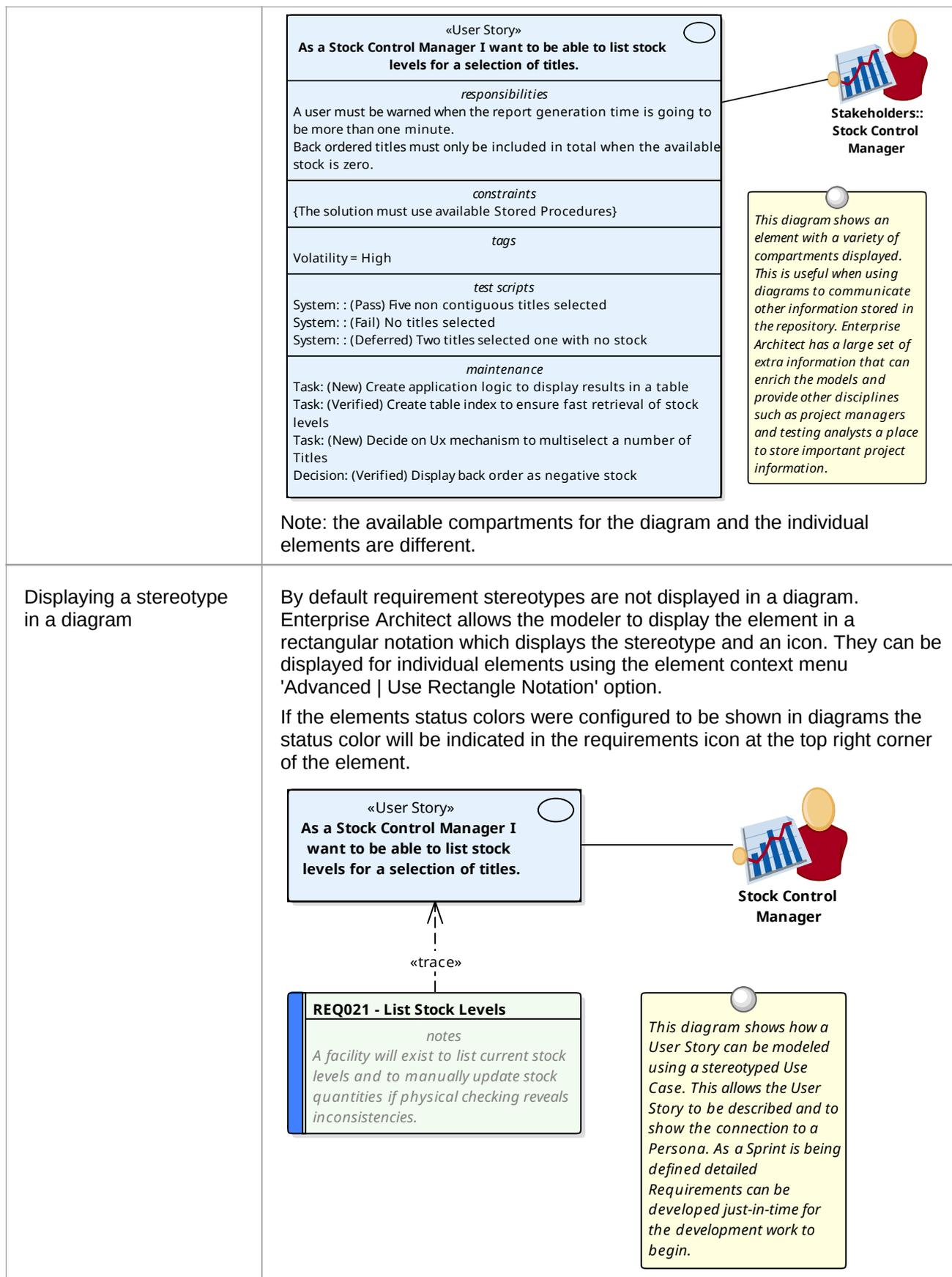
A diagram is a powerful means of communication. For many stakeholders it will be the preferred method of viewing the contents of a repository and each stakeholder will typically want to see different information. Enterprise Architect provides great flexibility, allowing the modeler to tailor what is shown in a diagram. This includes whether to display the detailed notes of a requirement, the extended properties represented by **Tagged Values**, constraints, testing details and more. This feature is not only available for requirements but can be used with any diagram object in a repository. The customization can occur at two levels:

- Customize what is displayed for individual elements in a diagram
- Customize what is displayed for all elements in a diagram



Customize what is displayed in a diagram

Action	Description
Customize visible compartments	<p>By default, requirements extended properties, notes and other element information are not displayed in a diagram but it can be useful to display this in the diagram elements particularly when conducting requirement workshops or creating documentation. Enterprise Architect supports element compartments and any number of compartments can be displayed in diagram elements, including Notes, Tagged Values (extended properties), Constraints, Test and Maintenance items such as Defects, Changes, Issues, Tasks, Decisions and Events.</p> <p>There are two options to do this:</p> <ul style="list-style-type: none"> • To display the additional compartment on all elements in a diagram, double-click on the diagram background and select the 'Elements' tab of the 'Diagram Properties' dialog; select the compartments checkbox for each compartment to display and click on the OK button • To display the additional compartment on a specific element in a diagram, from the element context menu select the 'Features & Properties Feature and Compartment Visibility' option; select the compartments checkbox for each compartment to display in the 'Show Element Compartments' panel of the 'Feature Visibility' dialog, or for notes in the 'Element Notes' panel and click on the OK button. <p>The additional compartments are then displayed in the Requirement element(s) on the diagram.</p>



Notes

- When the display is customized at the level of the diagram these settings will take precedence over the element level settings
- When the display is customized at the level of the requirement the length of the notes text displayed can be further tailored to a particular number of characters and formatting options can be set

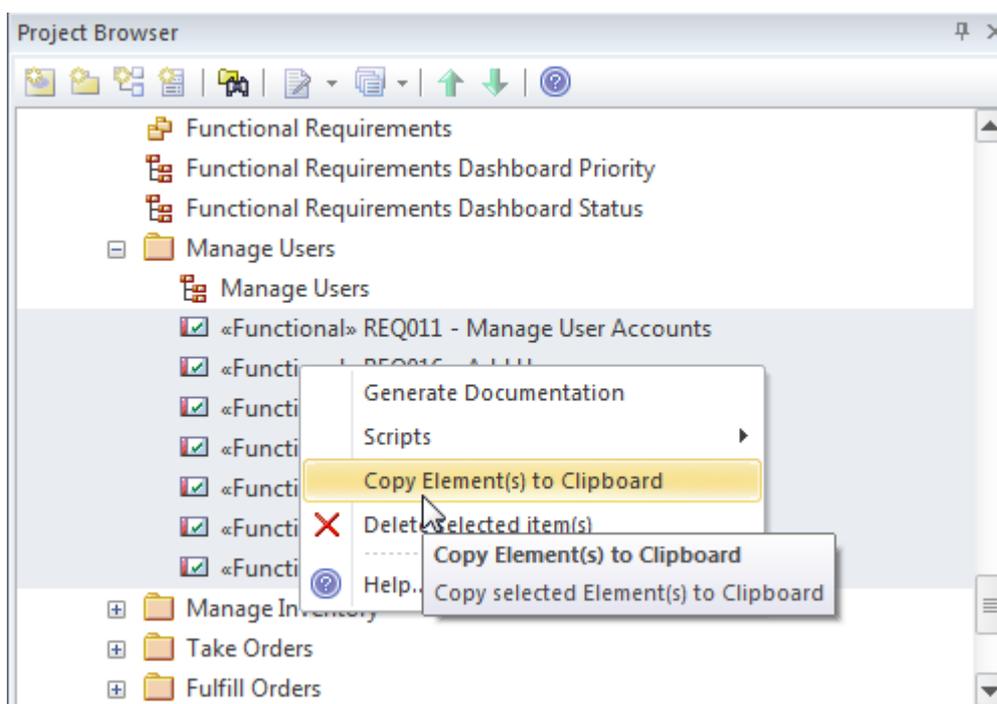
Import Requirements

A requirements analyst has a busy schedule of work from running stakeholder workshops to sifting through project documentation and updating project managers with the status of the requirement development so any opportunity to save time and work more efficiently is a usually welcomed. One of the most difficult tasks that the analyst faces is how to get a central repository of requirements when the source documents are often in a variety of formats and tools. Enterprise Architect has a range of mechanisms to import requirements from disparate sources. Requirements can be imported from these sources:

- Enterprise Architect models using copy and paste
- A text based document such as a word processor file
- A CSV file that can be exported from a Spread Sheet or similar tool
- An XMI file which has been exported from another model
- Rational Doors requirement management tool
- Any file source using a script to process the file
- A reusable asset server that has a register of requirements

Import Requirements by Copy and Paste Steps

This method of creating requirements allows you to copy requirements from an existing Enterprise Architect model by simply opening both models and copying the elements from one model to the other. You can copy a number of elements in the same Package or, if it is easier, you can copy an entire Package and all the elements it contains. When you copy elements they will be created as new elements in the target model and will be assigned new GUIDs.



Step	Action
1	Using the Project Browser select an individual requirement or multi-select a number of requirements in the same Package and right-click to display the context menu.
2	Select the 'Copy/Paste Copy Element(s) to Clipboard' option from the context menu.

	Enterprise Architect copies the elements to the clipboard.
3	Select the target location in the Project Browser and right-click to display the context menu.
4	Select 'Copy/Paste Paste Element(s) from Clipboard' from the context menu Enterprise Architect creates the new elements in the target location assigning new GUIDs to the elements.

Import Requirements from Text

Word Processors and other text tools are commonly used to develop requirements when a team is not equipped with a purpose built requirements modeling platform like Enterprise Architect. It is quite common to arrive at an elicitation workshop and find a customer or another stakeholder holding up a document saying that they have already started documenting their requirements. Fortunately Enterprise Architect has a convenient way of importing these requirements. This procedure creates a new element in Enterprise Architect by converting a text heading into an element's name and the text under the heading into the element's notes. You can use this method to generate any type of element; however, it is particularly useful for importing requirements from a requirements specification document. If there are a large number of requirements it may be more appropriate to use the **Scripting** method to import the requirements or if they are in a table to export them to a spreadsheet and use the CSV import mechanism.

Import Requirements from Text Steps

Step	Action
1	Ensure you have requirements diagram open.
2	Open the document file containing the text you want to generate Requirement elements from (this can be opened in any common text editing tool).
3	Highlight the required heading and associated text and drag them from the text file into the diagram. The Toolbox Shortcut menu displays.
4	Navigate through the menus and select the relevant element type, in this case Requirement. (If the diagram you are dragging onto is not a requirement diagram you will have to navigate to the requirement.)
5	Enterprise Architect creates a Requirement element in the diagram, and displays the 'Properties' dialog with the section heading in the 'Name' (or equivalent) field and the text in the 'Notes' field; the element is also added to the diagram's parent Package or element in the Project Browser .

Import Requirements from a CSV file

Spreadsheets are often the default tool that analysts will use for requirements development when they are not equipped with a more sophisticated tool such as Enterprise Architect. So it is quite common to find that a number of members of a team, including customers, have been entering requirements into spreadsheet

before Enterprise Architect has been installed. Fortunately Enterprise Architect has a built in facility to import these requirements so that they can be managed in a purpose built requirements modeling platform and the spread sheets can be decommissioned. Columns in the spread sheet will typically contain the name, description and additional properties of the requirements and these will need to be mapped to fields inside Enterprise Architect using a specification.

Access

Project Browser | Right-click on Package | Import/Export | **CSV Import/Export**

Import Requirements from a CSV file Steps

Step	Action
1	Select a Package in the Project Browser where you want the Requirements imported.
2	Choose an Import Specification from the list a create a new one.
3	Select the source CSV file that contains the Requirements.
4	Select Run to import the Requirements.

Import Requirements from XMI

Enterprise Architect supports the XML Metadata Interchange (XMI) exchange format which is governed by the Object Management Group (OMG) and provides a convenient way of exchanging models or model fragments. It is quite common for requirements that have been developed for one system to be applicable to another system; this is particularly true of non-functional requirements. Each project typically needs a unique set of requirements that have their own lifetime so these are best exported from one model and imported to another. XMI provides a suitable exchange format for this purpose and the XMI can be easily exported from one model and imported into another.

Access

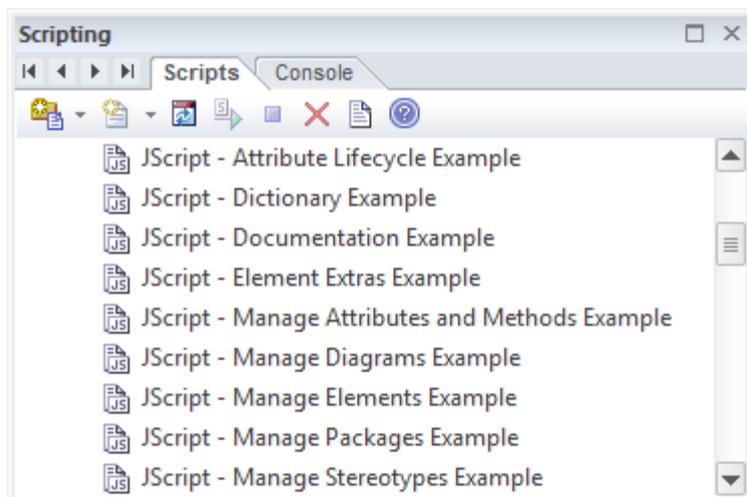
Project Browser | Right-click on Package | Import/Export | Import Package from XMI file

Import Requirements from XMI Steps

Step	Action
1	Select the location in the Project Browser where you want the content inserted, and right-click to display the context menu.
2	Select 'Import/Export Import Package from XMI File' from the context menu. The 'Import Package from XMI' dialog displays.
3	Select the filename of the XMI file to import, and select the appropriate options.
4	Select the Import button to import the Package Enterprise Architect will prompt you to confirm the import, select 'Yes' to import the file. If there are unsaved diagrams you will be prompted to save diagrams before continuing. If any elements in the import file exist in the model you will be warned, and will not be able to import the file unless you select the 'Strip GUIDs' option.

Import Requirements from Any Format

Enterprise Architect can import requirements from a wide range of file formats but on occasions the structure of the requirements in the source document, or the number of requirements can make using one of these built-in mechanisms difficult. Fortunately the requirements can be imported from any file format using the powerful and flexible scripting facility which is available in the core product or by developing an Add-in.



Import Requirements from Rational Doors

The Model Driven Generation (MDG) technology for Doors provides a lightweight bridge between Enterprise Architect and IBM (formerly Telelogic) Doors. This allows the analyst to import the Requirements from Doors

into an Enterprise Architect repository and to keep them synchronized with Doors. The entire hierarchy of Requirements will be imported and individual requirements can be linked to model elements such as Use Cases and Components. The Enterprise Architect model can be synchronized with Doors by re-importing the requirements. Any deleted items will be added to a 'Trash Can' Package but will not be deleted from the model or the diagrams.

Move Requirement External

Elements in Enterprise Architect can have internal requirements (responsibilities) which define what the element must accomplish. These can overlap or duplicate more formal requirements that the system in general must meet, so a modeler could decide to make an element's internal requirement into an external requirement element. This is often done to allow a number of elements to implement the need expressed in the requirement. This can be achieved using the move external function.

Access

On a diagram or in the **Project Browser**, right-click on element and select:

Properties > Requirements or
Properties Page | Requirements

Change an element's internal requirement into an external requirement element

Step	Action
1	Right-click on the internal requirement to change to an external requirement. A short context menu displays.
2	Click on the 'Move External' menu option. The 'Find Package' dialog displays.
3	Locate and click on the Package to place the new Requirement element in.
4	Click on the OK button . A new Requirement element is created in the target Package, with a Realization connector from the current element to the Requirement. In the 'Properties' dialog, the Requirement is now marked with a 'Yes' in the 'External' column, and the dialog fields are disabled. To edit its details, double-click on the Requirement. The Requirement element's own 'Properties' dialog displays.

Notes

- When an internal requirement is made into an external requirement element it is still viewable from within the element; any change to the external requirement will be reflected in the internal requirement and vice versa
- When an internal requirement is made into an external requirement element, the 'Stability' field and its value for the internal requirement are translated into the **Stability Tagged Value** in the external requirement

Move Requirements Internal

There are a number of situations where it is useful to define requirements inside an element. Requirements are typically created as elements in the **Specification Manager**, or as part of a Requirements diagrams or directly in the **Project Browser**. Enterprise Architect allows you to move (copy) an External Requirement into an element creating an Internal Requirement. This is quite commonly done so down-process workers like developers can see the Functional and Non Functional Requirements when working with a Use Case or Component. It can also be used as a device to list a series of applicable requirements under an element in a report. For example high level Business Requirements could be moved internal to a Business Process and if a report were generated the Business Requirements would be listed directly under the Business Process.

Change an element's internal requirement into an external requirement element

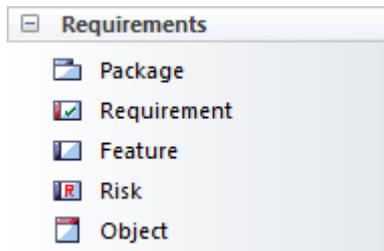
Step	Action
1	Ensure you have a diagram open with the element you wish to be the container for the Requirements visible.
2	In the Project Browser select the External Requirement you wish to change to an Internal Requirement.
3	Drag and drop the Requirement from the Project Browser on top of the element in the diagram.
4	A new Requirement element is created in the target element, with a Realization connector from the target element to the External Requirement.

Notes

- When an external requirement is made into an internal requirement element it is still viewable within the **Project Browser**. Any change to the external requirement will be reflected in the internal requirement and vice versa.

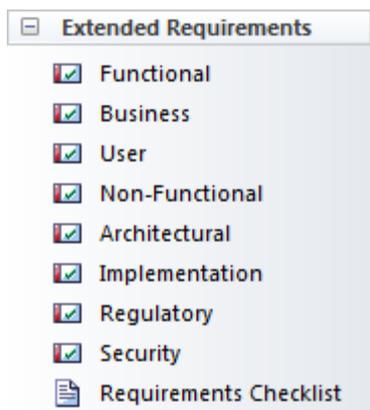
Recording Requirement Types

Enterprise Architect supports a wide range of element types and in addition to the elements defined by the various standards Enterprise Architect has added a series of extended elements that can be used for documenting a wide range of requirement types. When a Requirements diagram is created or opened Enterprise Architect will display the Requirements Toolbox which contains a number of standard and extended requirement elements. These elements can be dragged and dropped onto the diagram to create new elements including a Package that acts as a container for requirement elements and Features, Risks and Object elements.



Extended Requirements

There is also a convenient set of extended requirements that can be used to model a variety of diagram types ranging from Functional, Business, User to Architectural and Non-Functional requirements. These come pre-configured with tagged values that allow you to specify additional properties of the requirements.



Analysis

The analysis phase of Requirements development ensures that the Requirements discovered in the Elicitation phase have been articulated correctly and have the correct format, level of detail and properties and form a cohesive and correct set. As a result of the disparate sources and methods of elicitation the Requirements recorded in the elicitation phase will need some massaging and balancing - it is quite common for example to find repeated or overlapping requirements or to have omitted to record the concerns of one or more stakeholders. Tools such as the **Relationship Matrix** and the **Traceability window** will reveal omissions and issues with requirements. The **Element Discussion window** and **Model Mail** facility will also provide mechanisms for discussing the Requirements with other team members.

Target +	REQ011 - Manage User Accounts	REQ012 - Provide Online Sales	REQ013 - Manage Deliveries	REQ014 - ShoppingBasket	REQ015 - Process Credit Card Payment	REQ016 - Add Users	REQ017 - Remove User	REQ018 - Report on User Account	REQ019 - Manage Inventory	REQ020 - Receive Books	REQ021 - List Stock Levels	REQ022 - Order Books
+ Source												
Add New Titles												
Add To Shopping Basket				↑								
Close Account							↑					
Create Account						↑						
Create Orders												↑
Delete User							↑					

Models Used to Document Requirements

One of the most important aspects of the requirements engineering discipline is to communicate knowledge and ensure that all stakeholders have a clear and unambiguous understanding of the problem and the proposed solution. This can be challenging because the stakeholders typically cross organizational boundaries and have a myriad of backgrounds spanning from senior business executives to low level engineers. This heterogeneous audience will need a variety of communication devices to ensure the various stakeholders are able to engage with the requirements and are able to understand them. Enterprise Architect is a modeling platform with a formidable range of tools and features that can be used to model requirements in almost any way. These include modeling stakeholders, requirements, user stories, user interfaces and a wide range of other models.

Requirements models

Model	Description
Textual Requirements	Textual requirements can be modeled using the requirements element and users can choose to work with the elements in a text tool such as the Specification Manager , directly in the Project Browser in a hierarchy or visually in a diagram. The requirement element can be connected to other elements to describe a hierarchy of requirements or to business goals and also use cases and user interface models. Using the Specification Manager Enterprise Architect allows the modeler to create, analyze and manage requirements in a text tool that resembles working in a spread sheet but which is more powerful and gives the analyst access to other models including the glossary and the domain models.
Stakeholders	Stakeholders can be modeled using UML Classes and descriptions can be added that describe the stakeholder. Stakeholders are possibly the most important entities in the requirements engineering discipline and creating elements to represent them in the model allows them to be used as the owner of requirements and business rules. They can be placed onto diagrams allowing them to be visible in elicitation and prioritization workshops.
Glossary	A Glossary can be created and managed using the Project Glossary ensuring that important project and domain terms are accessible right inside the model. These terms can be inserted into the notes fields of elements including Use Case and User Story descriptions.
Use Cases	Use Cases can be modeled in a use case diagram and can be connected to a range of other elements including user interface models, User Requirements and Components. The Use Cases can be kept light-weight by just completing the description or they can be fully-dressed using the powerful Scenario Builder Tool . Use Cases often present a problem for the requirements analyst because the diagrams are typically drawn in a diagramming tool and the text is written in a word processor making it inaccessible to other model elements. Using Enterprise Architect's Scenario Builder the Use Case descriptions can be completed inside the Use Case itself inside the modeling tool. The tool can also produce behavior diagrams that represent the Use Case Scenarios automatically from the model.
User Stories	User Stories can be modeled using a stereotyped Use Case element and the text of the story can be completed in the description field. The Users and Personas can also be modeled and related to the story. Enterprise

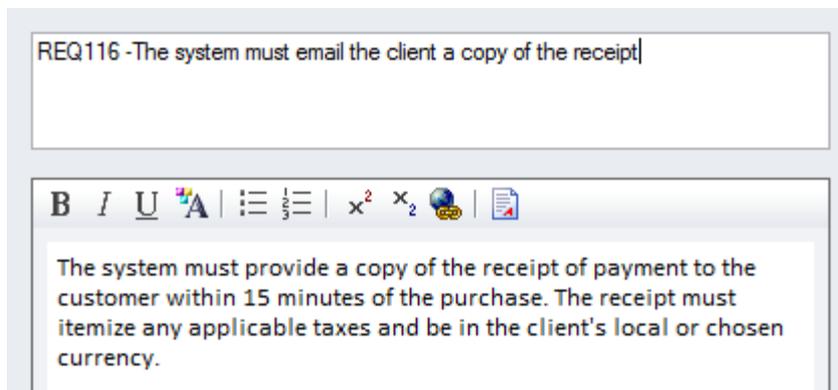
	Architect allows the modeler to work with the stories in text form or in diagrams. Functional requirements can be added in preparation for handing to the development team for an iteration and these can be managed inside or outside the user story.
Domain Models	Domain models can be modeled using a UML Class diagram and the important entities in the business domain can be recorded, detailed and related to other elements. Creating domain models early in a project helps stakeholders make sense of all the important entities in a domain and can be used to generate a Data Dictionary. The domain elements can be created as links in textual statements of requirements creating an articulated model that facilitates communication and understanding.
Process Models	Business processes are a useful way of recording the activities of a business including the events that trigger them to happen, the information that is produced or consumed, the outcomes and the roles that carry out the work. Enterprise Architect supports BPMN, UML and SysML Activity diagrams that can be used for this purpose.
Storyboards	Storyboards can be modeled using graphic elements in diagrams and a slide-show can be created to walk through the story.
Wireframes	Wireframes can be modeled using the powerful Wireframing diagram, which has built-in support for popular hand held devices such as Apple and Android phones and tablets, and also for modeling dialog windows and web pages. Using Enterprise Architect's Wireframing tool, an analyst can create powerful models of the arrangement of the application's content, describing interface elements and navigational mechanisms. Analyst and experienced designers have typically worked in isolation from other disciplines, but using Enterprise Architect the models can be created and maintained inside the same tool as the other requirement models, allowing traces to be created between other elements and the controls and content in the Wireframes.
User Profiles and Personas	User Profiles and Personas can be modeled using a stereotyped Actor element which allows descriptions and properties to be added that describe the persona.
System Events and Responses	A system will typically respond to a number of events and can also be responsible for creating events such as raising an alert or sending a data stream. These can be modeled in Enterprise Architect using BPMN or UML and SysML Activity diagrams.
System Interfaces	System Interfaces can be modeled using Provided and Required Interfaces and Ports on a Component diagram which describe how the software or hardware system interacts with other systems or how the internal Components of a system communicate. Enterprise Architect has rich support for modeling the interfaces and error codes and other behavior can be modeled. The interfaces can be linked to data definitions and Application Programmer Interface (API) specifications and a range of model elements including Use Cases and Requirements. The Interfaces can be added to documentation such as the System Requirements Specification and this document can be automatically generated from the model.

