Enterprise Architect

User Guide Series

The Model Repository

Sparx Systems Enterprise Architect models are held in relational databases. A simple project with a file-based repository (MS JET or Firebird) can be seamlessly scaled up to a global project on DBMS servers (such as MySQL) and a Cloud-based repository.

Author: Sparx Systems
Date: 13/11/2023
Version: 16.1

CREATED WITH ENTERPRISE ARCHITECT

Table of Contents

The Model Repository 13
Repository Overview 21
Brief Introduction 30
Managing Connections to Projects 34
Obtain Repository Connection String 40
Project Shortcuts 41
Create Project Shortcut 46
Capture Current Work Environment 50
Encrypt Repository Password 53
File Based Projects 54
Copy a Base Project 65
Sharing File Based Projects 67
Share Enterprise Architect Projects 70
Refresh View of Shared Project 72
Share Projects on Network Drive 74
Distributed Development 76
Replication 78
QEA Replication 80
Designate a Master Repository - QEA 83
Create Replicas - QEA 85
Synchronize Replicas - QEA 87
Resolve Conflicts - QEA 90
Remove Replication - QEA 94
EAP Replication 96

Designate a Master Repository - EAP 100
Create Replicas - EAP 102
Synchronize Replicas - EAP 104
Remove Replication - EAP 106
Upgrade Replicas - EAP 109
Resolve Conflicts - EAP 111
Create a Project in a MS Access 2007 Database 114
Server Based Repositories 123
Create a Project in a MariaDB Database 127
Native Connection to MariaDB repository 133
ODBC Connection to MariaDB repository 136
Create a Project in a MySQL Database 143
Native Connection to MySQL repository 150
ODBC Connection to MySQL repository 153
Create a Project in an Oracle Database 159
Native Connection to Oracle Repository 166
OLE DB Connection to Oracle Repository 169
ODBC Connection to Oracle Repository 174
Create a Project in a PostgreSQL Database 180
Native Connection to PostgreSQL repository 186
ODBC Connection to PostgreSQL repository 189
Create a Project in a SQL Server Database 196
Create a SQL Server Repository 203
Native Connection to SQL Server Repository 206
OLE DB Connection to SQL Server Repository 210
SQL Server Security Permissions 216
Open Server Connections - Native Connection 219

Connect to a Repository via ODBC 224
The WAN Optimizer 228
Pro Cloud Server Repositories 231
Cloud Repositories Overview 233
Pro Cloud Server Setup 238
Pro Cloud Server Installation 241
Installing New Versions 247
Migrating from 32 bit to 64 bit 254
Pro Cloud Server Configuration 259
Pro Cloud Server Configuration Client - Application 268
Pro Cloud Server Logon Screen 273
Database Managers 275
Add Database Manager - Native 277
Add Database Manager - ODBC 283
Add Database Manager - Firebird 287
Add Database Manager - ODBC 289
Add Database Manager - SQL Server 293
Add Database Manager - Oracle 299
Configure Database Manager 301
Configure Worker 308
Obtain Database Manager Connection String 311
Migrating OLEDB/ODBC connections to Native 313
Extended Server Properties 318
Access Control 326
Pro Cloud Server Licensing 329
Create Request 334

Add a New License 337
Remove a License 339
Renew a License 340
Manage Allocations 341
Port Configuration 344
Add a Port Definition 347
Using HTTPS (SSL) 359
Self-Signed SSL Certificates 363
Using Your Own Certificate Authority (CA) 366
Pro Cloud Server Configuration Client - Web 372
WebConfig - Login 378
WebConfig - Home Screen 380
WebConfig - Server Settings 386
WebConfig - Pro Cloud Server Licenses 396
WebConfig - New License Request 401
WebConfig - Add a License 404
WebConfig - Remove a License 406
WebConfig - Renew a License 407
WebConfig - Token Allocations 408
WebConfig - Configure Ports 411
WebConfig - Add or Edit a Port 414
WebConfig - Integration Plug-ins 418
WebConfig - Add or Edit an Integration Data Provider 425
WebConfig - Edit Model Bindings 437
WebConfig - Manage EA Floating Licenses 439
WebConfig - Floating License Groups 444

WebConfig - Add or Edit a Group 447
WebConfig - Group Entitlements 452
WebConfig - Adding a Model Connection 455
WebConfig - Edit Model Connection 460
WebConfig - Troubleshooting 466
Firewall Settings 470
Using Multiple Configurations 472
Pro Cloud Server using IIS (Optional) 478
Floating License Server 490
Install and Configure 491
Floating License Configuration Client 497
Floating License Server Connections 499
Set Shared Key Lease Periods 502
Migrate Shared Keys between Stores 504
Add Shared Keys 506
Release Shared Keys 508
Delete Shared Keys 509
Manage Group Configuration 510
Manage Group Details 513
Default Passwords for Floating License Server 520
Manage Group License Entitlement 522
Troubleshooting 525
Creating a Cloud Repository 532
Connecting Enterprise Architect to a Cloud Model 537
Pro Cloud Server Features 541
WebEA 545
WebEA Quick Start Guide 547

WebEA Quick Start Guide - IIS 560
WebEA Installation and Configuration 574
How to configure automatic viewable components 587
Manual EA Worker Configuration 593
How to configure WebEA models 597
Configure WebEA models - via Web Browser 600
Edit WebEA Model Connection settings 605
Configure WebEA models - via Text Editor 627
How to configure WebEA Localization 670
Configuring WebEA to Trust Your Own Certificate Authority (CA) 673
Single Sign-On (SSO) with WebEA 675
Configuring IIS for WebEA Windows Authentication 679
Updating WebEA to a New Version 683
WebEA User Guide 687
Login 690
WebEA's layout 695
WebEA - Model View 698
WebEA - Browser 707
WebEA Main View - Diagram 710
WebEA Main View - Object Properties 712
WebEA Main View - Object List 719
WebEA - Properties View 723
Object Creation in WebEA 728
Add Test Records 734
Add Resource Records 739

Add a Feature 743
Add Change Requests 747
Add a Document 751
Add a Defect 755
Add an Issue Notification 759
Add a Task 763
Add a Risk 767
Edit Objects 770
Discussions in WebEA 782
Reviews in WebEA 786
WebEA - Collaborate 794
Reviews 796
Discussions 798
Journals 800
Chat 802
Mail 805
WebEA - Search 810
Search 811
Watchlist 815
WebEA's Main Menu 820
Relationship Matrix 822
Troubleshooting WebEA 828
Prolaborate 847
Integrate Data from External Providers 850
Install and Configure 858
Integration Plug-ins 879
Add/Edit Data Provider 890

