Enterprise Architect

User Guide Series

Model Repository

Author: Sparx Systems

Date: 30/06/2017

Version: 1.0

CREATED WITH ENTERPRISE ARCHITECT

Table of Contents

Model Repository

Enterprise Architect is a rich modeling platform that can be used in a wide range of modeling and design situations, from a single user accessing a file based repository to a large and distributed team of thousands of modelers working all over the world using a Cloud Based Repository. The repositories can be scaled up and down, and a project that started with a single user can be seamlessly scaled up to a large multi-user repository.

Enterprise Architect has been built to support inter-disciplinary modelers collaborating to create a single and unified view of a system or enterprise, and the platform has numerous built in features to support this team development. By taking advantage of the base UML modeling language, in addition to the numerous extension languages such as BPMN and SysML, plus highly effective frameworks such as TOGAF and UAF/UPDM, modelers, designers, managers and others can build comprehensive, inter-connected models that fully expresses the capabilities, structure and behavior of fully functioning, multi-layered, complex systems and/or enterprises.

In addition, there are features that allow modelers to discuss the model, such as the element discussion window, and review work that has been in progress in the Team Review window. Modelers can even send Model Mail to communicate ideas. Model Security can be implemented, which facilitates collaboration and prevents work accidentally being overridden.

There are a range of project management features, such as tasks and resource allocation and a Gantt Chart to view progress on tasks over time, and a Project Calendar that keeps track of important project events.

Integration is so easy and the feature sets so powerful that strategic thinkers, business managers, architects, information and data modelers, testers, software engineers and many others will derive great benefit from using the tool and incorporating their models into the overall vision.

Key Concepts

Repository Format	Enterprise Architect models are stored in **standard relational databases**. This approach has been built into Enterprise Architect from the very first version. Using a relational database provides a huge advantage over other tools that are limited in scope and capability by relying on a simple text file based model storage system. With support for two file based repository formats (MS JET and Firebird) plus a number of industry standard DBMS servers (such as SQL Server, MySQL and Oracle) plus Cloud based storage, Enterprise Architect provides fast, scalable, flexible and transparent access to models in a wide range of deployment scenarios.
Modeling Languages	Enterprise Architect is based on industry standard UML, a specification maintained by the Object Management Group (OMG). Although initially conceived as a modeling language primarily for software development, over the years the capabilities and formal (standard) extensions have allowed the UML to develop into a rich set of constructs and modeling behaviors that provide very rich and detailed support for everything from software, to business processes, embedded software, enterprise architecture, organizational charts, strategic model, mind mapping, requirements management and much more. The success of UML has been proven over and over again in the real world and the number of new technologies and modeling profiles that use UML as the underlying 'meta-language' are tribute to the power and effectiveness of UML.
Team Development	From the very first version of Enterprise Architect the goal has been to provide the

best and most accessible support for Team based development. A wealth of tools and capabilities have been designed into the platform over the years to support security, scalability, information sharing, concurrent access, management, reporting and querying. With the growth of the internet and distributed development scenarios, Enterprise Architect has evolved to be the pre-eminent platform for developing models and designs in a globally distributed situation. The Cloud Server based deployment offers high availability, excellent performance and simple configuration to rapidly bring diverse and distributed teams into a single, living repository where work can be easily shared, reviewed, discussed and managed.

Scalability

By leveraging the power of today's DBMS servers and Cloud Based technologies, Enterprise Architect can be easily scaled up to truly epic proportions. Millions of objects can be stored and accessed by large and dispersed teams in real time. The level of scalability is limited solely by the choice of DBMS or Cloud Technology and the capabilities of the network and server infrastructure the model is deployed on. In practice, over the many years that Enterprise Architect has been used in industry, modelers and designers have taken advantage of the powerful in-built scalability to construct stunning and often huge models that capture an enormous amount of information about complex and highly inter-connected systems and enterprises.

Security

Due to the nature of the Repository used by Enterprise Architect and the possible deployment scenarios, security can be managed in a number of ways. In the Corporate edition of Enterprise Architect and above, there is built-in support for defining users and groups that permit locking and management of elements and Packages within the model. This will be discussed later in the section on Security. In addition, it is also possible to restrict general access to models by using file based security access for the MS JET and Firebird based models (.EAP and FDB) and DBMS server authentication and access for the larger database server based systems. As a further refinement it is also possible to restrict access to a model using the HTTPS protocol when connecting to a Cloud Based server.

Versioning

Enterprise Architect uses industry standard XMI (XML Interchange) format files to store and version model Packages. By integrating the import and export of XMI model fragments into a version control system (such as SVN) supported by Enterprise Architect, you can maintain multiple versions of the same model in different repositories and exercise fine control over the development and design process. Given the nature of this versioning system, there are several different scenarios that can be used to maintain different models and how teams access information. For example, where the versioning system is used as the single point of truth and de-facto Repository, it could be convenient to use many workgroup or single user based Repositories that reflect the actual model at a particular point in time. In contrast, where the versioned files are mainly held for backup and baseline purposes only, a larger, Team based DBMS server would be used to provide the model sharing and collaboration capabilities. So whatever the actual needs are, Enterprise Architect has a number of solution scenarios that can be adapted to provide the right level of change management and collaborative development.

Example Repository Tasks

Task

Create a project in a simple, file-based repository (.EAP and .FEAP files)
Create the project in one of a range of DBMS repositories - these provide for larger models with more concurrently connected users (Corporate and extended editions)
Manage change within the model
Maintain the integrity of the project data
Share the reference data used across the project, between models and between projects
Allow each user to configure their personal preferences for how project tools display and behave on their workstation
Document your model through document or web reports, which you can tailor to your purposes

Repository Overview

An Enterprise Architect repository is stored in a relational database; this can be a file based database such as MS Access or Firebird, or a database management system (DBMS) such as Oracle or MS SQL Server, on a network or in a Cloud Based Repository. It is common to start with a File Based Repository and, when the project gets some momentum and there are a number of modelers accessing the repository, to transfer it to a DBMS. Whichever type of repository is implemented, the user interface remains the same and users can seamlessly move from one type of repository to another and continue to contribute as a team member on this powerful modeling platform.

Concepts

Concept	Description
Repository as a File	In any version of Enterprise Architect you can store a project in a single file with a .eap or .feap extension. • A .eap file is a Microsoft JET database, so you can also open it using MS Access 97, 2000, 2003 or 2013, or any other reporting tool that can work with JET databases • A .feap file is a Firebird project file
Repository in a DBMS (Database Management System)	In the Enterprise Architect Corporate edition (and above) you can also use a suitable DBMS repository for projects. DBMS projects have the same logical structure as .eap and .feap files, but provide much greater scalability and concurrent access. You connect to a DBMS using ADO/ODBC (see *Connect to a Data Repository*). Please note that when configuring an ODBC data source on a 64 bit machine, you should access the ODBC data manager using the 'Configure > User Tools > ODBC Data Sources' ribbon option, as this will load the correct ODBC manager to access 32 bit data sources as used by Enterprise Architect.
Repository in the Cloud	In the Corporate Edition (and above) of Enterprise Architect you can further enable the distributed capabilities of Enterprise Architect and use the Internet or local network to connect to a Cloud Based server on which one or more Repositories have been installed and configured for access. The Cloud Server is free software available from Sparx Systems and is relatively simple to set up and configure on a suitable Windows based machine. Firebird files or any DBMS server are the required Repository types supported by the Cloud Server, and these can reside on the same machine for best performance, or on another machine to maximize local and network access. Cloud based repositories provide the ultimate in distributed development and minimal client configuration. It is in fact much easier to deploy a Firebird based repository on a Cloud Server for distributed development than to set up and manage a full DBMS solution.
Models	Enterprise Architect uses this terminology when discussing Repositories and their contents. 1. A Model is a related group of content that expresses some focused aspect of a

	system or enterprise that is related by the modeling tools and paradigm used. For example, you might build a 'Usecase Model' that describes behavior and scenarios using UML Use Cases. Process models, architectural models, behavioral models, structural models, Class models and so on are all Models within the current Project.
	2. A Project is a group of models that has a single unifying purpose. For example, a Project to design the navigation systems for a new aircraft or automobile would be considered a single Project. Within this Project there would be multiple models. Projects typically define levels of security, necessary types of views and models, necessary procedures and other management tasks to deliver the product. Project management capabilities in Enterprise Architect are quite rich and allow for resource management, security, change management, testing and other typical project management tasks.
	3. A Repository is a single File, DBMS database or Cloud Server address that contains one or more Projects. Typically, file based repositories such as MS JET and Firebird models contain a single Project with multiple models that can be shared by a small workgroup or only accessed by a single modeler. DBMS and Cloud based Repositories will often contain more than one Project, especially where the Projects are in some way inter-related or co-dependent.
Adding Initial Content	After creating and accessing your new project in whatever form the Repository takes, you can now use the Model Wizard to create Model Packages that range from generic and simple views to more complex and pattern based content. You can add Models to a project from the Project Browser by: • Right-clicking on an existing model and selecting the 'New Model' or 'Add a New Model using Wizard' options • Right-clicking on a Package and selecting the 'Add | Add a New Model using Wizard' option • Clicking on an existing model, pressing the Insert key and selecting the 'New Model' or 'Add a New Model using Wizard' menu options • Clicking on a Package, pressing the Insert key and selecting the 'Add a New Model using Wizard' menu option
Opening Existing Projects	Existing Projects are accessible via the 'Recent' list under the 'File Management' icon ( ) and on the Start Page, and through the 'Open Project' option under the 'File Management' icon. Connecting to an existing DBMS or Cloud based Repository is discussed in more detail under the relevant sections on working with those Repository types. For new users, you can use the 'Start > Help > Help > Open the Example Model' ribbon option to open and explore the Enterprise Architect Example Repository supplied with Enterprise Architect. This example is not so much a fully worked example as a collection of models that show how you can leverage the power of the Enterprise Architect platform and the UML to create a wide range of visual and text based models.
DBMS Repositories	You can connect to any of these data repositories: • MS Access 97, 2000, 2003 and 2013 (in all editions - .eap files are stored in Microsoft JET databases) • Access 2007 • SQL Server 2000 onwards • SQL Server Express 2005 onwards • MySQL • MariaDB

	• Oracle 9i, 10g, 11g and 12c
	• PostgreSQL
	• Adaptive Server Anywhere
	In brief, to create a new data repository, you first create a new database with the DBMS management software, then run supplied scripts to create the logical structure.
	You then use Enterprise Architect data transfer functions to move a project from a .eap, .feap or DBMS model into the new project. Details are available under the *Server Based Repositories* section.
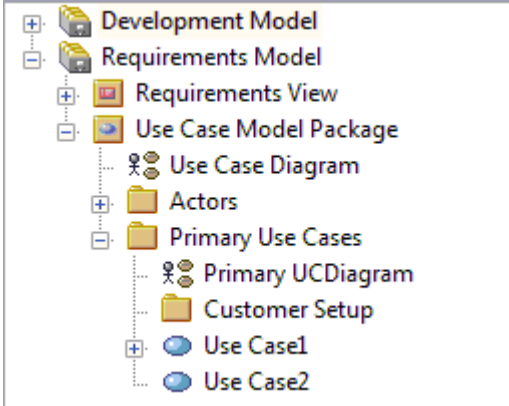
Projects Defined

An Enterprise Architect Project is a repository for storing, manipulating and managing one or more Models. A single repository can contain many models, and a Repository can be either file based, hosted in a larger DBMS system or based in the Cloud.

Concepts

Concept	Detail
Project	A project can contain a single model, or a number of models, each of which defines a particular system or process. A model contains the diagrams, elements, relationships and associated metadata that define the structure and function of the system or process. These components are organized into a hierarchy of Packages, which help to group and manage related components.
Model	Different aspects of the process or system - or their development - are defined by Model Packages, which you generate from templates specifically structured to support the aspects that the Model Packages represent, such as requirements or deployment. You can generate these templated Packages at any level of the hierarchy, but as they are created with their own content they are more useful at the top levels.
View	The top-level Packages in a model can also be Views, which represent partitions of the model that you define yourself. You can start with standard Views such as Class or Component, or create whatever partitions are appropriate to your model.
Example Project Structure	 Each View or Model Package contains Packages; Use Case Model Package contains: • Actors and • Primary Use Cases It also contains the diagram Use Case Diagram, which could be an overview of the Package structure or function. Each Package itself can contain one or more diagrams, one or more Packages, and several elements; the Primary Use Cases Package contains the: • Primary UCDiagram • Customer Setup Package

	• Use Case1 element
	• Use Case2 element
	Each subordinate Package also contains diagrams, elements and (if necessary) further Packages; the elements are related by connectors created in the diagrams, and each element and connector has properties, attributes, operations and extensions defined in the respective 'Properties' dialogs.
	Note that models do not have to be constructed this way. Depending on the methodology you are following and information you need to model, the project structure might look completely different. In addition, many architectural frameworks such as TOGAF and ArchiMate specify their own project structure as part of their overall solution architecture.

Opening a Project

An Enterprise Architect project is used for storing and managing the components of one or more UML models. The Desktop and Professional versions of Enterprise Architect work on file-based projects (.eap files). If you are using the Corporate edition (or above), you can also use one of a number of DBMSs such as Oracle or MySQL, or a Cloud Based server to host the project repository.

When you select to open a project, you can:

- Select a shortcut to a recent project, to open it
- Browse for another existing project to open
- Remove a shortcut from the list of recently-opened projects
- Specify connection details for opening a project hosted on a DBMS
- Connect to a project via the Cloud
- Create a new Enterprise Architect project file

Access

Ribbon	> Open Project
Keyboard Shortcuts	Ctrl+O

Options

Field/Button	Action
Local File	Click on this button to open a file browser dialog, and locate and select an existing .EAP or .FEAP file; the .EAP file can be a project file or a shortcut to a project hosted on a DBMS. Alternatively, click on the drop-down arrow and select the 'New Project' option to create a new project.
Connect to Server	Click on this button to specify connection details for opening an Enterprise Architect project that is hosted on a DBMS. The 'Windows Data Link Properties' dialog displays; start to define the connection to the project's server-based repository. You can also click on the drop-down arrow and select: • 'Connection Wizard' - the 'Data Link Properties' dialog again displays • 'Connection String' - the 'Connection String' dialog displays; type in or paste the connection string and click on the OK button to connect directly to the project
Connect to Cloud	Click on this button to connect to a project through the Cloud. The 'Cloud Connection' dialog displays, on which you specify the server, URL and model name to access via the Cloud.
Recent Projects	This panel lists the most recently opened projects, up to a maximum of ten. To open one of these projects either double-click on the name, or click on it and then click

	on the Open button.
	Each project name is shown with its access path. You can edit the access path by right-clicking on it and selecting the 'Edit Connection String' context menu option; make the changes on the 'Connection String' dialog, and click on the OK button. This list is also displayed: • On the Start Page, under the 'Recent' heading • As a drop-down menu under the Open Project button on the Default Tools toolbar • With the list of options
Remove Selection from List	Click on a project name and click on this button to remove that project name from the 'Recent' list.
Show this Dialog at Startup	Select the checkbox to show this 'Open Enterprise Architect Project' dialog automatically upon starting Enterprise Architect. Deselect the checkbox to hide the dialog.
Default Project to Open	Click on a project name in the 'Recent' list and select this checkbox to open the selected project automatically whenever you start up Enterprise Architect.
Open	Click on this button to open the currently selected project in the Recent Project list.
Cancel	Click on this button to close this dialog without any further action.
Help	Click on this button to display this Help topic.

Notes

- Use of a DBMS and/or Cloud for hosting the model repository is available in the Corporate, Business and Software Engineering, Systems Engineering and Ultimate editions of Enterprise Architect

- Enterprise Architect supports these DBMS products for hosting model repositories:
    - SQL Server and SQL Server Express;
    - MySQL
    - Oracle 9i, 10g, 11g or 12c
    - Postgre SQL and
    - ASA

Project Shortcuts

Enterprise Architect enables you to create a desktop shortcut (or proxy file) to an Enterprise Architect project (Cloud, DBMS or file-based). Each shortcut is a file containing the connection string for the model. In addition, the shortcut also defines views that Enterprise Architect should open when it opens the model, as outlined here. In this manner it is a convenient and customizable way of quickly sharing a connection to a Cloud based model without requiring other users to know the details of the connections string, model name and other parameters.

Access

Ribbon	> Save As Shortcut

Topics

View	Detail
Diagrams	Define one or more specific diagram(s) to be loaded on opening the project.
Model Search	Open the Model Search with a specific text string and search type.
	For searches operating on the current tree selection, a diagram in the target Package must be opened first.
	If you use a custom SQL search, the SQL must include ea_guid AS CLASSGUID and the object type.
	You specify a single Model Search to open.
The Relationship Matrix with a saved profile	Open the Relationship Matrix with a saved profile.
	You specify a single Relationship Matrix profile to open.
The default Team Review	Open the default Team Review document.
	You specify the Team Review once.
Working Set	Open a working set.
	You specify a single Working Set to open.
	This is very similar to the shortcut itself, opening a defined set of diagrams and views. However, the working set can also open source code editors, therefore widening the capabilities of the shortcut alone.
	Working sets make it easy to customize the main views you want to open without having to resave your project short cuts. The working set is easy to tailor to your changing requirements, adding greater flexibility to any short cut that calls the working set.
	Also, working sets provide the capability of storing the currently-engaged files when closing a model, and reopening them in the context in which you were last using them. The basic project shortcut can also do this, but once the work environment is captured the shortcut returns to the same set up each time you use it. The working set always captures the current work environment each time the model

	closes.
	This option can be very useful in, for example, a coding environment when you want to return to the last files you were editing.
Workspace Layout	Apply a selected Workspace Layout, which opens and sets out the appropriate screens and windows for a specific area of work, such as model simulation.
	You specify a single Workspace Layout to open. However, by creating and using separate shortcuts you can open Enterprise Architect in exactly the configuration you need for the work you want to do - modeling, coding, or debugging, for example.
An example shortcut	You might create a shortcut to open, in sequence:
	• A Development module
	• The Model Search for a simple search on the term Issue
	• The module Issues diagram
	• The module 'Changes' diagram
	Enterprise Architect opens the appropriate windows in the sequence in which you list the options, displaying the last view in the list as the active view.
	In this example, the project opens with the Enterprise Architect work area showing the two diagram tabs and the 'Model Search' tab, and with the 'Changes' diagram displayed in the Diagram View.

Notes

- If specified, the shortcut views override any default diagram defined for the model or current user

- A shortcut does not affect the original Enterprise Architect .exe file or icon, or any other shortcut you might have defined; you can use all of these independently

- If you are using a database repository other than MS Access 97, 2000, 2003 or 2013, you can configure the shortcut to encrypt the password used to set up the connection between Enterprise Architect and the repository; the Enterprise Architect user does not have the real password, thereby preventing them from accessing the repository using other tools such as Query Analyzer or SQLPlus

Create Project Shortcut

You can create a shortcut to an Enterprise Architect project (either a DBMS project or file based project). The shortcut can specify additional windows and diagrams to open up automatically every time the shortcut is run, to create a working environment in advance for other users.

Shortcuts are stored with a .EAP extension, but are actually small text files that tell Enterprise Architect what project to open and what initial views and windows to display.

Access

Ribbon	: Save as Shortcut

Create a project shortcut

Step	Action
1	Open Enterprise Architect.
2	Open the required project.
3	Select the ' |Save as Shortcut' option. The 'Save Project Shortcut' dialog displays.
4	Click on the \[...\] button at the end of the 'Target File' field. The 'Save Project As' dialog displays.
5	Browse for the appropriate file location and, in the 'File name' field, type an appropriate filename. All shortcuts are .EAP files, regardless of whether the model itself is a .EAP file or a DBMS model.
6	Click on the Save button to return to the 'Save Project Shortcut' dialog.
7	Click on the Add Other button and select the required option to define: • A diagram to open • A Relationship Matrix profile to open • The Team Review • A Model Search to perform • A working set to apply • A workspace layout to apply
8	The appropriate browser or dialog displays to define the view to display; enter the details and click on the OK button. The view is added to the 'Actions when model is opened' field; the entry is automatically selected, with a

	tick in the checkbox.
9	Repeat steps 7 and 8 for as many additional views as you require.
10	Review the items in the 'Actions when model is opened' field. • If you decide not to have an item in the shortcut, deselect its checkbox • If you want to clear all selected items, click on the Include None button Unselected entries are deleted when you save the shortcut.
11	If you decide to change the sequence and/or make a different view display first in the Diagram View: • Click on the appropriate entry • Click on the 'Up Hand' or 'Down Hand' buttons
12	Click on the OK button to save the shortcut.

Notes

- When you subsequently open the 'Save Project Shortcut' dialog, it lists the currently-opened views in the order in which they were opened; you can add further views or remove them from the shortcut

Capture Current Work Environment

You can capture the current Enterprise Architect work environment in your shortcut. This maintains the work environment from that point, and is useful if you intend to close and re-open the model and return to the same configuration many times.

If you simply want to return to whatever work environment you happen to be in each time you close the model, you should use a work set that captures the current environment, either by itself or as part of a broader project shortcut.

Access

Ribbon	: Save as Shortcut

Capture your current work environment

Step	Action
1	Open Enterprise Architect.
2	Open the required project and work in it.
3	At the point at which you decide to capture your work environment in a shortcut, ensure that: • You have opened all diagrams you require • If necessary, you have opened the Team Review, Model Search (with appropriate search term and type) and/or Relationship Matrix (at the appropriate profile) • The view you want to resume work on is the last one opened
4	Select the 'Save As Shortcut' menu option. The 'Save Project Shortcut' dialog displays, showing a list of actions derived from the views you currently have open.
5	If you accessed Enterprise Architect via a shortcut, the 'Target File' field displays the file location of that shortcut. Otherwise, click on the [...] button at the end of the 'Target File' field. The 'Save Project As' dialog displays.
6	Browse for the appropriate file location and, in the 'File name' field, type an appropriate filename. All shortcuts are .EAP files, regardless of whether the model itself is a .EAP file, .FEAP file or DBMS model.
7	Click on the Save button to return to the 'Save Project Shortcut' dialog.
8	In the 'Actions when model is opened' field, click on the Include All button.
9	If you also want to save current window positions click 'Add Other' followed by 'Add Workspace Layout'.

		The dialog shown allows you to select an existing layout or save the current layout.
10		Click on the OK button to save the shortcut.

Encrypt Repository Password

If your model is developed on a DBMS repository, the 'Save Project Shortcut' dialog has an 'Encrypt Connection String' check box.

You can create the shortcut actions and, if necessary, select the checkbox to encrypt the database connection string.

You distribute the shortcut file to the database users who are to access the model. The users then have an encrypted string that prevents them from directly accessing the database using other tools.

File Based Projects
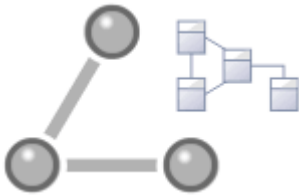
Enterprise Architect provides a convenient light-weight, low-barrier and portable file based repository out-of-the-box, which is ideal for modelers who want to start work immediately; with the file based repository you can be modeling within a few minutes. When a number of colleagues want to contribute to the models, the file can be moved to a file-share and it is possible to work with this setup even when the model becomes quite large and the number of users increases to ten or more. All editions of Enterprise Architect support this type of repository in the form of a .eap file in an Access database, or a .feap file in a Firebird repository.

You can create your own repository from scratch, but a default repository (EABase.eap) is conveniently supplied, which is located in the Enterprise Architect installation directory. The file acts as a template for creating new projects and can be copied to give you a head start in creating a repository. You can tailor this repository to suit your organization or create your own base repository. A File Based Repository can be easily transferred to a DBMS based repository using the Project Transfer feature, allowing a project to be scaled up when the team increases in size or organizational policy dictates it.

File-based project tasks

Task	Detail
Create .EAP Project Files	Select one of: • > New Project menu option • 'Local File | New Project' option on the 'Manage Projects' dialog, or • 'New File' option on the Start Page All of these options display the 'New Project' dialog; select a directory and enter a file name for your project. Once the project has been saved, the Model Wizard displays, which makes a selection of Model Patterns available; select the Model Patterns to use. Enterprise Architect adds a model containing the selected Model Packages to the Project Browser.
Create .FEAP Project Files	As for a .eap file, except that in the 'New Project' dialog, in the 'Save as Type' field, click on the drop-down arrow and select the .feap file extension. Enterprise Architect will check your input and display an error message if: • Your Firebird repository is not located on the local drive; a Firebird repository is not appropriate for a network shared project • The file path contains characters that are not in the system codepage
Location of default (EABase.eap) repository template	The default installation directories, depending on which version you have installed, are: • Registered version: C:\Program Files\Sparx Systems\EA • Trial version: C:\Program Files\Sparx Systems\EA Trial • Lite version: C:\Program Files\Sparx Systems\EA Lite A base project contains templates and reference data from which you can quickly

	develop your own project.
	Note that in addition to deriving a new model from EABase.eap. it is also possible to start a new project by copying an existing one. While it is possible to simply 'copy' the .EAP file at the file system level, this results in two identical models and should only be used when that is the required behavior; for example, when distributing a model to a client for verification. If you use the built in functions of Enterprise Architect to create a new model based on an existing one, Enterprise Architect will modify the unique identifiers (GUIDS) for all elements and related constructs so that the new model is essentially unique, and not a simple copy of the existing one.
Configure Project	Having created your project, you can set a range of project parameters to define defaults, tailor the project to particular coding languages, and ensure consistent development and use of the project.
Create Custom Templates	You can customize any Enterprise Architect project as a template project with company standards, tutorials, frameworks and any other common piece of modeling already in-built; with careful planning you can save yourself many hours of work at project start-up.

Best Practices for File Based Repositories

Single User	Simple file based repositories (.eap and .feap files) are best suited to single user development with the model file placed on the modelers local drive. This is fast, very efficient and provides a powerful means of getting started in modeling a particular solution. Many models will start out as single user repositories and only later be migrated into a DBMS or Cloud based scenario for larger team access. If you need to quickly start work on a project and do some exploratory work, or you are working alone as a consultant, the .eap or .feap file format is ideal.
Small Workgroup	With the Professional version of Enterprise Architect and above it is possible to share a .eap file at the network file level. This is really only a suitable solution for small workgroups (up to, say, 5 members) who are collaborating on a shared model. This solution is very much at the mercy of the quality of the network connection and could be less than ideal if collaborating users increase in number and/or the model grows very large. As Enterprise Architect on the user's machine is still running all the queries and performing all database operations on a network file, a slow network will result in less than ideal performance. As team size and model size increases it is best to upsize to either a DBMS or Cloud based architecture.
In Conjunction with Version Control	One additional scenario in which file based repositories can be used to great effect is with a version control system such as SVN. If the SVN system holds the master content, then individual developers/modelers can check out the material of interest to them into a local .eap or .feap file. When they have made their changes they can migrate their work back into the shared master. This style of development is quite effective, and as the import and export of model material from the version control system works best with single user local project files, it is a good match and a good use of the two technologies.
Copy of Repository for Storage or Review	A further use of the single file repository is for distribution to clients and others for review and comment, where the original model(s) is located in a DBMS or Cloud that the recipient would not have access to. As it is relatively easy to bulk copy a DBMS model to a local .eap file, this is a good method for passing on a model to a

	client or another interested party. If some of the material is restricted, it is possible to delete sections from this single file repository after the bulk copy is complete and before passing it on.
Replication	If network access is limited and DBMS and Cloud based solutions are not possible, another (less than optimal but still effective) team based solution is to use the built in replication features of the .EAP file format. By creating a design master and distributing replicas to other modelers, it is possible to effectively share and contribute to a model without having direct shared access to the common model. On the down side, someone will have to perform the model merges as required and if there are replication conflicts (two changes to the same item) this will require manual resolution.

Notes

- You can also add Model Packages to a project by clicking on the Project Browser header bar and selecting the 'New Model From Pattern' menu option

Copy a Base Project

When you create a new project, you can use the Model Wizard to define the structure and contents. Alternatively, you can copy an existing template or base project which is already largely set up, containing company standards, tutorials, frameworks and any other common modeling structures. You can also copy an operational .EAP or .FEAP project file to a new file location under a new name to, for example:

- Provide separate copies for individual team members
- Create an evaluation or distribution version of the project

If you intend to use an existing project file as the template for a new project, it is important to use this method rather than simply copying the .EAP file using Windows Explorer. This process provides the option to reset all the unique identifiers for Packages and elements, so that your new project is truly unique - otherwise you create an exact copy of the original project.

Access

Open the template project or existing operational project then;

Ribbon	: Save Project As

Create a new project from a base project

Option	Action
Target Project	Type the new project filename.
	You can either type in the file path as well, or click on the Browse button and browse for the required file path.
	If this is to be a shared project, store the file on a shared network resource such as a Network Server or Workgroup Server.
Source Project	Defaults to the path and name of the currently-open template or project.
Reset New Project GUIDs	If the target project is to be a different project from the source (such as when you copy a base template), select this checkbox to replace all GUIDs from the base model with fresh GUIDs in the new model.
	However, if the new project is to be a direct copy of the source, or based on a project that is under version control, we recommend that you deselect this checkbox to prevent duplication of Packages when the Get Latest facility is used.
Save As	Click on this button to save and open the new project.

Sharing File Based Projects

Enterprise Architect offers a diverse range of functionality designed specifically for sharing projects in team-based and distributed development environments, through network deployment of model repositories, DBMS based repositories, Cloud based models, replication and XMI Import/Export.

Features

Feature	Detail
Network Deployment	Network deployment is possible under three different schemas: • File based repositories • DBMS server based repositories, or • Cloud based deployment (recommended) DBMS server based repositories offer better: • Response times than files on networks, due to the inherent structure of the DBMS • Solutions when networking problems are encountered, as they have the ability to backtrack transactions caused by external breakdowns
Replication	Replication enables data interchange between .EAP based repositories and is suitable for where many different users work independently in parallel development. Modelers merge their changes into a Design Master only as required; it is recommended that a backup is carried out prior to replication. Replication cannot be performed on repositories stored on a DBMS server.
XMI Import and Export	XMI Import/Export can be used to export and share discrete Packages between developers; XMI enables the export of Packages into XML files which can then be imported into any model. Package control can be used to set up Packages for version control and to enable batch export of Packages using XMI; version control enables a repository to be maintained by a third-party source code control application that is used to control access and record revisions.

Notes

• DBMS Repository support is available with the Corporate, Business and Software Engineering, System Engineering and Ultimate editions of Enterprise Architect

Share Enterprise Architect Projects

The most efficient way of using Enterprise Architect to manage a team development is to share a project amongst a team of designers, developers and analysts.

Facilities

Facility	Detail
By Sharing a project	• Many people can work on the model at the same time and contribute their particular skill • Team members can always see what the latest changes are, keeping the team informed and up to date with the project status
You can share a project in three ways	• Place the project in a shared network directory • Use replication • Use a shared DBMS-based repository

Notes

• Project Sharing and Replication are available in the Professional, Corporate, Business and Software Engineering, System Engineering and Ultimate editions of Enterprise Architect

• DBMS repositories are supported in the Corporate, Business and Software Engineering, System Engineering and Ultimate editions of Enterprise Architect

Refresh View of Shared Project

When a user of a shared model checks out a Package and makes changes, other users can see those changes by refreshing their view of the Package or the changed diagram within the Package, in a number of ways.

Refresh the view

Object	Options
Project	• Right-click on the Package name in the Project Browser and select the 'Contents | Reload Current Package' option, or • Select the ![icon] (Reload Project) icon in the Project Tools toolbar, or • Press Ctrl+Shift+F11, or • Close the project and reopen it
Diagram	• Select the 'Start > Workspace > Views > Reload View' ribbon option, or • Right-click on the opened diagram tab in the Diagram View, and select the 'Reload <diagram name>' option

Share a Project

The easiest way to share a project amongst a work group of developers and analysts is to place the project file on a shared network drive, to which people connect from their workstations.

Individual developers and analysts can then open and work on the project concurrently.

Network Issues

Enterprise Architect accepts a number of concurrent connections without issue; however, there are points you should consider:

Issue	Description
Lock Outs	There can be occasional 'lock-outs' when one user tries to access or update something another user is in the process of modifying.
Other Changes	Changes to the Project Browser (and other project views) are not automatically updated; to compensate for this, users must occasionally reload their project to see changes made by other users.
Diagrams	If two or more people work on the same diagram concurrently, unexpected results can occur; it is best to allow only one analyst to work on a diagram at a time.
System crashes	(.EAP files only) If a user's machine crashes, the network suffers an outage or a machine is turned off unexpectedly, the project file might require repair to compensate for the sudden inconsistency; a repair facility is provided ('Configure > Model > Check Integrity > Manage .EAP File > Repair .EAP File') to carry out this task.

Notes

- Firebird-based projects (.feap files) are not suitable for sharing over a network

Distributed Development

Enterprise Architect supports distributed development using two different techniques.

Replication

Using replication, geographically separated analysts can update and modify parts of the model in replicas, then merge these back together at a central location.

XMI Import/Export

Using XMI-based import/export, you can export discrete Packages to XML to share among the development team; this has several benefits over replication:

- You can assemble a model from only the parts necessary to get your job done

- You can assemble a full model if required

- You can assemble a model from different Package versions for different purposes (such as customer visible, internal release only)

- You can roll-back parts of a model as required

- There is less chance of 'collisions' between developers if each works on a discrete Package

- The process is controllable using a version control system, or through Package control

XMI based import/export is UML1.3 / XMI1.1 compliant; you can also write XML based tools to manipulate and extract information from XML files to enhance the development process.

XMI-based Import/Export is accessed through the 'Publish > Model Exchange > Export XMI' and 'Import XMI' ribbon options.

Replication

Apart from sharing Enterprise Architect projects in real time over a network, you can also share projects using replication.

Guide

Facility	Detail
Replication	Replication is a powerful means of sharing projects between isolated or mobile users; it enables different users to work independently of one another, and to merge their changes at a later time. In Replication: • A project is converted to a design master, then replicas are made of the master • Users take the replicas away, modify the project, then bring their replicas back to be synchronized with the master file
Use Replication	To use replication: 1. Convert the base project to a design master. 2. Create replicas from the design master. 3. Take the replica away and work on it as required, then bring it back for synchronization with the design master. 4. Synchronize the replicas, during which all changes to both the master and the replica are propagated in both directions, so they both finally contain the same information.
Enterprise Architect Merge Rules	Enterprise Architect follows these rules in merging: • Additions are cumulative; that is, two replicas each creating three new Classes result in six new Classes after merging • Deletions prevail over modifications; if one replica changes a Class name and other deletes the Class, merging replicas results in both files losing the Class Conflicting modifications appear in the 'Resolve Replication Conflicts' dialog.
Upgrades and Replicas	When you upgrade your version of Enterprise Architect, you must not open a replica until you have opened the design master and then synchronized the replicas with the master; you cannot directly upgrade a replica.
Avoid Change Collisions	If two or more people make changes to the same element, Enterprise Architect arbitrarily overwrites one person's change with the other's; to avoid this, different users should work on different Packages. However, since Enterprise Architect does not enforce this rule, it is possible for users' work to conflict; to minimize difficulties, please note these guidelines: • If users are likely to have worked in the same area of the model, they should both witness the synchronization and confirm that they are happy with the net result • If small pieces of information have been lost, they should be typed into one of the merged models after synchronization • If a large piece of information has been lost (for example, an overwritten large Class note) use the 'Resolve Replication Conflicts' dialog

| Disable or Remove Replication Features | If you have converted a project to a design master but now want to disable the replication features, you can remove Replication; ensure that you back up all your files first. |

Design Masters

A design master is the first converted Enterprise Architect project that supports replication. You create the master project from which you create replicas that can be modified independently of the master project and re-merged later.

Access

Ribbon	Configure > Model > Check Integrity > Manage .EAP File > Make Design Master

Create a design master

Step	Action
1	Take a back-up of the required Enterprise Architect project.
2	Select the project in the Project Browser.
3	Select the 'Make Design Master' menu option and follow the on-screen instructions.

Create Replicas

A replica is one of several copies of the design master of an Enterprise Architect project. You create the copy of the master project for you or another user to modify independently and re-merge later.

Access

Ribbon	Configure > Model > Check Integrity > Manage .EAP File > Create New Replica

Create a replica

Step	Action
1	First create a design master, then select the 'Create New Replica' menu option and follow the on-screen instructions.
2	Edit the replica over time and, when required, return the file for merging with the design master.

Notes

- In the Corporate, Business and Software Engineering, System Engineering and Ultimate editions of Enterprise Architect, if security is enabled you must have Manage Replicas permission to create a replica

Synchronize Replicas

Synchronizing replicas combines the changes made to each file. You can:

- Merge the changes made to each replica with the design master, so that a new set of replicas with all changes can be generated and distributed
- Combine the changes made to two replicas, should it be necessary for two team members to combine their work

Access

Ribbon	Configure > Model > Check Integrity > Manage .EAP File > Synchronize Replicas

Synchronize replicas

Step	Action
1	Open the design master project file (or the first required replica).
2	Select the 'Synchronize Replicas' menu option.
3	Locate and select the (second) required replica to merge the open project and the replica.

Notes

- Information is copied both ways, including deletes, updates and inserts; both projects end up containing identical information
- If this process generates 'conflicting changes' errors, you should review and, if necessary, resolve these conflicts
- In the Corporate, Business and Software Engineering, System Engineering and Ultimate editions of Enterprise Architect, if security is enabled you must have Manage Replicas permission to create a replica

Remove Replication

Replication makes many changes to the database structure of your model, so the model file becomes considerably larger with additional information; you might, therefore, decide not to use the replication feature any more.

Access

Ribbon	Configure > Model > Check Integrity > Manage .EAP File > Remove Replication

Remove replication from your model

Step	Action
1	Ensure that you have open a temporary repository (not the one having replication removed) - the menu option is not available if no repository is open.
2	Select the 'Remove Replication' menu option. The 'Remove Replication Wizard' dialog displays.
3	Enter the full path and file name of the project to have replication removed. Click on the Next button. If the project displays Version Number 7.0 this indicates that Auditing is enabled. Auditing should be disabled before proceeding (see *Notes*).
4	Enter the full path and file name of the base Enterprise Architect model (with no replication) to act as template. Click on the Next button.
5	Enter the full path and required file name for the output file. Click on the Next button.
6	Select whether to have a log file created and, if so, enter a file name for the log file.
7	Click on the Run button to begin removing replication. Enterprise Architect creates a new project containing all the model information. Your model has now had replication removed, and should be considerably smaller.

Notes

- You cannot remove replication from a model with Auditing enabled - if you want to remove replication:

1. Disable Auditing

2.   If prompted to do so, allow Enterprise Architect to roll back the database version

3.   Remove replication

Upgrade Replicas

A new release of Enterprise Architect might contain changes to the underlying project structure, such as more tables or changed queries. If you use Replication, you must take care with your upgrade.

Considerations

Consideration	Detail
Open Design Master Project first	After installing the new version of Enterprise Architect, it is very important that you open the design master before opening any of the replicas with the updated version.
Changes to the database design	Changes to the database design in a replicated project can ONLY be done to the design master; trying to update a replica at best does nothing, and at worst causes the update of the master to fail.
Propagate Changes	Design changes are propagated through to the replicas the next time the replicas are synchronized with the master.
Alternatively	One other strategy is to remove replication from a copy of the replica set, upgrade that project and convert it into a new design master from which new replicas are created.

Resolve Conflicts

If two or more people each work on the same model object in their respective replicas between synchronizations, the replication engine has problems in resolving which change is the master. You need to select which of two conflicting changes you should save to the Design Master and/or replicas, where a substantial piece of information has been overridden by a user and you want to retrieve it.

Access

Ribbon	Configure > Model > Check Integrity > Manage .EAP File > Resolve Replication Conflicts

Considerations

Consideration	Detail
Avoid the problem	Ensure that each team member always works in a separate area of the model within their replica.
Check for conflicts	After synchronizing replicas, open the 'Resolve Conflicts' dialog (see next table) and check if there were any conflicts.
Response to conflicts	When a project record has been modified in different ways by different users, the replication engine selects one of the conflicting values based on rules within the JET replication manager.
	However, the replication engine stores the discarded changes and flags the conflict on the 'Resolve Conflicts' dialog so that you can choose to roll in the discarded change instead.
	Normally it is not necessary or desirable to examine conflicts, since they represent relatively inconsequential pieces of information that can very easily be modified through the normal Enterprise Architect interface; for example, by moving a diagram element.
	The only case in which the 'Resolve Conflicts' dialog should be used is where a substantial piece of information has been overridden by a user, and you want to retrieve it.

Resolve conflicts in changes to the same model object in two separate replicas

Step	Action
1	Synchronize a replica with the Design Master, and display the 'Resolve Conflicts' dialog.
2	In the Table with Conflicts list, click on the entry that is likely to contain the lost information.

3	Click on each entry in the Conflicting Records list. When the lost information appears in the Conflict Details list, click on the Overwrite with Conflict button.
4	In the 'Conflicting Records' list, you can also copy each Row ID to the clipboard by right-clicking on the row and selecting the 'Copy to Clipboard' menu option. You can locate the object that is in conflict in the Project Browser or diagrams, by right-clicking on it and selecting the 'Find in Project Browser' or 'Find in Diagrams' option.

Server Based Repositories

As a repository becomes larger, or the number of concurrent users increases or organizational policy dictates, it might be more appropriate to use a database management system (DBMS) to store the repository. A File Based Repository can be created and used in any edition of Enterprise Architect; however, if you decide to use a DBMS based repository you will need to use the Corporate, Business and Software Engineering, Systems Engineering or Ultimate editions. Enterprise Architect has a function to transfer your repository from a file based repository to a server based one, helping you to get started quickly and with no changes in the user-interface.

Note that the performance of the Repository as experienced by end users will depend very much on the quality and power of the server machine and the network infrastructure on which the DBMS and user are located. Using a DBMS over a very high latency (10ms or higher) network can result in significant delays and visibly inferior performance. When network latency is an issue, Sparx Systems recommends using a Cloud Based Server as the interactions are optimized to reduce the effect of network latency.

Also it is important to note that all models are quite different and although Sparx Systems does its best to maximize performance based on what is expected to be held in a repository, sometimes this is not quite sufficient. In these rare cases a review of the database indexes would be good practice to maximize data retrieval and access. This will ensure that end users receive the best possible performance even when models contain millions of constructs.

Set up a Project on a DBMS repository

To set up your project on a DBMS repository, you work through these stages:

1. Set up your DBMS software and create a repository.

2. Create the required tables in your repository, by running a script supplied on the Sparx Systems website.

3. (For certain DBMS products) Set up an ODBC driver to enable connection to the repository.

4. Transfer the project from the source file to the DBMS repository; the source file can be:
    - a .EAP or .FEAP base model, to begin a new project from scratch, or
    - a previously-developed project file, to move an existing project into the DBMS repository

5. Connect to your repository.

Notes

- You cannot move a model from a source .eap file of an Enterprise Architect version earlier than 3.5.0 without updating it first

- Before proceeding, you must have MDAC 2.6 or higher installed on your system

- (Optional, but recommended) before actually transferring the project structure from the file to the repository, perform a Project Data Integrity Check on the file

Create a Project in a MariaDB Database

To create a project in a repository on MariaDB, you work through these stages:

- Stage 1: Create an empty database repository and set up the data tables
- Stage 2: Set up the ODBC Driver
- Stage 3: Perform a project integrity check on the project file you are using as a base (optional, but recommended)
- Stage 4: Transfer the data
- Stage 5: Connect to the repository and open the project

Prerequisites

- A machine with MariaDB version 10.1.2 or higher installed and running,
- MariaDB ODBC

Create a MariaDB Repository

Create the repository as outlined here.

Step	Action
1	Create a new empty database and configure it to your installation's defaults.
2	Load the EASchema_1220_MySQL.sql file from the Sparx Systems website into your SQL management console of choice (such as HeidiSQL).
3	Run the script to create the required database schema.

Set Up the ODBC DSN

Step	Action
1	Create a suitable ODBC Data Source to point to your new database. Select the extended option: • 'Return matched rows instead of affected rows'

Perform a Project Integrity Check

Step	Action
1	In Enterprise Architect, open the file-based project or template from which you are creating the project on the repository.

2	Select and execute the 'Configure > Model > Check Integrity > Project Integrity' ribbon option.
	This ensures your project data is 'clean' and free from errors before being copied to the repository.

Transfer the Project Data

Step	Action
1	Open Enterprise Architect.
	(If the 'Open Project' dialog displays, click on the Cancel button to open with no project loaded.)
2	Select the 'Configure > Model > Transfer > Project Transfer' ribbon option.
	The 'Project Transfer' dialog displays.
3	In the 'Transfer Type' panel, select 'File to DBMS'.
4	In the 'Source Project' field, click on the [...] button and browse for the name of the project file to copy to the repository.
	If the .EAP file has Replication enabled, this must be removed before performing the transfer.
5	At the right of the 'Target Project' field, click on the [...] button.
	The 'Datalink Properties' dialog displays.
6	Select 'Microsoft OLE DB Provider for ODBC Drivers' from the list.
	Click on the Next button.
7	In the 'Use Data source name' field, click on the drop-down arrow and select the ODBC Data Source you configured to point to your new database.
	Click on the OK button. The 'Project Transfer' dialog redisplays.
8	If required, select the 'Logfile' checkbox and type a path and filename for the data transfer log file.
9	Click on the Transfer button to begin the data transfer process.
10	When the process is complete, you have created a project on a MariaDB database and can now connect to it and open it from Enterprise Architect.

Create a MariaDB Repository

Creating a MariaDB database is fully documented in the product information provided with your MariaDB installer.

Having created your MariaDB database, use the SQL script provided by Sparx Systems to create the required table definitions for Enterprise Architect.

You can obtain the scripts from the Sparx Systems website, on the:

- Registered Corporate edition 'Resources' page (Registered users)
- Corporate edition 'Resources' page (Trial users)

Third Party Tools

If you are unfamiliar with MariaDB and DBMS systems in general, you might want to consider a suitable front end tool. HeidiSQL is installed with MariaDB, providing a convenient graphical user interface to enable the creation of databases, the execution of scripts, and backups and restores.

You might, therefore:

1. Run HeidiSQL and create a new database

After creating a MariaDB data repository in Enterprise Architect, you must set up the MariaDB ODBC driver.

Notes

- This feature is available in the Corporate, Business and Software Engineering, System Engineering and Ultimate editions
- Ensure that the collation is set to the alphabet you use, such as Latin or Cyrillic

Set up a MariaDB ODBC Driver

After you have created a repository on your MariaDB database, you set up the MariaDB ODBC driver so that you can connect to a project on that repository from Enterprise Architect.

Prerequisites

Install:

- MariaDB DBMS and repository
- MariaDB ODBC driver software

Set up the ODBC Driver

Your MariaDB driver is now available to connect to the repository from Enterprise Architect.

Step	Action
1	Under both 32-bit and 64-bit operating systems, Enterprise Architect requires 32-bit ODBC drivers to connect to a repository through ODBC. You can quickly load the correct 32-bit ODBC Data Source Administrator by selecting the 'Configure > User Tools > ODBC Data Sources' ribbon option. The ODBC Data Source Administrator window displays.
2	Click on the Add button. The 'Create New Data Source' dialog displays, enabling you to add a new DSN.
3	Select the appropriate MariaDB ODBC Driver from the list.
4	Click on the Finish button. The 'MariaDB Connector/ODBC' dialog displays.
5	Enter these configuration details: • A data source name for the connection • A description (optional) • The host address of the DBMS server • User name and password • The database name on the selected server
6	To set the advanced options, click on the Details>> button.
7	Select these checkboxes (where provided): • Return matched rows instead of affected rows ('Connection' or 'Cursors/Results' tab) • Allow big result sets ('Connection' tab)
8	Click on the Test Connection button to confirm that the details are correct.
9	If the test succeeds, click on the OK button to complete the configuration. If the test does not succeed, review your settings.

Connect to a MariaDB Data Repository

To access a project in your MariaDB data repository, you must connect to the data repository from Enterprise Architect.

Prerequisites

- The MariaDB repository and the project already exist
- You have SELECT, INSERT, UPDATE, DELETE, EXECUTE and SHOW VIEW access permissions
- The MariaDB ODBC driver has been set up

Access

Ribbon	: Open Project
Keyboard Shortcuts	Ctrl+O

Connect to the repository

Step	Action
1	In the 'Open Project' dialog, select the 'Connect to Server' checkbox.
2	Click on the [...] button. The 'Data Link Properties' dialog displays.
3	Select 'Microsoft OLE DB Provider for ODBC Drivers' from the list.
4	Click on the Next>> button. The 'Connection' tab displays.
5	Click on the 'Use data source name' radio button and on the drop-down arrow in its field. From the list, select the ODBC driver you have set up to connect to your MariaDB repository.
6	If required, type in a User name and Password.
7	If required, type in an initial catalog.
8	Click on the Test Connection button to confirm that the details are correct.
9	If the test does not succeed, revise your settings. If the test succeeds, click on the OK button; the 'Connection Name & Type' dialog displays.

10	Give the connection a suitable name so that you can recognize it in the 'Recent Projects' panel on the 'Open Project' dialog.
11	If required, select the 'Encrypt Connection String' checkbox.
	This encrypts and hides the connection details of the database from the users that the connection string is given to.
12	If required, select the 'Lazy Load' checkbox to not load the full project view when the model is loaded; instead, only the parts that are necessary to display the visible portion of the tree are loaded.
	With this setting, the model loads faster and users can begin work sooner, but at the expense of later small delays as Enterprise Architect loads specific portions of the model.
13	If required, select the 'Use WAN Optimization' checkbox.
	To improve performance over a Wide Area Network, remote database calls can be routed through a WAN Optimizer that compresses the data returned from the repository, reducing transfer time.
	If you select this checkbox, complete the next three fields (see your administrator for the correct values); otherwise go to step 17.
14	In the 'Server' field, type the network name or address of the optimizer server.
15	In the 'Port' field, type the port on which the server is running on the remote machine.
16	In the 'DSN' field, type the data source name of the database as it appears on the remote machine.
17	Click on the OK button to complete the configuration and open the project.
	This also adds the project name to the 'Recent' list on the 'Start Page'; from now on, you can open the project again just by clicking on this name.

Notes

- This feature is available in the Corporate, Business and Software Engineering, System Engineering and Ultimate editions of Enterprise Architect

Create a Project in a MySQL Database

To create a project in a repository on MySQL 5, you work through these stages:

- Stage 1: Create an empty database repository and set up the data tables
- Stage 2: Set up the ODBC Driver
- Stage 3: Perform a project integrity check on the project file you are using as a base (optional, but recommended)
- Stage 4: Transfer the data
- Stage 5: Connect to the repository and open the project

Prerequisites

- A machine with MySQL version 5 or higher installed and running,
- MySQL ODBC - either 5.1.5, or higher than 5.2.4 (other versions are not recommended)

Create a MySQL Repository

MySQL supports two different storage engines - InnoDB and MyISAM. In older versions of MySQL MyISAM was the default storage engine, but from MySQL v5.5 onwards InnoDB is the default. As of Enterprise Architect v13, Sparx Systems will no longer provide updates to the MyISAM script and recommend all users to choose InnoDB, as it supports transactions and UTF8.

Step	Action
1	Create a new empty database and configure it to your installation's defaults.
2	Load the EASchema_1220_MySQL.sql file from the Sparx Systems website into your SQL management console of choice (such as MySQL Workbench).
3	Run the script to create the required database schema.

Set Up the ODBC DSN

Step	Action
1	Create a suitable ODBC Data Source to point to your new database. Select these extended options: • 'Return matched rows instead of affected rows' • 'Allow big result sets'

Perform a Project Integrity Check

Step	Action
1	In Enterprise Architect, open the file-based project or template from which you are creating the project on the repository.
2	Select and execute the 'Configure > Model > Check Integrity > Project Integrity' ribbon option. This ensures your project data is 'clean' and free from errors before being copied to the repository.

Transfer the Project Data

Step	Action
1	Open Enterprise Architect. (If the 'Open Project' dialog displays, click on the Cancel button to open with no project loaded.)
2	Select the 'Configure > Model > Transfer > Project Transfer' ribbon option. The 'Project Transfer' dialog displays.
3	In the 'Transfer Type' panel, select 'File to DBMS'.
4	In the 'Source Project' field, click on the [...] button and browse for the name of the project file to copy to the repository. If the .EAP file has Replication enabled, this must be removed before performing the transfer.
5	At the right of the 'Target Project' field, click on the [...] button. The 'Datalink Properties' dialog displays.
6	Select 'Microsoft OLE DB Provider for ODBC Drivers' from the list. Click on the Next button.
7	In the 'Use Data source name' field, click on the drop-down arrow and select the ODBC Data Source you configured to point to your new database. Click on the OK button. The 'Project Transfer' dialog redisplays.
8	If required, select the 'Logfile' checkbox and type a path and filename for the data transfer log file.
9	Click on the Transfer button to begin the data transfer process.
10	When the process is complete, you have created a project on a MySQL database and can now connect to it and open it from Enterprise Architect.

Create a MySQL Repository

Creating a MySQL database is fully documented in the product information provided with your MySQL installer.

Having created your MySQL database, use the SQL script provided by Sparx Systems to create the required table definitions for Enterprise Architect.

You can obtain the scripts from the Sparx Systems website, on the:

- Registered Corporate edition 'Resources' page (Registered users)
- Corporate edition 'Resources' page (Trial users)

Third Party Tools

If you are unfamiliar with MySQL and DBMS systems in general, you might want to consider a suitable front end tool. MySQL Administrator is one such tool, providing a convenient graphical user interface to enable the creation of databases, the execution of scripts, and backups and restores.

You might, therefore:

1. Run MySQL Administrator and create a new database, then
2. Run MySQL Query Browser, and open and execute the MySQL repository script

After creating a MySQL data repository in Enterprise Architect, you must set up the MySQL ODBC driver.

Notes

- This feature is available in the Corporate, Business and Software Engineering, System Engineering and Ultimate editions
- Ensure that the collation is set to the alphabet you use, such as Latin or Cyrillic

Set up a MySQL ODBC Driver

After you have created a repository on your MySQL database, you set up the MySQL ODBC driver so that you can connect to a project on that repository from Enterprise Architect.

Prerequisites

Install:

- MySQL DBMS and repository
- MySQL ODBC driver software version 5.1.5 or higher than 5.2.4

Set up the ODBC Driver

Your MySQL driver is now available to connect to the repository from Enterprise Architect.

Step	Action
1	Under both 32-bit and 64-bit operating systems, Enterprise Architect requires 32-bit ODBC drivers to connect to a repository through ODBC. You can quickly load the correct 32-bit ODBC Data Source Administrator by selecting the 'Configure > User Tools > ODBC Data Sources' ribbon option. The ODBC Data Source Administrator window displays.
2	Click on the Add button. The 'Create New Data Source' dialog displays, enabling you to add a new DSN.
3	Select the appropriate MySQL ODBC Driver from the list.
4	Click on the Finish button. The 'MySQL Connector/ODBC' dialog displays.
5	Enter these configuration details: • A data source name for the connection • A description (optional) • The host address of the DBMS server • User name and password • The database name on the selected server
6	To set the advanced options, click on the Details>> button.
7	Select these checkboxes (where provided): • Return matched rows instead of affected rows ('Connection' or 'Cursors/Results' tab) • Allow big result sets ('Connection' tab)
8	Click on the Test Connection button to confirm that the details are correct.
9	If the test succeeds, click on the OK button to complete the configuration. If the test does not succeed, review your settings.

Connect to a MySQL Data Repository

To access a project in your MySQL data repository, you must connect to the data repository from Enterprise Architect.

Prerequisites

- The MySQL repository and the project already exist
- You have SELECT, INSERT, UPDATE, DELETE, EXECUTE and SHOW VIEW access permissions
- The MySQL ODBC driver has been set up

Access

Ribbon	: Open Project
Keyboard Shortcuts	Ctrl+O

Connect to the repository

Step	Action
1	In the 'Open Project' dialog, select the 'Connect to Server' checkbox.
2	Click on the [...] button. The 'Data Link Properties' dialog displays.
3	Select 'Microsoft OLE DB Provider for ODBC Drivers' from the list.
4	Click on the Next>> button. The 'Connection' tab displays.
5	Click on the 'Use data source name' radio button and on the drop-down arrow in its field. From the list, select the ODBC driver you have set up to connect to your MySQL repository.
6	If required, type in a User name and Password.
7	If required, type in an initial catalog.
8	Click on the Test Connection button to confirm that the details are correct.
9	If the test does not succeed, revise your settings. If the test succeeds, click on the OK button; the 'Connection Name & Type' dialog displays.

10	Give the connection a suitable name so that you can recognize it in the 'Recent Projects' panel on the 'Open Project' dialog.
11	If required, select the 'Encrypt Connection String' checkbox.
	This encrypts and hides the connection details of the database from the users that the connection string is given to.
12	If required, select the 'Lazy Load' checkbox to not load the full project view when the model is loaded; instead, only the parts that are necessary to display the visible portion of the tree are loaded.
	With this setting, the model loads faster and users can begin work sooner, but at the expense of later small delays as Enterprise Architect loads specific portions of the model.
13	If required, select the 'Use WAN Optimization' checkbox.
	To improve performance over a Wide Area Network, remote database calls can be routed through a WAN Optimizer that compresses the data returned from the repository, reducing transfer time.
	If you select this checkbox, complete the next three fields (see your administrator for the correct values); otherwise go to step 17.
14	In the 'Server' field, type the network name or address of the optimizer server.
15	In the 'Port' field, type the port on which the server is running on the remote machine.
16	In the 'DSN' field, type the data source name of the database as it appears on the remote machine.
17	Click on the OK button to complete the configuration and open the project.
	This also adds the project name to the 'Recent' list on the 'Start Page'; from now on, you can open the project again just by clicking on this name.

Notes

- This feature is available in the Corporate, Business and Software Engineering, System Engineering and Ultimate editions of Enterprise Architect

Create a Project in an Oracle Database

To create a project in a repository on Oracle 9i, 10g, 11g or 12c, you work through these stages:

- Stage 1: Create an empty database repository and set up the data tables
- Stage 2: Connect to the Oracle repository; Sparx Systems recommends using Oracle Provider for OLE DB - connecting with either Microsoft OLE DB Provider for ODBC or Microsoft OLE DB Provider for Oracle could cause unexpected behavior or performance issues due to known limitations of those drivers
- Stage 3: Perform a project integrity check on the project file you are using as a base (optional, but recommended)
- Stage 4: Transfer the data
- Stage 5: Connect to the repository to open the project

Prerequisites

- A machine with Oracle 9i, 10g or 11 g installed and running
- Oracle Client installed on the client machine, please ensure that the OLE/DB drivers are installed

Create Database Repository

Step	Action
1	Create the empty database.
2	Load the EASchema_1220_Oracle.sql file from the Sparx Systems website into your SQL management console of choice (such as Oracle SQL Developer).
3	Run the script to create the required database schema.

Perform Project Integrity Check

Step	Action
1	Open the file-based project or template.
2	Select and execute the 'Configure > Model > Check Integrity > Project Integrity' ribbon option. This ensures your project data is 'clean' before uploading.

Transfer the project data to the repository

Step	Action

1	Open Enterprise Architect. (If the 'Open Project' dialog displays, click on the Cancel button to open with no project loaded.)
2	Select the 'Configure > Model > Transfer > Project Transfer' ribbon option. The 'Project Transfer' dialog displays.
3	In the 'Transfer Type' panel, select 'File to DBMS'.
4	At the right of the 'Source Project' field, click on the [...] button and browse for the name of the project file to copy to Oracle. If the file has Replication enabled, this must be removed before performing the transfer.
5	At the right of the 'Target Project' field, click on the [...] button. The 'Datalink Properties' dialog displays.
6	Select 'Oracle Provider for OLE DB' from the list. Click on the Next button.
7	On the 'Connection' page of the 'Datalink Properties' dialog, enter the Oracle service name in the 'Data Source' field, and the user name and password as required. Click on the OK button. The 'Project Transfer' dialog redisplays.
8	If required, select the 'Logfile' checkbox and type a path and filename for the data transfer log file.
9	Click on the Transfer button to begin the data transfer process.
10	When the process is complete, you have created a project on a Oracle database and can now connect to it and open it from Enterprise Architect.

Notes

- When transferring a project to Oracle you must have access rights to execute the CREATE SEQUENCE command

Create an Oracle Data Repository

Creating an Oracle database is fully documented in the product information provided with your Oracle installer.

Before creating an Oracle data repository, install the appropriate version of Oracle (9i, 10g, 11g or 12c) and MDAC 2.6 or higher, and obtain access permission to create a new database.

Having created your Oracle database, use the SQL script (Oracle_EASchema.sql) provided by Sparx Systems to create the required table definitions and indexes for Enterprise Architect. You can obtain the script from the Sparx Systems website, on the:

- Registered Corporate edition 'Resources' page (Registered users), or
- Corporate edition 'Resources' page (Trial users)

When you eventually connect to the Oracle database from Enterprise Architect, Sparx Systems recommends using Oracle Provider for OLE DB. Connecting with either Microsoft OLE DB Provider for ODBC or Microsoft OLE DB Provider for Oracle could cause unexpected behavior or performance issues due to known limitations of those drivers.

Third Party Tools

If you are unfamiliar with Oracle and DBMS systems in general, you might want to consider a suitable front end tool. You could connect to the database and execute the scripts with a program such as Oracle SQL*Plus or SQL Plus Worksheet.

Notes

- This feature is available in the Corporate, Business and Software Engineering, System Engineering and Ultimate editions
- The OLE DB client must be the 32-bit version
- Set the collation to the alphabet you use, such as Latin or Cyrillic

Set up an Oracle ODBC Driver

After you have created a repository on your Oracle database, you can set up the Oracle ODBC driver so that you can connect to a project on that repository from Enterprise Architect, in preference to using an OLE DB connection.

Note that Sparx Systems recommends using Oracle Provider for OLE DB. Connecting with either Microsoft OLE DB Provider for ODBC or Microsoft OLE DB Provider for Oracle could cause unexpected behavior or performance issues due to known limitations of those drivers.

Prerequisites

Install:

- Oracle DBMS and repository

Set up the Oracle ODBC driver

Your Oracle driver is now available to connect to the repository from Enterprise Architect.

Step	Action
1	Under both 32-bit and 64-bit operating systems, Enterprise Architect requires 32-bit ODBC drivers to connect to a repository through ODBC. You can quickly load the correct 32-bit ODBC Data Source Administrator by selecting the 'Configure > User Tools > ODBC Data Sources' ribbon option. The ODBC Data Source Administrator window displays.
2	Click on the Add button. The 'Create New Data Source' dialog displays, enabling you to add a new DSN.
3	Select 'Oracle in OraDB11g_home1' from the list (or similar, depending on the ODBC installation).
4	Click on the Finish button. The 'Oracle ODBC Driver Configuration' dialog displays.
5	Enter these configuration details: • A data source name for the connection • A description (optional) • The TNS Service Name (click on the drop down arrow and select from the list) • User ID
6	Click on the Test Connection button and enter the Oracle user password to confirm that the details are correct.
7	If the test succeeds, click on the OK button to complete the configuration. If the test does not succeed, review your settings.

Connect to an Oracle Data Repository (ODBC)

To access a project in your Oracle 9i, 10g, 11g or 12c data repository, you connect to the data repository from Enterprise Architect.

Note that Sparx Systems recommends using Oracle Provider for OLE DB (see *Connect to an Oracle Data Repository (OLE DB)*). Connecting with either Microsoft OLE DB Provider for ODBC or Microsoft OLE DB Provider for Oracle could cause unexpected behavior or performance issues due to known limitations of those drivers.

Prerequisites

- The Oracle repository and the project already exist
- You have SELECT, UPDATE, INSERT and DELETE access permissions
- The Oracle ODBC driver has been set up

Access

Ribbon	: Open Project
Keyboard Shortcuts	Ctrl+O

Connect to an Oracle Data Repository using an ODBC Driver

Step	Action
1	In the 'Open Project' dialog, select the 'Connect to Server' checkbox.
2	Click on the [...] button. The 'Data Link Properties' dialog displays.
3	Select 'Microsoft OLE DB Provider for ODBC Drivers' from the list.
4	Click on the Next>> button. The 'Connection' tab displays.
5	In the 'Data source' field, click on the drop-down arrow and select the data source name.
6	Type in the User name and Password.
7	Click on the Test Connection button to confirm that the details are correct.
8	If the test does not succeed, revise your settings. If the test succeeds, click on the OK button; Oracle prompts you for the password.

9	Type in the password.
	The 'Connection Name and Type' dialog displays.
10	Give the connection a suitable name so that you can recognize it in the 'Recent Projects' panel on the 'Open Project' dialog.
11	If you want to encrypt and hide the connection details of the database from the users that the connection string is given to, select the 'Encrypt Connection String' checkbox.
12	The Lazy Load facility does not load the full project view when the model is loaded; instead, it loads only the parts that are necessary to display the visible portion of the tree. With this set, the model loads faster and users can begin work sooner, but at the expense of later small delays as Enterprise Architect loads specific portions of the model.
	If you want to use the Lazy load facility, select the 'Lazy Load' checkbox.
13	To improve performance over a Wide Area Network, remote database calls can be routed through a WAN Optimizer that compresses the data returned from the repository, reducing transfer time. If you are using a WAN, and want to apply this facility, select the 'Use WAN Optimization' checkbox; otherwise go to step 16.
	If you select this checkbox, complete the next two fields (see your administrator for the correct values).
14	In the 'Server' field, type the network name or address of the optimizer server.
15	In the 'Port' field, type the port on which the server is running on the remote machine.
16	Click on the OK button to complete the configuration and open the project.
	This also adds the project name to the 'Recent' list on the Start Page; from now on, you can open the project again just by clicking on this name.

Notes

- This feature is available in the Corporate, Business and Software Engineering, System Engineering and Ultimate editions

Connect to an Oracle Data Repository (OLE DB)

To access a project in your Oracle 9i, 10g, 11g or 12c data repository, you connect to the data repository from Enterprise Architect.

Prerequisites

- The Oracle repository and the project already exist
- You have SELECT, UPDATE, INSERT and DELETE access permissions

Access

Ribbon	: Open Project
Keyboard Shortcuts	Ctrl+O

Connect to an Oracle Repository using OLE DB

Step	Action
1	In the 'Open Project' dialog, select the 'Connect to Server' checkbox.
2	Click on the [...] button. The 'Data Link Properties' dialog displays.
3	Select 'Oracle Provider for OLE DB' from the list. **Do not** select 'Microsoft OLE DB Provider for Oracle'; Enterprise Architect might not work as expected.
4	Click on the Next>> button. The 'Connection' tab displays.
5	In the 'Data source' field, click on the drop-down arrow and select the data source name (the service name of the Oracle database).
6	Type in the User name and Password.
7	Click on the Test Connection button to confirm that the details are correct.
8	If the test does not succeed, revise your settings. If the test succeeds, click on the OK button; the 'Connection Name and Type' dialog displays.
9	Give the connection a suitable name so that you can recognize it in the 'Recent Projects' panel on the

	'Open Project' dialog.
10	If you want to encrypt and hide the connection details of the database from the users that the connection string is given to, select the 'Encrypt Connection String' checkbox.
11	If required, select the 'Lazy Load' checkbox to not load the full project view when the model is loaded; instead, only the parts that are necessary to display the visible portion of the tree are loaded.
	With this setting, the model loads faster and users can begin work sooner, but at the expense of later small delays as Enterprise Architect loads specific portions of the model.
12	To improve performance over a Wide Area Network, remote database calls can be routed through a WAN Optimizer that compresses the data returned from the repository, reducing transfer time. If you are using a WAN, and want to apply this facility, select the 'Use WAN Optimization' checkbox; otherwise go to step 15.
	If you select this checkbox, complete the next two fields (see your administrator for the correct values).
13	In the 'Server' field, type the network name or address of the optimizer server.
14	In the 'Port' field, type the port on which the server is running on the remote machine.
15	Click on the OK button to complete the configuration and open the project.
	This also adds the project name to the 'Recent' list on the Start Page; from now on, you can open the project again just by clicking on this name.

Notes

- This feature is available in the Corporate, Business and Software Engineering, System Engineering and Ultimate editions

Create a Project in a PostgreSQL Database

To create a project in a repository on PostgreSQL 7, 8 or 9, you work through these stages:

- Stage 1: Create an empty database repository and set up the data tables
- Stage 2: Set up the PostgreSQL ODBC Driver
- Stage 3: Perform a project integrity check on the project file you are using as a base (optional, but recommended)
- Stage 4: Transfer the data
- Stage 5: Connect to the repository to open the project

Prerequisites

- A machine with PostgreSQL 7.3.2 or higher installed and running
- psqlODBC, version 7.03.01.00 or higher has been installed (do not use version 8.3.4 or 8.4.1)

Create Database Repository

Step	Action
1	Create the empty database.
2	Load the EASchema_PostgreSQL_1220.sql file from the Sparx Systems website into your SQL management console of choice (such as pgAdminIII).
3	Run the script to create the required database schema.

Set Up the ODBC DSN

Step	Action
1	Create a suitable ODBC Data Source to point to your new database. Select these extended options: Page 1: Disable Genetic Optimizer - **Uncheck** Use Declare/Fetch - **Check** Unknowns as LongVarChar - **Check** Bools as Char - **Uncheck** Max Varchar - **1024** Max LongVarChar - **1000000** Page 2:

	bytea as LO - **Check**
	Disallow Premature - **Check**
	Protocol - **7.4+**

Perform a Project Integrity Check

Step	Action
1	In Enterprise Architect, open the file-based project or template from which you are creating the project on the repository.
2	Select and run the 'Configure > Model > Check Integrity > Project Integrity' ribbon option. This ensures your project data is 'clean' before being copied to the repository.

Transfer the project data to the repository

Step	Action
1	Open Enterprise Architect. (If the 'Open Project' dialog displays, click on the Cancel button to open with no project loaded.)
2	Select the 'Configure > Model > Transfer > Project Transfer' ribbon option. The 'Project Transfer' dialog displays.
3	In the 'Transfer Type' panel, select 'File to DBMS'.
4	In the 'Source Project' field, click on the [...] button and browse for the name of the project file to copy to the repository. If the .EAP file has Replication enabled, this must be removed before performing the transfer.
5	At the right of the 'Target Project' field, click on the [...] button. The 'Datalink Properties' dialog displays.
6	Select 'Microsoft OLE DB Provider for ODBC Drivers' from the list. Click on the Next button.
7	On the 'Use Data Source Name' field, click on the drop-down arrow and select the ODBC Data Source you configured to point to your new database. Click on the OK button. The 'Project Transfer' dialog redisplays.
8	If required, select the 'Logfile' checkbox and type a path and filename for the data transfer log file.
9	Click on the Transfer button to begin the data transfer process.

10	When the process is complete, you have created a project on a PostgreSQL database and can now connect to it and open it from Enterprise Architect.

Notes

- During the transfer, if an error message displays reporting '...nonstandard use of \\ in a string literal...', set the server variable in the postgresql.conf file to: escape_string_warning = off

Create a PostgreSQL Repository

Creating a PostgreSQL database is fully documented in the product information provided with your PostgreSQL installer.

Having created your PostgreSQL database, use the SQL script provided by Sparx Systems to create the required table definitions for Enterprise Architect. You can obtain the scripts from the Sparx Systems website, on the:

- Registered Corporate edition 'Resources' page (Registered users)
- Corporate edition 'Resources' page (Trial users)

Third Party Tools

If you are unfamiliar with PostgreSQL and DBMS systems in general, you might want to consider a suitable front end tool. One such tool is pgAdminIII. It provides a convenient graphical user interface to enable creation of databases, execution of scripts and restores.

After creating a PostgreSQL data repository in Enterprise Architect, you must set up the PostgreSQL ODBC driver.

Notes

- This feature is available in the Corporate, Business and Software Engineering, System Engineering and Ultimate editions
- Ensure that the collation is set to the alphabet you use, such as Latin or Cyrillic

Set up a PostgreSQL ODBC Driver

After you have created a repository on your PostgreSQL database, you set up the PostgreSQL ODBC driver so that you can connect to a project on that repository from Enterprise Architect.

Prerequisites

Install:

- PostgreSQL DBMS and repository
- PostgreSQL ODBC driver software version 7.03.01.00 or above (note that versions 8.3.4 and 8.4.1 of the PostgreSQL ODBC Driver are not supported)

Set up the ODBC driver

Your PostgreSQL driver is now available to connect to the repository from Enterprise Architect.

Step	Action
1	Under both 32-bit and 64-bit operating systems, Enterprise Architect requires 32-bit ODBC drivers to connect to a repository through ODBC. You can quickly load the correct 32-bit ODBC Data Source Administrator by selecting the 'Configure > User Tools > ODBC Data Sources' ribbon option. The ODBC Data Source Administrator window displays.
2	Click on the Add button. The 'Create New Data Source' dialog displays, enabling you to add a new DSN.
3	Select 'PostgreSQL UNICODE' from the list.
4	Click on the Finish button. The 'Postgre SQL Connector/ODBC' dialog displays.
5	Enter these configuration details: • A data source name for the connection • The actual name of the database • The host address of the DBMS server • User name • A description (optional) • The password
6	To set the advanced options, click on the Datasource button and set the options on 'Page 1' as shown:

Note: from v9.5.4 onwards some options shown in the above image are no longer available and can be ignored.

7	If you are using PostgreSQL version 8 to 9.5.4, set the options on Page 2 as shown:

- The 'Disallow Premature' checkbox should be checked

	• In the 'Protocol' panel, the '7.4+' radio button Note: from v9.5.4 onwards the two options listed above are no longer available and can be ignored.
8	Click on the OK button to complete the configuration.

Connect to a PostgreSQL Data Repository

To access a project in your MySQL data repository, you must connect to the data repository from Enterprise Architect.

Prerequisites

- The PostgreSQL repository and the project already exist
- You have SELECT, UPDATE, INSERT and DELETE access permissions
- The PostgreSQL ODBC driver has been set up

Access

Ribbon	: Open Project
Keyboard Shortcuts	Ctrl+O

Connect to the repository

Step	Action
1	In the 'Open Project' dialog, select the 'Connect to Server' checkbox.
2	Click on the [...] button. The 'Data Link Properties' dialog displays.
3	Select 'Microsoft OLE DB Provider for ODBC Drivers' from the list.
4	Click on the Next>> button. The 'Connection' tab displays.
5	Click on the 'Use data source name' radio button and on the drop-down arrow in its field. From the list, select the ODBC driver you have set up to connect to your PostgreSQL repository.
6	Click on the Test Connection button to confirm that the details are correct.
7	If the test does not succeed, revise your settings. If the test succeeds, click on the OK button; the 'Connection Name & Type' dialog displays.
8	Give the connection a suitable name so that you can recognize it in the 'Recent Projects' panel on the 'Open Project' dialog.
9	If required, select the 'Encrypt Connection String' checkbox.

		This encrypts and hides the connection details of the database from the users that the connection string is given to.
10		If required, select the 'Lazy Load' checkbox to not load the full project view when the model is loaded; instead, only the parts that are necessary to display the visible portion of the tree are loaded.
		With this setting, the model loads faster and users can begin work sooner, but at the expense of later small delays as Enterprise Architect loads specific portions of the model.
11		If required, select the 'Use WAN Optimization' checkbox.
		To improve performance over a Wide Area Network, remote database calls can be routed through a WAN Optimizer that compresses the data returned from the repository, reducing transfer time.
		If you select this checkbox, complete the next three fields (see your administrator for the correct values); otherwise go to step 15.
12		In the 'Server' field, type the network name or address of the optimizer server.
13		In the 'Port' field, type the port on which the server is running on the remote machine.
14		In the 'DSN' field, type the data source name of the database as it appears on the remote machine.
15		Click on the OK button to complete the configuration and open the project.
		This also adds the project name to the 'Recent' list on the Start Page; from now on, you can open the project again just by clicking on this name.

Notes

- This feature is available in the Corporate, Business and Software Engineering, System Engineering and Ultimate editions

Create a Project in an Access 2007 Database

In setting up a project in an Access 2007 repository, you could be working in one of two situations:

- You are able to work in Access 2007 itself, in which case you can convert your .eap project directly to an Access 2007 .accdb database file, which you can connect to and open in Enterprise Architect
- You have access to an empty Access 2007 .accdb file, and need to transfer a .eap project into that .accdb file

In either case, before you convert or transfer the project data, you could perform a project integrity check on the .eap file you are using as a base (optional, but recommended).

Prerequisites

- The 'Microsoft Office 12.0 Access Database Engine OLE DB Provider' ODBC driver installed

Perform a Project Integrity Check

Step	Action
1	Open the base project or template .eap file.
2	Select and run 'Configure > Model > Check Integrity > Project Integrity'. This ensures your project data is 'clean' before uploading.

Convert a .eap file in Access 2007

Step	Action
1	Open MS Access 2007 and open the source .eap file within this product. Allow Access to convert the .eap file to a .accdb file. This forms the Access 2007 repository.

Transfer the project data into an empty .accdb file

Step	Action
1	Open Enterprise Architect. (If the 'Open Project' dialog displays, click on the Cancel button to open with no project loaded.)
2	Select the 'Configure > Model > Transfer > Project Transfer' menu option. The 'Project Transfer' dialog displays.
3	In the 'Transfer Type' panel, select 'File to DBMS'.

4	In the 'Source Project' field, type the name of the .EAP file to copy to the repository.
	If the .EAP file has Replication enabled, this must be removed before performing the transfer.
5	At the right of the 'Target Project' field, click on the [...] button.
	The 'Datalink Properties' dialog displays.
6	Select 'Microsoft Office 12.0 Access Database Engine OLE DB Provider' from the list.
	Click on the Next button.
7	On the 'Data Source Details' page of the 'Datalink Properties' dialog, type in the full path to the Access 2007 .accdb file.
	Click on the OK button to return to the 'Project Transfer' dialog.
8	If required, select the 'Logfile' checkbox and type a path and filename for the data transfer log file.
9	Click on the Transfer button to begin the data transfer process.
	When the process is complete, you have created your project in an Access 2007 database and can now open it directly from Enterprise Architect, browsing for the .accdb file location in the 'Open Project' dialog.

Notes

- This feature is available in the Corporate, Business and Software Engineering, System Engineering and Ultimate editions
- If you do not have Access 2007, you can download the Access Database Engine from the Microsoft downloads site
- Ensure that the collation is set to the alphabet you use, such as Latin or Cyrillic

Create a Project in a SQL Server Database

To create a project on a repository in SQL Server 2005 and above, or SQL Server Express 2005 and above, work through these stages:

- Stage 1: Create an empty database repository and set up the data tables
- Stage 2: Perform a project integrity check on the project file you are using as a base (optional, but recommended)
- Stage 3: Transfer the data
- Stage 4: Connect to the repository to open the project

Prerequisites

- A machine with SQL Server installed and running
- MDAC 2.6 or higher on the client machine
- Permissions to create databases on SQL Server
- ALTER permission on the t_image table in Enterprise Architect, to be able to import model images

Create a SQL Server Repository

Step	Action
1	Create a new empty database.
2	Load the EASchema_1220_SQLServer.sql file from the Sparx Systems website into your SQL management console of choice (such as Microsoft SQL Server Management Studio).
3	Run the script to create the required database schema.

Perform a Project Integrity Check

Step	Action
1	In Enterprise Architect, open the file-based project or template from which you are creating the project on the repository.
2	Select the 'Configure > Model > Check Integrity > Project Integrity' ribbon option, and execute the check. This ensures your project data is 'clean' before being copied to the repository.

Transfer the project data to the repository

Step	Action

1	Open Enterprise Architect. (If the 'Open Project' dialog displays, click on the Cancel button to open with no project loaded.)
2	Select the 'Configure >Model > Transfer > Project Transfer' ribbon option. The 'Project Transfer' dialog displays.
3	In the 'Transfer Type' panel, select '.EAP to DBMS'.
4	In the 'Source Project' field, type the name of the project file to copy to the repository. If the .EAP file has Replication enabled, this must be removed before performing the transfer.
5	At the right of the 'Target Project' field, click on the [...] button. The 'Datalink Properties' dialog displays.
6	Select 'Microsoft OLE DB Provider for SQL Server' from the list. Click on the Next button.
7	On the 'Data Source Details' page of the 'Datalink Properties' dialog, type in the server name, database name and any security details required. Click on the OK button. The 'Project Transfer' dialog redisplays.
8	If required, select the 'Logfile' checkbox and type a path and filename for the data transfer log file.
9	Click on the Transfer button to begin the data transfer process.
10	When the process is complete, you have created a project on a SQL Server database and can now connect to it and open it from Enterprise Architect.

Create a SQL Server Repository

Creating a SQL Server database is fully documented in the product information provided with your SQL Server installer.

Having created your SQL Server database, use the SQL script (SQLServer_EASchema.sql) provided by Sparx Systems to create the required table definitions for Enterprise Architect. You can obtain the script from the Sparx Systems website, on the:

- Registered Corporate edition 'Resources' page (Registered users)
- Corporate edition 'Resources' page (Trial users)

If you are unfamiliar with SQL Server and DBMS systems in general, you might want to consider a suitable front end tool, such as SQL Server Management Studio.

SQL Enterprise Manager example

Step	Action
1	In SQL Server Management Studio, locate the server on which to create your new project; for example: DBSERVER02\SQLEXPRESS.
2	Right-click and choose the 'New Database' option.
3	Enter a suitable name for the database. Set any file options as required. Ensure that the database collation is set to the alphabet you use, such as Latin or Cyrillic, and case-insensitive.
4	Click on the database to select it, then select the 'New Query' menu option.
5	In the Query window, use the 'Open File' dialog to locate the SQLServer_EASchema.sql script file.
6	Click on the Open button. In the drop-down menu, check that you have selected the correct database to run the script in.
7	Click on the Execute button; SQL Server executes the script, which creates the base tables for an Enterprise Architect project.

Notes

- This feature is available in the Corporate, Business and Software Engineering, System Engineering and Ultimate editions
- You should have ALTER permission on the t_image table in Enterprise Architect, to be able to import model images
- When creating a project in a SQL Server database you must have 'db_ddladmin' permission in order to execute the SET IDENTITY_INSERT (table) {ON | OFF} command

SQL Server Security Permissions

The security model implemented by Microsoft's SQL Server is quite powerful and highly configurable, supporting many different possible solutions for securing the data contained in SQL Server databases and ensuring it is only accessible to users with the required permissions. For a more detailed description and explanation of SQL Server permissions, see the SQL Server documentation.

Enterprise Architect users who plan to add, edit and delete contents in a SQL Server repository must have permissions to execute SELECT, UPDATE, INSERT and DELETE statements on all Tables in the Enterprise Architect database. The easiest way to achieve this is to grant the database roles of 'db_datareader' and 'db_datawriter' to each user.

Additional Permissions for Project Transfers

When an Enterprise Architect repository is transferred into a SQL Server based repository, it is necessary for Enterprise Architect to execute a number of SET IDENTITY_INSERT (table) {ON | OFF} commands during the process. This means the user performing the transfer must have a high level of security, in the role of 'db_ddladmin'.

Does Enterprise Architect support Windows Authentication?

Enterprise Architect does support Windows Authentication. However, the type of authentication is determined by the configuration of the connection used, and not by Enterprise Architect.

Windows Authentication to SQL Server is commonly used by Enterprise Architect users, but this requires that all Windows users in Enterprise Architect be defined on the SQL Server server and be granted the security roles 'db_datareader' and 'db_datawriter' for the repository.

Connect to a SQL Server Data Repository

To access a project in your SQL Server data repository, you must connect to the data repository from Enterprise Architect.

Prerequisites

- The SQL Server repository and the project already exist
- You have SELECT, UPDATE, INSERT and DELETE access permissions
- You have ALTER permission on the t_image table in Enterprise Architect, to be able to import model images

Access

Ribbon	: Open Project
Keyboard Shortcuts	Ctrl+O

Connect to the repository

Step	Action
1	In the 'Open Project' dialog, select the 'Connect to Server' checkbox.
2	Click on the [...] button. The 'Data Link Properties' dialog displays.
3	Select 'Microsoft OLE DB Provider for SQL Server' from the list.
4	Click on the Next>> button. The 'Connection' tab displays.
5	Type in the server details, including Server Name, User Name and Password.
6	Click on the 'Select the database on the server' option and on the drop-down arrow. From the list, select the project to connect to.
7	Click on the Test Connection button to confirm that the details are correct.
8	If the test does not succeed, revise your settings. If the test succeeds, click on the OK button; the 'Connection Name & Type' dialog displays.
9	Give the connection a suitable name so that you can recognize it in the 'Recent Projects' panel on the

		'Open Project' dialog.
10		If required, select the 'Encrypt Connection String' checkbox.
		This encrypts and hides the connection details of the database from the users that the connection string is given to.
11		If required, select the 'Lazy Load' checkbox to not load the full project view when the model is loaded; instead, only the parts that are necessary to display the visible portion of the tree are loaded.
		With this setting, the model loads faster and users can begin work sooner, but at the expense of later small delays as Enterprise Architect loads specific portions of the model.
12		If required, select the 'Use WAN Optimization' checkbox.
		To improve performance over a Wide Area Network, remote database calls can be routed through a WAN Optimizer that compresses the data returned from the repository, reducing transfer time.
		If you select this checkbox, complete the next two fields (see your administrator for the correct values); otherwise go to step 15.
13		In the 'Server' field, type the network name or address of the optimizer server.
14		In the 'Port' field, type the port on which the server is running on the remote machine.
15		Click on the OK button to complete the configuration and open the project.
		This also adds the project name to the 'Recent' list on the Start Page; from now on, you can open the project again just by clicking on this name.

Notes

- This feature is available in the Corporate, Business and Software Engineering, System Engineering and Ultimate editions

Create a Project in a Sybase ASA/SQL Anywhere Database

To create a project in a repository on Sybase Adaptive Server Anywhere 8 or 9, or SQL Anywhere 10, 11 or 12, you work through these stages:

- Stage 1: Create an empty database repository
- Stage 2: Set up the ASA ODBC Driver
- Stage 3: Perform a project integrity check on the project file you are using as a base (optional, but recommended)
- Stage 4: Transfer the data
- Stage 5: Connect to the repository and open the project

Prerequisites

- A machine with Adaptive Server Anywhere installed and running
- Install SQL Anywhere Studio, this also installs the ASA ODBC driver

Create Database Repository

Step	Action
1	Create a new empty database.
2	Load the EASchema_1220_ SybaseASA.sql file from the Sparx Systems website into your SQL management console of choice (such as Sybase Central).
3	Run the script to create the required database schema.