Requirements Naming and Numbering

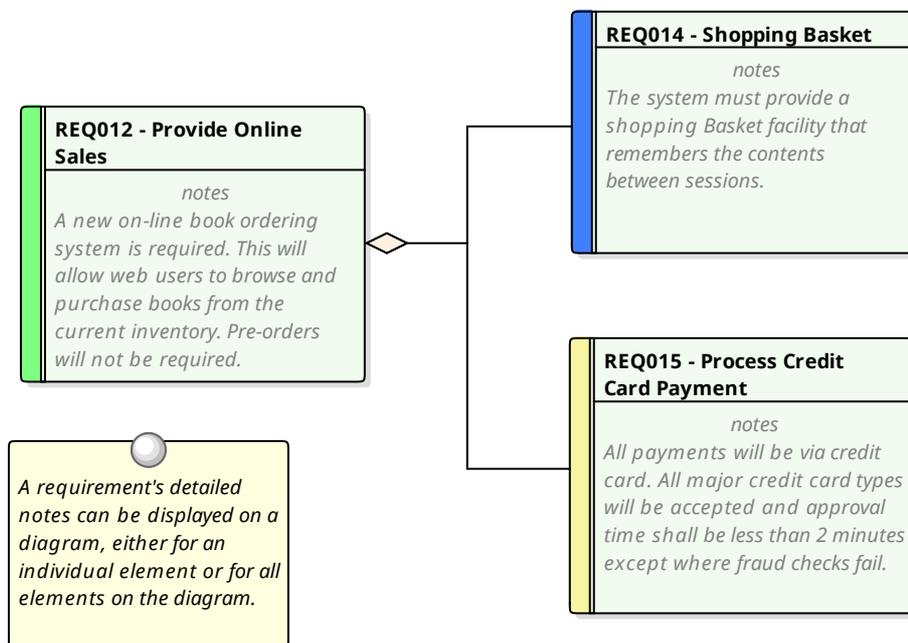
Requirements are fundamental to the definition of a problem (or opportunity) and the solution must be traced back to this definition.

Requirement Names and Descriptions

There are many schemes that are used to name requirements and Enterprise Architect is flexible enough to support any scheme that is used. There are two locations you can add textual information to a Requirement: the element name, which has a limit of 255 characters, and the 'Notes' field, which is effectively unlimited. Some schemes specify that a single definition of the requirement is entered and no notes are needed. Other schemes prescribe a short name and specify that the requirement is clarified with detailed text. If notes are not used it is common practice to use some type of numbering system so the Requirement can be referred to unambiguously.



When Requirements appear in diagrams the name will by default be displayed but a modeler can choose to display any one of a number of the requirement's compartments including the notes. This technique creates expressive diagrams that reveal the details of the requirement and help the reader or reviewer to understand the Requirement more fully.



Sequential Numbering

Good practice often recommends that Requirements are given a sequential number when they are created so they can be referred to in stakeholder workshops, change requests, conversations with System Integrators or implementation teams. Using a name in this situation is often unwieldy and subject to error so a sequential number is preferred. Enterprise Architect has a facility called **Auto Names and Counters** for this purpose that can be used to assign a sequential number to any element type including Requirements. It includes a prefix definition, a counter and a suffix definition allowing numbers such as: 'REQ007 - Manage Inventory' to be created. These can be further refined to numbering systems such as this architectural requirement: 'ARR134 - Payloads for internal component interfaces must use an XML format'.

Type: Requirement

Name: Prefix: REQ, Counter: 001, Suffix: - Apply on creation

Alias: Prefix: , Counter: , Suffix: Apply on creation

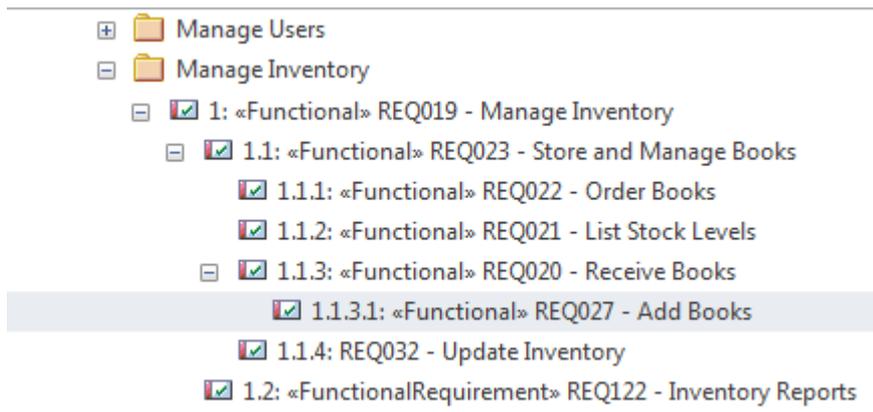
Buttons: Save, Close, Help

The counter is added to the name and is displayed in all views of the repository including the **Project Browser, Relationship Matrix, Search Results** and Diagrams.

Numbered Hierarchies

When Requirements are written in word processors they typically use a numbering scheme called Outline Numbering, which assigns a number to the first level heading such as: '4 Inventory Requirements' and then a sub-heading is numbered by adding a period and a number such as '4.1 Stock Levels' and again down a another level '4.1.1 List Stock Levels'. Enterprise Architect has a facility called Level Numbering that applies

hierarchical numbering to the elements in a Package. This is a useful mechanism that is displayed in a number of locations including the **Project Browser**, the **Specification Manager** and Diagram and Packages Lists. It must be remembered, however, that if the order or the level of the elements in the Package is altered they will be assigned new numbers based on their new position; this makes this mechanism unsuitable if immutable numbers are needed.

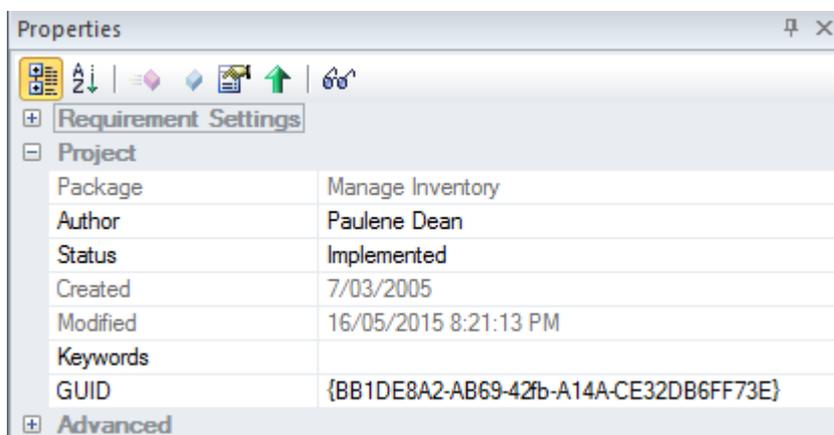


Numbered Packages

This is a hybrid method where Packages are used to create a high level naming and numbering structure and the Requirements in each Package are numbered using the Package identifier and a number to identify them. So Requirements for the Fulfillment of Orders could be contained in a Package named '2.4 Fulfill Orders' and an individual requirement in this package could be named 'FO-01 Process Credit Card Payments'. This would be manually maintained or a Script could be written to ensure that the numbers were correctly assigned.

Globally Unique Identifier

Every element, diagram and connector in an Enterprise Architect repository is given an immutable and unique reference in the form of a Globally Unique Identifier (GUID). The GUID is assigned to the element when it is created and is guaranteed to be unique across time and space. Thus requirements can ultimately be referred to by this unique identifier. While the GUID is a powerful and irrefutable way of referring to a Requirement it is not practical to use it in discussion with stakeholders because of its length and the fact that it is not memorable. The GUID's purpose is to be able to track and manage a Requirements provenance particularly when Enterprise Architect is used to generate Requirements to other tools. It is also used as the identifier in the XMI exchange format.



Proprietary Numbering Systems

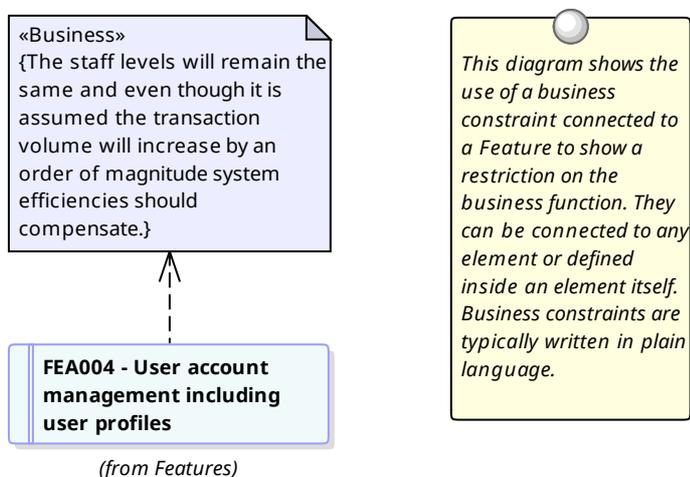
There will be projects or programs of work that will for regulatory or commercial reasons specify a proprietary numbering system that must be used with Requirements. For this reason one of Enterprise Architects in-built schemes may not suffice and in this situation the user can create their own numbering scheme using the **Scripting** facility in combination with **Tagged Values**.

Model Assumptions and Constraints

When an analyst is working through the information acquired from the elicitation process they will typically come across statements or conditions that are best described as Assumptions that have been made or Constraints that will restrict the solution in some way. These are not Requirements but have an important role in the requirements development process because they have the ability to affect the solution and must be understood.

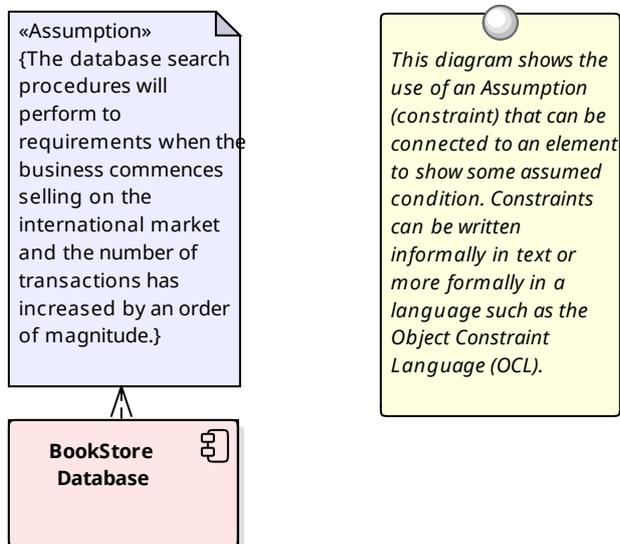
Business Constraints

A Business Constraint is a business restriction or limitation imposed on the choices that can be made for the design, implementation or deployment of the solution. They are typically restrictions of budget, time, resources but can be any type of limitation such as the context of the business deployment where the solution must not change the way that branch staff currently work. A Business Constraint may also limit the access or presentation of information such as 'Only the last four digits of a credit card number can be displayed in reports.' There is some overlap with business rules and the analyst should be careful to separate the two notions. Business Constraints can be modeled in Enterprise Architect using a Constraint or a stereotyped requirements element and they can be related to one or more model elements using a dependency relationship. Constraints can also be created as a property of an element using the element **Properties window**.



Assumptions

An assumption is a statement that is believed to be true but that has not yet been verified. It is imperative that assumptions are modeled and attempts are made to verify them as they have the potential, if proved to be false, to significantly change the definition of the problem and therefore the solution. They can be statements made about the way things are currently done or they could pertain to a future process or solution. Assumptions are similar to Risks and good practice would prescribe them being managed in a similar way to Risks. Attempts should be made to verify them as early as possible in the requirements development phase. An example of an assumption is: 'The User will know the meaning of toolbox icons as used in other Windows applications'. Based on this assumption the solution designer might plan not to implement tool tips for the icons. Assumptions can be modeled in Enterprise Architect using a Constraint element, available from the Common toolbox page or as a stereotyped Requirements element. They can be related to one or more model elements using a dependency relationship.

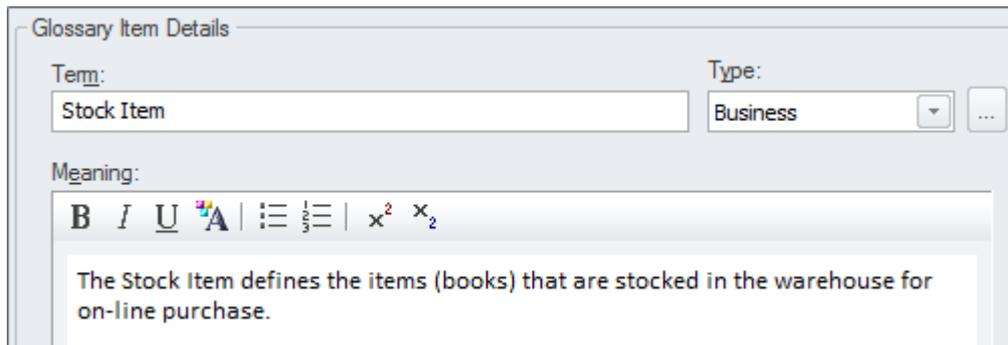


Technical Constraints

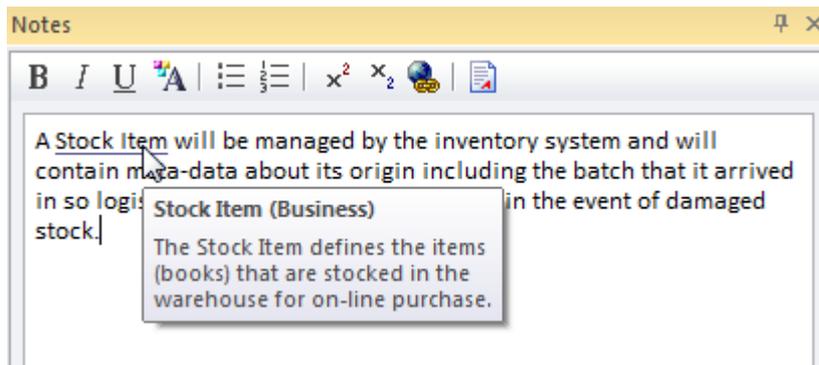
A technical constraint is any restriction on the choices that can be made for the architecture, design, implementation or deployment of the solution. They can start with principles defined in the enterprise architecture that restrict the types of platforms, programming languages and decisions to buy or build part of the solution. They could also be restrictions on the type of protocols or standards that the solution must implement or comply with. Restrictions on file sizes and formats can also place limitations on the solution choices. There is some overlap with non-functional requirements and the analyst should be careful to separate the two notions. Technical constraints can be modeled in Enterprise Architect using a Constraint element available from the Common toolbox page or as a stereotyped Requirements element. They can be related to one or more model elements using a dependency relationship.

Create a Glossary

One of the fundamental precepts of requirements engineering is to promote understanding and convergence of thought and to remove ambiguity. It is quite common for members on a project team to have quite disparate understanding of domain concepts. This can be easily rectified by creating a **Project Glossary** early in the project and ensuring the glossary is accessible to the project team. Enterprise Architect has a built in Glossary that allows the requirements analyst to define terms and meanings, and to create categories (types) of terms.



One of the features that makes the glossary powerful is the ability to link from text fields such as element notes to terms in the Glossary, and a rollover in the text displays the meaning.



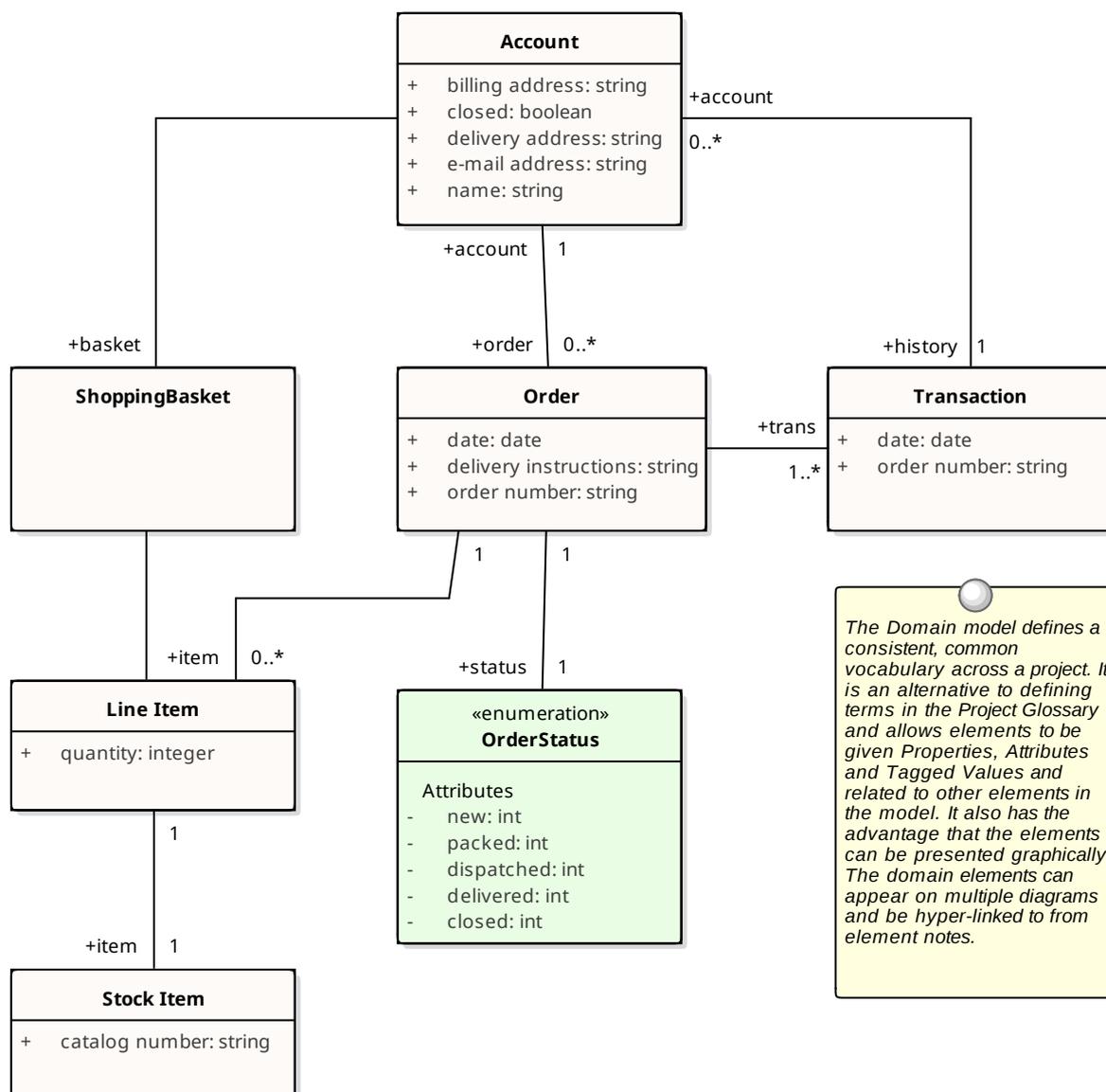
Notes

Experienced modelers tend to define the terms of a domain in a Domain Model and reserve the glossary for project and process terms and their definitions such as Use Case, Software Requirements Specification, Metric etc. This allows relationships between domain terms to be defined and data elements (Attributes) can be added including datatypes creating a more expressive representation than would be possible in the Glossary.

Create a Domain Model

Requirements analysis will reveal a number of business terms that must be defined if the requirements are to be understood and clarified. There are a number of options for recording these terms, including a Domain Model or the **Project Glossary**, which is a purpose built lexicon through which you can list, define and categorize terms. The Domain Model (business object model) is another powerful mechanism for describing the important terms of the business providing a single definition of the terms and their relationships that is accessible to all project staff from high level business managers to low level engineers. One of the advantages of using a domain model is that the terms are modeled as elements, allowing them to be linked to other elements within the domain model itself or to elements in other parts of the models.

Enterprise Architect has extensive support for modeling a domain using a **UML Class Diagram** or the Classes can be created directly in the **Project Browser** and displayed in a list using the **Package List window**.



The Domain Model elements can be added to other diagrams to make them more expressive and show important information such as what data is being consumed or produced by a business process or what data is being passed between application interfaces.

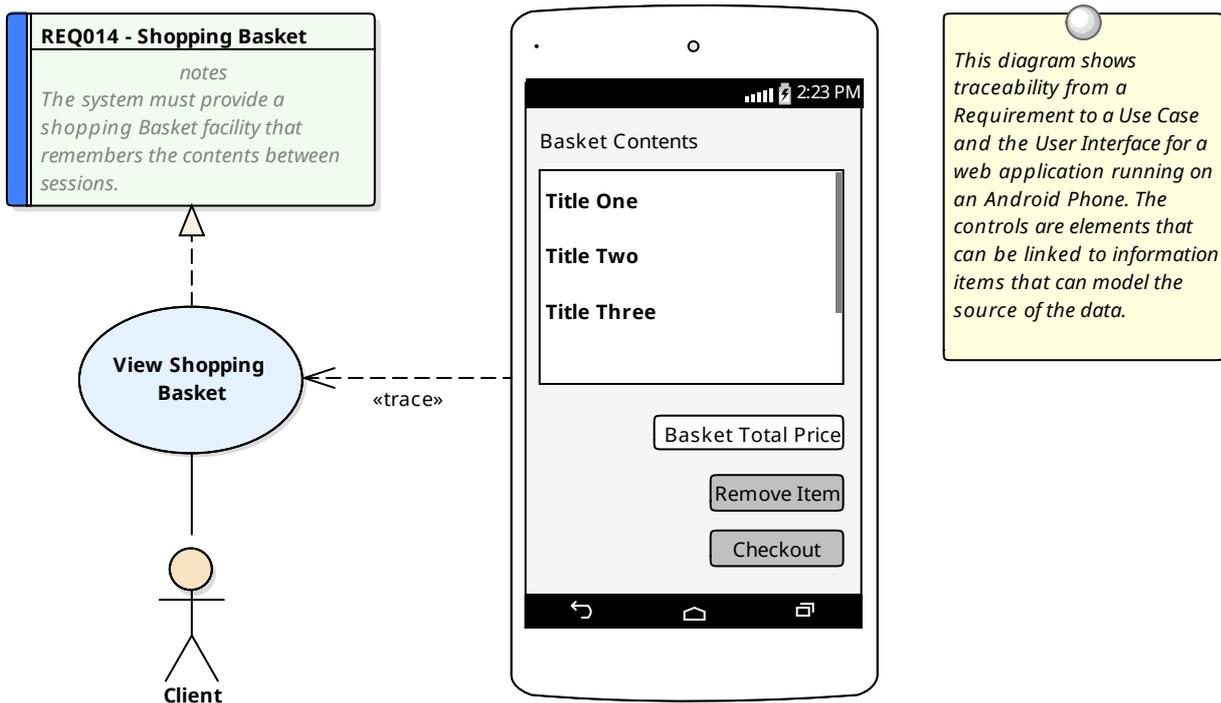
Enterprise Architect allows you to link to the domain elements (or any element) from the text of a requirement's notes. (This facility is available from any element.)

Notes

Experienced modelers tend to define the terms of a domain in a Domain Model and reserve the Glossary for project and process terms and their definitions such as Use Case, Software Requirements Specification, Metric etc.

Model the User Interface

The User Interface for a system can be modeled in a variety of ways depending on whether the interface is for the Web or for Win32 applications or for Web Applications targeting mobile devices.



Prioritize the Requirements

Prioritization of requirements is imperative to the success of a project as it ensures that analysis, development, testing and implementation resources are focused on the most critical aspects of the system. Prioritization is a decision process that allocates a priority to the each requirement, the most common criteria for the categorization is business value. Business value is typically determined by the cost-benefit analysis of the value the implemented requirement will produce for the organisation or its customers. Some other factors include: policy or regulatory compliance, urgency, business or technical risk and likelihood of success.

Changing the Priority Collaboratively

The process of selecting criteria and assigning priority is typically collaborative, and is often done in a workshop with stakeholders or their representatives debating the categorization. In previous eras this was a laborious and difficult process, but Enterprise Architect has some powerful features for working with requirement properties, including priority. There are a number of windows that support working with the requirements in a list and editing the priority in-line, automatically filtering or sorting the list of requirements based on the newly assigned priority, including the Package List and the **Diagram List**. The **Specification Manager** is a useful tool for this purpose and provides a text based interface where the requirements and their notes can be viewed and priorities can be selected from a drop down list. The interface also displays a number of other useful properties that are typically useful for prioritization, such as Status and Complexity.

Item	Priority	Notes	Status
REQ012 - Provide Online Sales	High	A new on-line book ordering system is required. This will allow web users to browse and purchase books from the current inventory. Pre-orders will not be required.	Validated
REQ014 - Shopping	Medium	The system must provide a shopping Basket facility that remembers the contents between sessions.	Approved

Requirement Priority Property

There are a wide range of criteria that can be used for prioritization and each organisation and project will typically use some type of weighted average to determine the priority. Enterprise Architect has flexible and complete support for requirement prioritization as each element has a built in Priority property that can be set to indicate its priority, allowing the user to select the allocated priority from a drop down list

Author: Paulene Dean

Difficulty: Medium

Priority: Medium

Version: High
Medium
Low

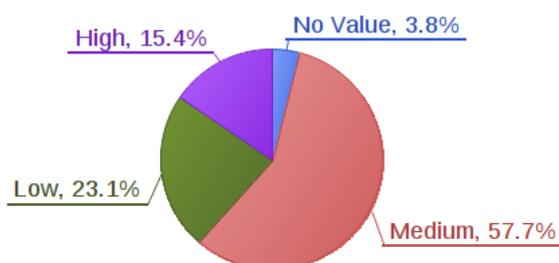
Phase: 1.0

The list of priorities is conveniently pre-loaded when you install Enterprise Architect but these can be edited or completely revised to suit an organization or project. They can even be imported as reference data from a previous project or if the current project was created based on a template the organization's priorities could be pre-loaded from the base model.

Dashboard Diagrams

Enterprise Architect has a series of dashboard diagrams that can be used to create a compelling view of the Priority of requirements in a Package with the option to include sub-Packages. There are a number of pre-configured charts that can be used to display the ratio of Priority values for Requirements in a part of the model. Filters add another level of user configuration allowing a modeler to, for example, exclude requirements of a particular Status or ensure only requirements for the current phase are displayed.