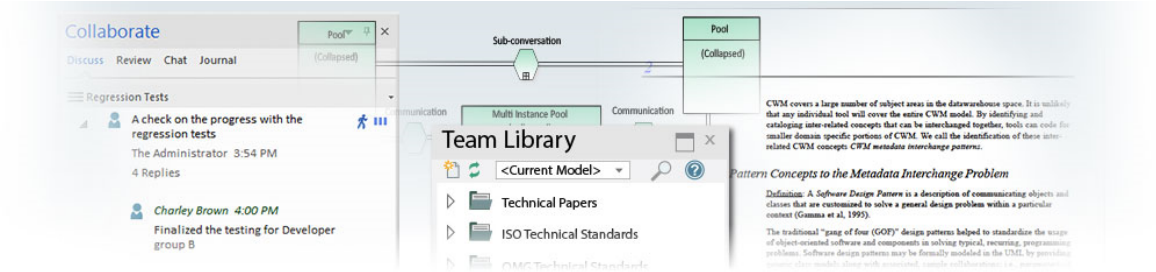
Data Provider Advanced Authentication 905
Walkthrough: ServiceNow Integration 908
Walkthrough: Jira Integration 921
Walkthrough: DevOps Integration 934
Walkthrough: Jama Integration 948
Walkthrough: Polarion Integration 958
Walkthrough: Enterprise Architect Integration 968
Troubleshooting 979
Information Accessed 986
Navigate External Data 989
External Item Details 993
Configuration 997
Linking Items 1001
Item Hyperlinks to WebEA 1007
Writing a Custom Integration Plugin 1010
ISBPIIntegrationPlugin interface 1014
CreateWebEAHyperlink 1019
GetAuthorisationDetails 1021
GetDefaultFieldMapping 1024
GetDefaultTypeMapping 1027
GetFields 1030
GetIcon 1032
GetItem 1034
GetItemDiscussion 1037
GetItemList 1039
GetItemNotes 1042
GetItems 1044

GetMenuList 1047
GetTypes 1049
ItemLinked 1051
ItemUnlinked 1053
ItemUpdated 1055
PostNewDiscussion 1057
PostNewItem 1059
PostOAuthCode 1061
PostUpdateItem 1063
PostUpdateItemNotes 1065
RefreshOAuthToken 1067
SetAuthorisation 1069
SetCallbacks 1072
SetConfiguration 1074
Custom SBPI Services 1078
ISBPIServicePlugin interface 1082
HandleRequest 1084
SetCallbacks 1088
SetConfiguration 1090
Example Script 1093
Visibility Levels 1096
Installation and Configuration 1099
User Guide 1105
Troubleshooting 1110
Pro Cloud Server Standard Features 1111
PCS Troubleshooting 1114
Project Maintenance 1127

Maintenance Overview 1130
Check Data Integrity 1131
Reset Auto Increments 1136
Upgrade a Project 1140
Upgrade Replicas 1142
Run SQL Patches 1143
Rename a Project 1145
Compact a .QEA/.QEAX Project 1146
Compact a .EAP/.EAPX Project 1149
Re-Index a Firebird Project 1151
Repair a Project 1153
Reference Data 1155
Reference Overview 1157
UML Types 1162
Stereotype Settings 1164
Shape Editor 1168
Tagged Value Types 1171
Cardinality 1174
General Types 1177
Status Types 1179
Constraint Types 1183
Constraint Status Types 1185
Difficulty Types 1188
Priority Types 1191
Test Status Types 1195
Requirement Types 1198
Scenario Types 1201

Maintenance Status Types 1204
People 1207
Project Authors 1209
Select Users 1212
Project Roles 1214
Project Resources 1217
Project Clients 1220
Risk Types 1223
Maintenance 1226
Problem Types 1227
Testing Types 1230
Metrics and Estimation 1232
Auto Names and Counters 1233
Data Types and Namespaces 1239
Data Types 1241
Coding Data Types 1242
Namespaces 1245
Sharing Reference Data 1247
Import Reference Data 1249
Export Reference Data 1253
Linking Reference Data 1255
Shared Security Reference Data 1262
More Information 1265

The 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 Discuss & Review window and to

review work that has been in progress in the Library window. Model Security can be implemented, which facilitates collaboration and prevents work accidentally being overridden.

There is a range of project management features, such as tasks and resource allocation, 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 rich, 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 (SQLite 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 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, Unified and Ultimate Editions of Enterprise Architect, 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

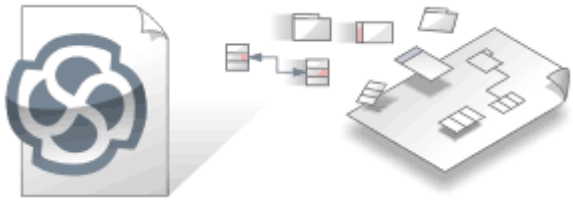
larger, Team based DBMS server would be used to provide the model sharing and collaboration capabilities. So whatever the actual needs are, Enterprise Architect has a number of solution scenarios that can be adapted to provide the right level of change management and collaborative development.

Example Repository Tasks

Task
Create a project in a simple, file-based repository (.QEA 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



Enterprise Architect repositories contain valuable information about an enterprise and the projects that it runs. The information could be at a strategic, tactical or operational level, and the models can be based on a range of disciplines from business through technology to engineering. The models can represent the current or future state of the enterprise or one of its business or technology divisions or services. The repository will become the central hub of corporate knowledge and, through integration with a range of other tools, can provide a unified view of all enterprise knowledge and its connections.

As a modeler you create content in the form of graphs of interconnected elements that are visualized on diagrams, lists and specialized representations such as Matrices, Kanban Boards, Charts, Graphs, Road Maps and more. Stakeholders outside the modeling groups are able to view these visualizations and collaborate through discussions, reviews and chat, giving feedback and asking questions and so participating in the evolution of the models.

An Enterprise Architect Repository is stored in a relational database; this can be a file based database such as SQLite or Firebird (MS Access in older versions), or a database management system (DBMS) such as MS SQL Server,

MySQL, PostgreSQL or Oracle. Enterprise Architect can connect to DBMS-based repositories via a direct connection or with the use of the Pro Cloud Server - of the type usually referred to as Cloud Based Repositories. 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 modeling platform.