Set up the ODBC DSN

Step	Action
1	Create a suitable ODBC Data Source to point to your new database.

Perform Project Integrity Check

Step	Action
1	In Enterprise Architect, open the file-based project or template from which you are creating the project on the repository.
2	Select and run 'Configure > Model > Check Integrity > Project Integrity'.

	This ensures your project data is 'clean' before being copied to the repository.

Transfer the Project Data

Step	Action
1	Open Enterprise Architect. (If the 'Open Project' dialog displays, click on the Cancel button to open with no project loaded.)
2	Select the 'Configure > Model > Transfer > Project Transfer' ribbon option. The 'Project Transfer' dialog displays.
3	In the 'Transfer Type' panel, select '.EAP to DBMS'.
4	In the 'Source Project' field, type the name of the project file to copy to the repository. If the .EAP file has Replication enabled, this must be removed before performing the transfer.
5	At the right of the 'Target Project' field, click on the [...] button. The 'Datalink Properties' dialog displays.
6	Select 'Microsoft OLE DB Provider for ODBC Drivers' from the list. Click on the Next button.
7	In the 'Use Data source name' field, click on the drop-down arrow and select the ODBC Data Source you configured to point to your new database. Click on the OK button. The 'Project Transfer' dialog redisplays.
8	If required, select the 'Logfile' checkbox and type a path and filename for the data transfer log file.
9	Click on the Transfer button to begin the data transfer process.
10	When the process is complete, you have created a project on an ASA database and can now connect to it and open it from Enterprise Architect.

Create an Adaptive Server Anywhere Repository

Creating an ASA database is fully documented in the product information provided with your ASA installer.

Having created your ASA database, use the SQL script (ASA_EASchema.sql) provided by Sparx Systems to create the required table definitions for Enterprise Architect. You can obtain the scripts from the Sparx Systems website, on the:

- Registered Corporate edition 'Resources' page (Registered users)
- Corporate edition 'Resources' page (Trial users)

Third Party Tools

If you are unfamiliar with ASA and DBMS systems in general, you might want to consider a suitable front end tool. Sybase Central is one such tool, that can be installed along with the DBMS. It provides a convenient graphical user interface to enable creation of databases, execution of scripts and restores.

After creating an ASA data repository in Enterprise Architect, you must set up the ASA ODBC driver.

Notes

- This feature is available in the Corporate, Business and Software Engineering, System Engineering and Ultimate editions
- Ensure that the collation is set to the alphabet you use, such as Latin or Cyrillic

Set up an ASA ODBC Driver

After you have created a repository on your ASA database, you set up the ASA ODBC driver so that you can connect to a project on that repository from Enterprise Architect.

Prerequisites

Install:

- Adaptive Server Anywhere - SQL Anywhere Studio 8 or higher, and create a repository
- ASA ODBC driver software (installed with the ASA DBMS)

Set up the ODBC Driver

Your ASA driver is now available to connect to the repository from Enterprise Architect.

Step	Action
1	Under both 32-bit and 64-bit operating systems, Enterprise Architect requires 32-bit ODBC drivers to connect to a repository through ODBC. You can quickly load the correct 32-bit ODBC Data Source Administrator by selecting the 'Configure > User Tools > ODBC Data Sources' ribbon option. The ODBC Data Source Administrator window displays.
2	Click on the Add button. The 'Create New Data Source' dialog displays, enabling you to add a new DSN.
3	Select 'Adaptive Server Anywhere' or 'SQL Anywhere' from the list.
4	Click on the Finish button. The 'ASA Connector/ODBC' dialog displays.
5	Enter these configuration details: • A data source name for the connection, on the 'ODBC' tab • User name and password on the 'Login' tab (dba, sql are the defaults when ASA is installed) • The server name and the path to the database, on the 'Database' tab • The network protocol on the 'Network' tab (if the database is on a network location)
6	Return to the 'ODBC' tab and click on the Test Connection button to confirm that the details are correct.
7	If the test succeeds, click on the OK button to complete the configuration. If the test does not succeed, review your settings.

Connect to an ASA Data Repository

To access a project in your Adaptive Server Anywhere (ASA) data repository, you must connect to the data repository from Enterprise Architect.

Prerequisites

- The ASA repository and the project already exist
- You have SELECT, UPDATE, INSERT and DELETE access permissions
- The ASA ODBC driver has been set up

Access

Ribbon	: Open Project
Keyboard Shortcuts	Ctrl+O

Connect to the repository

Step	Action
1	In the 'Open Project' dialog, select the Connect to Server button.
2	Click on the ⟨...⟩ button. The 'Data Link Properties' dialog displays.
3	Select 'Microsoft OLE DB Provider for ODBC Drivers' from the list.
4	Click on the Next>> button. The 'Connection' tab displays.
5	Click on the 'Use data source name' radio button and on the drop-down arrow in its field. From the list, select the ODBC driver you have set up to connect to your ASA repository.
6	Click on the Test Connection button to confirm that the details are correct.
7	If the test does not succeed, revise your settings. If the test succeeds, click on the OK button; the 'Connection Name & Type' dialog displays.
8	Give the connection a suitable name so that you can recognize it in the 'Recent Projects' panel on the 'Open Project' dialog.

9	If required, select the 'Encrypt Connection String' checkbox.
	This encrypts and hides the connection details of the database from the users that the connection string is given to.
10	If required, select the 'Lazy Load' checkbox to not load the full project view when the model is loaded; instead, only the parts that are necessary to display the visible portion of the tree are loaded.
	With this setting, the model loads faster and users can begin work sooner, but at the expense of later small delays as Enterprise Architect loads specific portions of the model.
11	If required, select the 'Use WAN Optimization' checkbox.
	To improve performance over a Wide Area Network, remote database calls can be routed through a WAN Optimizer that compresses the data returned from the repository, reducing transfer time.
	If you select this checkbox, complete the next three fields (see your administrator for the correct values); otherwise go to step 15.
12	In the 'Server' field, type the network name or address of the optimizer server.
13	In the 'Port' field, type the port on which the server is running on the remote machine.
14	In the 'DSN' field, type the data source name of the database as it appears on the remote machine.
15	Click on the OK button to complete the configuration and open the project.
	This also adds the project name to the 'Recent' list on the Start Page; from now on, you can open the project again just by clicking on this name.

Notes

- This feature is available in the Corporate, Business and Software Engineering, System Engineering and Ultimate editions

The WAN Optimizer

You can significantly improve Enterprise Architect's performance in a Wide Area Network (WAN) by reducing the amount of data transmitted and, in turn, the number of network calls made. To achieve this, you can use the Sparx Systems Wide Area Network (WAN) Optimizer, a lightweight server installed on a Local Area Network (LAN) connection to a Database Management System (DBMS) that hosts an Enterprise Architect repository. You can configure the server to listen for client connections on a particular port; it acts as a local proxy to execute queries and return the results in a compressed format to the client.

In this diagram, transmission between Enterprise Architect and a DBMS is depicted first without and then with the WAN Optimizer.
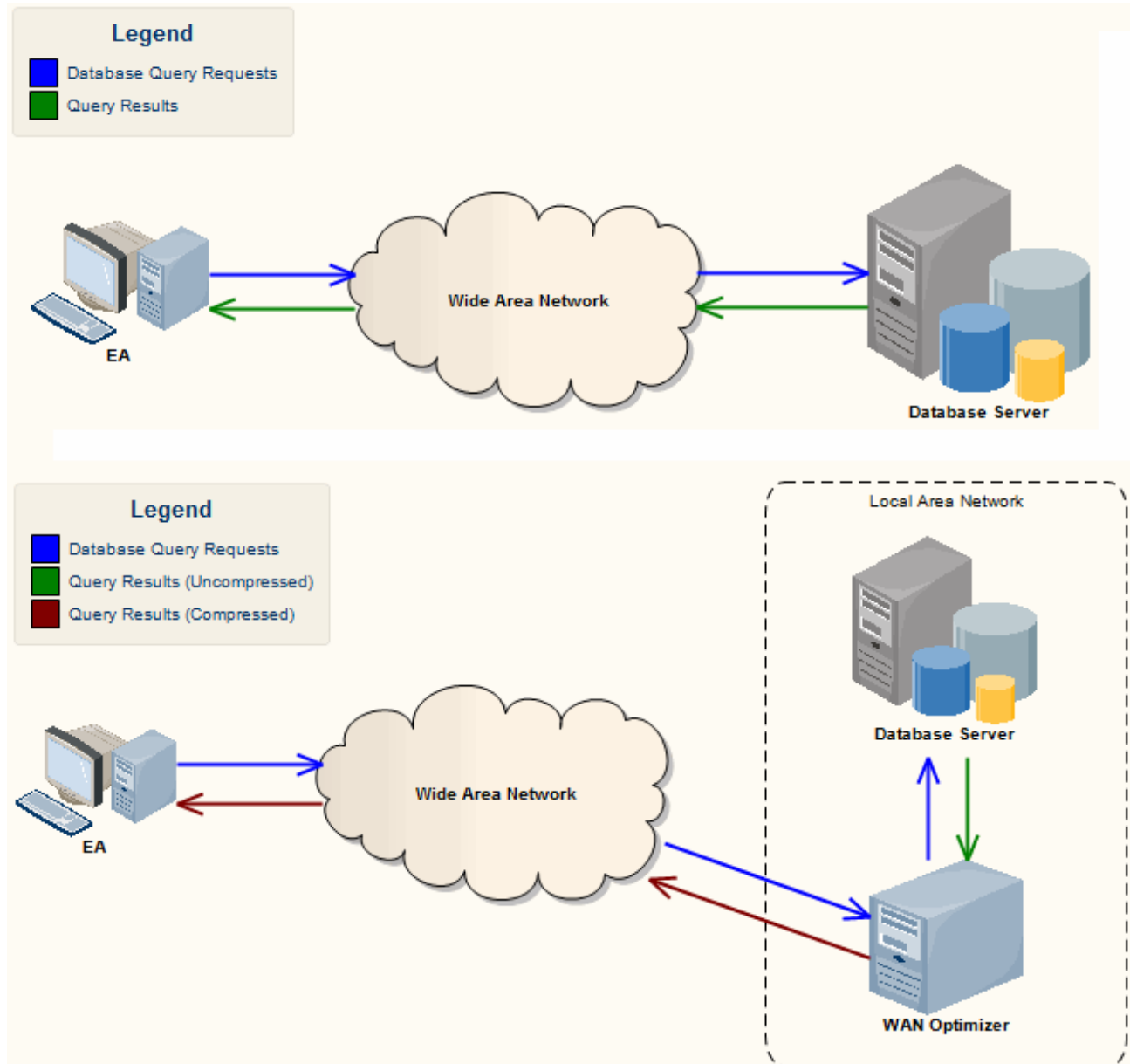
You can download the WAN Optimizer installer from the 'Downloads' page of the Registered Users section of the Sparx Systems website. The Wan Optimizer Service installer package provides two installable features for the target machine:

- WAN Optimizer Service - the installer also helps register and start the service on the target machine, and add it to the Windows Startup folder
- WAN Optimizer Admin Client - to enable an administrator to administer and configure the service from a remote

client

The Optimizer has its own *Sparx Systems WAN Optimizer User Guide*. See that Guide for more information on:

- WAN Optimizer Components
- Installing and Starting the WAN Optimizer Service
- Configuring the Service
- Troubleshooting

The WAN Optimizer forms the basis of the Sparx Systems Cloud Services, which in turn is the core of the separately-purchased Pro Cloud Server. If you are using the Clod products, you have no need to use the WAN Optimizer.

Notes

The WAN optimizer forms the basis for the Sparx Systems Cloud Services, which in turn is the core of the Sparx Systems Pro Cloud Server. Both of these facilities provide the basic function of the WAN Optimizer, but do so faster and more securely; they also have many more additional features.

Whilst you can still use the WAN Optimizer as described here, we recommend that you review the Cloud Services products and use one of them instead.

Cloud Based Repositories

With a simple connection to the internet or web - the 'Cloud' - you can access and work on projects stored in repositories at remote locations anywhere in the world, or access local repositories from such locations, using the Enterprise Architect installation on your machine. The advantages of working in this way include:

- You, as a user, do not need to have database drivers installed on your machine or to create a database connection, as you do to work directly on projects held on a DBMS server; your system administrator performs the driver set-up and connection once only, during server configuration

- The http and, especially, https connections apply a firewall and all the security facilities provided by the web server - whether Apache or IIS - to your project work, such as URL authorization, domain authentication and IP security

- The Cloud Server can be configured to encrypt all communications using standard TLS/SSL protocols; you can be certain that your data is not intercepted during transmission across insecure networks

Using the Cloud can help to reduce lag time for common tasks such as opening Packages and editing model data. Requests are cached and sent together, rather than individually, which minimizes communication delays.

Using the Cloud server is most beneficial when:

- You and other users want to minimize the set-up requirements each user has on their workstation

- You want to expose models held outside your private network

- You and other users are accessing your models over slow connections

Notes

- This facility is integrated with the Corporate, Business and Software Engineering, Systems Engineering and Ultimate editions of Enterprise Architect

- An enhanced version of Cloud Services, called Pro Cloud Server, is available for purchase under a separate license; this version supports the WebEA facility, in which users can review models through a web browser on various portable devices, without the need for an Enterprise Architect workstation client

Introducing Cloud Services

The Sparx Systems Cloud Services solution provides seamless access to Enterprise Architect models in repositories held anywhere in the world, providing high performance for remote access, secure encrypted links and optimization for high latency WAN connections.

Cloud-based repositories provide easy access to models, not only for people within your local team but also for remote team members, external customers and consultants with Internet access, anywhere around the world. Collaborative and distributive modeling and design are dramatically enhanced by using a Cloud-based repository, which can provide the shared backbone for a wealth of development scenarios.

This section will:

1. Familiarize you with the concepts of the Cloud Services

2. Discuss considerations of when and where you should use a Cloud server

3. Walk you through the process of setting up a server

4. Walk you through the process of connecting for the first time

5. Highlight some of the additional functionality available through the Cloud server

6. Outline working with Active Directory using IIS

Overview

Enterprise Architect models are stored in databases, such as MySQL, SQL Server and Oracle. Traditional installations of Enterprise Architect require each user to install the appropriate drivers for each DBMS, and in some cases define an ODBC DSN with the database connection details.

When connecting to a Cloud Server, the procedure for a user to connect to a model is dramatically simplified. For users connecting to models stored in a Cloud Server, the process is as simple as entering a single URL to the provided Cloud repository. No special drivers or further configuration is required.

Connecting to a Cloud Server provides a number of key benefits:

1. Improved performance for models used for distributed development. The Cloud Server provides benefits to connections that involve high latency and reduced data transfer speeds.

2. The process of setting up drivers and connections is performed once by an administrator during the server configuration. The only set-up required on a user machine is to install Enterprise Architect and to connect to a model on the Cloud server.

3. Database servers no longer have to be exposed through a firewall; the Cloud server can be run from inside the corporate firewall. All model connections are created using HTTP, allowing firewalls to completely isolate your database server.

4. A Cloud server can be configured to encrypt all communication. Using standard TLS/SSL protocols, you can be confident that your data is not intercepted during transmission on insecure networks.

5. A Cloud server can be configured to provide HTTP-level authorization to any model taken directly from the model user list. Even when the model is exposed on a public network, you can be assured that only authorized users are able to access your model.

6. A Cloud server can be configured to provide read-only access to any model; for example, for clients required to review a model.

Security Considerations

The Sparx Systems Cloud Server runs as a Windows Service, accepting network connections from Enterprise Architect clients and sending the data required by the system back over the network. As with any web connected service, there are security concerns that must be addressed when setting up a new service. To help you minimize risks, consider these points:

- If any data is considered private, always use an HTTPS connection and require user authentication; there is an option on each of the service's database configurations to prompt for this

- There is an implicit trust in sharing a model with anyone; security is available in models, which prevents a wide range of possible interactions, but with Enterprise Architect's flexibility determined users can circumvent this

  In particular, Model Search SQL queries can be run in a number of places that allow data to be read that would not otherwise be accessible, including user IDs and hashes of their passwords; to block this type of access for a list of users, you could use **Global Authentication** instead of **Model Authentication**, as discussed in the *Cloud Server Configuration Help* topic

Sparx Systems Cloud Services Products

Sparx Systems offer two configurations of their Cloud Services product.

- The core product, Sparx Systems Cloud Services, is integrated with your Corporate, Business and Software Engineering, Systems Engineering or Ultimate edition of Enterprise Architect, although it is separately installed; it contains the Cloud Server, a Management Client, optional HTTP support, and Firebird ODBC drivers

- Sparx Systems Pro Cloud is a separately purchased and licensed product that contains the core Sparx Systems Cloud Services plus the Sparx Systems Keystore and floating license server, Row-level security for Oracle and SQL Server based models, and WebEA - a web based application that allows users to access Enterprise Architect models through a browser, while on portable devices or remote desk-top systems that do not have Enterprise Architect installed

  Sparx Systems Pro Cloud is available in three editions, to support small, medium and large projects with differing numbers of models; the editions have the same installer but require different access keys to install them:
    - Small Business, which supports up to three Firebird repositories; Row Level
      Security is not available in this edition, as the feature is for Oracle and SQL Server
      databases
    - Team Server, which supports up to five repositories of any type
    - Enterprise Server, which supports any number of repositories of any type, within any
      restrictions imposed by your server and network

Webinar

Before you go on to set up the Cloud Server, Sparx Systems recommend that you watch the 'Introduction to Sparx Systems Cloud Services' webinar, which you can access by clicking on this link:

http://www.sparxsystems.com/resources/webinar/model-repository/cloud/introduction/introduction-cloud-services.html

You can also watch this webinar, which demonstrates setting up a MySQL repository on the Amazon Web Services (AWS) Cloud using Amazon Relational Database Services (RDS), and creating and configuring an Enterprise Architect project in the repository via the Cloud connection.

http://sparxsystems.com/resources/webinar/partners/cloud-server/amazon/deploy-enterprise-architect-model-on-amazon-cloud-aws.html

Cloud Server Installation

This information applies to both the free Sparx Systems Cloud Service and the separately-purchased editions of the Pro Cloud Server (which incorporate the Sparx Systems Cloud Service). The basic installation process is quite straightforward and you, as the System or Network Administrator, can install and/or configure some of the components at a later time, allowing you to set up your working service in stages.

If you have been using the Sparx Systems Cloud Service or the Sparx Systems WAN Optimizer and you want to replace it with Pro Cloud Server on the same machine, it is recommended that you specifically uninstall the Sparx Systems Cloud Service and/or Sparx System WAN optimizer to avoid any conflicts. The Cloud Services components of Pro Cloud Server will use the same configuration, directories and ports as the original installation.

- If you want to install the Sparx Systems Cloud Service, you can download the installer from the 'Registered User' section of the Sparx Systems website:

  http://www.sparxsystems.com.au/registered/index.html#cloud-services

  Click on the 'Download and try today' link; a Cloud Services icon is added to your desktop

- If you have purchased an edition of the Pro Cloud Server, you will receive an email explaining the download procedure for the installer, which will also be held in the 'Registered User' section of the website and, when downloaded, will add a Pro Cloud Server icon to your desktop

Both installing the Cloud Service and editing the configuration files require you to have Administration rights; right-click on the Cloud Services icon or Pro Cloud Server icon and select 'Run as Administrator'.

The 'Sparx Systems (Pro) Cloud Services Setup' dialog displays.

Installation Steps

The 'Sparx Systems (Pro) Cloud Services Setup' dialog helps you to install the components of the Cloud Service. When it displays:

1. Click on the Next button.

2. Select the option to accept the End User Licence Agreement.

3. Click on the Next button on that page, and on the next two pages of the dialog, to display the 'Select Features' page.

4. For a simple default installation on the one machine, go straight to step 6.

5. If you want to install or block specific components to, for example, have different components on separate machines, refer to the *Cloud Server Components* table and the *Installation Options* table, then return here to step 6.

6. Click on the Next button. The system installs the Cloud Services components and displays a message to indicate success.

7. Click on the Finish button. Now refer to the *Cloud Server Configuration* Help topic.

Cloud Server Components

The 'Select Features' page lists three (Cloud Services) or five (Pro Cloud Server) components. Each component can be controlled separately by clicking the small image in front of the component name and selecting the option you require.

Component	Description
Sparx Cloud Services / Sparx Pro Cloud Services	This component is the Windows service that will accept connections from Enterprise Architect and the management client. For Pro Cloud Server, this includes the Sparx Systems Keystore Server. The default option is to install this component.
	Generally, the only time you would block installation of this component is when

	you have already installed it on one machine and are now installing other components on a different machine.
Cloud Management Client / Pro Cloud Management Client	The Management Client supports the execution of management tasks, including the configuration of which databases a Cloud Service should connect to, and setting a number of server options.
	The default option is to install the component; Sparx Systems recommends that the Management Client should be installed whenever the Cloud Service is installed.
WebEA	(For a Pro Cloud Installation); this option installs the WebEA application files (PHP, CSS and Javascript) that enable a web user to view an Enterprise Architect model on a mobile, android or remote desktop machine without having to install Enterprise Architect on that device.
	The default option is to install this component; you can unselect this component if you do not plan to use WebEA.
RLS	(For a Pro Cloud installation) Row Level Security; this option installs a set of SQL scripts to be applied to your MS SQL Server or Oracle based Enterprise Architect models that require the Visibility Levels feature.
	The default option is install this component; you can unselect this component if you do not plan to use the Visibility Level functionality.
IIS Integration	An optional component providing integration with IIS.
	The default option is to not install this component; however, if your server requires advanced authentication methods available through IIS then this component should be installed.
Worker	(For a Pro Cloud Installation); this option installs a special reduced-functionality version of Enterprise Architect that has the sole purpose of generating diagrams and linked documents that are viewable by the WebEA application. The Worker can be configured to run against a single repository every x number of seconds.
	The main benefit of this component is to allow clients to use builds of Enterprise Architect prior to build 1350 and still have their changes viewable in WebEA. However, if your entire user base is using Enterprise Architect build 1350 (or later) then the Worker feature is not required.
	The default option is to install this component; you can unselect this component if you do not plan to use Worker.

Installation options

Option	Description
Install feature locally	Select to install the feature on the current machine.
Install entire feature	Select to install the feature and all items that it owns in the tree; effectively the same as 'Install feature locally'.
Install on demand	An msi command that only installs the feature when a user tries to access it; not recommended for any Sparx Systems Cloud Services features.

| Don't install | Select this option to not install a component/feature. |

Cloud Server Configuration

This topic discusses the configuration of the Cloud Server itself, whether as the original Sparx Systems Cloud Services or as a component of the Pro Cloud Server. If you are installing the Pro Cloud Server, there are several other configuration files associated with the Keystore, WebEA and Row Level Security components of the server. The procedures for editing those files are described in the separate topics for those features.

Many configuration settings for the Cloud Server are set by directly editing the configuration file **SSCloudServices.config**, which is found in your Service directory. To edit the configuration file, open it in a text editor (using the 'Run as administrator' option).

Once you have opened the file, you can edit it to set configuration options, including the ports the server will listen on.

There are two different types of configuration option defined in the Cloud Service configuration file. The first type are global and apply to the entire Cloud Service, while the second type (which are enclosed in parentheses) apply to an individual port that the Cloud Service will listen on. Typically there are multiple port definitions, each enclosed in their own set of parentheses.

Note: It is important to realize that the Cloud Service only reads the configuration file once, when it starts up. If you manually change the options in the configuration file then the Cloud Service must be restarted.

Configuration Client Connection Settings

Normally the first settings you will see in the configuration file are to control how the Configuration Client will connect to the server. The default values are:

> SERVER_PORT=803

> SERVER_PASSWORD=

Use of the Configuration Client is discussed in the *Cloud Services Configuration Client* topic.

Setting	Description
SERVER_PASSWORD	**SERVER_PASSWORD** is the password to protect the administration functions of the server. Note: This can be changed within the Configuration Client, which means a full server restart will not be necessary.
SERVER_PORT	**SERVER_PORT** is used when you connect to the Configuration Client or opt to use the IIS integration instead of the integrated web-server. For more detail see the *Cloud Server Using IIS* topic. Note: When changing this, check firewall settings and other services using the port. Additionally, when the service is running on Wine you should not use privileged ports - those below 1024. Under Wine, most applications are only able to listen on ports above 1024.

General Settings

The next list of settings includes the default global settings across the entire service:

> DBMAN_DEFAULTMAXSIMQUERIES=10

> AUDIT_TIME_PERIOD=3600

> TEMP_DIRECTORY=%SERVICE_PATH%\Temp

> LOGGING_LEVEL=SYSTEM

LOGGING_DIRECTORY=%SERVICE_PATH%\Logs

LOGGING_FILECOUNT=3

LOGGING_FILESIZE=1048576

Setting	Description
DBMAN_DEFAULTMAX SIMQUERIES	The default for the maximum number of simultaneous queries that can be run at a time for any configured database. This can be changed directly within the Management Client (see **Default Max Simultaneous Queries** under *Global Server Options* in the *Cloud Services Configuration Client* topic). There is no explicit limit for this default value. Note: As this can be set directly within the Configuration Client, a full server restart will not be required if you change it.
AUDIT_TIME_PERIOD	The number of seconds between the recording activities by system logs on each database.
TEMP_DIRECTORY	The location to write temporary files before they are sent to clients. You should not generally have to change this.
LOGGING_LEVEL	Determines how verbose the server should be when writing log files. The valid values are: OFF, FATAL, WARNING, INFO and SYSTEM. The value can be changed directly within the Management Client. (See **Log Level** under *Global Server Options* in the *Cloud Services Configuration Client* topic). Note: As this can be set directly within the Configuration Client, a full server restart will not be required if you change it.
LOGGING_DIRECTORY	Defines where the log files are to be stored. The default is set to: =%SERVICE_PATH%\Logs Note: The =%SERVICE_PATH% refers to the directory where the Cloud service is installed.
LOGGING_FILECOUNT	Determines the maximum number of rolling log files that should be kept. When the file count is exceeded, the oldest file is automatically deleted.
LOGGING_FILESIZE	Determines the maximum file size of each log file. When the logging file size is reached a new log file is created. For more details on using the logs see the *Cloud Server Troubleshooting* topic.
LICENSE	(For a Pro Cloud Installation); this option defines the licensing information of your Pro Cloud Server and will be supplied to you after your purchase has been processed correctly.

Port Settings

Using the Cloud Server you can define an arbitrary number of different ports on which to listen for connections from Enterprise Architect, each with a different configuration. Each port is denoted in the configuration file, with open and close parentheses, on their own lines.

(

SERVER_PORT=804

REQUIRE_SSL=0

DEFAULT_MODEL=

MODEL_AUTHENTICATION=

GLOBAL_AUTHENTICATION=user model

OSLC_SUPPORT=1

)

Setting	Description
SERVER_PORT	The port on which the server will listen for HTTP connections; each connection must be unique and not used by any other services on the machine. You must check that no firewalls are blocking this port on the client or server. You can use the standard HTTP port (80) or HTTPS port (443). Note: When changing this, check firewall settings and other services using that port. Additionally, when the service is running on Wine you should not use privileged ports - those below 1024. Under Wine, most applications are only able to listen on ports above 1024.
REQUIRE_SSL	Should be set to **1** to enable HTTPS on this port; HTTPS should be enabled for all connections that are being exposed on public networks. HTTPS requires a private key file (**server.pem**), to be included in the same directory as the configuration file, before it will run. Note: This unique file must be user-created. See *Creating a Self-Signed Certificate using OpenSSL*.
DEFAULT_MODEL	Allows a single model to be exposed on a port, making it possible to use a different port for each model. Model names are discussed further in the *Connecting Enterprise Architect to a Cloud Server* topic.
MODEL_AUTHENTICATION	Can be set to **1** to request HTTP authorization using the user security defined in the Enterprise Architect model being connected to. Passwords must be explicitly and individually assigned in that model using the *Maintain Users* procedure; the default administrator password and any passwords imported from Windows Active Directory do not work. Note that if you are not using SSL to connect, the usernames and passwords will be sent in plain text. If the model does not have security enabled, the Cloud user is not prompted for a password. This option is mutually exclusive with **GLOBAL_AUTHENTICATION**.
GLOBAL_AUTHENTICATION	Can be set to the name of an Enterprise Architect model with security enabled that will provide the list of users for all models accessed by the connection. This is helpful if you want to provide multiple models but only manage one list of users. Passwords must be explicitly and individually assigned in the reference model using the *Maintain Users* procedure; the default administrator password and any passwords imported from Windows Active Directory do not work. This option is mutually exclusive with MODEL_AUTHENTICATION.
OSLC_SUPPORT	Enabled by default. It allows models to be queried using the 'Open Services for Lifecycle Collaboration' standard. This is discussed further in the *OSLC Requirements Management* topic. Set to **0** to disable.

Restarting the Sparx Cloud Server

If you make any changes to the configuration file you must restart the server for the changes to take effect (unless otherwise stated). A server restart is carried out in the Windows Services application.

Depending on the server operating system, there are two methods for restarting the Cloud Server:

1) Using Window Services. This is available in all versions of Windows (see 'Control Panel | Administrative Tools | Services').
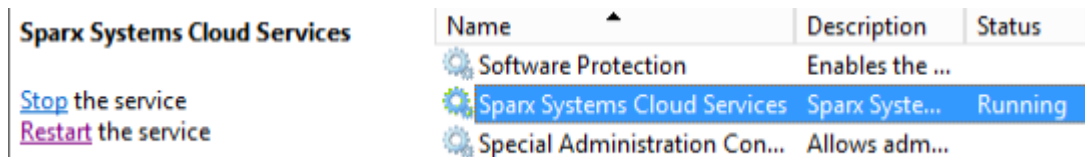
Figure 8: Start and Stop options for Cloud Services in the Windows Services view

2) Using the Server Manager on Windows Server 2012.
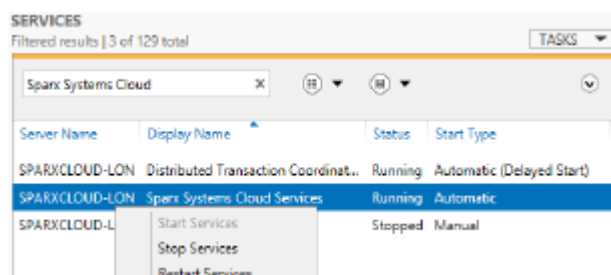
Firewall Settings

When setting up a server, you must check that the Firewall on the server is set to allow the incoming ports for the database connections that you have created.

For example, in the SSCloudServices.config file the ports default to 804 and 805 and are set as operative. If you have a firewall you must set these ports as enabled for inbound traffic.
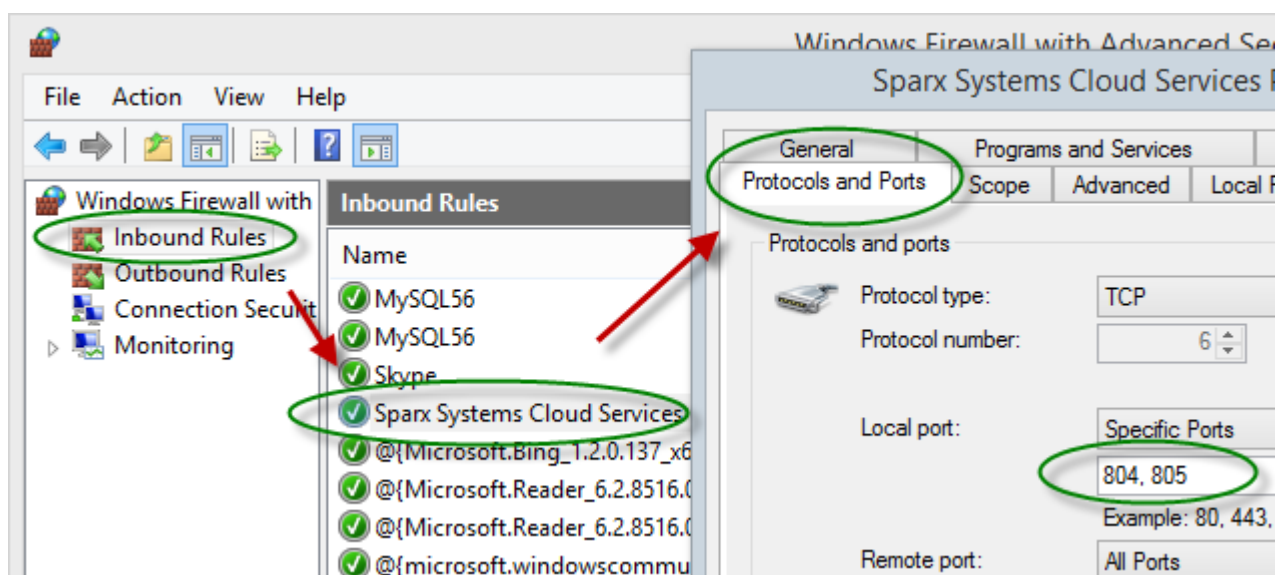
For information on creating the Inbound Rule to identify the ports you have specified in the config file (the 'Sparx

Systems Cloud Services' rule in the illustration), see the Windows online topic [Create an Inbound Port Rule](#).

Note: A common cause of failure is that other services are already using the allocated ports. This is especially likely when using the default http (80) and https (443) ports. Check that no other services are using the allocated ports.

Using Multiple Configurations

The Sparx Systems Cloud Service can be run multiple times with different configurations by passing these additional arguments.

Argument	Description
--config / -c	The argument following this will be interpreted as the path to the configuration file to read on startup. This allows a secondary instance of the cloud server to run on a different set of ports, use an altered logging level or different directories.
	When multiple configurations are being used, you need to ensure that each instance reads a configuration file that specifies different paths. Otherwise only the first instance will start successfully.
	By default this references SSCloudServices.config in the directory containing the service executable.
--path / -p	The argument following this will be interpreted as the absolute path to search for other resources. Changing this parameter changes where the configuration file is read from, along with all paths in the configuration file that specify %SERVICE_PATH%. Setting this to a directory other than the default is an easy way to separate multiple running instances of the cloud service.
--registry / -r	The argument following this will be interpreted as a registry path within HKEY_LOCAL_MACHINE where the models the cloud server provides are specified. Changing this parameter allows using different cloud servers to provide access to different models.
	By default this is Software\Sparx Systems\SQLBridge.
standalone	This argument allows starting the SSCloudServices.exe as a standalone executable instead of requiring it to be launched as a Windows service.
	This can be important if Windows firewall is blocking the service, as it will prompt to create a firewall exception.
	Additionally, Wine by default closes services automatically when the last user process closes, but using the 'standalone' argument to run the server as an application will ensure that the service isn't terminated when no more user applications are running. Use the command line:
	wine SSCloudServices.exe standalone
	Or, if you are launching from a terminal and want to close it:
	wine SSCloudServices.exe standalone & disown or
	nohup wine SSCloudServices.exe standalone
	(The 'disown' parameter and 'nohup' command both close the terminal but ignore the 'hangup' message that would otherwise disconnect Cloud Services again.)

Self-Signed Certificate using OpenSSL

In order to use a secure connection (HTTPS) to your model, a server certificate is required. For a production environment, particularly one providing access to external users, you should obtain a certificate from an appropriate certificate authority. However, to help with initial setup and testing purposes you can generate your own certificate, using these instructions.

The Cloud Services installer provides a simple batch file (openssl.exe) that assumes that OpenSSL is available on the windows path and is appropriately configured. You can create a second batch file into which you paste this code, and run it with the target hostname as a parameter. It will generate an appropriate server.pem key file, which can then be placed in the service install directory with the SSCloudServices.config file (..\Sparx Systems\Cloud Services\Service).

```
echo off
if not "%1" == "" goto generate
echo ERROR: No target specified
echo USAGE: %0 url
echo EXAMPLE %0 localhost
goto end
:generate
echo on
openssl genrsa -out %1.key 2048
openssl req -new -x509 -key %1.key -out %1.cert -days 3650 -subj /CN=%1
copy /b %1.cert+%1.key server.pem
:end
```

Notes

If you need to assign passwords to user IDs for a model or models that will be accessed via the Cloud, then:

1. Open the reference model using a direct connection or via a Cloud connection on a port that does **not** have either MODEL_AUTHENTICATION or GLOBAL_AUTHENTICATION set.

2. Enable security and assign a new administrator password.

3. Open the Cloud Server Management Client for the new database and set the checkbox 'Require a secure and authenticated connection'. (Now that you have a valid account, this model will no longer be accessible without https and http level authentication.)

4. Connect to the model on a port that **does** have an _AUTHENTICATION setting and use the *Maintain Users* procedure to assign passwords to the user IDs in the model.

Cloud Services Configuration Client

At any point after installation you can connect to the Cloud Service using the **Cloud Services Configuration Client** (SSCloudServicesClient.exe).

By default this is accessible from: '...\Program Files (x86)\Sparx Systems\Cloud Services\Client'.

The Cloud Services Configuration Client can be used from any machine on your local network; however, communications between it and the Cloud Service are not secure, so it should not be used from outside a secure network.

The Cloud Services Configuration Client is used to configure various settings related to the Cloud service and the repositories it connects to. When the application first starts, a prompt displays to log into a Cloud service (see *Logon screen* for more details). Once you have logged in, the 'Sparx Systems Cloud Services Configuration Client' dialog displays.

The Cloud Services Configuration Client dialog

This dialog shows a list of all defined repositories for the Cloud Server and, for each one, the number of active user connections and the total number of connections possible. It is also the entry point for all management functions; that is, new repositories can be created, existing ones edited and old ones removed.
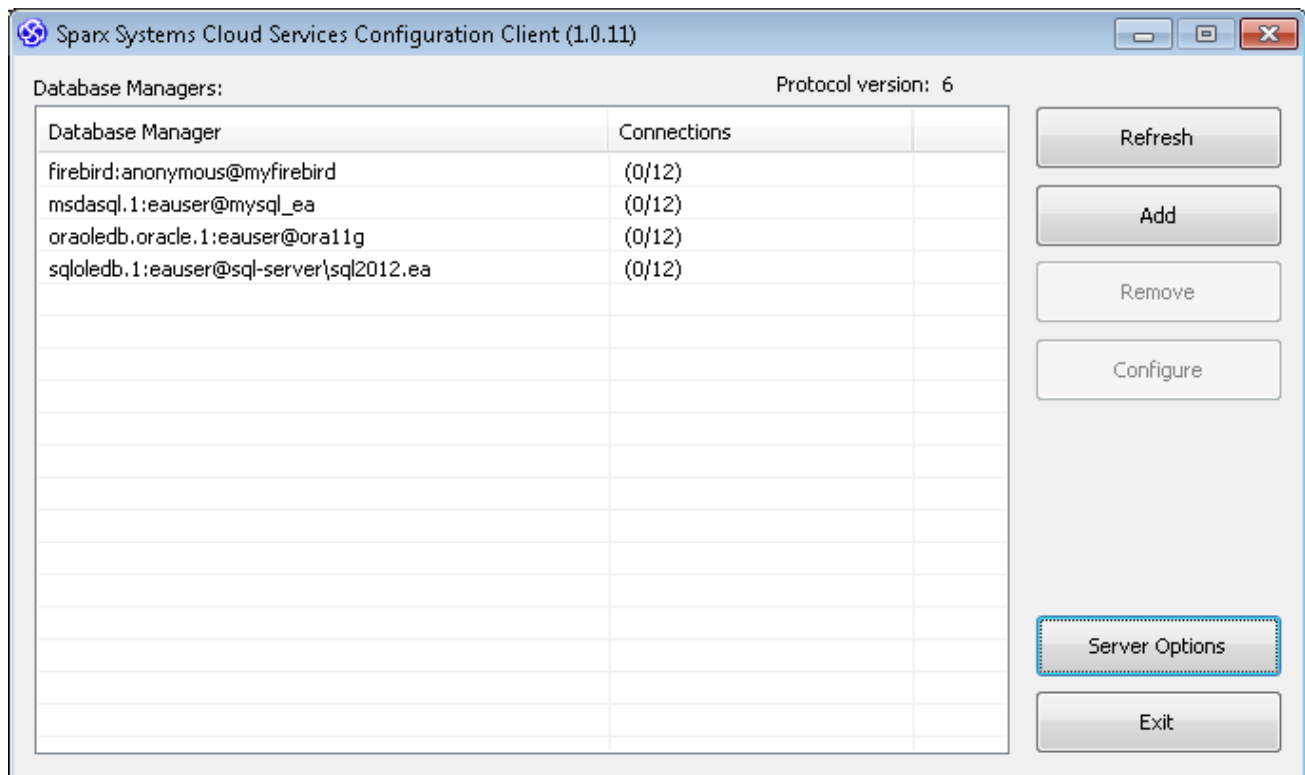
Sparx Systems Cloud Services support connections to these DBMS products:

- Firebird

- Microsoft SQL Server

- MySQL

- Oracle

- PostgreSQL

Item	Description

Database Managers	This panel lists the defined repositories. Each row contains two columns of information - the database connection string of the repository and the number of active and available connections.
Refresh	Click on this button to reload the contents of the 'Database Managers' list.
Add	Click on this button to display the dialog on which you define a new database.
Remove	Click on this button to permanently delete the selected database from the Cloud Server.
Configure	Click on this button to display the dialog on which you change the settings for the currently selected database.
Server Options	Click on this button to display the dialog on which you change the Cloud Server global options.

Cloud Services Logon Screen

The Cloud Services Logon screen is the first to be shown when launching the Management Client, and at a minimum requires the address and port of the Cloud Server to connect to.
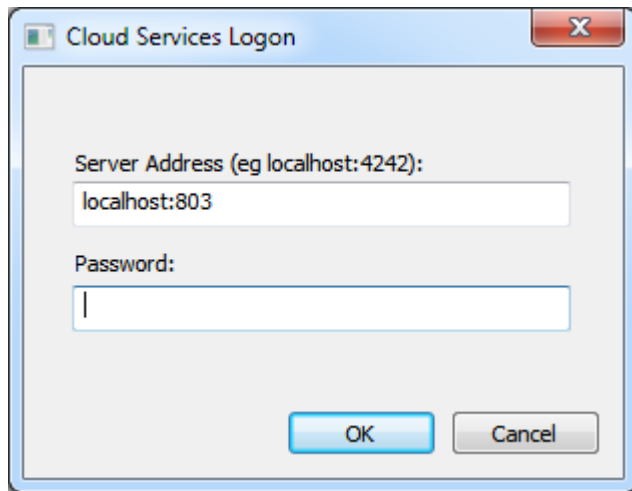
Cloud Services Logon

Option	Description
Server Address	The **Server Address** format is: *<ServerURI>:<Port>* The server URI can be 'localhost', an IP address, a DNS name or a machine name; for example, MyWebServer. If you are working on the web server, 'localhost' will be sufficient.
Password	Enter the password you defined in *Management Client Connection Settings*, in the *Cloud Server Configuration* topic. By default this password is blank; you can reset it to a new value, as described in the Help topic *Global Server Options*.
OK	Click on this button to log in to the Management Client.

Add Database Manager

The Add Database Manager screen displays when you click on the Add button on the 'Sparx Systems Cloud Services Configuration Client' dialog; it helps you to define a new database connection to either a local Firebird repository or a DBMS repository (via ODBC or OLE/DB).
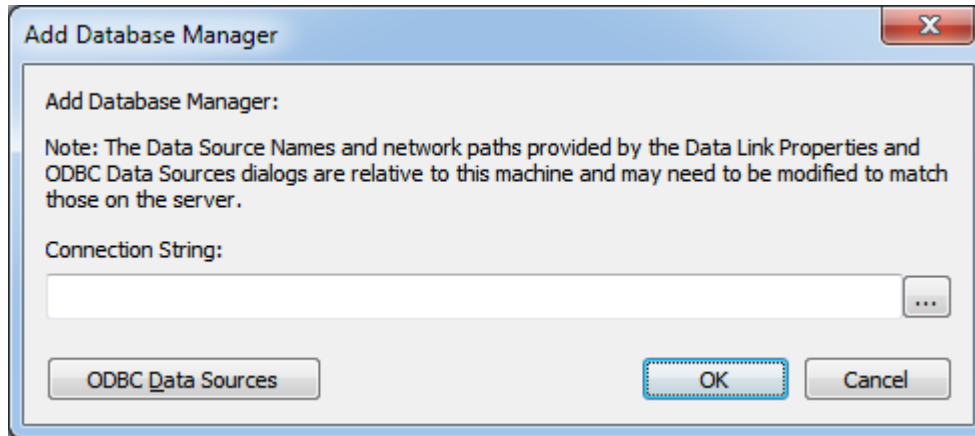
This note is shown on the Add Database Manager screen:

"Note: The Data Source Names and network paths provided by the Data Link Properties and ODBC Data Sources dialogs are relative to this machine and may need to be modified to match those on the server."

This means if the Cloud Services Client application is running from a remote machine (that is, not the machine running

the Cloud Server) the screens that are shown by clicking on the [...] button and ODBC Data Sources button are relative to the current machine and not the Cloud Server machine, so care should be taken.

By default and as a safety precaution, whenever a new Database is defined it is automatically configured to be disabled. You can enable the database during definition by selecting the 'Enabled' checkbox on the Configure Database Manager screen.

Adding New Databases
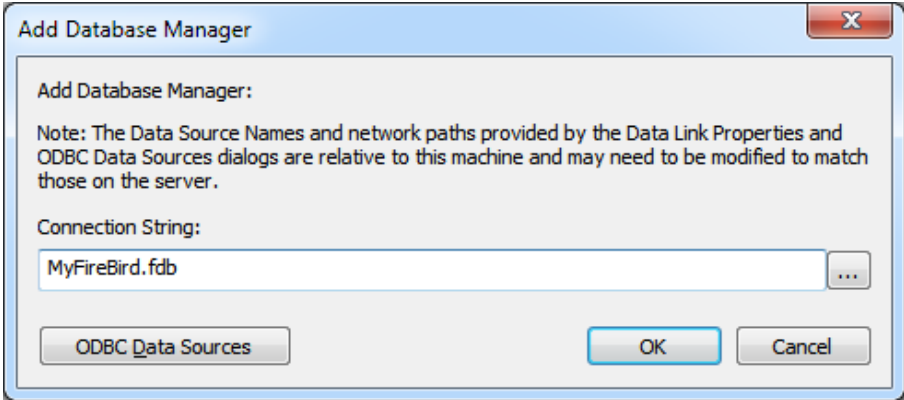
It is important to note that the Cloud Server uniquely identifies Databases by a single identifying value; this value is referred to as the database 'Alias'. For:

- Firebird definitions the 'Alias' is the filename minus the extension

- ODBC based definitions the 'Alias' is the 'Data Source'

- Oracle OLE DB based definitions the 'Alias' is the 'net service name'

- SQLServer OLE DB based definitions the 'Alias' is the 'Initial Catalog'

Therefore, take care when defining database connections, to ensure that you do not create duplicate database aliases. However, if two or more database definitions do share the same alias then all Enterprise Architect clients will connect to the first database found with the requested alias.

Item	Description
Firebird Database	Sparx Systems, in the 'Introduction to Sparx Systems Cloud Services' webinar, demonstrate the Cloud Services with a Firebird database as it is so easy to set up and use, and it works extremely well with Sparx Systems Cloud service technology. Firebird is a free, open source rational database that runs on a variety of platforms. Firebird is very sophisticated and offers great concurrency, scalability and performance. It can be used in an enterprise setting and has been used in production systems, under a variety of names, since 1981.

	To discover more about the Firebird database, please visit this website:
	http://www.firebirdsql.org/en/about-firebird/
	You can view the webinar by clicking on this link:
	http://www.sparxsystems.com/resources/webinar/model-repository/cloud/introduction/introduction-cloud-services.html
	You can define a connection to a Firebird database by entering its filename (and extension) only; for example,'MyFireBird.fdb'.
	The Cloud Server assumes all Firebird databases physically reside under the directory %SERVICE_PATH%\Models\. The Cloud Server is, by default, installed to the folder of:
	...\Program Files (x86)\Sparx Systems\Cloud Services\
	In this case all Firebird models will be located at:
	...\Program Files (x86)\Sparx Systems\Cloud Services\Models\
ODBC Database	You can define a connection to a database under a DBMS such as MySQL or PostgreSQL, by specifying the connection string to the database. If you are running the Management Client on the same machine as the server you can:
	1. Click on the [...] button to open the 'Data Link Properties' dialog to build the connection string.
	2. Select the provider 'Microsoft OLE DB Provider for ODBC Drivers' and click on the Next >> button.
	3. Select the pre-defined System DSN from the 'Use data source name' field.
	4. Fill in the user name and password and select the 'Allow saving password' checkbox.
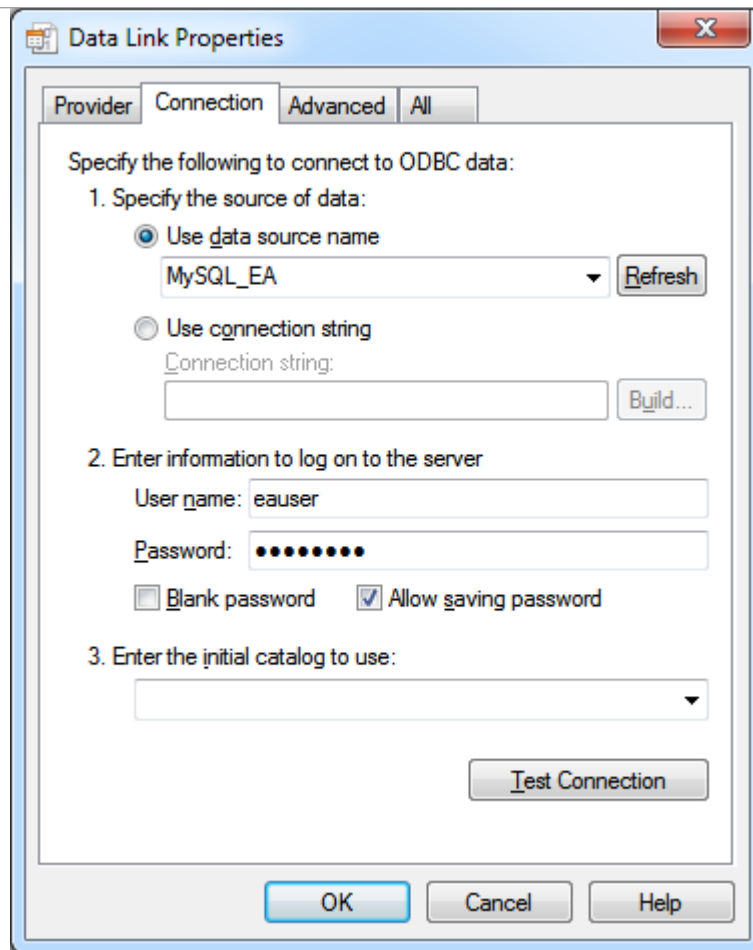
Notes:

- Define all ODBC DSNs as SYSTEM DSNs, because the Cloud Server runs as a Windows Service and User DSNs might not be available to the Cloud Server
- The step for filling in the user name and password might not be required if the user name and password have been saved with the ODBC DSN
- The ODBC Data Sources button displays the Windows 'ODBC Data Source Administrator' dialog, which you can use to set up a new ODBC system data source to be listed on the 'Data Link Properties' dialog (if you are running the administration client on the same machine as the server); it is particularly helpful because it shows the 32-bit ODBC data source list that matches Enterprise Architect and the Cloud Services

Oracle Database (via OLE DB)	You can define a connection to an Oracle Database (via OLE/DB) by specifying the connection string to the database. If you are running the Management Client on the same machine as the server you can:

1. Click on the [...] button to open the 'Data Link Properties' dialog to build the connection string.
2. Select the provider 'Oracle Provider for OLE DB' and click on the Next >> button.
3. Type in the name of the 'Data Source'; this is the Oracle net service name as defined in the TNSNAMES.ORA.
4. Fill in the user name and password and select the 'Allow saving password' checkbox.

Note:

- The Cloud Server identifies databases by the 'Data Source' value and Oracle OLE DB defines the 'Data Source' as the net service name, so it is only possible to access one Oracle repository per net service name; the workaround to this limitation is to define multiple net service names in TNSNAMES.ORA that differ only by name

SQLServer Database (via OLE DB using SQL Server authentication)	Microsoft SQL Server supports two different forms of database authentication: - Windows NT authentication, and - SQL Server authentication The Cloud Service will work with either, but SQL Server authentication is easier to configure. Prior to defining the Cloud Service connection, define a SQL Server user with db_datareader and db_datawriter roles to the database that will act as the Enterprise Architect repository. You can define a connection to a SQLServer Database by specifying the connection string to the database. If you are running the Management Client on the same machine as the server you can: 1. Click on the [...] button to open the 'Data Link Properties' dialog to build the connection string. 2. Select 'Microsoft OLE DB Provider for SQL Server' and click on the Next >> button. 3. Select or type in the instance of SQL Server.

4. Select the 'Use a specific user name and password' checkbox and fill in the user name and password, then select the 'Allow saving password' checkbox.

5. Select the database that is the Enterprise Architect repository in 'Select the database on the server'.
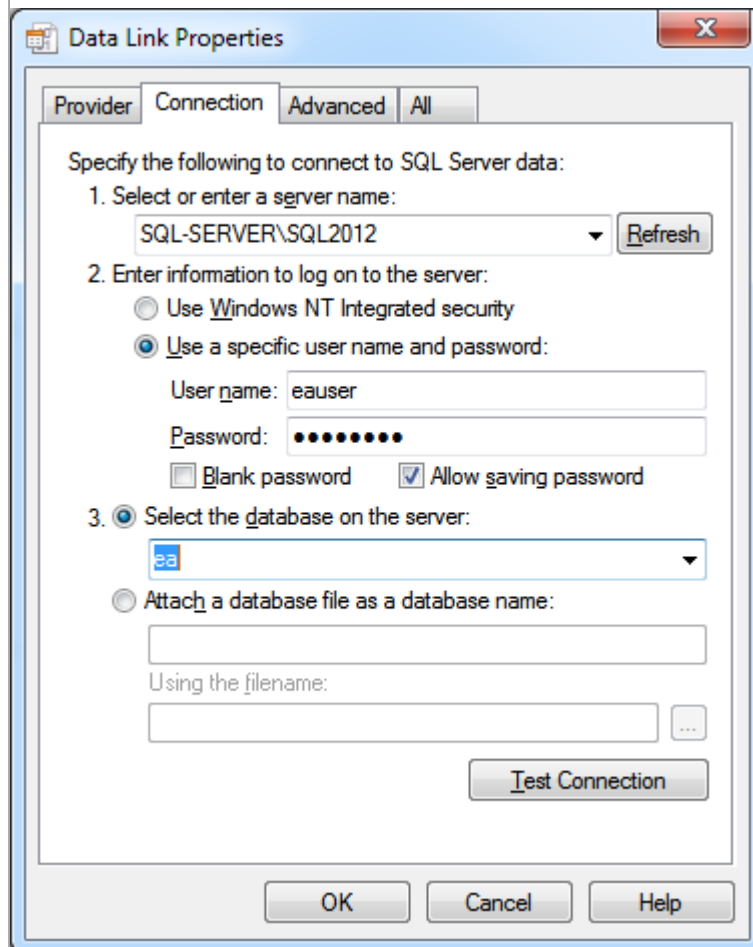
SQLServer Database (via OLE DB using Windows authentication)	Microsoft SQL Server supports two different forms of database authentication:
	• Windows NT authentication and
	• SQL Server authentication
	The Cloud Service will work with either, but Windows authentication requires some special consideration. The Cloud Server runs as a Windows Service, and by default all Windows Services run as 'Local System Account'; therefore, if Windows authentication is to be used, the Sparx Systems Cloud Service should be configured to run under a predefined network user (configured with the Windows Services console) and this network user must have access rights to the database (configured with Microsoft SQL Server Management Studio or similar).
	You can define a connection to a SQLServer Database by specifying the connection string to the database. If you are running the Management Client on the same machine as the server you can:
	1. Click on the [...] button to open the 'Data Link Properties' dialog to build the connection string.
	2. Select the provider 'Microsoft OLE DB Provider for SQL Server' and click on the Next >> button.
	3. Select or type in the instance of SQL Server.
	4. Select the 'Use Windows NT Integrated security' checkbox.
	5. Select the database that is the Enterprise Architect repository in 'Select the

database on the server'.
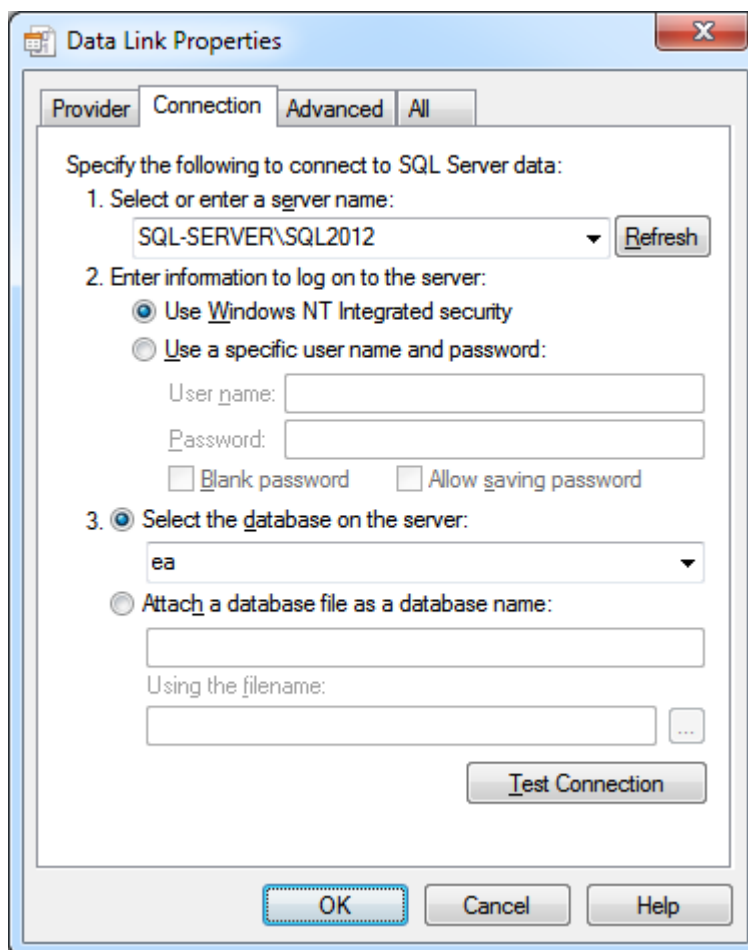
Note:
- Because the Cloud Service is run under a single account, all database access will be under that account, so it will not be possible to keep track of what records are updated in the database by individual users

Configure Database Manager

The Configure Database Manager screen is shown when you click on the Configure button on the 'Sparx Systems Cloud Services Configuration Client' dialog. It helps you to modify the various options for the selected Database.

Options

Option	Description
Database Manager	Read-only display showing the description of the database as shown in the 'Database Manager' List. The format of the description quickly describes the database, as shown here: {odbc provider}:{user}@{datasource}
Enabled	Select this checkbox to allow the Cloud service to respond to requests for the connection.
Alias	The database Alias. This is the value all Enterprise Architect clients will need to select in the 'Model Name' field of the 'Cloud Connection' screen. If necessary, you can edit this Alias to enable users to reference it from Enterprise Architect and

	WebEA.
Database Type	Read-only display showing the DBMS of the database, such as Firebird, MySQL, Postgres, SQLSvr or Oracle.
Connection String	Read-only display showing the connection string used to connect to the database.
Max Simultaneous Queries	Type in the maximum number of connections that can be open. You can use the up/down arrows to adjust this figure to balance resource usage with concurrent user performance. There is no explicit limit for this value, but the higher the value the more resources are consumed on your server, without a matching increase in benefit in user performance. Connections are held so briefly that the reduction in user wait time due to more connections being opened is not usually noticeable. If you are reaching the number of connections you have defined here, increase it by small increments (units) rather than large ones (tens).
Minimum EA Build	Optional. If you want to ensure that all users of the model are on an appropriate build of Enterprise Architect, type in the lowest build number required.
OSLC Access Code	Optional. Type in the access code that all OSLC clients, including WebEA, are required to send in an HTTP header. Leave blank to indicate that the OSLC clients are not required to send an access code.
Require a secure and authenticated connection	Select this checkbox to ensure this database is only used by https connections using http authentication.
Read-only connection	Select this checkbox to prevent users from modifying the database referenced by this connection.
Enable extended OSLC interface for WebEA	Select this checkbox to enable the model for access by WebEA through the Pro Cloud server.
Save	Click on this button to save the settings on the dialog.
Run Scheduled Tasks	Select this checkbox to enable the service to update Time Series charts on a schedule, or any other scheduled tasks that might be supported in the future.

Global Server Options

The Configure Server screen displays when you click on the 'Server Options' button on the Main screen. It helps you to configure the global options applied to the Cloud Server. Each of these options can also be set in the configuration file; however, the service does not require a restart if the options are changed using this screen.

Options

Option	Description
Protocol Version	Read-only display showing the highest level of protocol supported by this version of the server.
Audit Time Period	Read-only display showing the number of seconds paused between audit reports.
Logging File Count	Read-only display showing the number of log files that should be retained. All logging is written to files named 'SparxCloudServicesLog-X.log' where X represents the file number. File 'SparxCloudServicesLog-1.log' is always the current file. When the maximum file size is reached, all existing files are renamed by incrementing the file number and a new 'SparxCloudServicesLog-1.log' created.
Logging File Size	Read-only display showing the maximum file size of each log file. When the file

	reaches this maximum, a new file is created.
Log Level	Click on the drop-down arrow and select the level of information to be written to the log file.
Logging Directory	Read-only display showing the physical directory into which the log files are saved.
Temp Directory	Read-only display showing the physical path of the temp directory.
Default Max Simultaneous Queries	Provides a default value for the setting of 'Max Simultaneous Queries' when defining connections for a new database configuration.
Set New Password	Select this checkbox to change the password for using the admin client for this server; type in and confirm the new password.

Connecting Enterprise Architect to a Cloud Service

Once your Cloud Service has been set up with at least one port listening for communication and at least one model you can connect to, you can connect to the model in Enterprise Architect. When you first open Enterprise Architect, the 'Manage Projects' dialog displays. The Connect to Cloud button on the top right provides access to Cloud connections.
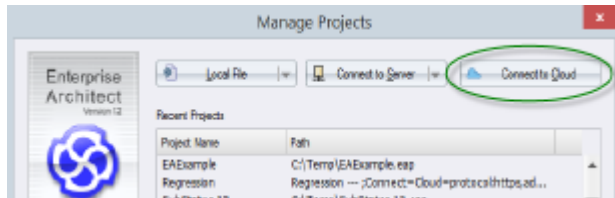
There is also a Cloud Connection link on the 'Start Page'.
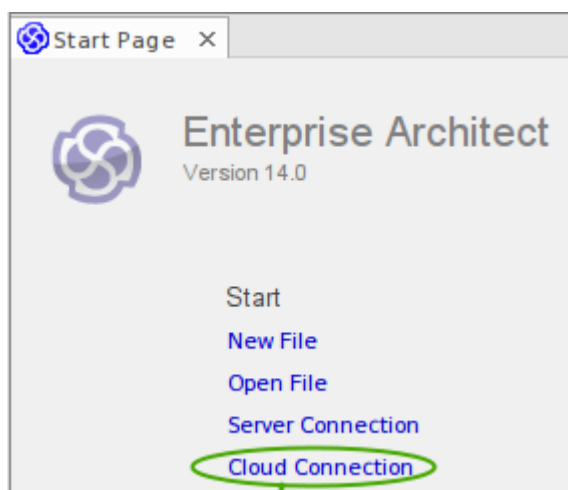
Cloud Connection Dialog

The 'Cloud Connection' dialog prompts you for the details of the model.

Option	Action
Name	Type the name by which to refer to this model. It does not have to match any values on the server, and will appear in your recent models list.
Protocol	Click on the drop-down arrow and select a secure (https) or insecure (http) connection, as supported by the server you are connecting to. When you select the protocol, the port will be automatically set to match the default for that protocol.
Server	Type in the machine name (sparxcloud.com) or IP address (192.168.0.20) of your server.
Port	Define the port number the service is listening on. This should match the settings in the configuration file on the server.

Note: For a server, the default settings will use port 804 for http and port 805 for https. |
| Model Name | Type in the name of the model available on the server.

Note: Generally a system administrator will provide you with a list of available models. However, for those with permission to use the Management Client the model name to access the given database can be found in the read-only 'Alias' field on the Configure Database Manager screen in the Cloud Services Configuration Client application. |

Pro Cloud Server

Sparx Systems Pro Cloud Server is an enhanced, separately-purchased and licensed edition of Sparx Systems Cloud Services. Once the Cloud Service has been installed and configured, it runs on the installed machine as a Windows service, listening for requests for data from Enterprise Architect.

In order to support the new Pro Cloud Server functionality, the service has these enhancements:

- The ability to define individual database managers as 'Enable extended OSLC interface for WebEA'
- The 'Configure Server' screen in the Configuration Client displays the number of Pro model licenses and their expiry dates
- The ability to define a 'Default Schema' option against individual database managers

As well as providing all the features and benefits of Sparx Systems Cloud Services, the Pro Cloud Server also includes:

- The Sparx Systems Key Store and a floating license server
- WebEA, a web based application that allows users to view and discuss elements in Enterprise Architect models through a web browser, while on portable devices or remote desk-top systems
- Row Level Security for Oracle and SQL Server based models, so that it is possible to restrict the access of a user or group of users to parts of the model that are only relevant to them

Sparx Systems Pro Cloud is available in three editions, to support small, medium and large projects with differing numbers of models; the editions have the same installer but require different access keys to install them:

- Small Business, which supports up to three Firebird repositories; Row Level Security is not available in this edition, as the feature is for Oracle and SQL Server databases
- Team Server, which supports up to five repositories of any type
- Enterprise Server, which supports any number of repositories of any type, within any restrictions imposed by your server and network

Installation

The Sparx Systems Pro Cloud Server product is installed using a standard Windows Installer package. While it is possible to change the install location of the Pro Cloud Server, most users elect to keep the default of C:\Program Files (x86)\Sparx Systems\Cloud Services\WebEA\ (assuming a 64bit machine is being used).

The base components of the Pro Cloud Server will create three sub-folders inside the installation folder. These are:

- Client Folder, containing the client GUI to configure the Cloud Service
- ISAPI Folder, containing runtimes of APIs needed by the Cloud Service for IIS integration
- Server Folder, containing the Windows service and a folder each for log files and local databases

Depending on what options are selected while installing Pro Cloud Server, there could also be other folders. The additional folders will be discussed in the installation and configuration topics for the base Pro Cloud Server and for each specific feature.

WebEA

One of the most requested features from the Enterprise Architect user base recently has been the ability to access Enterprise Architect models on mobile devices or from desktops without the need to install the Enterprise Architect application.

Sparx Systems has developed WebEA using PHP, Javascript and CSS to provide a solution that can be deployed easily, that runs on the widest variety of systems, and that doesn't require any special hardware or additional licensing.

WebEA has a large array of configuration options that give the Enterprise Architect customer the power to control

exactly how each model can be accessed and what types of actions can be performed.

Key Store

The Sparx Systems Key Store and floating license server has been available for many years, providing users of Enterprise Architect with the ability to manage their floating licenses. Up until the creation of Pro Cloud Server, the Key Store was only offered as an independent Windows service. Now that it has been integrated into the Pro Cloud Server, users only need to install, configure and maintain one service for all their Enterprise Architect needs.

Visibility Levels

The 'Visibility Levels' feature within Enterprise Architect gives the model managers the ability to restrict access to certain sections of a model. Only models hosted by a Pro Cloud Server are able to make use of this feature. Visibility Levels are implemented by taking advantage of the Row Level Security feature included in Oracle 8+ and Microsoft SQL Server 2016 (onwards).

Row Level Security is a security feature that controls which users can access which rows of data in a database. Traditional database security only allows for the ability to control which users have access to read, write or delete data down to the table level. Since Row Level Security functionality is implemented by the database, it is impossible for users to bypass security implemented at the application level by writing their own scripts or databases queries. That is, the database ensures that users can only view and update data they are authorized to access.

Since Row Level Security is functionality implemented by the database, most of the changes needed for Enterprise Architect and the Pro Cloud Server to support it are made directly to the database without needing to change the application code base, which means there is less to go wrong from an application point of view.

WebEA

Sparx System's WebEA is an application designed to display the data from Enterprise Architect models in a web browser, which means users can review and comment on a model in real time, on a range of mobile devices or a remote work station, without needing to install Enterprise Architect.

WebEA is a component of the Sparx Systems Pro Cloud Server, which is a separately-installed and licensed product, to complement Enterprise Architect. WebEA makes use of the PHP, HTML, CSS3 and javascript technologies and requires a web server (such as Apache) to host it. The topics of this chapter explain how to install and configure WebEA in detail, but if you are interested in getting started quickly see the *WebEA Quick Start Guide* for the necessary steps.

Requirements

- Users of WebEA requires a HTML5 / CSS3 compatible web browser

- The WebEA interface requires Javascript to be enabled in the web browser

- A client's device will require network connectivity to the web server hosting WebEA

- Users of WebEA will require this information in order to connect to and log into a WebEA model:
    - The complete URL; that is, machine name/IP number and optionally the port number
      and/or the path to WebEA files, depending on how the web server has been configured
    - The name of the model to access
    - (Optional) an access code necessary to open a model, if one has been configured
    - (Optional) the user ID and password required to open a model, if user security has been
      applied

WebEA Installation and Configuration

The Sparx Systems Pro Cloud Server product is installed using a standard Windows Installer package that includes a number of optional components. WebEA is one of these components and is installed by default. It is possible to change the install location of WebEA; however, if the defaults are used WebEA will be installed into the C:\Program Files (x86)\Sparx Systems\Cloud Services\WebEA\ folder (assuming a 64 bit machine is being used).

See the *Cloud Server Installation* and *Cloud Service Configuration* Help topics for a full description of how to install and configure the Sparx Systems Cloud Service.

Important: If you are installing the Pro Cloud Server on a machine that already has the Sparx Systems Cloud Services or Sparx Systems WAN Optimizer installed then these should be uninstalled first.

WebEA Server Requirements

In order to run WebEA in your environment you must have:

- A Web Server (that is, Apache or IIS) capable of hosting a PHP v5.0 (or later) website
- A Web Server with these standard PHP extensions installed and enabled: core, curl, date, gettext, hash, json, libxml, mbstring, pcre, session, standard and tokenizer
- A Web Server with PHP sessions enabled
- Communication between the web server and the server hosting the Pro Cloud Server
- A base folder (on the web server) to store the WebEA website pages

Configuration Steps

The installation of the WebEA application should have already been completed by following the steps outlined in the *Cloud Server Installation* topic, making sure that the WebEA component was installed.

In additional to installing the Pro Cloud server, WebEA also needs a web server to host it. In the simplest configuration the web server would be installed on the same machine as the Pro Cloud Server.

Configuring WebEA applies these steps:

Step	Action
1	Decide what models are going to be available to WebEA users.
	At the same time it is worth considering these points, which will determine the configuration options of each model.
	• Is user security currently configured on the model?
	• Should WebEA users be prompted for their model user and password?
	• Should WebEA users be able to view element discussions?
	• Should WebEA users be able to add element discussions?
	• Should WebEA users be able to join and participate in reviews?
	• Should WebEA users be able to insert new objects, and if so, what types?
	• Should WebEA users be able to add new element features, and if so, what types?
	• Should WebEA users be able to edit element notes?
	• Should WebEA users be able to edit test elements?
	• Should WebEA users be able to edit resource allocation elements?
	• What is the default list of watchlist items for the model?

2	Ensure a database manager is defined for each model decided on in step 1, in the Cloud Service's database manager list. The process to do this is as outlined for the core Sparx Systems Cloud Service, in *Add Database Manager*.
3	Enter your License key into the SSCloudServices.config file (by default found in the 'C:\Program Files (x86)\Sparx Systems\Cloud Services\Service\' folder). To edit this file you will require Administrator privileges. **Tip:** Add this line, replacing the zeroes with your license key. LICENSE={00000000-0000-0000-0000-000000000000}
4	For each Cloud Services ports that will be accessed by WebEA models, check that the option 'OSLC_SUPPORT' is set to '1' in the SSCloudServices.config file, and save any changes. **Tip:** Add/edit the OSLC_SUPPORT option to enable OSLC on the given port. OSLC_SUPPORT=1
5	Ensure that the Windows service 'Sparx Systems Professional Cloud' has been restarted since making changes to the configuration file (in the previous steps), so that new options are loaded.
6	For each of the models decided on in step 1, select the model on the 'Sparx Systems Cloud Services Configuration Client', click on the Configure button and select the 'Enable extended OSLC interface for WebEA' checkbox.
7	Download/install and configure your web server for PHP, according to your requirements. **Tip:** XAMPP is the most common Apache/PHP web server that includes all the required components; it is available for Windows/Linux and OS X at https://www.apachefriends.org/download.html. WebEA has been tested on all of the currently available versions of PHP, but if your environment has no special requirements then Sparx Systems would recommend v7.1.1 for Windows. All further Quick Start points will assume this configuration. Note: WebEA only requires the minimum XAMPP installation, Apache and PHP; it does not require MySQL, FileZilla Mercury, Tomcat or any of the other optional components, so if you have no need for these components Sparx Systems would recommend to not install them.
8	Define and configure a website for WebEA in your web server. As there are many different web servers and configurations, configuring the web site for WebEA, deciding what port the web server will listen to, and making sure the website is secure from external attacks, are all outside the scope of this documentation. **Tip:** Using the default XAMPP configuration, the port for the default website is 80 and the base folder is at 'C:\xampp\htdocs'. XAMPP will already have added files to this folder; WebEA does not require these files, so if you have no need for them, move them to a backup location.
9	Copy the contents of the entire WebEA installation folder ('C:\Program Files (x86)\Sparx Systems\Cloud Services\WebEA\') to the base folder of the website. **Tip:** Copy WebEA folders and files from 'C:\Program Files (x86)\Sparx Systems\Cloud Services\WebEA\' to 'C:\xampp\htdocs'.
10	Due to the sensitive nature of some of the information contained in the WebEA configuration file, we recommend that the '\includes\' folder within the WebEA folder is severely restricted so that only PHP can access the files within. For example, in Apache this is achieved with this entry in the Apache configure file: <Directory "{website_path}/includes"> AllowOverride none options none Require all denied

	</Directory> Once your web server is configured and the WebEA application files copied into the base folder, restart/start the web server. **Tip:** Using the XAMPP control panel, Stop/Start Apache.
11	Edit the '\includes\webea_config.ini' file and define each model decided on in step 1. See the *How to configure WebEA models* Help topic for more details on what each option means in WebEA's config file. **Tip:** Edit 'C:\xampp\htdocs\includes\webea_config.ini' file, and update all parameters with the prefix 'SSCS_' to that of a defined repository in your Cloud Service.
12	On a remote machine, open an internet browser and enter the URL of http://{yourwebserver}/login.php, where the value '{yourwebserver}' is either the name or IP number of the machine hosting the WebEA website. This should display the WebEA login screen that lists the default 5 models (Model One (no updates), Model One with discussions, and so on).
13	Other considerations: • If you require different lists of models (or different options) for different user bases, then it is possible to define multiple websites, each with their own dedicated port and base folder, using Apache's virtual hosts. This could mean user group A could access its set of models with the URL http://{yourwebserver}:180/ while user group B would use the URL http://{yourwebserver}:200/ • Firewall rules will have to be adjusted to allow traffic between the [Client] and [Web Server] • Firewall rules will have to be adjusted to allow traffic between the [Web Server] and [Pro Cloud Server]

Model Configuration

Action	Description
Configure Enterprise Architect model options	In all editions of Enterprise Architect prior to build 1350, diagrams and linked documents did not exist in a format that was viewable by browsers. In order for users of WebEA to view these components, Enterprise Architect must be configured to generate (and store) viewable versions of them whenever changes are made. The options to generate viewable diagrams and linked documents are defined at the model level. With the model open in Enterprise Architect, select the 'Configure > Model > Options' ribbon option. On the 'Manage Project Options' dialog, refer to the 'Data Cache' options at the foot of the 'General' page. Data Cache ☑ Auto create Diagram Image and Image Map (each time a Diagram is saved) ☑ Auto create HTML Page (each time a Linked Document is saved) Batch create Diagrams and/or Linked Documents cache: Create The first option (if enabled), in addition to saving a diagram normally will also generate and save a static image of the diagram. The second option (if enabled), in addition to saving a Linked Document normally

	will also generate and save the document in HTML format.
	If these options are not enabled, diagrams and linked documents will not be viewable from within WebEA.
	When initially configuring a model to auto create viewable diagrams and linked documents, it is advisable to click on the Create button to generate the initial set of viewable diagrams and linked documents. While this is not mandatory, viewable versions of all existing diagrams and linked documents will not exist until they are saved.
	Note: This approach is relevant when all Enterprise Architect users are using a build later than 1350. If, however, some of your user base is using older versions of Enterprise Architect then updates to diagrams and linked documents will not be reflected immediately when saved. See the *How to configure automatic viewable components* Help topic for the recommended approach to overcome this situation.
Configure model users	It is possible for WebEA users to connect to user security enabled models in one of two ways:
	• Using their model User Login and Password; that is, the same credentials they use when connecting to the model with Enterprise Architect, or
	• If the model has been configured in WebEA to not require login credentials then all users will be logged in under the one user account

Acknowledgements

WebEA incorporates certain components developed by third parties. Sparx Systems acknowledges the contribution made to the WebEA product by these components.

• jQuery v1.12.3 - jQuery Foundation

• NicEdit ©2007-2008, Brian Kirchoff, under license from MIT

• Date Picker for jQuery v5.1.1, Keith Wood, under license from MIT

How to configure automatic viewable components

While Sparx Systems recommend that users should always use the latest release of Enterprise Architect, we do acknowledge that in some situations this is hard to achieve. Therefore, in order for WebEA (and the Pro Cloud Server) to function correctly when users are using builds of Enterprise Architect earlier than 1350, the Pro Cloud Server includes a custom version of Enterprise Architect (referred to as the EA Worker) that will automate the generation of WebEA viewable diagrams and linked documents.

Each instance of an EA Worker is capable of connecting to a single model, and typically runs on the same server as the Pro Cloud Service.

How to run EA Worker

The 'EA Worker' application does not have a GUI of any kind, therefore it is started with two command line options:

1) "cloudupdate:" - the frequency (measured in seconds) of how often the application should check for new or modified diagrams and linked documents.

2) A connection string to the model, defined as a .EAP file.

An example command line is:

    C:\> EA.exe /cloudupdate:30 "C:\EA\shortcuts\model1.eap"

Typically the easiest and most efficient method to run the EA Worker is with a Windows batch/command file.

Model shortcuts for EA Worker

Within Enterprise Architect it is possible to save model connections (no matter what type of DBMS is in use) as a .EAP file. These model shortcuts are not .EAP file databases, but plain text files that contain a connection string to a model.

An example .EAP shortcut to a model without security:

    EAConnectString:model1 --- ;Connect=Cloud=protocol:http,address:localhost,port:804;Data Source=model1;

However, when the model has security enabled the connection string requires a USER and PASSWORD parameters:

    EAConnectString:model1 --- ;Connect=Cloud=protocol:http,address:localhost,port:804,user:fred,pwd:pwdxxx;Data Source=model1;

EA Worker Configuration Steps

Step	Description
1	Create a .EAP shortcut to the model.
	The creation of a .EAP shortcut file can be achieved by either:
	1) Manually creating a plain text file, renaming its extension to .EAP and entering the complete connection string as shown (making sure to replace all values enclosed in {}):
	EAConnectString:{modelname} --- ;Connect=Cloud=protocol:http,address:{servername},port:{port},user:{username},pwd:{password};Data Source={modelname};
	or
	2) In Enterprise Architect, open a Cloud connection to the model in question and then choose 'Save As

		Shortcut...' from the drop-down menu.
2		Create a batch/command file.
		Create a plain text file, give it a meaningful name, and change its extension to .CMD. Then edit the file and enter this text, making sure to replace {seconds} with the number of seconds and {filepath} with the location of the fully qualified .EAP created in the previous step:
		"C:\Program Files (x86)\Sparx Systems\Cloud Services\Worker\EA.exe" /cloudupdate:{seconds} "{filepath}"
3		Run the batch/command file.
		Double click on the file created in the previous step.
		Sparx System recommends that this batch/command file is added to Startup so that it is automatically run when the server starts.

Lifecycle of EA Worker

The EA Worker application is designed to connect to the specified model automatically when it starts; if there is a problem with the connection the application ends. If the connection string is valid and the EA Worker connects to the model it will run continually until the server is restarted or until the process is manually terminated.

How to configure WebEA models

All WebEA configuration is controlled by a single file, which must reside in the '\includes\' folder and which must have the name 'webea_config.ini'.

The configuration file is a plain text file. Options are grouped under section headers, which are enclosed by square brackets. Each option must be defined on a separate line and follow the syntax:

   {option name} = "{option value}"

Note that:

- Option values are enclosed by double quotes

- The configuration file supports comments; the semi-colon character starts a comment

- The default 'webea_config.ini' file included in the installation package contains samples of how models can be configured

Why are there so many options?

In order to offer the greatest level of flexibility in how WebEA models can be accessed and used, Sparx Systems provides a large number of configuration options, which can be daunting when the complete list is viewed for the first time. However, to minimize your effort in understanding and managing the options, most are also grouped by prefix (sscs, login, add, edit and wl), which helps you to quickly understand what a particular option is used for.

The Model List

The first section in the configuration file is '[model_list]'; this section defines the model names that will be displayed to WebEA users on the 'Login' page. There is no limit to the number of models that can be defined; however, when there are more than ten models the login screen will automatically show a vertical scroll bar.

The model names have no direct link to the physical server or database name of the models they are describing, so you are free to use any meaningful name.

For example:

   [model_list]

   model1 = "Model One (no updates)"

   model2 = "Model One with discussions"

   model3 = "Model Two with user login"

   model4 = "Model Three with all options"

   model5 = "Sparx Systems Sample Model"

Note: it is important that the 'model*x*' definitions are consecutive; any out-of-sequence model numbers will be ignored and no further models will be added to the WebEA Login page.

Model options

For each numbered model found in the '[model_list]' section a matching '[modelx : settings]' section is expected, further down the configuration file. All options within this section relate to that model only; that is, options can not be shared between models.

For example:

[model1 : settings]

sscs_protocol = "http"

sscs_server = "localhost"

sscs_port = "80"

sscs_db_alias = "model1"

auth_code = ""

login_prompt = "false"

default_diagram = ""

show_discuss = "false"

Model connection settings

Probably the most important settings for any model are the connection-related ones, prefixed with sscs_ (which stands for 'Sparx Systems Cloud Service'). This group of settings defines how WebEA will connect and communicate with the Cloud Service for data retrieval.

All communication between WebEA and the Cloud Service is performed via the PHP cURL library, therefore some of the connection settings relate directly to the cURL options.

Option	Description
sscs_protocol	Defines the protocol to be used to connect to the Sparx Systems Cloud Service. **Options:** http or https **Default:** http **Example:** sscs_protocol = "http"
sscs_port	Defines the TCP port used to connect to the Sparx Systems Cloud Service. This port number must match one of the ports the Sparx Systems Cloud Service is listening to or, in the case of an IIS setup, the IIS port servicing this repository. **Default:** 80 **Example:** sscs_port = "804"
sscs_server	Defines the fully qualified machine name or IP address of the server hosting the Cloud Service. **Default:** localhost **Example:** sscs_server = "sparxpublic.com" or sscs_server = "197.168.0.10"
sscs_use_ssl	Defines if all communications between WebEA and Cloud Services should use SSL (secure socket layer). When this value is true, the protocol should be set to "https". **Options:** true or false **Default:** false **Example:** sscs_user_ssl = "false"
sscs_db_alias	Defines the database alias name of the model repository hosted on the Cloud Service. This corresponds to the 'Database Alias' identified in the Cloud Services Configuration Client or the 'Model Name' on the Enterprise Architect's Cloud Connection screen. The value is case sensitive. **Default:** ea **Example:** sscs_db_alias = "RepositoryName"

sscs_enforce_certs	When SSL is set to true this option defines whether or not cURL will enforce that the certificate installed in Cloud Services is correctly signed by a Certificate Authority. Note: If you are using a self-signed certificate for Cloud Services, this option must be set to false. **Options:** true or false **Default:** true **Example:** sscs_enforce_certs = "false"
sscs_model_user	Defines the model user login (as defined in Enterprise Architect User Security) that will be used in combination with the sscs_model_pwd option to validate if the user/password combination can access the model on the Cloud Server. **Default:** **Example:** sscs_model_user = "webuser"
sscs_model_pwd	Defines the password (as defined in Enterprise Architect User Security) that will be used in combination with the sscs_model_user option to validate if the user/password combination can access the model on the Cloud Server. **Default:** **Example:** sscs_model_pwd = "pwdxxx"

Model general settings

These options define how WebEA users can access and display objects within the model.