Elements by Priority



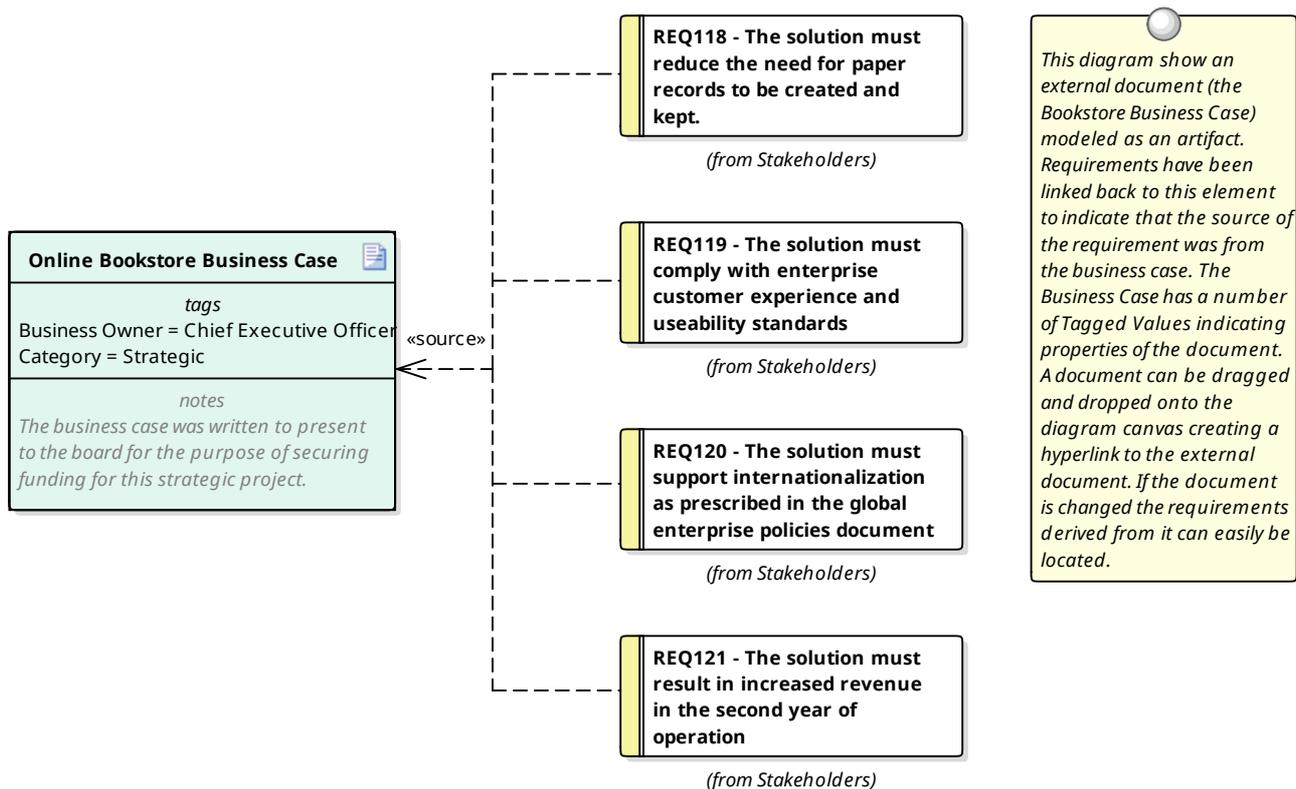
This diagram shows a Pie Chart element depicting element priority for all the requirements in a selected package. It provides a useful summary for a requirements manager and is dynamically updated when the Priority changes and the diagram is reopened. There are a range of other pre-defined charts and user defined charts can also be added. A filter has been added to exclude all elements other than Requirements.

Extension for Setting Priority

When there a large number of requirements and the basis for the prioritization has been well defined a script or add-in could be developed that could assign the Priority automatically based on a specified algorithm. Additional properties such as business value could be assigned using **Tagged Values** and any requirements that did not have the requisite properties set could be written to a log or presented as a search result list. The script or add-in could be used on multiple projects giving a consistent and unbiased result.

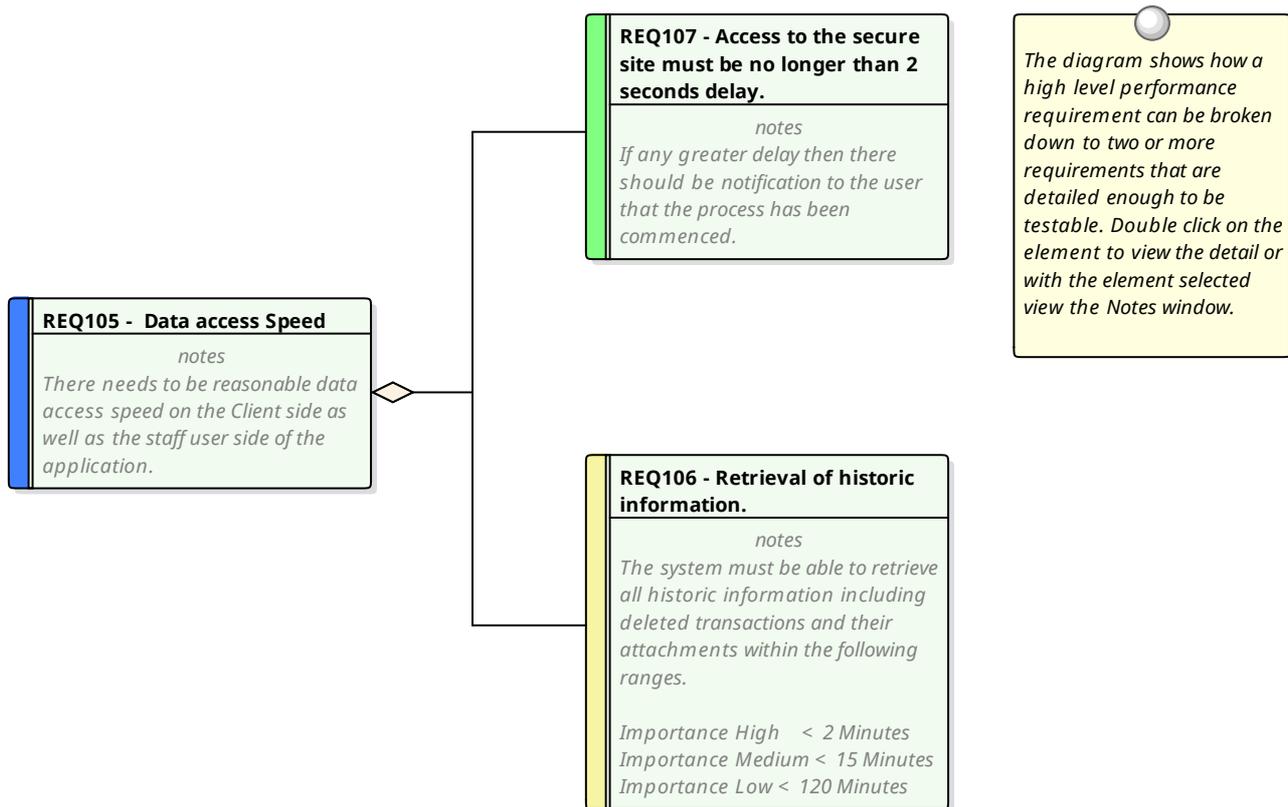
Specification

Requirements are typically derived from a wide range of sources and there often a number of requirements analysts working on a project and this will tend to make the Requirements disparate. Ensuring the Requirements are consistent and of high quality is critical to the success of a project.



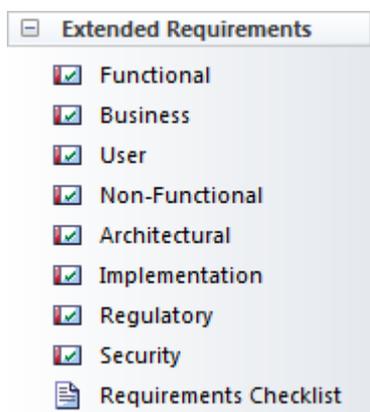
Specify Quality Attributes

Non Functional Requirements and Quality Attributes are important inputs into the Architecture of a system and provide the criteria that defines what the stakeholder's expectations are about how well the system will operate. Regardless of how well the Functional Requirements have been defined and implemented if the Quality Attributes aren't built into the architecture and implemented the system is unlikely to be satisfactory to its stakeholders and users.



Adding Non Functional Requirements

Enterprise Architect conveniently has a number of Requirements such as the Non Functional, Architecture, Implementation and Security Requirement types available from the Extended Requirements page of the Requirements Toolbox. These can be dragged onto a diagram or created directly in the **Project Browser** or **Specification Manager**.



Requirements Packages

Quality Attributes are often grouped into sets such as those pertaining to Security or to Performance or one of the '-ilities' such as Reliability. A requirements Package can be used to group these elements together which provides a convenient categorization for reporting. The Packages can be seen in the **Project Browser** and can also be added to a diagram with the option to display the individual Requirements inside the Package.

Extensibility

- + REQ100 - System must be easily extendible
- + REQ101 - Other product types options can be added easily.
- + REQ102 - System must be able to cope with regular retail sales

Reliability

- + REQ112 - 2000 hours mean time between failure.
- + REQ113 - Must be recoverable quickly.
- + REQ114 - 99.999% accuracy.
- + REQ115 - 99.999% precision.

Security

- + REQ108 - Processed information must be kept secure.
- + REQ109 - All transactions must be secure.
- + REQ110 - Wherever possible existing security definitions should be used.
- + REQ111 - Physical storage locations should be secure.

The Non Functional Requirements have been defined into a number of packages and these can be displayed in a diagram including the Requirements they contain.

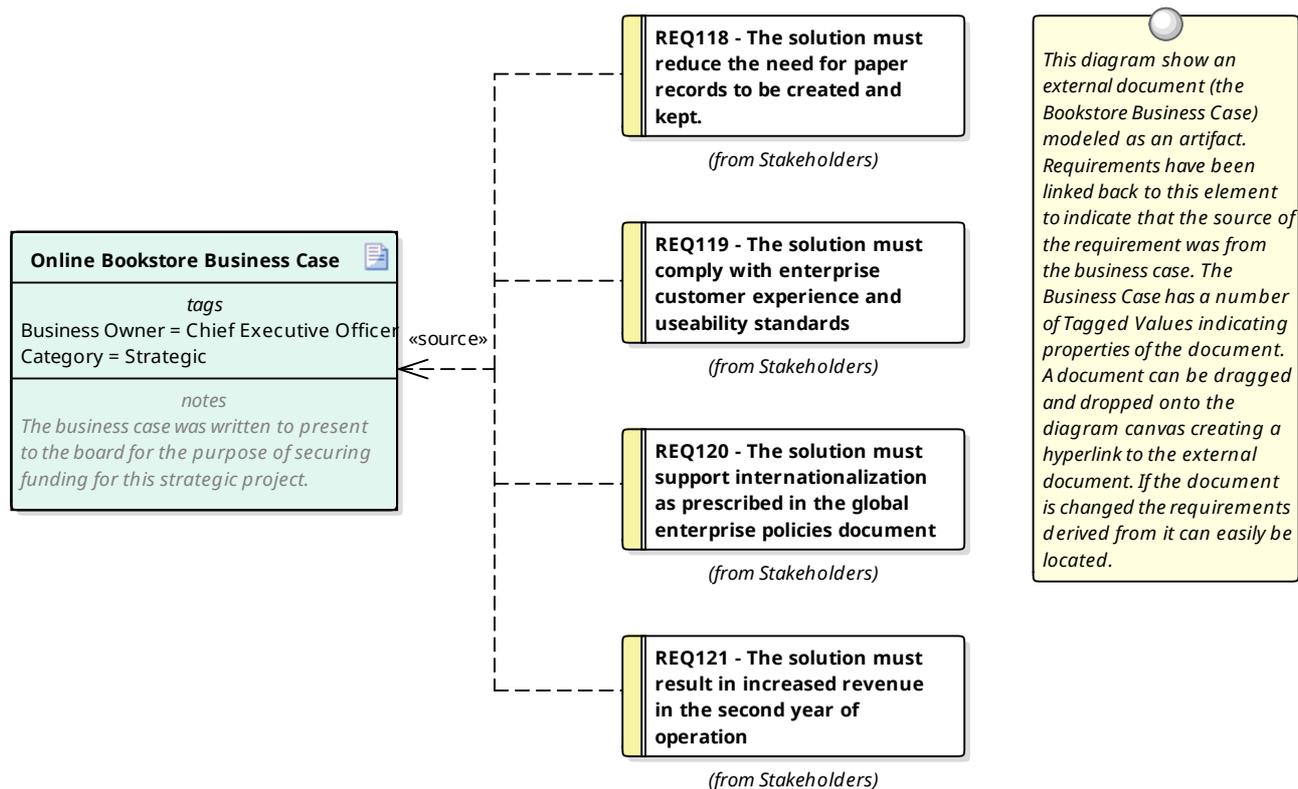
Requirement Sources

Much of the literature on requirements development talks of 'gathering' requirements which implies that the analyst is walking through an orchard picking low hanging fruit. In practice the process is quite the opposite and the requirement analyst needs to have all the skills and tenacity of a detective to discover the sources of requirements. The requirement sources and the documents and artifacts that are discovered are first class citizens and can be modeled in Enterprise Architect to provide a register of the requirement sources. This is a list of common requirements sources:

- Interviews with users and other stakeholders
- Observations of users performing tasks
- Business Case or Proposal
- Concept of Operation or Vision document
- Procedure manuals and user task lists
- Enhancement Requests for the existing system
- Marketing material and product definitions
- Analysis of a market leader or competitor's products

Modeling Requirement Sources

Requirement sources are typically documents or other file based artifacts and are best modeled as UML Artifacts using **Tagged Values** to add meta-data to the element to record such information as its type, location, who it is governed by and any other relevant properties. The document and files will typically be stored in a variety of places across an organizations networks, but so long as there is a file path or a URL they can be opened from within Enterprise Architect by a single key stroke. The artifact will be opened either inside or outside Enterprise Architect using the appropriate editor for the file type.



Creating artifacts to model requirement sources

Step	Action
1	<p>Ensure you have a diagram open that will be the canvas where you will create the elements. Open a file system browser such as MS Windows Explorer and drag-and-drop the file to the diagram canvas.</p> <p>A window will be displayed.</p>
2	<p>Select Artifact (External) from the window.</p> <p>Enterprise Architect will add a new Artifact to the diagram and prompt you to enter a name.</p>
3	<p>Enter an appropriate name for the requirement source or leave blank to use the file name for the element name.</p>

Elaborate the Requirements

Once the information that acts as a precursor for the Requirements has been analyzed and the Requirements have been created in Enterprise Architect they must be elaborated with properties and additional information to make them meaningful to the stakeholders who own them and the implementation teams that will use them to design and build the required system.

Setting Requirement Properties

When Requirements are created they will be given a name, but will often benefit from having additional information recorded about the requirement that will add clarification and details needed by the Stakeholders or the implantation team. These details can be entered into the 'Notes' field for the element.

REQ116 -The system must email the client a copy of the receipt|

B I U A | | x^2 x_2

The system must provide a copy of the receipt of payment to the customer within 15 minutes of the purchase. The receipt must itemize any applicable taxes and be in the client's local or chosen currency.

In addition there a number of built-in properties that can be set for each requirement, the most important of these are listed in this table; other user-defined properties can be added using **Tagged Values**.

Type:

Status:

Alias:

Keywords:

Author:

Difficulty:

Priority:

Version:

Phase:

Important Requirement properties

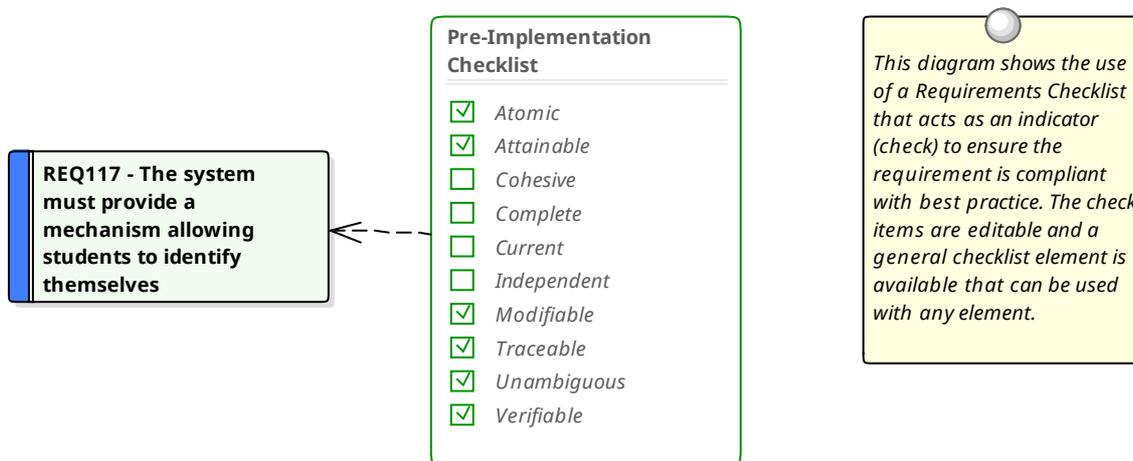
Property	Description
Status	The Status is used to record the condition of the requirement; the modeler can select it from a drop-down list of predefined values. Setting the Status

	helps project managers and others involved in planning determine if a Requirement is ready to be included in a development Package of work or iteration.
Alias	The Alias can be used to record a sequence number or identifier for the Requirement. A modeler can choose to display the Alias in a number of locations including diagrams and reports.
Priority	The Priority is used to set the precedence or relative importance of a Requirement and the modeler can select it from a drop-down list of predefined values. Setting the Priority helps project managers and others involved in planning to determine those Requirements that should take precedence over others for implementation.
Difficulty	The Difficulty is a measure that attempts to describe the relative effort that would be required to realize the requirement in a system implementation. The modeler can select a value from a drop-down list of predefined values. Setting the Difficulty helps project managers and others involved in planning determine the effort needed to implement the Requirement.
Phase	The Phase is a value that can be set to indicate the stage of development at which the Requirement will be implemented. The modeler can enter any value into this field. Setting the Phase helps project managers and others involved in planning determine which Package of work or iteration the Requirement should be included in.

Validation

Requirements validation is necessary to make sure the Requirements are of a high standard and suitably define the Customer's problem (or opportunity) and are sufficient for the implementation teams to design and implement the product. It is imperative that the requirements have the desired level of quality and are complete and necessary. There are a number of ways that Requirements can be validated but probably the two most common ways are to: perform team reviews and to assign test cases to the requirements.

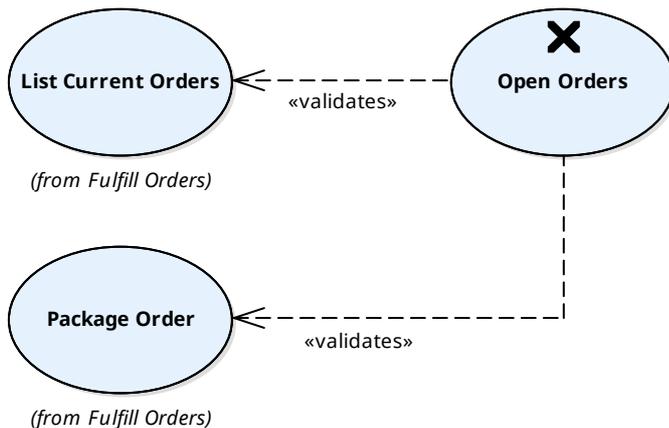
The team reviews are typically conducted by team members or other analysts who have some familiarity with the domain but were not themselves responsible for the requirements development or management. Enterprise Architect has a powerful tool to assist with this process called the **Team Review** which works across the entire model and allows reviewers to record their findings and to reference model elements. There is also a Requirements Checklist element available from the Extended Requirements page of the Requirements Toolbox that provides a useful mechanism for checking the quality of Requirements.



Test Cases can be defined at a number of levels from User Acceptance tests down to Unit tests. Defining the test cases early in the requirements development process creates a double check on the Requirements because when test cases are defined issues with the Requirements are often uncovered. Enterprise Architect has a number of facilities to define test cases and a modeler will need to decide which is the most appropriate for the endeavor.

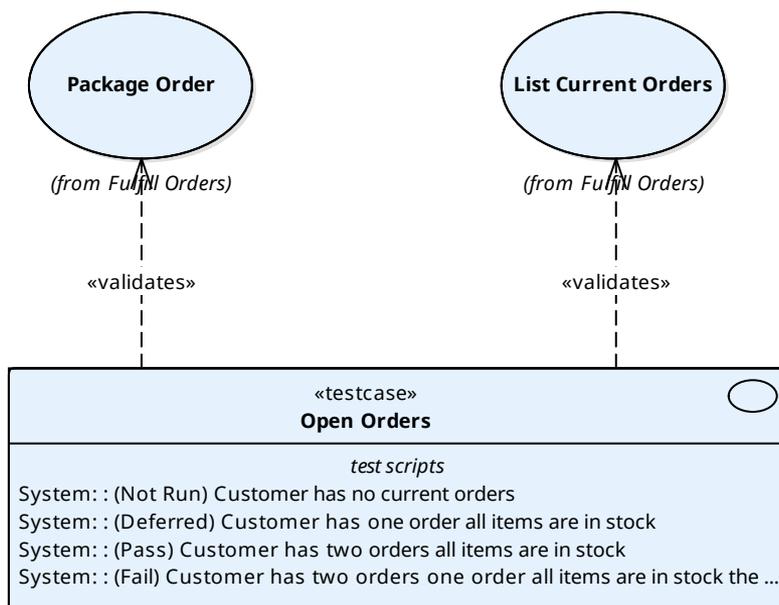
Derive Test Cases

The definition of Test Cases acts as a double check on the quality of Requirements as test designer bring their own unique and often orthogonal view of the requirements. Best practice suggests that the testing team should be independent and isolated from the requirements team thus providing a fresh set of eyes on the Requirements. It is quite common for one Test Case to test a number of requirements or for a Requirement to be tested by one or more Test Cases. These relationships can be modeled in Enterprise Architect using the Test Case element which can have detailed test scripts defined.



Detailing the Test Cases

Any element in Enterprise Architect can have tests defined and these can be one of a number of types such as Load, Regression or Standard and can be defined as a class of test such as User Acceptance, System, Unit etc. The details of the tests including when they were run and the status of the test indicating whether it passed or failed can be recorded.

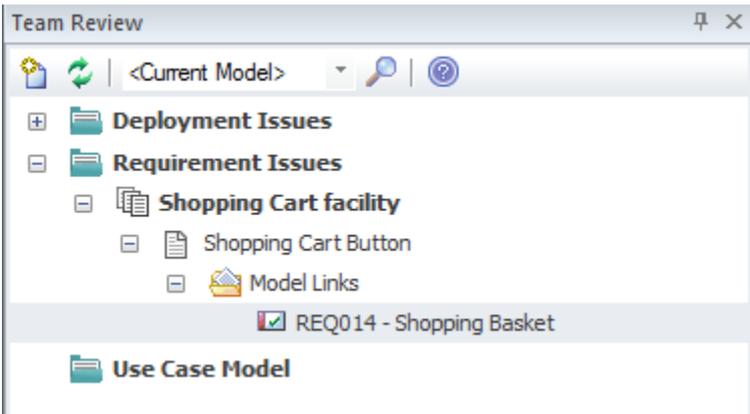


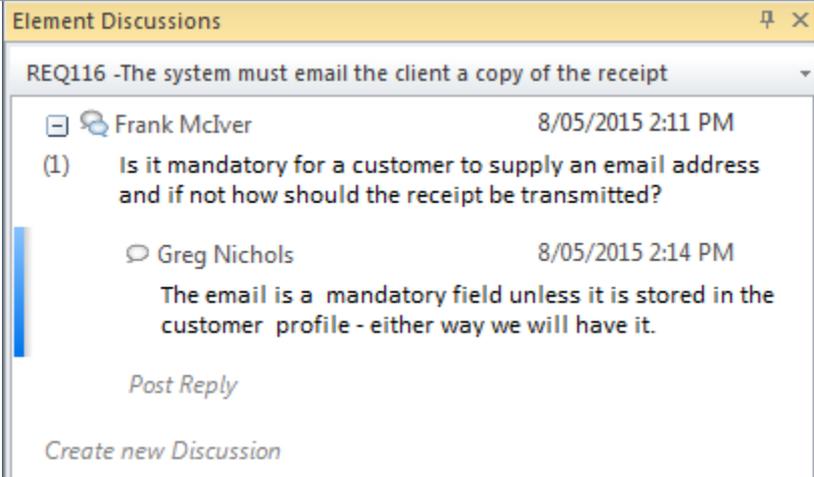
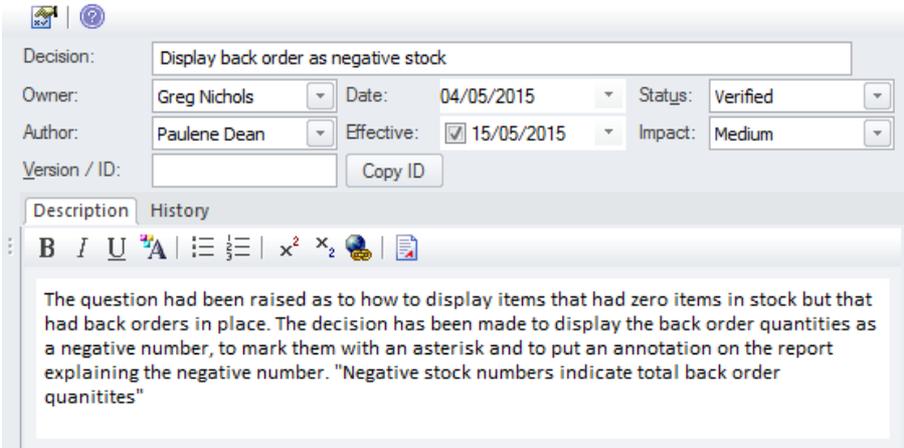
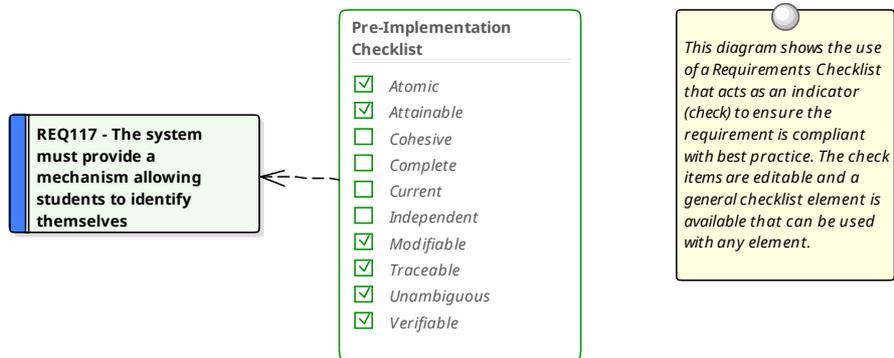
This diagram shows a Test Case presented in rectangular notation. A dependency relationship stereotyped with validate has been used to model the fact that the test case validates the functional Requirements expressed in the Use Cases. A number of Scripts have been written in the Test Case and the data describing the tests including the runs have been recorded.