Concepts

Concept	Description
Repository as a File	In any edition of Enterprise Architect you can store a project in a single file with a .qea or .feap extension. Older versions of Enterprise Architect supported .eap/.eapx file, however since the release of Enterprise Architect version 16 this format is deprecated. • A .qea file is a SQLite database • A .feap file is a Firebird database • A .eap/.eapx files are Microsoft JET databases, with .eap being a MS Access 97 based file and .EAPX being a MS

	Access 2000
Repository in a DBMS (Database Management System)	In the Enterprise Architect Corporate, Unified and Ultimate Editions you can also use a suitable DBMS repository for projects. DBMS projects have the same database schema/structure as .qea and .feap files, but provide much greater scalability and concurrent access. Enterprise Architect connects to a DBMS using either a 'Native' connection (introduced in EA v16) or ADO/ODBC (see *Connect to a Data Repository*). Sparx Systems recommends the use of Native connections going forward. Note that when configuring an ODBC data source on a 64 bit machine, you should use the matching ODBC data manager for your Enterprise Architect installation. For example, if you have installed the 64 bit version of Enterprise Architect then your ODBC DSN will need to have 64 bit definitions. To ensure the correct version of the ODBC data manager is used, use Enterprise Architect's 'Settings > User Tools > ODBC Data Sources' ribbon option.

Repository in the Cloud	In the Corporate, Unified and Ultimate Editions of Enterprise Architect you can further enable the distributed capabilities of Enterprise Architect and use the Internet or local network to connect to a Pro Cloud Server on which one or more Repositories have been installed and configured for access. The Pro Cloud Server is free software available from Sparx Systems, and is relatively simple to set up and configure on a suitable Windows-based machine. Cloud-based repositories provide the ultimate in distributed development and minimal client configuration. In fact, clients only need the URL of the server and database name in order to connect to the Cloud based repository. The Pro Cloud Server supports the same File based and DBMSs that Enterprise Architect does, ie Firebird, MariaDB, MySQL, Oracle, PostgreSQL, SQLite and SQLServer. If a DBMS server is being used, to ensure the best performance for Enterprise Architect clients, Sparx Systems recommend that the Pro Cloud Server and database servers should reside on the same local area network and be joined by a high

	speed connection.
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 view and model, necessary procedures and other management tasks to deliver the product. Project management capabilities in Enterprise Architect are

	very 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 SQLite 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 (Start Page 'Create from Pattern' tab) 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 Browser window by: • Right-clicking on an existing model and selecting the 'Add a Model using

	Wizard' option • Right-clicking on a Package and selecting the 'Add a Model using Wizard' option • Clicking on an existing model, pressing the Insert key and selecting the 'Add a Model using Wizard' menu option • Clicking on a Package, pressing the Insert key and selecting the 'Add a Model using Wizard' menu option
Opening Existing Projects	Existing Projects are accessible via the 'Recent Models' list or 'Open Project' option under the 'File Management' icon (), or the 'Recent' list or Manage Projects button on the Start Page. 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: • Firebird • MariaDB • Microsoft Access 2007 onwards • Microsoft SQL Server 2005 onwards (including Express and Azure) • MySQL • Oracle 9i onwards • PostgreSQL In brief, to create a new database repository, you first create a new database with the DBMS management software, then run supplied scripts to create the schema/structure. The database then requires initial amount of data for Enterprise Architect to function correctly. The initial data can be a copy of an existing repository or the bare minimum from a number of different sources. Details are available under the *Server Based Repositories* section.

Brief Introduction

Enterprise Architect is used to model business, technology and engineering systems. You create models for a variety of reasons including specifying requirements, analyzing design options, describing architecture, designing programming classes and databases, documenting an existing system and much more. All the artifacts associated with these models are stored in the repository, not as files, but in tables in a relational database. These models can all be stored and accessed from a single repository, sometimes called a project. The repository can be of two fundamental types:

- File-based Database - such as SQLite *.qeax files, Firebird *.feap files or MS Access *.eapx files
- Server-based RDBMS - such as MS SQL Server MySQL, PostgreSQL or Oracle

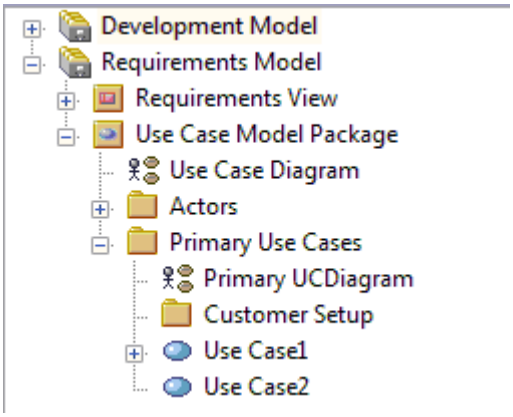
Enterprise Architect makes connections to these databases

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; the Use Case Model Package contains: • Actors and • Primary Use Cases It also contains the 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.

Managing Connections to Projects

An Enterprise Architect project is used for storing and managing the components of one or more UML models. The Professional Edition of Enterprise Architect works on file-based projects (.qea, .feap or .eap/.eapx files). If you are using the Corporate Edition (or above), you can also use one of a number of DBMSs such as MySQL, SQL Server, Oracle, 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

Other	Start Page : Manage Projects

Options

Field/Button	Action
Local File	Click on this button to open a file browser dialog, and locate and select an existing .QEA, .QEAX, .EAP, .EAPX 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 a server-based Enterprise Architect repository. The 'Open Server Connection' 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: • 'Native Connection' - the 'Open Server Connection' dialog displays • 'ODBC Connection Wizard' - the

	Windows 'Data Link Properties' dialog displays; you can also use this method to define the repository connection • '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 Note that you can also display the 'Connection String' dialog directly, by selecting the 'URL' option on the Start page.
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 select it in the list 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. Similarly, if the project has a connection string (not a shortcut), you can change the name of the project by right-clicking on it and selecting the 'Edit Project Name' menu option; type a new name in the 'Edit Project Name' dialog and click on the OK button. The list of recent projects 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	Click on a project name in the 'Recent'

Project to Open	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 Projects' 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, Unified and Ultimate Editions of Enterprise Architect
- Enterprise Architect supports these DBMS products for hosting model repositories:
  - MariaDB
  - MySQL
  - Oracle
  - PostgreSQL and
  - SQL Server, SQL Server Express and SQL Server on Azure

Obtain Repository Connection String

A 'connection string' defines all the connection information that Enterprise Architect requires to open a repository.