Option	Description
auth_code	Defines the authorization code all WebEA users must enter before they can access the model. If this value is blank, WebEA users are not prompted to enter a code. This option behavior is not dependant on the 'login_prompt' setting, the two settings can be used in combination with each other or independently. The most secure method would be to define an authorization code and set 'login_prompt' to true.
login_prompt	Defines whether or not all WebEA users must enter a model user/password before they can access the model. **Options:** true or false **Default:** false **Example:** login_prompt = "false"
login_allow_blank_pwd	When the 'login_prompt' option is set to true, this option defines whether or not WebEA will attempt to send blank passwords to the Cloud Services for validation. **Options:** true or false **Default:** false **Example:** login_allow_blank_pwd = "false"
participate_in_reviews	When set to true, 'participate_in_reviews' allows users of WebEA to Join reviews that will automatically assign all discussions to the joined review. In order to make full use of this option SHOW_DISCUSS and ADD_DISCUSS would also have to

	be set to true. **Options:** true or false **Default:** false **Example:** participate_in_reviews = "false"
recent_search_days	Defines the number of days the predefined 'Recent ...' searches should consider. For example if this value was set to '7' then whenever a user invokes the search for 'Recent elements', all elements modified in the previous 7 days will be returned. **Options:** any positive integer value **Default:** 3 **Example:** recent_search_days = "10"
use_avatars	When the 'use_avatars' option is set to true, all discussion messages are displayed with the creating user's avatar. When set to false a set of generic images representing discussions and replies is used instead. **Options:** true or false **Default:** true **Example:** use_avatars = "false"
prop_sec_<section>_visibl e	Defines whether or not a property section is visible. The property view includes numerous sections that can be optionally hidden, which is particular useful if a model is being configured for a group of users that are not interested in the finer details of elements or Packages. <section> can be one of: • location • relationships • taggedvalues • testing • resourcealloc • attributes • operations • runstates • changes • defects • issues • tasks • events • decisions • efforts • risks • metrics **Options:** true or false **Default:** true **Example:** prop_sec_testing_visible = "true"
prop_sec_<section>_expan	Defines whether or not a property section is expanded by default when the model is

ded	opened.
	The property view includes numerous sections that can be optionally expanded or collapsed; these options set the default values for the model.
	<section> can be one of:
	• info
	• location
	• relationships
	• taggedvalues
	• testing
	• resourcealloc
	• attributes
	• operations
	• runstates
	• changes
	• defects
	• issues
	• tasks
	• events
	• decisions
	• efforts
	• risks
	• metrics
	Options: true or false
	Default: false for all sections except 'info' and 'location', which are true
	Example: prop_sec_testing_expanded = "true"
default_diagram	Defines the GUID of the diagram that should be automatically loaded when users log onto the model. If this option is not defined or an empty value is specified, WebEA assumes that no diagram should be loaded and displays the model root instead. The GUID of any diagram can be obtained from WebEA by loading the diagram properties.
	Note: The entire GUID (including {}) must be used.
	Example: default_diagram = "{C89E174A-19B0-47c7-B4A8-8A98071CD838}"
default_main_layout	Defines the default layout of container items for the current model. WebEA can display child items of a container in one of two layouts. Either Icon view (the default) or List view.
	Options: icon or list
	Default: icon
	Example: default_main_layout = "list"
show_discuss	Defines whether or not discussions on Packages and elements should be displayed within the WebEA interface.
	Options: true or false
	Default: false
	Example: show_discuss = "true"

Model update settings

This group of settings determine what insert/edit options are available to WebEA users.

Note: Within the Cloud Service it is possible to configure a model as 'Read-only'; this setting in the Cloud Service will override any WebEA update setting, forcing them to false.

Option	Description
add_discuss	Defines whether or not new discussions on Packages and elements can be added to the model. Note: This option is forced to false if 'show_discuss' has been set to false. **Options:** true or false **Default:** false **Example:** add_discuss = "true"
add_objects	Defines whether or not new objects can be added to the model. **Options:** true or false **Default:** false **Example:** add_objects = "true"
add_objecttype_<type>	Defines whether or not elements of the given object type can be added to the model. These options are forced to false if 'add_objects' has been set to false. <type> can be one of: • package • review • actor • change • component • feature • issue • node • requirement • task • usecase **Options:** true or false **Default:** false **Example:** add_objecttype_actor = "true"
add_object_features	Defines whether or not element features can be added to the model. **Options:** true or false **Default:** false **Example:** add_object_features = "true"
add_objectfeature_<type>	Defines whether or not element features of the given type can be added to the model.

	These options are forced to false if 'add_object_features' has been set to false.
	<type> can be one of:
	• tests
	• resources
	• changes
	• defects
	• issues
	• tasks
	• risks
	Options: true or false
	Default: false
	Examples: add_objectfeature_changes = "true"
edit_objectfeature_resources	Defines whether or not the details of element resource allocations can be edited in the model.
	Options: true or false
	Default: false
	Example: edit_objectfeature_resources = "true"
edit_objectfeature_tests	Defines whether or not the details of element tests can be edited in the model.
	Options: true or false
	Default: false
	Example: edit_objectfeature_tests = "true"

Model watchlist settings

The watchlist functionality allows users to quickly see a wide variety of objects that have changed recently in the model. WebEA users can configure their personal preferences to control what sort of objects they are interested in 'watching'.

All watchlist settings are prefixed by 'wl_' and represent the default value for the model. However, once a user saves their own watchlist settings the model defaults will no longer be used.

In order to persist the watchlist preferences across sessions, WebEA saves them in a browser cookie. If the user's browser does not allow the saving of cookies, personal watchlist settings can not be persisted across sessions.

Option	Description
cookie_retention	Defines the period over which a WebEA user's set of Watch list options are stored. The value is the number of days that the cookies will be retained for.
	Options: any positive integer value
	Default: 365
	Example: cookie_retention = "180"
wl_period	Defines the default activity period of the model watchlist. The value is a number of days; for example, look for activity in the last 10 days.
	Options: any positive integer value
	Default: 0
	Example : wl_period = "10"

wl_recent_discuss	Define whether or not the watch list should display a count of elements that have had discussions during the activity period. **Options:** true or false **Default:** false
wl_recent_reviews	Define whether or not the watch list should display a count of review elements that have been added or modified during the activity period. **Options:** true or false **Default:** false
wl_recent_diagram	Define whether or not the watch list should display a count of added/modified diagrams during the activity period. **Options:** true or false **Default:** false
wl_recent_element	Define whether or not the watch list should display a count of added/modified elements during the activity period. **Options:** true or false **Default:** false
wl_resalloc_active	Define whether or not the watch list should display a count of elements with active resource allocations during the activity period. **Options:** true or false **Default:** false
wl_resalloc_today	Define whether or not the watch list should display a count of elements with resource allocations ending during the activity period. **Options:** true or false **Default:** false
wl_resalloc_overdue	Define whether or not the watch list should display a count of elements with resource allocations that fall overdue during the activity period. **Options:** true or false **Default:** false
wl_test_recentpass	Define whether or not the watch list should display a count of elements with tests that passed during the activity period. **Options:** true or false **Default:** false
wl_test_recentfail	Define whether or not the watch list should display a count of elements with tests that failed during the activity period. **Options:** true or false **Default:** false
wl_test_recentdefer	Define whether or not the watch list should display a count of elements with tests that were deferred during the activity period. **Options:** true or false

	Default: false
wl_test_recentnotchk	Define whether or not the watch list should display a count of elements with tests that were not checked during the activity period. **Options:** true or false **Default:** false
wl_test_notrun	Define whether or not the watch list should display a count of elements with tests that were not run during the activity period. **Options:** true or false **Default:** false
wl_change_verified	Define whether or not the watch list should display a count of elements with verified changes during the activity period. **Options:** true or false **Default:** false
wl_change_requested	Define whether or not the watch list should display a count of elements with requested changes during the activity period. **Options:** true or false **Default:** false
wl_change_completed	Define whether or not the watch list should display a count of elements with changes completed during the activity period. **Options:** true or false **Default:** false
wl_change_new	Define whether or not the watch list should display a count of elements with new changes added during the activity period. **Options:** true or false **Default:** false
wl_change_incomplete	Define whether or not the watch list should display a count of elements with incomplete changes during the activity period. **Options:** true or false **Default:** false
wl_defect_verified	Define whether or not the watch list should display a count of elements with verified defects during the activity period. **Options:** true or false **Default:** false
wl_defect_requested	Define whether or not the watch list should display a count of elements with defects reported or 'requested' during the activity period. **Options:** true or false **Default:** false
wl_defect_completed	Define whether or not the watch list should display a count of elements with defects completed during the activity period.

	Options: true or false **Default:** false
wl_defect_new	Define whether or not the watch list should display a count of elements with defects added during the activity period. **Options:** true or false **Default:** false
wl_defect_incomplete	Define whether or not the watch list should display a count of elements with incomplete defects during the activity period. **Options:** true or false **Default:** false
wl_issue_verified	Define whether or not the watch list should display a count of elements with verified issues during the activity period. **Options:** true or false **Default:** false
wl_issue_requested	Define whether or not the watch list should display a count of elements with issues reported or 'requested' during the activity period. **Options:** true or false **Default:** false
wl_issue_completed	Define whether or not the watch list should display a count of elements with issues completed during the activity period. **Options:** true or false **Default:** false
wl_issue_new	Define whether or not the watch list should display a count of elements with issues added during the activity period. **Options:** true or false **Default:** false
wl_issue_incomplete	Define whether or not the watch list should display a count of elements with incomplete issues during the activity period. **Options:** true or false **Default:** false
wl_task_verified	Define whether or not the watch list should display a count of elements with verified tasks during the activity period. **Options:** true or false **Default:** false
wl_task_requested	Define whether or not the watch list should display a count of elements with requested tasks during the activity period. **Options:** true or false **Default:** false
wl_task_completed	Define whether or not the watch list should display a count of elements with tasks

	completed during the activity period. **Options:** true or false **Default:** false
wl_task_new	Define whether or not the watch list should display a count of elements with tasks added during the activity period. **Options:** true or false **Default:** false
wl_task_incomplete	Define whether or not the watch list should display a count of elements with incomplete tasks during the activity period. **Options:** true or false **Default:** false
wl_event_requested	Define whether or not the watch list should display a count of elements with requested events during the activity period. **Options:** true or false **Default:** false
wl_event_completed	Define whether or not the watch list should display a count of elements with events completed during the activity period. **Options:** true or false **Default:** false
wl_event_high	Define whether or not the watch list should display a count of elements with events set to high during the activity period. **Options:** true or false **Default:** false
wl_decision_verified	Define whether or not the watch list should display a count of elements with verified decisions during the activity period. **Options:** true or false **Default:** false
wl_decision_requested	Define whether or not the watch list should display a count of elements with requested decisions during the activity period. **Options:** true or false **Default:** false
wl_decison_completed	Define whether or not the watch list should display a count of elements with decisions completed during the activity period. **Options:** true or false **Default:** false
wl_decision_new	Define whether or not the watch list should display a count of elements with decisions added during the activity period. **Options:** true or false **Default:** false

wl_decision_incomplete	Define whether or not the watch list should display a count of elements with incomplete decisions during the activity period.
	Options: true or false
	Default: false

How to configure WebEA Localization

The physical text values of all labels and messages within WebEA are defined in a single file that is read when the web pages are being built. This allows WebEA customers to tailor the content of web pages displayed to their users.

WebEA's default language is English. As pages are built on the server, the WebEA application will check whether there is a defined localization for a string and use that. If a localized version of the string is not defined then WebEA will use the default English value.

The WebEA localization definitions are saved in the file 'webea_strings.ini' in the '\includes\' folder. The localization file is saved in plain text format with each localized string being defined on a separate line with the syntax:

   {default_english_string} = "{localized_string}"

Note that:

- Localized string values are enclosed by double quotes

- The configuration file supports comments; the semi-colon character starts a comment

- The supplied 'webea_strings.ini' file includes a definition of all strings used within the WebEA application, therefore there is no need to add any new lines to the file

For example, on the login page the first label is "Select Model" in the English version; however, a French user would want to change this text to:

   Select Model = "Sélectionnez le modèle"

Replacements

The wording of some labels/messages within WebEA requires the names of items to be added to the string displayed. Such replacements are denoted by the %s placeholder. The localized versions of these strings should also include the %s placeholder in the appropriate position.

The other standard replacement that WebEA supports is the replacement of %NN% with a number. For example, in the Watchlist when WebEA detects that elements have been modified recently, this message is used:

   %NN% recently modified element%S% = "%NN% recently modified element%S%"

%NN% will be replaced with a count of recently modified elements.

Plurals

Given the nature of the English language, some of the localized strings support the ability to display words in their plural form. WebEA supports the special substitution of %S% which will replace, %S% with 's' when the supplied number is greater than 1. Using the same example as above, when a single element has been modified recently the message displayed on the watchlist page is:

   1 recently modified element

However, when there are 5 recently modified elements, WebEA would generate the message:

   5 recently modified elements

Troubleshooting WebEA

If WebEA users experience a problem, an error message will be displayed on the screen. This topic provides a list of common WebEA error messages that could occur. Each entry describes the situation that has occurred and provides guidance on how to correct the error.

WebEA is not accessible

On initial set up, check that the web server can communicate with the Pro Cloud Server on the specified IP address and port.

See *Browser Test* on the *Cloud Server Troubleshooting* Help page.

Check that the IP address and port are correctly set in the webea_config.ini file.

In routine operation, also confirm that the server is running.

WebEA Error Messages

Issue	Comments
Server could not be found	*Server could not be found on the network!* This error usually occurs when WebEA does not receive any kind of response from the specified server name (or IP number). Possible reasons: • The server is turned off • The server is behind a firewall and WebEA is unable to make a connection to it
Connection refused	*There was no response from the server, check that Pro Cloud Service is running!* Possible reasons: • Pro Cloud Service is not installed • Pro Cloud Service is not running • Pro Cloud Service is not listening on the expected port
OSLC not supported	*Your server is not configured to support OSLC.* Possible reasons: • WebEA communicates with Sparx Systems Cloud Service via OSLC, but the defined Sparx Systems Cloud Service has been configured to not allow OSLC requests
Unknown Protocol	*Unknown protocol! Get system administrator to confirm that the correct protocol and port have been configured.* Possible reasons: • There is a mismatch between the connection settings; for example, 'sscs_use_ssl' has been set to true but a protocol of HTTP was specified
Incorrect login details	*The specified login details were not valid for the current model!* Possible reasons: • The supplied user ID and password combination did not match any user

	credentials defined in the model
Connection timed out	*Request Error: Connection timed out after 5001 milliseconds.* Possible reasons: • Invalid Cloud Server settings (sscs_*), WebEA can't connect to the defined server/port • Apache/PHP web server requires a proxy server to reach the Cloud Service machine and the proxy settings defined in the .config file are incorrect/invalid
Failed to Connect	*Request Error: Failed to connect to xxx.xxx.xxx.xxx port yy: Connection refused* Possible reasons: • The 'sscs_port' option in the webea_config.ini file does not match the port setting in your Pro Cloud Service configuration file (SSCloudServices.config); make them the same Note that, if the Cloud Service is being re-directed via IIS, you must ensure this matches the IIS port used for the Pro Cloud Service.
The configured database is not defined	*The configured database is not defined in the Cloud Service.* Possible reasons: • The 'sscs_db_alias' option in the webea_config.ini file does not correspond to the 'Database Alias' defined in *Add Database Manager;* correct the values so that they do correspond
Problem reading the model root	*Problem reading the model root.* Possible reasons: • The configured port for the current model is not a port the Pro Cloud Server is listening to • The WebEA options for the current model do not match a Pro Cloud Server configuration
No response when selecting an Object	After WebEA loads the initial page (the model root or default diagram), there is no response to selecting objects**.** Possible reasons: • The user's browser does not have javascript enabled; re-enable it • The cURL PHP module/extension is not installed and enabled on the server; install/enable cURL on your web server • The PHP version is wrong; it must be greater than PHP 5.0 with all extensions supported
An unexpected response was received	*An unexpected response was received, check the cloud service configuration!* This error normally occurs while attempting to login or connect to a model from the initial page. The error message will display whenever the response from the configured Cloud Service does not return the expected XML message. Check that the defined Sparx Systems Cloud Service server name and port are the details of an actual Cloud Service and not those of a web server.
The selected database does not have the extended OSLC	*The selected database does not have the extended OSLC interface enabled.* This error normally occurs if the selected model is not enabled for the Extended OSLC interface, which must be enabled using the configuration client. Possible reasons:

	• The database has not been enabled for the extended OSLC interface
	• The Pro Cloud Server's license has expired and the server is now operating as the free, core edition of the Cloud Service
	• The License key entered is invalid and the Pro Cloud Server is operating as the core edition

Problems loading pages

Issue	Comments
Pages don't load correctly	After the server has been updated to a new version of WebEA, clients experience problems with pages loading incorrectly. Possible reasons: • The user's browser is using a cached version of WebEA's client side files which is causing unpredictable results The solution is to clear the browser cache, to force the browser to download the latest version of the application files from the server; the process for clearing the cache is different for each browser/device - consult the user manual for your browser
Login page in Internet Explorer	The 'Login' page is not drawn correctly and appears different to other browsers; that is, the selection boxes next to each model name are not square. All login attempts to models with authentication fail saying 'No model selected!' Possible reasons: • IE is running in 'Compatibility View' mode, which does not support HTML5 and CSS3; turn off 'Compatibility View' at least for the server hosting WebEA

WebEA User Guide

Sparx System's WebEA is an external interface to Enterprise Architect models through which users can review and comment on a model, in real time via the web, on a range of mobile devices or a remote work station, without needing to install Enterprise Architect. The reviewer can be anywhere in the world, and does not need to be in an office, just somewhere where internet access is available. It is ideal for stakeholders and clients to see the work in which they are invested, and to comment on and discuss aspects of the model.

WebEA models can be configured to allow users to update element notes, add element features (such as tests and resource allocations) and even create Project Management elements (to record issues, changes and similar), or use cases, requirements and system components.

WebEA is the interface to the model, so although you are using your web browser, the data shown is the same as that shown when you use Enterprise Architect. This means that all model updates occur in real time, so WebEA users can be confident that they are seeing the latest version of the model structure.

Access

Open the web browser on your device and, in the address bar, type in the complete WebEA URL. Press the Enter key or tap the Go key.

These are some examples of the complete URL that might be needed to open WebEA.

   http://demo.sparxpublic.com

In the case where the web server has the IP of 11.22.33.44 and has been configured to use the non-standard port of 800, the complete URL would be:

   http://11.22.33.44:800

If the WebEA application files were placed in a sub-folder called 'WebEA' under the web server's root folder, then the complete URL would be:

   http://11.22.33.44:800/WebEA/

If the entered URL is a valid, the WebEA Login page is shown, which will show all the models that are available.
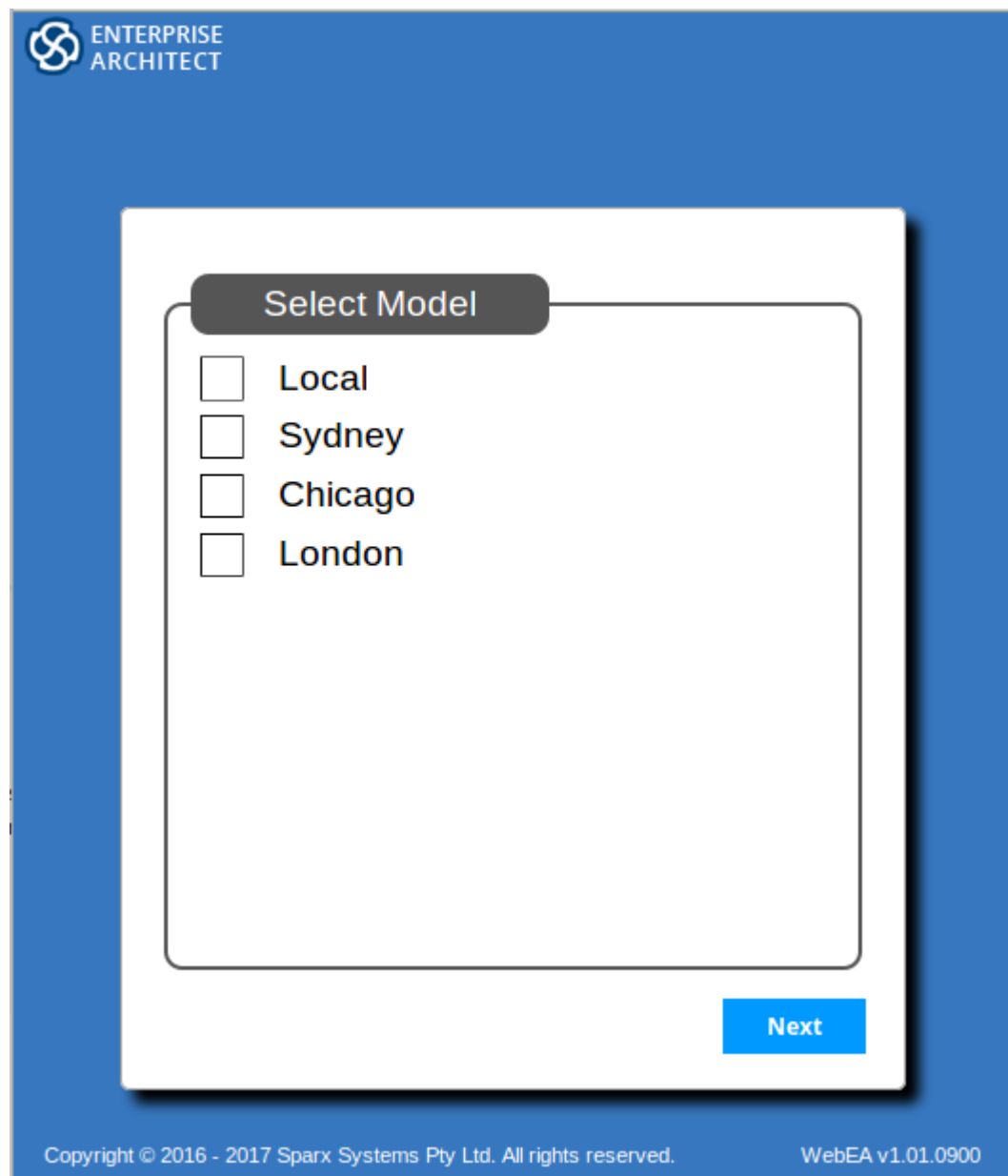
Acknowledgements

WebEA incorporates certain components developed by third parties. Sparx Systems acknowledges the contribution made to the WebEA product by these components.

- jQuery v1.12.3 - jQuery Foundation
- NicEdit ©2007-2008, Brian Kirchoff, under license from MIT
- Date Picker for jQuery v5.1.1, Keith Wood, under license from MIT

Login

WebEA websites are configured to allow access to one or more Enterprise Architect models, each of which can be configured independently to only grant access to users with valid logon credentials. WebEA user validation supports two forms of security, 1) an access code as well as 2) standard Enterprise Architect model security, these two validation methods can be used independently or in combination on a per model basis.

Prerequisites
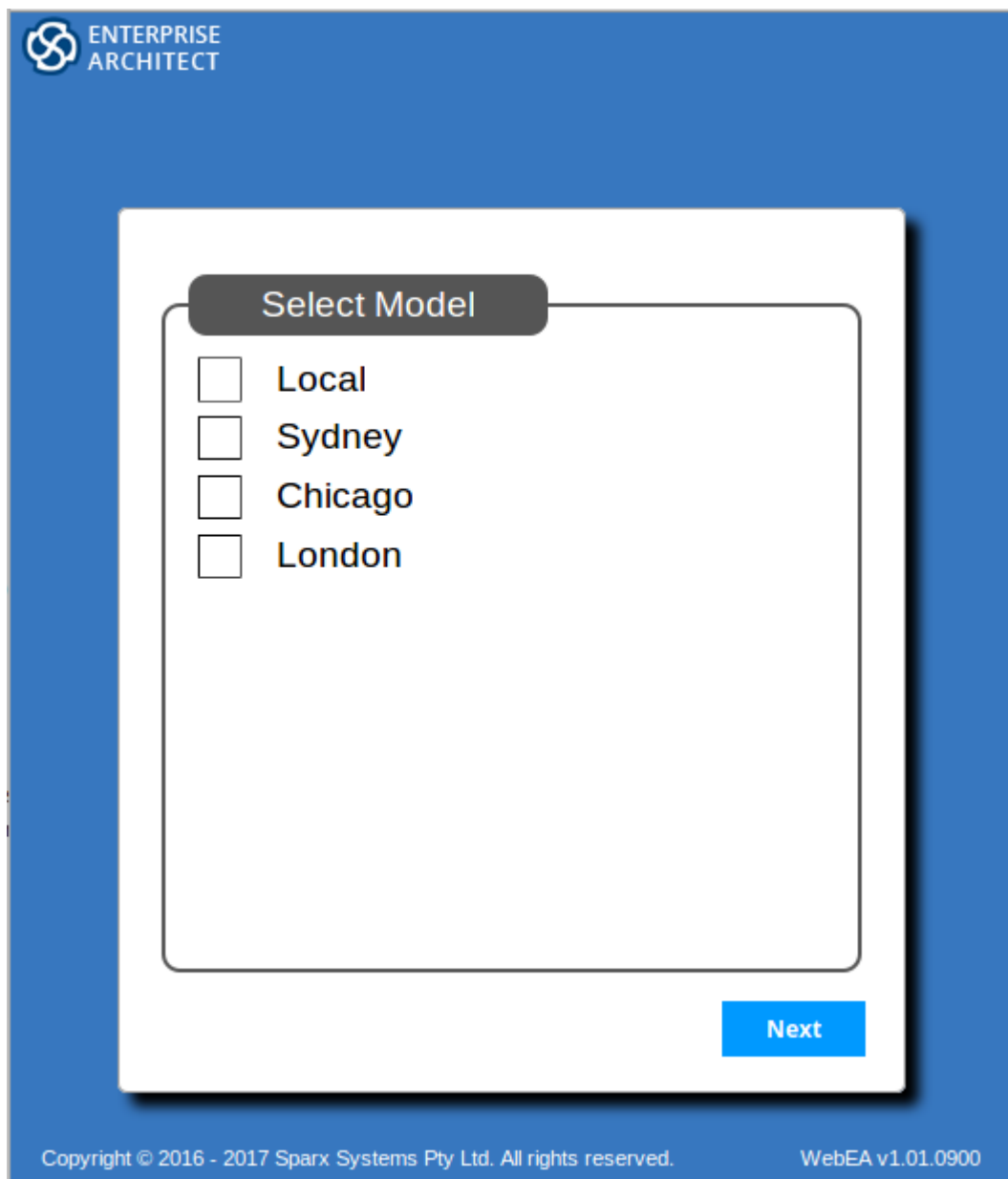
Users of WebEA will need to be advised of their login credentials for each model which will include:

- An access code, or
- A user ID and password, or
- Possibly all three

This data will be provided to you by your Web Administrator or by the Project Manager responsible for the model.

Logging In

The first page that is displayed in WebEA is the 'Login' page, which lists the models available to you.
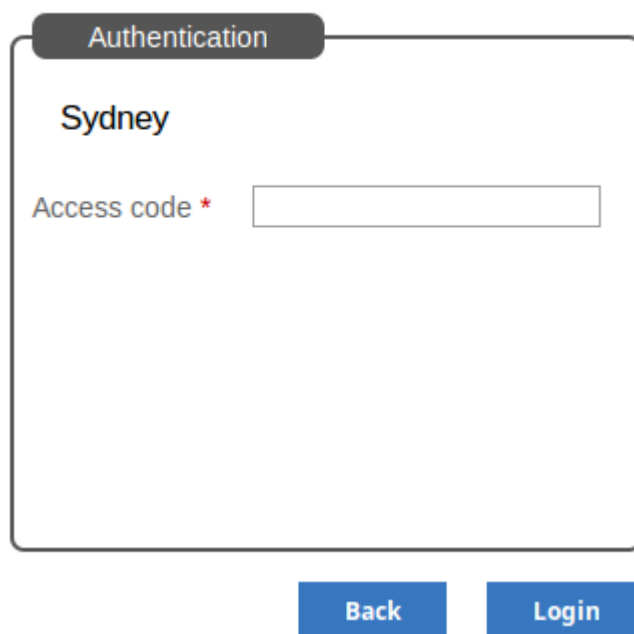
Select the model you want to open, and click or tap on the Next button.

If access to your model requires only an access code, then the 'Authentication' page prompts you for this.
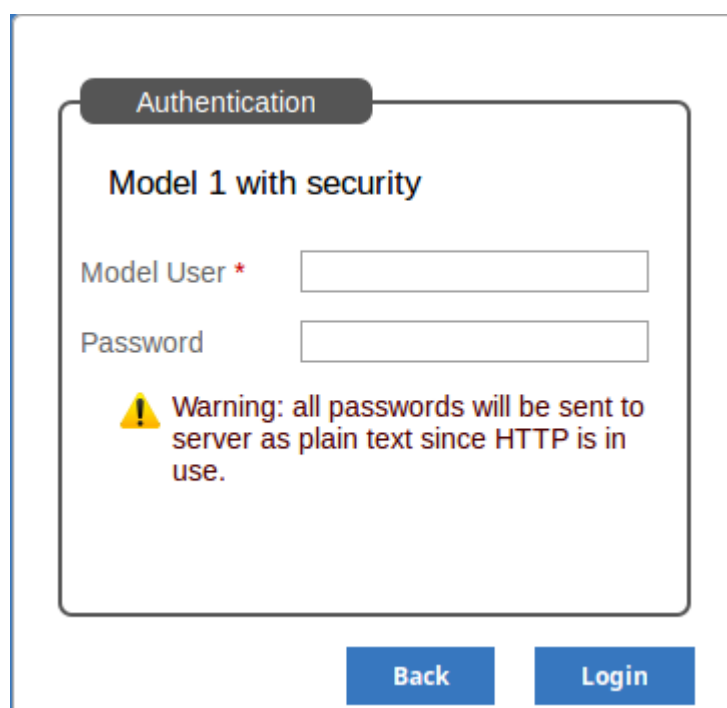
Note: The red asterix (*) denotes a mandatory field.

If User Security has been applied to the model, the 'Authentication' page prompts you for your user ID and password.

If the model requires both an access code and a user id/password, the 'Authentication' page prompts for both together:

If you do not have the authorization data, click or tap on the Back button to go back and select another model.

Otherwise, type in your access code and/or user ID and password, and click on the Login button.

WebEA's layout

Having logged on to a model in WebEA, you will be initially shown either the model root icon, a list of the model root nodes, or a default diagram (if one is defined for the model), depending on how the model has been configured to display on opening. From this point you can start navigating the model by selecting objects or links, review the organization and contents of the model elements, and read and add to discussions or reviews of particular structures.

Note: WebEA only supports single click when using a PC, or single touch when using a touch-enabled device.

This example shows the WebEA screen when the root node icon of the example model is displayed, and the equivalent list view.
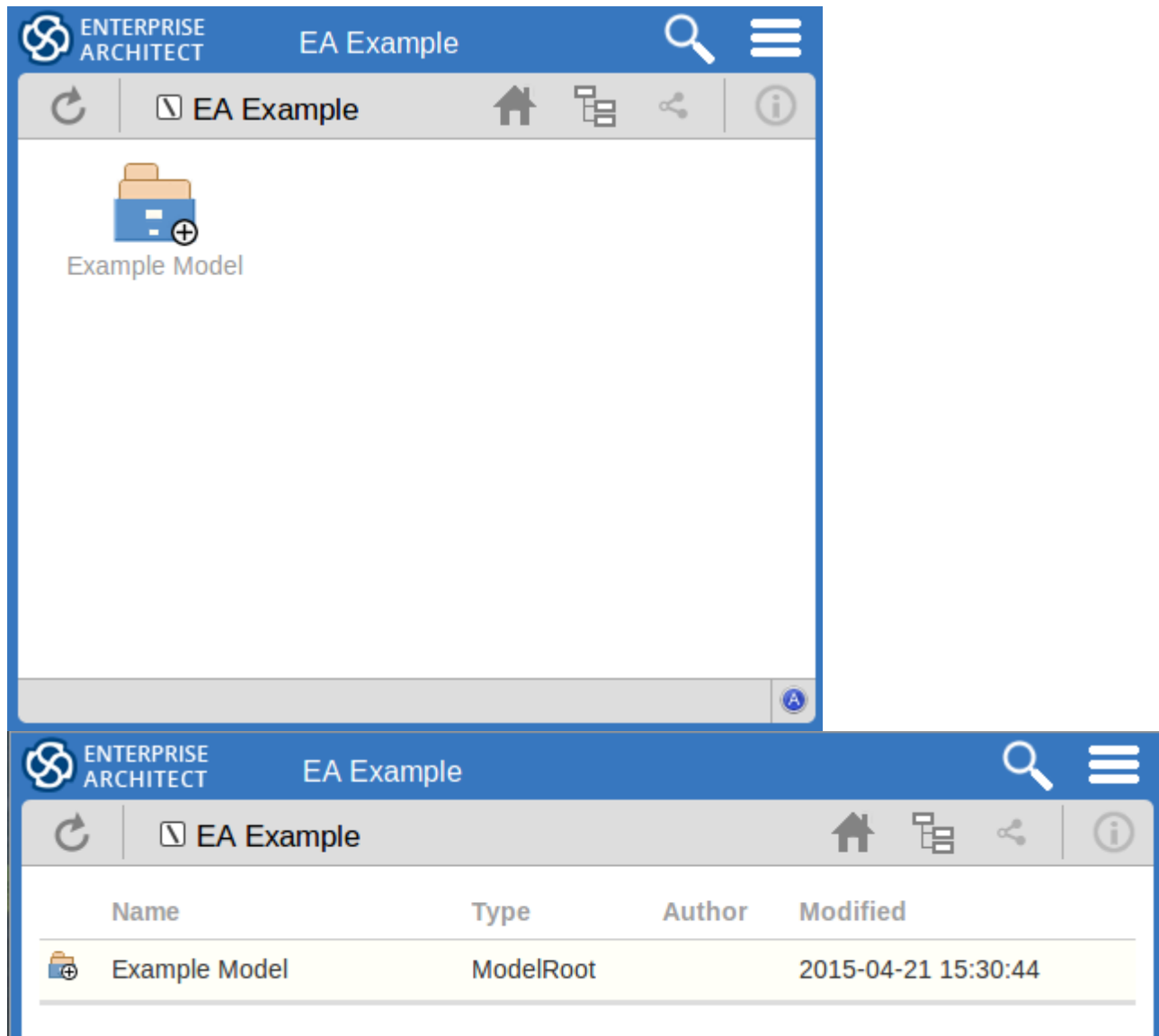
The WebEA view can be broken into four sections: heading, navigation bar, content area and status bar.

Note: In order to support small screen devices, some layout sections move or merge with others to minimize the amount of screen real estate needed. For example, the heading section merges with the Navigation bar when the screen height goes below 400 pixels.

Heading

WebEA's heading area displays the application logo in the top right corner, the name of the currently opened model, a search icon and a 'hamburger' icon. The search and 'hamburger' icons contain further options that are accessible when selected.

Search

The search icon provides access to the search functionality offered by WebEA. The functionality includes a number of standard searches, the Watchlist and the custom search.

Menu	Description
Goto item	Select to display the 'Goto WebEA item' dialog, into which you paste (Ctrl+V) a WebEA item's unique ID that you have copied earlier, using the 'Direct Link' icon in the page Navigation Bar (see the *Navigation Bar* table in this topic). Click on the Go button to load the details of the specified object.
Reviews	Select to display a list of all recently modified Review elements. Note: The number of days that are considered 'recent' is defined as a configuration option for the model.
Discussions	Select to display a list of all elements to which a Discussion item has been recently added.
Diagrams	Select to display a list of all recently modified diagrams.
Elements	Select to display a list of all recently modified elements (new Discussion items are not regarded as modifications to the element).
Watchlist	Select to display the 'Watchlist summary'. The Watch List facility in WebEA helps you to quickly identify recent activity in the model.
Custom search	Select to display the 'Custom Search' dialog. WebEA provides a facility that enables you to search through the model for a specific item or group of items that have certain characteristics.

Hamburger

The hamburger icon provides access to these options:

Menu	Description
Wide view	This is the default viewing option for devices with small screens; that is, phones and tablets when rotated to their narrow dimension. For further details, see the *Content Area - Properties View* section of this topic.
	This is the default viewing option for devices with screens wider than 800 pixels

Split view	(approximately), such as desktops.
	For further details, see the *Content Area - Properties View* section of this topic.
Icon view	Choose this option if the display of the model content is in list format, and you prefer it to be in icon format (as illustrated at the start of this topic).
List view	Choose this option if the display of the model content is in icon format, and you prefer it to be in list format (as illustrated at the start of this topic).
Logout	Choose this option if you want to disconnect from the current model. After the logout process has run, the 'Login' dialog automatically displays.

Navigation Bar
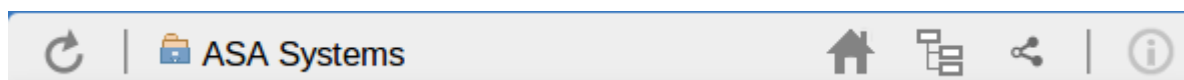
The Navigation bar provides information and buttons that assist in navigating an Enterprise Architect model. The various buttons available in the Navigation bar are explained in this table.

Icon	Description
Refresh	Click this button to refresh the current view. For example, if you are currently viewing a diagram and a user in Enterprise Architect has made changes to it since you opened it, simply click the Refresh button; WebEA will display the newly updated diagram.
Current object	This is a read-only text field that displays the name and object type (displayed as an image) of the current object.
Home	Click this button to reload the 'home' object for the current model. The home object is either the model's default diagram (if defined in the configuration file) or the model root Package.
Object Path	Click this button to display the complete path of parent nodes, Views, Packages and elements for the current object, all the way to the model's root node.
	You can click on any item in the list (except for the last one, which is the current object) and navigate directly to that object.
	Note: When viewing the root node there will be only one entry, which is not selectable.
Direct Link	Click this button to display the 'Link to WebEA item' dialog, which shows the current item's unique ID and full URL. You can highlight the ID and press Ctrl+C to copy it, then at a later point use it in the 'Goto WebEA item' dialog to return directly to that page, as described in the *Search* table earlier.
	Alternatively, you can copy the full URL and send it to another user if you want them look at a particular item.
Properties	Click this button to display the properties of the currently opened Package or diagram.
	Note: This button is only enabled when properties can be shown for the current item.

Content Area

WebEA's content area adjusts and adapts to display a custom layout for the currently selected item, depending on its type. For example, when a diagram is selected the content area consists entirely of the diagram image, but when a Package is selected the content area displays a view of child objects in that Package. The next three sections describe the main 'views' that are displayed in the content area.

Content Area - Object View

In Object View the child objects of a container are represented in one of two viewing styles, Icon or List view. In Icon view each object is represented by an image (and object name), in List view each object is represented as a row in a table, which includes columns for name, type, author and modified date. Each model has the ability to define a default viewing style in the configuration file; however, the user can change their style for the current session using the menu items in the 'Hamburger'.

Icon view

As an example, a root node with the name 'ASA Finance' would be represented by:

Whereas a View object with the name 'Class Model' would be represented by:
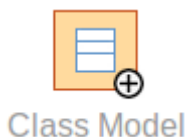
In Object View mode, objects are displayed in the predefined order: Packages, diagrams and elements; if there are multiples of any object type then they will be sorted in alphabetically order. This is different to Enterprise Architect's behavior, which can allow elements to be manually ordered.

As an example:
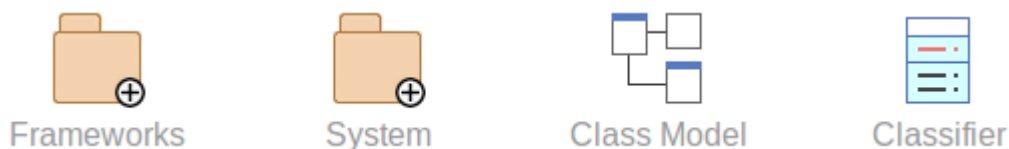
While navigating the model structure with WebEA you might see a number of symbols applied to the object images. Each symbol has a meaning: the ⊕ symbol indicates that the object contains children, whilst the 🔒 symbol indicates the object is locked to the current user.

The action taken when an item is selected is dependent on the object type of the item.

- For Packages, the default action is to navigate down one level of the Project Browser and display the child objects contained within that Package

- For diagrams the default action is to open the diagram image

- For elements the default action is to display the Properties view for the selected element

To open the properties of a Package while viewing it in the Object View, click on the ⓘ button in the Navigation bar.

In the case where a user navigates to a Package that does not have children, the message 'No child element(s)' will be displayed.

Adding new elements

If the model has been configured to allow the addition of elements and the current user has the required permissions, the

<New> icon will be shown as the last element of a Package's object view. For more details see the *Add Elements and Features in WebEA* Help topic.

Content Area - Diagram View

As mentioned earlier, when in Diagram View mode the content area displays the entire diagram as a single image. Objects on the diagram are selectable and, based on the object type, behave as explained here:

- Packages: opens the select Package in Object View mode
- Diagrams: opens the selected diagram image
- Elements: opens the selected element in the Properties view
- Hyperlink to external URLs: navigates away from WebEA and loads the external URL

Note: To open the properties of a diagram while viewing the diagram image click the ⓘ button in the Navigation bar.

Content Area - Properties View

The 'Properties' view has two formats, depending on which option you have selected from the 'Hamburger' menu - 'Wide view' or 'Split view' - and also depending on the size of your screen; narrow screens such as mobile phone screens in portrait orientation will automatically use the 'Wide' view, but adopt your setting in landscape orientation.

Wide View:

In 'Wide' view the main properties are displayed first, followed by the extended properties details.
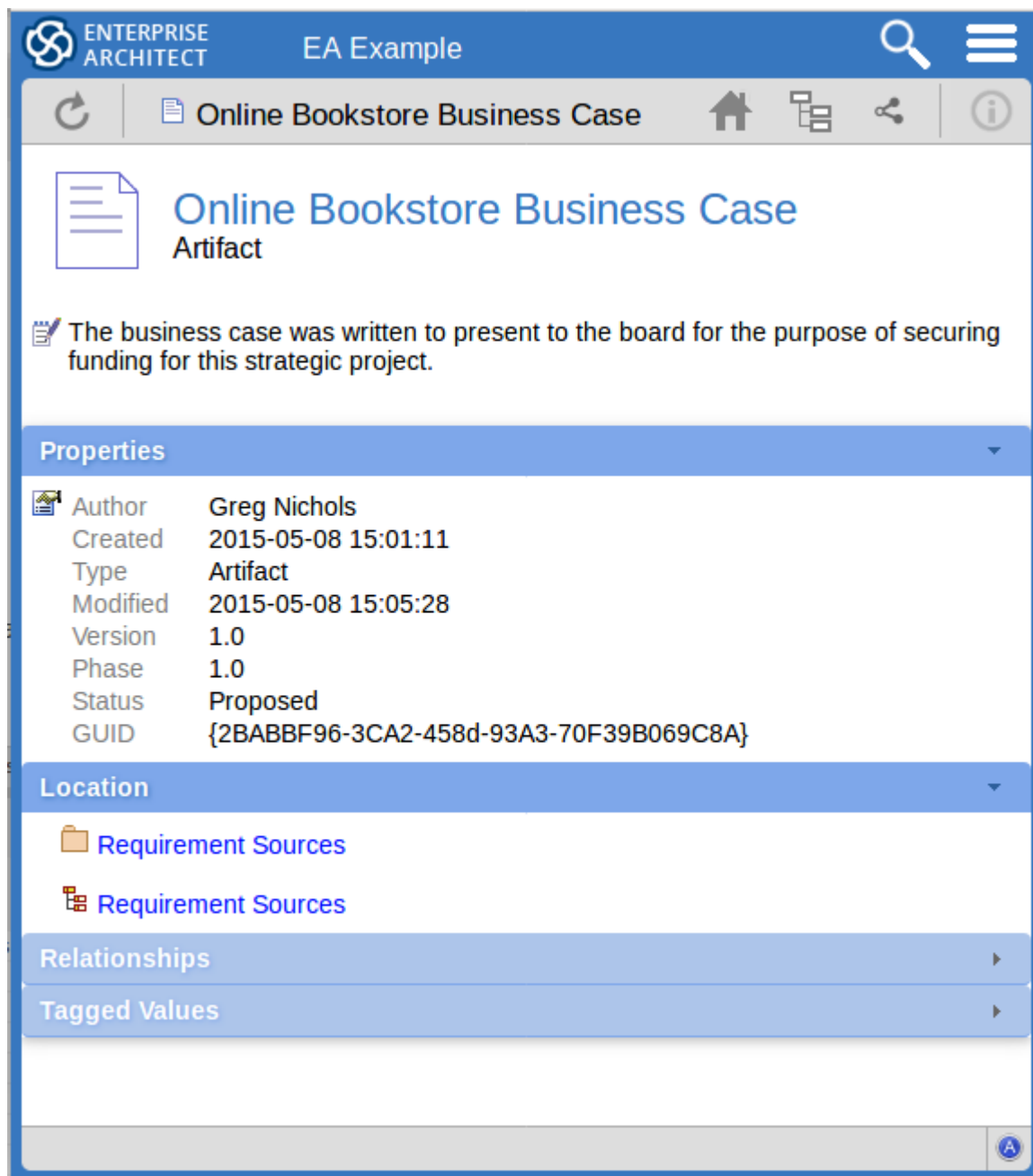
Split View:

In 'Split' view the display has two columns, the main properties on the left hand side and the extended properties on the right.

Notes:

- This option is disabled when the current device's screen does not support more than 800px
- The split percentage allocated to each section of the screen is fixed at 66% for the main properties and 33% for the extended properties
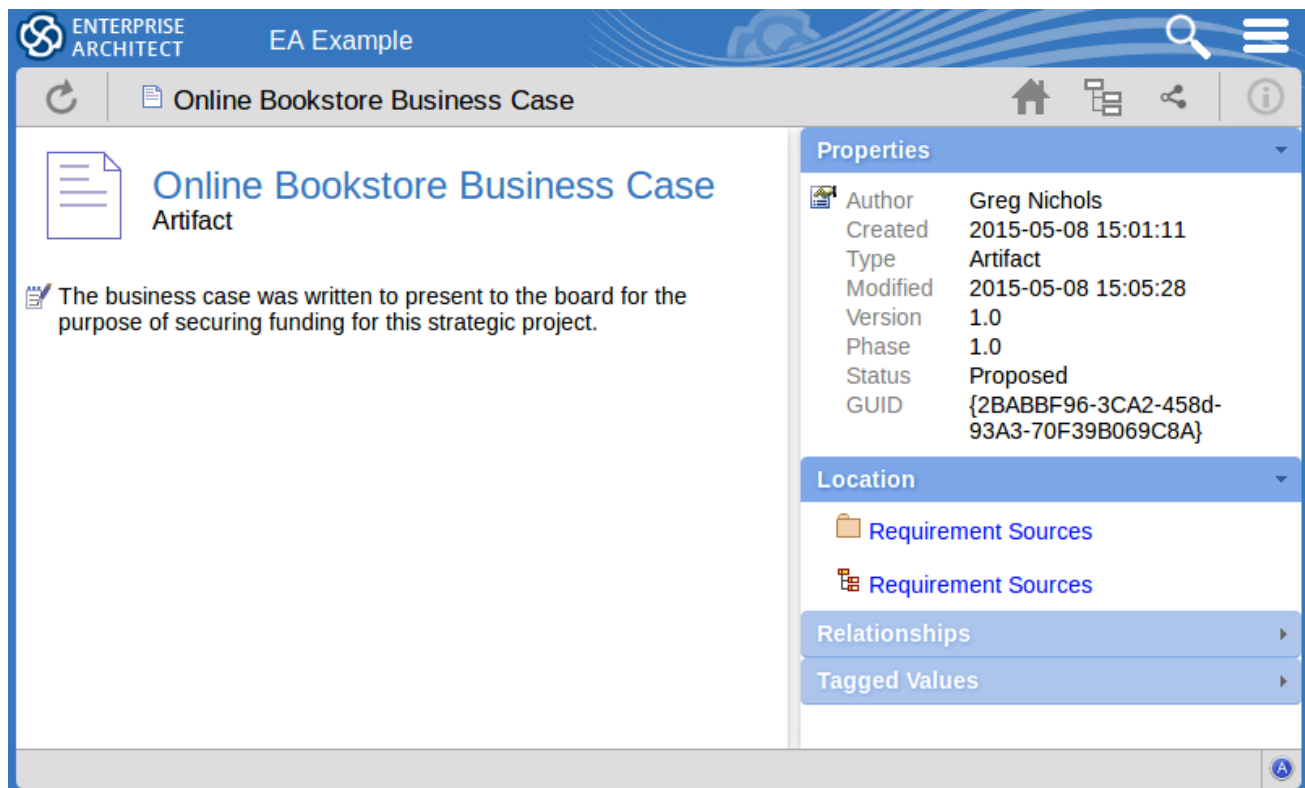
Main Properties:

The main properties section displays an image representing the object's type, its name, type and stereotype, followed by linked/encrypted/stored document information, the object notes and then Requirements, Constraints and Scenarios.

- If the model has been configured to allow it, there will be a element 'hamburger' icon that allows the creation of object features; see the *Add Elements and Features in WebEA* topic for further details

- The object's 'Notes' are formatted as rich text, with the previously applied fonts, colours and formatting

- If the element has a regular Linked Document applied, a line with 'Linked Document' and an Open button will be displayed immediately under the object's name and type fields; clicking on the Open button will display the contents of the linked document

- If the element has an encrypted Linked Document applied, in addition to the 'Linked Document' label and Open button a password textbox displays; if you enter the correct password and click or tap on the Open button the contents of the encrypted document will be displayed

- If the element has a stored Document applied, a 'Download' button (along with the label 'Stored Document') will be displayed; clicking on the Download button will save the document to the user's machine where they can use the appropriate application to view it

- If the element has a image asset applied, a 'Download' button (along with the label 'Image Asset') will be displayed; clicking on the Download button will save the image to the user's machine where they can use the appropriate application to view it

Hyperlinks in the object 'Notes' text will take you to whatever they are linked to within the model, and external sites (subject to your web browser allowing pop-ups) will open in a new browser tab or window.

Hyperlinks within linked documents will take you to whatever they are linked to within the model, and external sites (subject to your web browser allowing pop-ups) will open in a new browser tab or window.

Extended Properties:

In both Wide and Split views, the extended properties information is presented in horizontal panels that identify a type of information such as the object's Operations, Tagged Values, Change Requests, Location or Relationships (depending on the object type and, in some cases, which types of object and property you have access to).

You can collapse and expand each panel to hide or show that type of information, by clicking or tapping anywhere on the panel header. The collapsed/expanded state of each panel persists throughout your session. These panels will only be displayed when information of that type has been defined:

- Properties: general information about the object, such as author, time and date created and last modified, the version, phase, status, alias and GUID

- Location: information relating to the current object's parent and any diagrams that the current object is in

- Relationships: a list of incoming and outgoing connections for the current object; it is possible to display the properties of each connector and the involved object

    The relationship Properties page resembles this:

- Tagged Values: a list of defined Tagged Values for the current object

- Testing: a list of all defined tests for the current object; if you have the required permissions and the model has been configured correctly, clicking or tapping the pencil icon will allow you to edit the details of the selected test

- Resource Allocation: a list of all defined Resource Allocations for the current object; if you have the required permissions and the model has been configured correctly, clicking or tapping the pencil icon will allow you to edit the details of the selected resource allocation

- Attributes: a list of all defined attributes for the current object

- Operations: a list of all defined operations for the current object

- Changes: a list of all defined changes for the current object

- Defects: a list of all defined defects for the current object

- Issues: a list of all defined issues for the current object

- Tasks: a list of all defined tasks for the current object

- Events: a list of all defined events for the current object

- Decisions: a list of all defined decisions for the current object

- Effort: a list of all defined effort items for the current object

- Risks: a list of all defined risks for the current object

- Metrics: a list of all defined metrics for the current object

- Discussions: a list of all formal review discussions and informal discussions on the element; if the current model has been configured to allow it, you can contribute to discussions on the current object - for further information on these facilities, see the *Reviews in WebEA* and *Discussions in WebEA* Help topics

Status bar

The WebEA Status bar is displayed at the very bottom of the page, and is an area reserved for the display of information about the application. However, typically it will only consist of a single 'A' button in the right corner which when clicked or tapped will display the 'WebEA About' screen.

Review element

If you have joined a review, the name of the Review element will display in the status bar on the left side of the About button.

Add Elements and Features in WebEA

Depending on how the current model has been configured in the WebEA configuration file and your security access to the model, you might have the ability to create a range of objects in the model through WebEA. These objects include complete elements such as Use Cases, Requirements, Components, Changes and Issues, and features of elements such as Tests, Decisions, Defects and Events. You can therefore define additional scope or problems that have arisen, directly in the model, with precision and in detail, adding to the accuracy and quality of the solution that the model represents.

The Project Manager responsible for the model should guide you in how to create objects and in the purpose of you doing so.

In addition to adding new elements you might also be able to edit the Notes of any object regardless of type, and the details of element Tests and Resource Allocations for any element in the model, whether you have created it or not. See the *Edit Objects* Help topic.

Adding Elements

You can quickly establish if you have the ability to create elements in WebEA, by expanding a Package in the Hierarchy

view; if the <New> icon displays, you can create elements.

Expand the model Packages until you display the contents of the Package to which you want to add the element, and

click on the <New> icon. The 'Add Element' page displays.

Field	Description
Element type	Click on the drop-down arrow and select the type of element to create. Depending on your security permissions and the model's options, you can create some or all of

	these element types:
	- Package
	- Review
	- Actor
	- Change
	- Component
	- Feature
	- Issue
	- Node
	- Requirement
	- Task
	- Use Case
Name	Type in an appropriate name for the element, preferably giving some indication of its purpose or function.
Notes	Type in a more detailed description of the element, such as why you are creating it and what other elements it might be associated with.
	You can format this text using the facilities provided in the Notes toolbar. These are:
	- Bold
	- Italic
	- Underline
	- Numbered list
	- Bulleted list
	- Superscript text
	- Subscript text
	- Set color of selected text
Add	Click on this button to add the new element to the Package in which you are creating it. This message displays:
	Object added
	Either:
	- Work through the fields again, defining another new element in this Package, or
	- Click on the [icon] icon and select the Package line to return to the 'Hierarchy' view of the Package, now containing the new element

Adding Element Features

You can, again, quickly establish if you have the ability to create element features in WebEA, by displaying the

'Properties' view for an element; if the [icon] icon displays in the right corner of the element name/notes panel, you can add element features.

Display the 'Properties' view of the element to which you want to add a feature, and click on the [icon] icon. A menu

displays offering options to add to the element each feature that you have the access permissions to create. These features might include:

- Tests
- Resources
- Changes
- Defects
- Issues
- Tasks, or
- Risks

Select the appropriate option; a dialog displays in which you complete the fields to define the feature for the element. Each type of feature has a different dialog; refer to these topics for the details appropriate to each dialog.

- *Add Test Records*
- *Add Resource Records*
- *Add Change Requests*
- *Add a Defect*
- *Add an Issue Notification*
- *Add a Task*
- *Add a Risk*

Add Test Records

Add a Test

In the drop-down menu, select the 'Add Test' option. The 'Add Test to <element name>' dialog displays.

Mandatory fields on the dialog are indicated by a red asterisk (*).

Complete the fields as directed here.

Field	Action

Test name	Type an appropriate and meaningful name for the Test.
Class type	Click on the drop-down arrow and select the appropriate test group; that is: • Unit • Integration • System • Acceptance • Scenario, or • Inspection
Type	Click on the drop-down arrow and select the appropriate test type; that is: • Load • Regression or • Standard
Status	Click on the drop-down arrow and select the appropriate test status; that is: • Not Run (the most likely value for a new record from you) • Pass • Fail • Deferred or • Cancelled
Run by	If the test has been run, click on the drop-down arrow and select the name of the person who ran it.
Last Run	If the test has been run, click on the calendar icon and select the date on which the test was most recently run.
Checked by	If the test has been run, click on the drop-down arrow and select the name of the person who checked the test execution and results.
Description	Type in a description of the test, what it is intended to show and why it is necessary. You can format this text using the facilities provided in the Notes toolbar. These are: • Bold • Italic • Underline • Numbered list • Bulleted list • Superscript text • Subscript text • Set color of selected text
Input	Type a description of what actions or parameters are required as input to this test. You can format this text using the facilities provided in the Notes toolbar.
Acceptance Criteria	Type a description of what criteria must be satisfied to indicate that the test has

	passed.
	You can format this text using the facilities provided in the Notes toolbar.
Results	Type a description of the expected and/or actual results of this test.
	You can format this text using the facilities provided in the Notes toolbar.
Add	Click on this button to submit the details and create the Test record on the selected element.
	This message is displayed:
	Test <name> was successfully added to <element name>.
	Either:
	• Work through the fields again, defining another new Test on this element, or
	• Click on the browser's back button to return to the 'Properties' view of the element, which will be refreshed to show the new record in the 'Testing' panel

Add Resource Records

Add a Resource

In the drop-down menu, select the 'Add resource' option. The 'Add resource allocation to <element name>' dialog displays.

Mandatory fields on the dialog are indicated by a red asterisk (*).

Complete the fields as directed here.

Field	Action
Resource name	Click on the drop-down arrow and select the name of the person to be allocated as the resource.
Role	Click on the drop-down arrow and select the role of the resource in this work; for example: - Application Analyst - Business Analyst - C++ Programmer - Project Manager
Start Date	Click on the calendar icon and select the date on which the resource is expected to start the assigned work.
End Date	Click on the calendar icon and select the date on which the resource is expected to

	complete the assigned work.
% Complete	Click on the up and down arrows to change the field value to the percentage completion of work that the resource has achieved.
Description	Type in a description of the resource allocation - what the work is, why this (kind of) resouce is required to do it, and any criteria that must be satisfied. You can format this text using the facilities provided in the Notes toolbar. These are: • Bold • Italic • Underline • Numbered list • Bulleted list • Superscript text • Subscript text • Set color of selected text
Add	Click on this button to submit the details and create the Resource Allocation record on the selected element. This message is displayed: Resource allocation for <name> was successfully added to <element name>. Either: • Work through the fields again, defining another new Resource Allocation on this element, or • Click on the browser's back button to return to the 'Properties' view of the element, which will be refreshed to show the new record in the 'Resource Allocation' panel

Add Change Requests

Add a Change Request

In the drop-down menu, select the 'Add change' option. The 'Add change to <element name>' dialog displays.

Mandatory fields on the dialog are indicated by a red asterisk (*).

Complete the fields as directed here.

Field	Action
Name	Type an appropriate and meaningful name for the requested change.
Status	Click on the drop-down arrow and select the appropriate request status; that is: • New • Verified, or • Complete

Priority	Click on the drop-down arrow and select the appropriate priority value; that is: - High - Medium, or - Low
Requested by	Click on the drop-down arrow and select the name of the person requesting the change.
Requested on	Click on the calendar icon and select the date on which the request was made.
Completed by	If the change has been completed, click on the drop-down arrow and select the name of the person who completed it.
Completed on	Click on the calendar icon and select the date on which the requested change was completed.
Version	Type in a version number or letter to indicate any evolution of the same change. Generally, '1.0' indicates the first instance of a change.
Description	Type in a description of the change, what it is intended to be and why it is necessary. You can format this text using the facilities provided in the Notes toolbar. These are: - Bold - Italic - Underline - Numbered list - Bulleted list - Superscript text - Subscript text - Set color of selected text
History	If there is any background to this change, such as previous versions, document them here (you might copy the 'Description' from the last version or iteration of this change, for example). It is recommended that you include a date stamp and user initials in the entry. You can format this text using the facilities provided in the Notes toolbar.
Add	Click on this button to submit the details and create the change request on the selected element. This message is displayed: Change <name> was successfully added to <element name>. Either: - Work through the fields again, defining another new change request on this element, or - Click on the browser's back button to return to the 'Properties' view of the element, which will be refreshed to show the new record in the 'Changes' panel

Add a Defect

Add a Defect Record

In the drop-down menu, select the 'Add defect' option. The 'Add defect to <element name>' dialog displays.

Mandatory fields on the dialog are indicated by a red asterisk (*).

Complete the fields as directed here.

Field	Action
Name	Type an appropriate and meaningful name for the reported defect.
Status	Click on the drop-down arrow and select the appropriate report status; that is: • New • Verified, or • Complete

Priority	Click on the drop-down arrow and select the appropriate priority value; that is: • High • Medium, or • Low
Reported by	Click on the drop-down arrow and select the name of the person reporting the defect.
Reported on	Click on the calendar icon and select the date on which the report was made.
Resolved by	If the defect has been resolved, click on the drop-down arrow and select the name of the person who resolved it.
Resolved on	Click on the calendar icon and select the date on which the reported defect was fixed.
Version	Type in a version number or letter to indicate any evolution of the same defect. Generally, '1.0' indicates the first instance of a defect.
Description	Type in a description of the defect, what it is, what the solution might be, and why it is a problem. You can format this text using the facilities provided in the Notes toolbar. These are: • Bold • Italic • Underline • Numbered list • Bulleted list • Superscript text • Subscript text • Set color of selected text
History	If there is any background to this defect, such as previous occurences, document them here (you might copy the 'Description' from the last version or iteration of this defect report, for example). It is recommended that you include a date stamp and user initials in the entry. You can format this text using the facilities provided in the Notes toolbar.
Add	Click on this button to submit the details and create the defect report on the selected element. This message is displayed: Defect <name> was successfully added to <element name>. Either: • Work through the fields again, defining another new defect report on this element, or • Click on the browser's back button to return to the 'Properties' view of the element, which will be refreshed to show the new record in the 'Defects' panel

Add an Issue Notification

Add an Issue

In the drop-down menu, select the 'Add issue' option. The 'Add issue to <element name>' dialog displays.

Mandatory fields on the dialog are indicated by a red asterisk (*).

Complete the fields as directed here.

Field	Action
Name	Type an appropriate and meaningful name for the reported issue.
Status	Click on the drop-down arrow and select the appropriate issue status; that is: • New • Verified, or • Complete

Priority	Defaults to 'High'. If necessary, click on the drop-down arrow and select the appropriate priority value; that is: • Medium, or • Low
Raised by	Defaults to the first name in the list of project users. If necessary, click on the drop-down arrow and select the name of the person raising the issue.
Raised on	Defaults to today's date. If necessary, click on the calendar icon and select the actual date on which the issue was raised.
Completed by	If the issue has been resolved, click on the drop-down arrow and select the name of the person who resolved it.
Completed on	Click on the calendar icon and select the date on which the issue was resolved.
Version	Type in a version number or letter to indicate any evolution of the same issue. Defaults to '1.0', indicating the first instance of an issue.
Description	Type in a description of the issue, what it is, what the solution might be, and why it is a problem. You can format this text using the facilities provided in the Notes toolbar. These are: • Bold • Italic • Underline • Numbered list • Bulleted list • Superscript text • Subscript text • Set color of selected text
History	If there is any background to this issue, such as previous occurences, document them here (you might copy the 'Description' from the last version or iteration of this issue, for example). It is recommended that you include a date stamp and user initials in the entry. You can format this text using the facilities provided in the Notes toolbar.
Add	Click on this button to submit the details and create the issue report on the selected element. This message is displayed: Issue <name> was successfully added to <element name>. Either: • Work through the fields again, defining another new issue on this element, or • Click on the browser's back button to return to the 'Properties' view of the element, which will be refreshed to show the new record in the 'Issues' panel

Add a Task

Add a Task Record

In the drop-down menu, select the 'Add task' option. The 'Add task to <element name>' dialog displays.

Mandatory fields on the dialog are indicated by a red asterisk (*).

Complete the fields as directed here.

Field	Action
Name	Type an appropriate and meaningful name for the new task.
Status	Click on the drop-down arrow and select the appropriate task status; that is: • New • Verified, or • Complete

Priority	Defaults to 'High'. If necessary, click on the drop-down arrow and select the appropriate priority value; that is: • Medium, or • Low
Requested by	Click on the drop-down arrow and select the name of the person requesting the task. This defaults to the name selected for the previous record.
Reported on	Defaults to today's date. If necessary, click on the calendar icon and select the correct date on which the request was made.
Completed by	If the task has been completed, click on the drop-down arrow and select the name of the person who completed it.
Completed on	Click on the calendar icon and select the date on which the task was completed.
Version	Type in a version number or letter to indicate any evolution of the same task. Defaults to '1.0', indicating the first instance of a task.
Description	Type in a description of the task, what it is, what the objective might be, and why it is a required. You can format this text using the facilities provided in the Notes toolbar. These are: • Bold • Italic • Underline • Numbered list • Bulleted list • Superscript text • Subscript text • Set color of selected text
History	If there is any background to this task, such as previous occurences, document them here (you might copy the 'Description' from the last version or iteration of this task, for example). It is recommended that you include a date stamp and user initials in the entry. You can format this text using the facilities provided in the Notes toolbar.
Add	Click on this button to submit the details and create the task record on the selected element. This message is displayed: Task <name> was successfully added to <element name>. Either: • Work through the fields again, defining another new task request on this element, or • Click on the browser's back button to return to the 'Properties' view of the element, which will be refreshed to show the new record in the 'Tasks' panel

Add a Risk

A Risk is a condition that could cause the disruption, loss or compromise of a system. It can refer to either technical or business risks.

Add a Risk Record

In the drop-down menu, select the 'Add risk' option. The 'Add risk to <element name>' dialog displays.

Mandatory fields on the dialog are indicated by a red asterisk (*).

Complete the fields as directed here.

Field	Action
Name	Type an appropriate and meaningful name for the new risk.
Type	Type in the risk type.
Weight	Defaults to 1. Click on the up and down arrows to set a different weighting on the risk. There is no limit to the value that you can set, but within Enterprise Architect the usual value is a small number between, say, 1.0 (normal) and 5.0 (heavily weighted).
Description	Type in a description of the risk, what it is and what the mitigation might be. You can format this text using the facilities provided in the Notes toolbar. These are: • Bold • Italic • Underline
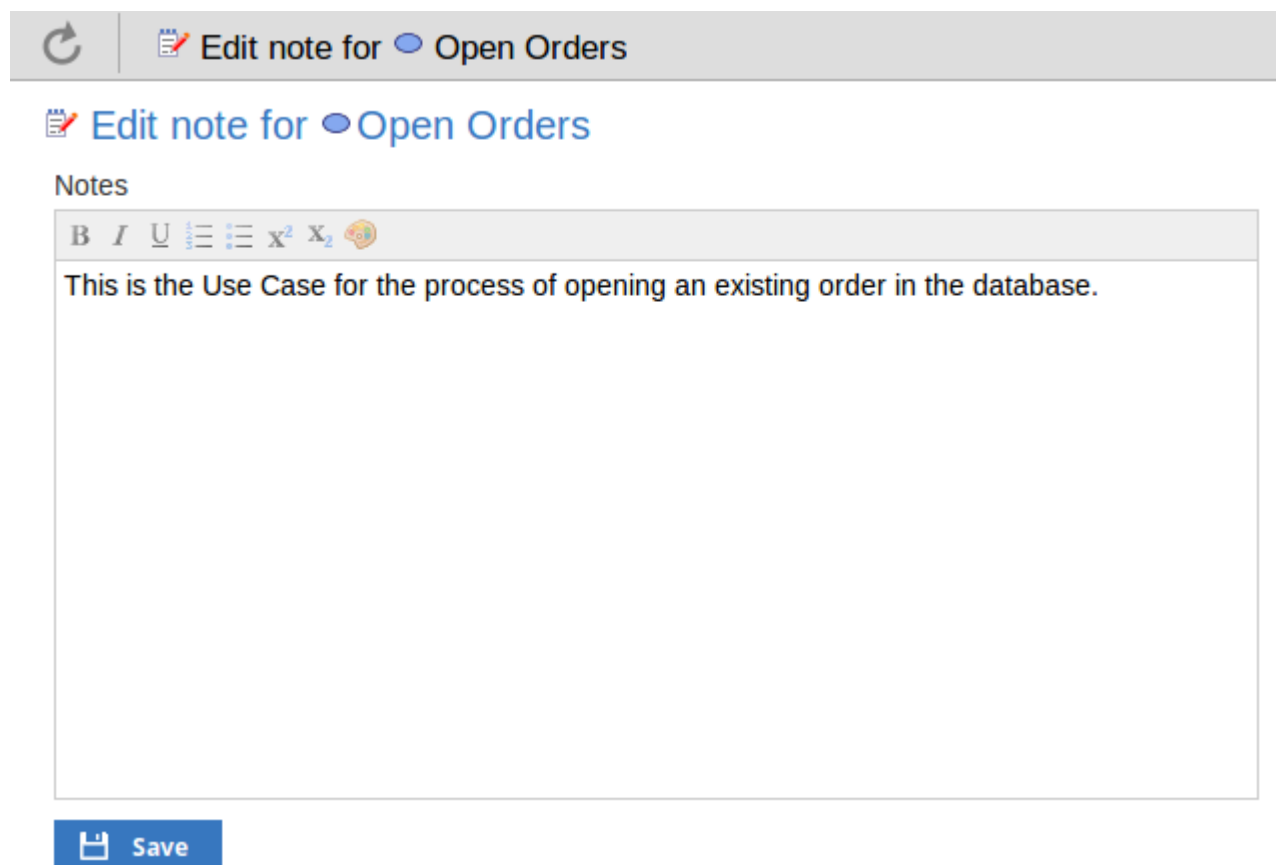
	• Numbered list
	• Bulleted list
	• Superscript text
	• Subscript text
	• Set color of selected text
Add	Click on this button to submit the details and create the risk record on the selected element.
	This message is displayed:
	Risk <name> was successfully added to <element name>.
	Either:
	• Work through the fields again, defining another new risk on this element, or
	• Click on the browser's back button to return to the 'Properties' view of the element, which will be refreshed to show the new record in the 'Risks' panel

Edit Objects

Within any model in which you have the security permissions to create elements, you can also edit the Notes, Resource Allocation and Testing details of any object in which they occur. On the right of the item field for these three properties you will see a ✏ icon, which indicates that you have the ability to edit that property.

Edit Notes

To edit the notes of an object, click on the ✏ icon for the field. The 'Edit note for <object type>' page displays.

You can add, delete and change text, and use the formatting tools in the Notes toolbar; that is, for a selected text string, add:

- Bold
- Italic
- Underline
- Numbered List
- Bulleted List
- Superscript
- Subscript
- Text color

You can also right click on the selected text and select options from the browser context menu to:

- Undo the last change

- Move (cut) the text string into the buffer for pasting

- Copy the text string into the buffer for pasting

- Paste text from the buffer into the 'Notes' field

- Delete the selected text

- Select all the text in the 'Notes' field

- Search for the selected text string on the internet using Yahoo!

- View the selected text in the object source code (of value to model coders)

- Turn spell-checking on and off (if on, possibly misspelt text is underlined in red)

- Add a different language dictionary to check spelling against (this would probably have been done already for your web browser)

- Inspect the object in which the text appears (of value to model coders)

Whilst you can edit the text of a hyperlink and delete the hyperlink, you cannot change the linked object or add a new hyperlink. The browser context menu also provides options for working with the hyperlink, but the WebEA interface cannot support these functions.

When you have finished editing the text, click on the Save button. A confirmation message briefly displays and you are returned to the object 'Properties' view.

Edit Tests

The tests assigned to an element support a range of situations including user acceptance and scenario. You could be asked to define or perform tests on parts of a model of relevance to you as a customer or user, or you might want to add information obtained from other people you have asked to perform an evaluation. Therefore, you might need to edit test records to update them with your information.

To edit a test record, display the appropriate element in the 'Properties' view, expand the 'Testing' panel and click on the

icon. The 'Edit Test for <element name' page displays.

Edit Test for <element name> page

Mandatory fields on the dialog are indicated by a red asterisk (*).

Complete the fields as directed here.

Field	Action
Test name	If necessary, type a new name for the Test.
Class type	In necessary, click on the drop-down arrow and select a different test group for the test; that is:

	• Unit
	• Integration
	• System
	• Acceptance
	• Scenario, or
	• Inspection
Type	If necessary, click on the drop-down arrow and select a new test type; that is: • Load • Regression or • Standard
Status	If necessary, click on the drop-down arrow and select a new test status; that is: • Not Run • Pass • Fail • Deferred or • Cancelled
Run by	If the test has been run, click on the drop-down arrow and select the name of the person who ran it.
Last Run	If the test has been run, click on the calendar icon and select the date on which the test was most recently run.
Checked by	If the test has been run, click on the drop-down arrow and select the name of the person who checked the test execution and results.
Description	Type in a description of the test, what it was intended to show and - if the test has been run - what the test situation and results were. You can format this text using the facilities provided in the Notes toolbar, as described in the *Edit Notes* section.
Input	Type a description of what actions or parameters are or were required as input to this test. You can format this text using the facilities provided in the Notes toolbar, as described in the *Edit Notes* section.
Acceptance Criteria	Type a description of what criteria must be or were satisfied to indicate that the test has passed. You can format this text using the facilities provided in the Notes toolbar.
Results	Type a description of the expected and/or actual results of this test. You can format this text using the facilities provided in the Notes toolbar.
Save	Click on this button to submit the details and update the Test record on the selected element. A confirmation message briefly displays and you are returned to the object 'Properties' view

Edit Resource Allocation

The resources allocated to an element can include reviewers in various roles, and you might be allocated as a resource yourself. Alternatively, you could have allocated another person as a reviewer. Therefore, you might need to edit a resource allocation to record either your own progress in a review, or the progress of the resource you have allocated.

To edit the resource allocation, display the appropriate element in the 'Properties' view, expand the 'Resource Allocation' panel and click on the ✎ icon. The 'Edit resource allocation for <element name' page displays.

Edit resource allocation for <element name> page

Notice that this page has a few more fields than the initial 'Add resource allocation ...' page.

Mandatory fields on the dialog are indicated by a red asterisk (*).

Complete the fields as directed here.

Field	Action

Resource	If necessary, click on the drop-down arrow and select the name of a different person to be allocated as the resource.
Role	If necessary, click on the drop-down arrow and select a different role for the resource in this work.
Start Date	If necessary, click on the calendar icon and select a different date on which the resource started or is expected to start the assigned work.
End Date	If necessary, click on the calendar icon and select a different date on which the resource completed or is expected to complete the assigned work.
% Complete	Click on the up and down arrows to change the field value to the percentage completion of work that the resource has achieved.
Expected	Type in the number of whole time units the task is expected to take. The value must be an integer; you cannot record part units.
	The unit of time you use will be identified by the Project Manager, and depends on the granularity of the work being recorded. Most tasks are completed in a number of hours or a number of days; use the smallest practical unit that you can record as a whole number, and use the same units for the expected, allocated and actual times.
Allocated	Type in the number of whole time units the task can be spread over. For example, you might expect the task to take 2 days of work performed at any point over a period of 5 days.
Actual	If the task is 100% complete, type in the number of whole time units it actually took.
Description	If necessary, edit the description of the resource allocation, perhaps to explain the changes you have made to earlier fields.
	You can format this text using the facilities provided in the Notes toolbar, as explained in the *Edit Notes* section.
	Before changing the text, you might want to copy it to paste into the 'History' field for editing there.
History	Optionally, paste or type in an explanation of what has previously taken place with this Resource Allocation. You can also format this text, using the facilities in the Notes toolbar.
Save	Click on this button to submit the details and update the Resource Allocation record on the selected element.
	A confirmation message briefly displays and you are returned to the object 'Properties' view.

Reviews in WebEA

Review elements are a convenient way for groups of users to collaboratively discuss one or more model objects and capture those discussions as a formal review. A Review element is a custom element that captures all discussions that you participate in over a period of time. That period begins when you select to 'join' the review, and ends either when you select to leave the review, or when you log out of the current model. You can also return to the review in a subsequent work session and your continuing discussions are also captured. The Review element would typically have a start and end date set to define the duration of the review.

When a Review element is created, a child diagram is also created, automatically. Both Enterprise Architect and WebEA consider any element added to this diagram as requiring comment during the Review. When the properties of the Review element are loaded in WebEA the 'Objects in Review' section will contain a list of all elements that have already been commented on or have been added to the Review diagram. This list provides a convenient way to quickly navigate to each element in the Review.

When the properties of a Review element are loaded in WebEA, a section of information is presented immediately under the element's notes, providing a summary of the review such as the status, start and end dates, number of elements discussed and yet to be discussed, and a break down of review topics based on priority and status, as illustrated here:

The 'Objects in Review' section immediately follows the Summary section. This displays a list of elements to be discussed during the review. Elements that have not yet been discussed are presented first in the list, followed by elements that have been commented on, showing each review topic (and, when you click on the expansion arrow, its replies), as illustrated here:

The name of the element on which the discussion point is raised acts as a link to open that element's own 'Properties' display. The review discussions are displayed in the 'Reviews' panel. If you have currently joined a review then the 'Reviews' panel is filtered to show only discussions for that review.

If you are not in a review then the 'Reviews' panel might show multiple reviews in which the element has been discussed. In either case the review name acts as a link to the Review element.

You can only participate in a single review at any given time. If necessary you can leave a review and join it again at a later time or date.

Locating Reviews

There are a number of ways in which to locate existing Review elements to join, or to switch to your active review (the review that you have already joined).

To check on recent Review elements available in the model, click on the 'Search' icon in the WebEA header and select the 'Reviews' option. This conducts a search for recent Review elements and lists them in the 'Search Results' view, with the most recently changed element at the top of the list.

You can click on the name of a Review in the list to load the details of the selected Review element, and then join or leave the Review as appropriate.

The name of your active Review is displayed in the WebEA's Status bar, and you can click on the name (in the Status bar) to load the details of the active Review element.

Join a Review

While the property details of a Review element are loaded, the 'Join Review' button will be enabled.

Clicking on this button will join you to the review and link your discussions to the Review element. The display will automatically be adjusted to the change of state. These changes include:

- The addition of the joined Review element's name to the status bar
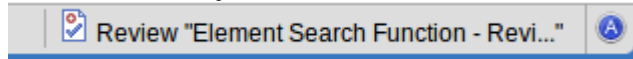
- The 'Review' panel filtering elements to the current Review element only
- The 'Discussion' panel's update/add actions being disabled
- The Join Review button being disabled, and the Leave Review button being enabled

There are some situations where the 'Join Review' button is already disabled for a Review element. If you are using WebEA on a desktop device, use the mouse to hover over the button to display a message explaining why the button is disabled. The reasons for not being able to join a review include:

- You are already joined to a review on a different Review element
- The loaded Review is closed

The Join Review button and Leave Review button are only visible on Review elements when the 'participate_in_review' configuration option is set to true for the current model. See the *How to configure WebEA models* Help topic for more information.

Add topics to a Review

When you have opened and read through the contents of a Review element, you can add your own review topics or respond to existing points. To do this you click on the name of the element you want to comment on, and display the Properties view for that element. The Properties view includes a 'Review' panel containing any existing review topics and a 'Create Review Topic' prompt.

Note that the panel also contains a reminder that - whilst you are joined to a review - the contents of the panel are filtered to show only the review comments for that review.

Add a Review Topic

To create a review topic on a new point, simply start typing in the blank field. If you want to have separate paragraphs,

press the Enter key to start each new line.

Note: WebEA does not support editing (or deleting) review comments, therefore you should take care to type messages in clearly the first time, and double check before submitting.

Depending on how the WebEA model is configured, you might connect to it using your personal user ID, or as a single generic User that is shared by multiple people. In the second case, Sparx Systems recommends that each user identifies themselves in each message, either at the beginning or at the end.

When you have finished typing your message, submit it by clicking on this button:

Whilst you are adding comments to a review, at any point you can return to a posted topic and add or change its status and priority. To the right of the review topic text are two icons. The first is the 'Status' icon, and the second is the 'Priority' icon. The appearance of the icons depends on the current setting for status and priority. To set a status or priority, simply click on the appropriate icon and select from the listed options. These are:

- Status

  ● Open

  ◆ Awaiting Review

  ▣ Closed

- Priority

  III High

  II Medium

  I Low

  — <none>

Add a reply to a comment

If you want to respond to either a review topic or a reply to that topic, click or tap on the appropriate message. A blank field and button underneath the conversation thread will appear; type in your message and click or tap on this button:

Your reply is added to the end of the thread.

Note: Currently the review threads are only read from the database when the page is loaded, which means if other users make additional comments these won't be displayed until the page is reloaded or refreshed.

Review another element

Click on the name of the Review element in the 'Reviews' panel or in the page footer, to display the Properties view for that review element. Select another element in the review from the list.

Leave the current Review

When you have finished contributing to the review, you can either return to the Review element Properties view and click on the Leave Review button, or log out of the current model.
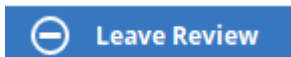
⊖ Leave Review

The Leave Review button will only be enabled when the details of the currently joined Review element are loaded.

Creating a Review

Typically, only a Project Manager will create Review elements in Enterprise Architect, notifying the relevant users of their location so that each user can join and leave the review(s) as necessary.

However, a WebEA user might create a Review element as a direction to the Project Manager to set up. In this case you create the Review element by clicking on the 'New' icon in the Package hierarchy view, selecting 'Review' as the element type. The procedure is as described in the *Add Elements and Features in WebEA* Help topic.

You require specific access permissions to create a Review element; those permissions are totally separate from the permission to join a review.

Discussions in WebEA

Enterprise Architect has a Discussion facility through which users can post short messages to have informal discussions with other users on a specific element or Package. As a WebEA user, if the model has been configured to allow discussions to be viewed and added, you can also read and participate in these discussions. The facility is very useful in informal reviews, as you can comment or respond immediately on a specific point as you come across it, without having to collate a list of comments and compose a relatively long document or email, or schedule a telephone call.

In WebEA the Discussion feature is presented as the last panel in the 'Properties' view for an element or Package.

Note that if you are joined to a formal review, you can read existing discussion points but you cannot respond to them or add new topics.

The input area shows as a blank field and a 'Discussion' button to save the entered text; if there has already been some discussion on the element, the comments of each discussion display above the input field.
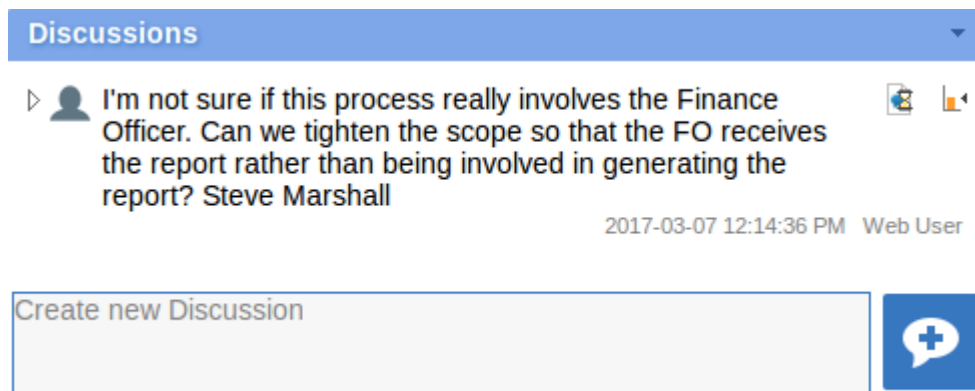
There are two types of discussion messages, an initial message (usually called a discussion) and replies. Replies are responses to an initial message, and are displayed as child items.

Avatars

In Enterprise Architect there is an application option for the current users that controls if avatars of security users are displayed with each discussion item. In WebEA this option is defined for a model in the configuration file; see the option *use_avatars*.

When a model has been configured to 'use avatars' and the model has security enabled, then instead of displaying generic blue speech bubbles the author's avatar is displayed, as shown here:

Starting a discussion

To start a discussion on a new point, simply start typing in the blank field. If you want to have separate paragraphs, press the Enter key to start each new line.

Note: WebEA does not support editing (or deleting) discussions, therefore you should take care to type messages in clearly the first time, and double check before submitting.

Depending on how the WebEA model is configured, you might connect to it using your personal user ID, or as a single generic User that is shared by multiple people ('Web User' in the first illustration). In the second case, Sparx Systems recommends that each user identifies themselves in each message, either at the beginning or at the end.

When you have finished typing your message, submit it by clicking on this button:

Replying to a discussion

If you are reading an existing discussion and want to respond to either the opening point or a reply to that point, click or tap on the appropriate message. A blank field and button underneath the conversation thread will display; type in your message and click or tap on this button:

Your reply is added to the end of the thread.

Note: Currently the discussion threads are only read from the database when the page is loaded, which means if other users make additional comments these won't be displayed until the page is reloaded or refreshed.

Status and Priority

If you are adding comments to a discussion, at any point you can return to a posted topic or reply and add or change its status and priority.

To the right of the discussion text are two icons. The first is the 'Status' icon, and the second is the 'Priority' icon. The appearance of the icons depends on the current setting for status and priority. To set a Status or Priority, simply click on the appropriate icon and select from the listed options. These are:

- Status

  ⊖ Open

  ◆ Awaiting Review

  ▣ Closed

- Priority

  Ⅲ High

  Ⅱ Medium

  Ⅰ Low

  ─ <none>

Notes

- As a WebEA user you cannot delete discussions or replies; however, users of Enterprise Architect can, so you might find that sometimes discussion postings disappear.
  This can be done for various reasons but the most common is that an inaccuracy was found in a message.

Create and check Watch Lists

The Watch List facility in WebEA helps you to quickly identify recent activity in the model. Each user can tailor their Watch List to monitor activity in any of a wide range of objects and features with various characteristics. Whenever the Watch List Summary is generated, all object types of interested are checked for recent activity and a count of matched objects displayed. All matched object counts can be clicked on to display the list of objects that match the particular activity type.

Note: that each model has a default set of activity types that have been enabled in the WebEA configuration file; each user can override the model defaults and save their customized settings.