Review Requirements

Requirements are typically discovered and written by a variety of team members and are commonly sourced from a variety of documents and stakeholders at disparate times. All of these factors lead to a tendency for the requirements to lack quality, be inconsistent and to contain issues. Best practice encourages early and continuous reviews of requirements to detect and remove defects before the requirements reach the design and implementation teams. Enterprise Architect provides sophisticated tools for reviewing requirements including the **Team Review** tool, **Element Discussions** and Maintenance items.

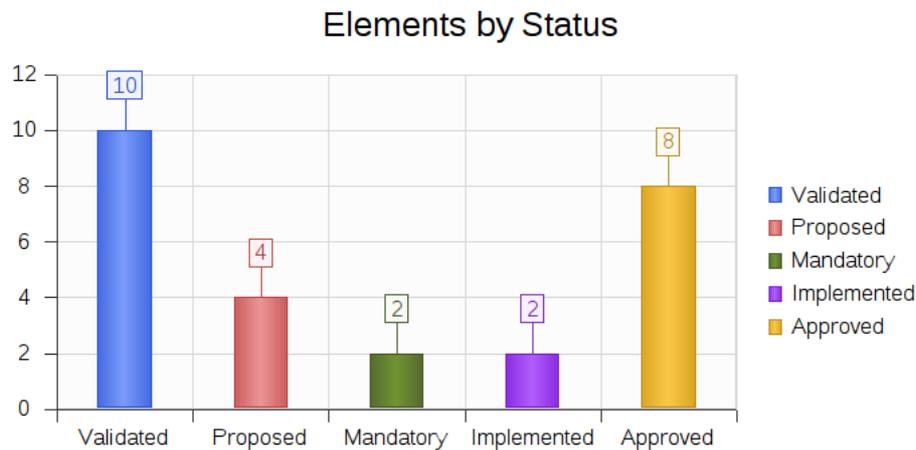
Requirement Review Tools

Review Tool	Description
Team Review	<p>The Team Review is a powerful tool that allows team members to collaborate and review the contents of the repository. It is particularly useful with requirements validation as it allows a set of requirements to be discussed and reviewed. Each review could be set up as a category and then different types of reviews could be setup as topics with posts being used to enter review items. Model elements can be added to a post, allowing - for example - a post to report two or more conflicting requirements. Team members would have the chance to post replies and view the linked elements. Statuses can be applied to indicate the progress through a workflow. The Team Review tool can be used in conjunctions with the Element Discussion window (where individual elements are managed) and the Discussion window (which contains a composite list of all element discussions in the repository).</p> 
Element Discussions	<p>During the progress of a requirements review there will inevitably be ambiguities, issues and defects detected. Before these are entered against the requirement as maintenance items it is common for analysts conducting the review to want to discuss these items with team members or stakeholders. The Element Discussion window is a useful place to record this information allowing other team members to post replies. This is a particularly powerful facility because in many requirement tools these discussions are typically written in the text of the element making it difficult to produce customer ready documentation until they are all removed. It is quite common for modelers to enter discussions even outside a formal review and these will assist the reviewers to resolve requirement defects or issues.</p>

	
<p>Element Maintenance Items</p>	<p>As Requirements are developed Changes will inevitable occur, Issues will be raised, Events will happen, Defects will be found, Tasks will need to be assigned and Decisions will need to be made. Enterprise Architect has a useful set of facilities to manage these items through the Project Maintenance facility. There is a window for each of these maintenance items where details of the item such as its Name, Description, Author Completion Date and more can be assigned.</p> 
<p>Requirement Checklist</p>	<p>Enterprise Architect has a convenient Requirements Checklist element available from the Extended Requirements Page of the Requirements Toolbox. This is a convenient way of recording visually whether a Requirement is of high quality and is a useful mechanism for team reviews.</p> 

Requirements Management

This comprises the activities to maintain a set of requirements that represent an accord or agreement between the project team and the customer. It also involves ensuring that the requirements are acceptable to the design and implementation teams and that they are sufficient so that what they specify can be implemented into working business, software or hardware systems. Enterprise Architect is a sophisticated platform for managing requirements and regardless of the domain, the size of the project or the method being followed there are tools that will make it straight forward to manage even large repositories of requirements in complex projects.



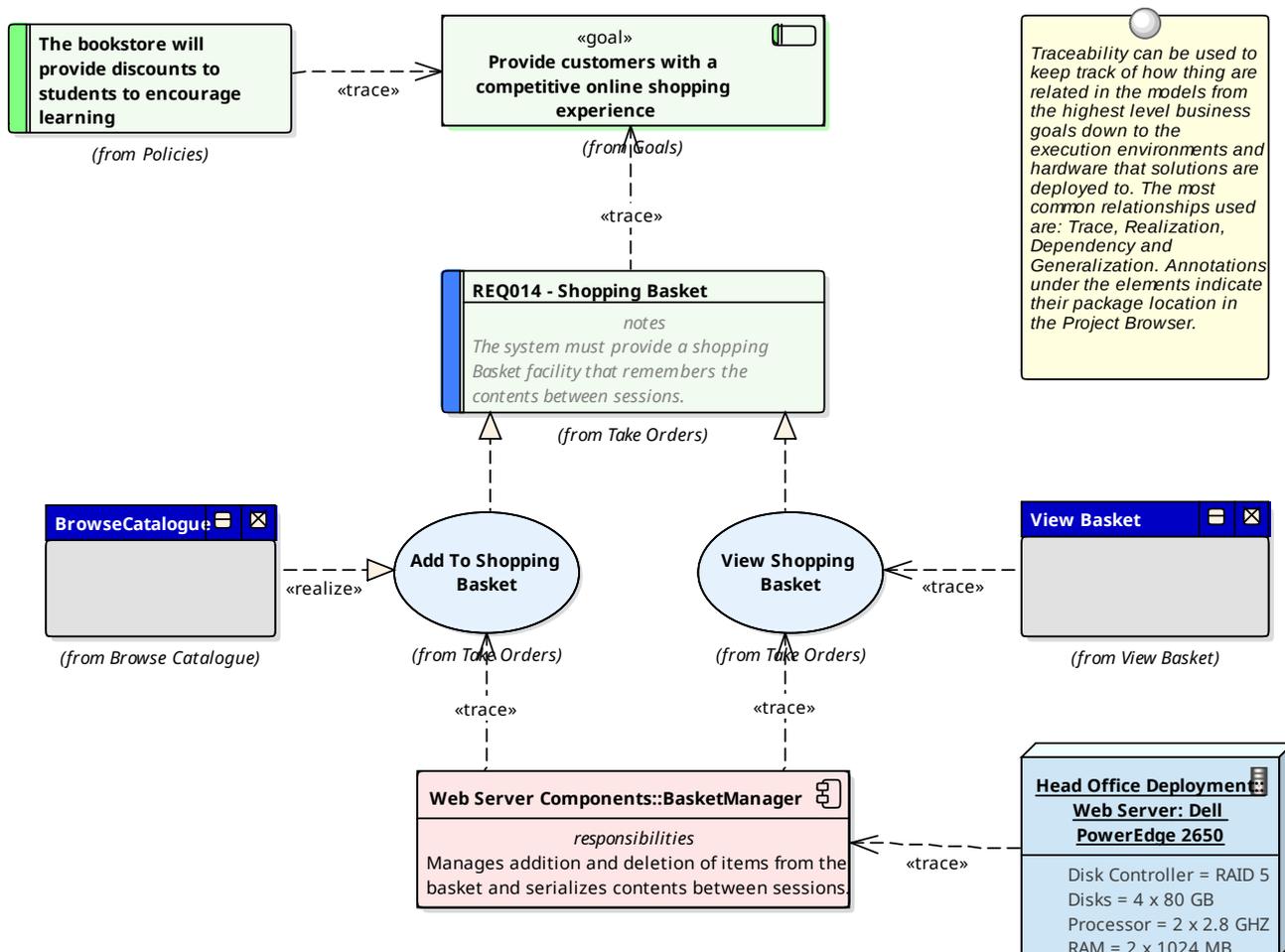
This diagram shows a Bar Chart element depicting element status for all the requirements in a selected package. It provides a useful summary for a requirements manager and is dynamically updated when the status changes and the diagram is reopened. There are a range of other pre-defined charts and user defined charts can also be added.

Tracing Requirements

Most requirements processes mandate that requirements are traced from high level concepts such as Business Drivers, Visions and Goals down to the parts of Components that implement them. For many projects this is an intractable problem because much of the information lives in a set of heterogeneous tools such as word processor documents, spread sheets, diagram tools, corporate presentation tools and more. Some project managers attempt to solve the problem by creating a spread sheet that acts as a register of all the disparate information but the management of this file takes up considerable project resources and is almost impossible to keep up to date. With Enterprise Architect there is the ability to model all of this project information in the one tool and to create easy to maintain and analyzable traces between all the elements from an organization's mission statement right down to the level of programming code if required.

Visualizing Traces in diagrams

Regardless of whether you have entered the project's Requirements using a diagram or using a text based tool like the **Specification Manager** or imported them from another tool, viewing the requirement traces in a diagram gives an easy to understand view of their relationships. The diagrams can be created easily by dragging and dropping elements from the **Project Browser** or automatically by using the **Insert Related Elements** function. This function can be configured and used to draw a graph of elements to any depth and can be restricted to selected element and connector types. It is a powerful productivity tool in a team environment and even modelers with deep knowledge of the domain and the repository are surprised at the connections that are displayed in the diagrams.



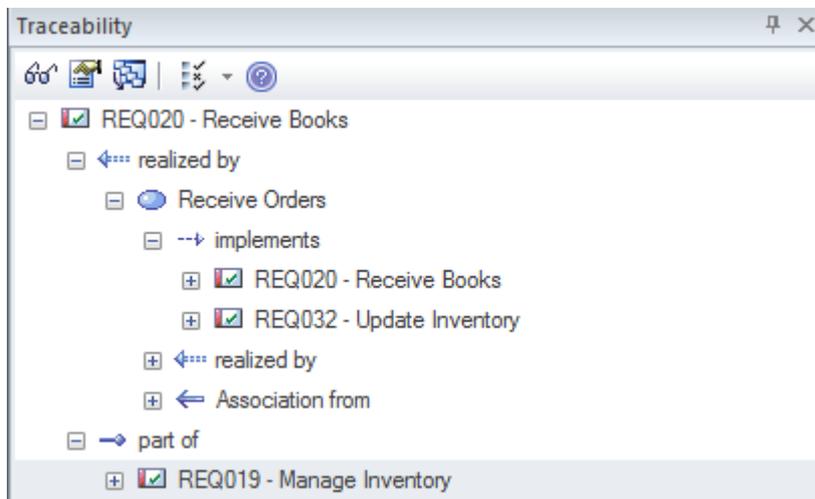
Visualizing Traces using the Relationship Matrix

The Relationship matrix is a powerful tool and provides an alternate way of visualizing the relationship between requirements and other elements or even between different levels or types of requirements. It is quite common for some stakeholders to prefer a spreadsheet like view of the Requirements and their relationships and the **Relationship Matrix** provides an excellent way of presenting the relationships without resorting to a diagram. In Use Case driven requirements methods Use Cases are said to realize one or more Requirements and these relationships can be displayed visually in the relationship matrix. The list of Use Case would appear on one axis of the matrix and the requirements would be listed on the other axis, a marker in the row and column intersection would display if a relationship exists indicating that a particular Use Case realizes a Requirement. Relationships between elements can be created or deleted using the relationship matrix and the matrix can be saved and reopened at any time or saved to a CSV file so it could be opened in a spreadsheet. Documentation can also be created that includes the relationship matrix providing a powerful communication tool for people who do not have access to the model.

Target +	REQ011 - Manage User Accounts	REQ012 - Provide Online Sales	REQ013 - Manage Deliveries	REQ014 - ShoppingBasket	REQ015 - Process Credit Card Payment	REQ016 - Add Users	REQ017 - Remove User	REQ018 - Report on User Account	REQ019 - Manage Inventory	REQ020 - Receive Books	REQ021 - List Stock Levels	REQ022 - Order Books
+ Source												
Add New Titles												
Add To Shopping Basket				↑								
Close Account							↑					
Create Account						↑						
Create Orders												↑
Delete User							↑					

Visualizing Traces using the Traceability Window

While diagrams and the **Relationship Matrix** allow modelers to view traces between requirement elements it is possible that the creators of these views of the repository have deliberately omitted elements from the view. For example a diagram does not need to show all the requirements owned by a particular stakeholder. The **Traceability window** will however present a complete and unabridged view of the relationships between elements. The element relationships will be displayed regardless of the location of the elements in the **Project Browser**.



Visualizing Traces using the Relationships Window

Modelers often choose to hide one or more relationships on a diagram for the purpose of making the diagram simpler to understand or to hide detail. The Relationships window is a useful window to have open as it will display all the relationships that exist between the elements in the diagram indicating whether they are visible or hidden in the diagram.

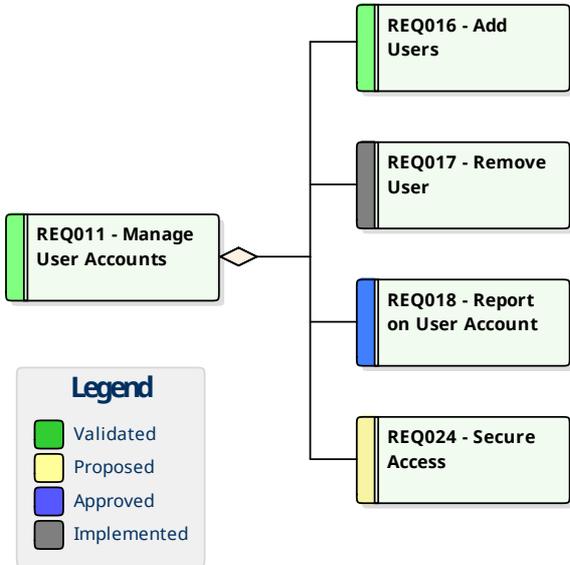
Relationships			
Relationship	Source	Target	View
Abstraction	REQ014 - Shopping Basket	Provide customers with a competitive ...	Visible
Realization	Add To Shopping Basket	REQ014 - Shopping Basket	Visible
Realization	View Shopping Basket	REQ014 - Shopping Basket	Hidden
Aggregation	REQ014 - Shopping Basket	(REQ012 - Provide Online Sales)	

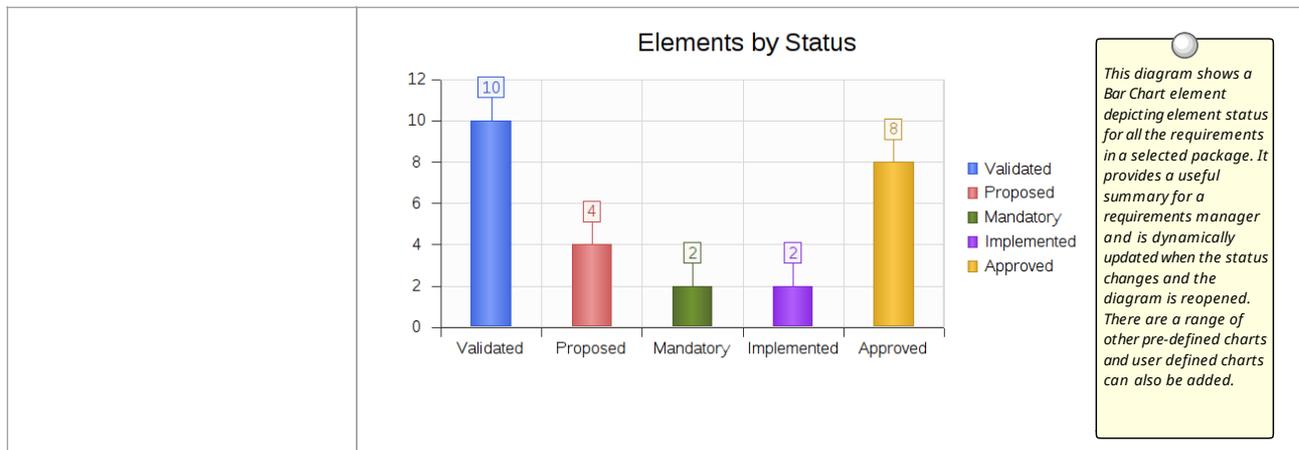
If relationships have been hidden in a diagram they can be made visible by selecting the 'Diagram | Visible Relations... ' option from the main menu.

Tracking Requirements

The status of a requirement is a fundamental indicator of where it is positioned in the requirement's development process. For example requirements that have a status of Proposed indicate that they are not yet ready and available for development work to begin. Enterprise Architect has a variety of tools to allow status to be tracked, analyzed and managed, starting with the fact that each requirement can be assigned a status and the list of status codes are completely configurable. The status is conveniently displayed in list views of the requirements including when using the **Specification Manager**. There are also a set of pre-defined and extensible dashboard charts and graphs that can be used to get a compelling visual representation of the status and other properties of requirements.

Tools for tracking requirements

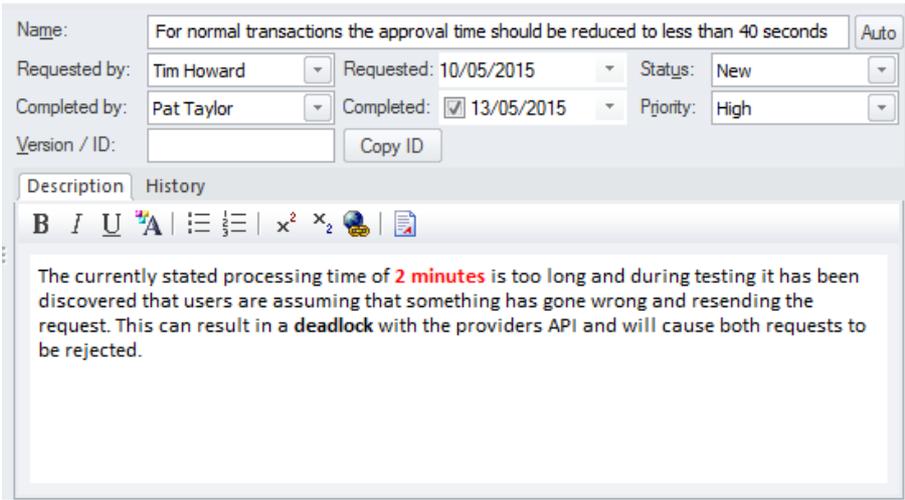
Tool	Description
Status Codes	<p>Status codes are a controlled list of statuses that can be applied to any element including Requirements. Enterprise Architect comes with a pre-defined list of codes but the list can be configured and codes in the list can be changed and deleted and new codes can be added. The status of Requirements can be displayed in a diagram as a color coded band on the side of the element.</p>  <p><i>This diagram show the use of the color code bar indicating the status of the requirement. It is a powerful visual cue that is useful in workshops and printed material to convey the status of a series of requirements in a diagram. The status codes and the colors are completely configurable and a legend can be placed on the diagram to convey their meaning.</i></p>
Dashboards charts and graphs	<p>Dashboard diagrams are an extended diagram type and allow high quality charts and graphs to be created to display repository information in a visually compelling way. Any number of diagrams and charts can be created and the data can be sourced from any level in the repository package hierarchy. Enterprise Architect comes with a toolbox page of pre-configured charts and graphs but new charts can be created based on any information in the repository.</p>

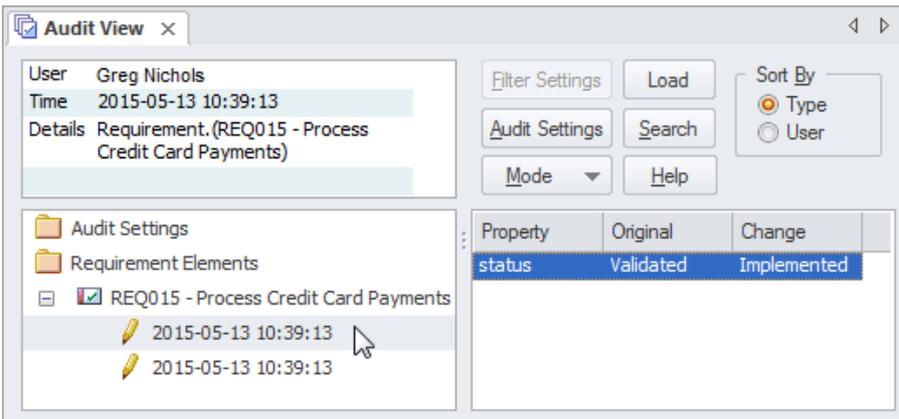


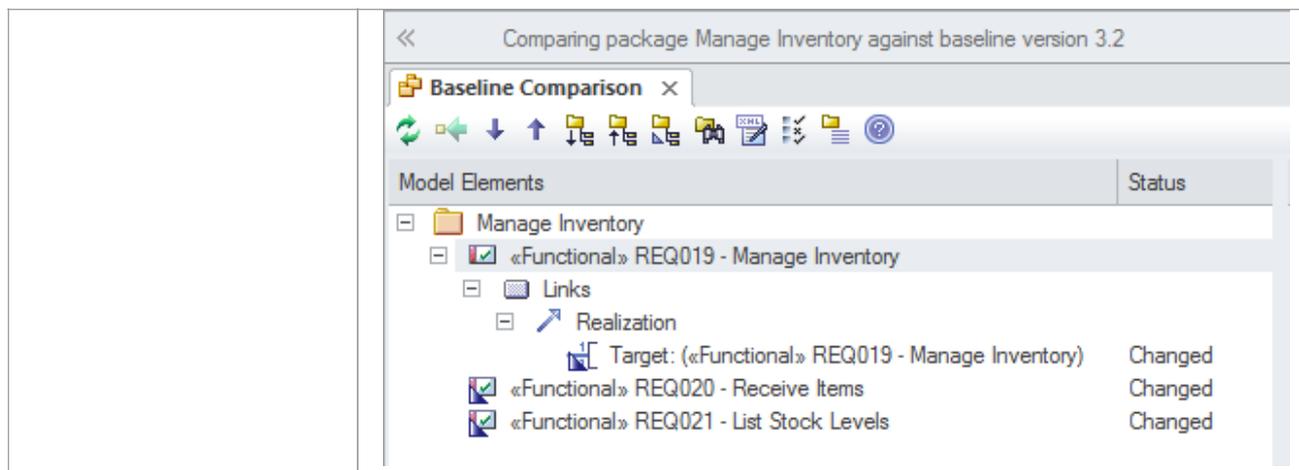
Managing Changing Requirements

It is inevitable that requirements will change during the specification and solution phases of a project and most requirements management processes have some type of mechanisms for embracing these changes. Typically a set of requirements will have been specified and groomed for the solution teams to implement; any subsequent changes are specified as Change Requests. Regardless of the rigor of the process being used inadvertent changes will occur and these need to be managed along with the Change Requests. Enterprise Architect is a sophisticated requirements management platform with a range of tools to assist the requirements manager. Change Requests can be managed in the Element **Maintenance window** which allows the requested change to be recorded and described, along with when and who requested it and who completed the change and when it was done. Inadvertent changes can be discovered and analyzed using a number of tool features including **Auditing**, **Baselines** and **Version Control**, these tools have some overlapping features and can be used in isolation or together. The built-in Security system will also assist in preventing inadvertent changes to models by allowing modelers to intentionally lock Packages and Elements in the model.