Connection strings are typically only available from 'Manage Projects'.

1. You can open the 'Manage Projects' dialog by:
   - Pressing Ctrl+O, or
   - Clicking on  > Open Project... or
   - Selecting the Start Page Manage Projects button
2. Find the repository in the 'Recent Projects' list. Note the 'Path' column will show the connection string; however, if it is too long for the screen it will be abbreviated and '...' shown.
3. Right-click on the repository and select the 'Edit Connection String' context menu option. This will show the 'Connection String' dialog.

When the 'Connection String' dialog is first shown the complete text of the connection string will be automatically selected. Press Ctrl+C to copy it to the clipboard.

It is usually a good idea to include this connection string in communications with Sparx Systems as it provides useful information about how you connect to the model, which can help Sparx Systems understand your environment.

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 Server or Cloud based model without requiring other users to know the full details of the server, 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	Open the Model Search with a specific

Search	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 Library	Open the default Library document. You specify the Library 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 shortcuts. The working set is easy to tailor to your changing requirements, adding greater flexibility to any shortcut 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 Library window • 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 Library window, 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.

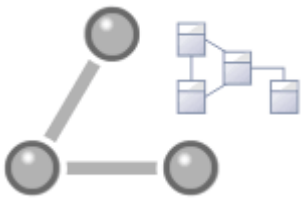
Encrypt Repository Password

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

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 a file-based repository you can be modeling within a few minutes. All editions of Enterprise Architect support these types of repository in the form of:

- A .qea or .qeax file as a SQLite database, the default type in Enterprise Architect Release 16.0 and later (recommended in both 32 bit and 64 bit versions)
- A .feap file as a Firebird repository in all editions of Enterprise Architect
- A legacy .eap or .eapx file in all editions of Enterprise Architect except Release 16.0 64 bit
- A .eadb file as an MS Access 2007+ database

You can create your own repository from scratch, but default repositories (EABase files) are conveniently supplied in .qea, .feap and .eap formats, located in the Enterprise Architect installation directory. The EABase files act as templates for creating new projects and can be copied freely to easily allow you to create new repositories. You can tailor these repositories to suit your organization or create your own base repository.

Note that:

- QEA files are based on the well regarded open source SQLite3 code base, which has many useful tools and is extendable to meet emerging and changing needs; it does not require third party or custom drivers to be installed
- QEA files are the default file-based format, recommended and preferred for single user, single desktop, file-based modeling
- QEA files work seamlessly across Windows, Linux and Mac (the last two using Wine) on both 32 bit and 64 bit versions
- QEA files also provide basic replication support for merging changes between a replica set
- QEAX files are recommended for very small work groups sharing a file on a network share
- EAP/EAPX files are not supported under Linux/Mac but might work with additional third-party drivers
- EAP/EAPX files are generally supported under Windows, but require additional drivers installed from Microsoft

QEA files (SQLite databases) have been incorporated to better support the 64 bit version of Enterprise Architect, and to avoid dependency on various operating system capabilities.

File-based Project Tasks

Firstly, to create any file in Enterprise Architect, select one of:

-  | New Project menu option

- 'Local File | New Project' option on the 'Manage Projects' dialog, or
- 'Create new' 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, using an appropriate file extension.

Once the project has been saved, the 'Create from Pattern' tab (Model Wizard) displays, which makes a selection of model patterns available; select the pattern(s) to use.

Enterprise Architect adds a model containing the selected model Packages to the Browser window.

Task	Detail
Create .QEA and .QEAX Project Files	From Enterprise Architect version 16.0 onwards, .qea/.qeax is the default and recommended file based format, due to its easy of setup and general performance. • Files with the extension of .QEA support single users. • The .QEAX support muilti-user access for a small group less than five users. The internal format of both files is identical, however the .QEAX extension indicates that file sharing will be enabled. A QEAX file can be renamed back to QEA - no data transfer is required.

Create .FEAP Project Files (Firebird)	When you create a .feap file in Enterprise Architect, the system 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
Create .EAP/.EAPX Project Files (MS Access Jet)	Create your project file with the .eap extension under JET 3.5 (does not support unicode text) or .eapx under JET 4 (does support unicode text). You can also copy and rename the EABase.eap or .eapx template file, and build your project up from that. Note that you cannot create or load a .eap or .eapx file in Enterprise Architect 16, 64 bit under Wine. Under any other operating system, if the 64 bit access driver is not installed, the system prompts you to convert the file to another format such as .feap.
Create .EADB Project Files	Introduced in Enterprise Architect v16, if you have MS Access 2007+ (or the MS Access database engine) installed, you can create a .eadb project file in these

	ways: • Select to create a new project file and give it the .eadb file extension • Copy and rename the EABase.eadb file (in the installation folder) and build up your project from that Prior to Enterprise Architect v16 the only way to make use of MS Access 2007+ databases was with the .ACCDB ODBC driver (see the next point). However, the .eadb extension is now internally recognized by Enterprise Architect, so once MS Access 2007+ (or the database engine) is installed, Enterprise Architect can work with these databases without any further configuration, thereby greatly improving the usability of this database format. The .eadb files can be opened in all editions of Enterprise Architect, including the Professional. Note: Enterprise Architect does not supply drivers for this repository format, they are available directly from the Microsoft site. see: driver issue page
Create .ACCDB Project Files	Enterprise Architect can use a MS Access 2007+ database (extension .ACCDB) as repository; however, the local machine

	requires either MS Access 2007 (or later) or the MS Access 2013 (or later) database engine installed. Having one of these products provides the local machine with a Microsoft Access ODBC driver that Enterprise Architect can make use of.
Location of default (EABase) repository template	The default installation directories, depending on which edition you have installed, are: • Registered Edition: C:\Program Files\Sparx Systems\EA • Trial Edition: C:\Program Files\Sparx Systems\EA Trial • Lite Edition: 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, it is also possible to start a new project by copying an existing one. While it is possible to simply 'copy' the EABase 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, there are a number of things that you might want to do to set it up for use by yourself or other team members. You might want to: • Enable Security and set up Users, Groups and Permissions • Set Project Options and Personal Preferences • Set up Technologies, Perspectives and Ribbon Sets • Create a Repository structure in the Browser • Use patterns to create initial project content • Set up Report Templates • Set up default database and programing language options
Create Custom	You can customize any Enterprise Architect project as a template project -

Templates	meaning that you can use it as the basis for creating other project files. This could include 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 (.qea, .feap and .eap files) are best suited to single user development with the model file placed on the modelers local drive. This is a fast and a very efficient 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 repository 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 .qea, .feap or .eap file format is ideal.
Small	With all versions of Enterprise Architect

Workgroup	it is possible to share a project file at the network file level. Note that this is a suitable solution for small workgroups (less than five members) who are collaborating on a shared model. The solution very much depends on the quality of the network connection as Enterprise Architect on the user's machine is still running all the queries and performing all database operations on a network file. Therefore, it is less suitable as collaborating users increase in number and/or the model grows larger, at which stage 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 .qea, .feap or .eap 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 project 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 possibility for team-based development is to use the built-in replication features of the 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.