Watch Lists Summary
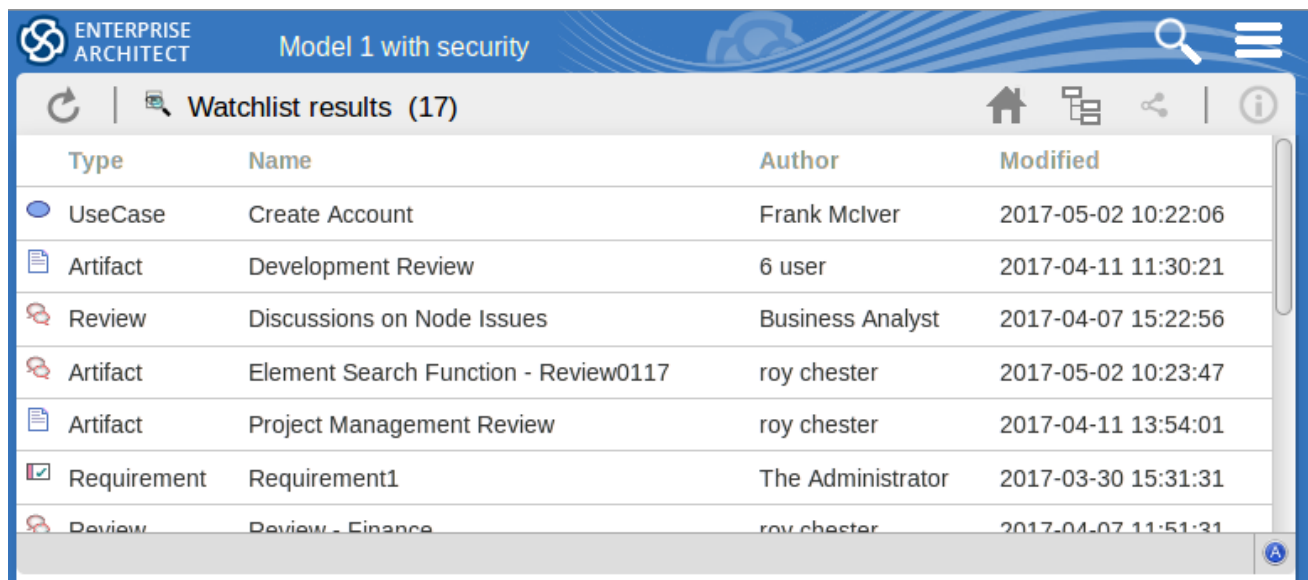
You can select the Watch List facility by selecting the 'Watchlist' option in the 'Search' menu. This immediately displays the 'Watchlist summary' view that shows:

- Any activity that has recently been detected, for

- All objects and features that are to be monitored

If there is no activity for a configured activity type, that activity type is omitted from the 'Watchlist Summary'. The display might look like this:
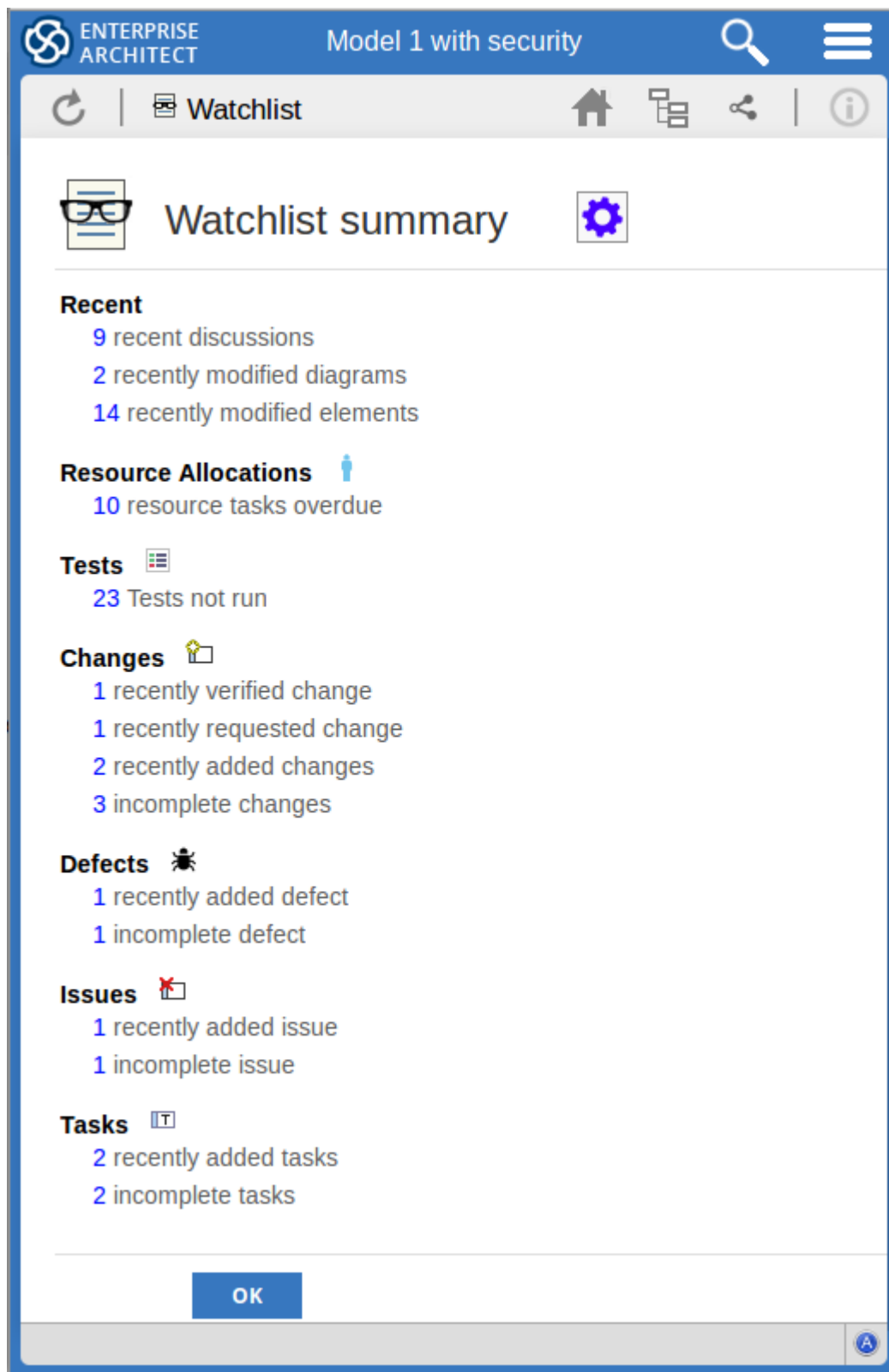
Every type of object on the report is represented by a line providing the number of objects that matches the given criteria. The number of the objects is a link which, if you click or tap it, displays a results page listing the specific objects that match the given criteria.

The watch list results page is the same as the custom search results page, in that if you then click or tap on an individual item in the list, the 'Properties' view will be displayed for the selected object. See the *WebEA's Layout* Help topic.

If you have finished checking the 'Watchlist summary', you can click on the OK button to return to the WebEA view you were displaying previously.

Alternatively, you can tailor the Watch List to provide information relevant to you; see *Configure the Watchlist*.

Configure the Watchlist

If you want to customize the activity types monitored in your Watchlist summary, click or tap the configure button located in the Watchlist summary header. The 'Watchlist configuration' page displays.

In the 'Number of Days' field, type the number of days that should be considered. This might be the number of days since you last used WebEA, or the number of days in a reporting period that you want to compare with.

For the remaining fields, select the checkbox against all items that you want to monitor and include in your watch list. The complete list is:

- Recent:
    - Discussions
    - Active reviews
    - Modified diagrams
    - Modified elements

- Resource Allocations:
    - Active
    - Ending today
    - Overdue

- Tests:
    - Recently passed
    - Recently failed
    - Recently deferred
    - Recently not checked
    - Not run

- Changes, Defects, Issues, Tasks and Decisions:
    - Recently verified
    - Recently requested
    - Recently completed
    - Recently added
    - Incomplete

- Events:
    - Recently requested
    - Recently added
    - Incomplete

Once you have finished choosing which items to monitor, click or tap on the Save button to save the configuration and redisplay the Watch List summary.

Note:
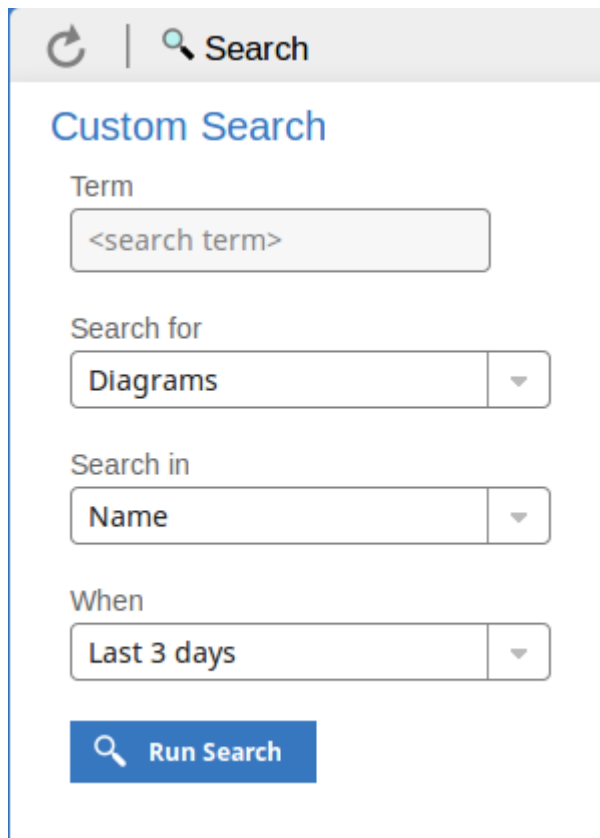
- All watch list options are saved based on current model and User ID for the current browser, and stored as cookies (where possible) so that the options are remembered between sessions

- If your browser does not allow the saving of cookies, then your customized watch list setting will only last as long as the current session

Search the model

WebEA provides a facility that enables you to search through the model for a specific item or group of items that have certain characteristics. This can help you to quickly locate objects of importance, such as elements you have created yourself (if you have the access privileges to do so).

You could use the Search facility to locate recent work on various objects, as identified by the Watchlist. See the *Create and Check Watchlists* Help topic.

You access the Search facility by selecting the 'Custom search' option from the main Search menu.

Search Specification

Define your search using these options.

Option	Description
Term	Enter the string of characters or digits you want to locate - name, Notes text or the complete object ID. Be aware that for some DBMSs - such as Firebird or Oracle - searches are case-sensitive.
Search for	Make a selection from the following object types: • 'Diagrams' • 'Elements' • 'Packages' • 'Discussions', or • 'Reviews' If you select 'Discussions', the 'Search in' and 'Term' options are not relevant and so

	are grayed out and unavailable. The search is for elements having active discussions within the specified time period. Note: the Discussion option are only available in models that have their *show_discuss* set to true in the WebEA configuration file, and the Review option is only available when *participate_in_reviews* is true.
Search in	Make a selection from the following list of fields to search: • Name • Name, Alias and Notes (the proper name, an assigned alternative name, and any text written in the object's 'Notes' field) • Author (the person who created the object) • ID (the GUID of the object, displayed as the last item in the 'Properties' panel when you expand it)
When	Make a selection on the period of time that should be searched. You can select to search for objects modified: • Today • Within the last 3 days • Within the last 7 days • Within the last 14 days • Within the last 30 days • Within the last 12 months • At any time
Run Search	Click or tap this button to execute the search and locate the objects that meet the search criteria.

Search Results

When you execute the search the results are displayed in the main content area.

The results provide the object name, type, author and last-modified date of each object that matches the search criteria. If you click on an item, the 'Properties' view will be loaded for that object.

Key Store

The Sparx Systems Key Store and floating license server has been available for many years, providing users of Enterprise Architect with the ability to manage their floating licenses. Up until the creation of Pro Cloud Server, the Key Store was only offered as an independent Windows service. Now that it has been integrated into the Pro Cloud Server, users only need to install, configure and maintain one service for all their Enterprise Architect needs.

In order to make use of the Key Store service that is provided with the Pro Cloud Server, you will require Enterprise Architect version 13.5 or later. It is expected that this will be available in late Q2 2017.

Topics that describe installation, use and problem-solving of the Key Store will be made available at the time Enterprise Architect 13.5 is released.

Install and Config

The Sparx Systems Pro Cloud Server product is installed using a standard Windows Installer package that includes a number of components. The Key Store and floating license server is included in the main 'Sparx Cloud Services' component, and consists of a number of extra configuration files over the standard Sparx Systems Cloud Service.

It is possible to change the install location of the Cloud Service; however if the defaults are used the Service will be installed into the C:\Program Files (x86)\Sparx Systems\Cloud Services\ folder (assuming a 64bit machine is being used).

In order to make use of the Key Store service that is provided with the Pro Cloud Server, you will require Enterprise Architect version 13.5 or later. It is expected that this will be available in late Q2 2017.

Information on configuring the key store/floating license server will be provided at the time Enterprise Architect 13.5 is released.

User Guide

In order to make use of the Key Store service that is provided with the Pro Cloud Server, you will require Enterprise Architect version 13.5 or later. It is expected that this will be available in late Q2 2017.

Information on how to use the Sparx Systems Key Store will be provided at the time Enterprise Architect 13.5 is released.

Troubleshooting

In order to make use of the Key Store service that is provided with the Pro Cloud Server, you will require Enterprise Architect version 13.5 or later. It is expected that this will be available in late Q2 2017.

Information on solving problems that might arise in the configuration and use of the Key Store will be provided at the time Enterprise Architect 13.5 is released.

Visibility Levels

The 'Visibility Levels' feature within Enterprise Architect provides the ability to restrict access to certain sections of a model. Only models hosted by a Pro Cloud Server are able to make use of this feature. Visibility Levels are implemented by taking advantage of the Row Level Security feature included in Oracle 8+ and Microsoft SQL Server 2016 (onwards).

Row Level Security is a security feature that controls which database users can access which rows of data in a database. Traditional database security only provides the ability to control which users have access to read, write or delete data down to the table level. Since Row Level Security functionality is implemented by the database, it is impossible for users to bypass security implemented at the application level, by writing their own scripts or databases queries. That is, the database ensures that users can only view and update data they are authorized to access.

Since Row Level Security is functionality implemented by the database, most of the configuration for Enterprise Architect and the Pro Cloud Server to support it is performed directly in the database without needing to change the application code base, which means there is less to go wrong from an application point of view.

Database / Application Users

It is important to understand the distinction between database and application users. The above discussion mentions that visibility levels are based on Database User permissions. In the Sparx System Cloud Server environment each Database Manager is defined with all of its connection details (including the user and password) to a given database. It is the database user, and is user used in the connection details that visibility levels are based on.

This concept is not to be confused with Enterprise Architect model users, ie the user ID/password that is entered when accessing a security enabled model. This is an application user.

Note: Enterprise Architect's model security is not mandatory, whereas all DBMS's require a database user to be defined in order to access data within the database.

How it works

In additional to their normal permissions, database users are granted access to one or more visibility levels.

A new column, VIS_LVL (Visibility Level), is added to 24 critical tables within the Enterprise Architect database. This field stores the visibility level that is needed to view/update this record.

For each of the 24 tables a security policy is defined that ensures each database user only sees records that their assigned visibility level is authorized to view.

Database triggers are then used to maintain the value of the VIS_LVL columns in all tables for all child objects and elements. For example, setting the visibility level of a Package in Enterprise Architect's Project Browser will update the selected Package and all of its children to the entered visibility level.

Note: The base scripts supplied by Sparx Systems support 20 'levels' of security (ie 1 to 20), however this can easily be adjusted by changing the number of rows that are inserted into the T_VISIBILITYLEVELS table at the beginning of the script. Level 0 is a special case, it represents the default value and implies that all database users (even those without a assigned visibility level) can view or update records of this level.

While the supplied scripts create could be adjusted to support more levels; however, this will need to be done manually.) The visibility levels are cumulative; for example, a user with level 3 can see all level 1, 2 and 3 rows.

Install and Config

The Sparx Systems Pro Cloud Server product is installed using a standard Windows Installer package that includes a number of optional components. One of these components is Visibility Levels, which is installed by default. It is possible to change the install location of Visibility Levels files; however, if the defaults are used the Visibility Levels files will be installed into the C:\Program Files (x86)\Sparx Systems\Cloud Services\RLS\ folder (assuming a 64bit machine is being used).

Requirements

In order to use Visibility Levels you must have:

- A licensed Pro Cloud Server
- An Oracle 8+ or SQL Server 2016+ based Enterprise Architect model/repository that uses the EA1220 schema definition
- An SQL based management console tool that supports the execution of SQL scripts
- Enterprise Architect build 1351 or later

What is in the package?

The Visibility Levels folder consists of a number of SQL script files that will either apply or remove the required database changes to an Enterprise Architect model so that it can support Visibility Levels.

The implementation of Row Level Security in Oracle is completely different to the implementation in SQL Server. While the end result behaves the same way, the physical objects that are needed by each DBMS are so different that each needs its own creation script.

Task	Script
Apply Visibility Level changes to Oracle model	EASchema_RLS_Updates_Oracle_1220.sql - Oracle SQL script that will update an existing Enterprise Architect Oracle based repository so that it supports Visibility Levels, by creating a set of functions, procedures, policies, triggers and columns.
Apply Visibility Level changes to SQL Server model	EASchema_RLS_Updates_SQLServer_1220.sql - SQLServer specific SQL script that will update an existing Enterprise Architect MS SQL Server based repository so that it supports Visibility Levels, by creating a set of functions, procedures, policies, triggers, roles and columns.
Remove Visibility Level changes from Oracle model	EASchema_RLS_Removal_Oracle_1220.sql - Oracle SQL script that will remove the functions, procedures, policies, triggers, roles and additional columns that were added to support Enterprise Architect's Visibility Levels. Note: all basic Enterprise Architect modeling data will remain unchanged by the removal script.
Remove Visibility Level changes from SQL Server model	EASchema_RLS_Removal_SQLServer_1220.sql - SQLServer SQL script that will remove the functions, policies, triggers, roles and additional columns that were added to support Enterprise Architect's Visibility Levels. Note: all basic Enterprise Architect modeling data will remain unchanged by the removal script.

How to execute

In order for an Enterprise Architect model to support Visibility Levels, the appropriate update script must be executed on that model's database. The script is standard SQL, so any tool can be used; however, you must connect using an ID with owner privileges because updates will be made to the physical schema.

Script Execution for Oracle

1. Identify the Enterprise Architect model database to which to apply Visibility Levels.
2. Ask all users to log out of the model.
3. (Optional step) Physically stop any further user access to the database; for example, disconnect the server from the network.
4. Perform a complete backup of the database.
5. Open your SQL console of choice and connect to the database as the owner.
6. Load the Oracle script ('EASchema_RLS_Updates_Oracle_1220.sql') into the console.
7. Review the comments at the top of the script, in particular the section that outlines the tasks that need to occur before running the script.
8. Make the needed adjustments to the script for your environment/preferences.
9. Execute the script.
10. Follow the set of instructions documented at the top of the script in the 'Do the following AFTER running this script' section
11. Perform a second complete backup of the database, as a restore point.
12. Define the Database Managers in the Pro Cloud Server (refer to the *Defining Visibility Level Aware Database Managers* table in the Visibility Levels *User Guide*).
13. (Optional step) If Step 3 was performed, re-enable regular user access to the database.
14. Open Enterprise Architect, connect to the Pro Cloud model with the highest Visibility Level (as defined in step 11) and set the Visibility Level on all Packages to their required values.

Script Execution for MS SQL Server

1. Identify the Enterprise Architect model database to which to apply Visibility Levels.
2. Ask all users to log out of the model.
3. (Optional step) Physically stop any further user access to the database; for example, disconnect the server from the network.
4. Perform a complete backup of the database.
5. Open your SQL console of choice and connect to the database as the owner.
6. Load the SQL script ('EASchema_RLS_Updates_SQLServer_1220.sql').
7. Review the comments at the top of the script, in particular the section that outlines the tasks that need to occur before running the script.
8. Make the needed adjustments to the script for your environment/preferences.
9. Execute the script.
10. Follow the set of instructions documented at the top of the script in the 'Do the following AFTER running this script' section
11. Perform a second complete backup of the database, as a restore point.

12. Define the Database Managers in the Pro Cloud Server (refer to the *Defining Visibility aware Database Managers* table in the Visibility Levels *User Guide*).

13. (Optional step) If Step 3 was performed, re-enable regular user access to the database.

14. Open Enterprise Architect, connect to the Pro Cloud model with highest Visibility Level (as defined in step 11) and set the Visibility Level on all Packages to their required values.

User Guide

The functionality of the Visibility Levels feature is primarily restricted to System Administrators, in order to setup and configure models to limit access to certain Packages in a model. Generally, users will not be aware of Visibility Levels; they simply won't see certain Packages in the model.

Defining Visibility Level Aware Database Managers

For the most part, defining Database Managers in the Professional ('Pro Cloud') version of Sparx Systems Cloud Service Configuration Client is the same process as for the regular version. These are the exceptions:

DBMS	Discussion
Oracle	All Oracle connections require a 'DefaultSchema={schema_name};' option manually added to the connection string, where {schema_name} is the physical name of the schema that owns the Enterprise Architect tables. For example: Provider=OraOLEDB.Oracle.1;Password=pass;Persist Security Info=True;User ID=user1;Data Source=ORA_SN;DefaultSchema=owner_user; In a typical Oracle connection to an Enterprise Architect model the 'User ID' option is both the owner of the tables and the user credentials that are being used to access the data. However, in a model using the Visibility Levels feature, these options are different, so the 'DefaultSchema' option has been added to define the table owner.
Microsoft SQL Server	All SQL Server connections require an 'Alias' field value that uniquely identifies the database manager on the current server. The 'Alias' value is the name by which Enterprise Architect users know the model. In previous versions of the Cloud Service, the database manager alias for SQL Server-based connections was automatically determined by the database name, which meant that each SQL Server database could only ever have one database manager defined in the Cloud Service. Now with the use of the 'Alias' field a single database can have an unlimited number of database managers defined.

Enabling the Extended OLSC Interface on a Database Manager

In order for the Pro Cloud Server to allow the use of Visibility Levels in a model the 'Enable extended OSLC interface for WebEA' option must be selected. This is achieved by opening the Sparx Systems Cloud Services Configuration Client application (SSProCloudClient.exe) connecting to the appropriate server. You then select the database manager in the list and click on the Configure button. On the resulting Configure Database Manager screen select the 'Enable extended OSLC interface for WebEA' checkbox.

Visibility Level Administration

When Enterprise Architect is used to connect to a model that has Visibility Levels the only new feature is the 'Set Visibility Level...' menu item available under the 'Package Control' item in the context menu of the project browser.

If your model has security enabled then all application users that require access to the 'Set Visibility Level...' menu item will need to be allocated the 'Visibility Level Admin' security permission, using standard Enterprise Architect permission assignment.

Set a Package's Visibility Level

From within Enterprise Architect, the Visibility Level of a Package can be set using the Project Browser context menu item 'Package Control | Set Visibility Level...'.

This menu item will only be available in models that have been set up for Visibility Levels and when they are being accessed via a Pro Cloud server connection.

Once a Visibility Level has been set on a Package, that Package and all of its children can be viewed, accessed and updated only by users with the corresponding access level or higher, set in user security.

To remove a visibility level from a Package (and all its children), simply set the Visibility Level back to zero.

Visibility Levels users can apply

Users can only set the Visibility Levels on Packages to levels the current database user has access to. For example, a Enterprise Architect client connected to a Cloud connection that uses a database user that has been assigned levels 3 & 4 can update the Visibility Level of any Package to the levels 0, 3 or 4.

Troubleshooting

Set Visibility Level menu item not available

The 'Set Visibility Level' menu item can be hidden from users for a number of reasons:

- You are not connecting to the model via a Cloud connection

- The Cloud Server you are connected to is not the Professional edition (ie Pro Cloud)

- The Cloud Server you are connected has an expired license and is now operating as the free edition

- The model you are connecting to is NOT Oracle or MS SQL Server based

- The model in the Cloud Server you are connecting to is not configured with the 'Enable extended OSLC interface for WebEA' option

- The model you are connecting to has not been configured/adjusted for Visibility Levels (ie the update script has not been run against the database)

- The currently logged on application user has not been granted access to the 'Visibility Level Admin' Enterprise Architect security permission.

Cloud Server Troubleshooting

Browser Test

For a simple check that the Cloud Service is operating on the specified ports defined in the configuration file, you can enter this address into a web browser:

<protocol>://<machineName>:<port>

For example, on the Cloud Server:

http://localhost:804/

Or

https://sparxcloud.com

If the port is enabled the following message is returned:

Sparx Systems Cloud Server

Congratulations, your server is now ready to host your models. Connect through the

configuration client to add or remove models.

If this message is not displayed, confirm the server and port name, check for other services listening on the specified port and check the server log file for error messages.

Errors shown in Enterprise Architect System Output

Error Message	Meaning
HTTP Status 401 Access Denied	Indicates a failure to authenticate with the server. With the built-in web server, this error will occur if an invalid username or password was provided, or no username or password were provided.
HTTP Status 403	Indicates an attempt to access a model that is marked on the server as 'Require authenticated and secure connection'. This might indicate that you are connecting using an *http* protocol instead of *https*, or that the port you are using is not providing authentication to the model you are connecting to. Check that you are using https. Check with the system administrator that the security authentication option is pointing to a model with security enabled.
HTTP Status 500 Unable to connect to service	The connection is not being accepted. • Check the server name • Check the firewall settings on the client and server • Check the ports the server is listening on
The database manager for this database was shut down	Enterprise Architect established a connection to a Cloud Service, and requested to connect to a known repository, but the repository is currently disabled. Ask your system administrator to check the 'Enabled' option for the database in question.
There is no database	Enterprise Architect established a connection to a Cloud Service, but the model

manager configured for the requested repository	name requested does not match any defined database on the server.
Unable to connect to Database	Enterprise Architect established a connection to a Cloud Service and requested to connect to a known repository; however, when the server attempted to open a connection to the database using the supplied details, it failed. Note: Whenever this error occurs, a Warning-level message that contains more detailed information will be written to the Cloud Server logs. If the repository is: • Using ODBC, ask your System Administrator to check that an ODBC connection is defined as a System DSN (not a User DSN) on the server, that all connection parameters are correct, and that the database username and password have been saved into the ODBC DSN or the connection string • Oracle, ask your System Administrator to verify that the user name and password were saved with the connection string and that all other options in the connection are correct • SQL Server connecting via OLE DB and Windows authentication, ask your System Administrator to verify that the user name the Cloud Service is running under has been granted permission to the SQL Server database (at a minimum it will need the roles db_datareader and db_datawriter) and that all other options in the connection are correct • SQL Server connecting via OLE DB and SQLServer authentication, ask your System Administrator to verify that the defined user name has been granted permission to the SQL Server database (at a minimum it will need the roles db_datareader and db_datawriter), that the option to save the password with the connection string was enabled, and that all other options in the connection are correct

Reading Server Logs

There are two key sources of troubleshooting information to determine the cause of connection errors. The first of these is the server logs.

When troubleshooting, it is recommended that the Cloud Services LOG_LEVEL property be set to the highest level, SYSTEM.

In the [INFO]: [AUDIT] section, each line has three parameters at the end. These are, from **right** to **left**:

- Acquisitions - the total number of times that the server received a request for this database since the last audit
- Max Active - the highest number of concurrent connections created
- Free - the current number of those connections that are not being used

Log Message	Explanation
Started database managers	When a Cloud Server is initially started, it writes the list of available models to the log file. For each model you have added, you should see a line such as: *[SYSTEM]: SUCCESS Added database manager msdasql.1:anonymous@mymodel* If the log file is missing any lines, or if the message doesn't read 'SUCCESS' this might indicate a problem with the model connection.
Open port list	When a Cloud Server is initially started, it writes to the log file the status of all ports on which it tries to listen. For each port, including the management port and any http/https ports, you should see a line such as: *[SYSTEM] SUCCESS Bound and listening on port 803.* If the log file is missing any lines, or if the message doesn't read 'SUCCESS' this might indicate a problem with the server port. A common error here is: *Unable to create secure server on port 443.* Possible causes of this are: - Keyfile not found at C:\Program Files (x86)\Sparx Systems\Cloud Services\Service\server.pem - The server.pem file is an invalid certificate (that is, a private key is missing) - Certificate Authority file not found or invalid at C:\Program Files (x86)\Sparx Systems\Cloud Services\Service\cacert.pem
Unable to connect to	During operation, the server will write to the log file any failed attempts to open a model. This will generally not occur until a user attempts to connect to the

database	database. Errors that can occur here are:
	REQUEST_CONNECT FAIL. Error (5): Unable to connect to database
	This means that the server attempted to open a connect but it failed. If you see this message, review the list of potential reasons outlined in the *Errors shown in Enterprise Architect System Output* section.

Cloud Service using IIS (Optional)

For most users, the built-in web server provided with the Cloud Service provides all the functionality they need. It is also the easiest method to set up and use.

However, the Cloud Service also provides the option of integrating with Internet Information Services (IIS). This is *only* recommended for users for whom the built-in authentication methods are insufficient, and who want to use Windows Active Directory or certificate-based authentication.

To configure IIS to host the Cloud Service, you must first set the SparxCloudLink.sseap file to refer to the Cloud Service, and then in Windows open the Internet Information Services (IIS) Manager and configure it to send requests to the appropriate module. Depending on your preferences you can choose either an HTTP Module or an ISAPI module.

Prior to setting up HTTP or ISAPI, these settings must be made in IIS:

- Application Pool settings
- Feature Settings

For a secure HTTPS setup (optional):

- Set up a Certificate
- Set up HTTPS

Configuration Settings

Before setting up your HTTP module or ISAPI module, you must first check that the sscloudservices.config file for the service is configured as required, and note the port used so that you can specify it when you go on to set the SparxCloudLink.sseap file to refer to the Cloud Service.

The directory path containing the *ISAPI dll* also contains the SparxCloudLink.sseap configuration file.

The path should resemble this example:

   "C:\Program Files (x86)\Sparx Systems\Cloud Services\ISAPI"

Edit this file in a text editor, running as an Administrator.

The default contents are:

   [cloud]

   server=localhost

   port=803

server:

The address to look up the machine running the Cloud Service. In most circumstances it is best to run the HTTP module and the Cloud Service on the same machine. In this case, the default value of localhost can be used. If the Cloud Service is running on a different machine, use the IP address or server name where the service is running.

port:

The port the Cloud Service is listening on for admin requests. By default this takes the value of 803, but this should be cross-referenced against your service configuration in the sscloudservices.config file.

Optional settings

These points are optional. For testing purposes you might prefer to leave these changes until any issues with IIS are resolved.

1. Clear the Sparx Services configuration file of reference to ports other than the admin port.
   In the SSCloudServices.config file, remove all the references to ports other than the administration port (default 803). In other words, remove the bracketed entries (…) from the config file.

2. Save the SSCloudService.Config file.

3. Restart the Service.

Application Pool Settings

HTTP or ISAPI will require an Application pool that is 32-bit and not managed code. This illustration shows the settings for 32-bit applications in the IIS Application Pools Advanced Settings view.
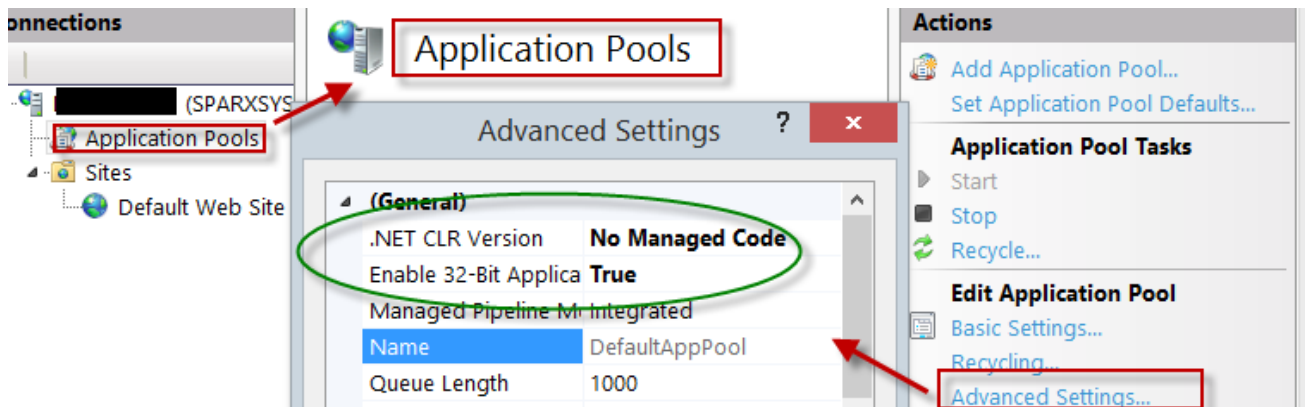
Feature Permissions

This illustration shows the access path and the 'Default Web Site | Handler' mappings to permit Script Execution.
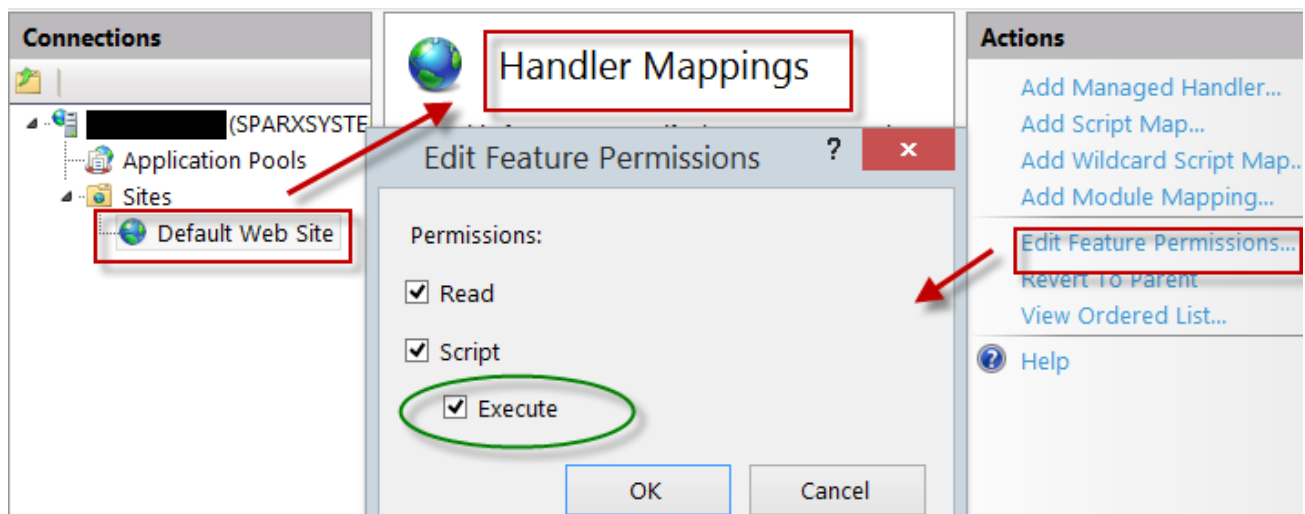
Setting up a Certificate

In order to run the HTTPS service you must set up a security certificate in IIS.

In the IIS Manager:

1. Under 'Connections', select the root connection (machine name).

2. Double-click on the 'Server Certificates' icon.

3. Click on 'Create Self Signed Certificate'.

4. Specify a simple, user-friendly name for the certificate.
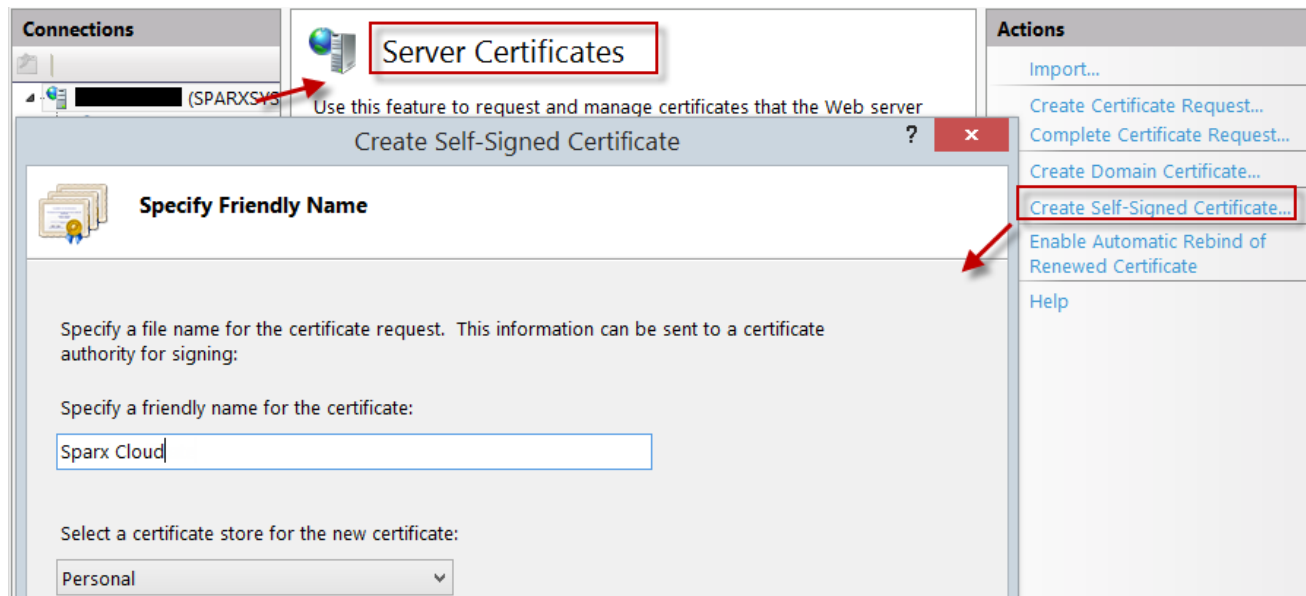
Setting up HTTPS

To set the bindings through which HTTPS will operate, you must include a port and a certificate in the site bindings.

In the IIS Manager:
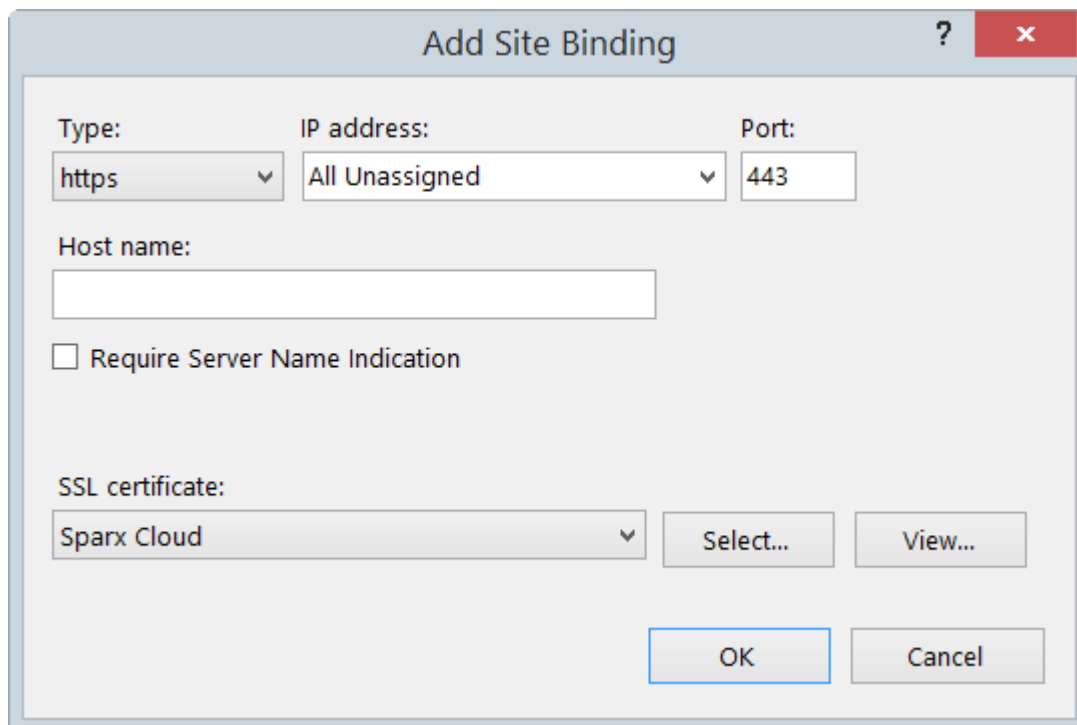
- Under 'Connections', select the 'Default Web Site' option
- Under 'Actions' click on the 'Bindings' option
- In the Site Bindings window click on 'Add'
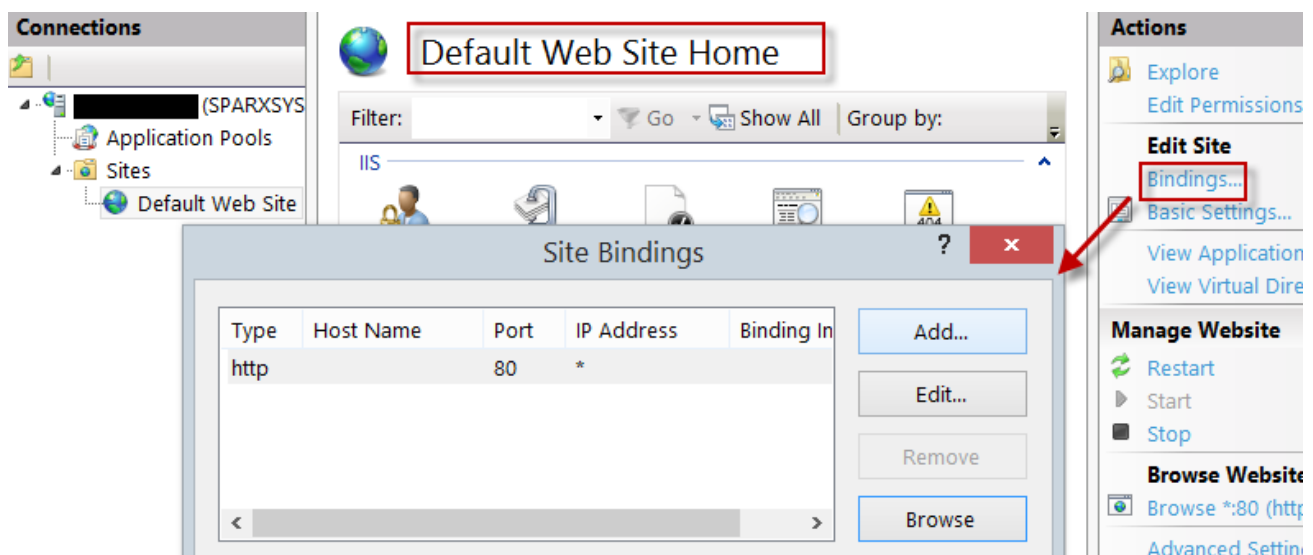
This will open the Add Site Binding window.

Set these parameters:

- Type: HTTPS
- Port: 443
- SSL Certificate: select the certificate created in *Setting up a Certificate*

HTTP Module

To set up the HTTP module in the Internet Information Services (IIS) Manager:

1. In the 'Connections' panel, select the machine properties (top of the tree).
2. Double-click on the 'Modules' icon in the middle panel.
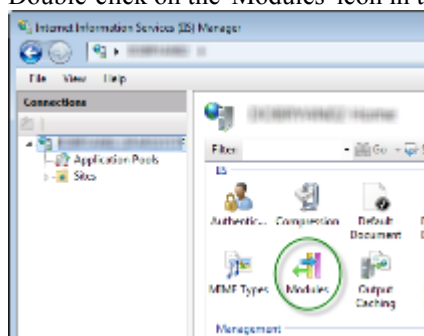
This returns the 'Modules' list and the 'Actions' view.

3. In the 'Actions' list, click on the 'Configure Native Modules…' option.
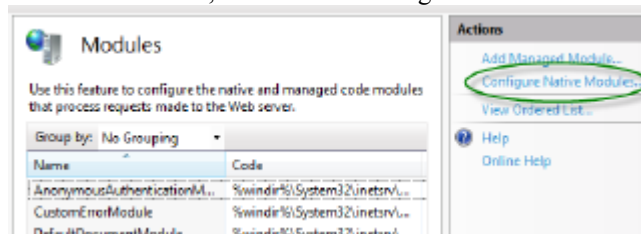
This opens the 'Configure Native Modules' view.

4. Click on the Register button to open the 'Register Native Module' dialog.
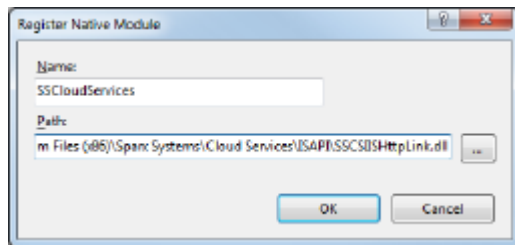
Type in the name and the path of the SSCSIISHttpLink.dll file.

5.   Click on the OK button. The 'SSCloudServices' checkbox will now be selected.
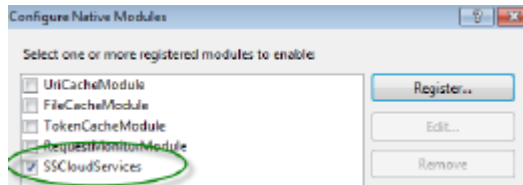
6.   Click on the OK button to close the 'Configure Native Modules' dialog.

7.   In the 'Connections' panel (see Step 1), click on your web site.

8.   Double-click on the 'Handler Mappings' in the middle pane.
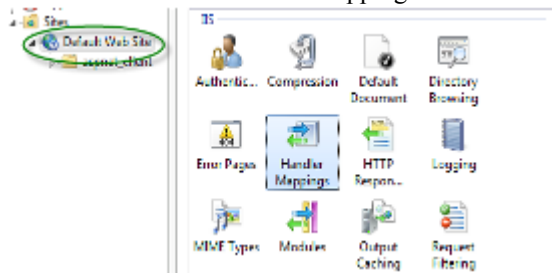
This opens the Handler Mappings view:
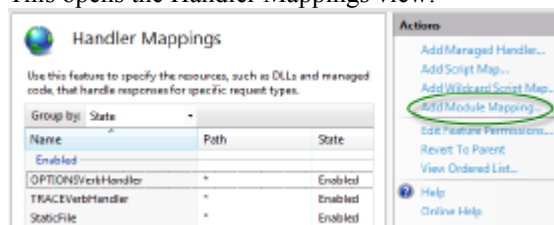
9.   In the 'Actions' list, click on the 'Add Module Mapping' option to open the 'Add Module Mapping' dialog.

10.   As illustrated, set the 'Request path' and 'Name'.

11.   In the 'Module' field, click on the drop-down arrow and select the module added in step 4.
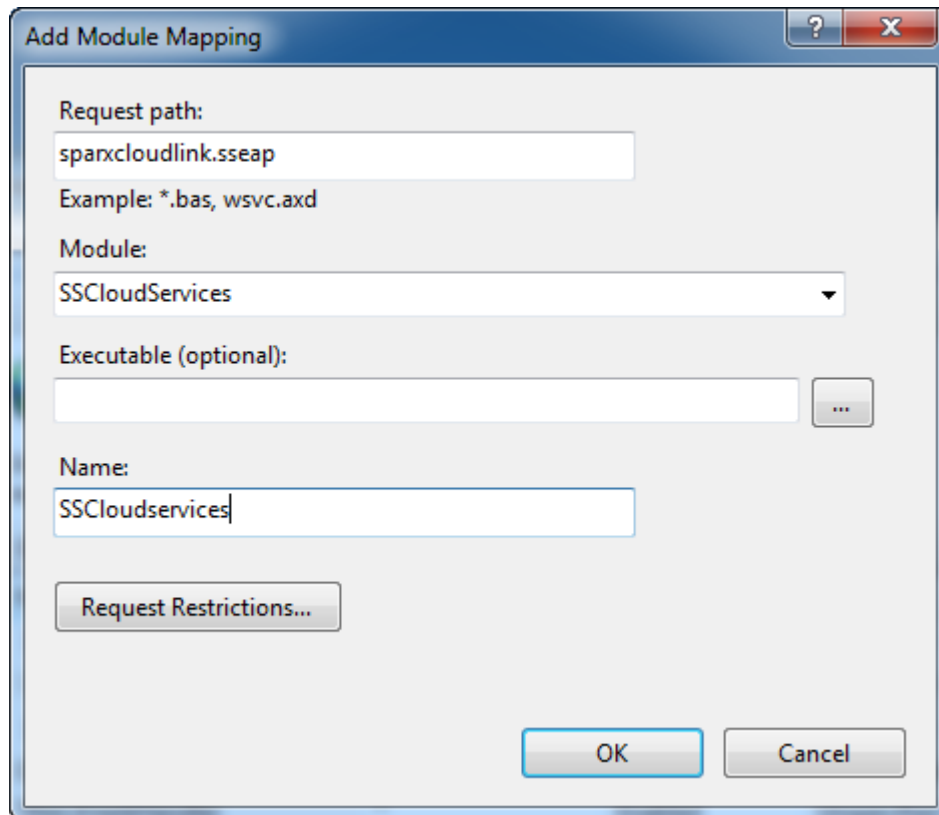
12. Click on the Request Restrictions button and, on the 'Request Restrictions' dialog, select the 'Access' tab. Select the 'Execute' radio button to enable Execute permission.
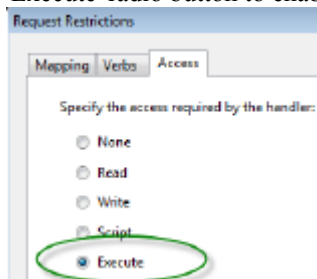
Note: The 'Mappings' tab should be left with the default settings; that is, 'Invoke handler only if request is mapped to' is not ticked.

13. Click on the OK button.

14. Close the 'Add Module Mapping' dialog by clicking on the OK button again.

To complete this HTTP module set up, see the *Configuration Settings* section at the start of this topic. You should then be able to connect to a model using Enterprise Architect via your IIS server using the HTTP module.

ISAPI Module

To configure an ISAPI module instead of the HTTP module:

1. In the 'Connections' panel, select the machine properties:
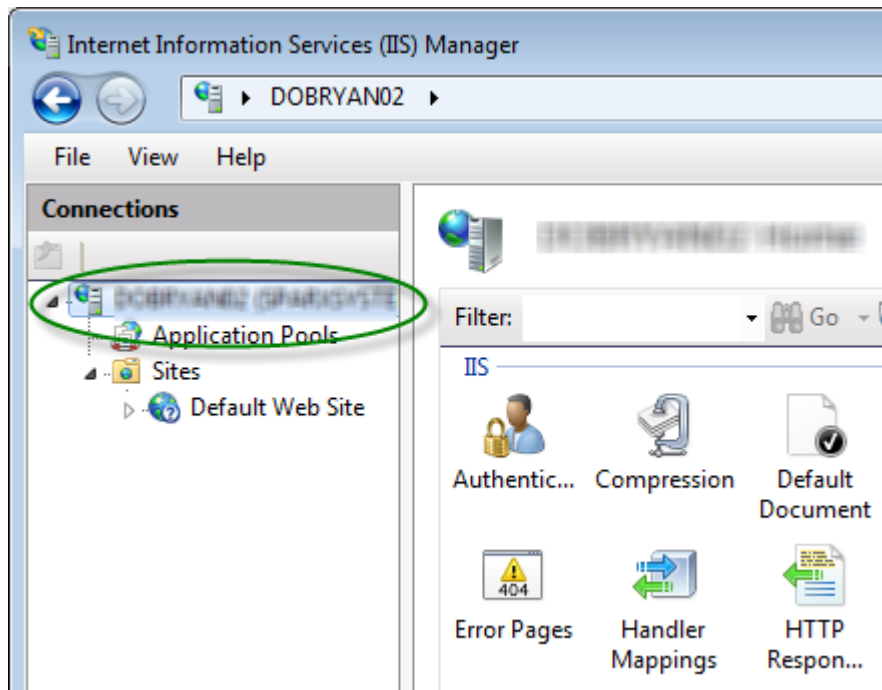
2. Double-click on the 'Handler Mappings' icon:

3. In the 'Actions' list, click on the 'Add Module Mapping' option.

4. In the 'Add Module Mapping' dialog, set 'Request path' to 'sparxCloudLink.sseap', 'Module' to 'IsapiModule', and 'Executable' to 'SSCSIISIsapiLink.dll', as shown.
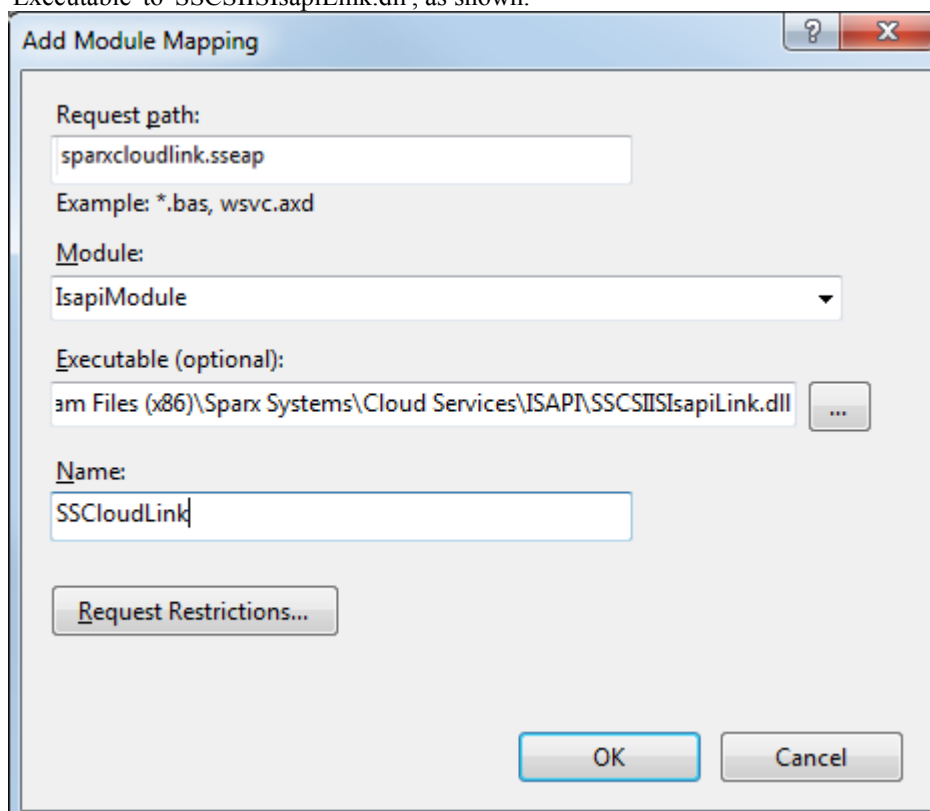
5. Click on the OK button to close the dialog. You should now be able to connect to a model using Enterprise Architect via your IIS server using ISAPI.

Additional Functionality Using the Cloud

The Cloud Services provide the facility of connecting to and using models over a secure http connection. They also make it possible to use three further facilities on or within your models:

- Open Services for Lifecycle Collaboration (OSLC)
- Reusable Asset Service (RAS)
- Scheduled Tasks

OSLC

OSLC is an initiative to provide easier integration between Requirement Management tools. It uses HTTP to list, add, modify and delete requirements. The service provider definition to direct any OSLC client to is:

<protocol>://<server>:<port>/<model_name>/oslc/sp/

For example, if you were connecting to a server running on your own machine using the default settings, the connection would be:

http://localhost:804/model/oslc/sp/

Reusable Asset Service (RAS)

The RAS portion of the Cloud Server helps you to define Packages that can be used in any model. When a Package is requested, Enterprise Architect and the Cloud Server will track cross-Package dependencies and make available everything required by that Package.

Scheduled Tasks

The Cloud Server includes optional support for running time-based updates to data. Currently, this is applied to updating a Time Series chart automatically to provide a dynamic view of how a model is changing over time. You can set different time scales for these updates, such as daily, weekly or monthly.

Notes

- This facility is available through the Corporate, Business and Software Engineering, Systems Engineering or Ultimate editions

OSLC Requirements Management

This text is derived from the OSLC Primer:

"Open Services for Lifecycle Collaboration (OSLC) is an open community creating specifications for integrating tools. These specifications allow conforming independent software and product lifecycle tools to integrate their data and workflows in support of end-to-end lifecycle processes."

"OSLC is based on the W3C Linked Data. One of the primary techniques for integrating tools using OSLC is Linking data via HTTP, which specifies creating, retrieving, updating and deleting (CRUD) lifecycle artifacts based on internet standards like HTTP and RDF using Linked Data model. Each artifact in the lifecycle, such as a requirement, is an HTTP resource that is manipulated using the standard methods of the HTTP specification (like GET, POST)."

Enterprise Architect acts as an OSLC Provider and supports the Requirements Management 2.0 specification of OSLC, which allows for creating, retrieving and querying the Requirements in a model accessed via a Cloud connection. With OSLC support, Requirements in an Enterprise Architect model can be identified and accessed using a unique URL that can be linked to resources in other lifecycle products and tools.

Enterprise Architect complies with these Requirements Management 2.0 base requirements:

- Resource Operations
- Service Provider Resource
- Partial Resource Representations
- Creation Factory
- Query Capability
- Query Syntax
- Error Responses
- RDF/XML Representations

Beta Implementation

This is a BETA release of OSLC Requirements Management and associated tools. As such, it is supplied without warranty of any kind and is subject to change without notice during the Beta period. Sparx Systems welcomes any feedback, issues, suggestions and comments on this implementation. As with all Beta software, please take due care when using OSLC Requirements Management in a production environment.

Notes

- This facility is available through the Corporate, Business and Software Engineering, Systems Engineering or Ultimate editions

Service Provider and Service Provider Resource

Enterprise Architect acts as an OSLC Provider that other OSLC-compliant tools (OSLC Consumer/Clients) access to link to its Resources. All OSLC Resources live in a Service Provider, which is a central organising concept of OSLC. In Enterprise Architect, each model that can be accessed via the Cloud connection is treated as an OSLC Service Provider. The Requirement elements in the model are the OSLC Resources.

The services offered by the Service Provider can be retrieved using the Service Provider Resource. A Service Provider Resource specifies the:

- URL to which you can POST representations to create new resources

- URL you can use to GET a list of existing resources

To retrieve the Service Provider Resource from an Enterprise Architect model connected via the Cloud, use the URL:

http://<server>/<model_name>/oslc/sp/

For example, the Service Provider Resource for a model called firebird_model, connected via the Cloud, would be accessed using the URL:

http://localhost:480/firebird_model/oslc/sp/

The retrieved resource might resemble this:

```xml
<?xml version="1.0" encoding="UTF-8"?>
- <rdf:RDF xmlns:dcterms="http://purl.org/dc/terms/" xmlns:oslc="http://open-services.net/ns/core#" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-
  ns#">
  - <oslc:ServiceProvider rdf:about="http://localhost:480/firebird_model/oslc/sp/">
      <dcterms:title>firebird:anonymous@firebird_model</dcterms:title>
    - <dcterms:publisher>
      - <oslc:Publisher>
          <dcterms:title>Sparx Systems</dcterms:title>
          <dcterms:identifier>http://www.sparxsystems.com.au/</dcterms:identifier>
        </oslc:Publisher>
      </dcterms:publisher>
    - <oslc:service>
      - <oslc:Service>
          <oslc:domain rdf:resource="http://open-services.net/ns/rm#"/>
        - <oslc:creationFactory>
          - <oslc:CreationFactory>
              <dcterms:title>Requirements Creation</dcterms:title>
              <oslc:creation rdf:resource="http://localhost:480/firebird_model/oslc/cf/"/>          Create new Requirements in the Model
              <oslc:resourceShape rdf:resource="http://localhost:480/firebird_model/oslc/rs/"/>      by sending POST request to this URL
            </oslc:CreationFactory>
          </oslc:creationFactory>
        - <oslc:queryCapability>
          - <oslc:QueryCapability>
              <dcterms:title>Requirements Query</dcterms:title>
              <oslc:queryBase rdf:resource="http://localhost:480/firebird_model/oslc/qc/"/>          Access the Requirements in the Model
              <oslc:resourceShape rdf:resource="http://localhost:480/firebird_model/oslc/rs/"/>      by sending GET request to this URL
            </oslc:QueryCapability>
          </oslc:queryCapability>
        </oslc:Service>
      </oslc:service>                                                                                Requirements representation is
    - <oslc:prefixDefinition>                                                                        available in this URL
      - <oslc:PrefixDefinition>
          <oslc:prefix>foaf</oslc:prefix>
          <oslc:prefixBase rdf:resource="http://xmlns.com/foaf/0.1/"/>
        </oslc:PrefixDefinition>
      </oslc:prefixDefinition>
    - <oslc:prefixDefinition>
      - <oslc:PrefixDefinition>
          <oslc:prefix>rdfs</oslc:prefix>
          <oslc:prefixBase rdf:resource="http://www.w3.org/2000/01/rdf-schema#"/>
        </oslc:PrefixDefinition>
      </oslc:prefixDefinition>
    - <oslc:prefixDefinition>
      - <oslc:PrefixDefinition>
          <oslc:prefix>dcterms</oslc:prefix>
          <oslc:prefixBase rdf:resource="http://purl.org/dc/terms/"/>
        </oslc:PrefixDefinition>
      </oslc:prefixDefinition>
    - <oslc:prefixDefinition>
      - <oslc:PrefixDefinition>
          <oslc:prefix>oslc_rm</oslc:prefix>
          <oslc:prefixBase rdf:resource="http://open-services.net/ns/rm#"/>
        </oslc:PrefixDefinition>
      </oslc:prefixDefinition>
    - <oslc:prefixDefinition>
      - <oslc:PrefixDefinition>
          <oslc:prefix>ss</oslc:prefix>
          <oslc:prefixBase rdf:resource="http://www.sparxsystems.com.au/"/>
        </oslc:PrefixDefinition>
      </oslc:prefixDefinition>
    - <oslc:prefixDefinition>
      - <oslc:PrefixDefinition>
          <oslc:prefix>oslc</oslc:prefix>
          <oslc:prefixBase rdf:resource="http://open-services.net/ns/core#"/>
        </oslc:PrefixDefinition>
      </oslc:prefixDefinition>
    - <oslc:prefixDefinition>
      - <oslc:PrefixDefinition>
          <oslc:prefix>rdf</oslc:prefix>
          <oslc:prefixBase rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#"/>
        </oslc:PrefixDefinition>
      </oslc:prefixDefinition>
    </oslc:ServiceProvider>
  </rdf:RDF>
```

Elements of the Service Provider Resource

Element	Description
oslc:Publisher	Specifies the OSLC Provider.
oslc:Service	Specifies the services offered by the OSLC Provider. Enterprise Architect supports these OSLC Requirement Management services: • Creation Factory - Used to create new Requirements in the model, by passing the Requirement's representation in RDF format using HTTP POST to the URL: http://<server>/<model_name>/oslc/cf/ • Query Capability - Used to list/query the Requirements in the model, by passing the query using HTTP GET to the URL:

	http://\<server>/\<model_name>/oslc/qc/
oslc:resourceShape	Specifies the Requirement's metadata; that is, its properties and constraints. These are available in the URL: http://\<server>/\<model_name>/oslc/rs/
oslc:prefixDefinition	Specifies the namespace prefixes and their namespace definitions.

Notes

- This facility is available through the Corporate, Business and Software Engineering, Systems Engineering or Ultimate editions

Resource Shape

'Resource Shape' specifies a Requirement's metadata; that is, the set of standard OSLC properties and custom Enterprise Architect properties. 'Resource Shape' can be accessed using the URL:

> http://<server>/<model_name>/oslc/rs/

The properties of both types specified in the 'Resource Shape' map to specific fields in the Requirement's 'Properties' dialog in Enterprise Architect. For each property, the 'Resource Shape' can also specify constraints.

OSLC Properties

(These are as defined in the Dublin Core Metadata Element Set.)

Property Name	Requirement element Properties dialog field
title	Short Description
description	Notes
subject	Key Words
creator	Author
created	Created
modified	Last Updated

Custom Enterprise Architect properties

Property Name	Requirement element Properties dialog field
alias	Alias
status	Status
difficulty	Difficulty
priority	Priority
type	Type
phase	Phase
version	Version

Constraints

Constraint	Meaning
name	The name of the property.
valueType	The type of value the property can have, such as string, dateTime or integer.
occurs	The cardinality of the property; that is, whether the property is optional or required.
maxSize	The maximum number of characters for a string valueType.
allowedValue	The list of values that can be assigned for the property.
readOnly	Determines whether a value for the property can be set by the client.

Notes

- This facility is available through the Corporate, Business and Software Engineering, Systems Engineering or Ultimate editions
- The property *identifier* refers to a Requirement's unique Enterprise Architect GUID
- The property *packageID* refers to the ID of the Package under which the Requirement exists in the model

Query Capability

Clients can query a model and retrieve the Requirements that match a specific criteria. In Enterprise Architect, the base URL for accessing the Query Capability is:

   http://<server>/<model_name>/oslc/qc/

A query string expressing the specific criteria should be added to the base URL and addressed to the model using an HTTP GET request. The response for this request will be in RDF/XML format. For example, all the Requirements in a model called firebird_model connected by the Cloud can be retrieved using the URL:

   http://localhost:480/firebird_model/oslc/qc/

The response for the request will resemble this:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:ss="http://www.sparxsystems.com.au/" xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:dcterms="http://purl.org/dc/terms/"
xmlns:oslc_rm="http://open-services.net/ns/rm#" xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#">
   <rdf:Description rdf:about="http://localhost:480/firebird_model/oslc/qc/">
      <rdfs:member>
         <oslc_rm:Requirement rdf:about="http://localhost:480/firebird_model/oslc/re/{58089E4F-E705-46ae-992E-0D876D7F5DF5}/">
            <rdf:type>http://open-services.net/ns/rm#Requirement</rdf:type>
            <dcterms:title>Requirement2</dcterms:title>
            <dcterms:description>Requirement2 Notes</dcterms:description>
            <dcterms:identifier>{58089E4F-E705-46ae-992E-0D876D7F5DF5}</dcterms:identifier>
            <dcterms:subject>risk, loading factor</dcterms:subject>
            <dcterms:creator>
               <foaf:Person>
                  <foaf:name>User1</foaf:name>
               </foaf:Person>
            </dcterms:creator>
            <dcterms:created>2014-01-06 11:29:23</dcterms:created>
            <dcterms:modified>2014-01-06 13:20:05</dcterms:modified>
            <ss:alias>Requirement2Alias</ss:alias>
            <ss:status>Proposed</ss:status>
            <ss:difficulty>Medium</ss:difficulty>
            <ss:priority>Medium</ss:priority>
            <ss:type>Display</ss:type>
            <ss:phase>1.0</ss:phase>
            <ss:version>1.0</ss:version>
            <ss:packageID>355</ss:packageID>
         </oslc_rm:Requirement>
      </rdfs:member>
      <rdfs:member>
         <oslc_rm:Requirement rdf:about="http://localhost:480/firebird_model/oslc/re/{7104C13D-841C-4068-B7EE-FB998C5BA4B7}/">
            <rdf:type>http://open-services.net/ns/rm#Requirement</rdf:type>
            <dcterms:title>Requirement1</dcterms:title>
            <dcterms:description>Requirement1 Notes</dcterms:description>
            <dcterms:identifier>{7104C13D-841C-4068-B7EE-FB998C5BA4B7}</dcterms:identifier>
            <dcterms:subject>risk, loading factor</dcterms:subject>
            <dcterms:creator>
               <foaf:Person>
                  <foaf:name>User1</foaf:name>
               </foaf:Person>
            </dcterms:creator>
            <dcterms:created>2014-01-06 11:01:58</dcterms:created>
            <dcterms:modified>2014-01-06 13:19:51</dcterms:modified>
            <ss:alias>Requirement1Alias</ss:alias>
            <ss:status>Proposed</ss:status>
            <ss:difficulty>Medium</ss:difficulty>
            <ss:priority>Medium</ss:priority>
            <ss:type>Functional</ss:type>
            <ss:phase>1.0</ss:phase>
            <ss:version>1.0</ss:version>
            <ss:packageID>355</ss:packageID>
         </oslc_rm:Requirement>
      </rdfs:member>
   </rdf:Description>
</rdf:RDF>
```

Elements of the response

In Enterprise Architect, the Query Capability supports these parameters:

- oslc.where
- oslc.select
- oslc.properties
- oslc.prefix

Element	Description

oslc_rm:Requirement	A Requirement and its properties (as defined in the Resource Shape).
	The attribute *rdf:about* on this element specifies the URL for accessing the Requirement. The format for the URL is: http://<server>/<model_name>/oslc/re/<requirement_GUID>/
Elements with namespace prefix dcterms	The standard properties from the Dublin Core Metadata Element Set.
Elements with namespace prefix ss	The custom Enterprise Architect properties.

Notes

- This facility is available through the Corporate, Business and Software Engineering, Systems Engineering or Ultimate editions

WHERE Parameter

The **oslc.where** Query parameter specifies the conditions that the resources must satisfy; it is similar to the WHERE clause of an SQL statement. This parameter must be appended to the base URI of the Query Capability in Enterprise Architect:

   http://<server>/<model_name>/oslc/qc/

The syntax for the **oslc.where** Query parameter (defined in BNF grammar and as specified in the *OSLC Query Syntax Specification*) is:

```
oslc_where     ::= "oslc.where=" compound_term
compound_term ::= simple_term (space? boolean_op space? simple_term)*
simple_term    ::= term | scoped_term
space          ::= " " /* a space character */
boolean_op     ::= "and"
term           ::= identifier_wc comparison_op value | identifier_wc space in_op space? in_val
scoped_term    ::= identifier_wc "{" compound_term "}"
identifier_wc ::= identifier | wildcard
identifier     ::= PrefixedName
PrefixedName  ::= /* see "SPARQL Query Language for RDF", http://www.w3.org/TR/rdf-sparql-query/#rPrefixedName */
wildcard       ::= "*"
comparison_op ::= "=" | "!=" | "<" | ">" | "<=" | ">="
in_op          ::= "in"
in_val         ::= "[" value ("," value)* "]"
value          ::= uri_ref_esc | literal_value
uri_ref_esc    ::= /* an angle bracket-delimited URI reference in which > and \ are \-escaped. */
literal_value ::= boolean | decimal | string_esc (LANGTAG | ("^^" PrefixedName))?
boolean        ::= "true" | "false"
decimal        ::= /* see "XML Schema Part 2: Datatypes Second Edition", http://www.w3.org/TR/xmlschema-2/ */
string_esc     ::= /* a string enclosed in double quotes, with certain characters escaped. See below. */
LANGTAG        ::= /* see "SPARQL Query Language for RDF", http://www.w3.org/TR/rdf-sparql-query/#rLANGTAG */
```

Examples

These example queries act on a model called *firebird_model* connected by the Cloud.

Example	Query
1	**Query**: http://localhost:480/firebird_model/oslc/qc/?oslc.where= dcterms:title="Requirement1" Retrieves all the OSLC properties of all Requirements having the title 'Requirement1'.
2	**Query**: http://localhost:480/firebird_model/oslc/qc/?oslc.where= dcterms:title in ["Requirement1","Requirement2"] Retrieves all the OSLC properties of all Requirements having the title 'Requirement1' or 'Requirement2'.
3	**Query**: http://localhost:480/firebird_model/oslc/qc/?oslc.where=dcterms:title="Requirement1" and dcterms:creator{foaf:name}="User1" Retrieves all the OSLC properties of all Requirements having the title 'Requirement1', created by 'User1'.
4	**Query**: http://localhost:480/firebird_model/oslc/qc/?oslc.where=dcterms:creator{foaf:name}="User1" and dcterms:created<"2014-01-07"

	Retrieves all the OSLC properties of all Requirements created by 'User1' prior to 7th January 2014.
5	**Query**: http://localhost:480/firebird_model/oslc/qc/?oslc.where= name="Requirement1" Returns an error response, as 'name' is not a valid property in Enterprise Architect.
6	**Query**: http://localhost:480/firebird_model/oslc/qc/?oslc.where=dcterms:created<"01-07-2 014" Returns an error response as the value of the property created is not in a valid format.

Notes

- Dates must be specified in the format YYYY-MM-DD
- This facility is available through the Corporate, Business and Software Engineering, Systems Engineering or Ultimate editions

SELECT Parameter

The **oslc.select** Query parameter specifies the Requirement properties to be retrieved; it is similar to the SELECT clause of an SQL statement. This parameter must be appended to the base URI of the Query Capability in Enterprise Architect:

http://<server>/<model_name>/oslc/qc/

The syntax for the **oslc.select** Query parameter (defined in BNF grammar and as specified in the OSLC Query Syntax Specification) is:

```
oslc_select ::= "oslc.select=" properties
properties  ::= property ("," property)*
property    ::= identifier | wildcard | nested_prop
nested_prop ::= (identifier | wildcard) "{" properties "}"
```

Examples

These example queries act on a model called *firebird_model* connected by the Cloud.

Example	Query
1	**Query:** http://localhost:480/firebird_model/oslc/qc/?oslc.select= dcterms:title Retrieves the OSLC property 'title' of all the Requirements in the model.
2	**Query:** http://localhost:480/firebird_model/oslc/qc/?oslc.select=dcterms:title,dcterms:created,dcterms:creator{foaf:name},ss:version Retrieves the OSLC properties 'title', 'creator' and 'version' of all the Requirements in the model.
3	**Query:** http://localhost:480/firebird_model/oslc/qc/?oslc.select=* Retrieves all the OSLC properties of all the Requirements in the model.
4	**Query:** http://localhost:480/firebird_model/oslc/qc/?oslc.select=dcterms:name Returns an error response, as 'name' is not a valid property in Enterprise Architect.

Notes

- This facility is available through the Corporate, Business and Software Engineering, Systems Engineering and Ultimate editions

Combine WHERE and SELECT Parameters

The **oslc.where** and **oslc.select** Query parameters can be combined in the base URI of the Query Capability to retrieve the required properties of all those Requirements that satisfy the specified condition. This is similar to using WHERE and SELECT clauses together in a SQL statement.

Examples

These example queries act on a model called *firebird_model* connected by the Cloud.

Example	Query
1	**Query:** http://localhost:480/firebird_model/oslc/qc/?oslc.where= dcterms:title="Requirement1" & oslc.select=dcterms:title Retrieves the OSLC property title of all Requirements that have the title 'Requirement1'.
2	**Query:** http://localhost:480/firebird_model/oslc/qc/?oslc.where= dcterms:title in ["Requirement1","Requirement2"] & oslc.select=dcterms:title,dcterms:created,dcterms:creator{foaf:name} Retrieves the OSLC properties 'title', 'created' and 'creator' of all Requirements that have the title 'Requirement1' or 'Requirement2'.
3	**Query:** http://localhost:480/firebird_model/oslc/qc/?oslc.where=dcterms:title="Requirement1" and dcterms:creator{foaf:name}="User1" & oslc.select=* Retrieves all the OSLC properties of all Requirements that have the title 'Requirement1', created by 'User1'.
4	**Query:** http://localhost:480/firebird_model/oslc/qc/?oslc.where=dcterms:created<"01-07-2014" oslc.select=dcterms:title Returns an error response, as the value of the property 'created' is not in the valid format.

Notes

- This facility is available through the Corporate, Business and Software Engineering, Systems Engineering and Ultimate editions
- Dates must be specified in the format YYYY-MM-DD

PROPERTIES Parameter

Enterprise Architect supports a technique called **Selective Properties**, through which clients can retrieve selected OSLC properties of a Requirement. This technique accepts a partial representation of the OSLC properties; that is, all properties or only some of them. The base URI for accessing Selective Properties of a Requirement in Enterprise Architect is:

http://<server>/<model_name>/oslc/re/<requirement_GUID>/

The syntax for the **oslc.properties** Query parameter (defined in BNF grammar and as specified in the OSLC Core Specification) is:

```
oslc_properties ::= "oslc.properties=" properties
properties      ::= property ("," property)*
property        ::= identifier | wildcard | nested_prop
nested_prop     ::= (identifier | wildcard) "{" properties "}"
wildcard        ::= "*"
identifier      ::= PrefixedName
PrefixedName    ::= /* see "SPARQL Query Lanaguage for RDF", http://www.w3.org/TR/rdf-sparql-query/#rPrefixedName */
```

Examples

These example queries act on a Requirement with the GUID {7104C13D-841C-4068-B7EE-FB998C5BA4B7} in a model called *firebird_model* connected by the Cloud.

Example	Query
1	**Query:** http://localhost:480/firebird_model/oslc/re/{7104C13D-841C-4068-B7EE-FB998C5BA4B7}/?oslc.properties=* Retrieves all the OSLC properties of the specified Requirement.
2	**Query:** http://localhost:480/firebird_model/oslc/re/{7104C13D-841C-4068-B7EE-FB998C5BA4B7}/?oslc.properties=dcterms:title,dcterms:creator{foaf:name} Retrieves the OSLC properties 'title' and 'creator' of the specified Requirement.
3	**Query:** http://localhost:480/firebird_model/oslc/re/{7104C13D-841C-4068-B7EE-FB998C5BA4B7}/?oslc.properties=dcterms:title,dcterms:creator{} Returns an error response as the property 'creator' is incomplete; it should be: dcterms:creator{foaf:name}

Notes

- This facility is available through the Corporate, Business and Software Engineering, Systems Engineering and Ultimate editions

PREFIX Parameter

Clients can use the **oslc.prefix** parameter to specify URI prefixes as used in OSLC property names. This parameter is appended to the base URI of the Selective Properties of a requirement:

http://<server>/<model_name>/oslc/re/<requirement_GUID>/

The syntax for the **oslc.prefix** parameter (defined in BNF grammar and as specified in the OSLC Core Specification) is:

```
oslc_prefix ::= "oslc.prefix=" prefix_defs
prefix_defs ::= prefix_def ("," prefix_def)*
prefix_def  ::= prefix "=" uri_ref_esc
prefix      ::= PN_PREFIX
PN_PREFIX   ::= /* see "SPARQL Query Lanaguage for RDF", http://www.w3.org/TR/rdf-sparql-query/#rPN_PREFIX */
uri_ref_esc ::= /* an angle bracket-delimited URI reference in which > and \ are \-escaped. */
```

Example Queries

These example queries act on a Requirement with the GUID {7104C13D-841C-4068-B7EE-FB998C5BA4B7} in a model called *firebird_model* connected by the Cloud.

Example	Query
1	**Query:** http://localhost:480/firebird_model/oslc/re/{7104C13D-841C-4068-B7EE-FB998C5BA4B7}/?oslc.prefix=otrm=<http://purl.org/dc/terms/\>& oslc.properties=otrm:title Retrieves the OSLC property title of the specified Requirement. Note that the prefix for the namespace http://purl.org/dc/terms/ has been specified as 'otrm' in the query.
2	**Query:** http://localhost:480/firebird_model/oslc/re/{7104C13D-841C-4068-B7EE-FB998C5BA4B7}/?oslc.prefix=otrm=<http://purl.org/dc/terms/\>,spx=<http://www.sparxsystems.com.au/\>& oslc.properties=otrm:title,spx:alias Retrieves the OSLC properties title and alias of the specified Requirement. Note that the prefix for the namespace http://purl.org/dc/terms/ has been specified as otrm and that of http://www.sparxsystems.com.au/ has been specified as spx in the query.
3	**Query:** http://localhost:480/firebird_model/oslc/re/{7104C13D-841C-4068-B7EE-FB998C5BA4B7}/?oslc.prefix=otrm=<http://purl.org/dc/terms/\>,spx=<http://www.sparxsystems.com.au/\>& oslc.properties=otrm:title,sx:alias Returns an error response as the namespace prefix sx on the property alias is undefined.

Notes

- This facility is available through the Corporate, Business and Software Engineering, Systems Engineering and Ultimate editions

Creation Factory

Enterprise Architect supports the **Creation Factory OSLC service**, through which clients can create new Requirements via HTTP POST. To create a new Requirement, the client POSTs a representation of the Requirement in RDF format to the Creation Factory URL. If the POST is successful, the HTTP location header of the response will contain the URL of the created Requirement. An unsuccessful POST will generate an error response.

The Creation Factory URL has the format:

http://<server>/<model_name>/oslc/cf/

Examples

These are some example representations of a Requirement in RDF format.

Example	RDF Representation
1	<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:oslc_rm="http://open-services.net/ns/rm#" xmlns:dcterms="http://purl.org/dc/terms/" xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:ss="http://www.sparxsystems.com.au/"> <oslc_rm:requirement> <dcterms:title>Requirement3</dcterms:title> <dcterms:description>Requirement Notes </dcterms:description> <dcterms:creator> <foaf:Person> <foaf:name>User1</foaf:name> </foaf:Person> </dcterms:creator> <ss:type>Functional</ss:type> <ss:packageID>355</ss:packageID> </oslc_rm:requirement> </rdf:RDF> **Result Description** Creates a new Requirement with the specified *name*, *notes*, *author* and *stereotype*.
2	<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:oslc_rm="http://open-services.net/ns/rm#" xmlns:dct="http://purl.org/dc/terms/" xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:ss="http://www.sparxsystems.com.au/"> <oslc_rm:requirement> <dct:title>Requirement4</dct:title> <ss:packageID>355</ss:packageID> <ss:difficulty>High</ss:difficulty>

	`<ss:priority>High</ss:priority>`
	`</oslc_rm:requirement>`
	`</rdf:RDF>`
	
	Result Description
	Creates a new Requirement with the specified *name*, *difficulty* and *priority*.

3	`<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"`
	`xmlns:oslc_rm="http://open-services.net/ns/rm#"`
	`xmlns:dct="http://purl.org/dc/terms/"`
	`xmlns:foaf="http://xmlns.com/foaf/0.1/"`
	`xmlns:ss="http://www.sparxsystems.com.au/">`
	`<oslc_rm:requirement>`
	`<dct:title>Requirement4</dct:title>`
	`<ss:packageID>355</ss:packageID>`
	`<ss:difficulty/>`
	`</oslc_rm:requirement>`
	`</rdf:RDF>`
	
	Result Description
	Produces an error response, as the property 'difficult' has an invalid (empty) value.

Notes

- This facility is available through the Corporate, Business and Software Engineering, Systems Engineering or Ultimate editions
- The OSLC properties 'title' and 'packageID' of the Requirement are mandatory and must be supplied in the RDF representation
- The value of the OSLC property 'packageID' should already exist in the model, so before creating a Requirement using Creation Factory, use the Query Capability URL to determine the existing values of 'packageID'
- The OSLC properties 'identifier', 'created' and 'modified' of the Requirement are read-only and cannot be set using the Creation Factory service

OSLC Architecture Management v2.0

This text is derived from the OSLC Primer:

"Open Services for Lifecycle Collaboration (OSLC) is an open community creating specifications for integrating tools. These specifications allow conforming independent software and product lifecycle tools to integrate their data and workflows in support of end-to-end lifecycle processes."

OSLC is based on the W3C Linked Data. One of the primary techniques for integrating tools using OSLC is Linking data via HTTP, which specifies creating, retrieving, updating and deleting (CRUD) lifecycle artifacts based on internet standards such as HTTP and RDF, using the Linked Data model. Each artifact in the lifecycle, such as a requirement, is an HTTP resource that is manipulated using the standard methods of the HTTP specification (for example, GET and POST).

Enterprise Architect acts as an OSLC Provider and supports the Architecture Management 2.0 specification of OSLC, which allows for creating, retrieving and querying Enterprise Architect Resources (Packages, elements, diagrams and connectors) in a model accessed via Pro Cloud Server. With OSLC support, resources in an Enterprise Architect model can be identified and accessed using a unique URL that can be linked to resources in other lifecycle products and tools.

Enterprise Architect complies with these Architecture Management 2.0 base requirements, as specified in the Architecture Management 2.0 specification:

- Service Provider Resource
- Absolute URIs
- RDF/XML Representations
- HTTP Rest Services
- Resource Creation Factory
- Resource Query Capability
- Link Type Query Capability
- Partial Resource Update
- Selective Properties
- Error Responses

Beta Implementation

This is a BETA release of OSLC Architecture Management and associated tools. As such, it is supplied without warranty of any kind and is subject to change without notice during the Beta period. Sparx Systems welcomes any feedback, issues, suggestions and comments on this implementation. As with all Beta software, please take due care when using OSLC Architecture Management in a production environment.

Notes

- This facility is available through the Enterprise Architect Pro Cloud Server, under Enterprise Architect Release 13.5 or higher

Resource Information Accessed Via OSLC Architecture Management

Using the OSLC Architecture Management 2.0 implementation in Enterprise Architect, you can retrieve a wealth of information on Packages, elements and diagrams as model resources.

Resource Information

This information on Resources can be retrieved:

Resource	Information
Package	• Properties • Requirements • Constraints • Scenarios • Discussions and Replies • Tagged Values • Tests • Resource Allocation • Changes • Defects • Issues • Tasks • Events • Decisions • Efforts • Risks • Metrics • Connectors
Element	Includes all the information that can be retrieved for a Package (as listed in the Package row), plus: • Run State • Attributes and Attribute Tagged Values • Operations and Operation Tagged Values • Operation Parameters
Diagram	• Properties • Diagram Image (Base64 encoded PNG file as a BLOB)

Managing Resources

With the OSLC Architecture Management 2.0 implementation in Enterprise Architect, it is possible to create and update

these Resources and Resource Features:

- Package
    - Discussions and Replies
    - Tagged Values
    - Tests
    - Resource Allocations
    - Maintenance Items (Changes, Defects, Issues, Tasks, Events, Decisions)
    - Project Management Items (Efforts, Risks, Metrics)

- Element
    - Attributes
    - Operations
    - Operation Parameters
    - Discussions and Replies
    - Tagged Values
    - Tests
    - Resource Allocations
    - Maintenance Items (Changes, Defects, Issues, Tasks, Events, Decisions)
    - Project Management Items (Efforts, Risks, Metrics)

- Connectors

Service Provider and Service Provider Resource

Enterprise Architect acts as an OSLC Provider that other OSLC-compliant tools (OSLC Consumer/Clients) access to link to its Resources. All OSLC Resources live in a Service Provider, which is a central organising concept of OSLC. In Enterprise Architect, each model that can be accessed via the Pro Cloud Server is treated as an OSLC Service Provider. The Packages, elements, diagrams and connectors in the model are the OSLC Resources.

When an OSLC Consumer tries to consume data from OSLC Provider, it has to know all the services that the OSLC Provider offers. This can be achieved via a Service Provider Resource; that is, the services offered by the Service Provider can be retrieved using the Service Provider Resource. A Service Provider Resource specifies the:

- URL to which you can POST representations to create new Resources

- URL you can use to GET a list of existing Resources

- URL from which you can get the metadata of:
   - XML that has to be POSTed to create new Resources
   - XML that represents the retrieved existing Resources

To retrieve the Service Provider Resource for an Enterprise Architect model connected via the Pro Cloud Server, use the URL:

   <protocol>://<server>/<model_name>/oslc/am/sp/

For example, the Service Provider Resource for a model called 'firebird_model', connected via the Pro Cloud Server, can be accessed using the URL:

   http://localhost:480/firebird_model/oslc/am/sp/

The retrieved resource will typically resemble this example:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:oslc="http://open-services.net/ns/core#"
    xmlns:dcterms="http://purl.org/dc/terms/"
    xmlns:ss="http://www.sparxsystems.com.au/oslc_ss#">
    <oslc:ServiceProvider rdf:about="http://mybuildvm:2480/firebird_model/oslc/sm/su/">
        <dcterms:title>firebird:anonymous@firebird_model</dcterms:title>
        <dcterms:publisher>
            <oslc:Publisher>
                <dcterms:title>Sparx Systems</dcterms:title>
                <dcterms:identifier>http://www.sparxsystems.com.au/</dcterms:identifier>
                <!--Enterprise Architect's OSLC Architecture Management 2.0 Implementation Version-->
                <ss:version>1.01.1390</ss:version>
                <!--Specifies whether Pro Cloud Server Connection to Enterprise Architect Model is Read-Only-->
                <ss:readonlyconnection>false</ss:readonlyconnection>
                <!--Specifies whether User Security is enable in the Enterprise Architect Model-->
                <ss:securityenabledmodel>true</ss:securityenabledmodel>
                <!--Specifies the Pro Cloud Server Edition-->
                <ss:prolicense>Enterprise</ss:prolicense>
                <!--Specifies the Pro Cloud Server Expiry Date -->
                <ss:prolicenseexpiry>2018-05-01</ss:prolicenseexpiry>
            </oslc:Publisher>
        </dcterms:publisher>
        <!--Specifies the OSLC Architecture Management 2.0 Services supported by Enterprise Architect-->
        <oslc:Service>
            <oslc:domain rdf:resource="http://open-services.net/ns/am#"/>
            <oslc:creationFactory>
                <oslc:CreationFactory>
                    <dcterms:title>Creation Factory for Enterprise Architect Package/Element</dcterms:title>
                    <!--Create new Resource (Package/Element) in the Enterprise Architect Model by POSTing Resource XML to this URL-->
                    <oslc:creation rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/cf/resource/"/>
                    <!--URL describing metadata of the Resource XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/resource/"/>
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/cfprofileresource/"/>
                    <oslc:usage rdf:resource="http://open-services.net/ns/am#Resource"/>
                    <oslc:resourceType rdf:resource="http://open-services.net/ns/am#default"/>
                </oslc:CreationFactory>
            </oslc:creationFactory>
            <oslc:creationFactory>
                <oslc:CreationFactory>
                    <dcterms:title>Creation Factory for Enterprise Architect Link</dcterms:title>
                    <!--Create new Connector in the Enterprise Architect Model by POSTing Connector XML to this URL-->
                    <oslc:creation rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/cf/resourcelink/"/>
                    <!--URL describing metadata of the Connector XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/cflinktype/"/>
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/cfprofilelinktype/"/>
                </oslc:CreationFactory>
            </oslc:creationFactory>
            <oslc:creationFactory>
                <oslc:CreationFactory>
                    <dcterms:title>Creation Factory for Enterprise Architect Element Attribute</dcterms:title>
                    <!--Create new Element Attribute in the Enterprise Architect Model by POSTing Attribute XML to this URL-->
                    <oslc:creation rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/cf/attribute/"/>
                    <!--URL describing metadata of the Attribute XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/cfattribute/"/>
                    <oslc:resourceType rdf:resource="http://www.sparxsystems.com.au/oslc_am#attribute"/>
                </oslc:CreationFactory>
            </oslc:creationFactory>
            <oslc:creationFactory>
                <oslc:CreationFactory>
                    <dcterms:title>Creation Factory for Enterprise Architect Element Operation</dcterms:title>
                    <!--Create new Element Operation in the Enterprise Architect Model by POSTing Operation XML to this URL-->
                    <oslc:creation rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/cf/operation/"/>
                    <!--URL describing metadata of the Operation XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/cfoperation/"/>
                    <oslc:resourceType rdf:resource="http://www.sparxsystems.com.au/oslc_am#operation"/>
                </oslc:CreationFactory>
            </oslc:creationFactory>
            <oslc:creationFactory>
                <oslc:CreationFactory>
                    <dcterms:title>Creation Factory for Enterprise Architect Operation Parameter</dcterms:title>
                    <!--Create new Element Operation Parameter in the Enterprise Architect Model by POSTing Operation Parameter XML to this URL-->
                    <oslc:creation rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/cf/operationparameter/"/>
                    <!--URL describing metadata of the Operation Parameter XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/cfoperationparameter/"/>
                    <oslc:resourceType rdf:resource="http://www.sparxsystems.com.au/oslc_am#operationparameter"/>
                </oslc:CreationFactory>
            </oslc:creationFactory>
            <oslc:creationFactory>
                <oslc:CreationFactory>
                    <dcterms:title>Creation Factory for Enterprise Architect Discussion</dcterms:title>
                    <!--Create new Package/Element Discussion in the Enterprise Architect Model by POSTing Discussion XML to this URL-->
                    <oslc:creation rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/cf/discussion/"/>
                    <!--URL describing metadata of the Discussion XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/cfdiscussion/"/>
                    <oslc:resourceType rdf:resource="http://www.sparxsystems.com.au/oslc_am#discussion"/>
                </oslc:CreationFactory>
            </oslc:creationFactory>
            <oslc:creationFactory>
                <oslc:CreationFactory>
                    <dcterms:title>Creation Factory for Enterprise Architect Discussion Reply</dcterms:title>
                    <!--Create new Package/Element Discussion Reply in the Enterprise Architect Model by POSTing Discussion Reply XML to this URL-->
                    <oslc:creation rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/cf/reply/"/>
                    <!--URL describing metadata of the Discussion Reply XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/cfreply/"/>
                    <oslc:resourceType rdf:resource="http://www.sparxsystems.com.au/oslc_am#reply"/>
                </oslc:CreationFactory>
            </oslc:creationFactory>
            <oslc:creationFactory>
                <oslc:CreationFactory>
                    <dcterms:title>Creation Factory for Enterprise Architect Resource Allocation</dcterms:title>
                    <!--Create new Package/Element Resource Allocation in the Enterprise Architect Model by POSTing Resource Allocation XML to this URL-->
                    <oslc:creation rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/cf/resourceallocation/"/>
                    <!--URL describing metadata of the Resource Allocation XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/cfresourceallocation/"/>
                    <oslc:resourceType rdf:resource="http://www.sparxsystems.com.au/oslc_am#resourceallocation"/>
                </oslc:CreationFactory>
            </oslc:creationFactory>
            <oslc:creationFactory>
                <oslc:CreationFactory>
                    <dcterms:title>Creation Factory for Enterprise Architect Test</dcterms:title>
                    <!--Create new Package/Element Test in the Enterprise Architect Model by POSTing Test XML to this URL-->
                    <oslc:creation rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/cf/test/"/>
                    <!--URL describing metadata of the Test XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/cftest/"/>
                    <oslc:resourceType rdf:resource="http://www.sparxsystems.com.au/oslc_am#test"/>
                </oslc:CreationFactory>
            </oslc:creationFactory>
            <oslc:creationFactory>
                <oslc:CreationFactory>
                    <dcterms:title>Creation Factory for Enterprise Architect Tagged Value</dcterms:title>
                    <!--Create new Package/Element Tagged Value in the Enterprise Architect Model by POSTing Tagged Value XML to this URL-->
                    <oslc:creation rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/cf/taggedvalue/"/>
                    <!--URL describing metadata of the Tagged Value XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/cftaggedvalue/"/>
                    <oslc:resourceType rdf:resource="http://www.sparxsystems.com.au/oslc_am#taggedvalue"/>
                </oslc:CreationFactory>
            </oslc:creationFactory>
            <oslc:creationFactory>
                <oslc:CreationFactory>
                    <dcterms:title>Creation Factory for Enterprise Architect Maintenance Item ( Change/Defect/Issue/Task/Event/Decision )</dcterms:title>
                    <!--Create new Package/Element Maintenance Item in the Enterprise Architect Model by POSTing Maintenance Item XML to this URL-->
                    <oslc:creation rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/cf/maintenanceitem/"/>
                    <!--URL describing metadata of the Change XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/cfchange/"/>
                    <!--URL describing metadata of the Defect XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/cfdefect/"/>
                    <!--URL describing metadata of the Issue XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/cfissue/"/>
                    <!--URL describing metadata of the Task XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/cftask/"/>
                    <!--URL describing metadata of the Event XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/cfevent/"/>
                    <!--URL describing metadata of the Decision XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/cfdecision/"/>
                </oslc:CreationFactory>
            </oslc:creationFactory>
            <oslc:creationFactory>
                <oslc:CreationFactory>
                    <dcterms:title>Creation Factory for Enterprise Architect Project Management Item ( Effort/Risk/Metric )</dcterms:title>
                    <!--Create new Package/Element Project Management Item in the Enterprise Architect Model by POSTing Project Management Item XML to this URL-->
                    <oslc:creation rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/cf/projectmanagementitem/"/>
                    <!--URL describing metadata of the Effort XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/cfeffort/"/>
                    <!--URL describing metadata of the Risk XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/cfrisk/"/>
                    <!--URL describing metadata of the Metric XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/cfmetric/"/>
                </oslc:CreationFactory>
            </oslc:creationFactory>
            <oslc:queryCapability>
                <oslc:QueryCapability>
                    <dcterms:title>Resource Query</dcterms:title>
                    <!--Retrieve Resources (Packages/Elements/Diagrams) from the Enterprise Architect Model by sending GET request with query parameters to this URL-->
                    <oslc:queryBase rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/qc/"/>
                    <!--URL describing metadata of the retrieved Resource XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/resource/"/>
                    <oslc:resourceType rdf:resource="http://open-services.net/ns/am#Resource"/>
                </oslc:QueryCapability>
            </oslc:queryCapability>
            <oslc:queryCapability>
                <oslc:QueryCapability>
                    <dcterms:title>Link Type Query</dcterms:title>
                    <!--Retrieve all valid Enterprise Architect Connector Types using this URL-->
                    <oslc:queryBase rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/lt/"/>
                    <!--URL describing metadata of the retrieved Link Type Query XML-->
                    <oslc:resourceShape rdf:resource="http://mybuildvm:2480/firebird_model/oslc/am/rs/linktype/"/>
                    <oslc:resourceType rdf:resource="http://open-services.net/ns/am#LinkType"/>
                </oslc:QueryCapability>
            </oslc:queryCapability>
        </oslc:Service>
        <!--Namespaces and their Prefixes-->
        <oslc:prefixDefinition>
            <oslc:PrefixDefinition>
                <oslc:prefix>rdf</oslc:prefix>
                <oslc:prefixBase rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#"/>
            </oslc:PrefixDefinition>
        </oslc:prefixDefinition>
        <oslc:prefixDefinition>
            <oslc:PrefixDefinition>
                <oslc:prefix>rdfs</oslc:prefix>
                <oslc:prefixBase rdf:resource="http://www.w3.org/2000/01/rdf-schema#"/>
            </oslc:PrefixDefinition>
        </oslc:prefixDefinition>
        <oslc:prefixDefinition>
            <oslc:PrefixDefinition>
                <oslc:prefix>oslc</oslc:prefix>
                <oslc:prefixBase rdf:resource="http://open-services.net/ns/core#"/>
            </oslc:PrefixDefinition>
        </oslc:prefixDefinition>
        <oslc:prefixDefinition>
            <oslc:PrefixDefinition>
                <oslc:prefix>oslc_am</oslc:prefix>
                <oslc:prefixBase rdf:resource="http://open-services.net/ns/am#"/>
            </oslc:PrefixDefinition>
        </oslc:prefixDefinition>
        <oslc:prefixDefinition>
            <oslc:PrefixDefinition>
                <oslc:prefix>dcterms</oslc:prefix>
                <oslc:prefixBase rdf:resource="http://purl.org/dc/terms/"/>
            </oslc:PrefixDefinition>
        </oslc:prefixDefinition>
        <oslc:prefixDefinition>
            <oslc:PrefixDefinition>
                <oslc:prefix>foaf</oslc:prefix>
                <oslc:prefixBase rdf:resource="http://xmlns.com/foaf/0.1/"/>
            </oslc:PrefixDefinition>
        </oslc:prefixDefinition>
        <oslc:prefixDefinition>
            <oslc:PrefixDefinition>
                <oslc:prefix>ss</oslc:prefix>
                <oslc:prefixBase rdf:resource="http://www.sparxsystems.com.au/oslc_ss#"/>
            </oslc:PrefixDefinition>
        </oslc:prefixDefinition>
    </oslc:ServiceProvider>
</rdf:RDF>
```

Elements of the Service Provider Resource

Element	Description
oslc:Publisher	Specifies the OSLC Provider.

oslc:Service	Specifies the services offered by the OSLC Provider. Enterprise Architect supports these OSLC Architecture Management services:
	
	• Creation Factory for Packages and Elements - Create new Packages or elements in the model by passing their resource representation in RDF/XML format using HTTP POST to the URL:
	    `<protocol>://<server>/<model_name>/oslc/am/cf/resource/`
	
	• Creation Factory for Package and Element Connectors - Create new Package or element connectors in the model by passing the resource representation in RDF/XML format using HTTP POST to the URL:
	    `<protocol>://<server>/<model_name>/oslc/am/cf/resourcelink/`
	
	• Creation Factory for Element Attributes - Create new element attributes in the model by passing their resource representation in RDF/XML format using HTTP POST to the URL:
	    `<protocol>://<server>/<model_name>/oslc/am/cf/attribute/`
	
	• Creation Factory for Element Operations - Create new element operations in the model by passing their resource representation in RDF/XML format using HTTP POST to the URL:
	    `<protocol>://<server>/<model_name>/oslc/am/cf/operation/`
	
	• Creation Factory for Element Operation Parameters - Create new element operation parameters in the model by passing their resource representation in RDF/XML format using HTTP POST to the URL:
	    `<protocol>://<server>/<model_name>/oslc/am/cf/operationparameter/`
	
	• Creation Factory for Package and Element Discussions - Create new Package/element Discussions in the model by passing the Discussion representation in RDF/XML format using HTTP POST to the URL:
	    `<protocol>://<server>/<model_name>/oslc/am/cf/discussion/`
	
	• Creation Factory for Package and Element Discussion Replies - Create new Package and element Discussion replies in the model by passing the Reply representation in RDF/XML format using HTTP POST to the URL:
	    `<protocol>://<server>/<model_name>/oslc/am/cf/reply/`
	
	• Creation Factory for Package and Element Resource Allocations - Create new Package and element Resource Allocations in the model by passing the Resource Allocation representation in RDF/XML format using HTTP POST to the URL:
	    `<protocol>://<server>/<model_name>/oslc/am/cf/resourceallocation/`
	
	• Creation Factory for Package and Element Tests - Create new Package and element Tests in the model by passing the Test representation in RDF/XML format using HTTP POST to the URL:
	    `<protocol>://<server>/<model_name>/oslc/am/cf/test/`
	
	• Creation Factory for Package and Element Tagged Values - Create new Package and element Tagged Values in the model by passing their resource representation in RDF/XML format using HTTP POST to the URL:
	    `<protocol>://<server>/<model_name>/oslc/am/cf/taggedvalue/`
	
	• Creation Factory for Package and Element Maintenance Items - Create new Package and element Maintenance Items (Changes, Defects, Issues, Tasks, Event, Decision) in the model by passing their resource representation in RDF/XML format using HTTP POST to the URL:

	- Change - <protocol>://<server>/<model_name>/oslc/am/cf/change/
	- Defect - <protocol>://<server>/<model_name>/oslc/am/cf/defect/
	- Issue - <protocol>://<server>/<model_name>/oslc/am/cf/issue/
	- Task - <protocol>://<server>/<model_name>/oslc/am/cf/task/
	- Event - <protocol>://<server>/<model_name>/oslc/am/cf/event/
	- Decision - <protocol>://<server>/<model_name>/oslc/am/cf/decision/
	
	• Creation Factory for Package and Element Project Management Items - Create new Package and element Project Management Items (Efforts, Risks and Metrics) in the model by passing their resource representation in RDF format using HTTP POST to the URL:
	  - Effort - <protocol>://<server>/<model_name>/oslc/am/cf/effort/
	- Risk - <protocol>://<server>/<model_name>/oslc/am/cf/risk/
	- Metric - <protocol>://<server>/<model_name>/oslc/am/cf/metric/
	
	• Query Capability - Query the Enterprise Architect Model for its resources by passing the query using HTTP GET to the URL:
	  <protocol>://<server>/<model_name>/oslc/am/qc/
	
	• Link Type Query Capability - Query the Enterprise Architect Model to retrieve all the supported Connector Types, using the URL:
	  <protocol>://<server>/<model_name>/oslc/am/lt/

oslc:resourceShape	Specifies the resource metadata; that is, its properties and constraints.
oslc:prefixDefinition	Specifies the namespace prefixes and their namespace definitions.

Notes

- This facility is available through the Enterprise Architect Pro Cloud Server
- Creation Factory Service will not be available when the Pro Cloud Server connection to the Enterprise Architect model is configured as a read-only connection

Resource Shape

As an OSLC Provider, the information returned by Enterprise Architect is in RDF/XML format. The OSLC Consumer needs to know the semantics of the information in this XML before it can process the XML. These semantics are explained using the Resource Shape.

Resource Shape specifies the metadata of a Resource; that is, the set of standard OSLC and custom Enterprise Architect properties that make up the Resource. To create a new Resource via OSLC, Enterprise Architect expects the RDF/XML representation of the Resource in a certain format. The metadata of this format is specified by the Resource Shape. Similarly, Resources retrieved from Enterprise Architect's OSLC interface will be in RDF/XML format. The metadata for the contents of this format is also specified by the Resource Shape. You can think of Resource Shape as a DTD or XSD for the RDF/XML response from Enterprise Architect.

The base URL for Resource Shape is :

<protocol>://<server>/<model_name>/oslc/am/rs/

A string expressing the specific Resource or Resource Feature should be added to the base URI and addressed to the model using a HTTP GET request. The response for this request will be the metadata in RDF/XML format.

Resource Shape for Resource/Resource Features Retrieved from the OSLC Interface

Resource/Feature	Resource Shape URL
Resource	<protocol>://<server>/<model_name>/oslc/am/rs/resource/
Resource Features	<protocol>://<server>/<model_name>/oslc/am/rs/resourcefeatures/
Stereotype Name	<protocol>://<server>/<model_name>/oslc/am/rs/stereotypename/
Run State	<protocol>://<server>/<model_name>/oslc/am/rs/runstate/
Attribute	<protocol>://<server>/<model_name>/oslc/am/rs/attribute/
Operation	<protocol>://<server>/<model_name>/oslc/am/rs/operation/
Operation Parameter	<protocol>://<server>/<model_name>/oslc/am/rs/operationparameter/
Requirement	<protocol>://<server>/<model_name>/oslc/am/rs/requirement/
Constraint	<protocol>://<server>/<model_name>/oslc/am/rs/constraint/
Scenario	<protocol>://<server>/<model_name>/oslc/am/rs/scenario/
Tagged Value	<protocol>://<server>/<model_name>/oslc/am/rs/taggedvalue/
Test	<protocol>://<server>/<model_name>/oslc/am/rs/test/
Resource Allocation	<protocol>://<server>/<model_name>/oslc/am/rs/resourceallocation/
Maintenance Items	• Change : <protocol>://<server>/<model_name>/oslc/am/rs/change/ • Defect : <protocol>://<server>/<model_name>/oslc/am/rs/defect/ • Issue : <protocol>://<server>/<model_name>/oslc/am/rs/issue/

	• Task : <protocol>://<server>/<model_name>/oslc/am/rs/task/ • Event : <protocol>://<server>/<model_name>/oslc/am/rs/event/ • Decision : <protocol>://<server>/<model_name>/oslc/am/rs/decision/
Project Management Items	• Effort : <protocol>://<server>/<model_name>/oslc/am/rs/effort/ • Risk : <protocol>://<server>/<model_name>/oslc/am/rs/risk/ • Metric : <protocol>://<server>/<model_name>/oslc/am/rs/metric/
Discussion/Reply	• Discussion : <protocol>://<server>/<model_name>/oslc/am/rs/discussion/ • Reply : <protocol>://<server>/<model_name>/oslc/am/rs/reply/
Diagram Image	<protocol>://<server>/<model_name>/oslc/am/rs/diagramimage/
Connector	<protocol>://<server>/<model_name>/oslc/am/rs/linktype/

Resource Shape for Resource/Resource Features that can be Created using the OSLC Interface

Resource/Resource Feature	Resource Shape URL
Resource	• Packages and UML Elements <protocol>://<server>/<model>/oslc/am/rs/cfresource/ • Profile Elements <protocol>://<server>/<model>/oslc/am/rs/cfprofileresource/
Stereotype Name	<protocol>://<server>/<model_name>/oslc/am/rs/cfstereotypename/
Attribute	<protocol>://<server>/<model_name>/oslc/am/rs/cfattribute/
Operation	<protocol>://<server>/<model_name>/oslc/am/rs/cfoperation/
Operation Parameter	<protocol>://<server>/<model_name>/oslc/am/rs/cfoperationparameter/
Connector	• UML Connectors <protocol>://<server>/<model_name>/oslc/am/rs/cflinktype/ • Profile Connectors <protocol>://<server>/<model_name>/oslc/am/rs/cfprofilelinktype/
Tagged Value	<protocol>://<server>/<model_name>/oslc/am/rs/cftaggedvalue/
Test	<protocol>://<server>/<model_name>/oslc/am/rs/cftest/
Resource Allocation	<protocol>://<server>/<model_name>/oslc/am/rs/cfresourceallocation/
Maintenance Items	• Change <protocol>://<server>/<model_name>/oslc/am/rs/cfchange/ • Defect

	<protocol>://<server>/<model_name>/oslc/am/rs/cfdefect/ • Issue <protocol>://<server>/<model_name>/oslc/am/rs/cfissue/ • Task <protocol>://<server>/<model_name>/oslc/am/rs/cftask/ • Event <protocol>://<server>/<model_name>/oslc/am/rs/cfevent/ • Decision <protocol>://<server>/<model_name>/oslc/am/rs/cfdecision/
Project Management Items	• Effort <protocol>://<server>/<model_name>/oslc/am/rs/cfeffort/ • Risk <protocol>://<server>/<model_name>/oslc/am/rs/cfrisk/ • Metric <protocol>://<server>/<model_name>/oslc/am/rs/cfmetric/
Discussion/Reply	• Discussion <protocol>://<server>/<model_name>/oslc/am/rs/cfdiscussion/ • Reply <protocol>://<server>/<model_name>/oslc/am/rs/cfreply/

Notes

- This facility is available through the Enterprise Architect Pro Cloud Server

User Credentials in a Security Enabled Model

If the model connected via Pro Cloud Server has User Security enabled, then you will have to validate the user credential before creating a Resource or Resource Feature. Once the user credential is validated, you will be provided with a User Security GUID that you will have to pass-in with the OSLC GET and POST Requests. Unless you pass-in the User Security GUID, the OSLC request on a Security Enabled model will not be processed by the Pro Cloud Server.

To get the User Security GUID:

- POST the User credentials (User ID and Password) to a specific URL; these are the same credentials that you will type into the 'Login to EA Repository' dialog in a security-enabled model

- Upon successful validation, the XML returned will contain the User Security GUID (amongst other information) in the XML element 'ss:useridentifier'

Validating User Credentials in a Security Enabled Model

Step	Action
1	Validate the User by POSTing to: URL: \<protocol\>://\<server\>/\<model_name\>/oslc/am/login/ POST Body: uid=\<USER ID\>;pwd=\<PASSWORD\>
2	If user validation succeeds, the response will typically be: \<?xml version="1.0" encoding="UTF-8"?\> \<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:ss="http://www.sparxsystems.com.au/oslc_am#" xmlns:foaf="http://xmlns.com/foaf/0.1/"\> \<ss:login\> \<!--boolean value that specifies whether the model allows Resource Creation/Update--\> \<ss:readonlymodel\>VALUE\</ss:readonlymodel\> \<!--boolean value that specifies whether you have valid Pro Cloud Server license--\> \<ss:validlicense\>VALUE\</ss:validlicense\> \<!--\<!--specifies the Full Name of the User in a Security Enabled Model--\> \<ss:userfullname\> \<foaf:Person\> \<foaf:name\>Web User\</foaf:name\> \</foaf:Person\> \</ss:userfullname\> \<!--specifies the User Security GUID--\> \<ss:useridentifier\>USER GUID\</ss:useridentifier\> \<!--boolean value that specifies whether you have permission to create/update Resources--\> \<ss:elementpermission\>VALUE\</ss:elementpermission\> \<!--boolean value that specifies whether you have permission to create/update Resource Tests--\> \<ss:testpermission\>VALUE\</ss:testpermission\> \<!--boolean value that specifies whether you have permission to create/update Resource Allocations--\> \<ss:resourceallocationpermission\>VALUE\</ss:resourceallocationpermission\> \<!--boolean value that specifies whether you have permission to create/update Resource Maintenance Items--\> \<ss:maintenanceitempermission\>VALUE\</ss:maintenanceitempermission\> \<!--boolean value that specifies whether you have permission to create/update Resource Project Management Items--\>

	`<ss:projectmanagementitempermission>VALUE</ss:projectmanagementitempermission>`
	  `</ss:login>`
	`</rdf:RDF>`

Passing User Security GUID with OSLC Requests

Once the User Credential is successfully validated, you pass in the User Security GUID in every OSLC request; that is, pass-in the User Security GUID:

- When retrieving a Resource or Resource Feature XML using GET Request as the value of the query string parameter 'useridentifier'
- When creating/updating a Resource or Resource Feature using POST RDF/XML as the value of the XML element 'ss:useridentifier'

Passing User Security GUID with GET Requests

For GET requests, pass-in the User Security GUID as the value of the query string parameter 'useridentifier':

- `<protocol>://<server>/<model_name>/oslc/am/OSLC REQUEST/?useridentifier=<USER GUID>`

Examples

No.	Example
1	Retrieve the Properties of an Enterprise Architect Package Resource: • Whose GUID is {5147E138-1F14-4738-B721-5FCC0F8821A8} • In a Security Enabled model • Where the User Security GUID is {72825AF9-E116-48f1-9DF8-77815E66B1A7} http://localhost:480/firebird_model/oslc/am/resource/pk_{5147E138-1F14-4738-B721-5FCC0F8821A8}/?useridentifier={72825AF9-E116-48f1-9DF8-77815E66B1A7}
2	Retrieve all Resources: • With the name Class1 • In a Security Enabled model • Where the User Security GUID is {72825AF9-E116-48f1-9DF8-77815E66B1A7} http://localhost:480/firebird_model/oslc/am/qc/?useridentifier={72825AF9-E116-48f1-9DF8-77815E66B1A7}&oslc.where=dcterms:title="Class1"

Passing in a User Security GUID with OSLC POST Requests

For OSLC POST requests, you need to pass-in the User Security GUID as a part of the RDF/XML accompanying the POST request.

Examples

No.	Example

1	In a Security Enabled model where the User Security GUID is {72825AF9-E116-48f1-9DF8-77815E66B1A7}, a Class 'Class1' is created under a Package with the Enterprise Architect GUID {965A54C2-6A89-46d7-AB7E-F192668010CA}. The Notes, Complexity and Version are set.
	
	<?xml version="1.0" encoding="UTF-8"?>
	
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	    xmlns:dcterms="http://purl.org/dc/terms/"
	    xmlns:foaf="http://xmlns.com/foaf/0.1/"
	    xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	  <oslc_am:Resource>
	    <!--Required Properties-->
	    <dcterms:title>Class1</dcterms:title>
	    <dcterms:type>Class</dcterms:type>
	
	<ss:parentresourceidentifier>pk_{259A54C2-6A89-46d7-AB7E-F192668010CA}</ss:parentresourceidentifier>
	    <dcterms:description>Created via OSLC</dcterms:description>
	    <ss:complexity>Easy</ss:complexity>
	    <ss:version>1.0</ss:version>
	
	<ss:useridentifier>{72825AF9-E116-48f1-9DF8-77815E66B1A7}</ss:useridentifier>
	  </oslc_am:Resource>
	</rdf:RDF>

2	In a Security Enabled model with the User Security GUID {72825AF9-E116-48f1-9DF8-77815E66B1A7}, an Actor 'Actor1' with the Enterprise Architect GUID {567854C2-6A89-46d7-AB7E-F192668010CA} has its name updated to 'Actor 1_New'.
	
	<?xml version="1.0" encoding="UTF-8"?>
	
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	    xmlns:dcterms="http://purl.org/dc/terms/"
	    xmlns:foaf="http://xmlns.com/foaf/0.1/"
	    xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	  <oslc_am:Resource>
	    <!--Required Properties-->

	`<dcterms:identifier>el_{567854C2-6A89-46d7-AB7E-F192668010CA}</dcterms:identifier>`
	
	`        <dcterms:title>Actor1_New</dcterms:identifier>`
	
	`<ss:useridentifier>{72825AF9-E116-48f1-9DF8-77815E66B1A7}</ss:useridentifier>`
	
	`   </oslc_am:Resource>`
	`</rdf:RDF>`

Notes

- This facility is available through the Enterprise Architect Pro Cloud Server
- An OSLC request will not be processed by the Pro Cloud Server unless the User Security GUID is passed-in with the OSLC request on a Security Enabled model
- 30 minutes of inactivity will log you out of a Security Enabled model; you will have to login again by POSTing the User Credentials to the credential validation URL

OSLC Access Code

It is possible to optionally set an OSLC Access Code on the connection to a model connected via Pro Cloud Server, using the 'Configure Database Manager' dialog. When this Access Code is set, Clients have to pass-in the code with each OSLC request via a custom HTTP GET/POST Header called EAO-Access-Code. The Pro Cloud Server will process the OSLC request only after validating the Access Code.

This example screen shows the Access Code (CC596BB6_9A3A_4322_8060_D40886338BE5 in the example) being passed in the HTTP GET Header via a popular REST Client called 'Postman'.

GET ⌄	http://localhost:480/firebird_model/oslc/am/sp/	Params	Send ⌄

Authorization	Headers (1)	Body	Pre-request Script	Tests

	Key	Value	Description	•••	Bulk Edit
☑	EAO-Access-Code	CC596BB6_9A3A_4322_8060_D40886338BE5			
	New key	Value	Description		

Creation Factory

Enterprise Architect supports the Creation Factory OSLC service, through which Clients can create these new Resources and Resource Features via HTTP POST:

- Resources
  - Packages
  - Elements
  - Connectors

- Resource Features
  - Package/Element Discussions, Discussion Replies, Resource Allocations, Tests, Tagged Values, Maintenance Items and Project Management Items
  - Element Attributes, Operations and Operation Parameters

To create a new Resource or Resource Feature, the Client POSTs a representation of the Resource or Resource Feature in RDF XML format to the Creation Factory URL. If the POST is successful, the HTTP location header of the response will contain the URL of the created Resource or Resource Feature. An unsuccessful POST will generate an error response.

Refer to the tables in this topic for the information used to create each Resource and Feature. Also see the *GUID Prefix Tables* and *Examples of RDF/XML Representation* topics.

Package

Type	Input
URL	\<protocol\>://\<server\>/\<model_name\>/oslc/am/cf/resource/

POST RDF/XML	\<?xml version="1.0" encoding="UTF-8"?\>
	 \<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	     xmlns:dcterms="http://purl.org/dc/terms/"
	     xmlns:foaf="http://xmlns.com/foaf/0.1/"
	     xmlns:ss="http://www.sparxsystems.com.au/oslc_am#"\>
	   \<oslc_am:Resource\>
	     \<!--Required Properties--\>
	     \<dcterms:title\>NAME\</dcterms:title\>
	     \<dcterms:type\>Package\</dcterms:type\>
	     \<ss:parentresourceidentifier\>GUID OF THE CONTAINING PACKAGE WITH GUID PREFIX\</ss:parentresourceidentifier\>
	     \<!--Required in a Security Enabled Model--\>
	     \<ss:useridentifier\>USER SECURITY GUID\</ss:useridentifier\>
	     \<!--Optional Properties--\>
	     \<dcterms:subject\>KEYWORDS\</dcterms:subject\>
	     \<dcterms:creator\>
	       \<foaf:Person\>
	         \<foaf:name\>AUTHOR\</foaf:name\>
	       \</foaf:Person\>
	     \</dcterms:creator\>
	     \<dcterms:description\>NOTES\</dcterms:description\>

	<ss:alias>ALIAS</ss:alias> <ss:status>STATUS</ss:status> <ss:complexity>COMPLEXITY</ss:complexity> <ss:stereotype> <ss:stereotypename> <ss:name>STEREOTYPE</ss:name> </ss:stereotypename> </ss:stereotype> <ss:phase>PHASE</ss:phase> <ss:version>VERSION</ss:version> </oslc_am:Resource> </rdf:RDF>
Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/cfresource/

Element

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/cf/resource/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	 <rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	     xmlns:dcterms="http://purl.org/dc/terms/"
	     xmlns:foaf="http://xmlns.com/foaf/0.1/"
	     xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	   <oslc_am:Resource>
	     <!--Required Properties-->
	     <dcterms:title>NAME</dcterms:title>
	     <dcterms:type>ELEMENT TYPE</dcterms:type>
	     <ss:parentresourceidentifier>GUID OF THE CONTAINING PACKAGE/ELEMENT WITH GUID PREFIX</ss:parentresourceidentifier>
	     <!--Required in a Security Enabled Model-->
	     <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	     <!--Optional Properties-->
	     <dcterms:subject>KEYWORDS</dcterms:subject>
	     <dcterms:creator>
	       <foaf:Person>
	         <foaf:name>AUTHOR</foaf:name>
	       </foaf:Person>
	     </dcterms:creator>
	     <dcterms:description>NOTES</dcterms:description>

<table>
<tr><td></td><td>

<ss:alias>ALIAS</ss:alias>

<ss:status>STATUS</ss:status>

<ss:complexity>COMPLEXITY</ss:complexity>

<ss:stereotype>

  <ss:stereotypename>

    <ss:name>STEREOTYPE</ss:name>

  </ss:stereotypename>

</ss:stereotype>

<ss:phase>PHASE</ss:phase>

<ss:version>VERSION</ss:version>

 </oslc_am:Resource>

</rdf:RDF>

</td></tr>
</table>

Resource Shape	• UML Elements: <protocol>://<server>/<model>/oslc/am/rs/cfresource/ • Profile Elements: <protocol>://<server>/<model>/oslc/am/rs/cfprofileresource/

Element Attribute

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/cf/attribute/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	    xmlns:dcterms="http://purl.org/dc/terms/"
	    xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	  <ss:attribute>
	    <!--Required Properties-->
	    <ss:resourceidentifier>GUID OF THE CONTAINING EA ELEMENT WITH GUID PREFIX</ss:resourceidentifier>
	    <dcterms:title>ATTRIBUTE NAME</dcterms:title>
	    <!--Required in a Security Enabled Model-->
	    <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	    <!--Optional Properties-->
	    <ss:classifiername>DATATYPE/CLASSIFIER NAME</ss:classifiername>
	    <dcterms:description>NOTES</dcterms:description>
	    <ss:alias>ALIAS</ss:alias>
	    <ss:classifierresourceidentifier>GUID OF THE CLASSIFIER EA ELEMENT WITH GUID PREFIX</ss:classifierresourceidentifier>
	    <ss:defaultvalue>DEFAULT VALUE</ss:defaultvalue>
	    <ss:stereotype>

	<ss:stereotypename> <ss:name>STEREOTYPE</ss:name> </ss:stereotypename> </ss:stereotype> <ss:scope>SCOPE</ss:scope> <ss:containment>CONTAINMENT</ss:containment> <ss:isstatic>IS STATIC</ss:isstatic> <ss:iscollection>IS COLLECTION</ss:iscollection> <ss:isordered>IS ORDERED</ss:isordered> <ss:isconst>IS CONST</ss:isconst> <ss:allowduplicates>ALLOW DUPLICATES</ss:allowduplicates> <ss:lowerbound>NUMERICAL VALUE</ss:lowerbound> <ss:upperbound>NUMERICAL VALUE</ss:upperbound> <ss:position>NUMERICAL VALUE</ss:position> </ss:attribute> </rdf:RDF>
Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/cfattribute/

Element Operation Parameter

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/cf/operationparameter/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	
	   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	
	   xmlns:dcterms="http://purl.org/dc/terms/"
	
	   xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	
	<ss:parameter>
	
	<!--Required Properties-->
	
	<ss:operationidentifier>GUID OF THE CONTAINING OPERATION WITH GUID PREFIX</ss:operationidentifier>
	
	<dcterms:title>OPERATION PARAMETER NAME</dcterms:title>
	
	<!--Required in a Security Enabled Model-->
	
	<ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	
	<!--Optional Properties-->
	
	<dcterms:description>NOTES</dcterms:description>
	
	<ss:classifiername>DATATYPE/CLASSIFIER NAME</ss:classifiername>
	
	<ss:classifierresourceidentifier>GUID OF THE CLASSIFIER EA ELEMENT WITH GUID PREFIX</ss:classifierresourceidentifier>
	
	<ss:defaultvalue>DEFAULT VALUE</ss:defaultvalue>
	
	<ss:paramdirection>DIRECTION</ss:paramdirection>

	`<ss:position>NUMERICAL VALUE</ss:position>`
	`</ss:parameter>`
	`</rdf:RDF>`

Resource Shape	`<protocol>://<server>/<model>/oslc/am/rs/cfoperationparameter/`

Connector

Type	Input
URL	`<protocol>://<server>/<model_name>/oslc/am/cf/resourcelink/`

POST RDF/XML	`<?xml version="1.0" encoding="UTF-8"?>`
	`<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"`
	`        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"`
	`        xmlns:dcterms="http://purl.org/dc/terms/"`
	`        xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">`
	`  <oslc_am:Resource>`
	`    <!--Required Properties-->`
	`    <dcterms:identifier>GUID OF THE SOURCE PACKAGE/ELEMENT WITH GUID PREFIX</dcterms:identifier>`
	`    <ss:{connector type} rdf:ID="ID"`
	`rdf:resource="<protocol>://<server>:<port>/<model>/oslc/am/resource/<GUID OF TARGET PACKAGE/ELEMENT WITH GUID PREFIX>/"/>`
	`  </oslc_am:Resource>`
	`  <rdf:Description rdf:about="#ID">`
	`    <!--Required in a Security Enabled Model-->`
	`    <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>`
	`    <!--Optional Properties-->`
	`    <dcterms:title>CONNECTOR NAME</dcterms:title>`
	`    <dcterms:type> CONNECTOR TYPE</dcterms:type>`
	`    <dcterms:description>NOTES</dcterms:description>`
	`    <ss:direction>DIRECTION</ss:direction>`
	`    <ss:stereotype>`
	`      <ss:stereotypename>`
	`        <ss:name>STEREOTYPE</ss:name>`
	`      <ss:/stereotypename>`
	`    </ss:stereotype>`
	`  </rdf:Description>`
	`</rdf:RDF>`

Resource Shape	• UML Connectors:
	  `<protocol>://<server>/<model>/oslc/am/rs/cflinktype/`
	• Profile Connectors:
	  `<protocol>://<server>/<model>/oslc/am/rs/cfprofilelinktype/`

Package/Element Discussion

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/cf/discussion/

POST RDF/XML	`<?xml version="1.0" encoding="UTF-8"?>`
	`<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"`
	    `xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"`
	    `xmlns:dcterms="http://purl.org/dc/terms/"`
	    `xmlns:foaf="http://xmlns.com/foaf/0.1/"`
	    `xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">`
	  `<ss:discussion>`
	    `<!--Required Properties-->`
	    `<dcterms:description>DISCUSSION TEXT</dcterms:description>`
	    `<dcterms:creator>`
	      `<foaf:Person>`
	        `<foaf:name>AUTHOR</foaf:name>`
	      `</foaf:Person>`
	    `</dcterms:creator>`
	    `<ss:resourceidentifier>GUID OF THE PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>`
	    `<!--Required in a Security Enabled Model-->`
	    `<ss:useridentifier>USER SECURITY GUID</ss:useridentifier>`
	    `<!--Optional Properties-->`
	    `<ss:status>DISCUSSION STATUS</ss:status>`
	    `<ss:priority>DISCUSSION PRIORITY</ss:priority>`
	    `<ss:reviewresourceidentifier>GUID OF THE REVIEW ELEMENT WITH GUID PREFIX</ss:reviewresourceidentifier>`
	  `</ss:discussion>`
	`</rdf:RDF>`
	
	Note:
	• If the Discussion is a part of a Review, then set the GUID of the Review Element in the ss:reviewresourceidentifier XML element

Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/cfdiscussion/

Package/Element Discussion Reply

Type	Input

URL	<protocol>://<server>/<model_name>/oslc/am/cf/reply/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	    xmlns:dcterms="http://purl.org/dc/terms/"
	    xmlns:foaf="http://xmlns.com/foaf/0.1/"
	    xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	  <ss:reply>
	    <!--Required Properties-->
	    <dcterms:description>REPLY TEXT</dcterms:description>
	    <dcterms:creator>
	      <foaf:Person>
	        <foaf:name>AUTHOR</foaf:name>
	      </foaf:Person>
	    </dcterms:creator>
	    <ss:discussionidentifier>GUID OF THE DISCUSSION WITH GUID PREFIX</ss:resourceidentifier>
	    <!--Required in a Security Enabled Model-->
	    <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	    <!--Optional Properties-->
	    <ss:reviewresourceidentifier>GUID OF THE REVIEW ELEMENT WITH GUID PREFIX</ss:reviewresourceidentifier>
	  </ss:reply>
	</rdf:RDF>
	
	Note:
	• If the Discussion Reply is a part of a Review, then set the GUID of the Review Element in ss:reviewresourceidentifier XML element

Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/cfreply/

Package/Element Resource Allocation

Type	Input
UML	<protocol>://<server>/<model_name>/oslc/am/cf/resourceallocation/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	    xmlns:dcterms="http://purl.org/dc/terms/"
	    xmlns:foaf="http://xmlns.com/foaf/0.1/"

	xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	<ss:resourceallocation>
	<!--Required Properties-->
	<ss:resourcename>
	<foaf:Person>
	<foaf:name>RESOURCE NAME</foaf:name>
	</foaf:Person>
	</ss:resourcename>
	<ss:role>RESOURCE ROLE</ss:role>
	<ss:startdate>YYYY-MM-DD FORMAT</ss:startdate>
	<ss:enddate>YYYY-MM-DD FORMAT</ss:enddate>
	<ss:resourceidentifier>GUID OF THE CONTAINING PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>
	<!--Required in a Security Enabled Model-->
	<ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	<!--Optional Properties-->
	<ss:percentagecomplete>PERCENT COMPLETE</ss:percentagecomplete>
	<ss:expectedtime>EXPECTED TIME</ss:expectedtime>
	<ss:allocatedtime>ALLOCATED TIME</ss:allocatedtime>
	<ss:expendedtime>EXPENDED TIME</ss:expendedtime>
	<dcterms:description>NOTES</dcterms:description>
	<ss:history>HISTORY</ss:history>
	</ss:resourceallocation>
	</rdf:RDF>
Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/cfresourceallocation/

Package/Element Test

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/cf/test/
POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	xmlns:dcterms="http://purl.org/dc/terms/"
	xmlns:foaf="http://xmlns.com/foaf/0.1/"
	xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	<ss:test>
	<!--Required Properties-->
	<dcterms:title>TEST NAME</dcterms:title>
	<dcterms:type>TYPE</dcterms:type>

	<ss:classtype>CLASS TYPE</ss:classtype>
	<ss:status>STATUS</ss:status>
	<!--Required in a Security Enabled Model-->
	<ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	<!--Optional Properties-->
	<ss:lastrun>YYYY-MM-DD FORMAT</ss:lastrun>
	<ss:runby>
	<foaf:Person>
	<foaf:name>PERSON NAME</foaf:name>
	</foaf:Person>
	</ss:runby>
	<ss:checkedby>
	<foaf:Person>
	<foaf:name>PERSON NAME</foaf:name>
	</foaf:Person>
	</ss:checkedby>
	<ss:resourceidentifier>GUID OF THE CONTAINING PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>
	<dcterms:description>NOTES</dcterms:description>
	<ss:input>HISTORY TEXT</ss:input>
	<ss:acceptancecriteria>ACCEPTANCE CRITERIA TEXT</ss:acceptancecriteria>
	<ss:results>RESULTS TEXT</ss:results>
	</ss:test>
	</rdf:RDF>
Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/cftest/

Package/Element Tagged Value

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/cf/taggedvalue/
POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	xmlns:dcterms="http://purl.org/dc/terms/"
	xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	<ss:taggedvalue>
	<!--Required Properties-->
	<dcterms:title>TAG NAME</dcterms:title>
	<ss:resourceidentifier>GUID OF THE CONTAINING PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>

<!--Required in a Security Enabled Model-->

<ss:useridentifier>USER SECURITY GUID</ss:useridentifier>

<!--Optional Properties-->

<ss:value>TAG VALUE</ss:value>

<dcterms:description>TAG NOTES</dcterms:description>

<ss:allowduplicates>BOOLEAN VALUE</ss:allowduplicates>

    </ss:taggedvalue>

</rdf:RDF>

Notes:
- dcterms:description can contain either Tag Notes or these Predefined Structured Types:
  - Boolean
  - Color
  - Const
  - Date (YYYY-MM-DD Format)
  - Directory
  - Enum
  - File
  - Float
  - Decimal
  - Double
  - Integer
  - Memo
  - RefGUID
  - RefGUIDList
  - String
  - Time (HH:MM Format)
  - Timestamp (YYYY-MM-DD HH:MM Format)
  - URL

Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/cftaggedvalue/

Package/Element Change

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/cf/maintenanceitem/

Post RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	      xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	      xmlns:dcterms="http://purl.org/dc/terms/"
	      xmlns:foaf="http://xmlns.com/foaf/0.1/"
	      xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	  <ss:change>
	    <!--Required Properties-->
	    <dcterms:title>CHANGE NAME</dcterms:title>

<table>
<tr><td></td><td>

```
        <ss:resourceidentifier>GUID OF THE CONTAINING
PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>
          <!--Required in a Security Enabled Model-->
          <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
          <!--Optional Properties-->
          <ss:requestedby>
            <foaf:Person>
               <foaf:name>PERSON NAME</foaf:name>
            </foaf:Person>
          </ss:requestedby>
          <ss:requestedon>YYYY-MM-DD format</ss:requestedon>
          <ss:status>STATUS</ss:status>
          <ss:priority>PRIORITY</ss:priority>
          <ss:completedby>
            <foaf:Person>
               <foaf:name>PERSON NAME</foaf:name>
            </foaf:Person>
          </ss:completedby>
          <ss:completedon>YYYY-MM-DD format</ss:completedon>
          <ss:version>VERSION</ss:version>
          <dcterms:description>NOTES</dcterms:description>
          <ss:history>HISTORY</ss:history>
        </ss:change>
      </rdf:RDF>
```

</td></tr>
<tr><td>Resource Shape</td><td><protocol>://<server>/<model>/oslc/am/rs/cfchange/</td></tr>
</table>

Package/Element Defect

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/cf/maintenanceitem/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	    xmlns:dcterms="http://purl.org/dc/terms/"
	    xmlns:foaf="http://xmlns.com/foaf/0.1/"
	    xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	  <ss:defect>
	    <!--Required Properties-->
	    <dcterms:title>DEFECT NAME</dcterms:title>
	    <ss:resourceidentifier>GUID OF THE CONTAINING PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>

	<!--Required in a Security Enabled Model-->
	<ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	<!--Optional Properties-->
	<ss:reportedby>
	  <foaf:Person>
	    <foaf:name>PERSON NAME</foaf:name>
	  </foaf:Person>
	</ss:reportedby>
	<ss:reportedon>YYYY-MM-DD format</ss:reportedon>
	<ss:status>STATUS</ss:status>
	<ss:priority>PRIORITY</ss:priority>
	<ss:resolvedby>
	  <foaf:Person>
	    <foaf:name>PERSON NAME</foaf:name>
	  </foaf:Person>
	</ss:resolvedby>
	<ss:resolvedon>YYYY-MM-DD format</ss:resolvedon>
	<ss:version>VERSION</ss:version>
	<dcterms:description>NOTES</dcterms:description>
	<ss:history>HISTORY</ss:history>
	 </ss:defect>
	</rdf:RDF>

Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/cfdefect/

Package/Element Issue

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/cf/maintenanceitem/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	    xmlns:dcterms="http://purl.org/dc/terms/"
	    xmlns:foaf="http://xmlns.com/foaf/0.1/"
	    xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	  <ss:issue>
	    <!--Required Properties-->
	    <dcterms:title>ISSUE NAME</dcterms:title>
	    <ss:resourceidentifier>GUID OF THE CONTAINING PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>
	    <!--Required in a Security Enabled Model-->

	<ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	<!--Optional Properties-->
	<ss:raisedby>
	  <foaf:Person>
	    <foaf:name>PERSON NAME</foaf:name>
	  </foaf:Person>
	</ss:raisedby>
	<ss:raisedon>YYYY-MM-DD format</ss:raisedon>
	<ss:status>STATUS</ss:status>
	<ss:priority>PRIORITY</ss:priority>
	<ss:completedby>
	  <foaf:Person>
	    <foaf:name>PERSON NAME</foaf:name>
	  </foaf:Person>
	</ss:completedby>
	<ss:completedon>YYYY-MM-DD format</ss:completedon>
	<ss:version>VERSION</ss:version>
	<dcterms:description>NOTES</dcterms:description>
	<ss:history>HISTORY</ss:history>
	</ss:issue>
	</rdf:RDF>

Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/cfissue/

Package/Element Task

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/cf/maintenanceitem/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	      xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	      xmlns:dcterms="http://purl.org/dc/terms/"
	      xmlns:foaf="http://xmlns.com/foaf/0.1/"
	      xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	  <ss:task>
	    <!--Required Properties-->
	    <dcterms:title>TASK NAME</dcterms:title>
	    <ss:resourceidentifier>GUID OF THE CONTAINING PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>
	    <!--Required in a Security Enabled Model-->
	    <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>

	<!--Optional Properties--> <ss:requestedby> <foaf:Person> <foaf:name>PERSON NAME</foaf:name> </foaf:Person> </ss:requestedby> <ss:requestedon>YYYY-MM-DD format</ss:requestedon> <ss:status>STATUS</ss:status> <ss:priority>PRIORITY</ss:priority> <ss:completedby> <foaf:Person> <foaf:name>PERSON NAME</foaf:name> </foaf:Person> </ss:completedby> <ss:completedon>YYYY-MM-DD format</ss:completedon> <ss:version>VERSION</ss:version> <dcterms:description>NOTES</dcterms:description> <ss:history>HISTORY</ss:history> </ss:task> </rdf:RDF>
Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/cftask/

Package/Element Event

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/cf/maintenanceitem/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	      xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	      xmlns:dcterms="http://purl.org/dc/terms/"
	      xmlns:foaf="http://xmlns.com/foaf/0.1/"
	      xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	  <ss:event>
	   <!--Required Properties-->
	   <dcterms:title>EVENT NAME</dcterms:title>
	   <ss:resourceidentifier>GUID OF THE CONTAINING PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>
	   <!--Required in a Security Enabled Model-->
	   <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	   <!--Optional Properties-->

| | ```
      <ss:reportedby>
        <foaf:Person>
          <foaf:name>PERSON NAME</foaf:name>
        </foaf:Person>
      </ss:reportedby>
      <ss:reportedon>YYYY-MM-DD format</ss:reportedon>
      <dcterms:type>TYPE</dcterms:type>
      <ss:priority>PRIORITY</ss:priority>
      <ss:resolvedby>
        <foaf:Person>
          <foaf:name>PERSON NAME</foaf:name>
        </foaf:Person>
      </ss:resolvedby>
      <ss:resolvedon>YYYY-MM-DD format</ss:resolvedon>
      <ss:version>VERSION</ss:version>
      <dcterms:description>NOTES</dcterms:description>
      <ss:history>HISTORY</ss:history>
    </ss:event>
  </rdf:RDF>
``` |
| Resource Shape | `<protocol>://<server>/<model>/oslc/am/rs/cfevent/` |

Package/Element Decision

Type	Input
URL	`<protocol>://<server>/<model_name>/oslc/am/cf/maintenanceitem/`
| POST RDF/XML | ```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
         xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:dcterms="http://purl.org/dc/terms/"
         xmlns:foaf="http://xmlns.com/foaf/0.1/"
         xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
  <ss:decision>
    <!--Required Properties-->
    <dcterms:title>DECISION NAME</dcterms:title>
    <ss:resourceidentifier>GUID OF THE CONTAINING
PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>
    <!--Required in a Security Enabled Model-->
    <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
    <!--Optional Properties-->
    <ss:owner>
``` |

	`<foaf:Person>`
	`<foaf:name>PERSON NAME</foaf:name>`
	`</foaf:Person>`
	`</ss:owner>`
	`<ss:date>YYYY-MM-DD format</ss:date>`
	`<ss:status>STATUS</ss:status>`
	`<ss:impact>IMPACT</ss:impact>`
	`<ss:author>`
	`<foaf:Person>`
	`<foaf:name>PERSON NAME</foaf:name>`
	`</foaf:Person>`
	`</ss:author>`
	`<ss:effective>YYYY-MM-DD format</ss:effective>`
	`<ss:version>VERSION</ss:version>`
	`<dcterms:description>NOTES</dcterms:description>`
	`<ss:history>HISTORY</ss:history>`
	`</ss:decision>`
	`</rdf:RDF>`

Resource Shape	`<protocol>://<server>/<model>/oslc/am/rs/cfdecision/`

Package/Element Effort

Type	Input
URL	`<protocol>://<server>/<model_name>/oslc/am/cf/projectmanagementitem/`

POST RDF/XML	`<?xml version="1.0" encoding="UTF-8"?>`
	`<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"`
	`xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"`
	`xmlns:dcterms="http://purl.org/dc/terms/"`
	`xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">`
	`<ss:effort>`
	`<!--Required Properties-->`
	`<dcterms:title>EFFORT NAME</dcterms:title>`
	`<ss:resourceidentifier>GUID OF THE CONTAINING PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>`
	`<!--Required in a Security Enabled Model-->`
	`<ss:useridentifier>USER SECURITY GUID</ss:useridentifier>`
	`<!--Optional Properties-->`
	`<dcterms:type>TYPE</dcterms:type>`
	`<ss:time>NUMERICAL VALUE</ss:time>`
	`<dcterms:description>NOTES</dcterms:description>`

	</ss:effort>
	</rdf:RDF>
Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/cfeffort/

Package/Element Risk

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/cf/projectmanagementitem/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	        xmlns:dcterms="http://purl.org/dc/terms/"
	        xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	  <ss:risk>
	    <!--Required Properties-->
	    <dcterms:title>RISK NAME</dcterms:title>
	    <ss:resourceidentifier>GUID OF THE CONTAINING PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>
	    <!--Required in a Security Enabled Model-->
	    <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	    <!--Optional Properties-->
	    <dcterms:type>RISK TYPE</dcterms:type>
	    <ss:weight>NUMERICAL VALUE</ss:weight>
	    <dcterms:description>NOTES</dcterms:description>
	  </ss:risk>
	</rdf:RDF>

Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/cfrisk/

Package/Element Metric

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/cf/projectmanagementitem/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	        xmlns:dcterms="http://purl.org/dc/terms/"

	xmlns:ss="http://www.sparxsystems.com.au/oslc_am#"> <ss:metric> <!--Required Properties--> <dcterms:title>METRIC NAME</dcterms:title> <ss:resourceidentifier>GUID OF THE CONTAINING PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier> <!--Required in a Security Enabled Model--> <ss:useridentifier>USER SECURITY GUID</ss:useridentifier> <!--Optional Properties-→ <dcterms:type>TYPE</dcterms:type> <ss:weight>NUMERICAL VALUE</ss:weight> <dcterms:description>NOTES</dcterms:description> </ss:metric> </rdf:RDF>
Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/cfmetric/

Notes

- This facility is available through the Enterprise Architect Pro Cloud Server

- In a Security Enabled model, you need to provide the User Security GUID (via the 'ss:useridentifier' XML element) in the POST XML to create a Resource/Resource Feature

- Enterprise Architect currently supports creating elements and connectors from these Profiles:
    - Archimate 3
    - BPMN 2.0
    - GML 3.3
    - SysML 1.4 GUID

Element Operation

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/cf/operation/
POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?> <rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:dcterms="http://purl.org/dc/terms/" xmlns:ss="http://www.sparxsystems.com.au/oslc_am#"> <ss:operation> <!--Required Properties--> <ss:resourceidentifier>GUID OF THE CONTAINING EA ELEMENT WITH GUID PREFIX</ss:resourceidentifier> <dcterms:title>OPERATION NAME</dcterms:title>

	<!--Required in a Security Enabled Model--> <ss:useridentifier>USER SECURITY GUID</ss:useridentifier> <!--Optional Properties--> <dcterms:description>NOTES</dcterms:description> <ss:alias>ALIAS</ss:alias> <ss:classifiername>DATATYPE/CLASSIFIER NAME</ss:classifiername> <ss:classifierresourceidentifier>GUID OF THE CLASSIFIER EA ELEMENT WITH GUID PREFIX</ss:classifierresourceidentifier> <ss:stereotype> <ss:stereotypename> <ss:name>STEREOTYPE</ss:name> </ss:stereotypename> </ss:stereotype> <ss:scope>SCOPE</ss:scope> <ss:isstatic>IS STATIC</ss:isstatic> <ss:isabstract>IS ABSTRACT</ss:isabstract> <ss:isreturnarray>IS RETURNARRAY</ss:isreturnarray> <ss:isquery>IS QUERY</ss:isquery> <ss:issynchronized>IS SYNCHRONIZED</ss:issynchronized> <ss:isconst>IS CONST</ss:isconst> <ss:ispure>IS PURE</ss:ispure> <ss:position>NUMERICAL VALUE</ss:position> <ss:behaviour>BEHAVIOR</ss:behaviour> </ss:operation> </rdf:RDF>
Resource Shape	\<protocol\>://\<server\>/\<model\>/oslc/am/rs/cfoperation/

Learn more

- GUID Prefix Tables
- Examples of RDF/XML Representations
- Packages
- UML Elements
- Domain Models
- Attributes
- Operations
- Define Parameters
- UML Connectors
- Informal Element Discussions
- Formal Review Elements
- Resource Allocation
- Create Test Records

- [Tagged Values](#)
- [Predefined Structured Types](#)
- [Create Maintenance Items](#)
- [Effort Management](#)
- [Risk Management](#)
- [Metrics](#)

GUID Prefix Tables

These tables provide the prefixes for the GUIDS of Resources and Resource Features of the various types.

Resource Types

Resource Type	GUID Prefix
Model Root	mr_
Package	pk_
Element	el_
Diagram	dg_
Link	lt_

Resource Feature Types

Resource Feature Type	GUID Prefix
Run State	rs_
Attribute	at_
Operation	op_
Operation Parameter	pr_
Requirement	rq_
Constraint	cn_
Scenario	sc_
Tagged Value	tv_
Test	ts_
Resource Allocation	ra_
Change	ch_
Defect	df_

Issue	is_
Task	tk_
Event	ev_
Decision	dc_
Effort	ef_
Risk	ri_
Metric	me_
Discussion/Reply	dr_

Examples of RDF/XML Representations

This topic provides a number of examples of the RDF/XML representation of Resource Creation Factory XML.

Create an element

Create an Actor called 'Actor1' in a Package with the Enterprise Architect GUID {259A54C2-6A89-46d7-AB7E-F192668010CA}.

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:dcterms="http://purl.org/dc/terms/"
        xmlns:foaf="http://xmlns.com/foaf/0.1/"
        xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
  <oslc_am:Resource>
    <!--Required Properties-->
    <dcterms:title>Actor1</dcterms:title>
    <dcterms:type>Actor</dcterms:type>
    <ss:parentresourceidentifier>pk_{259A54C2-6A89-46d7-AB7E-F192668010CA}</ss:parentresourceidentifier>
  </oslc_am:Resource>
</rdf:RDF>
```

Create an element in a Security Enabled model

Create a Class called 'Class1' in a Security Enabled model, under a Package with the Enterprise Architect GUID {965A54C2-6A89-46d7-AB7E-F192668010CA}, and set its Notes, Complexity and Version.

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:dcterms="http://purl.org/dc/terms/"
        xmlns:foaf="http://xmlns.com/foaf/0.1/"
        xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
  <oslc_am:Resource>
    <!--Required Properties-->
    <dcterms:title>Class1</dcterms:title>
    <dcterms:type>Class</dcterms:type>
    <ss:parentresourceidentifier>pk_{259A54C2-6A89-46d7-AB7E-F192668010CA}</ss:parentresourceidentifier>
    <!--Required in a Security Enabled Model-->
    <ss:useridentifier>{B6106F48-9659-4bff-B52A-C68F638BB1CE}</ss:useridentifier>
    <!--Optional Properties-->
```

```
    <dcterms:description>Created via OSLC</dcterms:description>
    <ss:complexity>Easy</ss:complexity>
    <ss:version>1.0</ss:version>
    <ss:useridentifier>{B6106F48-9659-4bff-B52A-C68F638BB1CE}</ss:useridentifier>
  </oslc_am:Resource>
</rdf:RDF>
```

Create a BPMN2.0 Activity

Create a BPMN 2.0 Activity called 'Retrieve Parts List' in a Package with the Enterprise Architect GUID {259A54C2-6A89-46d7-AB7E-F192668010CA}.

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:dcterms="http://purl.org/dc/terms/"
        xmlns:foaf="http://xmlns.com/foaf/0.1/"
        xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
  <oslc_am:Resource>
    <!--Required Properties-->
    <dcterms:title>Retrieve Parts List</dcterms:title>
    <dcterms:type>BPMN2.0::Activity(UML::Activity)</dcterms:type>
    <ss:parentresourceidentifier>pk_{259A54C2-6A89-46d7-AB7E-F192668010CA}</ss:parentresourceidentifier>
  </oslc_am:Resource>
</rdf:RDF>
```

Create a Use Case with Stereotype

Create a UseCase called 'Validate Login' with the Stereotype 'validator', in a Package with the Enterprise Architect GUID {259A54C2-6A89-46d7-AB7E-F192668010CA}.

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:dcterms="http://purl.org/dc/terms/"
        xmlns:foaf="http://xmlns.com/foaf/0.1/"
        xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
  <oslc_am:Resource>
    <!--Required Properties-->
    <dcterms:title>Validate Login</dcterms:title>
    <dcterms:type>UseCase</dcterms:type>
    <ss:parentresourceidentifier>pk_{259A54C2-6A89-46d7-AB7E-F192668010CA}</ss:parentresourceidentifier>
```

```
        <!--Optional Properties-->
        <ss:stereotype>
            <ss:stereotypename>
                <ss:name>validator</ss:name>
            </ss:stereotypename>
        </ss:stereotype>
    </oslc_am:Resource>
</rdf:RDF>
```

Create an Attribute

Create a UML attribute called 'attribute1' in a Class with the Enterprise Architect GUID {BE901CA9-FB3D-4af6-AEE2-B5CE6585F9D1}, and set its Type, Default Value and Scope.

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:dcterms="http://purl.org/dc/terms/"
        xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
    <ss:attribute>
        <!--Required Properties-->
        <ss:resourceidentifier>el_{BE901CA9-FB3D-4af6-AEE2-B5CE6585F9D1}</ss:resourceidentifier>
        <dcterms:title>attribute1</dcterms:title>
        <!--Optional Properties-->
        <ss:classifiername>int</ss:classifiername>
        <ss:defaultvalue>10</ss:defaultvalue>
    <ss:scope>Private</ss:scope>
</ss:attribute>
```

Create an enumeration Tagged Value

Create an enumeration-type Tagged Value (with the values 'Val1', 'Val2' and 'Val3') on a Class with the Enterprise Architect GUID {BE901CA9-FB3D-4af6-AEE2-B5CE6585F9D1}.

```
<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:dcterms="http://purl.org/dc/terms/"
        xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
    <ss:taggedvalue>
        <!--Required Properties-->
        <dcterms:title>enumTag</dcterms:title>
        <ss:resourceidentifier>el_{BE901CA9-FB3D-4af6-AEE2-B5CE6585F9D1}</ss:resourceidentifier>
```

```
    <!--Optional Properties-->
    <ss:value>Val2</ss:value>
    <ss:description> Type=Enum;Values=Val1,Val2,Val3</ss:description>
    <ss:allowduplicates>false</ss:allowduplicates>
  </ss:taggedvalue>
</rdf:RDF>
```

Create a Dependency Connector

Create a Dependency connector called 'Actor_DepCon' from the Actor 'Actor1' (with the Enterprise Architect GUID {FACC69CC-F55E-4a81-9645-239DD50A94BB}) to the Actor 'Actor2' (with the Enterprise Architect GUID {81E80D25-F8BB-4fca-A819-317962FC0672}).

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:dcterms="http://purl.org/dc/terms/"
        xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
  <oslc_am:Resource>
    <!--Required Properties-->
    <dcterms:identifier>el_{FACC69CC-F55E-4a81-9645-239DD50A94BB}</dcterms:identifier>
    <ss:Dependency rdf:ID="ID"
rdf:resource="http://localhost:480/firebird_model/oslc/am/resource/el_{81E80D25-F8BB-4fca-A819-317962FC0672}/"/
>
  </oslc_am:Resource>
  <!--Optional Properties-->
  <rdf:Description rdf:about="#ID">
    <dcterms:title>Actor_DepCon</dcterms:title>
    <dcterms:type>Dependency</dcterms:type>
    <dcterms:description>Created via OSLC</dcterms:description>
    <ss:direction>Source → Destination</ss:direction>
  </rdf:Description>
</rdf:RDF>
```

Create a BPMN 2.0 Sequence Flow connector

Create a BPMN 2.0 Sequence Flow connector from the BPMN 2.0 Activity 'Retrieve Parts List' (with the Enterprise Architect GUID {85F77073-A93A-496b-A3C6-9C20EE33A065}) to the BPMN 2.0 Activity 'Store Order' (with the Enterprise Architect GUID {A992EC7C-92C0-488e-B7B4-5FCBE87896AE}).

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:dcterms="http://purl.org/dc/terms/"
```

```
        xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
   <oslc_am:Resource>
      <!--Required Properties-->
      <dcterms:identifier>el_{85F77073-A93A-496b-A3C6-9C20EE33A065}</dcterms:identifier>
      <ss:ControlFlow rdf:ID="ID" rdf:resource="http://localhost:480/firebird_model/oslc/am/resource/el_{A9 GUID
92EC7C-92C0-488e-B7B4-5FCBE87896AE}/"/>
   </oslc_am:Resource>
   <rdf:Description rdf:about="#ID">
      <dcterms:type>BPMN2.0::SequenceFlow(UML::ControlFlow)</dcterms:type>
      <!--Optional Properties-->
      <ss:direction>Source → Destination</ss:direction>
   </rdf:Description>
</rdf:RDF>
```

Updating Resources and Resource Features

Enterprise Architect supports a Resource and Resource Feature update service, through which Clients can update the properties of existing Resources and Resource Features of these types via HTTP POST:

- Resources
     - Packages
     - Elements
     - Connectors

- Resource Features
     - Package/Element Discussions, Discussion Replies, Resource Allocations, Tests, Tagged Values, Maintenance Items and Project Management Items
     - Element Attributes, Operations and Operation Parameters

As per the OSLC Specification, the Client has to perform these actions to update an OSLC Resource:

- Fetch the RDF/XML representation of the Resource via HTTP GET

- Update the representation

- Use HTTP PUT to send the new representation to the Resource's URL

However, in Enterprise Architect, to update a Resource or Resource Feature the Client has to POST a representation of the Resource or Resource Feature in RDF XML format to the Update URL. If the POST is successful, the response will contain the URL of the updated Resource. An unsuccessful POST will generate an error response.

These are the steps involved in updating a Resource or Resource Feature in Enterprise Architect:

- The Client fetches the RDF/XML representation of the Resource or Resource Feature to update via HTTP GET

- The Client updates the Resource or Resource Feature properties in the RDF/XML representation

- The Client POSTs the updated RDF/XML representation to the Resource or Resource Feature's Update URL

Refer to the tables in this topic for the information used to update each Resource and Feature. Also see the *GUID Prefix Tables* and *Examples of Resource Update RDF/XML* topics.

Package

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/pu/resource/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	          xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" GUID
	          xmlns:dcterms="http://purl.org/dc/terms/"
	          xmlns:foaf="http://xmlns.com/foaf/0.1/"
	          xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	   <oslc_am:Resource>
	     <!--Required Property-->
	     <dcterms:identifier>GUID OF PACKAGE (TO UPDATE) WITH GUID PREFIX</dcterms:identifier>
	     <!--Required in a Security Enabled Model-->
	     <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	     <!--Updateable Properties-->
	     <dcterms:title>NAME</dcterms:title>

	<dcterms:subject>KEYWORDS</dcterms:subject>
	<dcterms:creator>
	  <foaf:Person>
	    <foaf:name>AUTHOR</foaf:name>
	  </foaf:Person>
	</dcterms:creator>
	<dcterms:description>NOTES</dcterms:description>
	<ss:alias>ALIAS</ss:alias>
	<ss:status>STATUS</ss:status>
	<ss:complexity>COMPLEXITY</ss:complexity>
	<ss:phase>PHASE</ss:phase>
	<ss:version>VERSION</ss:version>
	 </oslc_am:Resource>
	</rdf:RDF>

Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/resource/

Element

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/pu/resource/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	    xmlns:dcterms="http://purl.org/dc/terms/"
	    xmlns:foaf="http://xmlns.com/foaf/0.1/"
	    xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	 <oslc_am:Resource>
	  <!--Required Property-->
	  <dcterms:identifier>GUID OF ELEMENT (TO UPDATE) WITH GUID PREFIX</dcterms:identifier>
	  <!--Required in a Security Enabled Model-->
	  <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	  <!--Updateable Properties-->
	  <dcterms:title>NAME</dcterms:title>
	  <dcterms:subject>KEYWORDS</dcterms:subject>
	  <dcterms:creator>
	    <foaf:Person>
	      <foaf:name>AUTHOR</foaf:name>
	    </foaf:Person>
	  </dcterms:creator>

	<dcterms:description>NOTES</dcterms:description> <ss:alias>ALIAS</ss:alias> <ss:status>STATUS</ss:status> <ss:complexity>COMPLEXITY</ss:complexity> <ss:phase>PHASE</ss:phase> <ss:version>VERSION</ss:version> </oslc_am:Resource> </rdf:RDF>
Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/resource/

Element Attribute

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/pu/attribute/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	      xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	      xmlns:dcterms="http://purl.org/dc/terms/"
	      xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	  <ss:attribute>
	    <!--Required Properties-->
	    <dcterms:identifier>GUID OF ATTRIBUTE (TO UPDATE) WITH GUID PREFIX</dcterms:identifier>
	    <!--Required in a Security Enabled Model-->
	    <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	    <!--Updateable Properties-->
	    <dcterms:title>ATTRIBUTE NAME</dcterms:title>
	    <ss:classifiername>DATATYPE/CLASSIFIER NAME</ss:classifiername>
	    <dcterms:description>NOTES</dcterms:description>
	    <ss:alias>ALIAS</ss:alias>
	    <ss:classifierresourceidentifier>GUID OF THE CLASSIFIER EA ELEMENT WITH PREFIX</ss:classifierresourceidentifier>
	    <ss:defaultvalue>DEFAULT VALUE</ss:defaultvalue>
	    <ss:scope>SCOPE</ss:scope>
	    <ss:containment>CONTAINMENT</ss:containment>
	    <ss:isstatic>IS STATIC</ss:isstatic>
	    <ss:iscollection>IS COLLECTION</ss:iscollection>
	    <ss:isordered>IS ORDERED</ss:isordered>
	    <ss:isconst>IS CONST</ss:isconst>
	    <ss:allowduplicates>ALLOW DUPLICATES</ss:allowduplicates>
	    <ss:lowerbound>NUMERICAL VALUE</ss:lowerbound>

	<ss:upperbound>NUMERICAL VALUE</ss:upperbound> <ss:position>NUMERICAL VALUE</ss:position> </ss:attribute> </rdf:RDF>
Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/attribute/

Element Operation

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/pu/operation/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	    xmlns:dcterms="http://purl.org/dc/terms/"
	    xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	  <ss:operation>
	    <!--Required Property-->
	    <dcterms:identifier>GUID OF OPERATION (TO UPDATE) WITH GUID PREFIX</dcterms:identifier>
	    <!--Required in a Security Enabled Model-->
	    <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	    <!--Updateable Properties-->
	    <dcterms:title>OPERATION NAME</dcterms:title>
	    <dcterms:description>NOTES</dcterms:description>
	    <ss:alias>ALIAS</ss:alias>
	    <ss:classifiername>DATATYPE/CLASSIFIER NAME</ss:classifiername>
	    <ss:classifierresourceidentifier>GUID OF THE CLASSIFIER EA ELEMENT</ss:classifierresourceidentifier>
	    <ss:scope>SCOPE</ss:scope>
	    <ss:isstatic>IS STATIC</ss:isstatic>
	    <ss:isabstract>IS ABSTRACT</ss:isabstract>
	    <ss:isreturnarray>IS RETURN ARRAY</ss:isreturnarray>
	    <ss:isquery>IS QUERY</ss:isquery>
	    <ss:issynchronized>IS SYNCHRONIZED</ss:issynchronized>
	    <ss:isconst>IS CONST</ss:isconst>
	    <ss:ispure>IS PURE</ss:ispure>
	    <ss:position>NUMERICAL VALUE</ss:position>
	    <ss:behaviour>BEHAVIOR</ss:behaviour>
	  </ss:operation>
	</rdf:RDF>

Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/operation/

Element Operation Parameter

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/pu/operationparameter/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	      xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	      xmlns:dcterms="http://purl.org/dc/terms/"
	      xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	  <ss:parameter>
	    <!--Required Properties-->
	    <dcterms:identifier>GUID OF OPERATION PARAMETER (TO UPDATE) WITH GUID PREFIX</dcterms:identifier>
	    <!--Required in a Security Enabled Model-->
	    <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	    <!--Updateable Properties-->
	    <dcterms:title>OPERATION PARAMETER NAME</dcterms:title>
	    <dcterms:description>NOTES</dcterms:description>
	    <ss:classifiername>DATATYPE/CLASSIFIER NAME</ss:classifiername>
	    <ss:classifierresourceidentifier>GUID OF THE CLASSIFIER EA ELEMENT</ss:classifierresourceidentifier>
	    <ss:defaultvalue>DEFAULT VALUE</ss:defaultvalue>
	    <ss:paramdirection>DIRECTION</ss:paramdirection>
	    <ss:position>NUMERICAL VALUE</ss:position>
	  </ss:parameter>
	</rdf:RDF>

Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/operationparameter/

Connector

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/pu/resourcelink/
POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?> <rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"

	xmlns:dcterms="http://purl.org/dc/terms/" xmlns:ss="http://www.sparxsystems.com.au/oslc_am#"> <oslc_am:Resource> <!--Required Properties--> <dcterms:identifier>GUID OF THE SOURCE PACKAGE/ELEMENT WITH GUID PREFIX</dcterms:identifier> <ss:{connector type} rdf:ID="ID" rdf:resource="<protocol>://<server>:<port>/<model>/oslc/am/resource/<GUID OF TARGET PACKAGE/ELEMENT WITH GUID PREFIX>/"/> </oslc_am:Resource> <rdf:Description rdf:about="#ID"> <dcterms:identifier>GUID OF CONNECTOR (TO UPDATE) WITH GUID PREFIX</dcterms:identifier> <!--Required in a Security Enabled Model--> <ss:useridentifier>USER SECURITY GUID</ss:useridentifier> <!--Updateable Properties--> <dcterms:title>CONNECTOR NAME</dcterms:title> <dcterms:description>NOTES</dcterms:description> <ss:direction>DIRECTION</ss:direction> </rdf:Description> </rdf:RDF>
Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/linktype/

Package/Element Discussion

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/pu/discussion/
POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?> <rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:dcterms="http://purl.org/dc/terms/" xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:ss="http://www.sparxsystems.com.au/oslc_am#"> <ss:discussion> <!--Required Property--> <dcterms:identifier>GUID OF DISCUSSION (TO UPDATE) WITH GUID PREFIX</dcterms:identifier> <!--Required in a Security Enabled Model--> <ss:useridentifier>USER SECURITY GUID</ss:useridentifier> <!--Updateable Properties--> <ss:status>DISCUSSION STATUS</ss:status>

	<ss:priority>DISCUSSION PRIORITY</ss:priority>
	</ss:discussion>
	</rdf:RDF>
Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/discussion/

Package/Element Resource Allocation

Type	Input
UML	<protocol>://<server>/<model_name>/oslc/am/pu/resourceallocation/
POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?> <rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:dcterms="http://purl.org/dc/terms/" xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:ss="http://www.sparxsystems.com.au/oslc_am#"> <ss:resourceallocation> <!--Required Property--> <ss:key> <!--Details of Resource Allocation to update--> <rdf:Description> <ss:resourceidentifier>GUID OF CONTAINING PACKAGE/ELEMENT WITH GUID PREFIX </ss:resourceidentifier> <ss:resourcename> <foaf:Person> <foaf:name>AUTHOR</foaf:name> </foaf:Person> </ss:resourcename> <ss:role>ROLE</ss:role> </rdf:Description> </ss:key> <!--Required in a Security Enabled Model--> <ss:useridentifier>USER SECURITY GUID</ss:useridentifier> <!--Updateable Properties--> <ss:resourcename> <foaf:Person> <foaf:name>AUTHOR</foaf:name> </foaf:Person> </ss:resourcename> <ss:role>ROLE</ss:role> <ss:startdate>YYYY-MM-DD format</ss:startdate>

	<ss:enddate>YYYY-MM-DD format</ss:enddate>
	<ss:percentagecomplete>PERCENT COMPLETE</ss:percentagecomplete>
	<ss:expectedtime>EXPECTED TIME</ss:expectedtime>
	<ss:allocatedtime>ALLOCATED TIME</ss:allocatedtime>
	<ss:expendedtime>EXPENDED TIME</ss:expendedtime>
	<dcterms:description>NOTES</dcterms:description>
	<ss:history>HISTORY</ss:history>
	</ss:resourceallocation>
	</rdf:RDF>
Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/resourceallocation/

Package/Element Test

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/pu/test/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	 <rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	     xmlns:dcterms="http://purl.org/dc/terms/"
	     xmlns:foaf="http://xmlns.com/foaf/0.1/"
	     xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	   <ss:test>
	     <!--Required Property-->
	     <ss:key>
	       <!--Details of Test to update-->
	       <rdf:Description>
	         <ss:resourceidentifier>GUID OF CONTAINING PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>
	         <dcterms:title>TEST NAME</dcterms:title>
	         <ss:classtype>CLASS TYPE</ss:classtype>
	       </rdf:Description>
	     </ss:key>
	     <!--Required in a Security Enabled Model-->
	     <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	     <!--Updateable Properties-->
	     <dcterms:title>TEST NAME</dcterms:title>
	     <ss:classtype>CLASS TYPE</ss:classtype>
	     <dcterms:type>TYPE</dcterms:type>
	     <ss:status>STATUS</ss:status>
	     <ss:lastrun>YYYY-MM-DD FORMAT</ss:lastrun>

	<ss:runby>
	  <foaf:Person>
	    <foaf:name>PERSON NAME</foaf:name>
	  </foaf:Person>
	</ss:runby>
	<ss:checkedby>
	  <foaf:Person>
	    <foaf:name>PERSON NAME</foaf:name>
	  </foaf:Person>
	</ss:checkedby>
	<dcterms:description>DESCRIPTION</dcterms:description>
	<ss:input>INPUT</ss:input>
	<ss:acceptancecriteria>ACCEPTANCE CRITERIA</ss:acceptancecriteria>
	<ss:results>RESULTS</ss:results>
	</ss:test>
	</rdf:RDF>

Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/test/

Package/Element Tagged Value

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/pu/taggedvalue/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	    xmlns:dcterms="http://purl.org/dc/terms/"
	    xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	  <ss:taggedvalue>
	    <!--Required Properties-->
	    <dcterms:identifier>GUID OF TAGGED VALUE (TO UPDATE) WITH GUID PREFIX</dcterms:identifier>
	    <!--Required in a Security Enabled Model-->
	    <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	    <!--Updateable Properties-->
	    <dcterms:title>TAG NAME</dcterms:title>
	    <ss:value>TAG VALUE</ss:value>
	    <dcterms:description>TAG NOTES</dcterms:description>
	    <ss:allowduplicates>BOOLEAN VALUE</ss:allowduplicates>
	  </ss:taggedvalue>
	</rdf:RDF>

	Notes:
	'dcterms:description' can contain either Tag Notes or these Predefined Structured Types:
	- Boolean - Color - Const - Date (YYYY-MM-DD format) - Directory - Enum - File - Float - Decimal - Double - Integer - Memo - RefGUID - RefGUIDList - String - Time (HH:MM format) - Timestamp (YYYY-MM-DD HH:MM format) - URL
Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/taggedvalue/

Package/Element Change

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/pu/maintenanceitem/

Post RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	       xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	       xmlns:dcterms="http://purl.org/dc/terms/"
	       xmlns:foaf="http://xmlns.com/foaf/0.1/"
	       xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	  <ss:change>
	    <!--Required Property-->
	    <ss:key>
	      <!--Details of Change to update-->
	      <rdf:Description>
	        <dcterms:title>CHANGE NAME</dcterms:title>
	        <ss:resourceidentifier>GUID OF CONTAINING

	PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier> </rdf:Description> </ss:key> <!--Required in a Security Enabled Model--> <ss:useridentifier>USER SECURITY GUID</ss:useridentifier> <!--Updateable Properties--> <dcterms:title>CHANGE NAME</dcterms:title> <ss:requestedby> <foaf:Person> <foaf:name>PERSON NAME</foaf:name> </foaf:Person> </ss:requestedby> <ss:requestedon>YYYY-MM-DD format</ss:requestedon> <ss:status>New/Verified/Complete</ss:status> <ss:priority>High/Medium/Low</ss:priority> <ss:completedby> <foaf:Person> <foaf:name>PERSON NAME</foaf:name> </foaf:Person> </ss:completedby> <ss:completedon>YYYY-MM-DD format</ss:completedon> <ss:version>VERSION</ss:version> <dcterms:description>NOTES</dcterms:description> <ss:history>HISTORY</ss:history> </ss:change> </rdf:RDF>
Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/change/

Package/Element Defect

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/pu/maintenanceitem/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	    xmlns:dcterms="http://purl.org/dc/terms/"
	    xmlns:foaf="http://xmlns.com/foaf/0.1/"
	    xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	  <ss:defect>
	    <!--Required Property-->

<table>
<tr><td></td><td>

<ss:key>

  <!--Details of Defect to update-->

  <rdf:Description>

    <dcterms:title>DEFECT NAME</dcterms:title>

    <ss:resourceidentifier>GUID OF CONTAINING PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>

  </rdf:Description>

</ss:key>

<!--Required in a Security Enabled Model-->

<ss:useridentifier>USER SECURITY GUID</ss:useridentifier>

<!--Updateable Properties-->

<dcterms:title>DEFECT NAME</dcterms:title>

<ss:reportedby>

  <foaf:Person>

    <foaf:name>PERSON NAME</foaf:name>

  </foaf:Person>

</ss:reportedby>

<ss:reportedon>YYYY-MM-DD format</ss:reportedon>

<ss:status>STATUS</ss:status>

<ss:priority>PRIORITY</ss:priority>

<ss:resolvedby>

  <foaf:Person>

    <foaf:name>PERSON NAME</foaf:name>

  </foaf:Person>

</ss:resolvedby>

<ss:resolvedon>YYYY-MM-DD format</ss:resolvedon>

<ss:version>VERSION</ss:version>

<dcterms:description>NOTES</dcterms:description>

<ss:history>HISTORY</ss:history>

 </ss:defect>

</rdf:RDF>

</td></tr>
<tr><td>Resource Shape</td><td><protocol>://<server>/<model>/oslc/am/rs/defect/</td></tr>
</table>

Package/Element Issue

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/pu/maintenanceitem/
POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?> <rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"

	xmlns:dcterms="http://purl.org/dc/terms/"
	
	xmlns:foaf="http://xmlns.com/foaf/0.1/"
	
	xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	  <ss:issue>
	    <!--Required Property-->
	    <ss:key>
	      <!--Details of Issue to update-->
	      <rdf:Description>
	        <dcterms:title>ISSUE NAME</dcterms:title>
	        <ss:resourceidentifier>GUID OF CONTAINING PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>
	      </rdf:Description>
	    </ss:key>
	    <!--Required in a Security Enabled Model-->
	    <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	    <!--Updateable Properties-->
	    <dcterms:title>ISSUE NAME</dcterms:title>
	    <dcterms:title>ISSUE NAME</dcterms:title>
	    <ss:raisedby>
	      <foaf:Person>
	        <foaf:name>PERSON NAME</foaf:name>
	      </foaf:Person>
	    </ss:raisedby>
	    <ss:raisedon>YYYY-MM-DD format</ss:raisedon>
	    <ss:status>STATUS</ss:status>
	    <ss:priority>PRIORITY</ss:priority>
	    <ss:completedby>
	      <foaf:Person>
	        <foaf:name>PERSON NAME</foaf:name>
	      </foaf:Person>
	    </ss:completedby>
	    <ss:completedon>YYYY-MM-DD format</ss:completedon>
	    <ss:version>VERSION</ss:version>
	    <dcterms:description>NOTES</dcterms:description>
	    <ss:history>HISTORY</ss:history>
	  </ss:issue>
	</rdf:RDF>

Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/issue/

Package/Element Task

Type	Input

URL	`<protocol>://<server>/<model_name>/oslc/am/pu/maintenanceitem/`
| POST RDF/XML | ```xml
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:dcterms="http://purl.org/dc/terms/"
        xmlns:foaf="http://xmlns.com/foaf/0.1/"
        xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
  <ss:task>
    <!--Required Property-->
    <ss:key>
      <!--Details of Task to update-->
      <rdf:Description>
        <dcterms:title>TASK NAME</dcterms:title>
        <ss:resourceidentifier>GUID OF CONTAINING
PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>
      </rdf:Description>
    </ss:key>
    <!--Required in a Security Enabled Model-->
    <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
    <!--Updateable Properties-->
    <dcterms:title>TASK NAME</dcterms:title>
    <ss:requestedby>
      <foaf:Person>
        <foaf:name>PERSON NAME</foaf:name>
      </foaf:Person>
    </ss:requestedby>
    <ss:requestedon>YYYY-MM-DD format</ss:requestedon>
    <ss:status>STATUS</ss:status>
    <ss:priority>PRIORITY</ss:priority>
    <ss:completedby>
      <foaf:Person>
        <foaf:name>PERSON NAME</foaf:name>
      </foaf:Person>
    </ss:completedby>
    <ss:completedon>YYYY-MM-DD format</ss:completedon>
    <ss:version>VERSION</ss:version>
    <dcterms:description>NOTES</dcterms:description>
    <ss:history>HISTORY</ss:history>
  </ss:task>
</rdf:RDF>
``` |
| Resource Shape | `<protocol>://<server>/<model>/oslc/am/rs/task/` |

Package/Element Event

Type	Input
URL	`<protocol>://<server>/<model_name>/oslc/am/pu/maintenanceitem/`

POST RDF/XML	`<?xml version="1.0" encoding="UTF-8"?>`
	`<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"`
	    `xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"`
	    `xmlns:dcterms="http://purl.org/dc/terms/"`
	    `xmlns:foaf="http://xmlns.com/foaf/0.1/"`
	    `xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">`
	  `<ss:event>`
	    `<!--Required Property-->`
	    `<ss:key>`
	      `<!--Details of Task to update-->`
	      `<rdf:Description>`
	        `<dcterms:title>EVENT NAME</dcterms:title>`
	        `<ss:resourceidentifier>GUID OF CONTAINING PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>`
	      `</rdf:Description>`
	    `</ss:key>`
	    `<!--Required in a Security Enabled Model-->`
	    `<ss:useridentifier>USER SECURITY GUID</ss:useridentifier>`
	    `<!--Updateable Properties-->`
	    `<dcterms:title>EVENT NAME</dcterms:title>`
	    `<ss:reportedby>`
	      `<foaf:Person>`
	        `<foaf:name>PERSON NAME</foaf:name>`
	      `</foaf:Person>`
	    `</ss:reportedby>`
	    `<ss:requestedon>YYYY-MM-DD format</ss:requestedon>`
	    `<dcterms:type>TYPE</dcterms:type>`
	    `<ss:priority>PRIORITY</ss:priority>`
	    `<ss:resolvedby>`
	      `<foaf:Person>`
	        `<foaf:name>PERSON NAME</foaf:name>`
	      `</foaf:Person>`
	    `</ss:resolvedby>`
	    `<ss:resolvedon>YYYY-MM-DD format</ss:resolvedon>`
	    `<ss:version>VERSION</ss:version>`
	    `<dcterms:description>NOTES</dcterms:description>`
	    `<ss:history>HISTORY</ss:history>`
	  `</ss:event>`

	</rdf:RDF>
Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/event/

Package/Element Decision

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/pu/maintenanceitem/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	        xmlns:dcterms="http://purl.org/dc/terms/"
	        xmlns:foaf="http://xmlns.com/foaf/0.1/"
	        xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	  <ss:decision>
	    <!--Required Property-->
	    <ss:key>
	      <!--Details of Task to update-->
	      <rdf:Description>
	        <dcterms:title>DECISION NAME</dcterms:title>
	        <ss:resourceidentifier>GUID OF CONTAINING PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>
	      </rdf:Description>
	    </ss:key>
	    <!--Required in a Security Enabled Model-->
	    <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	    <!--Updateable Properties-->
	    <dcterms:title>DECISION NAME</dcterms:title>
	    <ss:owner>
	      <foaf:Person>
	        <foaf:name>PERSON NAME</foaf:name>
	      </foaf:Person>
	    </ss:owner>
	    <ss:date>YYYY-MM-DD format</ss:date>
	    <ss:status>STATUS</ss:status>
	    <ss:impact>IMPACT</ss:impact>
	    <ss:author>
	      <foaf:Person>
	        <foaf:name>PERSON NAME</foaf:name>
	      </foaf:Person>
	    </ss:author>

	<ss:effective>YYYY-MM-DD format</ss:effective>
	<ss:version>VERSION</ss:version>
	<dcterms:description>NOTES</dcterms:description>
	<ss:history>HISTORY</ss:history>
	</ss:decision>
	</rdf:RDF>
Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/decision/

Package/Element Effort

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/pu/projectmanagementitem/

POST RDF/XML	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	     xmlns:dcterms="http://purl.org/dc/terms/"
	     xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	  <ss:effort>
	    <!--Required Property-->
	    <ss:key>
	      <!--Details of Effort to update-->
	      <rdf:Description>
	        <dcterms:title>EFFORT NAME</dcterms:title>
	        <ss:resourceidentifier>GUID OF CONTAINING PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>
	      </rdf:Description>
	    </ss:key>
	    <!--Required in a Security Enabled Model-->
	    <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	    <!--Updateable Properties-->
	    <dcterms:title>EFFORT NAME</dcterms:title>
	    <dcterms:type>TYPE</dcterms:type>
	    <ss:time>NUMERICAL VALUE</ss:time>
	    <dcterms:description>NOTES</dcterms:description>
	  </ss:effort>
	</rdf:RDF>

Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/effort/

Package/Element Risk

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/pu/projectmanagementitem/

POST RDF/XML	```xml
	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	        xmlns:dcterms="http://purl.org/dc/terms/"
	        xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	  <ss:risk>
	     <!--Required Property-->
	       <ss:key>
	          <!--Details of Risk to update-->
	          <rdf:Description>
	             <dcterms:title>RISK NAME</dcterms:title>
	             <ss:resourceidentifier>GUID OF CONTAINING
	PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>
	          </rdf:Description>
	       </ss:key>
	     <!--Required in a Security Enabled Model-->
	     <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	     <!--Updateable Properties-->
	     <dcterms:title>RISK NAME</dcterms:title>
	     <dcterms:type>RISK TYPE</dcterms:type>
	     <ss:weight>NUMERICAL VALUE</ss:weight>
	     <dcterms:description>NOTES</dcterms:description>
	  </ss:risk>
	</rdf:RDF>
	``

Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/risk/

Package/Element Metric

Type	Input
URL	<protocol>://<server>/<model_name>/oslc/am/pu/projectmanagementitem/

POST RDF/XML	```xml
	<?xml version="1.0" encoding="UTF-8"?>
	<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
	        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
	        xmlns:dcterms="http://purl.org/dc/terms/"
	``

	xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
	
	<ss:metric>
	
	  <!--Required Property-->
	
	  <ss:key>
	
	    <!--Details of Metric to update-->
	
	    <rdf:Description>
	
	      <dcterms:title>METRIC NAME</dcterms:title>
	
	      <ss:resourceidentifier>GUID OF CONTAINING PACKAGE/ELEMENT WITH GUID PREFIX</ss:resourceidentifier>
	
	    </rdf:Description>
	
	  </ss:key>
	
	  <!--Required in a Security Enabled Model-->
	
	  <ss:useridentifier>USER SECURITY GUID</ss:useridentifier>
	
	  <!--Updateable Properties-->
	
	  <dcterms:title>METRIC NAME</dcterms:title>
	
	  <dcterms:type>METRIC TYPE</dcterms:type>
	
	  <ss:weight>NUMERICAL VALUE</ss:weight>
	
	  <dcterms:description>NOTES</dcterms:description>
	
	 </ss:metric>
	
	</rdf:RDF>

Resource Shape	<protocol>://<server>/<model>/oslc/am/rs/metric/

Notes

- This facility is available through the Enterprise Architect Pro Cloud Server

- In a Security Enabled model, you need to provide the User Security GUID (via the 'ss:useridentifier' XML element) in the POST XML to create a Resource or Resource Feature

- OSLC Specification uses HTTP PUT to update Resource Properties whereas Enterprise Architect uses HTTP POST for updating Resource Properties

Learn more

- GUID Prefix Tables
- Examples of Resource Update RDF/XML
- Packages
- UML Elements
- Domain Models
- Attributes
- Operations
- Define Parameters
- UML Connectors
- Informal Element Discussions

- [Formal Review Elements](#)
- [Resource Allocation](#)
- [Create Test Records](#)
- [Tagged Values](#)
- [Predefined Structured Types](#)
- [Create Maintenance Items](#)
- [Effort Management](#)
- [Risk Management](#)
- [Metrics](#)

Examples of Resource Update RDF/XML

This topic provides two examples of the RDF/XML representation of Resource Update XML.