Mechanisms for managing changing requirements

Mechanism	Description
Element change task and effort items	<p>Changes to requirements may happen inadvertently but it is more common for there to be an intentional change in response to a wide variety of factors such as Stakeholders revising their needs, the business changing or a problem being poorly understood. Inadvertent changes can be picked up using a number of tools but deliberate changes can be assigned using the Change item which can be recorded against each element. Once the impact of the change has been analyzed Tasks can be created to specify what needs to be done to implement the change and Effort can be assigned using the Requirements Effort item.</p>  <p>The screenshot shows a 'Name' field with the text 'For normal transactions the approval time should be reduced to less than 40 seconds' and an 'Auto' button. Below are dropdown menus for 'Requested by' (Tim Howard), 'Requested' (10/05/2015), 'Status' (New), 'Completed by' (Pat Taylor), 'Completed' (13/05/2015), and 'Priority' (High). There is also a 'Copy ID' button and a 'Version / ID' field. The description field contains the text: 'The currently stated processing time of 2 minutes is too long and during testing it has been discovered that users are assuming that something has gone wrong and resending the request. This can result in a deadlock with the providers API and will cause both requests to be rejected.'</p>
Auditing	<p>Auditing is a built-in tool, that when enabled, automatically records changes to the repository. It has a number of different modes and views and can be configured to assist in the management of Requirements. It can track what was changed in the model, who made the change and when it was made showing the before and after views. So if the text of a Requirement was updated or its status was changed this would be recorded. Auditing functionality overlaps with the Baseline tool but unlike the Baseline tool the changes are being recorded automatically and every</p>

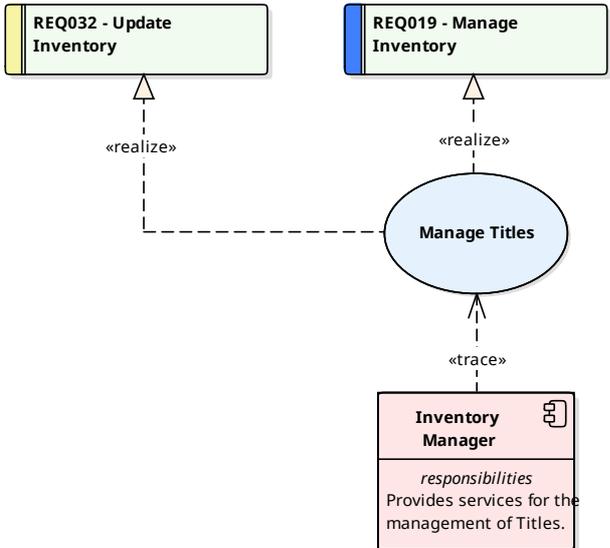
	<p>discreet change is recorded. In contradistinction the Baseline tool will only compare the current model to a Baseline regardless of how many intervening changes had been made. Auditing will not assist with the impact of the changes but just what changes have occurred, once the changes have been established tools such as the Relationship Matrix can be used to determine the impact.</p> 
Version Control	<p>Version Control can be implemented in Enterprise Architect to manage changes and revisions to any Package including Requirements Packages. Once implemented changes to Requirements will be recorded and a requirements analyst will be able to view previous version and roll back to these versions if required. There is some overlap between this tool feature and Auditing and Baselines. The difference between this facility and Auditing is that Auditing simply records the changes but does not allow you to revert to a previous version. The difference between Version Control and Baselines is that a modeler must intentionally create a baseline whereas with Version Control the changes are being recorded automatically in the background. Also with Baselines the intervening changes are not recorded, just the difference between the current requirement and the one captured in the Baseline.</p>
Baselines	<p>Baselines provide a powerful mechanism for managing changes to Requirements. Any number of baselines can be created and when requirement are changed these requirements can be compared to one of the baselines. Baselines are typically created at important milestones in a project such as after a stakeholder meeting or before a development iteration is commenced. When differences are found and these changes were not intended or contravene project management practice the requirements from the baseline can be restored to the current model. Baselines will not help with assessing the impact of a change but once a change has been identified tools such as the Relationship Matrix and element traces can be used to determine the impact of a change.</p>

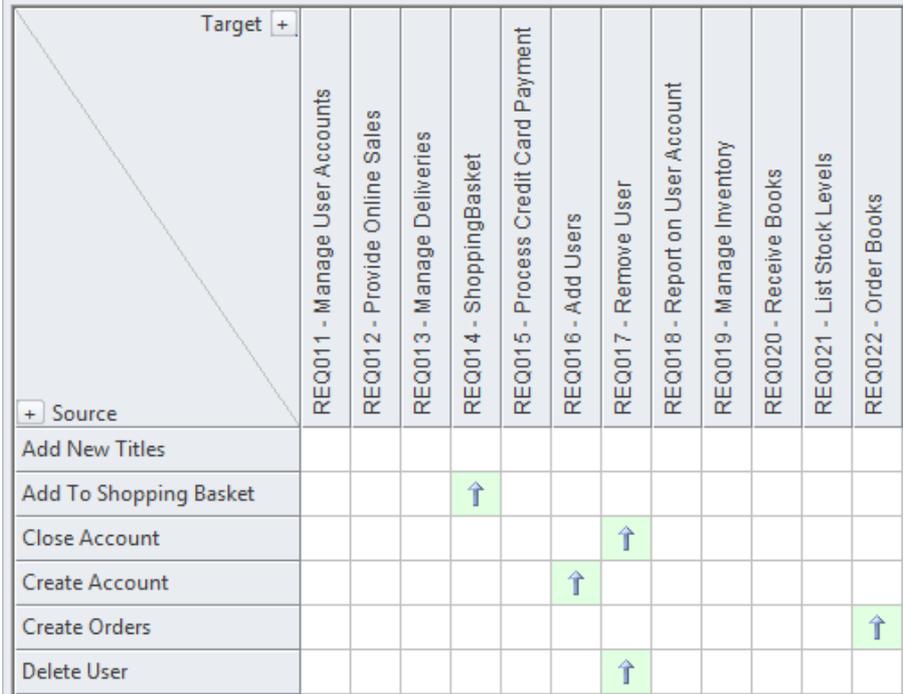
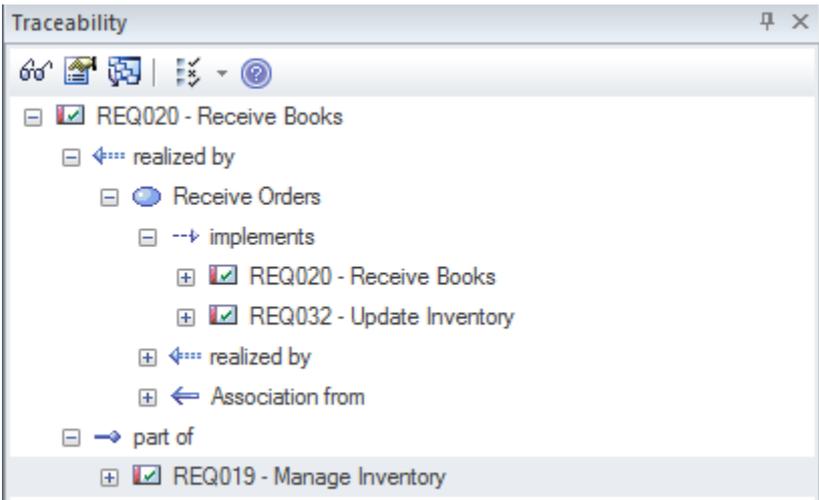


Impact Analysis of Changes

When the development of a system has started and requirements change there will be an impact of the change and the effect will need to be determined, understood and managed. Having traceability established both to up-process elements such as Stakeholders and Business Drivers and down-process elements such as Use Cases, Components, Test Cases and source code operations is critical to determining the impact of the change. Enterprise Architect has a number of facilities that can assist with this including the ability to visualize traces in diagrams, a powerful **Relationship Matrix**, a **Traceability window**, element Change, Task and Effort items that can be used to record impact and what is required to implement it.

Tools to record and analyze the impact of change

Tool	Description
<p>Analysis using requirement traces</p>	<p>The ability to visualize requirements and the way they are connected to other elements is a powerful tool for analyzing the impact of change. Requirements often form a hierarchy and when one requirement is affected it typically has a ripple effect to the requirement's children and being able to visualize this in a diagram or in a hierarchy is very useful. Requirements are also typically traced to up-process and down process elements and a diagram provides a way of viewing and analyzing these connections. The Insert Related Elements function can discover these connections and automatically draw and layout a diagram allowing the modeler to spend their time analyzing the impact.</p>  <p>This diagram shows the expressive power of putting disparate elements onto a diagram to show the traceability between different layers of a system. The traceability can be from the Requirements to the Use Cases that Realize them to the logical Components that will deliver the required functionality and more.</p>
<p>Analysis using a relationship matrix</p>	<p>The Relationship Matrix can be used to visualize the requirements and their connections by placing the Requirement on one axis of the matrix and the connected elements on the other. This method is very useful in workshops when working with people who may not be familiar with modeling languages like UML or who work better with spread sheet types of views. Any number of matrices can be created and their specification can be stored so they can easily be recalled.</p>

	
<p>Analysis using the traceability window</p>	<p>The Traceability window is a powerful window that shows the hierarchy of elements in the Repository. It is particularly useful because it unconditionally shows how elements are related to each other. Other views of the repository could be configured just to display particular elements for the purpose of communicating an idea whereas the Traceability window will display all relationship that an element participates in which makes it particularly useful for analyzing the impact of change.</p> 

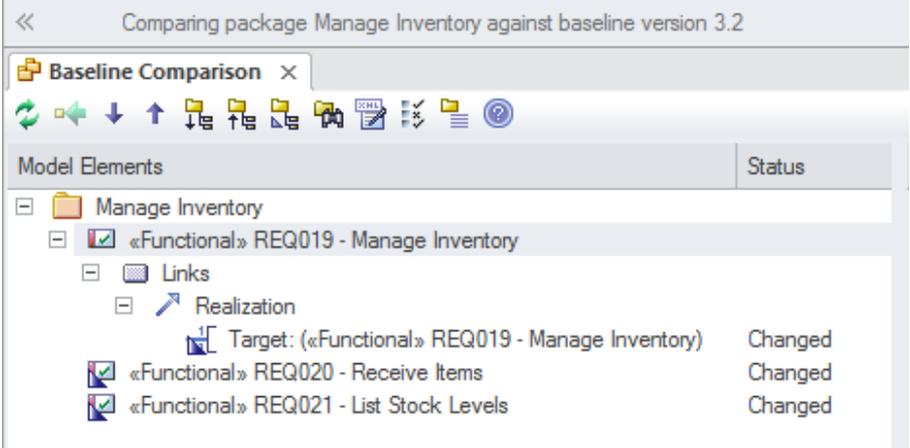
Requirement Volatility

There are ever increasing market place pressures to release products and systems as early as possible putting stress on project teams to develop, test and deploy products in shorter and shorter periods of time. The requirements processes have changed significantly in recent years to ensure that stable, correct and well-articulated specifications are provided to architects, designers and developers when they need them. There has been a move to iterative and incremental processes and this necessitates providing a set of requirements for an iteration which are stable. The churning of requirements is often an indicator that a problem is not clearly understood, that stakeholders have not compromised and there are unresolved political issues, the scope is not defined or the business itself is in fluctuation. Enterprise Architect has a number of mechanisms that can be used to assist with this problem. Enterprise Architect does not have a built-in property for requirement volatility (stability) but using the general purpose UML extension mechanism of **Tagged Values** a tag could be created to record this property.

- Note: Internal requirements do have a stability property but external requirements do not.

Mechanisms for managing requirement volatility

Mechanism	Description
Volatility as a Tagged Value	<p>Enterprise Architect provides a series of properties for requirements but additional properties can be created to record other properties such as a requirement's volatility or the source of the Requirement. This is achieved using the UML Tagged Value mechanism which allows any element including requirements to have one or more tags applied representing some property which can be assigned a value. Enterprise Architect has extended this mechanism and allows the modeler to create a list of values that can be chosen from a drop down list using the Predefined Structured Tagged Values. This allows a team to define their own list of volatility values such as extreme, high, medium low, minimal.</p>  <p>The screenshot shows a requirement element titled "REQ021 - List Stock Levels". It has a "tags" property set to "Volatility = Medium". Below the tags, there is a "notes" property with the text: "A facility will exist to list current stock levels and to manually update stock quantities if physical checking reveals inconsistencies."</p>
Using Baselines	<p>The Baseline facility is a powerful tool that enables a user to take a snapshot of a model or more typically a model fragment and then as the model is developed to compare the new version of the model to the baseline thus identifying anything that has changed since the baseline was taken. Baselines have general applicability but are particularly useful with requirements management where requirements are often said to be signed-off or frozen and any alterations to them must be registered as a change. The Baseline tool has a Compare utility that conveniently lists changes between the current model and the baseline.</p>

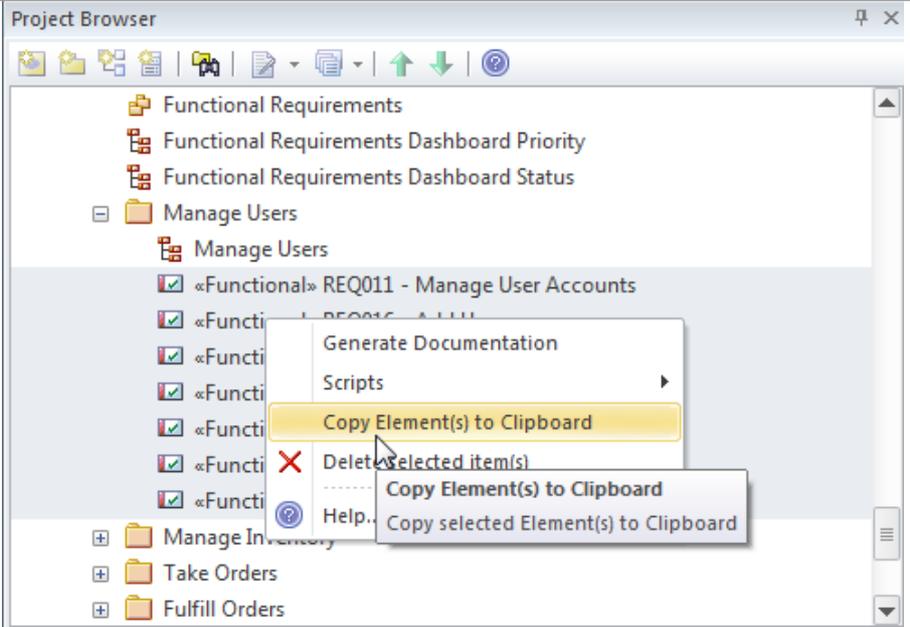
	
Searches for churning requirements	<p>Enterprise Architect has a sophisticated search facility that allows a user to search across a selected Package or the entire repository and locate elements that meet fined grained criteria. This can be used to locate requirements that have not changed by searching for a change in the modification date before a specified date, thus providing a list of stable requirements. Alternatively if volatility has been set using a Tagged Value, all elements with a specified volatility could be located. The search facility returns a list of elements that can be located in the Project Browser and the search can be used as the basis of a Model View that could be used to view volatile or alternatively non-volatile requirements.</p>

Requirements Reuse

The concept of reusing artifacts of a system development process has been written about in many papers and text books but has traditionally been confined to software components. In more recent years the notion of reusing specifications including requirements has started to get traction. The reuse is particularly important where organizations create a family of products with similar features or where there is a community of users within an industry or domain. Other types of requirements such as security and regulatory requirements will typically apply to a number of projects. Business Rules and Stakeholders Concerns will also typically apply across many projects and are best catalogued outside individual projects structures. Enterprise Architect provides a number of sophisticated mechanisms for managing the reuse of elements across projects including structuring the repository for reuse, importing requirements from other sources and a **Reusable Asset Service**.

Mechanism for requirements reuse

Mechanism	Description
Structuring the repository for requirements reuse	When you set up a repository you have the choice of structuring it for a single project or for multiple projects, which in turn could be organized by a number of programs of work. Enterprise Architect gives the modeler complete control on how the repository is structured allowing Packages to be set up above the level of projects where some requirements such as Business, Regulatory and Architectural Requirements can be added.
Creating a base model	When you create a new repository in Enterprise Architect a modeler has the option of creating a blank model, using the Model Wizard to help setup a repository structure or using a base model which acts as a template for the new model. The base model is a good place to store reusable assets such as Business, Regulatory and Architectural Requirements and also Policies and Business Rules. A base model can be created using the 'File Save Project As...' option in the main menu.
Importing requirements from other models	It is quite common to have a number of Enterprise Architect Repositories in an organization and it is very easy to copy and reuse requirements (or any other elements) from one model to another. This can be achieved by simply copying a selection of requirements or an entire Package from one repository to another or even from one project to another in the same repository. Enterprise Architect works like any other windows program, simply copying the selection to the clipboard and then allowing it to be pasted in another location in the same model or another open repository.

	
Using the Reusable Asset Service	<p>The Reusable Asset Service (RAS) is particularly useful for distributed teams and provides a simple and convenient mechanism for modelers to distribute or download reusable model structures and elements such as Requirements through a shared repository, accessible via a Cloud Service connection. Enterprise or organizational level Requirements could be stored in the RAS and different teams could incorporate them into their models, governance of the assets would typically be managed by the owner of the asset (register) at the Reusable Asset Service level.</p>

Requirements Documentation

A number of documents are commonly produced as part of the Requirements Engineering discipline, such as the Software (System) Requirements Specification and Use Case Report. These can be generated automatically from a requirements model using built-in templates. In addition a wide range of other documents can be produced using built-in or customized templates. The documentation facility in Enterprise Architect is highly configurable and many reports can be produced using the template system, but for more complex reports there is a facility called **Virtual Documents** that allows a publisher to model the structure of the document and to cherry pick content from anywhere in the repository, applying different templates to each section of the document. There is also a wide variety of options that can be applied at the template or document generation level, and the **Scripting** engine can be used to inject content into a document or to produce the entire report.

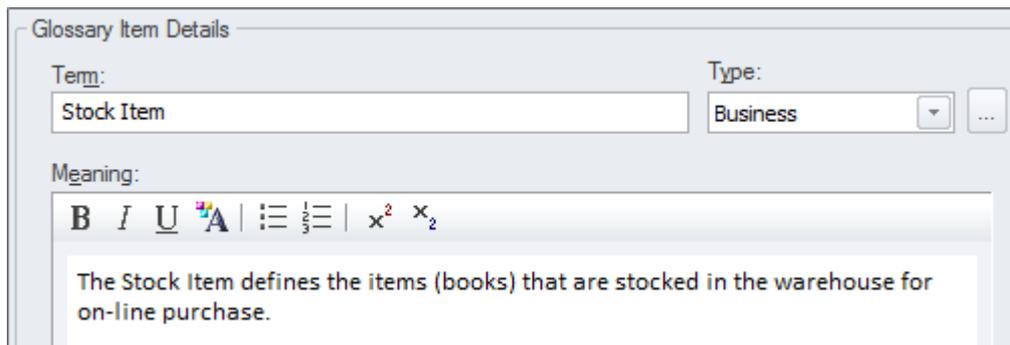
Requirement Report - Details

Fulfill Orders

Version 1.0 • Proposed

Project Glossary

A **Project Glossary** lists and defines the terms that are important for a project or program of work. The Project Glossary can be generated as an isolated document or it can be included as a section in one or more other documents. It provides a single point of truth for the important project terms and their meanings and when new documentation is generated the terms will automatically be updated. The Glossary can be generated to a Docx or PDF format or to Html which could be included in a project or organization level web site. The Glossary allows the modeler to categorize the terms into user defined Types and these can have styles applied when they are generated in documentation.



Glossary Item Details

Term: Type: ...

Meaning:

B *I* U **A** | | x^2 x_2

The Stock Item defines the items (books) that are stocked in the warehouse for on-line purchase.

Software Requirement Specification

This document describes the Requirements of the system and its behavior under defined conditions and the constraints that it must operate under; it will typically be read by a variety of stakeholders. There is a built-in Requirements template that can be used for this purpose and the modeler is free to create a new template which could be based on this or created from a blank template. When the document is comprised of content from a variety of locations in the project browser it would be most expedient to use the **Virtual Documents** facility which allows the user to create a model of the document (similar to a Master document in a Word Processor) which is comprised of a number of sections called model documents. These can contain content picked from anywhere in the project browser.

Software Requirements Specification

Online Bookstore

Version 1.0 • Proposed

Use Case Report

The creation of Use Case documentation has traditionally been a manual process and with the documents in many projects running into hundreds of pages their production consumes valuable project resources. These hand-crafted documents become difficult to maintain and remain isolated from other parts of the project such as Requirements, Business Rules and solution Components. Enterprise Architect has a powerful tool called the **Scenario Builder** that allows the modeler to specify Use Cases and Scenarios inside the model and these can be automatically generated to high quality documentation using built-in templates. There are two built-in templates that can be used for generating a Use Case report: one documents the Use Case at a summary level and the other at a detailed level.



Example content from a Use Case Report

The detailed Use Case report will list all the details of the Use Case and the detailed steps including Basic Paths, Alternate and Exception Scenarios. Other information including Internal Requirements and Pre and Post Conditions and other Constraints will also be included in the report. If a Behavioral diagram such as an Activity diagram has been automatically created this diagram will also be displayed in the report.

 Alternate. List Stock Levels by Publisher

The List Stock Levels by Publisher allows a user to obtain stock level information for a selected publisher. The Stock Control Manager and Storeroom Worker need this information to plan logistics and to ensure that stock remains at adequate levels to service incoming requests. There is also the need to predict the date that the stock items will fall below an acceptable level

Page 3 of 4

Use Case Details

19 May, 2015

SCENARIOS

based on purchase cycles and promotional periods.

1. User selects "List Stock Levels by Publisher"

Uses:

2. System returns a list of publishers to select from

Uses:

3. User Selects a publisher

Uses:

4. System returns a listing of titles and quantity in stock for the publisher

Uses:

Data Dictionary

Many processes specify the creation of a Data Dictionary that acts as a reference for all the information that will be consumed, stored or created by the system. The Data Dictionary can be created in Enterprise Architect using the **UML Class** diagram and Classes can be defined to represent the important elements of the domain including Attributes and Data Types. Enterprise Architect can generate high quality documentation from the Class diagram and data items and their descriptions can be listed and the Attributes of these Classes can also be detailed with Data Types and Multiplicities. Enterprise Architect provides built-in templates that allow the Data Dictionary to be created automatically and these templates can be edited or new ones created.

Account

Class in package 'Domain Model'

The Account defines the user details that are stored as clients. It provides all the information required for billing and product delivery and other information can be related to this record such as preferences and alerts. it is self-managed in the sense that clients are able to keep their own information up-to-date through a web interface ensuring that the system has the latest information.

Account
Version 1.0 Phase 1.0 Proposed
Benjamin Hutton created on 17/03/2005. Last modified 15/05/2015

INCOMING STRUCTURAL RELATIONSHIPS

Page 4 of 12

Data Modeling Report

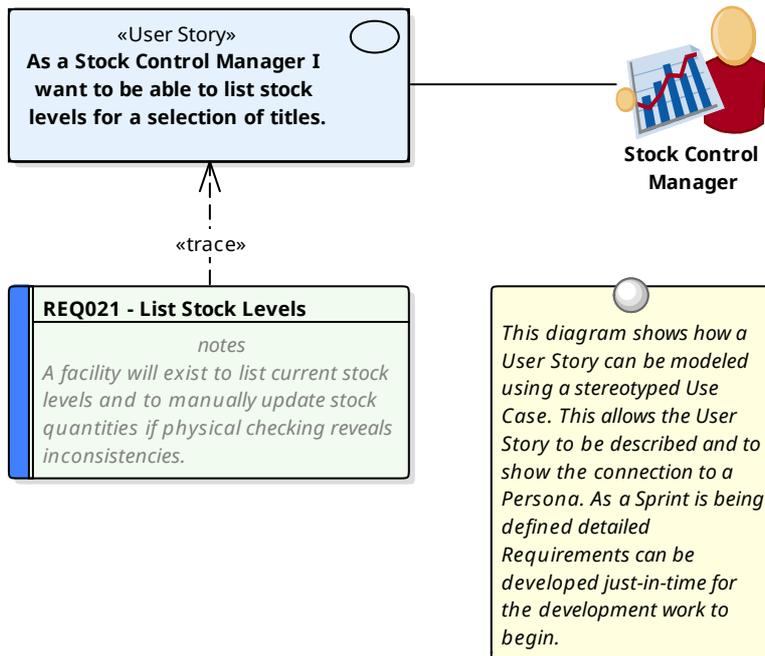
15 May, 2015

INCOMING STRUCTURAL RELATIONSHIPS

- ⇒ Collaboration from «control» View Account Details to Account
[Name is getAccountDetails(). Direction is 'Source -> Destination'.]
- ⇒ Collaboration from «control» Delete User to Account
[Name is retrieveAccountDetails(). Direction is 'Source -> Destination'.]
- ⇒ Collaboration from «control» Create New Account to Account
[Name is submitNewAccountDetails(). Direction is 'Source -> Destination'.]

Requirement Processes and Standards

Enterprise Architect is an open platform that supports any requirements engineering process. The tool has a rich feature set and is highly configurable and its flexible design means that what ever method is being used the modeler will find features to help. So whether the team is using formal Requirements, Use Cases, User Stories, Story-Boards in any combination Enterprise Architect can be used to develop, manage and document the requirements. The implementation of the UML extension mechanisms means that any type of requirement can be created and managed using built-in types or by using Stereotyped elements and **Tagged Values**.

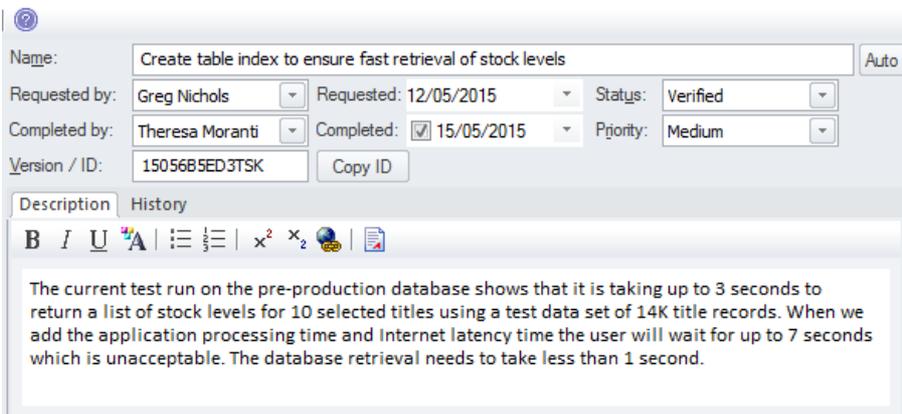


Agile Requirements Processes

There are a variety of Agile methods that have become prevalent for developing particular types of systems and the term Agile has come to encompass a group of software development methods that are iterative and focus on early development and delivery using customer and developer collaboration. Enterprise Architect has been built from the ground up as a flexible modeling platform that supports any software development methodology and has a wide range of tools and features that support Agile processes.