Up-sizing

A project to be easily scaled up when the team further increases in size or when organizational policy dictates. In particular, given that file based multi-user access is limited to the workgroup having fewer than five concurrent users, there are two good options:

- A file-based repository can be easily transferred to a DBMS-based repository using the Project Transfer feature
- The free Pro Cloud Server can be used with either a .feap file, or a free DBMS like MySQL, with both supporting unlimited users

Notes

- You can also add Model Packages to a project by clicking on the Browser window 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 (Start Page 'Create from Pattern' tab) 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 QEA, .QEAX, .FEAP, .EAP or .EAPX 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

Access

Either:

- In your file browser, right-click on the template project file or existing operational project file and select the 'Copy' and then 'Paste' menu options, or
- Use the Project Transfer facility to transfer the contents of the base, or source, project into a shell target project

In the first method you create a duplicate of the source project, with the same GUIDs. In the second method you have the option of not changing the GUIDs, or deleting and replacing all GUIDs in the new project so that the target model structures are unique and separate from the source structures.

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 Native/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.
XML Import and Export	XML Import and Export can be used to export and share discrete Packages between developers; Native/XMI file enables the export of Packages into XML files that 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, Unified 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, Unified and Ultimate Editions of Enterprise Architect

- DBMS repositories are supported in the Corporate, Unified 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 Browser window and select the 'Contents | Reload Current Package' option, or • Press F5 to refresh the Package, or • Select the  (Reload Project) icon in the Project Tools toolbar, or • Press Ctrl+Shift+F11, or • Close the project and reopen it
Diagram	• Select the 'Start > Appearance > View > Manage 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 Projects on Network Drive

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 Browser window (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	(.EAPX/.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 ('Settings > Model > Integrity > Manage .EAPX/.EAP File > Repair .EAPX/.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.

XML Import/Export

Using XML-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

XML-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.

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

Replication

Replication is a useful means of sharing projects between isolated or mobile users, enabling users or groups to work independently then merge their changes to a centralized master repository.

Although the Pro Cloud Server provides centralized repository access for distributed teams, there can be cases where parts of a group working collaboratively on a master repository, need separate repositories for working in disparate locations, but with the ability to synchronize their changes back into the master repository. This model synchronization process is achieved using model replication.

In Replication:

- A file-based repository is converted to a design master, then replicas are made of the master
- Users take the replicas away, modify the project, then return their replicas back to be synchronized with the master file.

Enterprise Architect supports replication on two file-based repository types:

- .qea and .qeax - SQLlite
- .eap and .eapx - Jet 3.5 and Jet 4.0 respectively

With a decreased support for Jet (MS Access databases), especially in 64 bit applications, replication using Enterprise Architect's SQLite file-based repository is the preferred option.

Access

Ribbon	Settings > Model > Transfer > Replication

QEA Replication

When using replication on the 64 bit version of Enterprise Architect it is necessary to use the SQLlite repository type. This includes the .qea and .qeax file types.

Access

Ribbon	Settings > Model > Transfer > Replication > QEA Replication

Guide

This table identifies the key steps in the process of setting up and using replication for .qea files.

Facility	Detail
Create a Master	Convert a base repository into a replication master repository.
Create Replicas	Create replicas from the master repository. These replicas can then be posted off-site and worked on it as required.
	When a work-cycle is complete, the

Synchronize	replica repositories can be returned for synchronization with the master repository. When synchronizing, all changes in the master and the replica are propagated in both directions, ensuring they both finally contain the same information.
QEA 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.
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.

Designate a Master Repository - QEA

Creating a master repository is the first step when working with Enterprise Architect's replication. This process converts an existing QEA repository to a master repository. This master repository provides a base for creating QEA replicas that can be modified independently and later synchronized back to the master.

Access

Ribbon	Settings > Model > Transfer > Replication > QEA Replication > Make Design Master

Create a design master

Step	Action
1	Create a back-up of the required .qea or .qeax repository.
2	Open the repository

3	Select the 'Make Design Master' menu option and follow the on-screen instructions.

Master usage

Once created you can work with the master and/or replica files as normal. Any changes applied on the master will be propagated to the replicas on synchronization. This can be useful where there are requirements or foundation classes that are only updated by higher level teams on the master.

Create Replicas - QEA

A replica QEA repository is one of several copies of a QEA master repository. You create replicas of the master repository for users to modify independently then synchronize these back to the master.

You can create as many replicas as needed. It is recommended to give these meaningful names to help identify each replica. For example if the master is called "Model.qea" then suitable names could be "Model-ReplicaDevGrpA.qea" or "Model-Janes-Replica.qea". This will help identify the replica when synchronizing changes back to the master and help avoid mistakes synchronizing the wrong file.

Access

Ribbon	Settings > Model > Transfer > Replication > QEA Replication > Create New Replica

Create a replica

Step	Action

1	First create a replication master repository, 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 master repository.
3	As the merge updates the replica with any changes on the master repository, for future work, ensure that you use the merged replica from the master.

Notes

- In the Corporate, Unified and Ultimate Editions of Enterprise Architect, if security is enabled you must have 'Manage Replicas' permission to create a replica.