Update the name of an element

Update the name of the Actor 'Actor1' (with the Enterprise Architect GUID {567854C2-6A89-46d7-AB7E-F192668010CA}) to 'Actor 1_New'.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:dcterms="http://purl.org/dc/terms/"
        xmlns:foaf="http://xmlns.com/foaf/0.1/"
        xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
   <oslc_am:Resource>
     <!--Required Properties-->
     <dcterms:identifier>el_{567854C2-6A89-46d7-AB7E-F192668010CA}</dcterms:identifier>
     <!--Updateable Properties-->
     <dcterms:title>Actor1_New</dcterms:identifier>
   </oslc_am:Resource>
</rdf:RDF>
```

Update the Properties of a Resource Allocation

In the Actor 'Actor1' (with the Enterprise Architect GUID {567854C2-6A89-46d7-AB7E-F192668010CA}), on a Resource Allocation with the name 'Sparx' and the Role 'Business Analyst', update the Start Date, End Date and Percent Complete.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:oslc_am="http://open-services.net/ns/am#"
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:dcterms="http://purl.org/dc/terms/"
        xmlns:foaf="http://xmlns.com/foaf/0.1/"
        xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
   <ss:resourceallocation>
     <!--Required Property-->
     <ss:key>
       <rdf:Description>
         <ss:resourceidentifier>el_{567854C2-6A89-46d7-AB7E-F192668010CA}</ss:resourceidentifier>
         <ss:resourcename>
            <foaf:Person>
```

```
                    <foaf:name>Sparx</foaf:name>
                </foaf:Person>
            </ss:resourcename>
            <ss:role>Business Analyst</ss:role>
        </rdf:Description>
    </ss:key>
    <!--Updateable Properties-->
    <ss:startdate>2017-05-01</ss:startdate>
    <ss:enddate>2017-05-15</ss:enddate>
    <ss:percentagecomplete>0</ss:percentagecomplete>
  </ss:resourceallocation>
</rdf:RDF>
```

Query Capability

Clients can query a model and retrieve the Resources that match a specific criteria. In Enterprise Architect, the base URI for accessing the Query Capability is:

<protocol>://<server>/<model_name>/oslc/am/qc/

A query string expressing the specific criteria should be added to the base URI and addressed to the model using an HTTP GET request. The response for this request will be in RDF/XML format. Currently, Enterprise Architect supports querying on Package, element and diagram properties only (and not on its features such as attributes, operations and scenarios).

Enterprise Architect supports these Query Capability parameters for specifying the criteria:

- oslc.where - specifies the conditions that the Resources must satisfy and is similar to the WHERE clause of a SQL statement

- oslc.select - specifies the Resource properties to be retrieved and is similar to the SELECT clause of a SQL statement

These parameters can be used separately or in combination to specify the criteria for retrieving the Resources.

Example

Retrieve all Resources of type Actor in the model:

- URL : http://localhost:480/firebird_model/oslc/am/qc/?oslc.where=dcterms:type="Actor"

The response is as shown.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
         xmlns:oslc_am="http://open-services.net/ns/am#"
         xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:dcterms="http://purl.org/dc/terms/"
         xmlns:foaf="http://xmlns.com/foaf/0.1/"
         xmlns:ss="http://www.sparxsystems.com.au/oslc_am#">
    <rdf:Description rdf:about="http://mybuildvm:2480/firebird_model/oslc/am/qc/">
        <rdfs:member>
            <oslc_am:Resource rdf:about="http://localhost:480/firebird_model/oslc/am/resource/el_{CE11BBAF-D7C5-416e-8C4A-8C1B1619C4D5}/">
                <rdf:type rdf:resource="http://open-services.net/ns/am#Resource"/>
                <!--Name of the Resource-->
                <dcterms:title>A1</dcterms:title>
                <!--Enterprise Architect Type for the Resource-->
                <dcterms:type>Actor</dcterms:type>
                <!--GUID of the Resource-->
                <dcterms:identifier>el_{CE11BBAF-D7C5-416e-8C4A-8C1B1619C4D5}</dcterms:identifier>
                <!--Author of the Resource-->
                <dcterms:creator>
                    <foaf:Person>
                        <foaf:name>Sparx</foaf:name>
                    </foaf:Person>
                </dcterms:creator>
                <!--Created Date of the Resource-->
                <dcterms:created>2017-05-01 09:24:16</dcterms:created>
                <!--Modified Date of the Resource-->
                <dcterms:modified>2017-05-01 09:24:16</dcterms:modified>
                <!--Status of the Resource-->
                <ss:status>Proposed</ss:status>
                <!--Complexity of the Resource-->
                <ss:complexity>Easy</ss:complexity>
                <!--Phase of the Resource-->
                <ss:phase>1.0</ss:phase>
                <!--Version of the Resource-->
                <ss:version>1.0</ss:version>
                <!--Type of the Resource-->
                <ss:resourcetype>Element</ss:resourcetype>
                <!--GUID of the Package that contains this Actor-->
                <ss:parentresourceidentifier>pk_{CBCDC240-73C2-4591-8C88-271143B6E887}</ss:parentresourceidentifier>
                <!--Use this link to retrieve the Features of this Actor-->
                <ss:features rdf:resource="http://localhost:480/firebird_model/oslc/am/features/el_{CE11BBAF-D7C5-416e-8C4A-8C1B1619C4D5}/"/>
                <!--Use this link to retrieve the Connectors of this Actor-->
                <ss:linkedresources rdf:resource="http://localhost:480/firebird_model/oslc/am/linkedresources/el_{CE11BBAF-D7C5-416e-8C4A-8C1B1619C4D5}/"/>
            </oslc_am:Resource>
        </rdfs:member>
    </rdf:Description>
</rdf:RDF>
```

Elements of the Response XML

Element	Description
oslc_am:Resource	Represents an OSLC Architecture Management 2.0 Resource and its properties (as defined in the Resource Shape). The attribute 'rdf:about' on this element specifies the URL for accessing the Resource. The format for this URL is: <protocol>://<server>/<model_name>/oslc/am/resource/<Resource GUID With Prefix>/
Elements with namespace prefix 'dcterms'	The standard properties from the Dublin Core Metadata Element Set.
Elements with namespace prefix 'ss'	The custom Enterprise Architect properties.

Notes

- This facility is available through the Enterprise Architect Pro Cloud Server
- In a Security Enabled model, you need to provide the User Security GUID (via the 'useridentifier' query string in the URL) in the GET request to retrieve the Resource/Resource Feature XML

oslc.where Parameter

[This topic under construction.]

The 'oslc.where' Query parameter specifies the conditions that the resources must satisfy; it is similar to the WHERE clause of an SQL statement. This parameter must be appended to the base URI of the Query Capability in Enterprise Architect as:

<protocol>://<server>/<model_name>/oslc/am/qc/?oslc.where=<conditions>

The syntax for the 'oslc.where' Query parameter (defined in BNF grammar and as specified in the OSLC Query Syntax Specification) is:

```
oslc_where     ::= "oslc.where=" compound_term
compound_term ::= simple_term (space? boolean_op space? simple_term)*
simple_term    ::= term | scoped_term
space          ::= " " /* a space character */
boolean_op     ::= "and"
term           ::= identifier_wc comparison_op value | identifier_wc space in_op space? in_val
scoped_term    ::= identifier_wc "{" compound_term "}"
identifier_wc ::= identifier | wildcard
identifier     ::= PrefixedName
PrefixedName  ::= /* see "SPARQL Query Language for RDF", http://www.w3.org/TR/rdf-sparql-query/#rPrefixedName */
wildcard       ::= "*"
comparison_op ::= "=" | "!=" | "<" | ">" | "<=" | ">="
in_op          ::= "in"
in_val         ::= "[" value ("," value)* "]"
value          ::= uri_ref_esc | literal_value
uri_ref_esc    ::= /* an angle bracket-delimited URI reference in which > and \ are \-escaped. */
literal_value ::= boolean | decimal | string_esc (LANGTAG | ("^^" PrefixedName))?
boolean        ::= "true" | "false"
decimal        ::= /* see "XML Schema Part 2: Datatypes Second Edition", http://www.w3.org/TR/xmlschema-2/ */
string_esc     ::= /* a string enclosed in double quotes, with certain characters escaped. See below. */
LANGTAG        ::= /* see "SPARQL Query Language for RDF", http://www.w3.org/TR/rdf-sparql-query/#rLANGTAG */
```

The Resource properties that can be queried are:

- dcterms:title
- dcterms:type
- dcterms:identifier
- dcterms:subject
- dcterms:creator{foaf:name} / dcterms:creator{*}
- dcterms:created
- dcterms:modified
- dcterms:description
- ss:status
- ss:complexity
- ss:stereotype{ss:name} / ss:stereotype{ss:fqname}
- ss:phase
- ss:version
- ss:parentresourceidentifier
- ss:resourcetype
- ss:linkedresources

Examples

Example	Query
1	http://localhost:480/firebird_model/oslc/am/qc/?oslc.where=dcterms:title="Class1" Retrieves all Resources with the name of 'Class1'.
2	http://localhost:480/firebird_model/oslc/am/qc/?oslc.where=dcterms:title="Class1" and dcterms:type="Class" Retrieves all Enterprise Architect 'Class' type Resources with the name of 'Class1'.
3	http://localhost:480/firebird_model/oslc/am/qc/?oslc.where=dcterms:type="Activity" and dcterms:created > "2017-05-01" or http://localhost:480/firebird_model/oslc/am/qc/?oslc.where=ss:resourcetype ="Element" and dcterms:type="Activity" and dcterms:created > "2017-05-01" Retrieves all Enterprise Architect Activity type Resources that were created after 2017-05-01.
4	http://localhost:480/firebird_model/oslc/am/qc/?oslc.where=dcterms:creator{foaf:name}="Sparx" Alternatives: • http://localhost:480/firebird_model/oslc/am/qc/?oslc.where=dcterms:creator{foaf:name="Sparx"} • http://localhost:480/firebird_model/oslc/am/qc/?oslc.where=dcterms:creator{*}="Sparx" • http://localhost:480/firebird_model/oslc/am/qc/?oslc.where=dcterms:creator{*="Sparx"} Retrieves all Resources created by Author 'Sparx'.
5	http://localhost:480/firebird_model/oslc/am/qc/?oslc.where=dcterms:creator{foaf:name}in ["Sparx",”SpxSys”] Alternatives: • http://localhost:480/firebird_model/oslc/am/qc/?oslc.where=dcterms:creator{foaf:name in ["Sparx",”SpxSys”]} • http://localhost:480/firebird_model/oslc/am/qc/?oslc.where=dcterms:creator{*} in ["Sparx",”SpxSys”] • http://localhost:480/firebird_model/oslc/am/qc/?oslc.where=dcterms:creator{* in ["Sparx",”SpxSys”]}

	Retrieves all Resources created by Author 'Sparx' or 'SpxSys'.

WebEA Quick Start Guide

The purpose of the WebEA Quick Start Guide is to provide the complete process for quickly installing and configuring WebEA in a new environment using the most common options.

The process is based on these assumptions:

- A clean Windows 7 (or later) based machine has been selected to host both the Pro Cloud Server and Web Server; that is, no other Web Servers or applications are configured to listen on port 80

- The latest version of XAMPP will be used to install Apache as the Web Server

- The Small Business edition of the Pro Cloud Server is being installed

- WebEA will access one brand new Firebird model (because it is the simplest, as it does not require any third party ODBC drivers)

1. Download the Pro Cloud Server

Step	Description
1.1	When you purchase the Sparx Systems Pro Cloud Server the sales team will send you a link to download the installer.

2. Install the Web Server

Step	Description
2.1	XAMPP is available at https://www.apachefriends.org/download.html. Download the latest version for Windows, to the machine's desktop.
2.2	The XAMPP installer should be run with Administration privileges; on the desktop, right-click on the XAMPP installer icon and select 'Run as Administrator'.
2.3	Proceed through the installation process, accepting the defaults until the Select Components screen displays. Deselect these checkboxes: MySQL, FileZilla, Mercury, Tomcat, Perl, phpMyAdmin, Webalizer and Fake Sendmail. WebEA only requires the minimum XAMPP installation of Apache and PHP. Click on the Next button.
2.4	The installation folder should be set to C:\xampp. Click on the Next button until the installation starts. When the installation finishes, select the option to open the Control Panel, then complete the installation.

3. Configure the Web Server

With the default XAMPP configuration, the default website will use port 80. Its base folder is C:\xampp\htdocs.

Step	Description

| 3.1 | Using the XAMPP control panel, start Apache. No errors should be reported at this stage. |

4. Install Pro Cloud Server

Step	Description
4.1	The Pro Cloud Server installer is a standard MSI; double-click on it to start the installation. Note: Depending on a number of factors, you might be prompted for the credentials of a user with elevated permissions; these must be entered before the installation will continue.
4.2	The 'Sparx Systems Pro Cloud Services Setup' dialog displays. Click on the Next button.
4.3	Read the license agreement and select the checkbox for indicating that you accept the terms. Click on the Next button.
4.4	Read the information about the Pro Cloud Server. Click on the Next button.
4.5	The 'Custom Setup' page displays. Click on the Next button to accept the defaults.
4.6	Click on the Install button to run the installation.

5. Configure the Pro Cloud Server

The Pro Cloud Server will have installed all of its files to the 'C:\Program Files (x86)\Sparx Systems\Cloud Services\' folder.

Step	Description
5.1	Open your text editor of choice with 'Administrator' privileges, then open the file 'C:\Program Files (x86)\Sparx Systems\Cloud Services\Service\SSCloudServices.config'.
5.2	Near the top of the file, locate the option "LICENSE=" and add the license key that was provided in purchase confirmation email. The completed field should have this format: LICENSE={00000000-0000-0000-0000-000000000000} Note: The SERVER_PORT option is set to 803; we will need this later.
5.3	Scan down the file looking for the first open parentheses (around line 28) and look for "SERVER_PORT=804". On the next line add: "OSLC_SUPPORT=1" This server port is the HTTP port that the Cloud Service will listen on, so this is the port that both WebEA and Enterprise Architect will use.
5.4	Save the changes to the SSCloudService.config file and close it.
	Open the standard Windows Services console (command "services.msc"), locate and select the service

5.5	"Sparx Systems Professional Cloud", then restart it.
5.6	Navigate to the folder C:\Program Files (x86)\Sparx Systems\Cloud Services\Client and run the SSProCloudClient.exe application.
	Enter the Server address of "localhost:803" and click on the OK button.
5.7	Click on the Add button and type the name 'fb-procloud.fdb' in the connection string field, then click on the OK button.
	Select the new entry in the list and click on the Configure button.
	On the Configure Database Manager screen, select the 'Enabled' and 'Enable extended OSLC interface for WebEA' checkboxes.
	Click on the Save button.
	For a more in depth explanation of how to add database managers, see the Help topic *Add Database Manager*.

6. Copy WebEA files to Website

Step	Description
6.1	Copy the folder C:\Program Files (x86)\Sparx Systems\Cloud Services\WebEA\ to C:\xampp\htdocs.

7. Configure WebEA to access the model

To give a better demonstration of the WebEA options, we will define three different WebEA model connections. All will connect to the same physical Firebird database.

Each model connection will have its own set of options that control how users will be able to interact with the data.

Step	Description
7.1	Open the file 'C:\xampp\htdocs\WebEA\includes\webea_config.ini' with a text editor.
7.2	In the "[model_list]" section, remove the 'model4' and 'model5' lines. Edit the remaining model names to:
	model1 = "Pro Cloud Model (no updates)"
	model2 = "Pro Cloud Model with discussions"
	model3 = "Pro Cloud Model with access code"
7.3	In the "[model1 : settings]" section change the SSCS_DB_ALIAS option to "fb-procloud". The remaining default options should remain unchanged.
7.4	In the "[model2 : settings]" section change the SSCS_DB_ALIAS option to "fb-procloud". The remaining default options should remain unchanged.
7.5	Delete the "[model3 : settings]" and "[model5 : settings]" sections, rename the "[model4 : settings] section header to [model3 : settings]", then set these options as shown:

	sscs_port = "804"
	sscs_db_alias = "fb-procloud"
	sscs_port = "804"
	sscs_use_ssl = "false"
	sscs_model_user = ""
	ssc_model_pwd = ""
	auth_code = "password01".
	login_prompt = "false"
7.6	Save the changes to the file webea_config.ini in C:\xampp\htdocs\WebEA\includes.

8. Configure your firewall(s) to allow client access

Step	Description
8.1	Firewall rules will have to be adjusted to allow traffic between the [Client] and [Web Server], on ports 80 and 443.
8.2	Firewall rules will have to be adjusted to allow traffic between the [Web Server] and [Pro Cloud Server], on ports 804 and 805.

9. Use Enterprise Architect to add content

The initial Firebird model we created earlier will be empty. The easiest way to add content to the model is with Enterprise Architect.

Step	Description
9.1	On a client machine with access to the web server, open Enterprise Architect v12 or later.
9.2	From the main menu select 'Open Project...' or, from the ribbon, select the '⊗ ▾ > Open Project' option.
9.3	Click on the Connect to Cloud button.
9.4	On the 'Cloud Connection' screen set these parameters: Name: fb-procloud Protocol: http:// Server: {your servername} Port: 804 Model Name: fb-procloud Click on the OK button.

Step	Description
9.5	Create a sample set of Model Views, Packages, diagrams and elements, so that they can be viewed in WebEA later.

10. Configure automatic viewable components

In a traditional Enterprise Architect model, saved diagrams and linked documents are not stored in a viewable format that browsers can use, so in order for WebEA to view them Enterprise Architect has to be configured to save HTML compatible versions of these components each time they are updated. The Pro Cloud Server includes a 'worker' application that can be automated to save HTML compatible versions of diagrams and linked documents.

Step	Description
10.1	On the Web Server machine, navigate to the folder C:\Program Files (x86)\Sparx Systems\Cloud Services\.
10.2	Create a new 'Scripts' folder and open it.
10.3	Create a new text file and name it 'fbprocloud.eap'. Notice the .EAP extension; this is important.
10.4	Open the file 'fbprocloud.eap' with a text editor, and paste the following text on a single line (that is, without any carriage returns): EAConnectString:fbprocloud ---;Connect=Cloud=protocol:http,address:localhost,port:804;Data Source=fb-procloud; Save the changes and close the file, but leave the text editor open - we will need it for the next step.
10.5	Using the text editor, create a new file called 'autogen_fbprocloud.cmd' (in the same folder as for the previous step) with the contents: "C:\Program Files (x86)\Sparx Systems\Cloud Services\Service\Worker\EA.exe"/cloudupdate:30 "fbprocloud.eap" Note: All text should be entered on a single line. Save the changes and close the file.
10.6	Double click on the 'autogen_fbprocloud.cmd' file to run it. Note: EA Worker does not provide any feedback to the console as it runs; the command window will remain open if everything is correct, but if it closes a problem has occurred while the Worker was trying to make a connection to the model. If this occurs, double check all your settings. Sparx Systems recommends that this batch/command file is added to Startup so that it runs automatically when the server starts. For a more in-depth explanation of how to configure automatic viewable components, see the Help topic *How to configure automatic viewable components*.

11. Test WebEA access

Step	Description
11.1	On a client machine, open an internet browser and enter the URL http://{yourwebserver}/WebEA/login.php, where the value '{yourwebserver}' is either the name or IP

		number of the machine hosting the WebEA website. The WebEA login screen displays, listing the three model connections we defined earlier: • Pro Cloud Model (no updates) • Pro Cloud Model with discussions • Pro Cloud Model with access code
	11.2	Select the first model connection - Pro Cloud Model (no updates) - and click on the Next button. This connects you to the model and displays the model views that are defined. From here you should be able to navigate through the model by selecting Packages, diagrams and elements. When you select an element, its complete details are displayed. Because of the way this model connection is defined, the bottom of the properties section will NOT contain the 'Discussions' section.
	11.3	Click on the 'hamburger' icon in the top right corner of the display, and select the 'Logout' option. Then select the second model connection - Pro Cloud Model with discussions - and click on the Next button. You should see the same model views as before; now navigate to an element and confirm that the 'Discussions' section is visible.
	11.4	Once again, click on the 'hamburger' icon in the top right corner, and select the 'Logout' option. Select the third model connection, 'Pro Cloud Model with access code' and click on the Next button. This time you should be presented with a prompt for the Access Code; enter 'password01' (as we defined in step 7.5) and click on the Login button. You should see the same model views as before. Navigate to an element; this time there should be a blue 'hamburger' menu in the element details area that allows the creation of element features such as tests and resource allocations. There should also be a pencil icon next to the element notes area that, when clicked on, will allow the modification of element notes.

Complete

This quick start procedure is now complete. It has shown you how to install and configure a basic installation of WebEA. The Sparx System's online Help documentation has more detailed instructions and explains all possible options that can be configured. See the section '*Model Repository | Cloud Based Repositories | Pro Cloud Server*' in the Enterprise Architect online Help.

Reuseable Asset Service (RAS)

Within a large organization, groups of users can be separated by geographical distance be on different networks. This can make it difficult to share common data, standards and modeling structures, without the complexity of using external version control tools or manually distributing XMI files between projects. Enterprise Architect has a Reusable Asset Service (RAS) that provides a simple and convenient mechanism for modelers to distribute or download reusable model structures, information, corporate directives and standards. The shared repository is accessible via a Cloud Service connection, making it accessible for distributed teams. The person who sets up the reusable data can retain governance of the resource or asset, whilst users of the service can easily review the currency of the information and download the latest versions into their models or file folders.

The RAS gives distributed teams convenient access to a single 'source of truth' for shared data, including project milestones, architectural frameworks and industry standards.

Reusable Assets

Reusable Assets consist of:

- Packages containing elements, diagrams and structures (as drawn from any point within the Project Browser), and

- Files in a range of text, code and graphic formats, including .eap files

An asset can be, for example:

- A common Class library or framework

- A set of common Requirements or Use Cases

- A draft specification document

- Marketing collateral

Storage Structure and Use

Reusable Assets are held in a remote registry, accessed through a Cloud Service connection. The registry contains any number of Storages, which can either be left unprotected for any user to create, or change-restricted by an Administrator password. Each Storage can contain any number of Asset Packages holding modeling structures, and files containing textual or graphical information and data. When a user creates a Storage, they can protect the contents from being updated in the Registry or downloaded into a model, using password protection defined by that user.

For each Package, the RAS automatically identifies the:

- Version of the Package held in the registry

- Diagrams and elements (including child Package elements) contained by the Asset Package

- Dependencies on parent Packages of any external elements that the Asset Package references

- Dependencies on MDG Technologies

Any user, regardless of password protection, can freely browse and identify the contents of a Package held in a Storage - including displaying the diagrams - without importing the material into a model. A user that has imported an Asset Package into their model can compare their model Package against any version of the Asset Package, to check for and assess any differences between them.

Notes

- The Reusable Asset Service is available in the Corporate, Business and Software Engineering, System Engineering and Ultimate editions of Enterprise Architect

Connect to the Asset Service

The Reusable Asset Service provides model structures and documents to you from one or more registries on remote systems. You access these registries through a Cloud connection to the appropriate server. The connection details should be provided to you by your System Administrator or Reusable Asset Service Administrator.

Access

Ribbon	Publish > Model Exchange > Reusable Assets

Connect to Assets

Option	Detail
Registry	This field normally defaults to the last-used registry name. If the field is empty, or if you want to switch to a different registry, click on the [...] button to the right of the field.
	The 'Cloud Connection' dialog displays.
	Using the information provided by your administrator, type in the server name, URL and the name of the model hosting the registry data. For example, if you wanted to download the UBL Frameworks into your model, you would specify the server 'sparxcloud.com' and the model name 'modelpatterns'.
	If advised by your Administrator, type in the user name and password for the server.
	Click on the OK button; the 'Registry' field now shows the server name.
Storage	If there is more than one Storage available in the Registry, this field defaults to the first listed in the Registry.
	To check for or select other Storages, click on the drop-down arrow at the right of the field. Click on the Storage to review. For example, if you were downloading UBL frameworks, you would select 'UBL' in the list.
	The 'Registry Browser' tab displays the Packages registered in the selected Storage.

Notes

* The Reusable Asset Service is available in the Corporate, Business and Software Engineering, System Engineering and Ultimate editions of Enterprise Architect

Browse Assets

When you select a Storage in the Reusable Asset Service, the 'Registry Browser' tab immediately displays a list of the Packages registered in that Storage. You can review and use these Packages using the context menu and buttons on the tab.

You can review the properties and contents of the selected Package in the Storage, and access any files in the Storage, using the tabs in the lower half of the Reusable Asset Service view, underneath the 'Registry Browser' tab.

If you have left the Registry Browser open for a while and there is a possibility that the Registry has been changed, you can click on the [icon] icon in the Reusable Asset Service toolbar to refresh the Browser with the latest contents of the Registry.

Access

Ribbon	Publish > Model Exchange > Reusable Assets > Registry Browser

Review Assets

Option	Detail
Toggle Filter Bar	Right-click on the column headings and select this option to show or hide the Filter Bar on the display.
<filter bar fields>	Type in the appropriate characters to list only Packages that have that string of characters in the values in the corresponding column.
Package	This column shows the name of the Packages held in the selected Storage.
Version	This column defaults to the most recent version of each Package held in the selected Storage. You can click on the drop-down arrow at the end of a field and select a different version of the corresponding Package, if any have been registered.
Last Registered	This column shows the date and time at which the currently-listed version of each Package was registered.
Registered By	This column shows the user name of the person who registered the currently-listed version of each Package.
Import	Click on a Package name and click on this button to begin to import either the Package alone or the Package and its dependent structures into your model. You can perform the same action by right-clicking on the Package name and selecting the 'Import from Registry' menu option. If the Package from the Registry already exists in your model, it is deleted and replaced by this import. For a:

	• 'Learning Center Library' Storage, the 'Download Library' dialog displays • 'Reference Data Library' Storage, the 'Import Reference Data' dialog displays • 'Source Code Library' Storage, the 'Save As' dialog displays
Compare with Package in Model	If you have previously imported a Package from the Registry into your model, you can right-click on the Package name in the Registry Browser and select this option to compare the imported Package with the Registry version, and show any differences. The system automatically selects the model Package from the Project Browser.
Find in Project Browser	If you have previously imported a Package from the Registry into your model, you can right-click on the Package name in the Registry Browser and select this option to highlight the corresponding Package in the Project Browser.

Notes

- Some of the other options available on the Registry Browser are used to set up assets in the Registry
- If the Storage is password-protected, a prompt for you to enter a password displays when you select to perform an operation on the data; in this situation, you need the 'Read-only' password to process the information from the Storage, and the 'All-access' password to change the information in the Storage
- The 'Registry Browser' tab will be empty for a 'Learning Center Library', 'Reference Data Library' or 'Source Code Library' type Storage - the contents of these Storages will be available in the 'Storage Files' tab of the Reusable Asset Service view

Asset Properties

When you select a Storage and a Package in the Registry Browser, the first four tabs in the lower half of the Reusable Asset Service view are updated with information from the Package. The 'Asset Properties' tab is a read-only view of the properties of the selected Package itself.

Access

Open the Reusable Asset Service window using one of the methods outlined here.

Select a Package, then click on the 'Asset Properties' tab to display properties for that Package.

Ribbon	Publish > Model Exchange > Reusable Assets

Review Package Properties

Field	Detail
Package	Displays the name of the selected Package.
GUID	Displays the Global Unique Identifier of the Package.
Version	Displays the version reference (such as 1.1 or Beta) of the Package.
Author	Displays the user ID of the person who created the Package (the Package Author).
Created Date	Displays the date and time at which the selected version of the Package was created (that is, registered) in the Registry.
Modified Date	Displays the date and time at which the selected version of the Package was last modified (that is, re-registered) in the Registry.
Comments	Displays any comment recorded against the Package when it was registered in the Storage.
Notes	Displays the notes recorded against the Package when it was registered in the Storage.

Notes

- The 'Asset Properties' tab will be empty for a 'Learning Center Library', 'Reference Data Library' and 'Source Code Library' type Storage

Package Contents

When you select a Storage and a Package in the 'Registry Browser', the first four tabs in the lower half of the Reusable Asset Service view are updated with information from the Package. The 'Contents' tab lists the diagrams and elements (including child Packages) held in the selected Package, listing the two types of object separately. You can organize the information within a column into alphabetical or reverse-alphabetical order for ease of reference, and use the Filter bar to filter the display to show only items with values containing specific characters or digits.

Access

Open the Reusable Asset Service window using one of the methods outlined here.

Select a Package, then click on the 'Contents' tab, to display the contents of that Package.

Ribbon	Publish > Model Exchange > Reusable Assets

Review Package Contents

Option	Detail
Content Type	Click on this button to toggle between listing the Package diagrams first and listing the Package elements first.
Toggle Filter Bar	Right-click on the column headings and select this option to show or hide the Filter Bar on the display.
<filter bar fields>	Type in the appropriate characters to list only elements and diagrams that have that string of characters in the values in the corresponding column.
Content Type: Diagram Content Type: Element	These are the headers for the two types of object listed in this tab. Click on the appropriate expansion box to hide or show the list of diagrams or elements under the heading.
Name	Displays the name of the element or diagram available in the Package.
Type	Displays the type of the element or diagram, such as UseCase or Use Case Diagram.
GUID	Displays the Global Unique Identifier of the element or diagram.
Content Type	Displays the object type of the item - Element or Diagram.
Find in Project Browser	Right-click on an element or diagram line and select this option to see if the element or diagram also exists in your model and, if it does, to highlight it in the Project Browser.
View Diagram	Right-click on a diagram name and select this option to display the diagram within a labeled frame, as an image.

	Alternatively, double-click on the diagram name.

Notes

- The 'Contents' tab will be empty for a 'Learning Center Library', 'Reference Data Library' or 'Source Code Library' type Storage

Package Dependencies

A Package held as an asset in the Reusable Asset Service registry might contain elements and diagrams that have relationships with objects in other Packages. The Packages containing these 'external' objects will also have been added to the Registry, to support the first Package, and will be listed in the 'Registry Browser' tab along with other, unrelated Packages. You can establish whether the selected Package in the Registry Browser has links to other Packages in the Registry, and which they are, by reviewing the 'Dependency' tab.

Package A depends on Package B if any of these constructs (or their Tagged Values) in Package A references elements in Package B:

- Elements
- Attributes
- Operations
- Operation Parameters
- Diagrams
- Connectors

Access

Open the Reusable Asset Service window using one of the methods outlined here.

Select a Package, then click on the 'Dependency' tab, to display a list of Dependencies for that Package.

Ribbon	Publish > Model Exchange > Reusable Assets

Check Package Dependencies

Option	Detail
Package	Displays the name of the related Package.
Version	Displays the latest registered version of the related Package that has links to the selected Package.
GUID	Displays the Global Unique Identifier of the related Package.
Find in Project Browser	Right-click on the 'Package' line and select this option to see if the Package also exists in your model and, if it does, to highlight it in the Project Browser.

Notes

- If one Package depends on another, and that second Package itself depends on another Package, both the second and third Packages are shown in the 'Dependency' tab
- The 'Dependency' tab will be empty for a 'Learning Center Library', 'Reference Data Library' or 'Source Code Library' type Storage

Package Technologies

It is possible that the asset Package in the Registry is associated with one or more Technologies, especially if the Package comes from a model developed as an extension or customization of the UML. You can check whether an asset Package has any associated Technologies, and what they are, by viewing the 'Technology' tab of the Reusable Asset Service view.

Access

Open the Reusable Asset Service window using one of the methods outlined here.

Select a Package, then click on the 'Technology' tab, to display a list of Technologies required to support that Package.

Ribbon	Publish > Model Exchange > Reusable Assets

Review Technologies

Option	Detail
Name	Displays the name of the Technology as recorded in the Registry.
Version	Displays the required version of the Technology to support the asset Package, available through the Registry.
ID	Displays the unique identifier of the Technology.
Type	Identifies whether the Technology is system-supplied and built in to Enterprise Architect (such as BPMN 2.0 or ArcGIS), or user-defined. A user-defined technology can be introduced from an external source or developed by a user within the system.
Registry Status	Indicates whether the Technology file is: • Available - the Technology file is registered in the current Storage, or • Unavailable: - The Technology file is not registered in the Storage, or - The Technology is a built-in Technology and is already integrated with your system, or must be purchased under license, or - The Technology is available from a URL and can be downloaded from that URL
Import Technology	If your local system does not have one of the Available Technologies, you can quickly import it from the Registry. 1. Right-click on the Technology name and select the 'Import Technology' option. 2. A prompt displays for you to confirm the import to your system; click on the Yes button. 3. A message displays to confirm that the Technology has been imported; click on the OK button.

| | You might have to restart Enterprise Architect to activate the Technology. |
| | Other than the remote Technologies, you cannot import a Technology flagged as 'Unavailable'. You can import a remote Technology from its URL site. |

Notes

- You can also import Available Technologies from the registry as part of the process of importing the asset Package

- If the Storage is password protected, a prompt to enter your password displays (this can be a Read-Only or a Complete Access password) after you confirm that you want to import the Technology; you cannot import the Technology without this password

- The 'Technology' tab will be empty for a 'Learning Center Library', 'Reference Data Library' or 'Source Code Library' type Storage

Storage Files

The 'Storage Files' tab of the Reusable Asset Service view simply lists the files that have been uploaded to the specified Storage. The files can be of any type, uploaded from any directory on any system connected to the registry. Typically they would be document or graphics files that provide standards, guidelines or design information on a model structure. The tab displays the file name and extension, and a comment on the nature or purpose of the file.

Files held in the Registry are compressed, so to view the contents you download them onto your local system.

Access

Open the Reusable Asset Service window using one of the methods outlined here.

Select a Package, then click on the 'Storage Files' tab to display a list of files associated with the Package, that are available for download.

Ribbon	Publish > Model Exchange > Reusable Assets

Review files in Draft/Complete type Storage

On the 'Storage Files' tab, check the 'Comments' field against each file that interests you.

To view the contents of a file, right-click on the file name and select the 'Import From Registry' option. The 'Save As' browser displays, through which you select the directory into which to copy the file.

Click on the Open button. The file is copied into the selected location, from which you can open it.

Review files in Learning Center/Reference Data/Source Code Library type Storage

On the 'Storage Files' tab, check the 'Comments' field against each file that interests you.

To import and install a Library file, right-click on the file name and select the 'Import From Registry' option. One of these dialogs will display:

- 'Download Libraries' dialog for a 'Learning Center' type Storage

- 'Import Reference Data' dialog for a 'Reference Data Library' type Storage

- 'Save As' dialog for a 'Source Code Library' type Storage

'Learning Center Library', 'Reference Data Library' and 'Source Code Library' type Storages do not contain any Assets. Therefore, clicking on the:

- 'Import' button in the Registry Browser will import and install a Library file from the Storage

- 'Delete' button in the Registry Browser will delete all Library files in the Storage

Notes

- If the Storage is password-protected, the system prompts you to enter a password before you begin importing a file from the Storage; in this case you require either a 'Read-Only' or a 'Complete Access' password

Compare an Asset to the Model

If you are developing a Package in your model against a standard structure, or using common elements from the Reusable Asset Service, you can check that your model conforms to the standard or incorporates any changes to the common elements, by performing a comparison between the asset Package and your model.

Access

Open the Reusable Asset Service window using one of the methods outlined here.

Select a Package, then right-click on the Package and choose 'Compare with Package in Model'.

Ribbon	Publish > Model Exchange > Reusable Assets

Compare Asset Package and Model

A prompt displays for you to confirm the comparison. Click on the Yes button.

The Baseline Comparison view displays, showing the element hierarchy in which differences have been detected between the Package in the model and the asset Package in the Registry (as the Baseline).

All facilities of the standard Baseline Comparison are available, including the ability to 'roll back' differences so that the element or Package in the model matches the Registry.

Notes

- If the Storage is password-protected, the system prompts you to enter a password before you begin comparing information in the Storage and in the model; in this case you require either a Read-Only or a Complete Access password

Import an Asset into the Model

The Reusable Asset Service provides common or standard information and data that you can import into your local model. The unit that you import is a selected version of the Asset Package; you can also, optionally, import:

- Other Packages on which the selected Asset Package depends, and/or
- Available Technologies that support full use of the model structures in the Asset Package

The Asset Package (with, if selected, any needed Packages) is imported into the currently-selected model Package in the Project Browser, unless the Asset Package already exists anywhere else in the Project. In this case, the system locates the existing Package and overwrites it with the imported Asset Package.

Access

Open the Reusable Asset Service window, using one of the methods outlined here.

On the 'Registry Browser' tab, either;

- Select an Asset Package and click on the Import button, or
- Right-click on an Asset Package and select the 'Import from Registry' option

Ribbon	Publish > Model Exchange > Reusable Assets

Import Asset Package

Before selecting an Asset Package to import into your model, click on the 'Version' drop-down arrow and select the appropriate version of the Package to import.

When you select to import an Asset Package into your model, a short menu displays from which you select to import:

- The Package alone or
- The Package with the related Packages on which it is dependent

A prompt then displays for you to confirm the import, and to warn you that if the Package already exists in the model, it will be overwritten by the import. Click on the Yes button to continue, or the No button to cancel the import.

If the Storage is password protected, after you confirm the import a prompt displays for your 'Read-Only' or 'Complete-Access' password; enter this and click on the OK button.

The 'Import from Registry' dialog displays, followed by confirmation that the import of the Package or Packages is complete. Click on the OK button.

If the Package depends on Technologies

If the Technologies associated with an Asset Package are not already on your system and/or enabled in your model, during the display of the 'Import from Registry' dialog the 'Import Technology From Registry' dialog also displays. Process this dialog as described.

Option	Detail
Name	Displays the name of the Technology as recorded in the Registry. The check box against each name is selected by default. If you want to import one technology but not another, deselect the checkbox of the technology to ignore.

Version	Displays the required version of the Technology to support the Asset Package, available through the Registry.

Registry Status	Indicates whether the Technology file is:
	• Available - the Technology file is registered in the current Storage, or
	• Unavailable - for one of these reasons:
	  - The Technology file is not registered in the Storage, or
	  - The Technology is a built-in Technology and is already integrated with your system, or must be
	    purchased under license, or
	  - The Technology is available from a URL and can be downloaded from that URL

Model Status	Indicates whether the Technology is already in your model and disabled. (If the Technology is in your model and enabled, there is no need to import it and it is not listed in the dialog.)
OK	Click on this button to import and/or enable the selected Technology or Technologies in your system. A status message displays to indicate that the Package and Technologies have been imported and the Technologies enabled. You might have to restart Enterprise Architect for the Technologies to take full effect. Click on the OK button to clear the message. In the Project Browser, the Package has been added to the model with (if selected) the Packages on which it is dependent.

Import Library

The Reusable Asset Service provides the ability to store Library files that you can import into your local model. The type of Library file available in a Storage will depend on the Storage type. When you import and install a Library from a:

• Learning Center Library storage, it will be available for use on all Enterprise Architect models in this system

• Reference Data Library storage, it will be installed into the current model

• Source Code Library storage, it will be saved in the specified location in the file system

Library from a Learning Center Library storage. Either:

• Click on the Import button to open the 'Download Libraries' dialog, showing all the Library files in this storage as selected, or

• Right-click on a Library file in the 'Storage Files' tab and select the 'Import from Registry' option to open the 'Download Libraries' dialog showing that specific Library file as selected

Library from a Reference Data Library storage. Either:

• Click on the Import button to open the 'Import Reference Data' dialog, showing all the reference data in that Library file as selected, or

• Right-click on a Library file in the 'Storage Files' tab and select the 'Import from Registry' option to open the 'Import Reference Data' dialog, showing all the reference data in that Library file as selected

Library from a Source Code Library storage. Either:

• Click on the Import button to open the 'Save As' dialog and select a location in the file system to save the imported Library to, or

• Right-click on a Library file in the 'Storage Files' tab and select the 'Import from Registry' option to open the 'Save As' dialog, then select a location in the file system to save the imported Library to

If the Storage is password protected, a prompt for you to enter your 'Read-Only or 'Complete-Access' password displays. Enter the password and click on the OK button.

Notes

- You can also import Technologies separately from the Package that is dependent on them, using the 'Technology' tab of the Reusable Asset Service view
- 'Learning Center Library', 'Reference Data Library' and 'Source Code Library' type Storages do not contain any Asset Packages

Set Up Registry Password

As a System Administrator, you can password-protect the Registry server to prevent users from modifying the Registry; that is, restrict users from creating, modifying and deleting Storages. Once you have set password protection, only those users who have a valid Administrator password can modify the Registry.

You can also change the password at regular intervals, for added security.

Access

Ribbon	Publish > Model Exchange > Reusable Assets : Set Administrator Password

Set the Administrator Password

Option	Detail
Enter old password	Disabled if no password exists for the Registry server.
	If the server has a current password, type it in to this field.
New password	Type in the new Administrator password.
Retype new	Type in the new Administrator password, for confirmation.
OK	Click on this button to set the password and close the dialog.
Cancel	Click on this button to abort the changes and close the dialog.

Notes

- To remove password protection from the Registry server, complete the 'Enter old password' field but leave the 'New password' and 'Retype new' fields blank
- If a Storage has a Complete Access password, then you can use either the Administrator password or the Complete Access password to modify the Storage

Set Up the Asset Service

The process of setting up assets and files in the Reusable Asset Service has a number of simple stages, typically:

- Identify the Registry
- Create the Storages, including copying an existing Storage as a template for a new one
- Set password protection on each Storage
- Register the Asset Packages and Storage Files, and update them as necessary

Access

Ribbon	Publish > Model Exchange > Reusable Assets

Identify Registry

The Registry server holds the Storages of the Reusable Asset Service. It will have been previously configured by your system administrator, to be accessed through a Cloud Connection. Your system administrator will provide the connection details for you to use.

In the Reusable Asset Service view, click on the ⎡...⎤ button to the right of the 'Registry' field. The 'Cloud Connection' dialog displays.

Using the information provided by your administrator, type in the server name, URL and the name of the model hosting the registry data. If advised by your administrator, also type in the user name and password for the server.

Click on the OK button; the 'Registry' field now shows the server name.

Create Storages

A Storage is a container within the Registry, holding related assets and files. There can be several Storages in the Registry, to support the different areas of work or purposes that your organization might define.

Step	Action
1	Click on the 🗂 icon in the Reusable Asset Service toolbar.
2	If necessary, enter the Administration password that enables update access to the Registry. Click on the OK button. The 'New Storage' dialog displays.
3	In the 'Name' field, type a name that indicates the purpose or content of the Storage.
4	In the 'Type' field, select: • 'Draft' if you intend to make significant changes to the content of the Storage, such as populating it with assets and/or files; you can overwrite existing Packages or files in a Draft Storage • 'Complete' if the Storage is ready for use; you cannot overwrite any Packages or files in a Complete Storage, although you can delete Packages and files and add new ones

	• 'Learning Center Library' if the Storage should contain only Enterprise Architect Learning Center Library files - this Storage type can be populated only with Library files and not with Asset Packages; you can overwrite existing Library files in a Learning Center Library storage • 'Reference Data Library' if the Storage should contain only Enterprise Architect Reference Data Library files - this Storage type can be populated only with Library files and not with Asset Packages; you can overwrite existing Library files in a Reference Data Library storage • 'Source Code Library' if the Storage should contain only Source Code Library files - this Storage type can be populated only with Library files and not with Asset Packages; you can overwrite existing Library files in a Source Code Library storage You can modify a Storage at a later time, and change this Type setting.
5	If you want to set Storage Access Passwords on this Storage, click on the check box against one or both password types, and provide and confirm the 'Complete Access' and/or 'Read-Only Access' passwords.
6	Click on the OK button. A prompt displays for you to confirm the creation of the Storage, with the type of access you have set on the Storage.
7	Click on the Yes button. A confirmation message displays; click on the OK button. The name of the new Storage displays in the 'Storage' field.

Password Protection

The Registry can be protected by a Registry password, so that only the Administrator can create and modify Storages. If no Registry password is set, a second level of password protection can be applied.

The owner of a Storage can apply password protection to restrict users in processing and modifying the contents of the Storage either intentionally or unintentionally. Password protection can be applied at two levels:

• Complete Access - users enter their password and can modify or delete the Storage itself, and within the Storage can register, update and delete Packages and files in the Registry, view Packages and import assets into their models

• Read-Only Access - users enter their password and can import assets into their models

You set the 'Complete Access' password before you set the 'Read-Only Access' password.

If password protection is set and a user has no password, they can only view the contents of the Storage. If no password protection is set, all users can freely access and modify the Storage and its contents.

You set the passwords on each Storage as you create that Storage, using the 'New Storage' dialog. Once the Storage has been created, you can change or remove an existing password using the 'Modify Storage Access' dialog, but you cannot add a password.

If a Storage is password protected, the system displays a prompt for the password when the user begins to:

• Change or delete a password on the Storage

• Copy the Storage to create a new Storage

• Delete the Storage

• Register a Package or file in the Storage

• Import a Package, file or Technology from the Storage

• Delete a Package or file from the Storage

• Compare an Asset Package with a Package in the model

The system displays the password prompt when any of those operations is performed for the first time after the user:

- Connects to a Registry
- Reloads the Registry Browser
- Selects a Storage in the 'Storage' field in the Reusable Asset Services view

If the user enters the correct password and therefore establishes their credentials, the system does not display a password prompt again for any valid operation that user performs until they:

- Select a different Storage in the 'Storage' field in the Reusable Asset Services view or
- Reload the Registry Browser from the Registry

Register Assets and Files

Whilst a Storage exists and, if the Storage is under password protection, you have the 'Complete Access' password, you can register Packages from a model and files from your system as assets in that Storage. If the Asset Package was developed using one or more MDG Technologies, you can optionally register those technologies in the Storage as well.

See the *Register New Assets* Help topic.

Copy Storages

It is possible to copy a Storage as the basis for creating another Storage, for example if you want to use the same set of assets in the context of larger set for a different department, work area or development phase.

Note: It is not possible to change the Type of a 'Learning Center Library', 'Reference Data Library' or 'Source Code Library' type Storage, nor is it possible to change a 'Draft' or 'Complete' type Storage into a 'Learning Center Library', 'Reference Data Library' or 'Source Code Library' type Storage.

Step	Action
1	In the 'Storage' field, select the Storage to be copied.
2	Click on the [icon] icon in the Reusable Asset Service toolbar.
3	If necessary, enter the Administrator password or your 'Complete Access' password and click on the OK button. The 'Save Storage As' dialog displays.
4	In the 'Name' field, type a name that indicates the purpose or content of the new Storage.
5	In the Type field, select: • 'Draft' if you intend to make significant changes to the content of the Storage, such as populating it with assets and/or files; you can overwrite existing Packages or files in a 'Draft' Storage • 'Complete' if the Storage is ready for use; you cannot overwrite any Packages or files in a 'Complete' Storage, although you can delete Packages and files and add new ones You can modify a Storage at a later time, and change this setting.
6	If you want to set Storage Access Passwords on this Storage, click on the check box against one or both password types, and provide and confirm the 'Complete Access' and/or 'Read-Only Access' passwords.
7	Click on the OK button. A prompt displays for you to confirm the creation of the Storage, with the type of access you have set on the Storage.

8	Click on the Yes button.
	A confirmation message displays; click on the OK button.
	The name of the new Storage displays in the 'Storage' field.

Change Storage Details

After you have set up a Storage, you can update it at a later stage to:

- Change the status or Type

- Change one or both of the existing 'Complete Access' and 'Read-Only' Access passwords

- Delete one or both of the passwords

It is not possible to add a password where one has not previously been set.

You cannot change or delete passwords unless you have the 'Complete Access' password yourself.

Note: It is not possible to change the Type of a 'Learning Center Library', 'Reference Data Library' or 'Source Code Library' type Storage, nor is it possible to change a 'Draft' or 'Complete' type Storage into a 'Learning Center Library', 'Reference Data Library' or 'Source Code Library' type Storage

Step	Action
1	On the Reusable Asset Service view, in the 'Storage' field click on the drop-down arrow and select the Storage to update.
2	Click on the [icon] icon in the Reusable Asset Service view toolbar.
3	If necessary, enter the Administrator password or your 'Complete Access' password and click on the OK button. The 'Modify Storage Access' dialog displays.
4	In the 'Type' field, click on the drop-down arrow and select: • 'Complete' if the Storage has now been set up and is ready for use; you cannot overwrite any Packages or files in the Storage, although you can delete them and add new Packages and files • 'Draft' if you intend to make significant changes to the content of the Storage; you can overwrite existing Packages or files
5	The 'Action' field defaults to 'None', to indicate no changes to the passwords. If you are changing or deleting the passwords, click on the drop-down field and select: • 'Change Password' to enable both password type fields ready for editing • 'Remove Storage Access Password' to simply delete the 'Complete Access' password altogether • 'Remove Read-Only Access Password' to simply delete the 'Read-Only Access' password altogether
6	If you have selected the 'Change Password' option, select the appropriate password checkbox or both checkboxes, and type in the new password(s). Re-type the password(s) in the 'Confirm Password' field(s).
7	Click on the OK button to save your changes and close the 'Modify Storage Access' dialog.

Delete a Storage

If a Storage has been created by mistake, or is no longer necessary in the Registry, you can delete it and its entire contents of Package and files in one action.

Step	Action
1	On the Reusable Asset Service view, in the 'Storage' field click on the drop-down arrow and select the Storage to delete.
2	Click on the ✕ icon in the Reusable Asset Service view toolbar. A prompt displays to confirm the deletion.
3	Click on the Yes button. If necessary, enter the Adminstrator password or your 'Complete Access' password and click on the OK button. A message displays to confirm that the Storage has been deleted.
4	Click on the OK button to clear the message. The Storage and its contents are cleared from the Registry.

Register New Assets

After you have set up your 'Draft' or 'Complete' Storages in the Reusable Asset Service Registry, you can register assets in them. The assets include:

- Packages of modeling structures, held in the Project Browser of your model; these Packages can include diagrams and subordinate Packages
- Any Packages containing model components on which the primary Asset Packages depend
- Any MDG Technologies that you used in developing the Asset Packages, and that you decide to register with the Packages
- Any text or graphics files you want to make available to your corporate community; these files are compressed before being added to the Storage

You register the first three items together. You can register files in the Storage during that same process, or separately.

You can register separate versions of the same Package. If you register a Package that already exists in the Registry under the same version reference, and the Storage has 'Draft' status, the existing version is overwritten. If you register the same Package with a different version number or reference, it is registered separately and users can access both versions of the Package from the Registry.

'Learning Center Library', 'Reference Data Library' and 'Source Code Library' type Storages cannot contain Packages of modeling structures - they can contain only Library files, which will appear in the 'Storage Files' tab. A Library file is :

- a zipped file containing Enterprise Architect-specific Learning Center files for the 'Learning Center Library' Storage type
- an XML file containing Enterprise Architect-specific reference data XML files for the 'Reference Data Library' Storage type
- a zipped file containing source code files for the 'Source Code Library' Storage type

Once registered, the Library file will be available in the 'Storage Files' tab of the Reusable Asset Service view.

Access

Ribbon	Publish > Model Exchange > Reusable Assets

Register Package as Asset

Step	Action
1	In the Reusable Asset Service view, select the appropriate Registry and Storage.
2	In the Project Browser for your model, click on the Package to register as an Asset.
3	On the 'Registry Browser' tab, click on the Register button. If the Storage is password protected, a prompt displays for your 'Complete Access' password. Type this in and click on the OK button.
4	The 'Register Package(s)' dialog displays, showing the: • Name and type of the Storage • Name and status of the selected Package (the status at this point being 'Pending')

	• The name, GUID and current version of the selected Package; you can type a different version number or text string if necessary If there is a version of the Package already in the Registry, and the Storage has 'Draft' status, you can either: • Overwrite the registered Package with an update of that version, or • Add the updated Package as another Asset under a different version number; in this case, the 'Packages to Register' panel shows the version number, registration date and comments on the latest version of the Asset Package You cannot overwrite an existing Package if the Storage has 'Complete' status. In this case, add the Package under a different version number.
5	In the 'Comments' field type a short comment and, in the 'Notes' field, a longer explanation of the Package being registered.
6	Click on the Check Dependency button and resolve the dependencies of the Package. At the end of this process, the selected Package has the status 'Ready' (for registration).
7	You can, at this point, also register files as assets of the Storage. See Step 4 of the next table, *Register File as Asset*. However, if you prefer you can follow the complete procedure separate from registering Packages.
8	Click on the Register button. A prompt displays to confirm registering the Package. Click on the Yes button. If the same version of the selected Package already exists in the Registry (if the Storage is a 'Draft') a prompt displays to confirm overwriting it. Click on the Yes button. A progress dialog displays. If the Package does not have any associated MDG Technologies, a message displays confirming that Package registration is complete. Go to step 10.
9	If the Package has one or more associated MDG Technologies, the 'Register MDG Technology' dialog displays, listing the Technologies and prompting you to register (OK) or ignore all of them. If you want to register some Technologies but not others, click on those you do not want to register, and click on the Remove button for each one in turn. Click on the OK button or Ignore button. A message displays confirming that Package and Technology registration is complete. Notes: • If a remote Technology is removed from registration (that is, the Technology is not registered along with the Package), the registered Package will refer to this Technology at its URL When importing the Package from Registry, the system will import this remote Technology from its URL (if the remote Technology is not already available in the importing model) • If a remote Technology is registered with the Package being registered, the system will store a copy of the Technology file in the Registry When importing the Package from Registry, the system will import the Technology stored in the Registry and will not import it from its URL (if the remote Technology is not already available in the importing model)
10	Click on the OK button. The 'Registry Browser' tab is updated with the newly-added details of the Package and any dependent Packages. You can review the details on the tabs on the lower half of the Reusable Asset Service view.

Register File as Asset

Step	Action
1	In the Reusable Asset Service view, select the appropriate Registry and Storage.
2	On the 'Registry Browser' tab, click on the Register button.
	If the Storage is password-protected, a prompt displays for your 'Complete Access' password. Type this in and click on the OK button.
	The 'Register Package(s)' dialog displays.
3	If any Packages are listed in the 'Packages to Register' panel, and you do not want to re-register them, clear the checkbox against each Package name.
4	Click on the Additional Files button.
	The 'Select Additional Files' dialog displays.
5	Click on the Add button.
	A 'Select' screen displays, on which you browse for and click on the required file. You can only select one file at a time on this browser.
6	Click on the Open button.
	The 'Add Comment' dialog displays.
7	Type a brief comment on the nature or purpose of the file in the Registry, and click on the OK button.
	Focus returns to the 'Select Additional Files' dialog, which now lists the selected file.
8	Repeat steps 5-7 for each Asset file to register.
	When you have finished, click on the OK button.
	The 'Register Package(s)' dialog redisplays.
9	Click on the Register button.
	A prompt displays to confirm registration of the files.
10	Click on the Yes button.
	A 'Progress' dialog displays while the files are registered, followed by a message confirming that the (Package and) File registration is complete.
11	Click on the OK button to clear the message, and to return focus to the Reusable Asset Service view.
	You can review the files on the 'Storage Files' tab in the lower half of the view.

Register Learning Center Library

Step	Action
1	In the Reusable Asset Service view, select the appropriate Registry and 'Learning Center Library' Storage.

2	On the 'Registry Browser' tab, click on the Register button.
	If the Storage is password protected, a prompt displays for your 'Complete Access' password. Type this in and click on the OK button.
3	The 'Register Learning Center Library' dialog displays.
	Click on the [...] button next to the 'File' field and select the Library file.
	Type in a short comment on the Library being registered, in the 'Comments' field.
4	Click on the Register button.
	A prompt displays to confirm registering the Library file. Click on the Yes button.
	If the same Library file already exists in the Registry, a prompt displays to confirm overwriting it. Click on the Yes button.
	A progress dialog displays followed by a message confirming that Library registration is complete.
5	Click on the OK button.
	The 'Storage Files' tab in the lower half of the Reusable Asset Server view is updated with the newly-added Library file.

Register Reference Data Library

Step	Action
1	In the Reusable Asset Service view, select the appropriate Registry and 'Reference Data Library' Storage.
2	On the 'Registry Browser' tab, click on the Register button.
	If the Storage is password protected, a prompt displays for your 'Complete Access' password. Type this in and click on the OK button.
3	The 'Register Reference Data Library' dialog displays.
	Press the [...] button next to the 'File' field and select :
	• From File System – to select an existing Reference Data XML file from the file system
	• From Model – to open the 'Export Reference Data' dialog for selecting categories that will be registered as the Reference Data Library once the Register button is clicked and a name is entered in the 'Enter Reference Data File Name 'dialog
	Type in a short comment for the Library being registered in the 'Comments' field.
4	Click on the Register button.
	A prompt displays to confirm registering the Library file. Click on the Yes button.
	If the same Library file already exists in the Registry, a prompt displays to confirm overwriting it. Click on the Yes button.
	A progress dialog displays followed by a message confirming that Library registration is complete.
5	Click on the OK button.
	The 'Storage Files' tab in the lower half of the Reusable Asset Server view is updated with the newly-added Library file.

Register Source Code Library

Step	Action
1	In the Reusable Asset Service view, select the appropriate Registry and 'Source Code Library' Storage.
2	On the 'Registry Browser' tab, click on the Register button.
	If the Storage is password protected, a prompt displays for your 'Complete Access' password. Type this in and click on the OK button.
3	The 'Register Source Code Library' dialog displays.
	Select the Library file by pressing the [...] button next to the 'File' field.
	Type in a short comment for the Library being registered, in the 'Comments' field.
4	Click on the Register button.
	A prompt displays to confirm registering the Library file. Click on the Yes button.
	If the same Library file already exists in the Registry a prompt displays to confirm overwriting it. Click on the Yes button.
	A progress dialog displays, followed by a message confirming that Library registration is complete.
5	Click on the OK button.
	The 'Storage Files' tab in the lower half of the Reusable Asset Server view is updated with the newly-added Library file.

Check Package Dependencies

As you register a Package in the Reusable Asset Service, the system provides a check on whether that Package is dependent on any external content; that is, whether it links to any modeling component held in Packages other than its own child Packages. You can perform three types of dependency check:

- Complete, which will consider Package A to be dependent on Package B if Package A contains any element, attribute, operation, operation parameter, diagram or connector (or their Tagged Values) that references elements in Package B.

- Normative, which is the same as a Complete Dependency Check except that it will not treat as a dependency for Package A any element from Package B that is added as a link on a diagram in Package A

- Package Dependency/Import, where only those Packages that are target of a Dependency/Package Import connector from Package A will be treated as dependent Packages for Package A.

You perform this dependency check as an integral part of the Asset Registration process (specifically, at step 6 of the Register Package as Asset process). The check automatically identifies, firstly, the Packages directly needed by the Asset Package, and then any Packages needed by those initial Packages, and so on until the needed Packages are not dependent on any other.

In practice, such dependency chains are likely to be quite short. If you know that the original Asset Package is dependent on two other Packages, one of which is subordinate to the other, you can simplify the process further by selecting the Parent or Ancestor Package first and thus avoiding the extra steps of selecting the two Packages separately.

Access
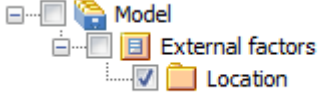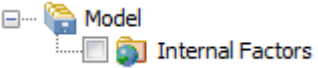
Open the Reusable Asset Service window, using one of the methods outlined here.

- In the Reusable Asset Service window, click on the 'Register' button

Ribbon	Publish > Model Exchange > Reusable Assets

Check Package Dependencies

Step	Action
1	Select the appropriate Registry, Storage and version of the model Package to register, and provide any appropriate comments or notes.
2	Click on the Check Dependency button. If there are no Packages on which the Asset Package depends, the system sets that Package to 'Ready'. Go to step 6. The 'Select Needed Package: <dependent package name>' dialog displays, showing the model hierarchy containing the Packages needed by the Asset Package. Each needed Package has a selected checkbox against it. If you know that a needed Package is a child or grandchild of another needed Package, click on the checkbox next to the 'higher' Package name so that the child is validated as part of the parent. In this example, if the Asset Package has dependencies on both External factors and Location, click on the 'External factors' check box to register both dependencies as External factors.

If the needed Package is already registered in the Registry, it is indicated by a globe icon:

In this case, select the 'Use latest version of selected package in Registry (if available)' checkbox to link to the registered version. For such Packages, no further dependency check is performed and, on the 'Register Package(s)' dialog, the Package's Status is set to 'Ready'.

3	Click on the OK button. The 'Register Package(s)' dialog updates to show the initial 'Asset Package status' as 'Ready', and to list the needed Packages each with the status of 'Pending'.
4	Click on the Check Dependency button again. If there are no Packages on which the needed Package depends, the system sets that Package to 'Ready' and sets the focus on the next Pending Package. If there are no more Pending Packages, go to step 6. If the needed Package is dependent on other Packages, the 'Select Needed Package: <dependent package name>' dialog displays again, showing the model hierarchy containing those Packages, each with a selected checkbox next to it. As in step 2, you can select a 'higher' Package if appropriate.
5	Click on the OK button. The 'Register Package(s)' dialog updates to show the first needed Package status as 'Ready', and to list any further needed Packages with the status of 'Pending.
6	If there are further needed Packages in the chain, repeat step 4. The system automatically identifies them and lists them for validation as in step 5. Otherwise the system sets the final needed Package status to Ready and you can continue to register files and MDG Technologies.

Notes

- A Package with status Pending cannot be registered; only Packages with status Ready can be registered
- If you are certain that there are no parent-child relationships between the needed Packages, you can work through these steps quickly by just clicking on the Check Dependency and OK buttons alternately until all Packages in the Package to Register list have the status Ready
- The system does not allow you to register an Asset Package that has dependencies on its own parent or grandparent Package; such dependencies would cause the Asset Package to be registered twice - by itself and as part of the parent Package - creating the risk of differences developing between the two instances

Update an Asset

Over time, you might need to revise the contents of the Registry. Such updates can include:

- Registering additional Asset Packages and files
- Registering a new version of an Asset Package
- Registering a new edition of a Storage file
- Registering new Libraries
- Deleting an Asset Package
- Deleting a Storage file
- Deleting a Library
- Changing the details of a Storage
- Deleting an entire Storage and the assets and files within it

Access

Ribbon	Publish > Model Exchange > Reusable Assets > Registry Browser

Update Tasks

Task	Detail
Register new Asset Packages and Storage files	You can add further Asset Packages and files to a Storage at any time.
Register a new version of an Asset Package	You register a new version of an Asset Package as if it were a new Asset. In the 'Current Version' field on the 'Register Package(s)' dialog, you type a different number or reference to that assigned to the currently-registered Asset.
	You can overwrite an existing Asset Package under the same version number only if the Storage has 'Draft' status. If the Storage has 'Complete' status, you cannot overwrite the existing Asset Package under the same version number. To replace the current version, you must delete the Asset from the Storage and register the Package again.
Delete Asset Package from Storage	Click on the Package name in the 'Registry Browser' tab and either:
	• Click on the Delete button or
	• Right-click and select the 'Delete From Registry' menu option
	In either case a short submenu displays, prompting you to select to delete:
	• The Asset Package only, or
	• The Asset Package and the Packages on which it is directly or indirectly dependant
	When you select the appropriate option, a prompt displays for you to confirm the deletion.
	Click on the Yes button. A message displays to confirm that the Package has been

	deleted from the Registry, and the name of the Package is removed from the 'Registry Browser' tab.
	If the Storage is password-protected, the system prompts you to enter a 'Complete Access' password before it begins deleting the Package(s).
	If any of the registered Packages depend on the Package being deleted, the system displays the names of these Packages and prompts for confirmation before deleting each Package.
Register a new version of a Storage File	You register a new version of a Storage file as if it were a new file under a different file name.
	You can overwrite an existing file under the same file name only if the Storage has 'Draft' status. If the Storage has 'Complete' status, you cannot overwrite the existing file. To replace the current version, you must delete the file from the Storage and register the file again.
Delete Storage Files from Storage	On the 'Storage Files' tab, right-click on the file name and select the 'Delete From Registry' option. A prompt displays for you to confirm the deletion.
	Click on the Yes button. A status message confirms the deletion, the file is removed from the Registry and the file name is removed from the 'Storage Files' tab.
	If the Storage is password-protected, the system prompts you to enter a 'Complete Access' password before it begins deleting the file.
Register a new Library File	You can add further Library files to a Storage at any time.
	When you register a Library file, the file name and location will be used as the unique identifier for the file. You can overwrite this Library file in the Registry by registering a file with the same name and from the same file system location.
Delete Library Files from Storage	To delete all Library files in Storage, click on the Delete button in the 'Registry Browser' tab.
	To delete a particular Library file, right-click on the Library file name in the 'Storage Files' tab and select the 'Delete From Registry' option.
	A prompt displays for you to confirm the deletion. Click on the Yes button. A status message confirms the deletion, the Library file(s) is removed from the Registry and the Library file(s) name is removed from the 'Storage Files' tab.
	If the Storage is password-protected, the system prompts you to enter a 'Complete Access' password before it begins deleting the Library file.
Change Storage Details	This task is as documented for setting up the Registry. Note that whilst you can copy a Storage as a new Storage, you cannot transfer assets directly between existing Storages.
Delete a Storage and its contents	This task is as documented for setting up the Registry. Note that whilst you can copy a Storage as a new Storage, you cannot transfer assets directly between existing Storages.
Refresh Registry Browser	If there is a possibility that another user has updated the Registry while you have had it open, click on the ⟳ icon in the Reusable Asset Service toolbar to refresh your display from the latest updates to the Registry.

Change Management

As a repository is developed it will become the data store for valuable organizational information assets, and it is imperative that this data is protected and maintained. Enterprise Architect has sophisticated tools to ensure the information is protected, including full integration with all the leading Version Control Systems, Baselines that are snapshots of your model that can be taken at important milestones, and Auditing that can track the finest changes to a model. A Project Transfer function helps you to make backups of models without involving information technology personnel. There are also Model Validation and Project Integrity Checkers to ensure the repository is maintained with a clean bill of health, and a powerful role based Security System to ensure users can collaborate easily.

Facilities

Facility	Detail
Version Control of Packages	Enterprise Architect Model Version Control helps you to: • Coordinate sharing of Packages between users, with either read-only access or update access, ensuring that work on different areas of the model is coordinated and synchronous rather than conflicting • Save and retrieve a history of changes to Packages To use version control in Enterprise Architect, you require a third-party source-code control application such as: • Subversion • CVS • MS Team Foundation Server (TFS), or • Any other version control product that complies with the Microsoft Common Source Code Control standard
Tracking Changes	Enterprise Architect provides two separate but complementary facilities for tracking changes to data across the project: • Auditing of model changes • Baselining and differencing to capture and roll back changes
Project Data Transfer	Enterprise Architect enables you to transfer project data between project data repositories either for: • Sections of the project (XMI and CSV) or • The whole project, row by row, table by table (in the Corporate, Business and Software Engineering, Systems Engineering and Ultimate editions of Enterprise Architect)

Version Control

Within Enterprise Architect, you can manage changes to and revisions of your projects by placing individual model Packages, view nodes or root nodes under version control. Version control provides numerous key facilities, including:

- Saving a history of changes to Packages

- The ability to retrieve previous revisions of Packages

- Propagating model updates between team members

- Coordinating the sharing of Packages between team members

You apply version control through a third-party source-code control application that manages access to and stores revisions of the controlled Packages. Once the version control software has been installed and configured, you must define a Version Control Configuration within your project. You can then use version control to manage changes to the Packages of your model.

Notes

- Sparx Systems strongly urge you not to manipulate version controlled Package files outside of Enterprise Architect; it is possible to leave the Package files in a state that Enterprise Architect cannot recognize

- Database replication should not be combined with version controlled Packages

- If the Packages under version control contain any alternative images and those images are subject to frequent change, you can set the 'Export alternate images' option on the 'Preferences' dialog to export the images to the version control repository when you check in the Packages; if the images are not subject to frequent change, do not select this option and instead use 'Export/Import Reference Data' to manage alternative images

Introduction

Enterprise Architect's version control integration provides several key facilities, including:

- Saving a history of changes made to your model's Packages
- Retrieving previous revisions of Packages
- Propagating model updates between team members
- Coordinating the sharing of Packages between team members

The steps involved in setting up a version control environment and applying version control to an Enterprise Architect project that is to be accessed by a number of users, are summarized in the Help topic 'Applying Version Control in a Team Environment'.

There are a number of factors to consider when setting up and using version control in your model development.

Factors to consider

Factor	Detail
System Requirements and Configuration	You apply version control through a third-party source-code control application that manages access to and stores revisions of the controlled Packages. Typically the configuration consists of: • A server component that manages a version control repository, and • Client components on the workstations, that manage local working copies of controlled files Enterprise Architect uses the client component to communicate with the server. A version control client must be installed on every machine where you run Enterprise Architect and want to access your version control system.
Version Control Usage	There are two main ways in which projects can be deployed: • Centralized Shared Model • Distributed Private Models Version control is employed in the same way for both scenarios; however, when using Private Model deployment you have the additional facility of propagating model updates throughout the team. Version Control can also be used to share standard Packages between different projects.
Team Deployment	Consider the process of setting up a version control environment and applying version control to a project to be accessed by a number of users.
Version Control Basics	Enterprise Architect enforces serialized editing of version controlled Packages, using the lock-modify-unlock mode of operation.
Applying Version Control to Models	Using version control consists of placing individual model Packages under version control, rather than version controlling the project as a whole.
Version Control and Project Reference Data	To share changes in reference data between users in a version-controlled project deployed as multiple private models, you periodically export the reference data

	from the model where the changes were made, and import it into the other models maintained by the team.
Offline Version Control	You can prevent the system from attempting to make any version control connections by choosing to Work Offline before loading a model.
	If Enterprise Architect is unable to connect a Version Control Configuration for any reason, it displays warning messages to notify you and provides 'offline' version control functionality for all Packages associated with the failed connection.

Notes

- Packages under version control are identified in the Project Browser by icons that indicate the current status of the Package

Version Control Usage

The version control facility can be used in many different ways, although these roughly fall into one of four types of use as discussed here.

Usage

Type	Details
Single Shared model	Users share a model stored in a central project file or DBMS repository. In this configuration you can view changes to other users' Packages without explicitly having to check them out, by simply refreshing your view of the model.
	Version control is used to:
	• Archive successive versions of your work to date
	• Maintain Package revision history
	• Recover from unwanted changes or accidental deletions, through an 'undo' facility
	• Regulate access to Packages
Multiple Private models	One model is created by a single user who configures it for version control. The model file is then distributed to other users, with each user storing their own private copy of the model.
	Version control is used to:
	• Propagate changes to the model, throughout the team
	• Archive successive versions of your work to date
	• Maintain Package revision history
	• Recover from unwanted changes or accidental deletions, through an 'undo' facility
	• Regulate access to Packages
Shared Packages	Individual users create separate models but share one or more Packages:
	• Users share Packages through version control
Standard Packages	A company might have a standard set of read-only Packages that are broadly shared:
	• Individual users retrieve Packages with the Get Package menu option

Version Control of Model Data

When applying version control in Enterprise Architect, you place individual model Packages under version control, and not the project as a whole.

All Enterprise Architect models are stored in databases - even the .eap file is an MS Jet database. In simple terms, the project file is a single entity of binary data. Being binary data, the project file would require the use of the lock-modify-unlock model of version control, which would mean that only a single user at a time could work on any given (version controlled) model. Therefore, it is not practical to apply version control to the database (.eap file) as a whole; this can also create problems for you:

- Most version control systems mark their controlled files as read only, unless they are specifically checked-out to you
- The .eap file is an MS Jet database, and Enterprise Architect must be able to open this file for read/write access when you load your model; the system displays an error message and fails to load the model if it is read-only

Version Controlling Packages in your Model

To overcome these limitations, Enterprise Architect exports discrete units of the model - the Packages - as XMI Package files, and it is these XMI files, not the project file, that are placed under version control. The XMI file format used by Enterprise Architect dictates that they too be treated as binary files - therefore it is not possible to merge the XMI files either; however, by splitting the model into much smaller parts, many users can work on separate parts of the model simultaneously.

Version Controlled Nested Packages

Version controlled nested Packages result in much smaller XMI files being exported, as the parent Packages' XMI files do not contain any content for the version controlled child Packages.

Version control of nested Packages, together with a model structure of small individual Packages, provides greater scope for multiple users to work concurrently, as individual users are locking much smaller parts of the model.

Version Control and Reference Data

Reference data is data that is used across a model or project; it is not Package-specific. Version control operates at Package level, and therefore does not capture changes in reference data. Where version control is used in a multiple private model set up, changes in reference data are not brought into the model when Packages are updated from version control.

In a Shared Model environment, all users are accessing the same project reference data. Changes in reference data can be shared between users in a version-controlled project deployed as multiple private models, by periodically exporting the reference data from the model where the changes were made, and importing it into the other models maintained by the team.

Reference data is exported and imported as an XMI file, which contains whatever types of reference data you want to transfer. You can place your project reference data under version control by exporting the data as an XMI file and apply version control to that file using your version control software external to Enterprise Architect.

Version Control Basics

Enterprise Architect implements version control of your model by exporting Package data from the project database to XMI Package files, which are placed under version control in the source-code control application. The XMI file format cannot be merged in the same way as ordinary text files can be merged, which is why Enterprise Architect must enforce serialized editing of version controlled Packages, as discussed here.

The Lock-Modify-Unlock Solution

Many version control systems use a lock-modify-unlock model to address the problem of different authors in a shared source overwriting each other's work. In this model, the version control repository allows only one person to change a file at a time, and access is managed using locks.

Harry must lock a file before he can begin making changes to it. If Harry has locked a file, Sally cannot also lock it, and therefore cannot make any changes to that file. All she can do is read the file, and wait for Harry to finish his changes and release the lock. After Harry unlocks the file, Sally can take her turn in locking and editing the file.

The Copy-Modify-Merge Solution

Subversion, CVS and a number of other version control systems use a copy-modify-merge model as an alternative to locking. In this model, each user's client contacts the project repository and creates a personal working copy - a local reflection of the repository's files and directories. Users then work simultaneously and independently, modifying their private copies. In due course, the private copies are merged together into a new, final version. The version control system often assists with the merging, but ultimately a person is responsible for making it happen correctly.

When Locking is Necessary

While the lock-modify-unlock model is generally considered a hindrance to collaboration, there are still times when locking is necessary.

The copy-modify-merge model is based on the assumption that files are contextually merge-able: that is, that the files in the repository are line-based text files (such as program source code). However, for files with binary formats, such as artwork or sound, it is often impossible to merge conflicting changes. In these situations, it really is necessary for users to take strict turns in changing the file. Without serialized access, some users end up wasting time on changes that are ultimately overwritten.

Applying Version Control in a Team Environment

The process of setting up a version control environment and applying version control to a project to be accessed by a number of users is summarized here.

Version Control - Process Overview

Step	Action
1	Install your version control product.
2	Create a version control repository.
3	Create a version control project to be used with your Enterprise Architect project.
4	Check-out a working copy of the version control project (a module, project or folder within the version control system) into a local folder. You must do this for every team member that is accessing the version controlled Packages, whether you are using a single shared model or each team member stores his own private copy of the model.
5	Before attempting to access the new version control environment from within Enterprise Architect, you should first verify that it functions correctly when used outside of Enterprise Architect.
6	Within Enterprise Architect, define a version control configuration to provide access to the working copy files. The name of the version control configuration must be the same across all machines throughout a team. That is, all version control access to a given Package must be through version control configurations with the same name, across all models and all users. The easiest way to perform this step, (throughout the team), is to have one user set up version control on the model and then share that model with the rest of the team. • In Shared Model deployment, all users connect to a single instance of the model database, so the model is shared automatically • In Private Model deployment, it is easiest to distribute copies of the original model (after version control has been set up) to all other members of the team Whenever you open a model (Private or Shared) that uses a version control configuration that is not yet defined on your workstation, a prompt displays to complete the definition for that configuration. This typically means specifying the local working copy directory and perhaps choosing the version control project associated with this Enterprise Architect project. Once this has been done, the version controlled Packages that already exist in the model are ready for use.
7	Configure Packages within the Enterprise Architect model for version control. That is, apply version control to individual Packages.
8	Check-out and check-in Packages as required.

Notes

- It is possible to use multiple version control configurations within the same model; different Packages can still use different version control configurations within the model, as long as any given Package is always accessed via the same version control configuration

Version Control Nested Packages

When you save a Package to the version control system, only stub information is exported for any nested Packages. This protects information in a nested Package from being inadvertently over-written by a top level Package.

Operations on Nested Packages

Operation	Detail
Checking Out	When you check out a Package, you do not modify or delete nested Packages; only the top level Package is modified.
	As a consequence of this behavior, if you check out or get a version controlled Package with nested Packages that are not already in your model, you see stubs in the model for the nested Packages only.
Get All Latest	If you select the 'Get All Latest' option from the version control menu, any new stubs are populated from the version control system.
Importing Models	You can populate a large and complex model, by 'getting' only the root Packages, then using 'Get All Latest' to recursively iterate through the attached and nested Packages.
	This is a powerful and efficient means of managing your project and simplifies handling very large models, even in a distributed environment.
	The 'Import a Model Branch' option combines the procedure steps into a single operation.

Notes

- It is recommended that, when sharing a version controlled model, you do not mix versions of Enterprise Architect later than version 4.5 with earlier versions; if this is necessary it is best to go to the 'Version Control Settings' dialog and deselect the 'Save nested version controlled packages to stubs only' checkbox, setting Enterprise Architect to the pre-version 4.5 behavior (for the current model only)

Add Connectors To Locked Elements

Generally, when working in a diagram containing locked elements, you cannot add a connector to a locked element. However, this depends on the lock status of the source and target elements (or more precisely, the lock status of the parent Packages of the source and target elements, when the source and target element are held in different Packages). There are scenarios in which a connector can be created on a locked element.

Lock Scenarios

Element Status	Add Connectors
Source unlocked, target unlocked	Yes, any kind of connector can be added
Source unlocked, target locked	Yes, except for Composition connectors
Source locked, target unlocked	No, except for Composition connectors
Source locked, target locked	No, prohibited for all connectors

Notes

- A connector can be added if its source is unlocked - you are modifying what the source can see
- The exception is Composition connectors, where the target (the parent) must be unlocked - you are modifying the parent by adding children

Project Browser Indicators

Packages under version control are identified in the Project Browser by icons that indicate the current status of the Package.

Indicators

Icon	Indicates that
	This Package is version controlled and not checked out to you. You cannot edit the Package (until you check out the Package yourself).
	This Package is version controlled and checked out to you. You can edit the Package.
	This Package is version controlled, but you checked it out whilst not connected to the version control server. You can edit the Package but there could be version conflicts when you check the Package in again.
	This Package is controlled and is represented by an XMI file on disk, but it is not under version control. You can edit this Package.

Offline Version Control
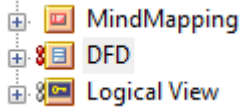
When loading a model that uses version control, Enterprise Architect normally initializes a connection to the version control system for each version control configuration defined in the model. If Enterprise Architect is unable to connect a version control configuration for any reason, it displays warning messages to notify you and provides offline version control functionality for all Packages associated with the failed connection.

You can prevent Enterprise Architect from starting to make any version control connections, by selecting to work offline menu before loading your model.

Access

Ribbon	Configure > Version Control > Work Offline

Working Offline

Concept	Discussion
Choosing to Work Offline	Selecting to work offline is useful if you know beforehand that Enterprise Architect cannot connect to your version control system. For example: If you are working on a laptop computer that is disconnected from your network, on an Enterprise Architect model that uses a large number of version control configurations, you can select to work offline before you load the model to avoid all the error messages that the system would normally display as each version control connection attempt failed.
	You can switch between working offline and working online at any time, either before or after a model is loaded, by toggling the 'Work Offline' menu option. Enterprise Architect disconnects or reconnects version control (depending on the connection availability) according to your selection.
Using Version Control Whilst Disconnected From the Version Control Server	Enterprise Architect 'remembers' the status of a model's version controlled Packages. Packages that were checked out to you prior to disconnecting from the server are still shown as checked out to you, even though you are no longer connected to the server. You can still edit these Packages as you normally would.
	Packages that were not checked out to you prior to disconnecting from the server are shown as version controlled and locked. You cannot edit these Packages until you check them out.
Offline Check Out	You can 'check-out' and edit a version controlled Package even when your machine is disconnected from the version control server. In this example, the colored 'figure 8' icon for DFD indicates that you have checked it out whilst off line.
	(The gray 'figure 8' icon shown against Logical View indicates that you have

	checked out a version-controlled Package on line.)
	You should be aware that the version control system, and therefore other users, have no way of knowing that you have 'checked-out' a Package whilst offline. It is not possible to merge changes to an XMI file that result from two users editing the same Package at the same time. If an offline checkout leads to two people editing the same Package at the same time, when the changes are brought back online the first-saved set of changes is lost.
Checking In a Package That Was Checked Out Offline	Once you reconnect your system to the version control server, if the Package you checked out offline is not currently checked out by another user, you can check in that Package. However, before Enterprise Architect checks in the Package, it compares the local working copy of the Package file with the latest revision in the repository. (These Package files remain unchanged in your work area until Enterprise Architect exports the Package again before checking in.) If the repository version remains unchanged from when you last updated your local copy, Enterprise Architect exports and checks in your Package without further prompting.
	On the other hand, if the repository now contains a file that has changed since you last updated your local copy, checking in your Package would overwrite those changes. Enterprise Architect displays a message warning you of the pending data loss and giving you the opportunity to abort the check in. At this point, you must decide whether to discard your own changes, using the Undo Check Out command, or continue with your check in and overwrite the changes that have been committed to the repository since you last updated your local copy from the repository.
	You can use the File Properties command to determine who checked in the last changes to this Package. This might help you to discover what changes have been uploaded and decide whose changes take precedence.
Update Before You Disconnect	Whenever you are connected to the version control server, you are always working with the latest version of a Package. This is because you cannot modify a Package until you check it out from version control, and checking it out loads the latest revision from the repository into your model.
	This cannot happen when you are disconnected from the version control server. You are working on whatever versions you have on your machine, dating back to the last time you updated your local copy of each version controlled Package. So, if you are planning to work on a model whilst disconnected from version control, it is a very good idea to make sure that you have the latest versions of all Packages before you disconnect. The 'Get All Latest' option makes this a simple task.

Version Control Branching

Currently, Enterprise Architect does not support Version Control Branching.

Work-arounds to achieve similar results might be possible for certain version-control products; contact Sparx Support for advice.

Contacts

User Type	Contact via
Registered users	http://www.sparxsystems.com/registered/reg_support.html
Trial users	support@sparxsystems.com

Version Control Product Setup

To control and maintain the different revisions of your project Packages, Enterprise Architect uses third-party version control products. Once your version control product is installed and a suitable environment has been created, Enterprise Architect can use that environment to control your project's Packages.

Typically, version control products consist of:

- A server component
- A client component

Enterprise Architect integrates with the version control client components for Subversion, CVS and MS Team Foundation Server command line clients, as well as for products having API clients that comply with the MS SCCI specification.

Version Control System Components

Component	Detail
Version Control Server	The server component maintains the controlled files in their many revisions, in a central repository. The server component is usually located on a server machine that is accessible to all team members who are using version control.
Server Configuration	The steps for configuring a version control server are, broadly: • Install the software • Create a repository • Create version control projects (or modules or folders for use with specific projects) • Configure user IDs and passwords For details on configuring any particular version control server, consult the appropriate documentation provided with the version control product.
Version Control Client	The client component manages the working copies of the controlled files, submitting files to or retrieving files from the server as required. A version control client must be installed on every machine on which you run Enterprise Architect and want to access your version control system.
Client Configuration	The steps for configuring a version control client are, broadly: • Install the software • Create a new directory for use as a local working copy folder • Log in to the version control server • Associate the working copy folder with its corresponding server repository folder For details on setting up a product-specific version control environment for use with Enterprise Architect, click on the appropriate link in the next column.

System Requirements

Enterprise Architect is a Windows-based application and requires a Windows-based version control client for integration. It is independent of the version control server component and the platform on which that runs.

Version Control Product Requirements

Product	Detail
Subversion	Subversion is free, open source software. Subversion server components are available to run on a wide range of different hardware and operating systems. Enterprise Architect is not affected by your choice of server components, but requires Subversion's Windows-based command line client for integration. There are many graphical user interface clients available for use with Subversion, such as TortoiseSVN; this type of client cannot be used directly for integration with Enterprise Architect, but can be very useful in preparing a working Subversion environment for use by Enterprise Architect. Binary Packages are available for download from: • http://subversion.apache.org/packages.html Subversion documentation is available from: • http://svnbook.red-bean.com/nightly/en/index.html
Concurrent Versions System (CVS)	CVS is free, open source software. CVS server components are available to run on a wide range of different hardware and operating systems. Enterprise Architect is not affected by your choice of server components, but requires CVS's Windows-based command line client for integration. There are many graphical user interface clients available for use with CVS, such as TortoiseCVS; this type of client cannot be used directly for integration with Enterprise Architect, but can be very useful in preparing a working CVS environment for use by Enterprise Architect. The software is available for download from: • http://www.nongnu.org/cvs/ CVS documentation is available from: • http://cvsbook.red-bean.com/cvsbook.html
Microsoft Team Foundation Server	Enterprise Architect is able to use either the: • Command line client for TFS, or • MS TFS-SCC client Your choice of client affects how you specify your Version Control Configuration. MS TFS-SCC clients are available for download from Microsoft's web pages: • Visual Studio 2005 Team Foundation Server MSSCCI Provider • Visual Studio Team System 2008 Team Foundation Server MSSCCI Provider

Common Source Code Control (SCC)-compatible products	Any version control product that provides a client that complies with the Microsoft Common Source Code Control standard, version 1.1 or higher, can be integrated with Enterprise Architect.
	These products are SCC-compatible and are known to successfully integrate with Enterprise Architect:
	• Accurev
	• ClearCase
	• MS Visual Source Safe
	• MS TFS-SCC
	• MKS Source Integrity
	• Perforce
	• Source Offsite
	• Snapshot CM
	Products that do not appear in this list should still integrate successfully with Enterprise Architect, if there is a client available for that product that complies with the MS SCC API specification.

Create a Subversion Environment

You can use Subversion as a version control provider for Enterprise Architect. The first step in doing this is for a Subversion administrator to install and configure the appropriate software. A number of basic tasks are performed in creating an operational Subversion environment, and useful tools are available for performing some of these tasks.

Tasks in Creating a Subversion Environment

Task	Detail
Install server components	Executable files for Subversion can be obtained from the Apache Software Foundation. Subversion server components are available to run on a wide range of different hardware and operating systems; Enterprise Architect is not affected by your choice of server components. VisualSVN is a package that can greatly simplify the installation, configuration and management of your Subversion server.
Create a repository	Please consult the official Subversion documentation.
Create Subversion users	Please consult the official Subversion documentation.
Create a new repository sub-tree	It is good practice to create a new repository sub-tree in Subversion for each new Enterprise Architect model being added to version control with Subversion. Users should create a new local working copy from the sub-tree to be used with that model TortoiseSVN can greatly simplify the process of creating new repository sub-trees.
Install client components	Executable files for Subversion can be obtained from the Apache Software Foundation.
Create a working copy folder	A working copy folder must exist on each users' machine, for Enterprise Architect to use when exporting and importing the version controlled Package files. It is this folder that is specified as the Local Project Path, when defining your Version Control Configurations. The working copy folder is the 'sandbox' where you modify the controlled files. The working copy folder is usually associated with a folder that exists within the version control repository. In Subversion, to create a local working copy you perform an initial check-out of a folder from the Subversion repository; this downloads a copy of the folder and its contents, to create your local working copy. TortoiseSVN can greatly simplify the initial check out of a working copy folder.
Setting up Subversion under Wine/CrossOver	The process of setting up and using Subversion with Enterprise Architect under Wine is almost identical to the process when running natively under Windows, apart from minor differences in installing the Subversion client and performing the initial check out of the working copy folder.

Notes

- Enterprise Architect relies on exclusive file locking when applying version control to its Packages; file locking was not introduced into Subversion until version 1.2, therefore Enterprise Architect does not work with Subversion releases earlier than Subversion 1.2

- Enterprise Architect can only communicate with the Subversion server using the Subversion command line client svn.exe

Create a new Repository Sub-tree

When you set up Subversion as your version control tool, it is good practice to create a new repository sub-tree in Subversion for each new Enterprise Architect model. The sub-tree can be used to control the Package files for your model.

Create a new sub-tree in the Subversion Repository

Step	Action
1	Use Windows Explorer to create a temporary directory on your PC file system, to be imported into the Subversion repository as a new repository sub-tree. The directory would have this structure:
2	Open a Windows command prompt, navigate to *tempDir* and issue the Subversion command *import*. For example: C:\Documents and Settings\user> cd \tempDir C:\tempDir> svn import . https://host.example.com:8443/repos/ --message "Repository Initialization"
3	On your PC, delete the temporary directory structure (tempDir) and all its contents.

Notes

- After the import is finished, the original tree is not converted into a Subversion working copy; you should delete the temporary structure and check out a fresh working copy of the tree
- The process can also be performed using TortoiseSVN's Repository Browser, which provides commands for simply creating new folders directly in the repository

Create a Local Working Copy

In order to use Subversion to provide version control of the Packages in a model, you need to prepare a functional SVN working copy folder that can be accessed through an Enterprise Architect version control configuration within that model.

Create a Subversion working copy folder

Step	Action
1	Choose or create a suitable directory on your system in which to create your Subversion working copy.
2	Open a command prompt window and navigate to the directory you have selected. For example: C:\> mkdir mySVNWorkSpace C:\> mkdir mySVNWorkSpace/myEAModelName C:\> cd mySVNWorkSpace/myEAModelName
3	Perform the initial check out from the Subversion repository, specifying the repository folder and local working copy folder, as well as your user name and password. For example: C:\> svn checkout "https://myserver:8443/svn/repos_folder "C:\mySVNWorkSpace/myEAModelName" --username myUserName --password myPassword (By specifying your Subversion username and password, you ensure that they are correctly cached by Subversion and available for use by Enterprise Architect.) If you specify the HTTPS protocol when performing the initial Subversion check out, a prompt displays to accept a security certificate; in this instance, press the P key to permanently accept the certificate. The nominated local folder is configured as a Subversion working copy folder. Any files already existing in the repository folder are downloaded to the working copy folder as working copy files.
4	Verify that your Subversion environment functions correctly.

Notes

- It is important that Subversion caches your username and password, so that it never has to prompt for user input; a check-out operation might not request authentication, and if it does not you should perform an action that does request authentication, such as adding and committing a dummy test file

- The process can also be performed using TortoiseSVN's Checkout command (which provides options to browse when specifying both your repository folder and your local folder); when prompted for authentication details by TortoiseSVN, make sure you place a check against the 'Save Authentication Data' option

Verify the SVN Workspace

After creating the Subversion local working copy to hold the working copies of your Package files, you can verify that it functions correctly before attempting to use it with Enterprise Architect. You need to be able to add files to Subversion, lock the files and commit changes to those files.

Verify correct operation of a Subversion working copy folder

Step	Action
1	Use Windows to open a command prompt window.
2	Select the directory you specified as the working copy, in the Subversion checkout command when preparing the SVN workspace. For example: C:\> cd mySVNWorkSpace
3	Create a test file, such as *Test.txt*, containing the text *Subversion Test*. For example: C:\> echo Subversion Test > Test.txt
4	Execute each of these Subversion commands: • svn add Test.txt • svn commit -m"Commit comment" Test.txt • svn update Test.txt • svn lock Test.txt • Use your preferred editor to modify the file and save the changes • svn commit -m"Second commit comment" Test.txt The commands should execute without any errors and without prompting the user for any extra input.

Notes

- Your environment must be set up such that you can perform these operations without ever being prompted for user ID or password; for further information, please see the Caching Client Credentials topic in the official Subversion documentation

Subversion Under Wine-Crossover

If you want to set up and use Subversion with Enterprise Architect under Wine, the process is almost identical to the process for setting up and using the systems under Windows. When running Enterprise Architect under Wine or CrossOver, you still use a Windows-based Subversion command line client.

There are some minor differences in the way you prepare the Subversion working environment, specifically in the way you install your Subversion client and the way you check out a working copy folder from the Subversion repository.

System Requirements

Sparx Systems has tested the use of Enterprise Architect with Subversion under Wine 1.2, on macOS 10.4 and 10.6.2, and on Ubuntu 10.04. All tests were passed.

When using Wine 1.2 on the Ubuntu 9.10 platform, Sparx Systems was able to use the svn: and file: protocols to communicate with the SVN server; but not the https: protocol.

Installing a Subversion Client

Wine is able to install applications from either a Windows .exe file, or a .msi installer file.

Place the installer for your Windows Subversion client in a convenient location on the native file system, then open a Wine console window from within Enterprise Architect and run the installer from within the Wine console. Your Subversion installation can then access the same C: drive and folders that Enterprise Architect is accessing.

Preparing a Subversion Environment Under Wine

Under Wine, you can install Subversion from either a Windows .exe file, or a .msi file. By performing your Subversion installation and initial check out from a Wine console window opened from within Enterprise Architect, you have access to the same C: drive and folders that Enterprise Architect is accessing.

Set up Subversion for use with Enterprise Architect, running under Wine

Step	Action
1	Start Enterprise Architect. You do not have to load a project at this point.
2	From the Start ribbon, select 'Workspace > Preferences > Customize > Tools: ' A new, blank entry is opened on the 'Tools' tab of the 'Customize' dialog.
3	Define the new menu item entry: • In the newly-opened 'Menu contents' field, type the name 'Wine Console' • In the 'Command' field, type 'wineconsole' • In the 'Arguments' field, type 'cmd' • Leave the 'Initial directory' field blank
4	Click on the Close button. The 'Customize' dialog closes.
5	From the Configure ribbon, select 'User Tools > Wine Console'. A Wine console window opens.
6	Type 'C:' and press the Enter key. The Wine console switches to the C: drive.
7	Issue the command to install your Subversion client. For example: C:\>/Installers/Subversion-client-1.6.12-1.win32.exe To install from a .msi file, use Wine's msiexec utility. For example: C:\>msiexec /i "Slik-Subversion-1.6.9-win32.msi" Installation of the Subversion command line client begins.
8	Create a folder to serve as the working copy folder to be used by Enterprise Architect. For example: C:\>mkdir C:\VC_workspaces\SVN_workcopy
9	Issue the command to perform the initial checkout from the Subversion repository, specifying the repository folder, working copy folder, username and password. For example: C:\>svn checkout "https://myServer:8443/svn/repos_folder" "C:\VC_workspaces\SVN_workcopy " --username myUserName --password myPassword

	(After specifying your Subversion username and password, they are correctly cached by Subversion and are available for use by Enterprise Architect.)
	If the HTTPS protocol is specified when performing the initial Subversion check out, you are prompted to accept a security certificate; in this instance, press the P key to permanently accept the certificate.
	The nominated local folder is configured as a Subversion working copy folder. Any files already existing in the repository folder are downloaded to the working copy folder as working copy files.
10	Type 'Exit' and press the Enter key.
	The Wine console window closes.
	You are now ready to load a project in Enterprise Architect and apply version control to it, using the normal Windows-based procedures.

Notes

- You should copy the installer for your Windows Subversion client to a convenient location on the native file system, so that you can easily refer to it from within the Wine console window in step 7

TortoiseSVN

TortoiseSVN is a Windows shell extension for Subversion; it provides icon overlays in Windows Explorer that are useful as a tool for observing the status of your Subversion controlled files. You can also use it to create your repository sub-trees and check out local working copies from within Windows Explorer, using simple menu commands.

You can download TortoiseSVN from http://tortoisesvn.net/downloads.html.

Notes

- Enterprise Architect can only communicate with the Subversion server using the Subversion command line client svn.exe

Create a TFS Environment

You can use Microsoft Team Foundation Server (TFS) as a version control provider for Enterprise Architect. The first step in doing this is for a TFS administrator to install and configure the TFS server and client applications. A number of basic tasks are performed in creating an operational TFS environment.

Tasks in Creating a TFS Environment

Task	Detail
Obtain and install TFS	Enterprise Architect uses the TFS command line client to integrate TFS version control. The TFS command line client is normally available as part of your Visual Studio installation.
Choose a TFS project	It is good practice to create a new TFS project, or least a new Source Control Folder within a project, for each Enterprise Architect project being added to version control with TFS. If you have a single Enterprise Architect project that contains many different models (for example, a DBMS hosted project with multiple model root nodes), you might choose to create a new TFS project for each separate model. For further information, please consult your TFS product documentation.
Create a TFS workspace	A working copy folder must exist on each users' machine, for Enterprise Architect to use when exporting and importing the version controlled Package files. It is this folder that is specified as the Local Project Path, when defining your Version Control Configurations. The working copy folder is the 'sandbox' where you modify the controlled files. The working copy folder is usually associated with a folder that exists within the version control repository. In TFS, the TFS workspace is used to map a local working folder on your PC to a Source Control Folder within a TFS project. TFS 2012 and VS 2012 (and later versions) feature a new type of workspace called 'local' workspaces. Do not attempt to use TFS 'local' workspaces with Enterprise Architect. You must use only 'server' workspaces for Enterprise Architect version control, as 'local' workspaces do not support the application of checkout locks to files. Enterprise Architect relies on the presence of checkout locks to ensure that Packages can only be checked out exclusively and that a given Package is not already checked-out in some other project (for instance, in a Private Model deployment). This is necessary because it is not practical to merge the XMI Package files that Enterprise Architect uses for version control. A single TFS workspace can map many different local folders, each one to a separate Source Control Folder. In this case, TFS can take a long time to work through and update the files in all of those folders, and the system might appear to 'freeze' whilst it waits for TFS to hand back program control. You can avoid this if you keep your version controlled Package files in a folder that is separate from other artifacts, such as source code files, creating a separate work space to use just for your Package files, or creating and mapping a separate folder for Package files within an existing workspace.
Configure exclusive check-outs	The XMI format files used for the version control of Enterprise Architect's Packages can not be merged like ordinary text files. Therefore, Enterprise Architect must enforce serialized editing of its version controlled Packages. As a

	consequence, it is important that TFS is configured to use 'exclusive checkouts' for XML files.
Verify the TFS Workspace	Enterprise Architect uses the TFS command line client to checkin and checkout files from the TFS repository. After creating the TFS workspace, it is important to verify that the command line client can be used to add, check-in and check-out files residing in the working copy folder that is mapped through this workspace.

Notes

- TFS can also be used with an SCC client; the MS TFS-SCC client is available for download from Microsoft's web site

- MDG Integration for Visual Studio 2005 or 2008 enhances TFS support by providing access to, for example, work items and bugs within both Enterprise Architect and the MDG Integration product

TFS Workspaces

In order to use TFS to provide version control of the Packages in a model, you prepare a functional TFS workspace that can be accessed through an Enterprise Architect version control configuration within that model. You use the TFS workspace to map a local working folder on your PC to a Source Control Folder within a TFS project repository.

It is assumed that have TFS 2013 open and a TFS Team Project already exists for you to use in version controlling the Packages of your Enterprise Architect project. You can create the TFS workspace through MS Visual Studio.

TFS 2013 and later versions support the use of Local workspaces. However, Local workspaces do not support the application of checkout locks. For this reason, Sparx Systems strongly advises against the use of Local workspaces when version controlling your Enterprise Architect Package files. Using Local workspaces carries a high risk of merge conflicts when checking in, which would almost certainly result in loss of data or a corrupted model database.

Map a local folder to a TFS Source Control Folder

Step	Action
1	Connect to your TFS server. From the TFS main menu, choose 'View | Team Explorer'. In the Team Explorer toolbar, click on the Connect to Team Projects button (fourth from left, with a power plug icon). The 'Team Explorer Connect' page displays.
2	Click on the link 'Select Team Projects...' The 'Connect to Team Foundation Server' dialog displays.
3	Click on the appropriate Team Foundation Server, Team Project Collection and Team Project, then click on the Connect button.
4	Click on the Home page of the Team Explorer, and then on the Source Control Explorer button. The Source Control Explorer view displays.
5	In the Source Control Explorer, click on the drop-down arrow in the 'Workspace' field in the toolbar, then click on 'Workspaces' at the bottom of the list. The 'Manage Workspaces' dialog displays.
6	Click on the Add button. The 'Add Workspace' dialog displays.
7	Click on the Advanced button. A number of new fields display on the dialog.
8	Type in an appropriate name for the new workspace and, if required, type in a comment.
9	Set the 'Location' field to 'Server'. This setting is important. This setting is not available in TFS 2010 and earlier releases, where all workspaces are Server based.
10	Click in the Source Control Folder column, then click on the Browse button to select a Source Control Folder. Select the Source Control folder (in the Team Project) to use for controlling Enterprise Architect

	Packages.
11	Click on the Browse button in the Local Folder column and create a new local folder.
	This is the working copy folder into which Enterprise Architect exports the Package files.
12	Click on the OK button.
	The new workspace is created and saved.
	The 'Add Workspace' dialog closes.
13	Click on the OK button.
	The 'Manage Workspaces' dialog closes.

Notes

- The local folder referenced in step 11 is the Working Copy Path that should be specified when defining an Enterprise Architect Version Control Configuration to use this TFS workspace

TFS Exclusive Check Outs

The XMI format files used for the version control of Enterprise Architect's Packages can not be merged as ordinary text files can. Therefore Enterprise Architect must enforce serialized editing of its version controlled Packages, and it is important that Team Foundation Server is configured to use 'exclusive checkouts' for XML files. Otherwise, TFS can return file statuses that make it look as if the Package file is not checked-out by another user when indeed it is.

You set exclusive checkouts in the TFS workspace through MS Visual Studio.

Configure TFS to enforce exclusive check outs for XML files

Step	Action
1	Using Visual Studio, from the main menu select 'View | Team Explorer'.
2	In the 'Team Explorer' pane, right-click on the TFS Server name that is controlling the Enterprise Architect Package files, then from the context menu select 'Team Foundation Server Settings | Source Control File Types'.
3	Select the entry for XML files (or create an entry if necessary) then click on the Edit button.
4	Deselect the 'Enable file merging and multiple check out' checkbox.

Verify the TFS Workspace

Enterprise Architect will be using the TFS command line client to commit file updates from the local working copy folder into the TFS repository. After creating the TFS workspace that maps your local working copy folder to a TFS repository folder, it is good practice to verify that the TFS workspace functions correctly before attempting to use it with Enterprise Architect. You need to be able to add files to TFS, place a check-out lock on the files and check-in changes to those files.

Verify correct operation of a TFS Workspace

Step	Action
1	Use Windows to open a command prompt window.
2	Select the directory you specified as the local working copy folder when creating the TFS workspace. For example: C:\> cd myTFSWorkingCopyFolder
3	Create a test file, such as *Test.xml*, containing the text "*my TFS Test"*. For example: C:\> echo my TFS Test > Test.xml (The filename extension ".xml" is used because the TFS server should have been configured to allow exclusive checkouts for XML files.)
4	Using the command line client, execute each of these TFS commands: • tf.exe add Test.xml • tf.exe checkin Test.xml /override:"Manual check in." /comment:"a message" • tf.exe checkout /lock:checkout Test.xml Use your preferred editor to modify the file and save the changes. • tf.exe checkin Test.xml /override:"Manual check in." /comment:"a message" • tf.exe get . /recursive

Notes

- All of the commands in step 4 should execute without any errors; in particular, you should be able to apply a checkout lock when checking out the file
- If the checkout command reports any errors, you might have to reconfigure the TFS workspace as a server based workspace, or you might have to configure the TFS Server to use 'exclusive checkouts' for XML files

Create a CVS Environment

You can use Concurrent Versions System (CVS) as a version control provider for Enterprise Architect. The first step in doing this is for a CVS administrator to install and configure the appropriate software. A number of basic tasks are performed in creating an operational CVS environment, and useful tools are available for performing some of these tasks.

Tasks in Creating a CVS Environment

Task	Detail
Install server components	Executable files for CVS can be obtained from March Hare Software.
	CVS server components are available to run on a wide range of different hardware and operating systems; Enterprise Architect is not affected by your choice of server components.
Create a repository	Please consult the official CVS documentation.
Create CVS users	Please consult the official CVS documentation.
Create a new repository module	It is recommended good practice to create a new repository module in CVS for each new Enterprise Architect model being added to version control with CVS. Users should create a new local working copy folder from the module to be used with that model.
	A repository module represents a project, or a set of related files in the repository.
	TortoiseCVS can greatly simplify the process of creating new repository sub-trees.
Install client components	Executable files for CVS can be obtained from March Hare Software.
	Enterprise Architect is a Windows based application - it requires a Windows based CVS command line client for integration.
Create a working copy folder	A working copy folder must exist on each users' machine, for Enterprise Architect to use when exporting and importing the version controlled Package files. It is this folder that is specified as the Local Project Path, when defining your Version Control Configurations.
	The working copy folder is the 'sandbox' where you modify the controlled files. The working copy folder is usually associated with a folder that exists within the version control repository. In CVS, to create a local working copy you perform an initial check-out of a folder from the CVS repository; this downloads a copy of the folder and its contents, to create your local working copy.
	TortoiseCVS can greatly simplify the initial check out of a working copy folder.
Setting up CVS under Wine/CrossOver	The process of setting up and using CVS with Enterprise Architect under Wine is almost identical to the process when running natively under Windows, apart from minor differences in installing the CVS client and performing the initial checkout of the working copy folder.

Notes

- If you do not already use CVS for version control, you should consider using Subversion instead; Subversion's client-server protocols provide a broader range of possibilities for connecting to remote servers, with easier set up of secure connections

- TortoiseCVS is a Windows shell extension; Enterprise Architect cannot use TortoiseCVS as its client, it must use the CVS command line client

Prepare a CVS Local Workspace

In order to use CVS to provide version control of the Packages in a model, you need to prepare a functional CVS working copy folder that can be accessed through an Enterprise Architect version control configuration within that model.

Prepare a CVS Working Copy Folder

Step	Action
1	Ask your System Administrator to install CVS and create a remote repository, with a module that you can use to control your Enterprise Architect Package files. Your administrator must create a username and password for you before you can make a connection.
2	Select or create a suitable directory to use as your CVS working copy directory.
3	Open a command prompt window and navigate to your CVS working copy directory. For example: C:\> mkdir myCVSWorkSpace C:\> cd myCVSWorkSpace
4	Log in to the remote CVS repository. For example: C:\myCVSWorkSpace> cvs -d:pserver:myUserID@ServerName:/reposPath login Replace *myUserID* with your CVS username, replace *ServerName* with the name of your CVS server and replace *reposPath* with the path to the repository on the server. A prompt for a password displays.
5	Enter your password. You are logged in to the CVS server.
6	Perform the initial checkout of the CVS repository module, into the local working copy directory. For example: C:\myCVSWorkSpace> cvs -d:pserver:myUserID@ServerName:/reposPath checkout moduleName (Replace *moduleName* with the name of the repository module that you want to check out.) A subdirectory is created in your current working directory, with the same name as the module being checked out. Any files already existing in the repository module are downloaded to the working copy folder as working copy files.
7	Verify that your CVS environment functions correctly.

Notes

- Much of the process can also be performed (more simply) using the TortoiseCVS command 'Make New Module'

Verify the CVS Workspace

After creating the CVS local working copy to hold the working copies of your Package files, you can verify that it functions correctly before attempting to use it with Enterprise Architect. You need to be able to add files to CVS, and commit changes to those files. You also need to be able to register as an editor of the file and retrieve the list of currently registered editors.

Verify correct operation of a CVS working copy folder

Step	Action
1	Use Windows to open a command prompt window.
2	Select the directory you specified as the working copy in the cvs checkout command, when preparing the CVS workspace. For example: C:\> cd myCVSWorkSpace
3	Create a test file, such as *Test.txt*, containing the text *CVS Test*. For example: C:\> echo CVS Test > Test.txt

4	Execute these CVS commands:
	• cvs add Test.txt
	• cvs commit -m"Commit comment" Test.txt
	• cvs update Test.txt
	• cvs edit Test.txt
	• cvs editors Test.txt
	The commands should execute without any errors and without prompting the user for any extra input.
	The editors command should produce output that resembles this:
	   Test1.txt myUserID Tue Aug 9 10:08:43 2009 GMT myComputer
	   C:\myCVSWorkSpace\moduleName

5	Take note of the userID that follows the filename. Enterprise Architect must find and use this user ID when you create your version control configuration.

Notes

• Your environment must be set up such that you can perform these operations without ever being prompted for input, such as user ID or password

TortoiseCVS

TortoiseCVS is a Windows shell extension for CVS; it provides icon overlays in Windows Explorer that are useful as a tool for observing the status of your CVS controlled files. You can also use it to create your repository modules and check out local working copies from within Windows Explorer using simple menu commands.

You can download TortoiseCVS from: http://www.tortoisecvs.org.

Notes

- Enterprise Architect must use the CVS command line client to communicate with the CVS server; it cannot use TortoiseCVS

Create an SCC Environment

You can use a Microsoft Common Source Code Control (SCC) compatible product as a version control provider for Enterprise Architect. The first step in doing this is for an administrator to install and configure the server and client applications. A number of basic tasks are performed in creating an operational SCC-based environment.

Tasks in Creating an SCC Environment

Task	Detail
Install and configure your chosen version control product	A version control server component is typically installed on a dedicated server machine. All Enterprise Architect users who require access to version control must be able to connect to the server machine.
	After installing the version control software, the administrator should also create version control user IDs for all users who require access to version control.
	For further information, consult the product documentation for your particular version control product.
Create a new SCC project	It is good practice to create a new SCC version control project, or least a new folder within a project, for each Enterprise Architect project being added to version control with SCC.
	If you have a single Enterprise Architect project that contains many different models (for example, a DBMS hosted project with multiple model root nodes), you might choose to create a new SCC version control project for each separate model.
	For further information, consult the product documentation for your particular version control product.
Configure your SCC project to support exclusive check-outs for .XML files	The XMI-format files used for the version control of Enterprise Architect Packages can not be merged in the same way as ordinary text files can. Therefore, Enterprise Architect must enforce serialized editing of its version controlled Packages. As a consequence, it is important that your version control application is configured to use 'exclusive checkouts' for XML files.
Create a local working copy folder	A working copy folder must exist on each users' machine, for Enterprise Architect to use when exporting and importing the version controlled Package files. It is this folder that is specified as the Local Project Path, when defining your Version Control Configurations.
	The working copy folder is the 'sandbox' where you modify the controlled files. The working copy folder is usually associated with a folder that exists within the version control repository. Your version control product provides some means by which you associate a working copy folder with a repository folder.
	For further information, consult the documentation for your particular version control product.

Notes

- When installing the client component software on users' PCs, check that the SCC client is also installed, as it might not be a part of the default installation

Upgrade at Enterprise Architect Version 4.5, Under SCC Version Control

If you are working in Enterprise Architect release 4.5 or later and you open an SCC version-controlled project that was created under a release earlier than 4.5, you must identify the SCC connection with a new unique ID. You can assign a name to the existing SCC configuration or associate the project with another configuration that has previously been assigned a unique ID.

By having a unique ID for each version control configuration, you can assign a configuration quickly and efficiently using configurations that have been created previously for other version controlled repositories. This enables you to configure the many Packages to use an existing version control repository; this can apply to Packages created for more than just one model enabling a great deal of flexibility.

Upgrade an existing SCC version controlled project created before release 4.5, in Enterprise Architect release 4.5 or later

Step	Action
1	Open the project that has an SCC version control configuration created in Enterprise Architect earlier than version 4.5. The 'Select or Create Unique ID for Version Control' dialog automatically displays.
2	On the dialog, either create an ID for an existing configuration or choose a previously created one from the 'Unique ID' drop-down list.
3	The existing SCC configuration is the initial value, represented by SCC-XXXXX; this number is not especially meaningful, therefore it is recommended that the configuration be given a meaningful name.
4	You can associate the version controlled Package with a previously-defined configuration by selecting an existing configuration from the Unique ID drop-down list (if one exists).
5	After you have assigned the unique ID, click on the OK button to load the model.

Version Control Setup

Once you or an Administrator have installed and configured the version control product software, to start using version control you must first define a version control configuration within your project in Enterprise Architect, to be used to control the Packages in the project. You can define any number of version control configurations in a single model, but any given Package is associated with only one configuration.

Access

Ribbon	Configure > Version Control > Settings
Context Menu	Right-click on Package > Package Control > Version Control Settings

Define Version Control Configuration

Step	Action
1	On the 'Version Control Settings' dialog, click on the New button.
2	In the 'Unique ID' field, type a suitable configuration name.
3	Select the Type of version control product you are connecting to, by clicking on the corresponding radio button.
4	At this point, the middle section of the dialog changes to display a collection of fields specific to the type of version control configuration you are defining. Enter details relating to the version control workspace that this configuration is to use.
5	Click on the Save button. The new configuration is added to the Defined Configurations list.
6	If you want to create another version control configuration, return to step 1. When you have finished defining your version control configurations, click on the Close button.

Notes

- Version control configuration details are stored in the user's Windows Registry settings, but each project stores a list of the configurations it uses, so that version control connections can be initialized as the project is being loaded

- If you are using the Corporate or extended editions of Enterprise Architect with security enabled, the administrator must also set up access permissions to configure and use version control

Re-use an Existing Configuration

Once a version control configuration has been defined for use in one project, it is possible to re-use that configuration in other projects to provide access to:

- An already existing version control environment (a working copy directory and its associated repository that is already in use)
- Version controlled Packages that were created (and version controlled) in another project

Access

Ribbon	Configure > Version Control > Settings
Context Menu	Project Browser | right-click on Package | Package Control | Version Control Settings

Select existing configuration

Step	Action
1	On the 'Version Control Settings' dialog, click on the New button. The various fields on the dialog are cleared, ready for data entry.
2	In the 'Unique ID' field, click on the drop-down arrow and select one of the previously-defined version control configurations. The details of the selected configuration are displayed in the dialog.
3	Click on the Save button. The configuration details are saved and are ready for use in the current project.

Version Control Settings

Defining version control settings within Enterprise Architect should only be performed after you have successfully created the version control environment that you intend to use to control your Package files. If you have not yet done so, please refer to the Help topic *Applying Version Control in a Team Environment* for a summary of what is required.

To set up a version control configuration on your model, or update an existing version control configuration, you define a number of settings that control how the status of your model is communicated to your version control system. You define these settings using the 'Version Control Settings' dialog.

Access

Ribbon	Configure > Version Control > Settings
Context Menu	Right-click on Package > Package Control > Version Control Settings

Configuration Options

Option	Action
This model is private	Select to specify that this model database is to be accessed by just a single user (Private Model).
	Leave unselected (the default) or deselect to specify that the database is to be accessed by multiple concurrent users (Shared Model).
	If in doubt, use the default setting.
Save nested version controlled packages to stubs only	Select to specify that the exported XMI file for a version controlled Package will contain Package stubs (place holders) for nested version controlled child Packages (recommended).
	Deselect to specify that the exported XMI file will contain the full content of nested version controlled child Packages.
For all packages, create placeholders for external references	Select to force all XMI 1.1 imports across the model to exclude incoming relationships and instead create external references.
	If the 'Create placeholders for missing External References during XMI 1.1/2.1 Import' checkbox is not selected in the XML Specifications options for a user, this field overrides that setting.
Unique ID	Specify a name that uniquely identifies the configuration. Either:
	• Type a name to identify a new configuration, or
	• Click on the drop-down arrow and select the name of a configuration previously defined in a different project (if any exist)
Type	Click on the appropriate radio button for the type of version control system you are associating with this configuration.
	The middle section of the dialog changes to display a collection of fields relating to the type of version control configuration you are defining.

	Set the type to SCC for: • MS Visual Source Safe • Rational Clear Case • Perforce • AccuRev • Any other SCC-compatible clients For any other product that you are using, select the type that matches the product - CVS, Subversion or TFS.
New	Click on this button to clear the fields and create a new version control configuration.
Save	Click on this button to save the details of a new or updated configuration.
Delete	Click on an entry in the Defined Configurations list and click this button to remove the definition of the selected configuration from this model.
Defined Configurations	Review a list of configurations that are in use in the current model.
In future, do not prompt for incomplete configurations	Select to specify that the user is not prompted to complete the definition of configurations that are not fully specified (the default). Deselect to prompt the user to complete configurations that are not fully defined.
Close	Close the 'Version Control Settings' dialog.
Help	Display this Help topic.

Notes

- It is important that, for any given version controlled Package file, any user accessing that file from any model uses version control configurations having the same Unique ID
- When you first open a model that was created by another user and that uses version control, the version control configuration(s) used by that model do not yet exist in your Windows registry settings; you need to complete the definitions of those configurations before you can use version control in that project
- If User Security is enabled, you must have 'Configure Version Control' permission to set up version control options for the current model
- It is possible to use multiple version control configurations in the same model

SCC Settings

When you are setting up your version control configurations on the 'Version Control Settings' dialog, and you set the configuration type to 'SCC', the dialog presents a set of fields specific to SCC-based configurations. You can then define details such as:

- The working copy folder to be used with the configuration
- The details necessary to connect to the SCC version control system

You set the version-control configuration type to SCC for version control providers such as:

- MS Visual Source Safe
- Rational Clear Case
- Perforce
- AccuRev
- Any other SCC-compatible clients

Access

Ribbon	Configure > Version Control > Settings: Type: SCC
Context Menu	Right-click on Package > Package Control > Version Control Settings: Type: SCC

Settings

Option	Action
Local Project Path	Displays the full path of the folder that contains the local (working) copies of the XMI Package files. This field is read-only; its value can only be set through use of the Select Path button.
Select Path	Click on this button to choose the Local Project Path, by opening a file browser dialog and navigating through the file system to the appropriate folder. • After you choose the appropriate folder path, the 'Select SCC Provider' dialog displays, listing all SCC providers that are installed on the current workstation; choose the SCC provider to use and click on the OK button • At this point, the SCC client opens its own dialog to prompt you for information; SCC products implement this functionality in varied ways, but typically you are prompted to log in to the version control system, then prompted to choose the SCC project to use and possibly a (server) folder contained within that project • At the conclusion of this process, all of the SCC details should be filled in; you can then save the definition by clicking on the Save button on the 'Version Control Settings' dialog
Current User	Displays the user name used to log on to the version control system that is accessed through this configuration.

	This field is read-only; the value it displays is retrieved from the SCC client.
SCC Provider	Displays the name of the SCC provider.
	This field is read-only; the value it displays is retrieved from the SCC client.
SCC Project	Displays the name of the SCC Project that this configuration attaches to.
	This field is read-only; the value it displays is retrieved from the SCC client.

Notes

- You define the SCC-specific details as part of the broader process of setting up a version control configuration on the 'Version Control Settings' dialog

CVS Settings

When you are setting up your version control configurations on the 'Version Control Settings' dialog, and you set the configuration type to CVS, the dialog presents a set of fields specific to CVS-based configurations. You can then define details such as:

* The working copy folder to be used with the configuration
* The path to the CVS command line client

Access

Ribbon	Configure > Version Control > Settings: Type: CVS
Context Menu	Right-click on Package > Package Control > Version Control Settings: Type: CVS

Settings

Option	Action
Working Copy Path	Displays the full path of the folder that contains the local (working) copies of the XMI Package files. This field is read-only; its value can only be set through use of the Select Path button.
Select Path	Click on this button to choose the working copy path, by opening a file browser dialog and navigating through the file system to the appropriate folder.
Current User	This field is read-only; its value is retrieved from a file named CVS\Root, located in the folder selected using the Select Path button.
CVS Exe Path	Displays the full path of the CVS command line client executable file. This field is read-only; its value can only be set through use of the Select Path button.
Select Path	Click on this button to specify the path to the CVS command line client, by opening a file browser dialog and navigating through the file system to locate the appropriate file.
VC Time-Out Value	Specify the amount of time that Enterprise Architect waits for a CVS command to complete; if the command does not complete within the allowed time, the system displays a Time-out error message, detailing the command that failed to complete. This single time-out value is applied to all Version Control Configurations (of types SVN, TFS and CVS) that the user accesses from this workstation.

Notes

- When connecting to a remote CVS repository, the 'Current User' field should display the user name used to log into that repository; if this does not happen, it indicates that Enterprise Architect cannot extract the user name from the file ...\WorkingCopyPath\CVS\Root and the configuration does not work correctly

- You define the CVS-specific details as part of the broader process of setting up a version control configuration on the 'Version Control Settings' dialog

SVN Settings

When you are setting up your version control configurations on the 'Version Control Settings' dialog, and you set the configuration type to 'Subversion', the dialog presents a set of fields specific to Subversion-based configurations. You can then define details such as:

- The working copy folder to be used with the configuration
- The path to the Subversion command line client

Access

Ribbon	Configure > Version Control > Settings: Type: Subversion
Context Menu	Right-click on Package > Package Control > Version Control Settings: Type: Subversion

Settings

Option	Action
Working Copy Path	Displays the full path of the folder that contains the local (working) copies of the XMI Package files. This field is read-only; its value can only be set through use of the Select Path button.
Select Path	Click on this button to choose the Working Copy Path, by opening a file browser dialog and navigating through the file system to the appropriate folder.
Subversion Exe Path	Displays the full path of the Subversion command line client executable file. This field is read-only; its value can only be set through use of the associated Select Path button.
Select Path	Click on this button to specify the path to the Subversion command line client, by opening a file browser dialog and navigating through the file system to locate the appropriate file.
VC Time-Out Value	Specify the amount of time that Enterprise Architect should wait for a Subversion command to complete; if the command does not complete within the allowed time, the system displays a Time-out error message, detailing the command that failed to complete. This single time-out value is applied to all Version Control Configurations (of types SVN, TFS and CVS) that the user accesses from this workstation.

Notes

- You define the Subversion-specific details as part of the broader process of setting up a version control configuration on the 'Version Control Settings' dialog

TFS Settings

When you are setting up your version control configurations on the 'Version Control Settings' dialog, and you set the configuration type to 'TFS', the dialog presents a set of fields specific to TFS-based configurations. You can then define details such as:

- The working copy folder to be used with the configuration
- The user name and password to log in to the TFS server
- The path to the TFS command line client

Access

Ribbon	Configure > Version Control > Settings: Type: TFS
Context Menu	Right-click on Package > Package Control > Version Control Settings: Type: TFS

Settings

Option	Action
Working Copy Path	Displays the full path of the folder that contains the local (working) copies of the XMI Package files. This field is read-only; its value can only be set through use of the associated Select Path button.
Select Path	Click on this button to choose the Working Copy Path. The file browser dialog opens, through which you navigate through the file system to the appropriate folder.
Server Name	Displays the name of the TFS Server that is associated with the working copy folder specified in the 'Working Copy Path' field. This field is read-only; Enterprise Architect retrieves the value it displays by querying the TFS client.
Workspace Name	Displays the name of the TFS Workspace that is associated with the working copy folder specified in the 'Working Copy Path' field. This field is read-only; Enterprise Architect retrieves the value it displays by querying the TFS client.
User Name	Type in the user name with which to log into the TFS Server.
Password	Type in the password with which to log into the TFS Server.
Display Name	TFS 2013 and later releases use a Display Name to report on who owns a checkout lock on a file. TFS 2010 uses AccountID when reporting lock owners and so does not require a Display Name. If using TFS 2013, type in your TFS Display Name as shown in the User column of

	the Visual Studio Source Control Explorer when you checkout a file.
Checkout Locks Required	This checkbox defaults to selected.
	TFS 2013 supports the use of Local workspaces, but these do not support checkout locks. If you want to use TFS Local workspaces (not recommended), you must deselect this checkbox. Otherwise, leave the checkbox selected.
	It is recommended that all workspaces used for version controlling Enterprise Architect Package files are set as 'Server' workspaces.
TFS Exe Path	Displays the full path of the TFS command line client executable file.
	This field is read-only; its value can only be set through use of the associated Select Path button.
Select Path	Click on this button to specify the path to the TFS command line client.
	The file browser dialog opens, through which you navigate through the file system to the appropriate folder.
VC Time-Out Value	Type in the number of seconds that Enterprise Architect should wait for a TFS command to complete; if a command does not complete within this allowed time, the system displays a Time-out error message, detailing the command that failed to complete.
	This single time-out value is applied to all version control configurations (of types SVN, TFS and CVS) that the user accesses from this workstation.

Notes

- If you automatically log in to TFS through a path external to Enterprise Architect (for example, through MS Integrated Security), you can leave the 'User Name' and 'Password' fields blank

- If the 'Password' field is blank, Enterprise Architect retrieves your Windows username and uses that value when determining whether a Package file is checked out to you or to another user

- TFS version control can also be accessed using the TFS MSSCCI client; to make use of the TFS MSSCCI client, please define an SCC based version control configuration

- You define the TFS-specific details as part of the broader process of setting up a version control configuration on the 'Version Control Settings' dialog

Use Version Control