Support for Agile Methods

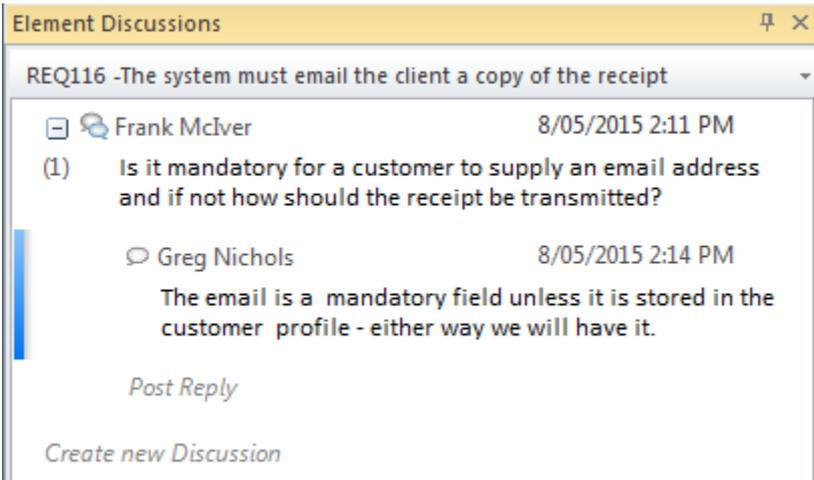
Method	Support
Scrum	<p>Scrum is a project management approach for managing Agile projects that is typically used in conjunction with other agile methods such as XP. The goal of Scrum is to deliver software releases in iterations, which provide the highest demonstrable business value. Enterprise Architect supports User Stories, textual requirements and product releases. Sprint backlogs can be managed using tagged values and Model Views. Estimates of effort can be recorded for Requirements in each of the backlogs and refined as the items are promoted towards the sprint using the element Effort item. A built in Gantt chart will automatically display the schedule for sprints and a series of built-in and customizable dashboards can show the progress of a sprint. The Product Owner, Scrum Master and Team Member roles can all be supported and the tool provides a cohesive platform for collaboration and requirements management.</p> <p>The diagram shows a User Story box with the text: «User Story» As a Stock Control Manager I want to be able to list stock levels for a selection of titles. This box is connected to a Persona icon labeled 'Stock Control Manager'. Below the User Story box is a Requirement box labeled 'REQ021 - List Stock Levels' with a 'notes' section containing the text: 'A facility will exist to list current stock levels and to manually update stock quantities if physical checking reveals inconsistencies.' A dashed arrow labeled '<trace>' points from the Requirement box to the User Story box. A callout box on the right contains the text: 'This diagram shows how a User Story can be modeled using a stereotyped Use Case. This allows the User Story to be described and to show the connection to a Persona. As a Sprint is being defined detailed Requirements can be developed just-in-time for the development work to begin.'</p>
Extreme Programming (XP)	<p>Extreme Programming relies on the User Story (User Card) to express Requirements and develops plans for iterations and releases. Enterprise Architect supports XP by allowing User Stories to be modeled using a stereotyped Use Cases. Iterations are supported by the Phase property built into every element, and a Gantt chart can be used to automatically display scheduled iterations and releases. There is support for developer tasks which can be created as element maintenance tasks for each User Story</p>

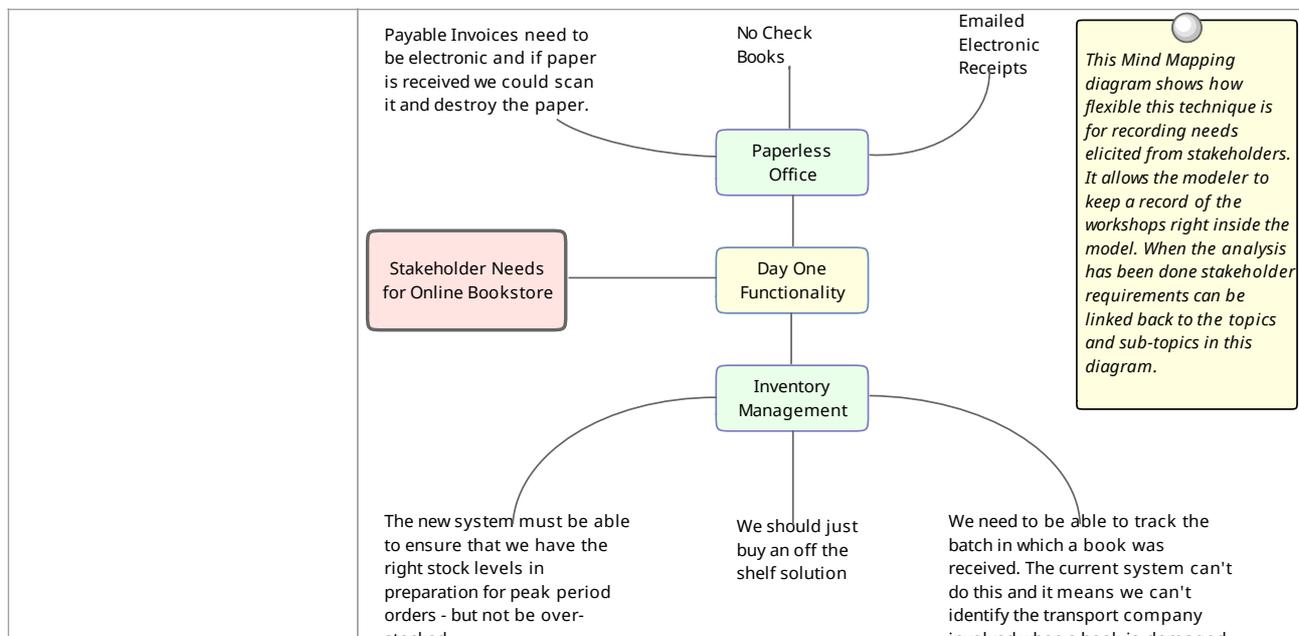
	<p>and can include status, priority, requested and completion dates, history and more. The Customer, Coach, Programmer and Tracker roles can all be supported and the tool provides a cohesive platform for collaboration including visual inspections of code and design and automatic documentation generation.</p> 
<p>Feature Driven Development (FDD)</p>	<p>Feature-driven development (FDD) is an iterative and incremental process that uses a feature to drive iterations and development. Domain models are created early in collaboration with subject matter experts, subject areas are defined and these are broken down into Feature Sets and these in turn into atomic features. A Feature is a piece of system functionality that is valued by the client and is used to drive design, development and implementation. During design a Feature or small Feature Set are assigned to Classes and to programmers and the Features are used to monitor progress within an iteration. Enterprise Architect has support for this method with the Feature element that is part of the Requirements Toolbox, Subject Areas and Feature Sets can be modeled as Packages, allowing a hierarchy to be created that can be displayed as a diagram, list, using the Relationship Matrix or in the Traceability window.</p> 

Business Analysis Body of Knowledge (BABOK)

The International Institute of Business Analysis publishes the Business Analysis Body of Knowledge (BABOK) Guide which contains descriptive and prescriptive information about how to perform the activities and tasks of a business analyst in a number of areas of knowledge. The guide is produced and reviewed by a large number of people around the world and acts as a guide for business analysis including requirements engineering. The guide defines a number of types of requirements and these can all be implemented in Enterprise Architect by defining the types in the requirements section of the 'General Types' dialog.

Support for Babok Knowledge Areas

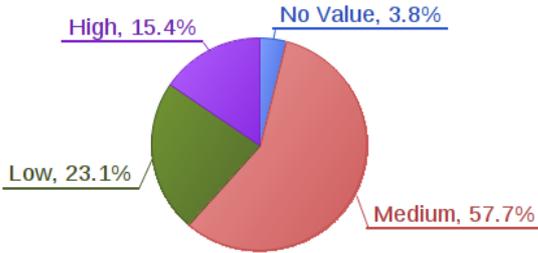
Knowledge Area	Description
Business Analysis Planning and Monitoring	<p>The Business Analysis Planning and Monitoring knowledge area is concerned with planning the approach to the analysis effort, the engagement with stakeholder, the governance of the activities and the how the Requirements and other information discovered during the process is managed and maintained. Enterprise Architect allows a team of analysts to define the approach that will be taken in the tool by defining processes, stakeholders and the governance mechanisms that will be used including the information that is collected.</p>
Elicitation and Collaboration	<p>The Elicitation and Collaboration knowledge area is concerned with the discovery and conformation of information obtained from stakeholders and a variety of other sources. The collaboration with stakeholders during the entire requirements life cycle is critical to the success of the change or endeavor. Enterprise Architect has a number of tools that can assist with the planning the elicitation including a Project Calendar, Gantt chart and a number of team collaboration tools such as Model Mail, Element Discussions and Slide Shows.</p>  <p>Information elicited during stakeholder workshops can be recorded using a MindMapping diagram and, once it has been analyzed, reports can be generated to a variety of formats for confirmation with stakeholders. Photos taken during user observations can be included in diagrams, creating a rich visual presentation.</p>



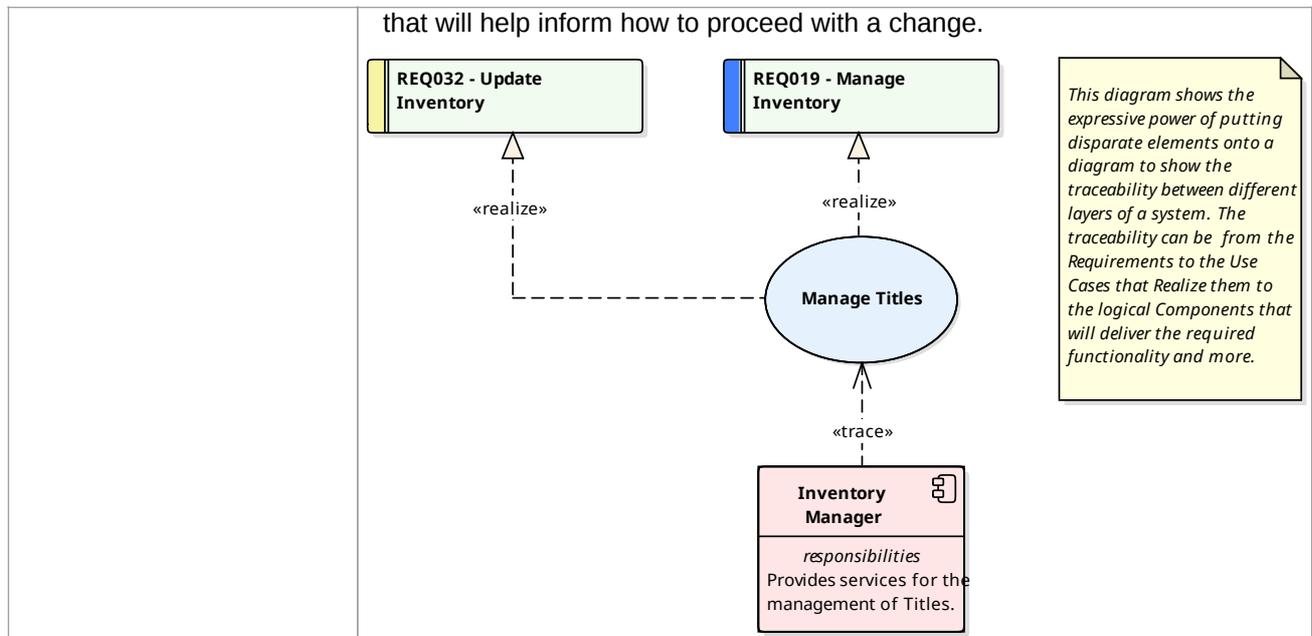
Requirements Life Cycle Management

The Requirements Life Cycle Management knowledge area is concerned with the management of requirements and ensuring that the requirements can be related to the solution designs and ultimately to the solution components during the entire life cycle of the solution from inception through to retirement. The management of requirements involves requirements governance, tracing the requirements, maintaining and prioritizing them and managing change and assessment. Enterprise Architect has sophisticated tools that enable requirements to be traced, maintained and prioritized. These include the Relationships Matrix, which can be configured to display sets of traces between source and target Packages; the **Traceability window**, which shows how elements are connected in the repository; and the Relationships window, which displays the relationships between elements in a diagram. The diagram is a powerful way to demonstrate and make visual underlying relationships including the connection between stakeholders and the elicited information.

Target +	REQ011 - Manage User Accounts	REQ012 - Provide Online Sales	REQ013 - Manage Deliveries	REQ014 - ShoppingBasket	REQ015 - Process Credit Card Payment	REQ016 - Add Users	REQ017 - Remove User	REQ018 - Report on User Account	REQ019 - Manage Inventory	REQ020 - Receive Books	REQ021 - List Stock Levels	REQ022 - Order Books
+ Source												
Add New Titles												
Add To Shopping Basket				↑								
Close Account							↑					
Create Account						↑						
Create Orders												↑
Delete User							↑					

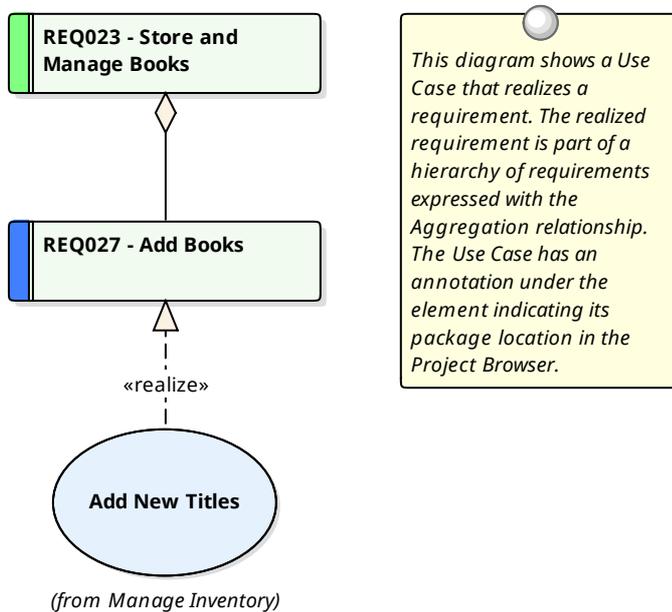
	<p>The requirements governance process could also be modeled using UML Activity diagrams or BPMN diagrams and this process could be reused across a range of projects and changes. There is also a Dashboard diagram that allows Requirements properties to be categorized and displayed in a series of charts and graphs which assist in the collaboration and management of properties like Status and Priority and to communicate this information to stakeholders.</p> <p style="text-align: center;">Elements by Priority</p>  <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p><i>This diagram shows a Pie Chart element depicting element priority for all the requirements in a selected package. It provides a useful summary for a requirements manager and is dynamically updated when the Priority changes and the diagram is reopened. There are a range of other pre-defined charts and user defined charts can also be added. A filter has been added to exclude all elements other than Requirements.</i></p> </div>
<p>Strategy Analysis</p>	<p>The Strategy Analysis knowledge area is concerned with the discovery and articulation of a business need and the description of the future state and various transition (intermediate) states that will be used to move from the current state to the defined future state. It involves strategic thinking and the determination of a number of possible solutions that will create value for the organization and its stakeholders. Enterprise Architect has tools to describe the possible solutions and the value and outcomes they will produce and to model the current and future states and what is needed to transition from one state to another.</p>
<p>Requirements Analysis and Design Definition</p>	<p>The Requirements Analysis and Design Definition knowledge area is concerned with how to organize and structure the information that is discovered during elicitation and how to record these as a set of coherent Requirements. It is also concerned with the prioritization of the requirements and finding solution options that can be evaluated against the potential benefit they will bring to the business. The articulation and evaluation of solution options is done in conjunction with the implementation teams but ultimately it is the business analyst that is responsible for conveying the options and their business value and recommending a solution to the stakeholders. Enterprise Architect is a fully featured requirements management tool and allows requirements to be created, maintained and prioritized. The Specification Manager can be used to input, maintain and visualize requirements in a view that resembles working in a word processor or spreadsheet.</p>

	<div style="background-color: #FFD700; padding: 2px; border: 1px solid black; margin-bottom: 10px;">Item</div> <h2 style="margin: 0;">1 REQ019 - Manage Inventory</h2> <p>The system MUST include a complete inventory management facility to store and track stock of books for the on-line bookstore.</p> <h3 style="margin: 0;">1.1 REQ122 - Inventory Reports</h3> <p>Inventory reports are required that detail the available stock for each item including back orders. Future stock level reports should be able to predict the quantity of stock at a specified future date.</p> <h3 style="margin: 0;">1.2 REQ023 - Store and Manage Books</h3> <p>A book storage and management facility will be required.</p> <h4 style="margin: 0;">1.2.1 REQ022 - Order Books</h4> <p>A book order facility will be required to allow on-line ordering from major stockist's.</p> <h4 style="margin: 0;">1.2.2 REQ021 - List Stock Levels</h4> <p>A facility will exist to list current stock levels and to manually update stock quantities if physical checking reveals inconsistencies.</p>
<p>Solution Evaluation</p>	<p>The Solution Evaluation knowledge area is concerned with the evaluation of an implemented solution and the assessment of whether it meets the business needs. The implementation is typically a partial solution or proof of concept and the analyst must work to determine if there are obstacles or removable constraints that are limiting the value that can be delivered by the solution. Enterprise Architect is a full life cycle tool that not only is a requirements management platform but also allows the modeler to describe the solution in detail including constraints and design decisions and rationales.</p> <div style="display: flex; justify-content: space-between; align-items: flex-start; margin-top: 20px;"> <div style="width: 45%;"> <div style="border: 1px solid black; background-color: #E6E6FA; padding: 5px; margin-bottom: 10px;"> <p>«Assumption» {The database search procedures will perform to requirements when the business commences selling on the international market and the number of transactions has increased by an order of magnitude.}</p> </div> <div style="text-align: center;">  </div> <div style="border: 1px solid black; background-color: #FFC0CB; padding: 5px; margin-bottom: 10px;"> <p>BookStore Database</p> </div> </div> <div style="width: 45%; border: 1px solid black; background-color: #FFFFE0; padding: 5px;"> <p><i>This diagram shows the use of an Assumption (constraint) that can be connected to an element to show some assumed condition. Constraints can be written informally in text or more formally in a language such as the Object Constraint Language (OCL).</i></p> </div> </div> <p>The solution components can be traced back to the functional, stakeholder and ultimately business requirements allowing the analyst to visualize the solution in the context of the problem and to record issues and decisions</p>



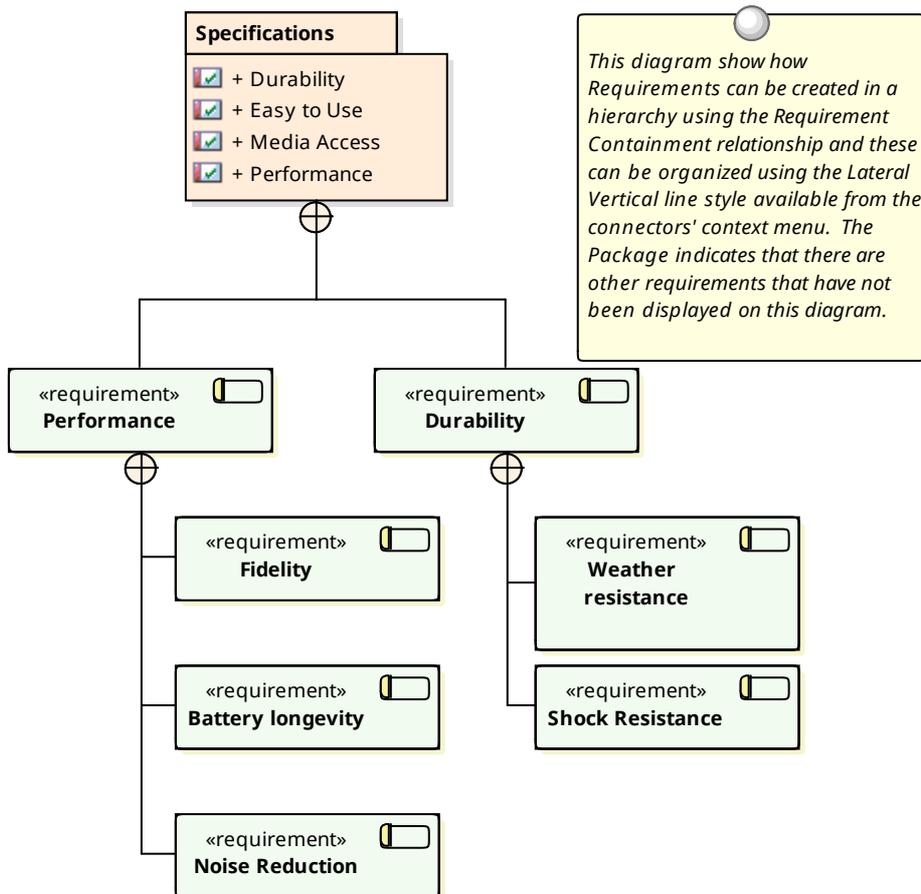
UML Requirements

The Unified Modeling Language does not specify an element for modeling Requirements other than the Use Case, which it states can be used to describe system usages. In very early versions of UML a Requirement was defined as a stereotyped comment, but this was later made obsolete. Enterprise Architect has, since very early versions, filled this gap in the UML specification by extending the language to include a Requirement element that allows this important concept to be modeled textually and graphically in diagrams and other views. Enterprise Architect has rich support for modeling Use Cases in compliance with the specification but allows the requirements analyst to specify other Requirements using the flexible Requirements element. When used together the Requirement element and the Use Case equip the analyst with a powerful palette to describe the Requirements for any system at any level. The Requirement element can be used to describe Business Drivers, Business Goals and Stakeholder Requirements, and these can be linked to Use Cases with the Realize connector. The Use Cases can be augmented with detailed Functional, Architectural and Implementation Requirements suitable for the implementation teams.



SysML Requirements

The Systems Modeling Language (SysML) specifies a rich set of modeling constructs to represent text-based requirements as elements which can be related to other elements in a model. Model Based Engineering has become important in recent years as the complexity of systems has increased and the engineering profession has recognized the need for more sophisticated mechanisms for describing and developing these complex systems. Enterprise Architect has full support for the SysML and the modeling of Requirements including support for a range of ways of presenting Requirements such as: Diagrams, Lists, Matrices, Hierarchies and documentation. There are even dashboard diagrams that present charts and graphs that summarize requirements properties such as Status, Priority and Difficulty in a visually compelling way.



MDG Link for DOORS

Welcome to the Model Driven Generator (MDG) Link™ for DOORS, which provides support for linking an Enterprise Architect model to an IBM® Rational® DOORS® requirements package.

Features

The MDG Link for DOORS is useful when you need to perform the management of Requirements external to your Model Driven Development. Using this Add-in you can interchange requirements defined within IBM DOORS with the traceable Requirements used within Enterprise Architect's Requirements Management features.

The MDG Link for DOORS supports these features:

Importing:

- Objects from IBM® Rational® DOORS®
- Object attributes from IBM® Rational® DOORS®
- Links between objects
- External links (attachments)
- **Discussions**

Exporting:

- Requirements and Use Cases
- Element properties and **Tagged Values**
- Connectors between elements
- Attachments

Synchronization:

- Of IBM® Rational® DOORS® requirements and Enterprise Architect elements through import and export (or vice-versa)

Obtaining the MDG Link for DOORS

The Enterprise Architect Ultimate edition contains the MDG Link for DOORS as an integrated component.

For all Enterprise Architect editions other than Ultimate, you can purchase the MDG Link for DOORS separately and download the installer from the Sparx Systems website. The product pages provide:

- A product overview
- A video demonstration of the product
- Pricing and purchasing information
- System requirements for the product
- A trial version of the product to explore for 30 days

When you purchase the product, you will receive download and installation instructions by email.

Enable MDG Link for DOORS

When you have installed the MDG Link for DOORS, and before you can access its facilities, you must enable the product for use.

1. In Enterprise Architect, select the 'Extensions | Manage **Add-Ins**' menu option.
2. Against the 'DoorsEASync' entry, select the 'Load on Startup' checkbox.
3. Click on the **OK button**.

Getting Started

After installing and/or enabling the MDG Link for DOORS, you can access the facilities of the technology in Enterprise Architect.

Access

Project Browser | Right-click on target Package | Extensions | Doors Extension or

Project Browser | Click on target Package | (Main Menu) Extensions | Doors Extension

Tasks

There are three principal tasks that you can perform through the MDG Link for DOORS.

Task	Description
Create a Module Link	<p>On the 'Doors Extension' menu you can select the 'Add/Edit Module' option to create a link between the selected Sparx Systems Enterprise Architect Package and an existing IBM® Rational® DOORS® module. You can also redirect a link to a different module.</p> <p>The link enables you to exchange requirements data between DOORS and Enterprise Architect, importing data from DOORS to Enterprise Architect, and exporting data from Enterprise Architect to DOORS.</p>
Export data to a DOORS module	<p>On the 'Doors Extension' menu you can select the 'Export to Doors' option to transfer requirements from a selected Enterprise Architect Package to a linked DOORS module.</p> <p>As part of this process you can create an export profile to define what the export should operate on.</p>
Import data from a DOORS module	<p>On the 'Doors Extension' menu you can select the 'Import from Doors' option to transfer requirements into a selected Enterprise Architect Package from a linked DOORS module.</p> <p>As part of this process you can create an import profile to define what the import should operate on.</p>

Create a Link to a DOORS Module

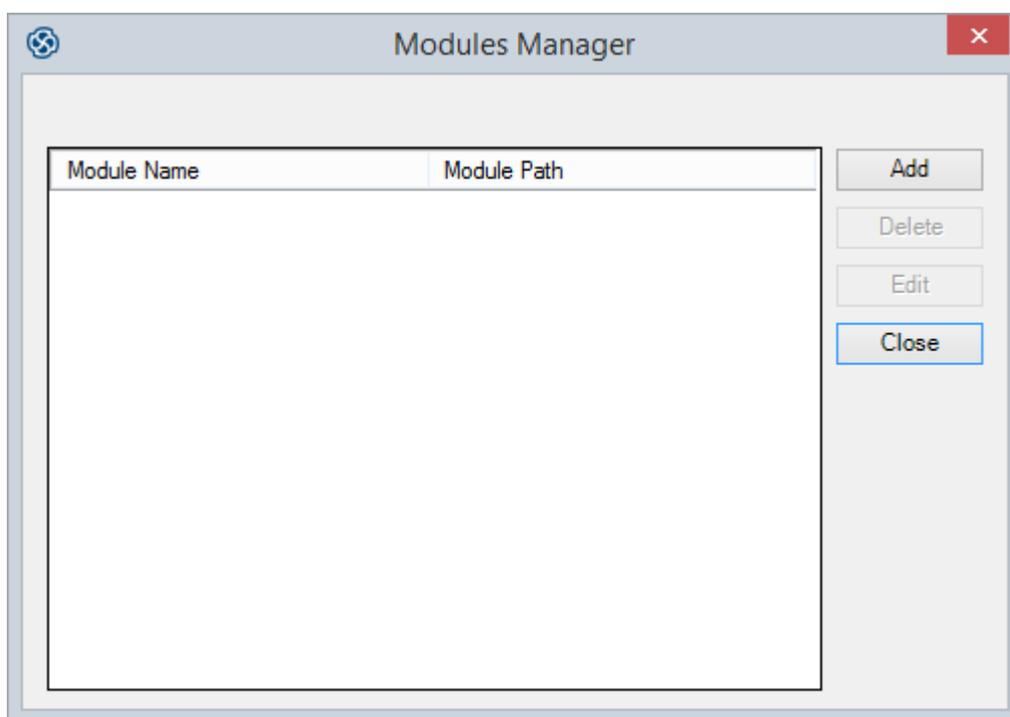
In the MDG Link for DOORS you can create a link between a selected Sparx Systems Enterprise Architect Package and an existing IBM® Rational® DOORS® module, which enables you to exchange requirements data between DOORS and Enterprise Architect. You can also redirect the link to a different module. Through this link you can import data from DOORS to Enterprise Architect, and export data from Enterprise Architect to DOORS.