Synchronize Replicas - QEA

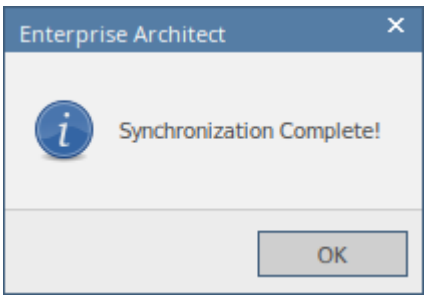
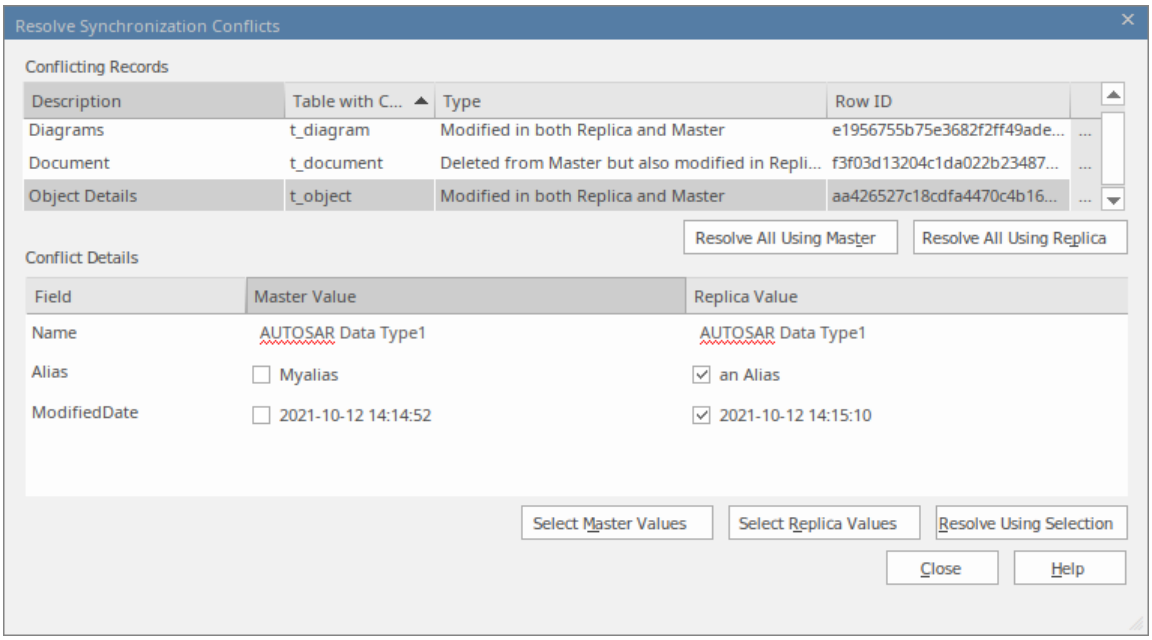
The process of synchronizing replicas to a master involves merging the changes made to each replica with the changes in the master repository. This updates each replica with all the changes ready for distribution.

Access

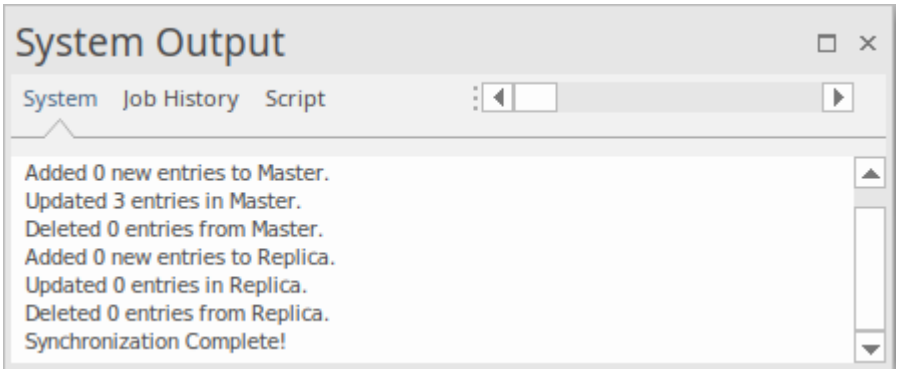
Ribbon	Settings > Model > Transfer > Replication > QEA Replication > Synchronize Replicas...

Synchronize replicas

Step	Action
1	Open the master repository file.
2	Select the 'Synchronize Replicas...' option.
3	In the file selection dialog, locate and select the required replica repository to merge with the master repository.

4	Where there are no conflicts, in that changes are on separate items in each repository, then the replication will finish with the dialog: 
5	Where there are conflicts occurring due to a change on the same item in the master and in the replica, for example, an Element's name is changed in both, then the *Resolve conflict* dialog is opened for user interaction in resolving the conflict(s).  For more details see the *Resolve Conflicts - QEA* Help topic.
6	The details on the number of updates on each side

are listed in the System Output window:

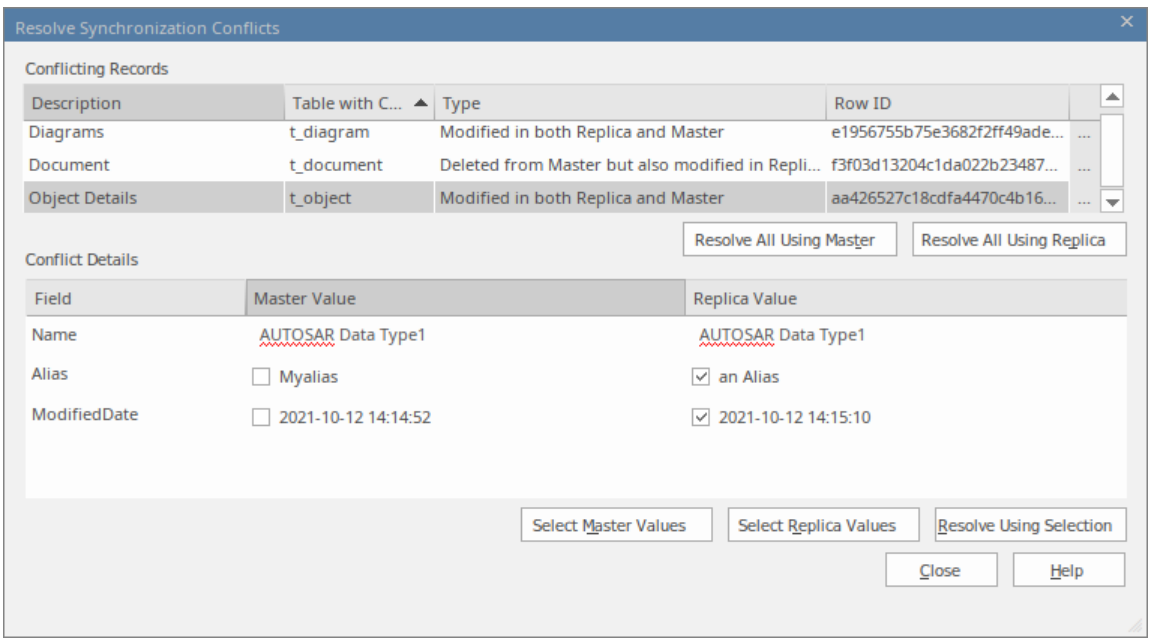


Notes

- 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, Unified and Ultimate Editions of Enterprise Architect, if security is enabled you must have 'Manage Replicas' permission to create a replica