Once your version control product is installed and you have created a suitable environment, Enterprise Architect can make use of that environment to control the Packages in your project. Version control provides a range of facilities, as outlined in this table.

Facilities

Facility	Detail
Define Version Control Settings	Version control configurations are used by Enterprise Architect to communicate with your version control system. You define one or more version control configurations in your project and then use those configurations to control the Packages in your project.
Configure a Package	To put a Package under version control you mark the Package as a controlled Package, specify the version control configuration to control it, and associate an XMI file with the Package.
Check In a Model Branch	Checks in all Packages involved in a particular unit of work, as a single operation. Checking-in updates the reference version of a Package or group of Packages in the model.
Check Out a Model Branch	Checks out all Packages within a selected model branch as a single operation, so that you can update modeling objects within them.
Check In a Package	Checks in the Package currently selected in the Project Browser. Checking-in updates the reference version of a Package or group of Packages in the model.
Check Out a Package	Checks out the version controlled Package currently selected in the Project Browser, so that you can update modeling objects within it.
Undo Check Out of a Package	Reverses the check-out of a Package, discarding any modifications that have been made by restoring the Package content to the latest revision held in version control.
Import a Package From Version Control	Retrieves Packages from version control that have been created by other users, or by you in another model, and imports them into your current model.
Apply Version Control to a Model Branch	Applies version control to all Packages within the selected model branch, in a single operation. In this context, a model branch is a Package in the Project Browser, and all of the Packages contained within it.
Export a Version Controlled Model Branch	Exports version control information about the root Package of a model branch, that is used to simplify the process of exporting and importing a hierarchy of Packages from one model to another.
Import a Model Branch From Version Control	Uses Enterprise Architect's Model Branch files, of which there are few, to retrieve information about the root Package file and to import the model branch. Model branch files can simplify the process of exporting and importing a hierarchy

	of Packages from one model to another.
View Package Revision History	Displays the change history of version controlled Packages. You can also check out a prior revision of the Package for editing, effectively rolling-back to a prior revision of the Package.
Validate Package Configurations	You can test the validity of the version control settings associated with each version controlled Package within your current model.
Resynchronize the Status of Version Controlled Packages	Re-synchronizes the version control status of Packages within your project with the status reported by your version control provider.

Notes

- Database replication should not be combined with version controlled Packages
- If the Packages under version control contain any alternative images, you can export the images to the version control repository when you check in the Packages, by setting the 'Export alternate images' option on the 'Preferences' dialog

Configure Controlled Package

Once your version control application is set up and you have version control configurations in place, you can place the individual Packages in your model under version control. To put a Package under version control, you:

- Flag the Package as a controlled Package
- Specify the version control configuration to control it and
- Associate an XMI file with the Package

You can then export and import the Package data to and from the file and issue commands to the version control system.

Access

Ribbon	Configure > Version Control > Configure Package
Context Menu	Right-click on Package > Package Control > Configure
Keyboard Shortcuts	Ctrl+Alt+P

Apply version control to a single Package

Step	Action
1	Click on the 'Control Package' checkbox to select it, indicating that this Package is to be controlled.
2	Click on the Version Control drop-down arrow and select the version control configuration to be used to control this Package.
3	The 'XMI Filename' field displays a default filename for the Package export file, based on the Package name. Optionally, modify the filename. Either: • Type a new name, or • Click on the [...] button to open a file selection dialog The target file must be located within the working copy folder of the selected version control configuration, or one of its sub-folders.
4	The 'Version ID' field defaults to '1.0'. Optionally, change this to a different version number.
5	The 'Owner' field defaults to your user name. Optionally, type or select the name of the user who actually owns the Package.
6	Click on the OK button. The 'Add Package to Version Control' dialog displays.

7	Optionally, clear the 'Keep checked out' checkbox.
	After applying version control, the Package either remains checked-out for editing, or is checked-in and locked against editing, depending on this setting.
8	Click on the OK button.
	The 'Add Comment' dialog displays.
9	Optionally, add any further comments to the default comment.
	Enterprise Architect provides a default comment that includes the current date & time.
10	Click on the OK button.
	The current Package is exported to the nominated XMI file, which is then committed to version control. The Package icon in the Project Browser is updated to reflect the Package's version control status.

Notes

- If you are using the Corporate or extended editions of Enterprise Architect with security enabled, these features are only available to users who have been granted permission to configure and use version control

Apply Version Control To Branches

It is possible to apply version control to all Packages within a selected model branch, in a single operation. In this context, a model branch is a Package that is currently selected in the Project Browser, and all of the Packages contained within it.

Access

Context Menu	Project Browser	right-click on Package	Package Control	Add Branch to Version Control

Apply version control to all Packages within a selected model branch

Step	Action
1	On the 'Apply VC to Branch' dialog, click on the drop-down arrow in the 'Version Control Configuration' field and select the configuration to use.
2	Optionally, tick the checkbox 'Export as Model Branch'. Once the version control operation is complete a Model Branch file (.EAB file) is created for this branch.
3	Click on the OK button. The system creates a number of sub-folders within the version control working copy folder, then exports all of the Packages within the selected model branch. The system generates filenames for the XMI files, based on the Package GUIDs.

Notes

- The version control configuration to be used in this operation must be defined within the model before selecting this command
- When invoked on the model root node, this command applies version control to every Package within the model

Package Version Control Options

When you have set up a Package for version control, you gain access to a range of version control operations that can be performed on that Package, such as:

- Open the dialog for working with baselines of the Package
- Check-in and check-out single Packages or a selected hierarchy of Packages
- Update Packages to the latest revision from the version control repository
- Inspect the revision history or properties of the XMI file associated with a Package
- Revert a Package to a previous revision
- Compare the current model content of a Package, against the latest revision of the Package in version control
- Import and export hierarchies of Packages (model branches) to and from the model, through the version control system
- Synchronize the status of a Package, with the version control system

Access

Context Menu	Right-click on version controlled Package > Package Control

Options

Option	Action
Configure	Apply or remove version control for the selected Package (specify version control settings) or specify a file for use in XMI Package Control. Shortcut: Ctrl+Alt+P
Package Baselines	Create a Baseline of the current Package, or compare the current Package with a previous Baseline. Shortcut: Ctrl+Alt+B
Check In Branch	Check-in Packages contained in the currently selected model branch (that is, the selected Package and all of its child Packages). The 'Select Packages to Check In' dialog lists all version controlled Packages within that branch that are checked out to you; you can then select Packages in the displayed list, to be submitted for check-in. You can also choose to keep the Packages checked-out after committing a new revision to version control.
Check Out Branch	Recursively check out all Packages contained within the currently selected model branch (that is, the selected Package and all of its child Packages) that are version controlled and checked-in.
Check In	Commit a new revision of the currently selected Package to the version control repository and lock the Package against further editing.

	Only available for Packages that you have checked-out yourself.
Check Out	Synchronize the currently selected Package with the latest revision from the version control repository and unlock the Package to allow editing. Only available for Packages that are not already checked-out (and whose associated Package file is not checked-out).
Undo Check Out	Restore the selected Package to the latest revision in the version control repository and lock the Package against further editing.
Put Latest	Commit a new revision of the currently selected Package to the version control system, while keeping the Package checked-out. This is equivalent to checking a Package in and immediately checking it back out again. Only available for Packages that you have checked-out yourself.
Get Latest	Synchronize the currently selected Package with the latest revision from the version control repository. Available only for Packages that are checked in.
Get All Latest	Update all of the version controlled Packages in the project, to the latest revision retrieved from version control. Only updates Packages that are currently checked in. Once the latest revisions are retrieved, the system scans all the controlled Packages and fixes any missing cross-references by comparing the Package with its XMI 1.1 file. If the cross-reference information in the XMI does not match the model, the system updates the model with the information from the XMI and records this update in the System Output window. You can roll back such updates by selecting the entry in the System Output window and using the context menu option 'Rollback Update' (or 'Rollback Selected Updates' if multiple entries are selected). • Closing the model clears the entries in the System Output window • An entry in the System Output window is also cleared as and when you roll-back the update for it
Scan XMI and Reconcile Model	Scan the Package XMI files associated with each of the project's controlled Packages and restore any diagram objects or cross-references that are detected as missing from the project. This function is useful in team environments where each user maintains their own private copy of the model database (that is, multiple private project files) and model updates are propagated through the use of controlled Packages. It provides no benefit when the model is hosted in a single shared database that is accessed by all team members. Each controlled Package is compared with its associated XMI file and, if the cross-reference information in the model does not match the XMI, the system updates the model with the information from the XMI and records the update in the System Output window. You can roll back such updates by right-clicking on the entry in the System Output window and selecting the 'Rollback Update' option (or 'Rollback Selected Updates' if multiple entries are selected). Closing the model clears the entries in the System Output window; an entry in the window is also cleared as and when you roll-back the update for it.

	This functionality is invoked automatically as part of the 'Get All Latest' operation.
	When working in an environment that uses a Private Model deployment and your model contains a significant number of cross-Package references, it is recommended that you invoke 'Scan XMI and Reconcile Model' from time to time, following the re-importation of controlled Packages - for example, after using 'Get Latest' to update a number of Packages - or after performing a number of Package check-outs.

	• As a general rule, avoid running 'Scan XMI and Reconcile Model' while you have uncommitted changes in your model; generally, you:
	   - Check-out a number of Packages
	   - Invoke 'Scan XMI and Reconcile Model'
	   - Make your modifications
	   - Commit any outstanding changes before you check-out more Packages and run 'Scan XMI and
	       Reconcile Model' again

File Properties	Display version control properties pertaining to the XMI export file associated with the currently selected Package; this also identifies who has checked out the Package.
File History	Display change history information for the currently selected Package. Revert to or check-out a prior revision of the Package.
Compare with Controlled Version	Compare the currently selected Package with the latest revision of its associated XMI file retrieved from version control.
Add Branch to Version Control	Apply version control to all Packages within a selected model branch, in a single operation. In this context, a model branch is a Package that is currently selected in the Project Browser, and all of the Packages contained within it.
Export as Model Branch	Export a newly created model branch from your own private copy of a model.
Import a Model Branch	Retrieve a model branch and import it into either the source model or another model.
Get Package	Access Packages in the version control repository that are not currently available in your model.
Re-synch Status With VC Provider	Update the version control status value recorded for the selected Package in the project to match the value reported by the version control provider, without performing an XMI import or export. Use this function when the Package's version control status recorded in your project is out of synchrony with the version control status reported by your version control provider.
Version Control Settings	Display the 'Version Control Settings' dialog.

Notes

• You set up version control using options from the project 'Version Control' submenu

- If the selected Package is not under version control, a different set of options is available
- If a version control configuration has not been defined for the model, no options for using version control are available, only the options for configuring version control

Check Out a Package

When you need to work on a version controlled Package, you check it out. The local XMI file associated with the Package is then checked-out from version control. No other user can check out the Package to make changes to it until it has been checked in again.

Access

Context Menu	Right-click on Package > Package Control > Check Out
Keyboard Shortcuts	Ctrl+Alt+L

Check out a single Package

The Package file is imported into your model, and the 'Package' icon is updated to reflect the change in the Package's version control status.

When working in a Private Model, if the system detects that the Package content in the model is already up to date with the latest revision of the Package file retrieved from version control, then the 'Import Package' dialog displays first. This dialog is not displayed for a Shared Model.

These options are available:

- Force Reload From XMI - reload the Package from XMI regardless of whether it is up to date or not

- Accept current package - select to skip the process of re-importing the Package from XMI

- Refresh model view - select to refresh the Project Browser and diagrams, by reloading the Package content from the project database

- Always use these settings - when selected, if you subsequently check out a Package that is found to be up to date, the same settings are applied again without displaying the dialog

Notes

- If you check out a version controlled Package whilst offline, the 'Package' icon has a red figure 8 in front of it

- If you have selected the 'Always use these settings' checkbox and you want to reconfigure the 'Import Package' dialog, press the Ctrl key whilst you select the 'Package Control | Check Out' context menu option; the dialog displays and you can change the settings

Undo Check Out of a Package

If you check out a Package and then decide not to proceed, you can undo the check-out and discard any modifications that have been made, by restoring the Package content to the latest revision held in version control. The Package returns to a checked-in state and subsequently can be checked out by any user, including yourself if, for example, you need to reverse incorrect changes before checking the Package out and starting again.

Access

Context Menu	Right-click on Package > Package Control > Undo Check Out

Undo check out of a selected Package

A confirmation dialog displays; click on the OK button.

The latest revision of the Package is retrieved from version control and re-imported into your model. The icon against the Package in the Project Browser is updated to reflect the change in the Package's version control status.

Check In a Package

When you have finished working on the contents of a Package under version control, and you want to return it to the model for other users to see, you check it in.

Access

Context Menu	Right-click on Package > Package Control > Check In
Keyboard Shortcuts	Ctrl+Alt+S

Check in a single Package

Step	Action
1	The selected Package is exported and the 'Add Comment' dialog displays. A default comment is provided that contains the current date and time. Optionally, modify the default check-in comment
2	Click OK. The Package file is checked-in to version control and the Package icon is updated to reflect the change in version control status.

Check Out a Model Branch

If you need to check out a number of related Packages involved in a particular unit of work, to update the contents, you can do so in a single operation by checking out the whole model branch that contains them.

Access

Context Menu	Right-click on Package > Package Control > Check Out Branch

Check out a sub-tree of model Packages

Step	Action
1	The selected root-node Package and all of its contained sub-Packages are recursively checked out. Any Packages that cannot be checked-out are listed in a message box, with a brief description of the problem; for example: *The Package is already checked out by user 'Fred'.*
2	When Project Security is enabled in Lock to Edit mode, Enterprise Architect prompts you to apply a User Lock throughout the selected model branch before proceeding.

Check In a Model Branch

If you need to check in a number of related Packages involved in a particular unit of work, and that you have updated, you can do so in a single operation by checking in the whole model branch that contains them. You can also commit new revisions of the affected Packages as you complete milestones, whilst keeping the Packages checked-out for further editing.

Access

Context Menu	Right-click on Package > Package Control > Check In Branch

Check in Packages from within a model branch

Step	Action
1	The 'Select Packages to Check-in' dialog lists all version controlled and checked-out Packages within the selected model branch. By default, the entire list is selected. Optionally: • Click an individual Package to select just that Package • Ctrl+click on an individual Package to add or remove it from the selection • Shift+click on a range of Packages to select them • Click on the All button to select all listed Packages • Click on the None button to clear the selection
2	Optionally, you can commit into version control a new revision of all selected Packages, while keeping those Packages checked out for further editing. To do this, select the 'Keep packages checked-out after committing new revision' checkbox.
3	Click on the OK button. The 'Add Comment' dialog displays. A default comment is provided that contains the current date and time. This comment is applied to all Packages that are checked in.
4	Optionally, modify the default check-in comment.
5	Click on the OK button. The selected Packages are exported and checked-in. The Package icons are updated to reflect any change in version control status. If you opted to keep Packages checked out, there is no change in status.

Update to the Latest Revision of Selected Package

When you are part of a team working in a Distributed Model environment, you will want to periodically update your model with the changes that other team members have committed into version control. You can transfer the other users' updates from version control into the selected Package in the Project Browser.

Access

Context Menu	Right-click on Package > Package Control > Get Latest

Update Package to latest revision

The local XMI file associated with the Package is updated to the latest revision from version control. The XMI file is imported into your model database, updating the Package in your model.

When working in a Private Model, if the system detects that the Package content in the model is already up to date with the latest revision of the Package file retrieved from version control, then the 'Import Package' dialog displays first. This dialog is not displayed for a Shared Model.

These options are available:

- 'Force Reload From XMI' - reload the Package from XMI regardless of whether it is up to date or not

- 'Accept current package' - select to skip the process of re-importing the Package from XMI

- 'Refresh model view' - select to refresh the Project Browser and diagrams, by reloading the Package content from the project database

- 'Always use these settings' - when selected, if you subsequently check out a Package that is found to be up to date, the same settings are applied again without displaying the dialog

Notes

- The 'Get Latest' command is disabled for any Package that is checked-out (to anybody) in the currently loaded project

- When using a Shared Model environment, where all users are connected to a single model database, you should reload the Package from the database, rather than using the 'Get Latest' command

- If you have selected the 'Always use these settings' checkbox and you want to reconfigure the 'Import Package' dialog, press the Ctrl key whilst you select the 'Package Control | Get Latest' context menu option; the dialog displays and you can change the settings

Update to the Latest Revision of All Packages

When you are part of a team working in a Distributed Model environment, you will want to periodically update your model with the changes that other team members have committed into version control. You can transfer the other users' updates to all version controlled Packages into the currently loaded project.

Access

Context Menu	Right-click on Package > Package Control > Get All Latest

Update all Packages in project to latest revision retrieved from version control

All the local XMI files for all the version control configurations used in the project are updated to the latest revision from version control. The system then scans the Packages in the model, to determine which ones are up to date and which are not, compared to the latest revisions of the associated Package files.

A prompt displays, providing these import options for Packages that are up to date:

- Import changed files only
- Always import
- Prompt for each file

Click on the OK button. The version controlled Packages in your project are updated according to the option you selected; if you chose the 'Prompt for each file' option, a prompt displays to confirm import of each file.

Notes

- There is no need to re-import Packages that are already up to date - re-importing Packages first deletes them from the project and then re-imports them from the XMI file, which is time consuming as well as unnecessary; we strongly recommend using the default option 'Import changed files only'

- The 'Get All Latest' command does not update any Package that is checked-out (to anybody) in the currently loaded project; otherwise, any changes not yet committed to version control would be discarded

- When using a Shared Model environment, where all users are connected to a single model database, the information in the model database is always the same as, or ahead of, what is committed into version control; in this situation, the Get All Latest command will simply refresh your view of the model database, by reloading diagrams or reloading Package content in the Project Browser

Include Other Users' Packages

Other users might be developing Packages in their own models that you could use in your model, or you might have other models containing Packages that you want to use in the current model. Unless you are sharing an SQL database or project file, those Packages are not automatically available to you. However, if the Packages have been placed into version control, you can import them into your model as children of one of your model's Packages.

Access

Context Menu	Right-click on Package > Package Control > Get Package

Import Packages from version control into current model

Step	Action
1	On the 'Get Shared File' dialog, click on the drop-down arrow of the 'Version Control Configuration' field and select the version control configuration associated with the Package to retrieve.
	The file list is populated with the names of files available through that configuration, for retrieval and import into your model.
2	Click on the Package file to import into your model and click on the OK button.
	The Package file is imported as a new child Package, under the parent Package you selected.

Notes

- You must have access to the Package files through the version control system and you must define a version control configuration through which to access those files

- The version control configuration must use the same unique ID that was originally used to add the Package to version control

- XMI Package files associated with Packages that are already part of your project, are NOT included in the list of files available for import

Export Controlled Model Branch

Applying version control to a model can result in many XMI files placed under version control. It might then be hard to locate and import the file corresponding to the root of a particular model branch. Using Model Branch Files (.eab files) overcomes this problem by making it easier to export and import Package hierarchies from one model to another.

You could export a newly created model branch from your own private copy of a model so that, for example:

- Another user can import that branch into their own private copy of the same model
- It can be imported for inclusion as a common branch in a number of different models

Access

Context Menu	Right-click on Package > Package Control > Export as Model Branch

Create a Model Branch File to represent a Package hierarchy stored in version control

Step	Action
1	On the 'Export as Model Branch' dialog, in the 'EAB Filename' field, type a name for your Model Branch File. Alternatively, click on the [...] button and browse for the file location. Note that the Package name is supplied as a default.
2	Click on the OK button. A branch file is created to represent the selected Package. The branch file is committed to version control using the same version control configuration that controls the Package you selected.

Notes

- You can specify any file name, including sub-folder names, as long as the file is contained in or below the working folder of your version control configuration
- The facility is only enabled for Packages that are already under version control

Import Controlled Model Branch

Applying version control to a model can result in many XMI files placed under version control. It could then be hard to locate and import the file corresponding to the root of a particular model branch, if you want to:

- Retrieve a model branch created by another user in a private copy of a model, to import it into your own private copy of the same model
- Retrieve a model branch that is common in many models, for inclusion in a new model

Model Branch Files overcome this problem by simplifying the retrieval of Package hierarchies for use in other models. You use Enterprise Architect's Model Branch Files, of which there are few, to retrieve information about the root Package file such as the name and type of the version control configuration for the selected Package, and the relative filename of the version controlled XMI file associated with the Package. The system then uses this information to import the branch into your model.

Prerequisites

Before you begin, you must have:

- An operational version control environment that can be accessed by Enterprise Architect, and
- All of the version controlled Package files and the model branch file associated with the model branch to import, in a valid and accessible working copy folder

Access

Context Menu	Right-click on target Package for import > Package Control > Import a Model Branch

Import a Model Branch

Step	Action
1	On the 'Import VC Model Branch' dialog, either: - Use the lower portion of the dialog to select a model branch file (this is the simpler option if the associated version control configuration has already been saved in the current model; continue to step 2) OR - Click on the Find a Model Branch (.EAB) file button (this option is useful when you have not yet defined the version control configuration that is associated with the model branch to be imported; see the *Manually Locating Model Branch Files* topic)
2	Click on the drop-down arrow in the 'Select a Version Control Configuration' field and select a configuration. A list of .eab files controlled by that configuration is displayed in the 'Select a Model Branch (.EAB) file' list.
3	Select the Model Branch File you need, then click on the OK button.

	The system imports the root Package specified in the Model Branch File and recursively imports and populates all the sub-Packages contained in the root Package.

Notes

- The Import a Model Branch command is only enabled for Packages that you (the current user) are able to edit, as the imported model branch is inserted into the model under your selected Package

Manually Locating Model Branch Files

When importing a Model Branch File from version control, you might not have the associated version control configuration saved in the model that is receiving the import. In this situation, it is simpler to manually browse the file system to locate the Model Branch File (.eab) and let Enterprise Architect derive the details of the configuration from the branch file you select.

Prerequisites

Before you begin, you must have:

- An operational version control environment that can be accessed by Enterprise Architect, and
- All of the version controlled Package files and the model branch file associated with the model branch to import, in a valid and accessible working copy folder

Access

Context Menu	Right-click on target Package for import > Package Control > Import a Model Branch : Find a Model Branch (.EAB) file

Locate the Model Branch File

Step	Action
1	Access the 'Import VC Model Branch' dialog, then browse for and select the Model Branch File that represents the model branch to import.
2	Click on the Open button. If the version control configuration referenced by the file is fully defined within the current model, the import commences at this point. Otherwise, Enterprise Architect displays a dialog prompting you to complete the required configuration.
3	Click 'Yes' to proceed with completing the definition of the version control configuration. The 'Version Control Settings' dialog is displayed.
4	Complete the definition of the configuration. (Typically this involves simply specifying the working copy folder.)
5	Click on the Save button. The configuration details are saved. The import of the model branch proceeds.

Notes

- The Import a Model Branch command is only enabled for Packages that you (the current user) are able to edit, as the imported model branch is inserted into the model under your selected Package

Review Package History

It is possible to review the change history of version controlled Packages by examining previous revisions. If necessary, you can check out one of these earlier revisions of a Package for editing, effectively rolling-back to that prior revision of the Package.

Access

Context Menu	Right-click on Package > Package Control > File History

Review change history of a version-controlled Package

Step	Action
1	For version control environments using Subversion, CVS or TFS command line clients, the 'File Version History' dialog displays. Click on a revision number in the 'Revisions' field, to select that revision and view its log entry. For version control environments using SCC based clients, your particular product opens its own 'File Version History' dialog.
2	On the 'File Version History' dialog, you can optionally click on either: • Check Out - the selected revision of the Package file is retrieved from version control and imported into your model as a Package that is checked out for editing; you can subsequently check in this revision as a new HEAD revision, effectively allowing you to revert the Package to a prior revision or • Retrieve - the selected revision of the Package file is retrieved from version control and imported into your model, but the Package remains flagged as checked-in and cannot be modified; subsequently checking out the Package updates it to the latest revision before it is unlocked for editing

Notes

• If the selected Package was already checked out in the current model, the Retrieve and Check Out buttons are disabled

• If the selected Package contains any sub-Package that is already checked-out in the current model, a warning will be displayed and the retrieval or check-out will not go ahead

• If you check out a prior revision of a Package, but do not want to commit it as a new revision, right-click on the Package and select Package Control | Undo Check Out

Review Package History - SCC Client

It is possible to review the change history of version controlled Packages by examining previous revisions. If necessary, you can check out one of these earlier revisions of a Package for editing, effectively rolling-back to that prior revision of the Package. The process for reviewing the change history of Packages configured for version control with an SCC client (including products such as Visual Source Safe, TFS-SCC, ClearCase, Perforce, AccuRev and MKS Source Integrity) differs from that for Subversion, CVS or TFS command line clients.

Access

Context Menu	Right-click on Package > Package Control > File History

Review change history of a version-controlled Package (SCC client)

Step	Action
1	The change history mechanism offered by the third party SCC provider displays. To import a prior revision of the Package into your model, use the 'SCC History' dialog to retrieve the revision, then close the dialog.
2	The SCC client notifies Enterprise Architect that a different revision has been retrieved. A prompt then displays, asking whether you want to check-out the prior revision.
3	Optionally, click on either: • Yes, to check out the prior revision - the selected revision of the Package file is retrieved from version control and imported into your model as a Package that is checked out for editing; you can subsequently check in this revision as a new HEAD revision, effectively reverting the Package to the prior revision OR • No, to import the prior revision as read-only - the selected revision of the Package file is retrieved from version control and imported into your model, but the Package remains flagged as checked-in and cannot be modified; subsequently checking out the Package updates it to the latest revision before it is unlocked for editing

Notes

- If the selected Package was already checked out in the current model, the system does not proceed with retrieving a prior revision

- If you check out a prior revision of the Package, but do not want to commit it as a new revision, right-click on the Package and select 'Package Control | Undo Check Out'

Retrieve Prior Revision - SCC Client

Depending on your version control product, retrieving a prior revision of a controlled Package can involve a number of prompts regarding overwriting the current local copy.

This example details retrieval of a prior revision from a TFS-SCC version control configuration.

Access

Context Menu	Right-click on Package > Package Control > File History

Example Procedure - retrieve prior revision, TFS-SCC client

Step	Action
1	Display the 'TFS File History' dialog.
2	Click on the Get button. The TFS-SCC client displays the 'Resolve Conflicts' dialog. This dialog offers the 'Automerge All XML Package files' option. DO NOT select this option. It is important to prevent any merging of Enterprise Architect's XML Package files.
3	Click on the Resolve button. The TFS-SCC client displays the 'Resolve writable file conflict' dialog.
4	Select the 'Overwrite local file/folder' option. The existing working copy of the Package file is overwritten by the prior revision retrieved from version control.
5	Click the OK button. The TFS-SCC client redisplays the 'Resolve writable file conflict' dialog; it should now show no conflicts.
6	Click on the Close button. The TFS-SCC client redisplays the 'File History' dialog.
7	Click on the Close button. Enterprise Architect displays a prompt, asking whether to check-out the prior revision.
8	Click on the: • Yes button to check-out the prior revision • No button to retrieve a read-only version of the Package, that is NOT checked-out and is NOT editable

Validate Package Configurations

Having defined the version control settings for your current model, you can test the validity of those settings associated with each version controlled Package within the model.

Access

Ribbon	Configure > Version Control > Check Configuration

Validate version control settings

Step	Action
1	The validation process scans the model database and verifies that the version control configuration associated with each version controlled Package is fully specified in the current model. It also queries the corresponding version control provider to find the status of the Package file associated with each version controlled Package.
	The results of the validation process are sent to the System Output window.
2	Open the 'Version Control Settings' dialog to complete the definition of any invalid or missing version control configurations.
3	Click on an error message in the System Output window to highlight the corresponding Package in the Project Browser.
4	Right-click a Package node and select 'Package Control	Configure Package' to open the 'Package Control Options' dialog.
	Correct any problems with the version control details for the Package.
	Correct any problems with the Package's associated XMI file.

Resynchronize the Status of Version Controlled Packages

It is possible to update the version control status of version controlled Packages within your project to re-synchronize with the status reported by your version control provider. This can be useful if you are creating copies of your project, where checking in a Package from one copy of the model leaves the Package in the second copy of the model with an out-of-date version control status.

For a given Package, the re-synchronization process queries the corresponding version control provider to find the status of the Package file associated with the version-controlled Package. If necessary, the process then updates the Package flags within the model database, to synchronize the Package status recorded in the model with the value reported by the version control provider.

Access

Ribbon	Configure > Version Control > Re-Synch Status (applies to all Packages in model)
Context Menu	Right-click on Package > Package Control > Re-synch Status With VC Provider (applies to single Package only)

Resynchronize version control status

Step	Action
1	The results of the re-synchronization process are sent to the System Output window.
2	Double-click on any result message to select, in the Project Browser, the corresponding Package.

Notes

- This process does not cause any Package data to be either exported from your model to the associated Package file, or imported from a Package file into your model's Package data

- If a Package has been checked-out and modified with Enterprise Architect, but your version control provider reports the Package file as checked-in, running this process marks the Package within Enterprise Architect as being checked-in, without exporting and committing the pending changes; subsequently checking-out the Package imports the latest revision of the Package file from version control, effectively discarding the uncommitted modifications from the model

- Similarly, if a Package file is checked-out to you in your local working copy folder, but not in the Enterprise Architect model, running this process marks the Package within the model as checked-out, but it does not import the associated Package file from the version control system; consequently, it is possible to check-in a Package from Enterprise Architect that is potentially out of date, compared to the latest revision of the Package file within the version control system

Tracking Changes

If you want to track changes to data across your project, you can use two separate but complementary facilities - Auditing and Baselines.

Facilities

Facility	Detail
Auditing of model changes	Auditing is a project-level feature, available in the Corporate, Business and Software Engineering, System Engineering and Ultimate editions, that enables you to record model changes in Enterprise Architect.
	By enabling this feature, model administrators can view a range of information regarding changes, such as:
	• Who changed an element
	• How many elements they changed
	• When they changed the data
	• What the previous values were, and
	• What type of elements they changed
Baselining and differencing to capture and roll back changes	The Enterprise Architect Corporate, Business and Software Engineering, System Engineering and Ultimate editions provide a facility to ' baseline ' or snapshot a model branch in XMI format at a particular point in time, and store it within the model in compressed format.
	More than one baseline can be stored against a single Enterprise Architect Package; using baselines, you can compare Packages at the current and earlier stages of development, using the Compare (Diff) utility.
	The Compare utility is available in the Professional, Corporate, Business and Software Engineering, System Engineering and Ultimate editions of Enterprise Architect; it enables you to compare the current model with:
	• A Baseline
	• An exported Enterprise Architect XMI file on disk
	• A version-controlled Enterprise Architect XMI file on disk

Auditing

Auditing is a project-level feature that model administrators can use to record model changes in Enterprise Architect. After switching Auditing on, you can view information on changes such as:

- Who changed an element
- How many elements they changed
- When they changed the data
- What the previous values were, and
- What type of elements they changed

Auditing does not record changes to:

- Document Templates
- Model Documents
- Baselines or
- Profiles

We provide a Quickstart procedure to help you enable Auditing and see it in action, from which point you can explore a number of set-up options and display features.

Features

Feature	Detail
The Audit View	To view what has been recorded by the audit, use the Audit View, which provides an interface to everything recorded by Auditing.
	If security is enabled, you must have Audit View permission to display data in the Audit View.
Model Views	You can also obtain a snapshot of selected items in the model, using the Model View facility.
	In the Corporate, Business and Software Engineering, Systems Engineering or Ultimate editions of Enterprise Architect, you can use this facility to automatically generate a snapshot at intervals and, if there are changes in the items collected by the defined search, to trigger a notification to you of such changes.
	This helps you to monitor workflow and other events of concern to you.
Document Report	You can generate a document report that includes the audit history information for the selected element or Package, by choosing the basic + audit template.
Audit History	Using Auditing, you can track changes to an element or connector over time.
	However, enabling Auditing also enables an 'Audit History' tab in the System Output window, which summarizes all changes made to the selected element or connector.
Performance Issues	By enabling Auditing on a project, you increase the time taken for most actions.
	For most modeling tasks, this increase is insignificant; however, there are some instances where the difference is more substantial.

Notes

- The Auditing facility is available in the Corporate, Business and Software Engineering, System Engineering and Ultimate editions

- Warning - If your site runs separate editions of Enterprise Architect, when Auditing is turned on in a project any Desktop or Professional edition users are locked out of the project; to restore access, turn Auditing off in the project from the Corporate, Business and Software Engineering, Systems Engineering or Ultimate edition instance of Enterprise Architect

Auditing Quickstart

It is very simple to enable Auditing and see it in action.

Access

Ribbon	Configure > Model > Audit : Audit Settings

Enable Auditing

Step	Action
1	In the 'Audit View', click on the Audit Settings button. The 'Audit Settings' dialog displays.
2	Select the 'Enable Auditing' checkbox.
3	Click on the OK button to close the 'Audit Settings' dialog.
4	Close the 'Audit View' dialog.
5	Change and save your project. • Add a new Package • Add a new Class • Add a new connector • Change the name of an element • Delete an element
6	Open the Audit View again and click on the Refresh (or Load) button to display a record of the changes you have made.

Auditing Settings

Using the 'Audit Settings' dialog, you can change what is recorded by the Auditing facility and:

- Define the areas of processing in Enterprise Architect to audit
- Administer your audit records
- Indicate the kind of model objects for which changes are to be recorded
- Configure Auditing to record changes to only certain types of elements

Access

Ribbon	Configure > Model > Audit : Audit Settings

Configure Settings

Option	Action
Enable Auditing	Select this checkbox to turn the Auditing facility on.
Audit XMI Import	Select this checkbox to record changes arising from XMI importing in the audit. As version control uses XMI, you must select this option to record changes from checking out Packages.
Audit XMI Export	Select this checkbox to record changes arising from XMI exporting in the audit. As version control uses XMI, you must select this option to record changes from checking out Packages.
Audit Reverse Engineering	Select this checkbox to record changes arising from reverse engineering in the audit.
Use Database Timestamp	Select this checkbox to use the database server's timestamp instead of each user's local timestamp; this improves security. This option is not available for project files.
Clear Logs	Click on this button to permanently delete all log data from the current project; use the Save Logs button first, to save the audit records outside the project. The system prompts you to specify whether to clear items logged over a specific period of time. • Click on the No button to clear all log items currently held in the database • Click on the Yes button to display the 'Time Filter' dialog, on which you select a standard time period or define your own This function can be accessed through the Automation Interface.
Save Logs	Click on this button to save a copy of the logged audit items currently held in the project. These items remain in the project; you can use the Clear Logs button to

	remove them.
	The system prompts you to specify whether to save items logged over a specific period of time.
	• Click on the No button to save all log items currently held in the database
	• Click on the Yes button to display the 'Time Filter' dialog, on which you select a standard time period or define your own
	This function can be accessed through the Automation Interface.
Load Logs	Click on this button to load a previously saved set of logs back into the project. A browser displays through which you select the log file to reload.
	If the same record exists in both project and log file, that record is not reloaded.
Core	Select this radio button to record changes to elements (including attributes and operations), Packages, connectors and some model-level information.
Standard	Select this radio button to record the same changes as for the Core option, plus changes to some of the 'housekeeping' data for diagrams - Name, Author, Version, Stereotype, Notes and Date Modified.
	You can check for changes to diagram content and structure using the Baseline facility for reviewing visual changes to diagrams.
Extended	Select this radio button to record the same changes as for the 'Standard' option, plus changes to project security.
Debug	Select this radio button to record the same changes as for the 'Extended' option plus changes to any database tables. Some of the auditing information will not be visible unless the Audit View mode is set to 'Raw'.
Maintenance	Select this radio button to audit changes to maintenance elements only; that is:
	• Package (element)
	• Requirement
	• Feature
	• Use Case
	• Actor
	• Note
	• Issue and
	• Change
Core Structural	Select this radio button to audit changes to maintenance elements plus certain structural elements; that is:
	• Package (structure)
	• Class
	• Interface
	• Signal
	• Node
	• Component
	• Artifact
	• Part
	• Port and

	• Device
All	Select this radio button to audit changes to all types of element.
Custom	Select this radio button to audit changes to element types that you specify.
	The Customize button is made available; click on this button to display a list of element types, and select the checkbox against each element type to include in the audit (or click on the Select All button to select every element type).
	Click on the OK button to save the selection.

Notes

- As the number of records increases, the performance of the Audit View reduces; it is recommended that audit records that are not regularly required are saved to file, then cleared from the project, to help ensure high performance

- Connectors are audited when they are connected to an element that is included in the Audit Options

- If security is enabled, you must have Audit View permission to turn Auditing on, and Audit Settings permission to change the audit settings

The Audit View
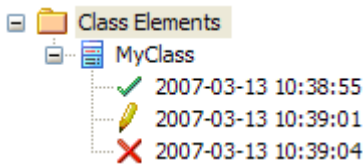
The Audit View provides an interface to the information that has been recorded by auditing.

Access

Ribbon	Configure > Model > Audit

Sections of the Audit View

The Audit View is divided into three main areas:

Section	Description
View controls	The view controls provide a variety of settings for controlling auditing and the display of audit records.
Audit tree	The audit tree displays the log items that have been recorded by auditing. What is displayed in the tree is affected by the view controls, such as: • Sorting • Filter (by time) • Mode • Auditing settings (what was actually recorded) If synchronizing with the Project Browser, it is also affected by the Package, diagram or element you have selected. In the audit tree: • The green tick indicates a creation • The yellow pencil indicates an edit • The red cross indicates a deletion Right-clicking an element in the audit tree (such as MyClass) displays a context menu that enables you to locate the selected element in: • The Project Browser • Any diagrams in which it exists
Record display	The record display is in two parts: the identity of the selected change, and the actual change made. The data in the record display is determined by the view controls and mode and, if synchronizing with the Project Browser, by the Package, diagram or element you

have selected.

Identity

The identity of a change consists of:

- The Windows username of the user that made the change; if security is enabled, the name is of the format WindowsUsername(SecurityUser)

- When the change was made

- The path of the change; for example: Class Class1 - Attribute Att1 - Attribute Constraint Constraint

Changes

Changes are displayed in a table format, showing the:

- Property (or data item) name

- Its original value before the change and

- Its value after the change

If you double-click on an item in the list of changes (or right-click and select the 'Show Differences' option) the Difference window displays:

This shows the specific changes that have been made, highlighting the edited, created, deleted or formatted characters.

Changes to the format of text in the element, made through the element 'Properties' dialog, are not apparent in the initial table; they are visible in the Difference window, identified by HTML formatting tags.
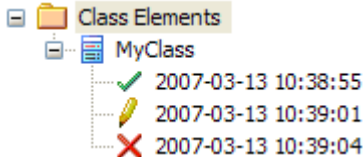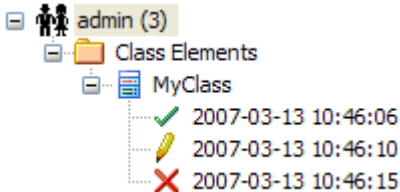
Notes

- If security is enabled, you must have Audit View permission to display data in the Audit View

Audit View Controls

The Audit View controls provide a variety of settings for controlling auditing and the display of audit records.

Options

Option	Action
Load or Refresh	Click to reload the Audit Tree, updated with any new audit results.
Search	Click to search through log items for a particular area - you can search by Name, Type or GUID. The items are loaded and filtered with the current Sort By, Time Filter and Mode settings. If you refresh the Audit View, you must run the search again.
Audit Settings	Click to open the 'Audit Settings' dialog.
Sort-by	Select the appropriate radio button for the required display setting: • Type - changes are grouped under element type (such as Class or Requirement), and then grouped under the changed element • User - changes are grouped under user name, each with the number of changes for that user Under each user name, changes are grouped as for the Type sort
Filter By Date/Time	Select to enable the Filter Settings button, to filter the audit results by time period. Changes that occur outside the selected filter period are not shown in the Audit View.
Filter Settings	Click to display the 'Time Filter' dialog, to set the filter time period: • Today - to display changes occurring today • Previous Hour - to display changes occurring in the last 60 minutes • Previous 24 Hours - to display changes occurring in the last 24 hours • Previous Week - to display changes occurring in the last 7 days • Previous 30 Days - to display changes occurring in the last 30 days • Previous Year - to display changes occurring in the last 365 days

	• Custom - to define your own time period, using the From and To fields The six pre-configured time periods automatically update when you click on the Refresh button; custom periods are static and do not automatically update. If you have set a filter period, and you deselect the 'Filter By Date/Time' checkbox, the period remains set; the custom time period, too, is retained so that you can re-use it or modify it later if required.
Status Text	Read to see which: • Mode has been selected and • Time filter is being applied to the data

Mode	Click to display a menu of options to change the mode of the Audit View.
	Select:
	• 'Standard' - to automatically synchronize with the Project Browser; where changes have been made, the Audit View reflects your selection from the Project Browser - if you click on:
	   - An element, the Audit View displays the history for that element
	   - A Package, the Audit View displays the history for that Package and its immediate children (but not the contents of nested Packages)
	   - A diagram, the Audit View displays the history for that diagram and its contents (which could be drawn from a wide area of the Project Browser)
	• 'Advanced' - to load large sets of log items independent of the Project Browser; in this mode, a special Audit Settings group can be displayed in the Audit Tree, which records:
	   - When Auditing has been enabled and disabled
	   - Who made the change
	   - The date and time of the change
	   - Changes to the Audit Settings
	   - When Audit Operations are executed
	   - Security changes (which can be browsed in the same way as other changes)
	• 'Deleted' - to display only deleted records, but otherwise data is shown as in 'Advanced' mode; records can be sorted by element type or by user as required
	• 'Raw' - to display all audit records in chronological order without sorting, enabling you to see a progression of changes; this can be especially useful in determining date-time inconsistencies
	Any search and filtering you define still apply, enabling you to view all of today's changes in order, or all changes for a particular element in order, or both
	Additional database information is displayed; this additional information might be insignificant or only in machine-readable format

Audit History Tab

When Auditing is turned on, an 'Audit History' tab is enabled in the System Output window. The tab shows a history of changes to whichever element or connector you have selected in the:

- Current diagram
- Project Browser
- Audit View, or
- Package Browser

As you select different elements or connectors, the 'Audit History' tab automatically updates to reflect your current selection. For each change made to the element or connector, the tab shows:

- Who made the change
- When the change was made
- Where the change was made
- The value of the characteristic before the change
- The value of the characteristic after the change

Access

Ribbon	Show > Window > System Output > Audit History

Notes

- To see this tab, you must have the Audit View open; the tab continues to display if you subsequently close the Audit View
- If security is enabled, you must have Audit View permission to display data on the 'Audit History' tab

Auditing Performance Issues

Enabling auditing on a project increases the time taken for most actions.

For most modeling tasks, this increase is insignificant; however, there are some situations where the difference is more substantial.

Operation Delays

Operation	Detail
Large Deletions	Deleting large Packages or Package structures, or large numbers of elements, takes significantly longer with auditing on. You might disable auditing before performing such a deletion.
XMI Imports	Importing XMI takes longer with auditing enabled. A project-level option is provided for disabling auditing of XMI Imports.
Reverse Engineering	Reverse engineering code takes longer with auditing enabled. A project-level option is provided for disabling auditing of reverse engineering.
Replication	You cannot remove replication from a model with Auditing enabled. If you have to remove replication, disable Auditing and - if prompted to do so - allow Enterprise Architect to roll back the database version; you can then remove replication.

Audit View Performance Issues

The Audit View can cause slow performance, generally because of the volume of records it has to process.

Considerations and Responses

Consideration	Detail
Navigating the Project Browser Within Auditing is Slow	Try setting the time filter to a period in the immediate past, such as Today, Previous 24 Hours or Previous Week; this time period updates each time you open or refresh the Audit View.
	Save log items outside the project with the Save Logs button; if you then clear the logs you have just saved, the load time of the Audit View is reduced.
	You can reload logs into the project at any time, using the Load Logs button.
The Audit View is Slow in Loading and Changing Modes	Try setting the time filter to a period in the immediate past, such as Today, Previous 24 Hours or Previous Week; this time period updates each time you open or refresh the Audit View.
	Save log items outside the project with the Save Logs button; if you then clear the logs you have just saved, the load time of the Audit View is reduced.
	You can reload logs into the project at any time, using the Load Logs button.
Navigating the Audit Tree is Slow	Close the 'Audit History' tab in the System Output window

Package Baselines

Enterprise Architect includes tools to help you manage and review changes to your models over time. These tools apply the concepts of Baselines, Differencing and Merges.

You use Baselines, Differencing and Merges essentially to compare two snapshots of a specific part of your project, to capture the differences between them and either roll back or incorporate selected changes or all changes.

Baselines

Enterprise Architect provides a facility to create a Baseline or 'snapshot' of the contents of a selected Package and its child Packages at a particular point in time; this enables you to later compare that branch of the model at that time with the current state of the branch.

Baselines are stored in the same XML format as is used for version control, but are stored within the project in compressed format.

You can also have parallel copies of parts of your model for team development, and create Baselines within each copy to merge changes into the project master.

Differencing

Differencing (Diff, or Compare) helps you to explore the differences between:

- The current state of a specific part of your project, and
- Previous or parallel versions captured in a Baseline or an XMI 1.1 file on disk

Merges

Once Differencing is complete, you can merge information from the Baseline into the current project; it is not possible to go the other way.

You can:

- Merge information manually, change by change
- Merge information automatically by electing to merge in all changes in one batch procedure
- Revert completely to the original Baseline by importing the stored XMI directly
- Merge information and elements from a Baseline in a different project, making it possible to keep multiple versions of a single model in synch

The merge options are available through the toolbar, context menus and the keyboard on the 'Compare Utility' tab, which shows the results of a comparison.

Visual differences in diagrams

Changes to a model might include:

- Adding or removing elements and connectors on a diagram, or
- Changing the position of elements or the overall layout of a diagram

You might believe that a diagram has changed, and select to compare it with a baseline using a context menu option from the Project Browser. Alternatively, you might perform a baseline comparison on a Package or a model and select from the comparison output any diagrams that are flagged as changed.

Notes

- Package Baseline facilities are available in the Corporate, Business and Software Engineering, Systems Engineering and Ultimate editions of Enterprise Architect

- The Enterprise Architect Corporate, Business and Software Engineering, System Engineering and Ultimate editions provide another facility, Auditing, which you can switch on to perform continuous monitoring of changes across the project; you can dovetail your use of each facility to meet the range of your change management requirements

- If a Package under version control forms part of a Baseline, and that Package is checked in to the model, you cannot merge the original data from the Baseline into that Package

- You can also obtain a snapshot of selected items in the model, using the Model Views facility; this facility enables you to automatically generate the snapshot at intervals and, if there are changes in the items collected by the defined search, to trigger a notification to you of such changes, which enables you to monitor workflow and other events of concern to you

- If security is enabled you must have 'Baselines - Manage' permission to create, import and delete Baselines, and 'Baselines - Restore' permission to merge data from a Baseline; security permissions are not required to select an existing Baseline and perform a comparison with the model

Baselines

Enterprise Architect provides a facility to 'Baseline' (snapshot) a model branch at a particular point in time for later comparison with the current Package state.

Baselines

Baseline comparison is most useful for determining the changes made to the model during development compared to some Baseline saved at a crucial point - for example the completion of a phase or version iteration.

More than one Baseline can be stored against a single Enterprise Architect Package.

Baselines are particularly useful during requirements management to check for changes, additions and deletions that have occurred since the start of the current work phase; knowing how a model has changed is an important part of managing change and the overall development process.

Baselines are stored within the model in compressed XML format; you can save a Baseline to an external XML file for storage or archive, or for distributing to other users working on models derived from a master project.

Baselines are generally used in conjunction with the Compare utility.

Scenario

A typical scenario for using Baselines would be:

- Create the base model branch to a sufficient point to create a Baseline (checkpoint); create and store the Baseline as Version 0.1a

- As work continues on development, managers and developers can check the current model branch against the Baseline for important modifications, additions and deletions; the Compare utility can be invoked from the Baseline dialog to check the current model branch against the stored version

- As required, minor Baselines can be created to check recent progress; these 'temporary Baselines' are useful for managing change when a lot of work is being done and it is important to only see what has changed in, for example, the last 24 hours

At sign-off or the move to a new version/phase, a major Baseline can be created to capture the new state of the model.

Minor Baselines created earlier can be deleted if required to save space

Considerations

- Baselines are based on the GUID or unique ID of a particular Package:

- Enterprise Architect checks for that ID as the root element within the XML document being used as a Baseline

- When you export a Package to XML, the Package you export is the root element; likewise when you create a Baseline, the current Package is the root Package of the XML Baseline

- When you save information in a version control system, the current version-controlled Package is again the root Package of the document

- It is not useful to create a Baseline by importing an XMI Package file created by version controlling a Package that itself contains version-controlled child Packages; that type of XMI Package file contains stubs for the child Packages, not full information on the child Packages and elements

- If a Package under version control forms part of a Baseline, and that Package is checked in to the model, you cannot merge the original data from the Baseline into that Package

XML files must be in the same format used by the Baseline engine - currently the UML 1.3 XMI 1.1 format (plus Enterprise Architect extensions), which contains all the information necessary to reconstruct a UML model, even a UML

2.x model

Notes

- The Baseline facility is available in the Corporate, Business and Software Engineering, System Engineering and Ultimate editions of Enterprise Architect
- The Compare utility is available in the Professional edition of Enterprise Architect, as well as in the Corporate and extended editions

Manage Baselines

Enterprise Architect provides a range of facilities for working with and managing Baselines.

Access

Ribbon	Design > Tools > Baselines > Manage Baselines or
	Design > Package > Manage > Baselines
Context Menu	Right-click on Package > Package Control > Package Baselines
Keyboard Shortcuts	Ctrl+Alt+B

Baseline Management

Create, select and process Baselines using the 'Package Baselines' dialog.

Option	Action
Current Baselines For Package: <Name>	Review the Baselines for the current model branch, listed by version reference with the highest alphabetical/numerical value at the top.
	If an entry is longer than the display area, a horizontal scroll bar displays at the bottom of the panel; use this to scroll to the text that is not shown.
Show Differences	Run the Compare utility on the selected Baseline and the current model branch or diagram, to display the differences between the two.
Restore to Baseline	Completely restore the model branch from the selected Baseline.
New Baseline	Create a new Baseline.
Delete Selected	Delete the selected Baseline.
Load Other Baselines	Display a drop-down menu that enables you to load Baselines from another model, in either a project file or a DBMS repository.
	• For project files, a browser displays; locate the required project file
	• For DBMS repositories, the Windows 'Data Link Properties' dialog displays; select the data provider and click on the OK button to display the 'Select Data Source' dialog, from which you select the required project
	In either case, the *Connected To:* message at the bottom of the 'Package Baselines' dialog changes to the name of the alternative model.
	To return the dialog to the original project, select the third option on the drop down list: 'Load From Selected Package'.
Import File	Import an XML 1.1 file from the file system as a new Baseline for this current model branch.

Export File	Export the selected Baseline to an XML file on disk.
Compare Model to File	Compare the selected model branch with an XML 1.1 file on disk; a browser displays, which you use to locate the file.
Options	Set filters to make the comparison more specific.

Notes

- Package Baseline facilities are available in the Corporate, Business and Software Engineering, Systems Engineering and Ultimate editions of Enterprise Architect

Create Baselines

This topic details the basics of creating new baselines of Model Packages.

Access

Select a Package in the Project Browser, then open the 'Baselines' dialog using one of the methods outlined here.

In the 'Baselines' dialog, click on the New Baseline button.

Ribbon	Design > Tools > Baselines > Manage Baselines : New Baseline or Design > Package > Manage > Baselines : New Baseline
Context Menu	Right-click on Package > Package Control > Package Baselines : New Baseline
Keyboard Shortcuts	Ctrl+Alt+B : New Baseline

Create a new baseline

Field	Action
Name	Display the Package name of the currently selected model branch.
Version	Type a unique version reference for this Baseline, which can consist of any alphanumeric characters. The 'Package Baselines' dialog sorts the Baselines according to the value of this field.
Include Sub-packages	Include the entire sub-Package hierarchy of this branch in the Baseline; this option defaults to selected. If you deselect the checkbox, only the immediate contents (XMI stubs) of the Package are included in the Baseline.
Note	Edit the default current time and date to any other value. The field is a single-line entry, for display on the 'Package Baselines' dialog (a one-line-per-entry list).
OK	Click to create a new Baseline and return to the 'Package Baselines' dialog.

Notes

- Package Baseline facilities are available in the Corporate, Business and Software Engineering, Systems Engineering and Ultimate editions of Enterprise Architect

The Compare Utility (Diff)

Enterprise Architect has a comprehensive and powerful built in Compare (diff) utility, which enables you to:

- Explore what has changed within a model over time

- Explore how previous versions of a model branch differ from what is currently in the model

- Perform a full model comparison by exporting all of Model A to XMI, then using 'Compare Model to File' from within the current model (Model B)

Comparing and checking model development at various points in the process is an important aspect of managing change and development, monitoring what is being modified and ensuring the development and design process is on track.

Using the Compare Utility you can:

- Compare a model branch in Enterprise Architect with a Baseline created using the Baseline functionality (Corporate, Business and Software Engineering, System Engineering and Ultimate editions)

- Compare a model branch in Enterprise Architect with a Baseline stored in a different model

- Compare a model branch in Enterprise Architect with an XML 1.1 file on disk created previously using the Enterprise Architect XML export facility (user selects file)

- Compare a model branch in Enterprise Architect with the current version-controlled XMI 1.1 file on disk as created when using Version Control in Enterprise Architect (file automatically selected)

Access

Select a Package in the Project Browser, then open the 'Baselines' dialog using one of the methods outlined here.

In the 'Baselines' dialog, click on the Show Differences button.

Ribbon	Design > Tools > Baselines > Manage Baselines : Show Differences or
	Design > Package > Manage > Baselines : Show Differences
Context Menu	Right-click on Package > Package Control > Package Baselines : Show Differences or
	Right-click on Package > Package Control > Compare with XMI File (for Package not under version control) or
	Right-click on Package > Package Control > Compare with Controlled Version (for Package under version control)
Keyboard Shortcuts	Ctrl+Alt+B : Show Differences

Differencing With Baselines

As a Baseline is stored within a model and contains all the information, elements and connections for a Package at a point in time, it can be used within Enterprise Architect to track changes to model elements over time.

The Differencing engine first builds a representation of the current Package in memory, based on what is currently in the model.

It then compares this with the stored Baseline, highlighting changes, new elements, missing elements and elements that have been moved to other Packages.

It is possible to filter the resultant output to display only one particular kind of change: for example, additions to the model.

If a Baseline has been created to ignore child Package content, a comparison between that Baseline and the model does not include any child Package content in the model.

See the Example comparison.

Notes

- This utility is available in the Professional, Corporate, Business and Software Engineering, System Engineering and Ultimate editions of Enterprise Architect
- You cannot compare the current model with an XMI 2.1 file; the utility can only compare with an XMI 1.1 file

Compare Options

You use the 'Compare Options' dialog to refine the output of the Compare utility when it compares the current model with a Baseline.

To display the dialog, either:

- Click on the Options button on the 'Package Baselines' dialog, or

- Click on the 'Compare Options' icon on the 'Compare Utility' tab toolbar

If the 'Compare Utility' tab shows the results of a Baseline comparison, when you click on the OK button the display refreshes to refine the information according to the options you have selected.

Notes

- Package Baseline facilities are available in the Corporate, Business and Software Engineering, Systems Engineering and Ultimate editions of Enterprise Architect

Options

Option	Action
Always Expand to Differences	Always display the list of elements fully expanded to show changes. If you deselect the checkbox, when the 'Compare Utility' tab is first opened it lists the Package contents to element level, and you expand each element as required to show the changed items. For large branches of the model, it is better to leave the checkbox unselected.
Show Elements that are	List elements that: • Have been changed since the Baseline was created • Are in the Baseline only (that is, have been deleted from the model since the Baseline was created) • Are in the model only (that is, have been created since the Baseline was created) • Have not changed since the Baseline was created (you might generally leave this checkbox unselected)
Suppress these Changes	Exclude: • Changes to diagrams • Changes to the 'Date Modified' field for an item • Changes to the 'Date Created' field for an item • Child items of a deleted item • Changes to advanced properties (defaults to selected)
Baseline Diagram Compare Options	Select the checkbox to always open the first parent Package for which there is a Baseline, when you select the diagram for comparison from the Project Browser.

Check Visual Changes to Diagrams

The Baseline Diagram Compare feature is a quick and easy way to visually compare a current diagram with an earlier version from a saved Baseline, and highlight any elements in the diagram that have been added, deleted, resized or moved.

You can then review these changes and optionally roll back each change if needed to its previous state from the Baseline.

The changes are identified on the 'Baseline Diagram Compare' dialog and on the diagram itself. If the diagram is not already open, the compare feature also opens the diagram.

Access

Ribbon	Layout > Diagram > Manage > Compare to Baseline
Context Menu	In Project Browser - Right-click on Package > Package Control > Package Baselines > <select baseline> : Show Differences > Locate and right-click on diagram name > Compare to Baseline
	In Project Browser - Right-click on Diagram > Compare Diagram to Baseline
	In open Diagram - Right-click on diagram background > Advanced > Compare Diagram to Baseline

Processing

In two of these access paths, you perform a comparison of a Package and Baseline, and then select the diagram from the results on the 'Baseline Comparison' dialog to display the 'Baseline Diagram Compare' dialog. Refer to the *Results* section and the *Options* table.

In the other access paths, you first select the diagram to check, and then might have the option of selecting the Package from which to use a Baseline, on the 'Load Baselines from Package' dialog.
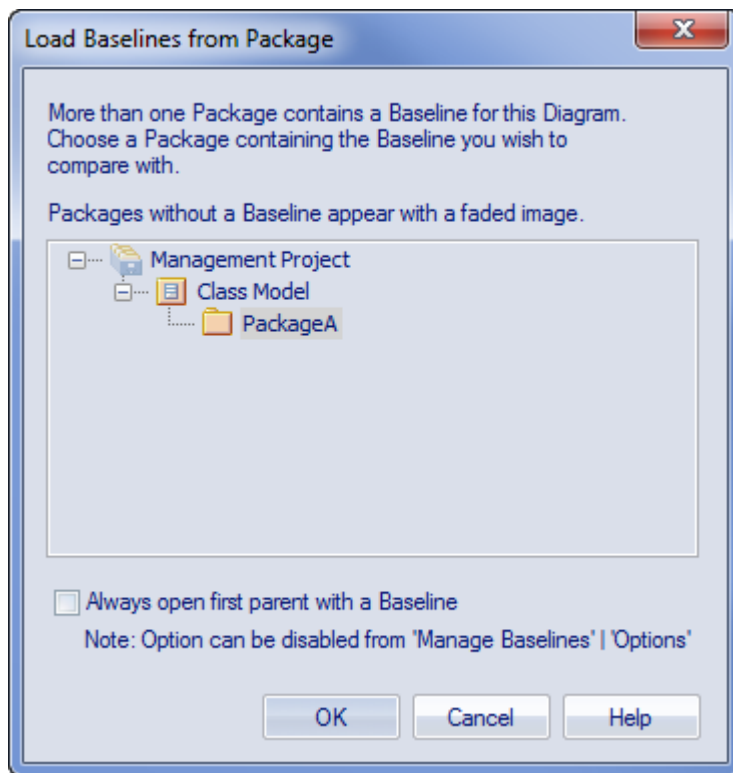
This dialog displays if you have NOT selected the:

- 'Always open first parent with a Baseline' checkbox on the dialog itself or
- 'Baseline Diagram Compare Options' checkbox on the 'Compare Options' dialog

(Selecting or clearing one of these checkboxes resets the other one as well.)

When you create a Baseline, it can be for a Package that contains one or more levels of child Package, and you might create Baselines for the Package(s) at each level. If the diagram you are checking is at a lower level in the hierarchy, there might therefore be a number of Baselines that contain information on the diagram, perhaps taken at different times and capturing different changes to the diagram. The 'Load Baselines from Package' dialog provides the facility to compare the diagram with one of a broader range of Baselines than just those from the diagram's immediate parent.

Click on the Package, and click on the OK button. In this case, **or** if the dialog did not display at all (the checkboxes were selected), the 'Baselines dialog displays.

Click on the required Baseline and on the Show Differences button. The 'Baseline Diagram Compare' dialog displays. Refer to the *Results* section and the *Options* table.

Results
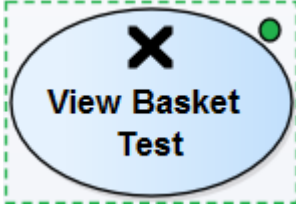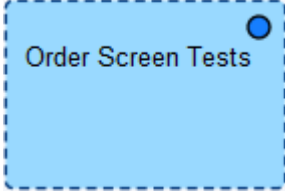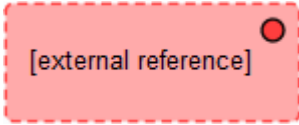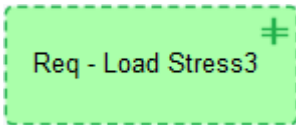
The 'Baseline Diagram Compare' dialog shows the elements that have been changed on the diagram, and what kind of change was made ('Status' field).
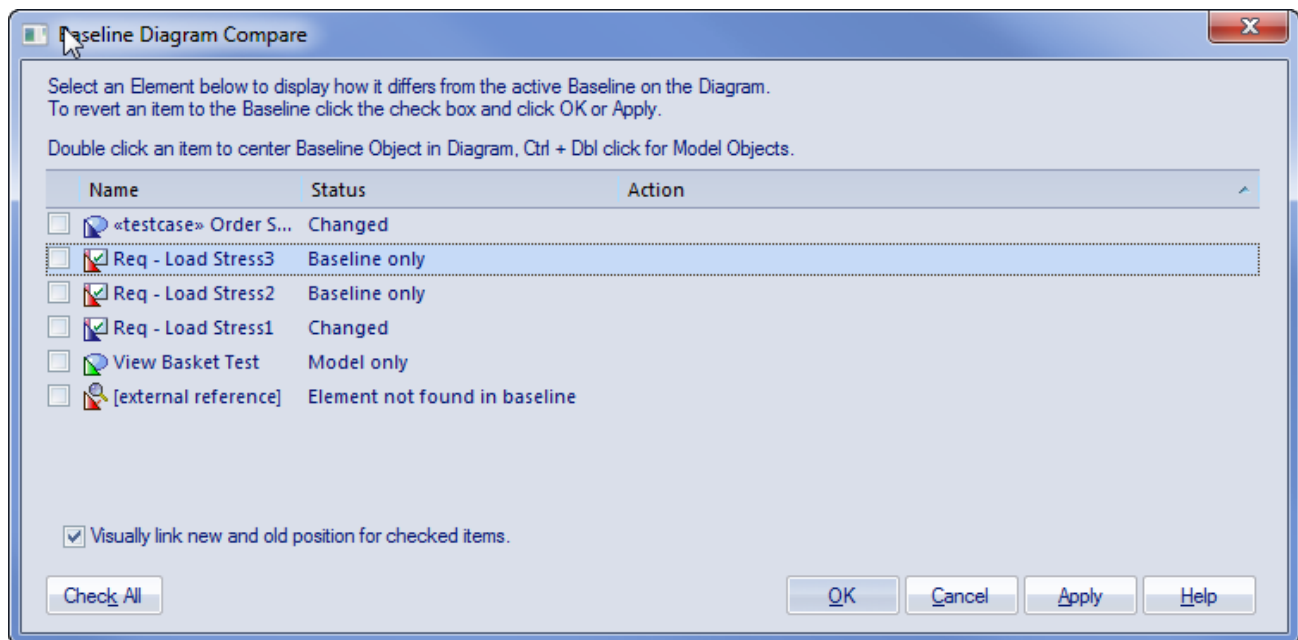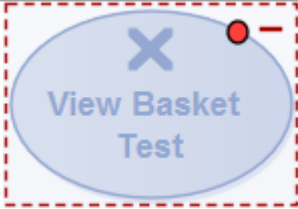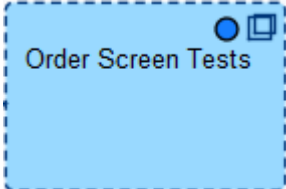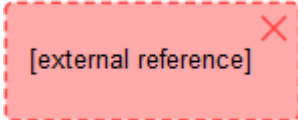
As you select elements on the dialog, images are shown on the diagram itself to indicate where the changed element was and what kind of change it underwent.

Options

Option	Detail
Select (click on) name of element	The 'Status' column indicates whether the element has been: • Moved or re-sized (Changed) • Deleted from the diagram (Baseline only) • Added to the diagram since the Baseline was captured (Model only), or • Deleted from its parent external Package, and there is no record in the current Baseline (because the Baseline is only for the current diagram's parent Package) This diagram-linked element has been deleted from the model. The element might be found in a different baseline either in a parent Package baseline or a different Package baseline outside of the current Package. If the external referenced element is restored to the model, the visual comparison will be able to resolve the missing diagram object in the current baseline. When an item is selected, the corresponding element on the diagram is highlighted as shown: • Deleted from the diagram • Added to the diagram

	   • Resized or moved to a new position • A deleted external element on the diagram The highlighted element on the diagram is marked with a colored dot, as shown, to indicate that it is in focus.
Position the diagram to show the selected element	To scroll the diagram so that you can see the original (Baseline) position of an element, double-click on the item in the list. To scroll the diagram so that you can see the current (model) position of the element, press and hold Ctrl while you double-click on the item.
Leave the changes in the item as they are	Ensure that the checkbox against the item is not selected. Click on the OK button.
Roll the changes back to the Baseline position	Click on the checkbox against each required item (or click on the Check All button to select every item). The 'Action' column displays the action required to roll each element's relationship to the diagram back to the Baseline relationship, and on the diagram the selected elements are represented as shown: • This deleted element will be restored • This added element will be removed

- This resized/repositioned element will be put back in its original position

- This element from another Package, deleted from the diagram, cannot be restored from this Baseline

The comparison automatically shows a blue direction arrow for each reposition or resize that has been checked. For a heavily edited diagram this might be confusing. However, you can hide the arrow for all elements except the one currently in focus; to do this:

- Deselect the 'Visually link new and old position for checked items' checkbox

To roll back the changes for all items for which a checkbox is selected:

- Click on the Apply button
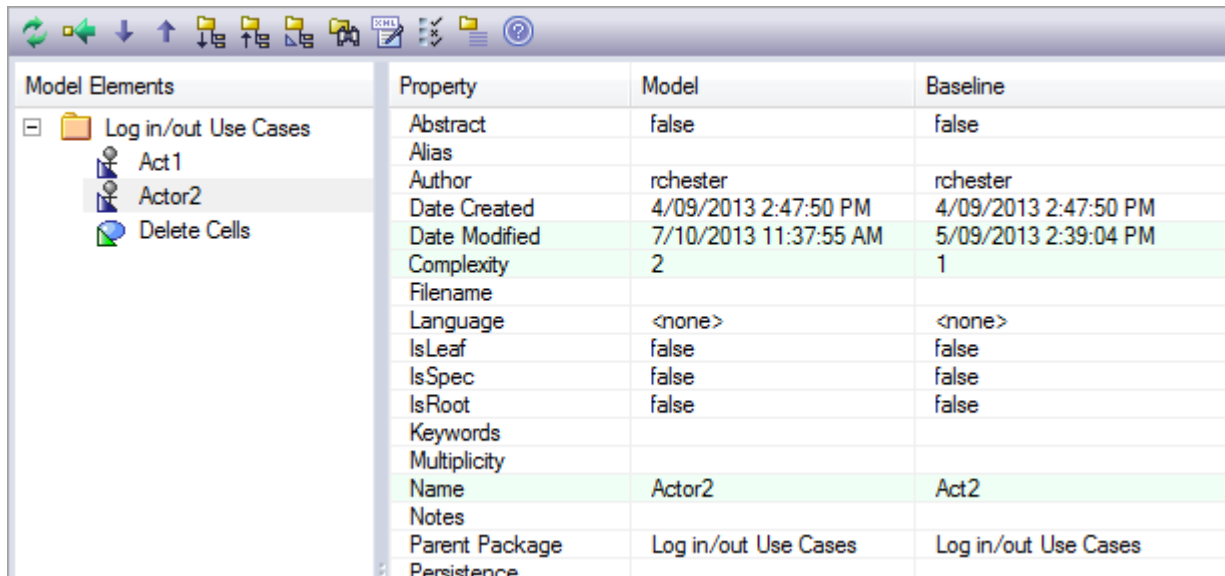
Notes

- Diagram Baseline facilities are available in the Corporate, Business and Software Engineering, Systems Engineering and Ultimate editions of Enterprise Architect

Example Comparison

This diagram shows the result of a comparison between a Package (Log in/out Use Cases) in the current project and that Package in a Baseline captured at an earlier date.

The results of the comparison are displayed on the 'Baseline Comparison' tab.

Review Changes

Aspect	Description
Interpretation	A hierarchy of model elements is displayed in the left-hand pane.
	From the triangle-based icons and the highlighted lines on the report, it is clearly visible which items in the hierarchy have, since the Baseline was captured, been:
	• Changed
	• Deleted from the model (in the Baseline only)
	• Added to the model (in the Model only) or
	• Switched to different Packages (changes in the Parent Package property)
	If you click on an item in the left hand pane, the right-hand pane displays a table of properties showing the values of those properties in the current model and in the Baseline.
	For each property where there is a difference between the model and the Baseline, the row is highlighted.
	The 'Compare Utility' tab enables you to perform operations (such as merging or rolling back changes) on the reported information, using the toolbar, context menu and keyboard.
Increase Level of Detail	The right panel of the 'Compare Utility' tab might, for some fields, display only part of the value.
	It might also not be immediately obvious what a change is.
	In either case, you can double-click on the property to display full details and to highlight the exact differences; this example shows the highlighted changes to the 'Name' property.

	Model Baseline
	Actor2 Act2
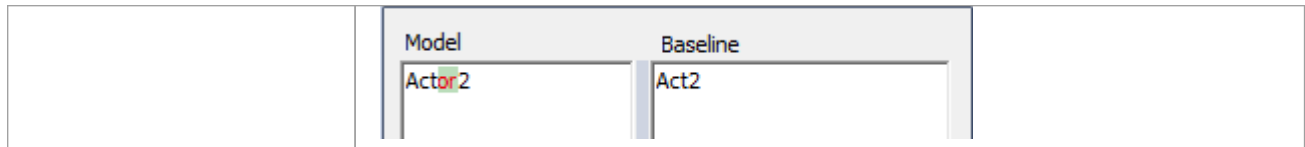
Notes

- The Compare utility is available in the Professional, Corporate, Business and Software Engineering, Systems Engineering and Ultimate editions of Enterprise Architect

Baseline Comparison Tab Options

The 'Baseline Comparison' tab enables you to perform operations on the reported information, using the toolbar, context menu, 'Merge' dialog and certain keyboard keys.

- The toolbar is at the top of the left-hand panel; the icons operate either on the comparison as a whole or on the currently-selected item in the left hand panel of the 'Baseline Comparison' tab
- Each item in the hierarchy has a context menu, which you display by right-clicking on the item; the options displayed depend on the level of the item in the hierarchy
- The 'Merge' dialog enables you to specify which changes to roll back in the model from the baseline
- You can use a selection of keyboard keys to move up and down the hierarchy, or to roll back changes

Toolbar Options

Option	Action
Refresh	Re-run the comparison to refresh the current display.
Merge To Model	Merge the values of the currently-selected item in the Baseline back into the model.
Next Change	Highlight the next changed item (this skips Moved items).
Previous Change	Highlight the previously-changed item.
Expand All	Fully expand the selected item.
Collapse All	Collapse the changed items in the selected item.
Expand To Changed Items	Expand the selected item to show changed items only (in the event that you have selected to also show unchanged items in the comparison).
Find in Project Browser	Highlight the item in the Project Browser.
Log To XML	Log the changes to an XML file. A browser displays, on which you specify the file name and location.
Compare Options	Display the 'Compare Options' dialog.
Manage Package Baselines	Display the 'Package Baselines' dialog.
Help	Display the Help topic Package Baselines.

Context Menu Options

Option	Action
Merge from Baseline Add from Baseline	Restore the item in the model to the Baseline state, or restore a deleted item from the Baseline.
Delete from Model	Remove a recently-created item from the model.
Merge From Baseline (with Options)	(For the root node of the hierarchy on the 'Compare Utility' tab.) Display the 'Merge' dialog, which you can use to specify options for rolling back the whole model branch to the Baseline state.
Refresh	(Object-level items.) Re-run the comparison to refresh the current display.
Find in Project Browser	Locate and highlight the item in the Project Browser.
Open Baseline Diagram Compare	(For a diagram listed in the comparison.) Display the Baseline Diagram Compare window, showing differences in diagram content and layout.
Expand All	Fully expand the selected item.
Expand To Changed Items	Expand the selected item to show changed items only.
Collapse All	Collapse the changed items in the selected item.
Log To XML	Log the changes to an XML file. A browser displays, on which you specify the file name and location.
Compare Options	Display the 'Compare Options' dialog.

Merge Dialog Options

Option	Action
Changed	Restore all changed items in the model branch to the Baseline state.
In Baseline Only	Restore all deleted items to the model branch from the Baseline.
In Model Only	Remove all recently-created items from the model branch.
Moved	Restore all moved items to their original locations, as identified in the Baseline.
Full Restore from XMI	Completely restore the model branch to the version held in the Baseline XMI 1.1 file, (using the 'XMI Import' function). (This option automatically selects all the other options)

Keyboard Keys

- Ctrl+ ↓ - expand and highlight the next changed item
- Ctrl+ ↑ - expand and highlight the previous changed item
- Ctrl+ ← - undo the changes for a selected item (roll back to the Baseline values)

Compare Projects

A number of operations can make changes to your project that you either want to monitor carefully or not have at all. Such events include:

- Recovering from a database crash
- Restoring a backup
- Performing a Project Data Transfer
- Importing from XMI, and
- Deleting model elements

You might have made a copy of the original project or the purpose of the operation is to generate a copy, in which case you can compare the size and row counts of the 'before' and 'after' copies as a convenient 'sanity check'. The copies can be on different platforms; you have options to:

- Compare a project file to another project file
- Compare a project file to a DBMS-based repository
- Compare two DBMS repositories

The comparison examines the number of project rows in each database, producing a report indicating the total records in each and the difference in record count between the two. If discrepancies are found, you must investigate further manually. The comparison does not examine the actual data in the tables.

Access

Ribbon	Configure > Model > Check Integrity > Project Compare

Compare two projects

Step	Action
1	On the 'Project Compare' dialog, select the radio button for the database types of the two projects you want to compare: • File to File • DBMS to File • File to DBMS • DBMS to DBMS
2	In the 'Source Project' and 'Target Project' fields, type the name or connection string for the source and target projects to compare.
3	Click on the Compare Projects button. The results of the comparison display in the panel at the bottom of the dialog.
4	If you want to print the results of the comparison, click on the Print List button.

Project Maintenance
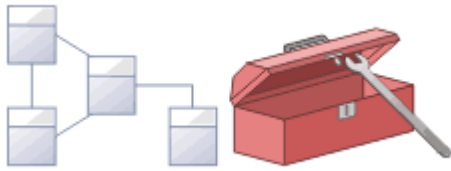
The repository is an important data store that must be maintained to ensure that it is running efficiently and that the information it contains has integrity. Enterprise Architect has built-in features that allow you to keep the repository in good health. These include tools to maintain the database itself, such as Repair and Compact functions (for file based repositories), and a series of tools to manage the data and models, such as model validation and data integrity checks. Reference data can be imported and exported from the repository which can be shared between repositories ensuring consistency. The Project Browser allows repositories to be refactored easily by dragging and dropping Packages, elements, features and diagrams to new locations. The Element Discussion and Team Review features allow the model to be critically analyzed and reviewed.

Facilities

Facility	Detail
Check the integrity of the data in a project	If you have a failed XMI import, network crash or other unforeseen event that could disrupt the integrity of information in the model, it is recommended that you run a Project Integrity Check.
Reset Auto Increment Columns in Tables	XMI Import/Export can affect the table auto increment column values and push them towards the maximum datatype value; you can re-sequence the columns to avoid this problem.
Upgrade an old project to enable use of new features	The structure of Enterprise Architect project files is occasionally changed to support more features; when this happens, existing project files must be upgraded to the new format to ensure correct operation and to take advantage of all the new features.
Rename a project	If you want to rename an Enterprise Architect project, you can do so through Windows Explorer.
Compact a project	After some time, a project .EAP file might benefit from compacting to conserve space.
Repair a project if it did not close properly	If a project has not been closed properly, such as during system or network outages, on rare occasions the .EAP file does not re-open correctly.

Notes

- You only rename, compact and repair models created as .eap files; these processes are not required for models stored in a DBMS

Check Project Data Integrity

If you have a failed XMI import, network crash or other unforeseen event that could disrupt the integrity of information in the model, it is recommended that you run the Project Integrity Check function to check that your project data is structurally complete.

You can select a variety of items to check. The integrity check examines all database records and ensures there are no 'orphaned' records or inaccurate or unset identifiers. This function does NOT check UML conformance, only the data relationships and repository structure.

You can run the integrity checker first in report mode to discover if anything should be corrected, and then run it again in 'recover/clean' mode.

When Enterprise Architect 'cleans' the model, it attempts to recover lost Packages and elements, and generates a new Package at the model root level called _recovered_. Check through any elements that are found and, if required, drag them into the model proper. If they are not required, delete them.

Access

Ribbon	Configure > Model > Check Integrity > Project Integrity

Check the data integrity of your project

Step	Action
1	Open the project.
2	Select the 'Project Integrity Check' menu option. The 'Project Integrity Check' dialog displays.
3	Select the checkbox for each check to run: • Package Structure • Object Structure • Object Features • All GUIDs • Cross References • Connectors • UML 2.0 Migration
4	Select either: • The 'Report Only' option to just view a report on the state of your model, or • The 'Recover/Clean' option to recover and clean your project; before selecting this option back up your project file first
5	To write a log of the integrity check, click on the Save Results button and select a log file.
6	Click on the Go button to run the check.

	If you want to display the resulting information in a more readable layout, you can resize the dialog and its columns.

UML 2.0 Migration

The UML 2.0 Migration check enables you to migrate the project from UML 1.3 semantics to UML 2.0 semantics. The migration process currently converts activities that are invocations of operations into called operation actions as per the UML 2.0 specification.

The UML 2.0 Migration option is an exclusive process that does not enable any of the other checks to be selected. When you click on the Go button to perform the migration, a prompt displays for you to confirm the operation.

Notes

- In the Corporate, Business and Software Engineering, System Engineering and Ultimate editions of Enterprise Architect, if security is enabled you must have Check Data Integrity permission to perform a data integrity check

Reset Table Auto Increment or Identity Columns

This topic explains possible impacts of XML Export/Import on table auto increment columns, and how to re-sequence the columns whose value approaches the maximum datatype value.

Impacts

Impact Due To	Impact
XML Export/Import	XML Export/Import can cause gaps in the numbering sequence of auto increment columns. Each XML Import deletes rows from several tables; the import then adds rows starting from the maximum value of the auto increment column. Repeated XML imports can result in the value of the auto increment approaching the maximum value of the database datatype; for example, SQL Server's int datatype has a maximum value of 2,147,483,647.
Replication	Large auto increment values can also arise where the project originated as an EAP replica or design master. The Jet engine assigns random values for auto increment columns with each XML Import into the project. These random values can approach the maximum range of the repository data type, which could present a problem when the EAP project is transferred to a repository.

Access

Ribbon	Configure > Model > Check Integrity > Reset IDs

Re-sequence auto increment columns

Step	Action
1	Open the project.
2	Select the 'Reset IDs' menu option. A dialog displays listing all non-empty tables that contain an auto increment or identity column. • The 'Rows' column shows the number of rows in the table • The 'Maximum ID' column shows the current maximum value of the auto increment column • The 'Action' column shows either 'No Action' or 'Reset', depending on how close the column value is to the datatype's maximum Tables requiring a reset are automatically selected in the list.

3	Click on the Go button to reset the auto increment column values.

Notes

- Connection to the project must be via a **direct** connection; IDs cannot be reset while the connection to the repository is via Cloud Services

- Project Auditing must be disabled before IDs can be reset

- MySQL repositories will require re-starting of the MySQL server so that the reset auto increment columns start from the reset value

- In the Corporate, Business and Software Engineering, System Engineering and Ultimate editions of Enterprise Architect, if security is enabled you must have Check Data Integrity permission to reset IDs

Upgrade a Project

The structure of Enterprise Architect project files is occasionally changed to support more features. In such cases, the project might have to be upgraded. Upgrading to the new file structure is a simple and quick process that brings your project to the current level to:

- Ensure correct operation and
- Support all the latest Enterprise Architect features

Initial Check

When you load a project that was created in an earlier release of Enterprise Architect (for example, an archived project) using a recent release of Enterprise Architect, the system determines whether the project should be upgraded and, if the upgrade is necessary, displays the Upgrade Wizard.

The Upgrade Project Wizard

The Upgrade Project Wizard takes you through the upgrade process and:

- Advises you of the necessity to upgrade
- Advises you to back up the current project; it is essential to back up before any changes are made
- Checks which upgrade path is required
- Guides you through the steps to perform the upgrade
- Opens the newly converted project

Notes

- If you are using replication in your models, and the upgrade wizard detects that the project you are opening is a replica and not a Design Master, a different upgrade path is required
- Once upgraded, the project cannot be opened with the version of Enterprise Architect in which it was created

Upgrade Replicas

Models that have replication features added might have to be upgraded differently from regular projects.

- If the model is a Design Master (the root model of all other replicas) then you can upgrade the model to suit the current version of Enterprise Architect; after upgrading a Design Master you should re-create the replicas, rather than synchronizing

- If the model is not a Design Master, you must first remove the replication features, then upgrade the project in the normal manner; the Upgrade Wizard guides you through the steps

Run SQL Patches

Occasionally, Sparx Systems might release a patch to correct a model fault.

The patch generally checks how many records are to be updated, and reports on what is to be done.

Access

Ribbon	Configure > Model > Check Integrity > Run Patch

Rename a Project

If you want to rename a project, you can only do so at the Windows file system level, using Windows Explorer.

Rename an Enterprise Architect project file

Step	Action
1	If you have the project open, close it.
2	Ensure no other users have the file open.
3	Open Windows Explorer and navigate to the project.
4	Rename the project file using Windows Explorer.
5	You should keep the file extension to preserve compatibility with the default project type, as installed in the registry at installation time.

Compact a Project

After a period of time, through general use, a project .eap file might occupy more disc space than necessary. You can move the project to a local drive and compact the file to recover the unused space.

Access

Ribbon	Configure > Model > Check Integrity > Manage .EAP File > Compact .EAP File

Compact a project

Step	Action
1	If you have the project open, shut it down.
2	Ensure no other users have the file open.
3	Select the 'Compact .EAP File' menu option.
4	Follow the on-screen instructions to complete the process.

Notes

- Always compact projects on a local drive, never on a network drive
- In the Corporate, Business and Software Engineering, System Engineering and Ultimate editions of Enterprise Architect, if security is enabled you must have Administer Database permission to compact a project

Repair a Project

If a project has not been closed properly, such as during system or network outages or on poor network connections, on rare occasions the .eap file does not re-open correctly. A message displays informing you the project is of an unrecognized database format or is not a database file. In such cases, you can move the project file to a local drive and repair it.

Access

Ribbon	Configure > Model > Check Integrity > Manage .EAP File > Repair .EAP File

Repair a project that was not closed properly

Step	Action
1	Ensure that all users are logged off the project you are attempting to repair.
2	Copy the project file to a local drive on your workstation.
3	In Enterprise Architect, select 'Start > Workspace > Preferences' to display the 'Preferences' dialog. On the 'General' page, deselect the 'Use Jet 4.0 - requires restart' checkbox.
4	Close and restart Enterprise Architect - you do not need to open any model, including the one you are repairing.
5	Select the 'Repair .EAP File' menu option, and follow the on-screen instructions.
6	Once you have repaired the project, it is recommended that you perform a data integrity check.

Notes

- Always repair projects on a local drive, never on a network drive
- The 'Compact a Project' option can also repair corrupt .eap files
- In the Corporate, Business and Software Engineering, System Engineering and Ultimate editions of Enterprise Architect, if security is enabled you must have Administer Database permission to repair a project

Reference Data

Reference data is used in Enterprise Architect to provide meta-information for the modeler, and is often seen in drop down lists and selection boxes. It includes such things as status codes, requirement types, model authors and more.

For convenience Enterprise Architect comes pre-configured with useful reference data, but it is best practice to review this existing data when setting up a project and either replace or edit it to ensure it is appropriate for the project.

There are a number of different types of reference data, some of which apply at a repository level - such as status codes and authors - and some that is more technical in nature, such as code engineering and database datatypes.

The reference data can be copied between repositories by exporting it from one repository and importing it into another; this mechanism can save time and ensure consistency between repositories.

Project reference data can be entered and edited through the 'Configure > Reference Data' ribbon and includes:

- Stereotypes
- Tagged Value Types
- Cardinalities
- People
- General Types
- Project Indicators
- Estimation Factors
- Datatypes

Notes

- In the Corporate, Business and Software Engineering, System Engineering and Ultimate editions of Enterprise Architect, if security is enabled you must have Manage Reference Data - Update permission to update and delete reference items

UML Types

Using the 'UML Types' dialog you can configure stereotypes, Tagged Value types and the cardinality list for your project. In more detail, you can:

- Add, modify and delete stereotypes
- Customize stereotypes appearance
- Updating shape scripts and metafiles associated with Stereotypes
- Create/modify tagged values
- Define cardinality rules

Access

Ribbon	Configure > Reference Data > UML Types

Stereotype Settings

Enterprise Architect has an extensive set of Standard Stereotypes that you can apply to any UML construct. Using the 'Stereotypes' tab of the 'UML Types' dialog, a Technical Developer can also customize the stereotypes for your project by adding, modifying and deleting them. For information on customizing stereotypes, see the *Custom Stereotypes* Help Topic.

Stereotypes can be modified to make use of metafiles (image files) or customized colors, or to make use of the Enterprise Architect Shape Scripts to make new object shapes to determine the shape and dimensions of the object.

Access

Ribbon	Configure > Reference Data > UML Types

Manage Stereotypes

Option	Action
Stereotype	Type in the name of the new stereotype, or click on an existing stereotype in the list in the right-hand panel.
Group name	For a stereotype that applies to an element feature (an attribute or operation) type in a name under which to group the features that have the stereotype. This is shown on diagrams in the attributes or operations compartment of the element. If you do not provide a group name, the label in the feature compartment defaults to the stereotype name.
Base Class	Click on the drop-down arrow and select an object type, so that the stereotyped object will inherit the base characteristics from that existing object type.
Notes	Type in any notes concerning the stereotype (and not the objects to which the stereotype is to be applied).
None	Select this radio button to use the default object appearance.
Metafile	Select this radio button to use an image file for the appearance of the stereotyped object.
Shape Script	Select this radio button to apply a custom shape to the stereotyped object, which you define using the Enterprise Architect Shape Scripting language.
Assign	Click on this button to select or define the associated metafile or Shape Script to be applied to the stereotyped object.
Remove	Click on this button to remove the associated metafile or Shape Script from the stereotype.
Fill	Click on the drop-down arrow and select a default color for the background of the

	object.
	This color will be applied to all occurrences of any object to which the stereotype has been applied; if the color is subsequently changed, the change is immediately applied to all occurrences of any object to which the stereotype was applied (as for changes to any other property of the stereotype).
	However, on elements created with the stereotype, the default color might be overridden by other color definitions of a higher priority that have been applied to the element, such as the 'Alternative Appearance' dialog (F4) definition.
Border	Click on the drop-down arrow and select a default color for the object border.
Font	Click on the drop-down arrow and select a default color for the stereotype font.
Reset	Click on this button to reset the stereotyped object appearance definition to the default object appearance.
Preview	Provides a visual representation of the stereotyped object definition.
New	Click on this button to clear the dialog fields so that you can define another stereotype.
Save	Click on this button to save the stereotype definition and add it to the list of current stereotypes.
Delete	Click on this button to immediately remove the selected stereotype from the list of defined stereotypes.

Shape Editor

A Technology Developer can use the Shape Editor to specify custom shapes via a scripting language; that is, to create Shape Scripts. These custom shapes are drawn instead of the standard UML notation. Each script is associated with a particular Stereotype, and is drawn for every element of that stereotype.

Access

Ribbon	Configure > Reference Data > UML Types > Stereotypes > [select or specify a stereotype] : Shape Script : Assign

Reference

Option	Action
Format	The format in which the script is written.
Import	Opens a Windows browser dialog, allowing for a script file to be loaded.
Export	Opens a Windows browser dialog, allowing for a script file to be saved.
Edit window	The area in which a script can be written.
Preview of main	A visual representation of the compiled script. Nothing displays until the Refresh button is clicked.
Next Shape	If a composite shape is defined within the Edit Window, clicking on the Next Shape button pages through the components of the shape.
Refresh	Parses your script and produces a visual example of the shape in the Preview of main window.
OK	Exit from the Shape Editor; don't forget to save your scripts from the 'Stereotype' tab.

Notes

- Shape Scripts adopt the same color gradient settings as normal elements, as defined in the 'Standard Colors' page of the 'Preferences' dialog
- If an element's appearance is modified by a Shape Script, many of the 'Advanced' context menu options for that element are disabled
- Once you have finishing writing your Shape Script, click on the OK button; to save the Shape Script you must click on the 'Stereotypes' tab

Tagged Value Types

Tagged Values are used in a variety of places within Enterprise Architect to specify additional information about an element or connector. The 'Tagged Value Types' tab of the 'UML Types' dialog enables a Technology Developer to rapidly create Tagged Values, using a range of predefined structured Tagged Values to create structured tags that adhere to a specific format. For example, for model features that use the predefined tag Boolean you can use the Tagged Values window to assign a value of True or False and no other value.

You can also add default Tagged Value names and create predefined reference data Tagged Value types and custom masked Tagged Value types.

Any Tagged Value names created display in the drop-down lists of Tagged Value names in the 'Tagged Value' dialogs for elements, operations and attributes.

Access

Ribbon	Configure > Reference Data > UML Types > Tagged Value Types

Create a Tagged Value Type

Field	Action
Tag Name	Type the new name of the Tagged Value.
Description	Type a description of the Tagged Value.
Detail	Type any additional information necessary.
New	Click to begin creating a new Tagged Value.
Save	Click to save the details of a Tagged Value.
Delete	Click to delete a Tagged Value from the Defined Tag Types list.
Defined Tag Types	A list of previously defined Tagged Values.

Notes

- You can transport these Tagged Value Type definitions between models, using the 'Configure > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options; Tagged Value Types are exported as Property Types

Cardinality

The 'Cardinality Values' tab of the 'UML Types' dialog enables you to add, modify and delete values in the default cardinality list.

The cardinality values are used to define the multiplicity of source and target elements in relationships. This is the range of instances of the role that can be active in the relationship; for example, one employee can be assigned to tasks; for the target role you define the range of instances (such as tasks) the employee could be assigned to.

The cardinality values are also used to define the multiplicity of a Classifying element; that is, the number of instances of the element that can exist. For example, the Class element 'Building Walls' might have a multiplicity of 2..n, meaning that at least two walls must exist (to support the roof) but there can be many walls if the building design required it.

The values have these formats:

- *, or 0..* - zero, one or many instances
- 0..n - zero or up to n instances, but no more than n
- n - exactly n instances
- n..* - n, or more than n instances.

Access

Ribbon	Configure > Reference Data > UML Types > Cardinality Values

Reference

Field	Action
Cardinality	Type the new name of the Cardinality Value.
New	Type a description of the Cardinality Value.
Save	Saves the Cardinality value and adds it to the Cardinality List.
Delete	Deletes a Cardinality value from the Cardinality List.
Cardinality List	A list of already defined Cardinality values.

Notes

- You can transport these cardinality values between models, using the 'Configure > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options

People

In your project, you can set up and configure the people and stakeholders involved in and relevant to this model; that is:

- Maintain Authors involved within a project
- Define role types that are captured within Enterprise Architect
- Record information on project resources
- Capture client details associated with the current model

Access

Ribbon	Configure > Reference Data > Project Types > People

Project Authors

You can define the people who are working on a project, such as the authors of specific elements.

Access

Ribbon	Configure > Reference Data > Project Types > People > Project Author(s)

Notes

- If you enter multiple names, Enterprise Architect adds them separately and in alphabetical order to the 'Defined Authors' list; if you then click on one of these names, Enterprise Architect displays that name only in the 'Name(s)' field
- If you type in a role, this is not added to the roles on the 'Project Roles' tab
- You can transport these author definitions between models, using the 'Configure > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options

Reference

Field	Action
Name(s)	Type the name of the person registered as a Project Author. If you are using a Windows Active Directory, you can select names from the directory; click on the […] button to display the 'Select Users' dialog. You can also type a list of names separated by semi-colons; this enables you to define a group of people sharing a role, such as a team of Developers, Testers or Analysts. Do not leave any spaces between the names and the semicolons.
Role	(Mandatory) Enter the role the Project Author plays in the project (such as Designer, Analyst, or Architect). You can type a role name or click on the drop-down arrow and select a role defined through the 'Project Roles' tab.
Notes	Type any additional notes concerning the Project Author.
New	Add further Authors.
Save	Add the new Author to the Defined Authors list.
Delete	Removes an Author from the Defined Authors list.
Defined Authors	Review the Project Authors already defined.

Select Users

If your company is using a Windows Active Directory, you can select the Project Author names from the local or corporate-wide directory.

Access

Ribbon	Configure > Reference Data > Project Types > People > Project Author(s) : [...]

Reference

Field	Action
Object Types...	Opens the 'Object Types' dialog, which provides a choice of object types that can be used.
Locations...	Defines the root location from which to begin a search.
Check Names	Matches object names listed in the Enter the object names to select section.
Enter the object names to select	Specify object names to search for.
Advanced...	Opens the 'Advanced' dialog to provide further search options. For more information on this dialog, please refer to Windows Help and Support.

Use to

- Select Project Authors from Windows Active Directory

Notes

- Multiple Entries can be typed into the 'Enter the object names to select' section; ensure that objects are separated by semicolons (for example: Name01; Name02)

Project Roles

People associated with a project play a role in analysis, design or implementation, such as Application Analyst, Architect, Developer and Project Manager. Project roles define the activities that resources can undertake.

Access

Ribbon	Configure > Reference Data > Project Types > People > Project Roles

Reference

Field	Action
Role	Type or select the name of the role.
Description	Type a description of the role.
Notes	Type any additional information related to the role.
New	Add further Roles.
Save	Add the new role to the Defined Roles list.
Delete	Removes a role from the Defined Role list.
Defined Roles	Review the Project Roles already defined. The 'Defined Roles' list is available for selection for any element in the model; for example, you can select roles on the 'Project Authors' page of the 'People' dialog, and the Resource Allocation window. You can also specify other roles on these dialogs, but such roles are not added to the 'Defined Roles' list.

Use to

- Define role types that are captured within Enterprise Architect

Notes

- Deleting a role has no effect on any Project Author definition having this role; the deleted role becomes a simple text entry in the Project Author definition
- You can transport these role definitions between models, using the 'Configure > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options

Project Resources

Resources are, for example, project authors, analysts, programmers and architects. That is, anyone who might work on the system over time, either adding to the model or programming and designing elements of the system outside Enterprise Architect.

Access

Ribbon	Configure > Reference Data > Project Types > People > Project Resources

Reference

Field	Action
Name	Type or select the name of the person listed as a resource. The resource name is available for use in Resource Management.
Organization	Type the name of the organization employing the resource.
Role(s)	Type or select the role the resource plays in the project (for example, Designer, Analyst, Architect).
Phone 1, Phone 2, Mobile, Fax	Type the contact telephone numbers for the resource.
Email	Type the email address for the resource.
Notes	Type any additional notes on the resource.
Available Resources	Review resources that have already been defined.
New	Add further resources.
Save	Add a new resource to the Available Resource list.
Delete	Delete a resource from the Available Resources list.

Use to

- Record information on project resources

Notes

- You can transport these resource definitions between models, using the 'Configure > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options

Project Clients

Project clients are the eventual owners of the software system.

Access

Ribbon	Configure > Reference Data > Project Types > People > Project Clients

Reference

Field	Action
Name	Type or select the name of the client.
Organization	Type the name of the organization that employs the client.
Role(s)	Type the role the client plays in the project (for example, Manager, Sponsor).
Phone 1, Phone 2, Mobile, Fax	Type the contact telephone numbers for the client.
Email	Type the email address of the client.
Notes	Type any additional notes on the client.
Defined Clients	Review clients that have already been defined.
New	Add details of further clients.
Save	Add a new client to the Defined Client list.
Delete	Delete a client record from the Defined Client list.

Use to

- Capture client details associated with the current model

Notes

- You can transport these client definitions between models, using the 'Configure > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options

General Types

When you create or edit the properties of an element, you define the type or status of the property by selecting from a drop-down list that initially contains system-provided values. You can add to or replace any of these system values with your own customized values. The properties you can customize are:

- Status
- Constraint
- Constraint Status
- Difficulty
- Priority
- Test Status
- Requirement
- Scenario

Each of these can be separately managed through the 'General Types' dialog.

Access

Ribbon	Configure > Reference Data > Project Types > General Types

Status Types

In an element 'Properties' definition, the status of the element in the development management process is defined in the element 'Status' field. You can select the appropriate value from a drop-down list, which initially contains the system values:

- Approved
- Implemented
- Mandatory
- Proposed and
- Validated

You can add to or replace any of these values with your own custom values. You can also assign a color band to each status type, and define the types of element that can display those colors.

Access

Ribbon	Layout > Diagram > Appearance > Configure Status Colors, or
	Configure > Reference Data > Project Types > General Types > Status

Customize Element Status Types

Field	Action
Status	Type the name of the status.
Description	Type a short description of the status.
Status Type Color	Using the drop-down arrow, select a color to be applied to the current status type.
Preview	Displays a visual representation of an element with the selected status type color.
Restore Default	Restore the status type color to its default setting.
Applies to...	By default, status colors only apply to Requirement, Issue and Change elements. You might decide to also apply these colors to other UML elements, such as Use Cases or Classes.
	To do this, click on the Applies to ... button and, in the 'Applied Status Colors' list, select the checkbox against each element type to which to apply the status colors.
New	Clear the dialog fields to create a new status.
Save	Save the status, and add it to the Type list.
Delete	Remove the currently selected status from the Type list.

Type	Lists the current status types and descriptions for each status.

Notes

- To display status colors on your diagrams, you must select the 'Show status colors on diagrams' checkbox on the 'Objects' page of the 'Preferences' dialog

- You can transport the status types (and the colors assigned to status types) between models, using the 'Configure > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options

- Requirement, Feature, Issue and Change elements have a status color compartment, whereas the status color for other elements is applied to the element shadow; to show the element shadows (and hence the status colors), select the 'Element Shadows on' checkbox on the 'Diagram Appearance' page of the 'Preferences' dialog

Constraint Types

In an element 'Properties' definition, each constraint on the element is defined by type in the 'Constraint' field. You can select the appropriate value from a drop-down list, which initially contains the system values:

- Invariant
- Post-condition
- Pre-condition
- Process and
- OCL

You can add to or replace any of these values with your own custom values.

Access

Ribbon	Configure > Reference Data > Project Types > General Types > Constraint

Customize Constraint Types

Option	Description
Constraint	Type the name of the constraint to define.
Description	Type a brief description of the constraint.
Note	Type any further details concerning the constraint.
New	Clear the fields to define a new constraint.
Save	Save the constraint definition and add it to the Defined Constraint Types list.
Delete	Remove the currently selected constraint from the Defined Constraint Types list.
Defined Constraint Types	Lists the currently-defined constraint types and their descriptions.

Notes

- You can transport customized constraint types between models, using the 'Configure > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options

Constraint Status Types

In an element 'Properties' definition, each constraint's status in the development process is defined in the constraint 'Status' field. You can select the appropriate value from a drop-down list, which initially contains the system values:

· Implemented

· Build

· Validated

· Approved

· Mandatory and

· Proposed

You can add to or replace any of these values with your own custom values.

Access

Ribbon	Configure > Reference Data > Project Types > General Types > Constraint Status Types

Customize Constraint Status Types

Option	Description
Status	Type the name of the constraint status to define.
Make Default	To make the selected constraint status the default for all constraint 'Status' fields, select the checkbox.
New	Clear the fields to define a new constraint status.
Save	Save the constraint status definition and add it to the Type list.
Delete	Remove the currently selected constraint status from the Type list.
Type	Lists the currently available constraint status types.
twinarrows	Move the selected constraint type higher or lower in the Type list. This defines the ordering used in the Status drop down combo boxes, and in sorting any report or list when the Constraint Status Type is used as the sort-by property.
Sort Alphabetically	Reorganize the entries in the Type list into alphabetical order.
Restore Defaults	Remove all customized constraint status values and restore the system defaults.

Notes

- You can transport customized constraint status types between models, using the 'Configure > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options

Difficulty Types

Developing a particular element might be a simple task or more complex. In the element Properties definition, the level of difficult of developing that element is defined in the 'Difficulty' field. You can select the appropriate value from a drop-down list, which initially contains the system values 'High', 'Medium' and 'Low'. You can add to or replace any of these values with your own custom values.

Access

Ribbon	Configure > Reference Data > Project Types > General Types > Difficulty

Customize Difficulty Types

Option	Description
Difficulty	Type the name of the Difficulty to define.
Make Default	To make the selected Difficulty the default for all 'Difficulty' fields, select the checkbox.
New	Clear the fields to define a new Difficulty type.
Save	Save the Difficulty definition and add it to the Type list.
Delete	Remove the currently selected Difficulty from the Type list.
Type	Lists the currently available Difficulty types.
twinarrows	Move the selected type higher or lower in the 'Type' list. This defines the ordering used in the 'Difficulty' drop down combo boxes, and in sorting any report or list when the Difficulty type is used as the sort-by property. For example: you have several Change elements on a diagram, and you select 'Diagram View'; then you sort on the 'Difficulty' column. The 'Diagram View' entries are grouped by difficulty, in the order you defined in this 'Type' field.
Sort Alphabetically	Reorganize the entries in the Type list into alphabetical order.
Restore Defaults	Remove all customized Difficulty values and restore High, Medium and Low.

Notes

- You can transport customized Difficulty types between models, using the 'Configure > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options

Priority Types

In an element Properties definition, the priority of developing that element is defined in the 'Priority' field. You can select the appropriate value from a drop-down list, which initially contains the system values 'High', 'Medium' and 'Low'. You can add to or replace any of these values with your own custom values.

Access

Ribbon	Configure > Reference Data > Project Types > General Types > Priority

Customize Priority Types

Option	Description
Priority	Type the name of the Priority to define.
Make Default	To make the selected Priority the default for all 'Priority' fields, select the checkbox.
New	Clear the fields to define a new Priority.
Save	Save the Priority definition and add it to the Type list.
Delete	Remove the currently selected Priority from the Type list.
Type	Lists the currently available Priority types.
twinarrows	Move the selected type higher or lower in the 'Type' list. This defines the ordering used in the 'Priority' drop down combo boxes, and in sorting any report or list when the Priority type is used as the sort-by property. For example: you have several Change elements on a diagram, and you select 'Diagram View'; then you sort on the 'Priority' column. The 'Diagram View' entries are grouped by priority, in the order you defined in this 'Type' field.
Sort Alphabetically	Reorganize the entries in the Type list into alphabetical order.
Restore Defaults	Remove all customized Priority values and restore High, Medium and Low.

Notes

- You can transport customized Priority types between models, using the 'Configure > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options

Test Status Types

All tests defined in the Testing window and 'Test Details' dialog have a status value to show what point in the testing cycle the test has reached. The system values you can select from the test 'Status' field are:

- Not Run
- Pass
- Fail
- Deferred and
- Canceled

You can add to or replace any of these values with your own custom values.

Access

Ribbon	Configure > Reference Data > Project Types > General Types > Test Status

Customize Test Status Types

Option	Description
Status	Type the name of the Test status to define.
Make Default	To make the selected test status the default for all test 'Status' fields, select the checkbox.
New	Clear the fields to define a new Test status.
Save	Save the Test status definition and add it to the Type list.
Delete	Remove the currently selected Test status from the Type list.
Type	Lists the currently available Test status types.
twinarrows	Move the selected type higher or lower in the 'Type' list. This defines the ordering used in the test 'Status' drop down combo boxes, and in sorting any report or list when the Test Status Type is used as the sort-by property. For example: you have run several tests on a Package, and you run a Test Details report; then you sort on the 'Status' column. The report entries are grouped by test status, in the order you defined in this 'Type' field.
Sort Alphabetically	Reorganize the entries in the Type list into alphabetical order.
Restore Defaults	Remove all customized Test status values and restore the system defaults.

Notes

- You can transport customized Test status types between models, using the 'Configure > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options

Requirement Types

In a Requirement definition, the requirement is categorized according to type or function, in the 'Type' field of the Requirement 'Properties' dialog. This helps to maintain a single set of typed requirements. You can select the appropriate 'Type' value from a drop-down list, which initially contains the system values:

- Display
- Functional
- Performance
- Printing
- Report
- Testing and
- Validate

You can add to or replace any of these types with your own customized values.

Access

Ribbon	Configure > Reference Data > Project Types > General Types > Requirement

Customize Requirement Types

Option	Description
Requirement	Type the name of the requirement.
Description	Type a short description of the requirement.
Weight	Type a weighting to apply to the requirement type.
New	Clears the dialog fields so that you can define a new requirement type.
Save	Saves the requirement type details and adds it to the Defined Requirement Types list.
Delete	Deletes a selected requirement type from the Defined Requirement Types list.
Defined Requirement Types	Lists the currently available requirement types.

Notes

- You can transport these requirement types between models, using the 'Configure > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options

Scenario Types

In the Scenario definitions for an element each scenario is of a specific type, which you define in the 'Type' field on the 'Scenario' page of the 'Properties' dialog. You can select the appropriate value from a drop-down list, which initially contains the system values:

- Exception
- Alternate and
- Basic Path

You can add to or replace any of these system types with your own customized values.

Access

Ribbon	Configure > Reference Data > Project Types > General Types > Scenario

Customize Scenario Types

Option	Description
Scenario Type	Type the name of the scenario type.
Description	Type a short description of the scenario type.
Weight	Type a value for the weighting to apply to the scenario type.
Notes	Type any additional information to describe the scenario type.
New	Clears the dialog fields so that you can define a new Scenario type.
Save	Saves the scenario type and adds it to the Defined Scenario Types list.
Delete	Deletes a selected scenario type from the Defined Scenario Types list.
Defined Scenario Types	Lists the currently available scenario types to offer in the drop-down list of a 'Scenario Type' field.

Notes

- You can transport customized scenario types between models, using the 'Configure > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options

Metrics and Estimation

TCF values, EFC values and Default Hour Rate for a project are controlled from the 'Estimation Factors' dialog.

Risk, metric and effort types for a project are controlled from the 'Project Indicators' dialog.

For further information on these see the Project Management and Project Resources topics, or specifically:

- Technical Complexity Factors
- Environment Complexity Factors
- Default Hours
- Effort Types
- Metric Types
- Risk Types

Maintenance

Within your project, you can set up Maintenance types that determine the base Problem Types to be used within the model.

Access

Ribbon	Configure > Reference Data > Project Types > Maintenance

Problem Types

NOT CURRENTLY USED

For the maintenance and change control screens, you can use the 'Maintenance' dialog to set the base Problem Types that are handled. Examples are hardware-related issues, performance problems, software bugs and network problems.

Access

Ribbon	Configure > Reference Data > Project Types > Maintenance > Problem Types

Define Problem Types

Field	Action
Problem Type	Type in the name of the problem type.
Description	Type a short description of the problem type.
Weight	Provide a weighting to apply to the problem type.
Note	Type any additional information on the problem type.
Defined Types	Displays all of the pre-defined and saved problem types.
New	Click on this button to clear the dialog fields so that you can add a new problem type.
Save	Click on this button to save the scenario details and add it to the 'Defined Types' list.
Delete	Click on this button to delete a selected scenario from the 'Defined Types' list.

Notes

- You can transport these problem types between models, using the 'Configure > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options
- You transport the problem types together with test types as a Maintenance Types file

Testing Types

You can use the 'Test Types' tab of the 'Maintenance' dialog to add testing types to the basic set that comes with Enterprise Architect. Typical test types are load tests, performance tests and function tests.

Access

Ribbon	Configure > Reference Data > Project Types > Maintenance > Test Types

Notes

- You can transport these test types between models, using the 'Configure > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options
- You can either export the test types together with the default problem types, as a Maintenance Types file, or separately as a Test Types file

Reference

Field	Action
Test Type	Type the name of the test type.
Description	Type a short description of the test type.
Weight	Type a weighting to apply to the test type.
Note	Type any additional information on the test type.
Defined Types	Displays all of the pre-defined and saved test types.
New	Add a new test type.
Save	Saves the scenario details and adds it to the 'Defined Types' list.
Delete	Deletes a selected scenario from the 'Defined Types' list.

Data Types

Different programming languages support different inbuilt data types. It is possible for you to manage and extend this set of inbuilt data types for a standard programming language, as well as define new programming languages for use within Enterprise Architect.

Access

Ribbon	Configure > Reference Data > Settings > Code Engineering Datatypes

Programming Language Datatype Options

Option	Description
Product Name	Click on the drop-down arrow and select the name of the programming language.
Add Product	Click on this button to display a prompt for the name of a new programming language. Type the name and click on the OK button.
Datatype	Type the language-specific name of the datatype.
Common Type	Type the common (or generic) name of the datatype; for example, the Java boolean datatype has a common datatype Boolean.
New	Click on this button to clear the fields to create a new datatype.
Save	Click on this button to save the newly created or updated datatype.
Delete	Click on this button to delete the selected datatype. You can delete your own custom data types, but you cannot delete any of the predefined data types.
Defined Datatypes for Programming Languages	Review the list of datatypes for the selected language, and click on any that you want to edit or delete. The details for the selected datatype display in the dialog fields.

Notes

- You do not need to set the options in the 'Size' panel
- Once you have defined at least one datatype for a new language, that language name is added to the drop-down list for the 'Language' fields in the 'Programming Languages Datatypes' dialog, in the Code Template Editor, and in the 'Properties' dialog for each Class element within the model

- You can transport these data types between models, using the 'Configure > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options

- In the Corporate, Business and Software Engineering, System Engineering and Ultimate editions of Enterprise Architect, if security is enabled you must have 'Configure Datatypes' permission to update and delete data types

Resources

The Resources window is a repository of project-standard and re-usable document generation templates, MDG Technologies, Relationship Matrix profiles, commonly-used model elements, stylesheets, UML Profiles and Patterns. The window provides facilities for creating and modifying these structures and facilities and, in some cases, for applying the facility to the current model directly from the window.

Access

Ribbon	Start > Explore > Resources
	Show > Window > Resources
	Configure > Reference Data > Resources
Keyboard Shortcuts	Alt+6

Resources available through the Resources window

Notes

- In the Corporate, Business and Software Engineering, Systems Engineering and Ultimate editions of Enterprise Architect, if security is enabled you must have 'Configure Resources' permission to maintain Resources window items

- For a number of folders in the Resources window, the right-click context menu also offers a 'Help' option that, when you select it, displays a brief description of the folder

Resource	Description
Document Generation	The Document Generation folder holds a number of sub-folders, each of which provides the facilities for generating a particular type of document or report from your model.
	• System Templates - this sub-folder itself has sub-folders containing system-defined report templates, Fragments, Cover Pages, Style Sheets and Table of Contents definitions You can right-click on these template names and select options to display the template contents or to copy the template as the basis of a user-defined template that you work on in the Document Template Editor, under a different name; you can also select the 'Override' option to create a copy of the system template as a user template with the same name
	• User Templates - this sub-folder also has sub-folders for report templates, Fragments, Cover Pages, Style Sheets and Tables of Contents, but these are the templates that you have created yourself using the Document Template Editor, either as new or by copying and editing another template You can right-click on these template names and select options to create, delete, rename, modify, move or copy a user-defined template; you have additional options to mark a report template as a Fragment, or a Fragment as a report template, if they are suitable for this reclassification You can also 'drag and drop' user-defined templates between user-defined groups, and between these groups and the appropriate standard 'type' folders; for example, from DavidCover to Cover Pages
	• Technology Templates - If you have loaded an MDG Technology containing

	document templates, those are also listed under the Technology name
	• Web Style Templates - this sub-folder contains the HTML style templates that you have created as alternatives to the system-supplied <default> template To create a new template, right-click on the folder name and select the 'Create HTML template' option, and provide a template name; this opens the HTML and CSS Style Editor, in which you create the template from HTML fragments You can right-click on the template names and select options to delete the template or modify the template contents
	• Defined Documents - this folder holds the Resource documents (report specifications) you have created in the Document Report Generator; you can right-click on the document name to generate documents individually or as a batch, open the generated document, or delete the report specification
	• Linked Document Templates - this folder holds the non-standard Linked Document templates, either user-defined or supplied with each MDG Technology enabled on the system; you can edit or delete any of these existing templates, or create new ones from scratch or by copying an existing template, including the system-provided linked document templates You can also re-assign the templates to different template groups, either another of the existing groups or a new one that you create as part of the assignment
	• Legacy Templates - if you are, or have been, using the original (legacy) Enterprise Architect report generator, this sub-folder contains the report templates that you use to generate the documents You can continue to create, modify, use and delete legacy style templates, but it is recommended that you switch to the later report generator
MDG Technologies	The *MDG Technologies* folder lists the MDG Technologies that have been imported directly into the Resources window. Each technology extends the modeling capabilities and facilities of Enterprise Architect. You can import Technologies into your project by right-clicking on the *MDG Technologies* folder and selecting the 'Import Technology' menu option. The 'Copy Technology to Application Data' dialog then gives you the option to import the Technology to: • 'User', for your individual use; the Technology is imported into the %APPDATA% folder and is not listed in the Resources window • 'Model', for all project users to access; the Technology is imported into the Resources window If you have any MDG Technologies in the folder, each has its own subfolder containing the Profiles, Patterns and code modules used in the technology. You can right-click on these and select options to delete the technology, review the Patterns or Profiles, or invoke the code modules. If you remove the technology, it is removed from the Resources window, the 'MDG Technologies' dialog, and the model. Consider the fact that some MDG Technologies can be large and might impose some delays on the workstation as they load each time a user connects to the model.
Matrix Profiles	The Matrix Profiles folder contains a list of Relationship Matrix Profiles that you or your team members have created using the Relationship Matrix or Specification Manager. You simply double-click on a Matrix Profile name to quickly open the Relationship Matrix under the settings defined in the Profile.
Favorites	The Favorites folder provides a shortcut to each commonly-used element that you have added to the folder, so that you can quickly and easily locate the elements and add them to diagrams.

Stylesheets	The *Stylesheets* folder contains XSL stylesheets that you or your team members have imported into the model. These XSL stylesheets support the optional step of converting XMI files exported from the model into alternative formats such as HTML, XSL or source code. Stylesheets imported here are then available in the drop-down lists on the 'XML Export' dialog and the 'Generate GML Application Schema' dialog. Combined with UML Profiles, this is a powerful means of extending Enterprise Architect to generate almost any content required. To import a stylesheet: 1. Right-click on the folder name and select the 'Import Stylesheet' option. 2. Locate and select the stylesheet file in the browser, and click on the Open button. 3. Provide a reference name for the file, and click on the OK button. Once the stylesheet is in the folder, you can select context menu options to reload it (if it has been changed at the source) or delete it.
UML Profiles	A Profile is a source of modified or extended element and connector types that you have developed from the UML base types. You can import each new Profile into this folder, and then create elements and connectors of the new types by dragging the icons from the Profile onto a diagram. Once you have Profiles in the folder, you can select options for arranging them in order of name or type, viewing a Profile description, or deleting a Profile. It is possible to synchronize the Tagged Values and constraints for any elements created from a Profile element in the Resources window. Importing and applying Profiles through the Resources window is no longer recommended; from Enterprise Architect release 7.0. it is recommended that you load and apply Profiles as components of MDG Technologies.
Patterns	The *Patterns* folder is a folder of Design Patterns that you or your team members have imported into the model. A Pattern captures complex new elements and features that you can drag into a diagram without having to retype or reconfigure each element. Within the folder, the Patterns are grouped by type. Once you have Patterns within a group, you can select options to delete a selected Pattern or all Patterns in the group, view a Pattern's details, or apply a Pattern to an empty diagram.

Favorites

If you have any elements that you or your team use frequently - such as the list of Actors in a system - you can add them to the Favorites folder in the Resources window, and use that as the source for conveniently dragging and dropping instances of or links to these elements into other diagrams. This makes it much easier to locate and use the elements as you create and manage your model.

▲ 📁 Favorites
    ▲ 🗐 Class
        🗐 AddressBook
        🗐 Contacts
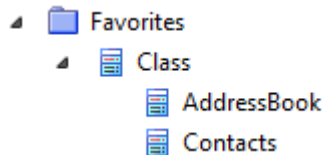
Access

Ribbon	Start > Explore > Resources > Favorites, or
	Show > Window > Resources > Favorites, or
	Configure > Reference Data > Resources > Favorites
Keyboard Shortcuts	Alt+6 | Favorites

Actions on the Favorites Folder

Action	Detail
Add an element to the Favorites Folder	• In a diagram, right-click on the element to add • From the context menu select the 'Find | Add to Favorites' option • Switch to the Resources window and expand the *Favorites* folder; the added element is listed within its element-type category (such as Class) You add elements to the *Favorites* folder individually.
Delete an element from the Favorites Folder	• Right-click on the element within the *Favorites* folder in the Resources window • From the context menu select the 'Delete Favorite' option • A confirmation prompt displays; click on the Yes button
View Properties of an element in the Favorites Folder	• Select and right-click on the element in the folder • From the context menu, select the 'Element Properties' option; the element 'Properties' dialog displays

Notes

• When you have had the *Favorites* folder open for some time and other users might have been adding to it, right-click on the folder name and select the 'Refresh Favorites' option to update your view of the folder with the new contents

Sharing Reference Data

You can conveniently update your models with reference data (including Glossary and Issue information) by exporting the data to and importing the data from .XML files to (for example):

- Copy glossaries from one model to another

- Add additional profiles by merging new stereotypes into the model

- Update reference data from files supplied by Sparx Systems as a maintenance release

- Copy resources, clients and so on from one model to another

You import data into the model automatically or manually from a reference data .XML file, exported from another model or an iteration of the current model.

Link Reference Data to a Shared Repository

If you have a number of projects that use the same sets of reference data, you can use a central project containing that common data as a shared repository. The common data includes system resources such as Security Users and Groups, Glossary terms or Data Types. Shared repositories make it possible for teams working on multiple projects to leverage a common and consistent set of system resources. Common standards and definitions can be applied across all projects using this repository. Any time a new project is started, that project can automatically inherit some or all of these common resources from the repository; there is no need to establish these resources again for each project.

Shared repositories apply to Database Management System (DBMS) based Enterprise Architect projects and can link all projects residing within one DBMS.

A new DBMS based project in Enterprise Architect is initially created with its own stand-alone repository. A single step process then allows that new project to be linked to the shared repository residing on its DBMS. Any older or pre-existing Enterprise Architect projects can also be linked to the shared repository, taking their resource data from the master version. This then allows for centralized management of the contents of the repository, promoting or enforcing common usage and standards.

Any changes to the shared repository are automatically available to every participating project.

Access

Ribbon	Configure > Model > Transfer > Shared Repository

Set up a shared repository

Step	Action
1	Create the script file to set up the shared repository, as explained in the next table.
2	Execute the generated SQL script using a third-party tool.

Select the repository tables to share

Option	Action
Script File	Type in the location into which the script file will be generated by this dialog, or click on the ⸝…⸝ button and search for it.
Main Repository	Type in the name of the current main repository.
Shared Repository	Type in the name of the database to be used as the shared repository. This name cannot be the same as the name of the main repository.
Table Groups	In the list of table groups that can be linked to the shared repository, select the

	checkbox against each table group to share in the repository.
Tables in each group	(Read only) Review the list of database tables in the currently-selected table group. All of these tables in the selected group will be available through the shared repository.
Generate	Click on this button to generate the SQL script to link the selected tables of the current (main) repository to the shared repository. The script is generated into the file location you specified in the 'Script File' field.
View Script	Click on this button to open the generated script file in the default Code Editor, and to close the dialog.
Close	Click on this button to close the dialog.

Notes

- This feature is not applicable to projects that are based on local project files (.eap or .feap)
- Shared repositories are supported for MySQL, SQL Server and Oracle

Export Reference Data

When you have a complete project set up, with fully defined project data, it is possible to capture either selected categories or all categories of that data, and export it to a single XML file. You can then review the data, and import it to another project or tool so that you do not have to define it all again. The exported data includes all instances of the selected data type in the project; for example, all defined cardinality values, or all Document Style Templates.

Access

Ribbon	Configure > Model > Transfer > Export Reference Data

Select and Export Reference Data

Step	Action
1	On the 'Export Reference Data' dialog, click on the '+' box against each table group you want to expand.
2	Select the checkbox against each table or table group to export. If you select the top-level, group-name checkbox, all tables in that group are selected for export.
3	Click on the Export button.
4	When prompted to do so, enter a valid file name with a .XML extension.
5	Click on the Save and OK buttons. This exports the data to the file; you can use any text or XML viewer to examine the file.

Notes

- You can resize the 'Export Reference Data' dialog; drag the dialog edges to the size you need
- If there are no instances of a selected data type in the project, the export does not generate any output for that data type in the XML file
- Currently, Standard Complexity Types cannot be directly edited and are therefore effectively standard for all models; they can be listed using the Predefined Reference Data Tagged Value type ComplexityTypes

Import Reference Data

It is possible to import reference data into your model from an .XML file that was exported from another model or from an iteration of the current model, either:

- Manually, whenever you know there is new or changed data to apply, or

- Automatically whenever the model is reloaded into Enterprise Architect (if the file has changed since the previous import)

The automatic import checks if the source file has changed since the last import; if the file has not changed, the import does not proceed. If the file has changed, the changed data is imported; however, you can configure the system to display a prompt for you to allow or cancel the import.

Access

Ribbon	Configure > Model > Transfer > Import Reference Data > Import File
	Configure > Model > Transfer > Import Reference Data > Shared File

Import reference data manually (Import File)

Step	Action
1	On the 'Import Reference Data' dialog, click on the 'Import File' tab and on the Select File button, then select the filename to import data from. This would be an XML file produced by the Enterprise Architect Data Exporter.
2	If you have selected a valid file, a list of available tables to import displays in the 'Select Datasets to Import' panel.
3	Click on one or more of the tables to import. Press Ctrl or Shift to click on multiple tables.
4	Click on the Import button to start the process. A message displays when the import is complete; generally the process is quite fast.

Import reference data automatically (Shared File)

Step	Action
1	On the 'Import Reference Data' dialog, click on the 'Shared File' tab.
2	If you are changing an existing configuration to import from a different XML file, click on the Clear button to clear the dialog fields.

3	Click on the Select File button and browse for the filename to import data from. This would be an XML file produced by the Enterprise Architect Data Exporter.
4	If you have entered the name of a valid file, a list of tables to import displays in the 'Datasets in File' panel. Click on one or more of the tables to import. Press Ctrl or Shift to click on multiple tables.
5	If you prefer to control whether or not the automatic import takes place, select the 'Always prompt before import' checkbox.
6	Click on the Import button to import the reference data now, and to enable the automatic check and import for subsequent reloads.

Notes

Concerning General Types data:

- For Statuses, Constraints, Requirements and Scenarios, the imported data is merged with the existing data; if a record already exists it is updated to the new values, and if the record does not exist, a new record is added - records are never deleted

- For Constraint Status Types, Difficulty, Test Status and Priority, the complete existing list is replaced by the imported list

Concerning Calendar events and event types, Team Review configurations, Patterns and Gap Matrix Profiles:

- The imported data is merged with the existing data; if a record already exists it is updated to the new values, and if the record does not exist, a new record is added - records are never deleted

Concerning Images; for MS SQLServer Repositories:

- Importing images as reference data into a SQLServer repository is not supported when connected via Cloud Services

- To import images as reference data you must have additional MS SQL Server permissions, so that you can perform 'Set Identity On/Off' commands; the role of db_ddladmin provides the required permissions; however, this approach might not be allowed in your environment, so discuss this requirement with your DBA to find an appropriate solution