Access

Project Browser | Right-click on target Package | Extensions | Doors Extension | Add/Edit Module or
Project Browser | Click on target Package | (Main Menu) Extensions | Doors Extension | Add/Edit Module

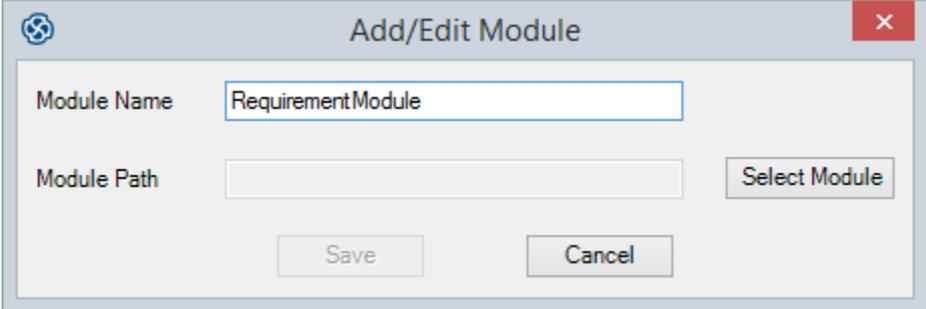
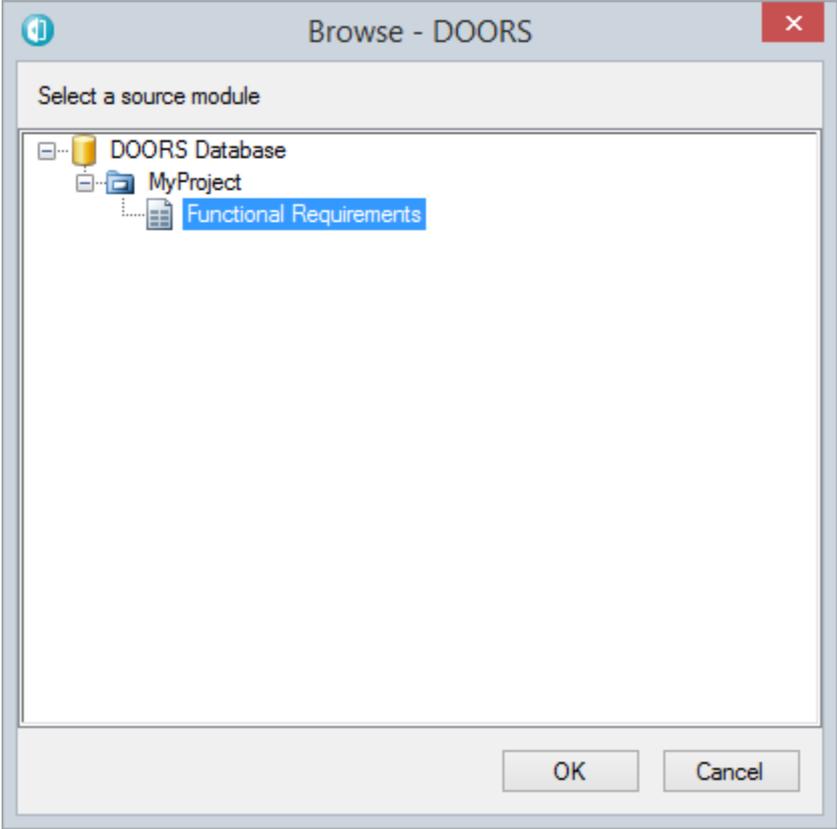
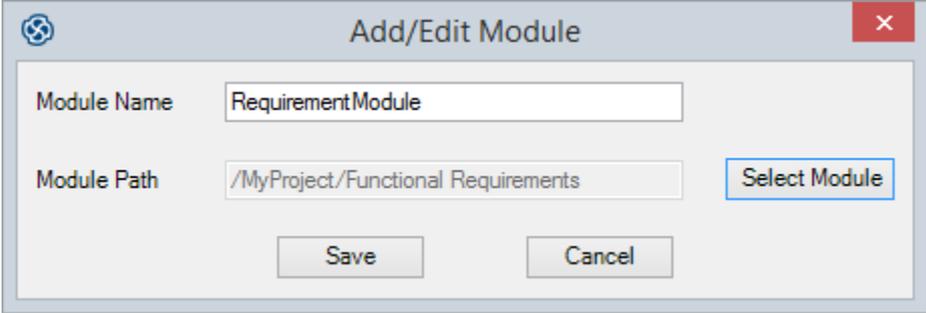
Modules Manager Dialog

You use the 'Modules Manager' dialog to locate an existing DOORS module and create a link to it.

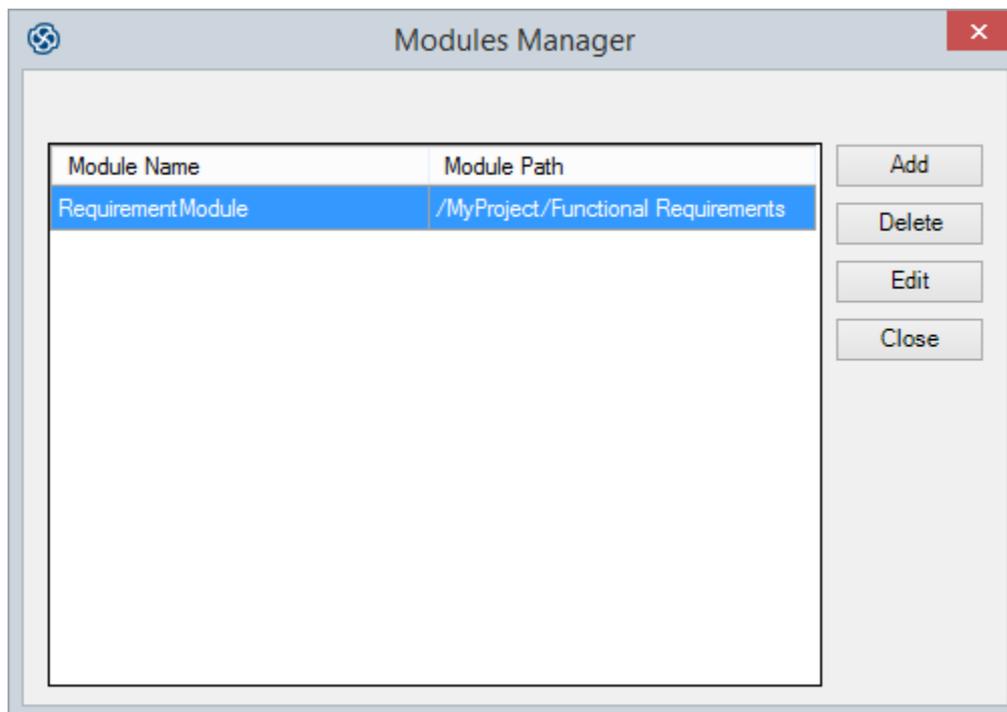


Create a link to a DOORS Module

Step	Action
1	On the 'Modules Manager' dialog, click on the Add button . The 'Add/Edit Module' dialog displays.

	
2	In the 'Module Name' field, type a name to identify the link to the module.
3	<p>Click on the Select Module button to open the 'Browse - DOORS' dialog (in DOORS itself).</p> 
4	Expand the module hierarchy as necessary to locate the required module, and click on it.
5	<p>Click on the OK button to return to the 'Add/Edit Module' dialog, which now displays the module path and name in the 'Module Path' field.</p> 

- 6 Click on the **Save button**. The 'Module Manager' dialog redisplay, showing the newly-created module link. You can now use this link to exchange requirements data between Enterprise Architect and DOORS.



Click on the **Close button** to close the dialog.

Edit or Delete a link to a DOORS module

If the existing link to the DOORS module is not appropriate, you can delete or redirect it.

Step	Action
1	On the 'Modules Manager' dialog, click on the link to change. <ul style="list-style-type: none"> To delete it, click on the Delete button and on the Close button; this ends the procedure To redirect it, click on the Edit button; the 'Add/Edit Module' dialog redisplay
2	On the 'Add/Edit Module' dialog, click on the Select Module button to display the 'Browse - DOORS' dialog (in DOORS).
3	Expand the hierarchy as necessary on the 'Browse - DOORS' dialog, and click on the replacement module for the link.
4	Click on the OK button to return to the 'Add/Edit Module' dialog, which now shows the module path of the replacement module. Note that you cannot edit the link name .
5	Click on the Save button . The 'Module Manager' dialog redisplay, showing the edited module link. Click on the Close button to close the dialog.

Notes

- You can also create and edit links to DOORS modules through the 'Export to Doors' and 'Import from Doors' dialogs, using the **Module Manager button**

Export Requirements to DOORS

Using the Sparx Systems MDG Link for DOORS, you can transfer all the Requirement elements under the selected Enterprise Architect Package to a linked IBM® Rational® DOORS® module, as DOORS objects.

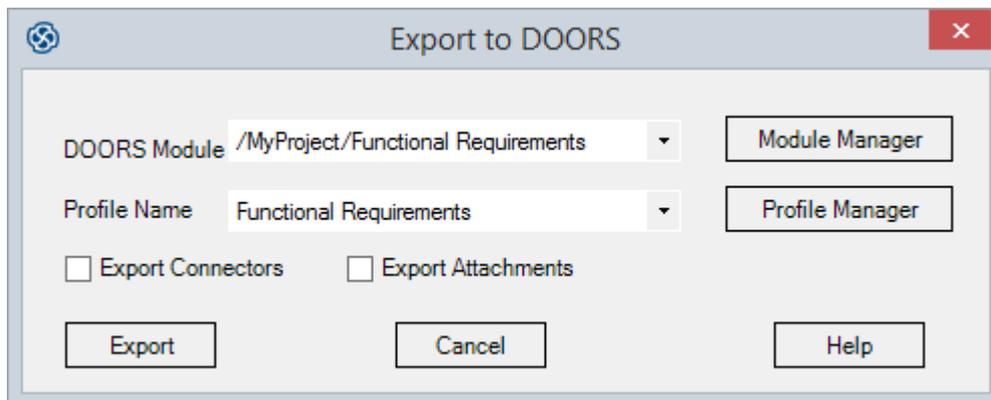
As part of this process you can create an export profile to define what the export should operate on. An export profile is a template in which you specify the Enterprise Architect element properties to be exported to the mapped fields in DOORS object properties. The profile can be reused when you import and synchronize the DOORS project into Enterprise Architect again.

Access

Project Browser | Right-click on target Package | Extensions | Doors Extension | Export to Doors or
 Project Browser | Click on target Package | (Main Menu) Extensions | Doors Extension | Export to Doors

Export to Doors Dialog

On the 'Export to Doors' dialog you set the module you are exporting into, and specify which export profile to apply and whether to also export connectors and attachments.



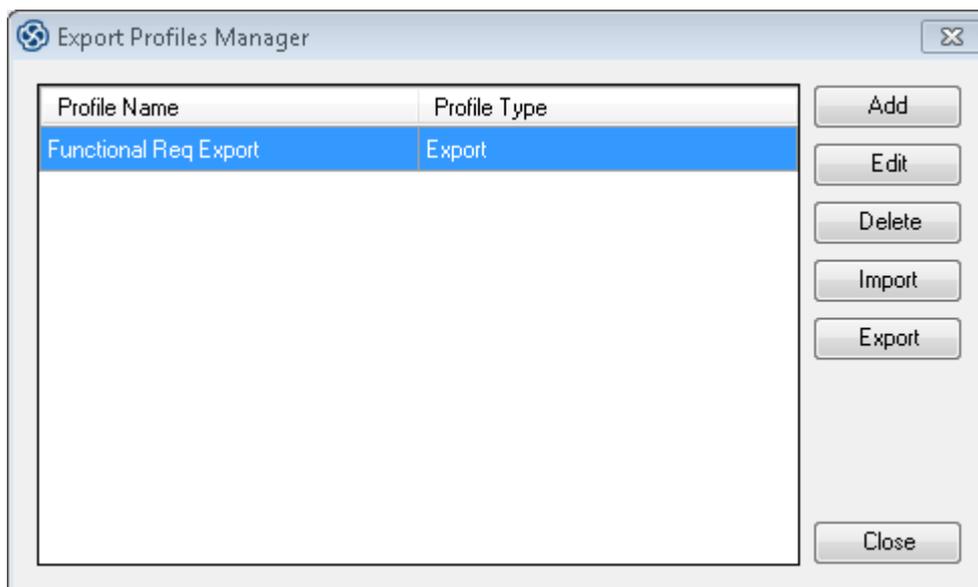
Export Options

Field/Button	Action
DOORS Module	Click on the drop-down arrow and select the linked DOORS module into which to export the Requirements.
Module Manager	If the required module is not listed, click on this button to display the 'Module Manager' dialog, and locate and link to the module.
Profile Name	Click on the drop-down arrow and select the appropriate export profile to use in this operation.
Profile Manager	If the export profiles listed are not appropriate, click on this button to display the 'Export Profiles Manager' dialog and create, edit or import the profile.

	(See the <i>Manage Export Profiles</i> table.)
Export Connectors	Select this checkbox to export any connectors between the Requirement elements.
Export Attachments	Select this checkbox to export any attachments that the Requirement elements might have.
Export	Click on this button to begin the export of Requirements from the selected Enterprise Architect Package to the specified DOORS module.
Cancel	Click on this button to close the 'Export to Doors' dialog without exporting any Requirements.
Help	Click on this button to display this Help topic.

Export Profiles Manager Dialog

When you click on the **Profile Manager button** on the 'Export to Doors' dialog, the 'Export Profiles Manager' dialog displays. You use this dialog to create or import new profiles, edit or delete existing profiles, and export profiles to your preferred file system.



Manage Export Profiles

Field/Button	Action
Add	Click on this button to add a new export profile. The 'Mapping Options' dialog displays; see the <i>Create or Edit an Export Profile</i> table.
Edit	Click on an existing profile name and click on this button to edit the

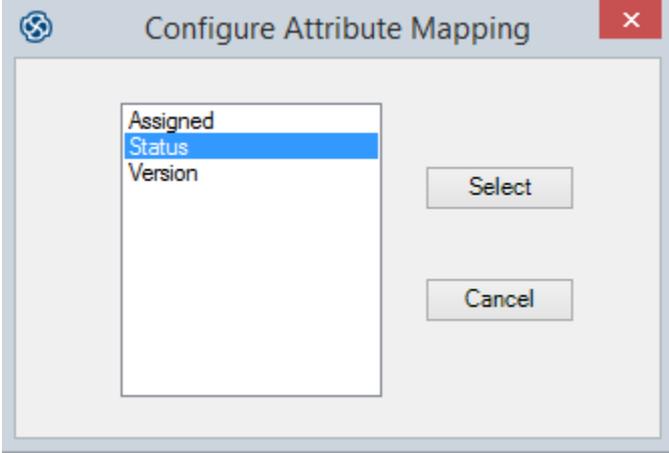
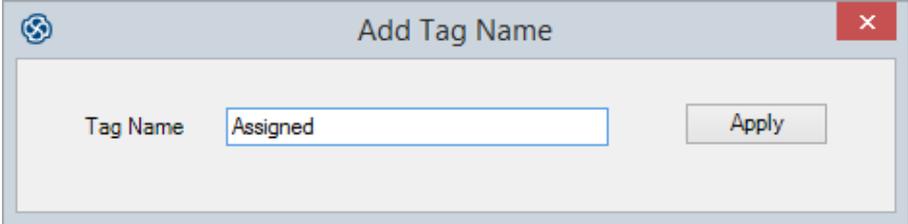
	parameters of that profile. The 'Mapping Options' dialog displays, showing the current parameters of the profile; see the <i>Create or Edit an Export Profile</i> table.
Delete	Click on an existing profile name and click on this button to immediately remove that profile from the list. There is no confirmatory prompt.
Import	Click on this button to display the 'Import a Profile - Doors Extension' browser. Browse for the location of the required profile file (.eProfile), and click on the file name and on the Open button . A status message displays. Click on the OK button ; if the operation was successful, the profile name displays on the 'Export Profiles Manager' dialog. If unsuccessful, the profile is not added to the list.
Export	Click on a profile name and click on this button to export that profile to a file system location as a .eProfile file. The 'Save As' browser displays. Browse to the required location and click on the Save button . A status message displays; click on the OK button to clear the message.
Close	When you have finished using the 'Export Profiles Manager' dialog, click on this button to return to the 'Export to Doors' dialog.

Export Mapping Options dialog

When you click on the **Add button** or **Edit button** on the 'Export Profiles Manager' dialog, the 'Mapping Options' dialog displays. This dialog maps Enterprise Architect element properties to IBM® Rational® DOORS® Object properties, and helps you to define which properties - including **Tagged Value** properties - to export. You can use the selected profile many times to update the DOORS module with changes in the Enterprise Architect Requirements.

Create or Edit an Export Profile

Field/Button	Action
Profile Name	Type in an appropriate name for the profile (if editing an existing profile, the name of the selected profile displays here).
Choose an Action	Defaults to 'Export'. You can click on the drop-down arrow and change this action to 'Import' if you are switching the direction of the data exchange.
Element Type	Click on the drop-down arrow and select either 'Requirement' or 'Use Case'.
Stereotype	Click on the drop-down arrow and select the type of Requirement or Use Case to filter for.
Enterprise Architect	Lists the Enterprise Architect element properties that could be exported.
DOORS	Lists the DOORS properties that can be exported to. Double-click on an empty property field to map that field to an Enterprise Architect property. If there are no available properties, an error message displays; otherwise, the 'Configure Attribute Mapping' dialog displays.

	
Add Custom Field	<p>Click on this button to select an Enterprise Architect Tagged Value name to export. The 'Add Tag Name' dialog displays.</p> 
Remove	<p>Click on a DOORS property value and click on this button to clear the value mapped to the corresponding Enterprise Architect property.</p>
Save Profile	<p>Click on this button to save the Profile definition you have created, and close the 'Mapping Options' dialog and return to the 'Export Profiles Manager' dialog.</p>
Cancel	<p>Click on this button to close the 'Mapping Options' dialog without saving any profile data you have entered.</p>

Import Requirements from DOORS

Using the Sparx Systems MDG Link for DOORS, you can transfer all the objects in a linked IBM® Rational® DOORS® module into the selected Enterprise Architect Package, as Requirement elements.

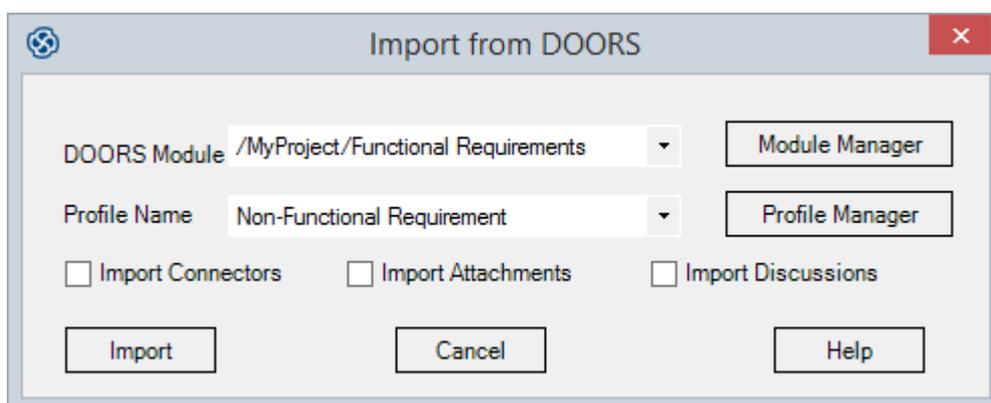
As part of this process you can create an import profile to define what the import should operate on. An import profile is a template in which you specify the DOORS object properties to be imported into mapped Enterprise Architect element properties. The profile can be reused when you export and synchronize the Enterprise Architect Package into DOORS again.

Access

Project Browser | Right-click on target Package | Extensions | Doors Extension | Import from Doors or
Project Browser | Click on target Package | (Main Menu) Extensions | Doors Extension | Import from Doors

Import from Doors Dialog

The 'Import from Doors' dialog helps you to specify which module you are importing from, which import profile you are using, and whether to import connectors, attachments and discussions.



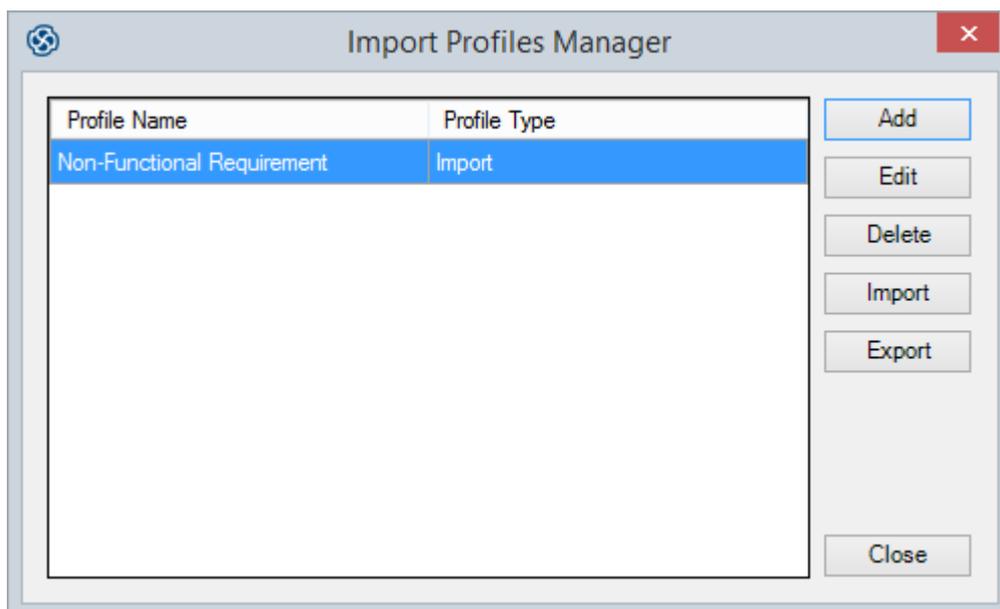
Import Options

Field/Button	Action
Doors Module	Click on the drop-down arrow and select from the list of DOORS modules that are linked to this project.
Module Manager	If the module you want is not listed in the 'Doors Module' field, click on this button to display the 'Modules Manager' dialog and locate and link to the module you require.
Profile Name	Click on the drop-down arrow and select the Import profile to apply to this operation.
Profile Manager	If an appropriate profile to use is not listed in the 'Profile Name' field, click

	on this button to display the 'Import Profiles Manager' dialog (see the Manage <i>Import Profiles</i> table).
Import Connectors	Select this checkbox to import any connectors (relationships) between the imported DOORS objects.
Import Attachments	Select this checkbox to import any attachments that the imported DOORS objects might have.
Import Discussions	Select this checkbox to import any discussions associated with the imported DOORS objects.
Import	Click on this button to begin the import process.
Cancel	Click on this button to close the 'Import from Doors' dialog without performing an import.
Help	Click on this button to display this Help topic.

Import Profiles Manager Dialog

When you click on the **Profile Manager button** on the 'Import from Doors' dialog, the 'Import Profiles Manager' dialog displays. You use this dialog to create or import new profiles, edit or delete existing profiles, and export profiles to your preferred file system.



Manage Import Profiles

Field/Button	Action
	Click on this button to add a new Import profile. The 'Mapping Options'

Add	dialog displays; see the <i>Create or Edit an Import Profile</i> table.
Edit	Click on an existing profile name and click on this button to edit the parameters of the selected profile. The 'Mapping Options' dialog displays, showing the current parameters of the profile; see the <i>Create or Edit an Import Profile</i> table.
Delete	Click on an existing profile name and click on this button to immediately remove that profile from the list. There is no confirmatory prompt.
Import	Click on this button to display the 'Import a Profile - Doors Extension' browser. Browse for the location of the required profile file (.eProfile), and click on the file name and on the Open button . A status message displays. Click on the OK button ; if the operation was successful, the profile name displays on the 'Import Profiles Manager' dialog. If unsuccessful, the profile is not added to the list.
Export	Click on a profile name and click on this button to export that profile to a file system location as a .eProfile file. The 'Save As' browser displays. Browse to the required location and click on the Save button . A status message displays; click on the OK button to clear the message.
Close	When you have finished using the 'Import Profiles Manager' dialog, click on this button to return to the 'Import from Doors' dialog.

Import Mapping Options Dialog

When you click on the **Add button** or **Edit button** on the 'Import Profiles Manager' dialog, the 'Mapping Options' dialog displays. This dialog maps IBM® Rational® DOORS® Object properties to Enterprise Architect Requirement or Use Case element properties, and helps you to define which properties to import. You can use the profile many times to update the Enterprise Architect Package with changes in the DOORS module objects.

Mapping Options

Profile Name:

Choose an Action:

Element Type:

Stereotype:

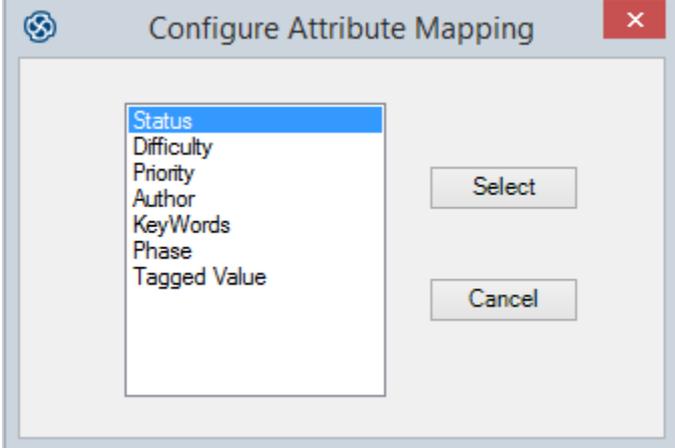
DOORS	Enterprise Architect
Absolute Number	Tagged Value
Assigned	
Created By	Tagged Value
Created On	
Created Thru	
Last Modified By	Tagged Value
Last Modified On	
Object Heading	Name
Object Short Text	Alias
Object Text	Notes
Status	
Version	Version

Remove

Save Profile Cancel

Create or Edit an Import Profile

Field/Button	Action
Profile Name	Type in an appropriate name for the import profile.
Choose an Action	Click on the drop-down arrow and select 'Import'. You can also reset the profile to 'Export' to use it for exporting Requirements from the Enterprise Architect Package to the DOORS module.
Element Type	Click on the drop-down arrow and select the element type to create or overwrite with the DOORS data - either 'Requirement' or 'Use Case'.
Stereotype	Click on the drop-down arrow and select any stereotype you want to set on the elements resulting from the import.
DOORS	Lists the DOORS object properties available in the selected DOORS module.
Enterprise Architect	Lists the Enterprise Architect properties mapped to the DOORS properties. You can double-click on an empty property field to display the 'Configure

	<p>Attribute Mapping' dialog, which provides a list of Enterprise Architect properties that can be mapped to the DOORS property.</p>  <p>Click on a property and on the Select button to map it to the DOORS property.</p>
Remove	<p>If necessary, click on a mapped Enterprise Architect property field and click on this button to clear the field and unmap the property.</p>
Save Profile	<p>Click on this button to save the profile definition you have created and return to the 'Import Profiles Manager' dialog.</p>
Cancel	<p>Click on this button to close the 'Mapping Options' dialog without saving the profile information.</p>

Additional Requirements Tools

In addition to the Key Tools listed in *Meet the Requirements* there are a number of other tools that can be used for Requirements development and management and team collaboration in the requirements engineering discipline. These include the ability to assign sequential numbers to requirements, importing Requirements from a spreadsheet file, creating requirements documentation and team collaboration features such as **Element Discussions**.