Resolve Conflicts - QEA

If two or more people each work on the same Element/Object in their respective replica/master repositories, then when attempting to synchronize these, the replication engine has a problem resolving which change is to be synched to both. To resolve this, you need to select which of the two conflicting changes should be saved to the master or to the replica. To facilitate this, at the end of the synchronisation, all conflicting changes will be displayed in the *Resolve Synchronization Conflicts* dialog:



Resolve Synchronization Conflicts

The *Resolve Synchronization Conflicts* dialog provides a list of conflicts and a variety of means to resolve these. This process can be worked through by:

- Resolving all to the master
- Resolving all to the replica

- Resolving on a record-by-record basis or, more narrowly, by individual fields.

For this process the two lists in the dialog cover:

List	Entries
Conflicting Records	The top list shows the database row that has a conflict, and the type of conflict. Click on the [...] button, on the end of the row, to either: • Copy details or • Find the item in the Browser or in a Diagram. Click on an entry to show the details in the *Conflict Details* list.
Conflicting Details	On selection of an item in *Conflicting Records* the bottom list *Conflicting Details* will show the individual fields that were modified in both the master and the replica repositories.

Resolving conflicts

Button	Action
Resolve All	On clicking the button *Resolve All Using*

Using Master	*Master* - all the changes in the master are written to the replica.
Resolve All Using Replica	On clicking the button *Resolve All Using Replica* - all the changes in the replica are written to the master.
Resolve Using Selection	In the *Conflict details list*, you have these options: • Click on the *Select Master Values* button, this ticks all items in the *Master Value* column • Click on the *Select Replica Values* button; this ticks all items in the *Replica Value* column • Use the tick-boxes to select specific values to be updated to both repositories After selecting one of these you can write the update using the button *Resolve Using Selection*.

Considerations

Consideration	Detail

Avoid the problem	Ensure that each team member always works in a separate area of the model within their replica.
Conflict cannot be resolved	If a conflict cannot be resolved yet, you can close the conflict window. Where a conflict is not resolved, the master and replica files will be left as they are. The Synchronization can be re-run later, and the conflict will be listed again.

Remove Replication - QEA

Given that the replication requires extra detail there are many additions to a replication repository's structure. This makes the repository size considerably larger. If, at some point, you decide not to use the replication feature any more, you can remove the replication data.

As the synchronization of the replicas should be completed before this, only a master can have the replication removed. The replica repositories, being duplicates, can be discarded.

Access

Ribbon	Settings > Model > Transfer > QEA Replication > Remove Replication

Remove replication from your model

Step	Action
1	Open the .QEA file.
2	Select: Settings > Model > Transfer > QEA Replication >

	Remove Replication

Notes

- When removing replication it is strongly recommended that you also reset the ID's to restore a normal functioning model.
  Background; the IDs in a replicated model use random numbers in the accepted range. So, a high number may have been assigned, meaning further increments for the identifiers may exceed the database identifier range. See the *Reset Auto Increments* Help topic for more detail.

EAP Replication

Replication is supported for .eap files and .eapx files. These files are based on the Microsoft Jet database formatting (MS Access), and support the Jet replication.

Note: with decreased support for Jet/DAO, especially in 64 bit applications, replication using Enterprise Architect's new SQLite file-based repository format (.qea file) is the preferred option.

Access

Ribbon	Settings > Model > Transfer > Replication

Guide

The process to set up replication is basically the same as for .eap files, as for .qea files. To use replication:

Facility	Detail
Create a Master	Convert the base project into a design master.
Create	Create replicas from the design master.

Replicas	These replicas can then be posted off-site and worked on it as required.
Synchronize	When a work-cycle is complete, the replica repositories can be returned for synchronization with the master repository. When synchronizing, all changes in the master and the replica are propagated in both directions, ensuring 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	If you have converted a project to a design master but now want to disable the

Replication Features	replication features, you can remove Replication; ensure that you back up all your files first.

Designate a Master Repository - EAP

Creating a master repository is the first step when working with Enterprise Architect's replication. This process converts an existing EAP repository to a master repository. This master repository provides a base for creating EAP replicas that can be modified independently and later synchronized back to the master.

Access

Ribbon	Settings > Model > Transfer > EAP Replication > 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 Browser window.
3	Select the 'Make Design Master' menu option and

	follow the on-screen instructions.

Create Replicas - EAP

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	Settings > Model > Transfer > EAP Replication > 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, Unified and Ultimate Editions of Enterprise Architect, if security is enabled you must have 'Manage Replicas' permission to create a replica

Synchronize Replicas - EAP

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	Settings > Model > Transfer > EAP Replication > 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, Unified and Ultimate Editions of Enterprise Architect, if security is enabled you must have 'Manage Replicas' permission to create a replica

Remove Replication - EAP

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	Settings > Model > Transfer > EAP Replication > Remove Replication

Remove replication from your model

Step	Action
1	Open a .EAP file (any .EAP file, other than the one having replication removed) - the menu option is not available if no file 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.

- When removing replication it is strongly recommended that you also reset the ID's to restore a normal functioning model.
  Background; the IDs in a replicated model use random numbers in the accepted range. So, a high number may have been assigned, meaning further increments for the identifiers may exceed the database identifier range. See the *Reset Auto Increments* Help topic for more detail.

Upgrade Replicas - EAP

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 - EAP

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	Settings > Model > Transfer > EAP Replication > 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 Browser window or diagrams, by right-clicking on it and selecting the 'Find in Project Browser' or 'Find in Diagrams' option.