More Requirements Tools

Tool	Overview
Auto Names and Counters	Use to assign a sequential number to a Requirement including a prefix and suffix.
Requirements Checklist	Provides a graphical list of checks that can be applied to individual requirements.
Import and Export Spreadsheets	A tool to import and export Requirements from Spreadsheet files in the CSV format.
Documentation	A powerful engine to automatically create high quality documentation directly from the model, using built-in or user defined templates.
Glossary	A feature to create and maintain a lexicon of terms and their meaning that can grouped by type and styled when included in documentation.
Auditing	Used to keep a trail of what has changed in a repository, who it was changed by and when.
Element Discussions	A facility to allow modelers to create posts and replies to discuss model elements.
Maintenance Items	A series of Items that can be applied to requirements to define such things as Changes, Issues, Defects and more.
Team Reviews	Allows reviews to be created for user defined categories and topics with links to model elements such as Requirements and Scenarios that can be referenced in the review.

Auto Names and Counters

Getting to know Auto Names and Counters

Introducing Auto Names and Counters

To aid, regulate and enforce a naming standard, Enterprise Architect includes some capabilities to configure the default names assigned to new elements of a specific type. This is a useful feature when dealing with complex and large sets of requirements, but is also relevant when dealing with smaller data sets. **Auto Names and Counters** can be used to assign a sequential number to any element type including Requirements. It includes a prefix definition, a counter and a suffix definition allowing numbers such as: 'REQ007 - Manage Inventory' to be created.

The screenshot shows a configuration dialog for 'Requirement' type elements. It has a 'Type' dropdown set to 'Requirement'. Below it are two sections: 'Name' and 'Alias'. The 'Name' section has fields for 'Prefix' (REQ), 'Counter' (001), and 'Suffix' (-), with an 'Apply on creation' checkbox checked. The 'Alias' section has empty fields for 'Prefix', 'Counter', and 'Suffix', with an 'Apply on creation' checkbox unchecked. On the right side, there are 'Save', 'Close', and 'Help' buttons.

Where to find Auto Names and Counters

Main Menu: Project | Settings | **Auto Names and Counters**

Usage of Auto Names and Counters

Analysts and others can use the sequential number for communicating unambiguously about the requirements without having to use the often long requirement name. The 'Apply on Creation' option must be checked to start using the auto numbering feature, this can also be used to temporarily suspend auto naming for example if other types of requirements are being entered that don't need to have sequential numbers assigned.

Options for Auto Names and Counters

There are options to define the prefix, counter and suffix for a requirement.

This screenshot is similar to the one above, but the 'Counter' field is set to '0001' instead of '001'. The 'Apply on creation' checkbox is checked.

Learn more about Auto Names and Counters

[Apply Auto Naming to Existing Elements](#)

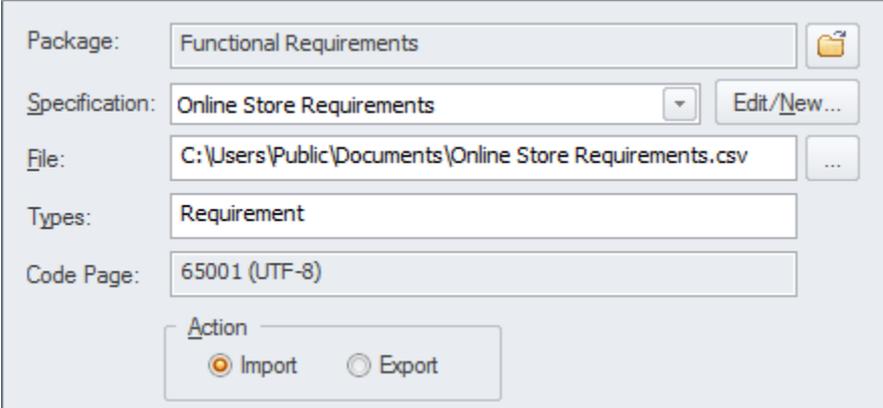
Import and Export Spreadsheets

Import and Export Spreadsheets

Introducing Import and Export Spreadsheets

This facility is a useful mechanism to import requirements that have been defined in a Spreadsheet or a Word Processor table into Enterprise Architect. Once in Enterprise Architect the requirements can be managed and traced to elements such as business drivers and Scenarios and Components. Alternatively Requirements in Enterprise Architect can be exported to a Spreadsheet for the purposes of providing them to a third party or for some type of numerical or statistical analysis. The mapping between fields in the Spreadsheet and the analogous properties in Enterprise Architect is completely configurable using a specification.

For more detailed information exchange, the Microsoft Office **Add-In** (available from Sparx Systems) provides additional functionality and integration points useful when dealing with complex requirements.



The screenshot shows a dialog box with the following fields and options:

- Package: Functional Requirements
- Specification: Online Store Requirements (with an Edit/New... button)
- File: C:\Users\Public\Documents\Online Store Requirements.csv (with a browse button)
- Types: Requirement
- Code Page: 65001 (UTF-8)
- Action: Import Export

Where to find Import and Export Spreadsheets

Main Menu: Package | Import/Export | **CSV Import/Export...**

Use of Import and Export Spreadsheets

This feature can be used to import or export Requirements from a CSV file. Before a tool like Enterprise Architect is installed analysts may have used a Spreadsheet or a table in their favorite word processor to record requirements, these can conveniently be imported using the CSV import facility. Alternatively Requirements sometimes need to be provided to a third party who will typically specify that they want them in a Spreadsheet file and this can be achieved using the export facility.

Options to Import and Export Spreadsheets

The import and export facility is completely configurable and has a user defined specification that can be used to determine how fields in the Spreadsheet file are mapped to Requirements properties in Enterprise Architect including being able to import and export fields to and from **Tagged Values** of the Requirement.

Learn more about Import and Export Spreadsheets

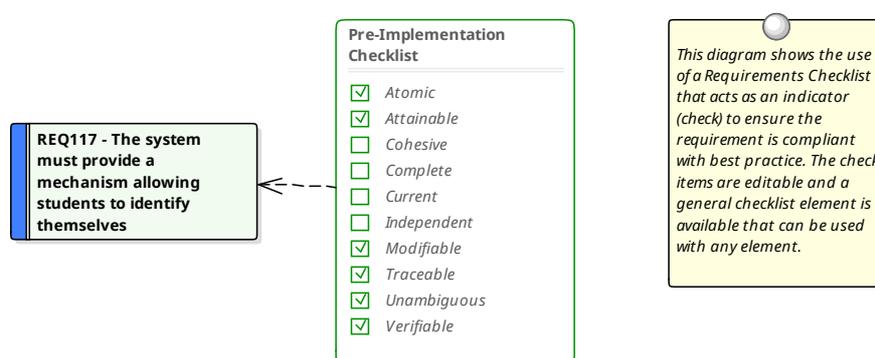
- [CSV Import and Export](#)

Requirements Checklist

Getting to know the Requirements Checklist

Introducing the Requirements Checklist

The Requirements Checklist is a convenient element that acts as a tally to indicate whether a Requirement complies with a set of predefined measures such as whether the Requirement is Atomic, Cohesive, Traceable and Verifiable. It can be assigned to any Requirement and the measures can be updated directly in the diagram. When working with requirements it is sometimes very useful to refer to a common set of 'best practices' and qualities that help define the nature of a well formed specification. The Requirement Checklist element is designed to meet this need.



Where to find the Requirements Checklist

Toolbox | More Tools... | Requirements | Extended Requirements Page | Requirements Checklist

Usage of the Requirements Checklist

Analysts and Requirements Managers can use the checklist to annotate whether one or more requirements meet a set of predefined checks.

Options for the Requirements Checklist

The list of measures is completely configurable and items can be added or removed from the list for each individual checklist by using the Checklist Tagged Value notes.

```
<Checklist>
  <Item Text="Atomic" Checked="True"/>
  <Item Text="Attainable" Checked="True"/>
  <Item Text="Cohesive" Checked="False"/>
  <Item Text="Complete" Checked="False"/>
  <Item Text="Current" Checked="True"/>
  <Item Text="Independent" Checked="False"/>
  <Item Text="Modifiable" Checked="True"/>
  <Item Text="Traceable" Checked="True"/>
  <Item Text="Unambiguous" Checked="True"/>
  <Item Text="Verifiable" Checked="True"/>
</Checklist>
```

Learn more about the Requirements Checklist

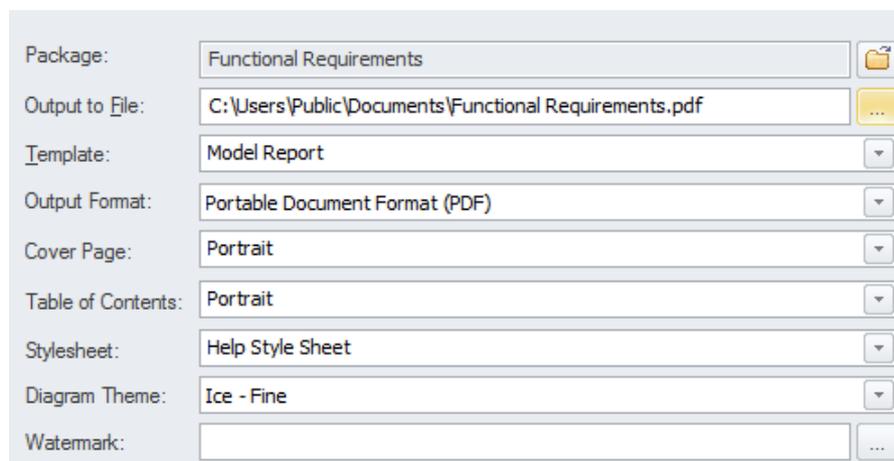
Requirement Checklist

Documentation

Getting to know Documentation

Introducing Documentation

The documentation features can be used to automatically generate a wide range of documentation directly from the models. These can be document based such as PDF and Docx format or html based. Flexible templates can be used to completely tailor the documents that are generated including company logos, tables of content, tables of element information and diagrams. Ad-hoc reports can also be created from a number of tools such as the Glossary and the Search Window.



Package:	Functional Requirements	...
Output to File:	C:\Users\Public\Documents\Functional Requirements.pdf	...
Template:	Model Report	▼
Output Format:	Portable Document Format (PDF)	▼
Cover Page:	Portrait	▼
Table of Contents:	Portrait	▼
Stylesheet:	Help Style Sheet	▼
Diagram Theme:	Ice - Fine	▼
Watermark:		...

Where to find Documentation

Main Menu: Package | Documentation | **Document Generator**/Publish as HTML/Glossary..

Use of Documentation

Modelers, Analysts, Architects, Project managers and others can use the facility to produce a wide range of document based publications and reports, such as a: System Requirements Specification, Use Case Report, Data Dictionary, Solution Architecture Description and more. It can also be used for Ad-hoc reporting to create reports such as a list of the most volatile requirements. Html documentation can also be published to allow stakeholders who don't have access to Enterprise Architect to view the models from an Intranet site that can just be placed on a file system without the need for a Web Server.

Options for Documentation

There are several options that can be set to tailor the information that is included in a generated document, including the ordering of elements and diagrams and hiding certain elements. Filters and word substitutions and other options can also be applied.

The screenshot displays two panels from the Enterprise Architect software. The 'Order' panel on the left contains three rows of dropdown menus: 'Packages by:' with 'Tree Order' and 'Ascending'; 'Elements by:' with 'Modified Date' and 'Descending'; and 'Diagrams by:' with 'Name' and 'Ascending'. Below these are 'Options' with various checkboxes, including 'Hide 'note-less' elements' (checked), 'Hide Diagram Borders' (checked), 'Skip root package', 'No bookmarks', 'Hide <Anonymous> elements', 'Use style defined in template for notes', 'Disable large OLE file support', 'Insert page breaks when generating a Master Document', 'Include child elements even if the parent element is filtered out' (checked), and 'Hide 'note-less' connectors'. At the bottom of the options are 'Diagram Format:' set to 'Metafile' and 'Adjust Heading Levels:' set to 'Heading 9', along with a 'Switch generator' button. The 'Filter' panel on the right includes 'Only include objects:' with 'Created', 'After', and '05/06/2015'; 'Where Package Phase:' with '>' and '3'; 'With element status:' with an empty dropdown; 'Connector Direction:' set to 'Both'; and 'Except where Query excludes Package:' with radio buttons for 'Custom SQL' (selected) and 'Custom Script'. A large empty text area is at the bottom of the filter panel.

Learn more about
Documentation

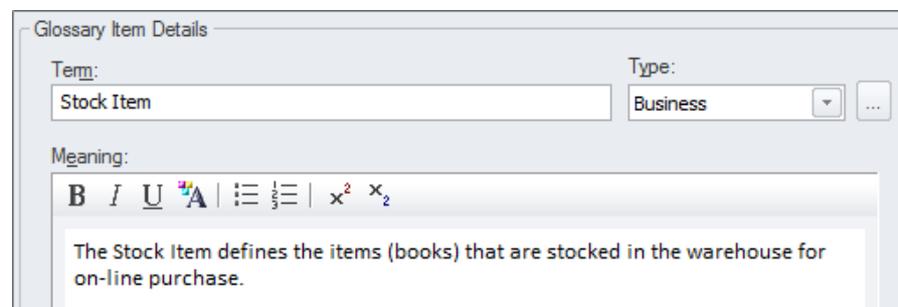
- [Report Generation](#)

Glossary

Getting to know the Glossary

Introducing the Glossary

The Glossary is a project level lexicon of the important terms and their meanings categorized by type. Any number of terms, their types and meanings can be defined and these can be referenced from the notes of model elements. The terms can be included in documentation or generated as a stand-alone report. When working with domain specific requirement specifications, architectures and other models it is essential that new terms and over-ridden meanings for common words or phrases are kept in a suitable dictionary format to ensure proper understanding of documentation and specifications.



Where to find the Glossary

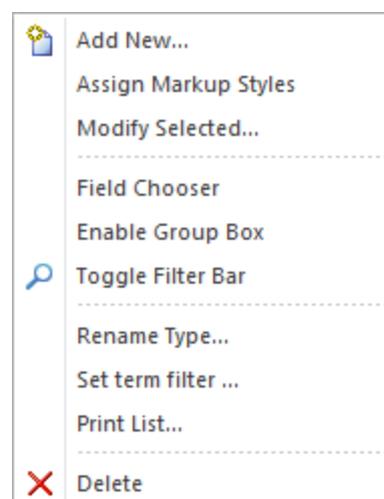
Main Menu: Project | Glossary

Usage of the Glossary

The project Glossary can be used to record the important terms of a project or domain grouped by the type of term, allowing business, technical and domain specific types to be defined. A glossary report can be generated as a stand alone report or the glossary can be included as a section of another document.

Options for the Glossary

The Glossary has a number of options to determine the terms that are displayed in the list and to define the style that is used for types of terms in documentation.



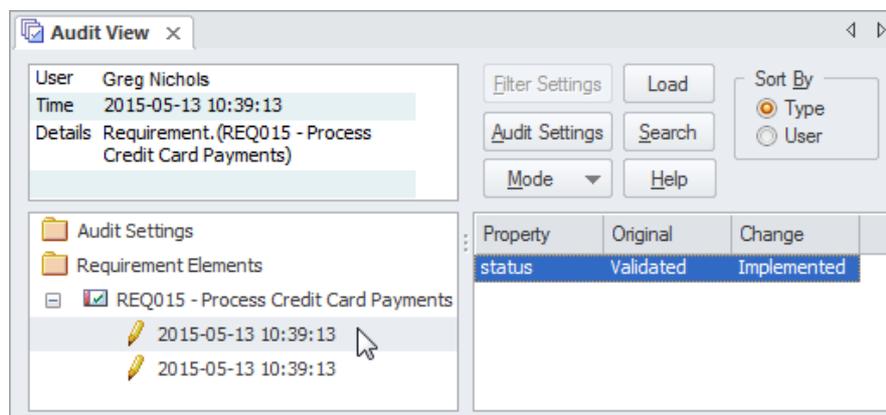
**Learn more about the
Glossary** [Project Glossary](#)

Auditing

Getting to know Auditing

Introducing Auditing

The **Auditing** feature can keep track of the changes to Requirements including what was changed, when it was changed and by whom. Auditing is by default disabled and must be enabled before the changes to requirements will be recorded. Once enabled it is a passive tool that silently records the changes to elements. It does not replace version control or baselines and in contradistinction to these tools it can not be used to return to a previous state of the model. Change management, governance and quality control are all aided by the use of Auditing.



Where to find Auditing

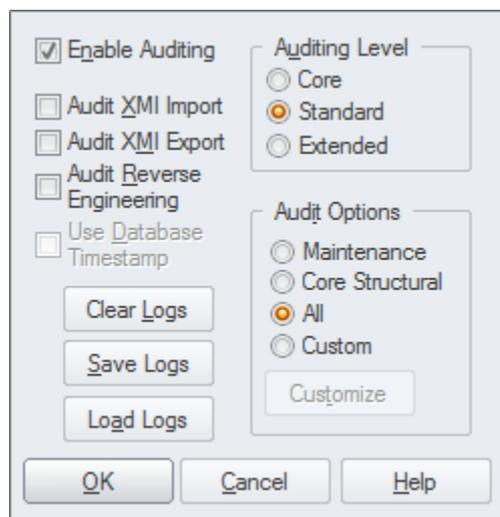
Main Menu: Project | **Auditing**

Use of Auditing

Auditing can be used to track what was changed in a model, who changed it and when. There are a number of modes and a repository administrator can use the settings to specify what is recorded in the audit. While a baseline can be used to show the difference between a model and a snapshot at a point in time, the Auditing tool records each individual change; it can not, however, be used to revert to a previous state.

Options for Auditing

There is a wide range of settings to configure auditing, starting with enabling or disabling the settings that determine which elements have an audit trail and the level of detail recorded. Audit logs can be exported from the repository to increase performance.



Learn more about
Auditing

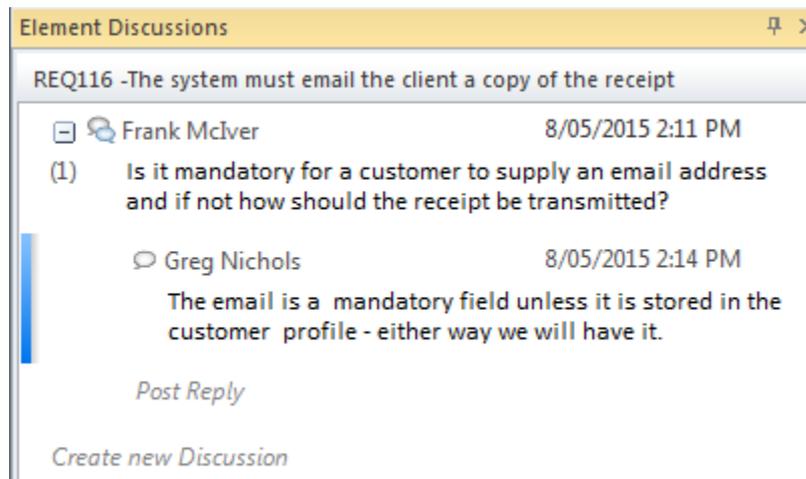
- [Auditing](#)

Element Discussions

Getting to know Element Discussions

Introducing the Element Discussions

The **Element Discussions** facility allows modelers to have conversations about elements posting discussions and replying to exiting posts. The discussions for all elements in the model are conveniently listed in the **Discussions Review** window allowing a modeler to see all the elements with posts.



Where to find Element Discussions

To post or view an element's discussion
Main Menu: Element | **Discussions**

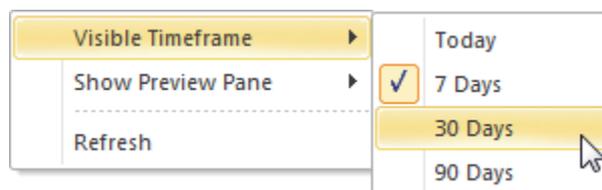
To view all discussions in the repository
Main Menu: Project | Discussions

Usage of Element Discussions

Element Discussions allow modelers to have conversations about elements without 'polluting' the element's notes with questions and modeling level comments such as 'Properties need to be added before the first release'. This features brings the collaborative modeling platform alive where modelers can add informal discussions about elements emulating being in a workshop.

Options for the Element Discussions

The **Discussion Review** window has a number of options to tailor the discussions that are listed including defining the timeframe allowing older and perhaps less relevant discussions to be hidden.



Learn more about Element Discussions

[Element Discussions](#)

Maintenance Items

Getting to know Maintenance Items

Introducing Maintenance Items

Element Maintenance Items can be used with any element including: Requirements, Components, Classes and other elements to capture problems, changes, issues, tasks, events and decisions that affect an individual element. Using maintenance items to track official changes, additions and deletions to sets of Requirements, Components, User Stories and other specifications provides a sound basis for overall project governance and traceability.

Decision:

Owner: Date: Status:

Author: Effective: Impact:

Version / ID:

Description History

B I U A | | x^2 x_2 |

The question had been raised as to how to display items that had zero items in stock but that had back orders in place. The decision has been made to display the back order quantities as a negative number, to mark them with an asterisk and to put an annotation on the report explaining the negative number. "Negative stock numbers indicate total back order quantities"

Where to find Maintenance Items

Main Menu: Element | Maintenance | Defects\Changes\Issues\Tasks\Decisions\Events

Usage of Maintenance Items

Element Maintenance Items can be used to record a wide range of information about elements including: changes, issues and tasks and more. These can be used to track the way an element is altered and maintained and can provide information about the Item including the rationale for the alteration, status and priority. While the Baseline and **Auditing** features record what has changed automatically the Maintenance Items provide the flexibility for a modeler to specify details manually.

Options for Maintenance Items

There are a range of maintenance items that can be used to add information to elements these include: Defects, Changes, Issues, Tasks, Decisions and Events. Items can have information such as who requested the item, who completed the item and status and priority.

Learn more about Maintenance Items

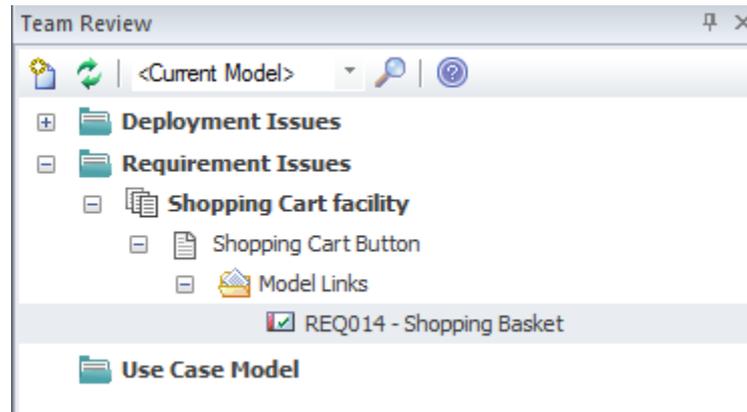
[Maintenance Items](#)

Team Reviews

Getting to know Team Reviews

Introducing Team Reviews

Team reviews provide an opportunity for developers, modelers, customers and stakeholders to comment and provide feedback on the work in progress or at the completion of a milestone or project.



Usage of Team Reviews

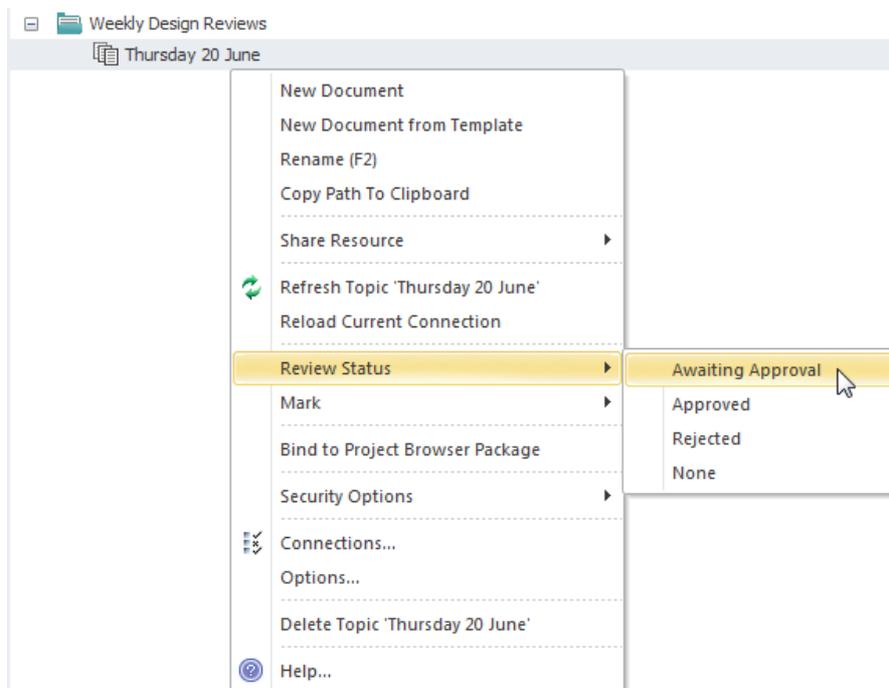
The **Team Review** feature can be used to conduct model reviews from any number of perspectives including walk-throughs, formal model reviews, or ad-hoc reviews.

Where to find Team Reviews

Main Menu: Project |Team Reviews

Options for Team Reviews

There is a wide range of settings available to configure the Team Reviews available from the Category and Topic context menus, including setting the status of the category or topic and other options. Diagrams, elements and element features can be conveniently dragged from the **Project Browser** to create model links that can be used by team members to hyperlink directly from the **Team Review** window to these items in the Project Browser.



Learn more about Team Reviews

[Team Review Tools](#)