Create a Project in a MS Access 2007 Database

With the release of Enterprise Architect v16, while it is still possible to utilize MS Access 2007 (.ACCDB) based repositories using an ODBC based connection, the introduction of the .EADB file type greatly simplifies the creation process. Therefore Sparx Systems recommends that all new MS Access 2007 based repositories should be created as .EADB. But these instructions remain for historical purposes.

The .ACCDB database format was introduced with MS Access 2007 and remains the format for all versions of MS Access since. In order for Enterprise Architect to be able to read data from a MS Access 2007, databases the local machine must have the appropriate ODBC driver (*Microsoft Access Driver (*.mdb, *.accdb)*) installed. The ODBC driver (and supporting file) are installed by either having MS Access 2007 (or later) or the MS Access 2013 (or later) redistributables (sometimes referred to as the Access Database Engine or ACE) installed. MS Access requires a license, while the redistributables can be freely downloaded from the Microsoft website.

To create a project in a MS Access 2007 database, you work through these stages:

- Stage 1: Obtain the EABase.accdb file
- Stage 2: Create ODBC DSN
- Stage 3: Connect to the Repository

- Stage 4: (optional) Transfer an existing repository into the MS Access 2007 database

Prerequisites

- *MS Access 2007* (or later) or *MS Access 2013 Access Database Engine* (or later) is installed on the local machine

1. Obtain the EABase.accdb file

Step	Action
1.1	Download *eabase-XXXX-accdb.zip* (where XXXX is the latest schema version number) from the Sparx Systems website, and unzip the file contained within to a temporary location.
1.2	Move file EABaseXXXX.accdb to an appropriate location on your local hard drive and at the same time rename the file to something meaningful for the current project.

2. Create ODBC DSN

Step	Action
2.1	The Windows Operating System includes two different versions of the ODBC administrative tool, one for 32 bit and the other for 64 bit applications. Therefore you need to open the 'Open Data Source Administrator' that corresponds to the architecture of your Enterprise Architect application. The easiest way to achieve this is to use Enterprise Architect's build-in ribbon option: 'Settings > User Tools > ODBC Data Sources'. The ODBC Data Source Administrator window displays.
2.2	The DSN can be saved as either a User or System DSN, if you have administrator privileges on the local machine then we would suggest saving the DSN as a System DSN, otherwise your only option is to create a User DSN.
2.3	Click on the Add button. The 'Create New Data Source' dialog displays, enabling you to add a new DSN.
2.4	Select 'MS Access Driver (*.mdb, *.accdb)' from the list. Note: Make sure NOT to select 'MS Access Driver (*.mdb)' *without* the '*.accdb'; this is the older driver

	and does not work with MS Access 2007 databases.
2.5	Click on the Finish button. The 'ODBC Microsoft Access Setup' dialog is displayed.
2.6	Enter these configuration details: • **Data Source Name**: (DSN), a unique name for the connection • **Description**: (optional) description of the DSN
2.7	Click on the Select button to display the Select Database screen. Use this to browse for and select the database file, which was renamed in step 1.2.
2.8	Click on the OK button on the Select Database screen.
2.9	Click on the OK button on the ODBC Microsoft Access Setup screen. This will save the ODBC DSN.
2.10	If the test succeeds, click on the OK button to complete the configuration. If the test does not succeed, review your settings.

3. Connect to the Repository

Step	Action
3.1	Open the 'Data Link Properties' dialog using one of these methods: • Keyboard Shortcut: Ctrl+O : Connect to Server • Start Page | Server Connection (direct to 'Open Server Connection' dialog) •  | Open Project : Connect to Server • Start Page : Manage Projects : Connect to Server
3.2	On the 'Provider' tab select 'Microsoft OLE DB Provider for ODBC Drivers', click on the Next >> button.
3.3	Select the ODBC DSN defined in Stage 2 in the 'Use data source name' combo; click on the Test Connection button.
3.4	Once step 3.3 is successful, click on the OK button to open the repository. If the test does not succeed, review your settings.

4. Transfer existing repository contents

This step is only required if your new MS Access 2007

repository should be loaded with the contents of another repository, if you are wanting

4.1. Perform a Project Integrity Check

To avoid any potential data issues with the source repository, a Project Integrity Check should be performed. While this step is optional, Sparx Systems do recommend that it is performed.

Step	Action
4.1.1	Using Enterprise Architect, open the source project .EAP/.EAPX file.
4.1.2	Select the 'Settings > Model > Integrity > Project Integrity' ribbon option.
4.1.3	Ensure that in the 'Action' panel 'Report Only' is selected, and that all checkboxes in the 'Checks to Run' panel are selected, then click on the Go button.
4.1.4	If Enterprise Architect detects that the existing model contains integrity issues, they will be listed in the main panel at the bottom of the dialog. These issues should be resolved before continuing with the

	project transfer.

4.2. Transfer Source Project to an Empty .ACCDB File

From Enterprise Architect Release 16.0 onwards, there is an alternative to the .accdb file that is easier to use; this is the Access .eadb file. Enterprise Architect uses .EADB as the extension to connect to the file using the Access ODBC driver. See the *File Based Projects* Help topic.

Note that if the .eap or .eapx file has replication enabled, this must be removed before performing the transfer. See the *Remove Replication* Help topic.

Step	Action
4.2.1	Open Enterprise Architect. (If the 'Open Project' dialog displays, click on the Cancel button to open with no project loaded.)
4.2.2	Select the 'Settings > Model > Transfer > Full Project Transfer via Connection' ribbon option. The 'Project Transfer' dialog displays.
4.2.	In the 'Transfer Type' panel, select 'File to DBMS'.

3	
4.2.4	In the 'Source Project' field, click on the  button and browse for and select the name of the .eap or .eapx file to copy to the repository.
4.2.5	At the right of the 'Target Project' field, click on the  button and click on the 'ODBC Connection Wizard' option. The 'Datalink Properties' dialog displays.
4.2.6	Select 'Microsoft Office 12.0 Access Database Engine OLE DB Provider' from the list. Click on the Next button.
4.2.7	Obtain an empty MS Access 2007 base project from the Sparx Systems *DBMS Based Repositories* web page, and rename the file to suit the intended purpose.
4.2.8	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.
4.2.	If required, select the 'Logfile' checkbox and type a path and filename for the data transfer log file.

9	
4. 2. 10	Click on the Transfer button to begin the data transfer process. When the process is complete, you will have created your project in an Access 2007 database and can open it directly from Enterprise Architect, browsing for the .EADB file location in the 'Open Project' dialog.

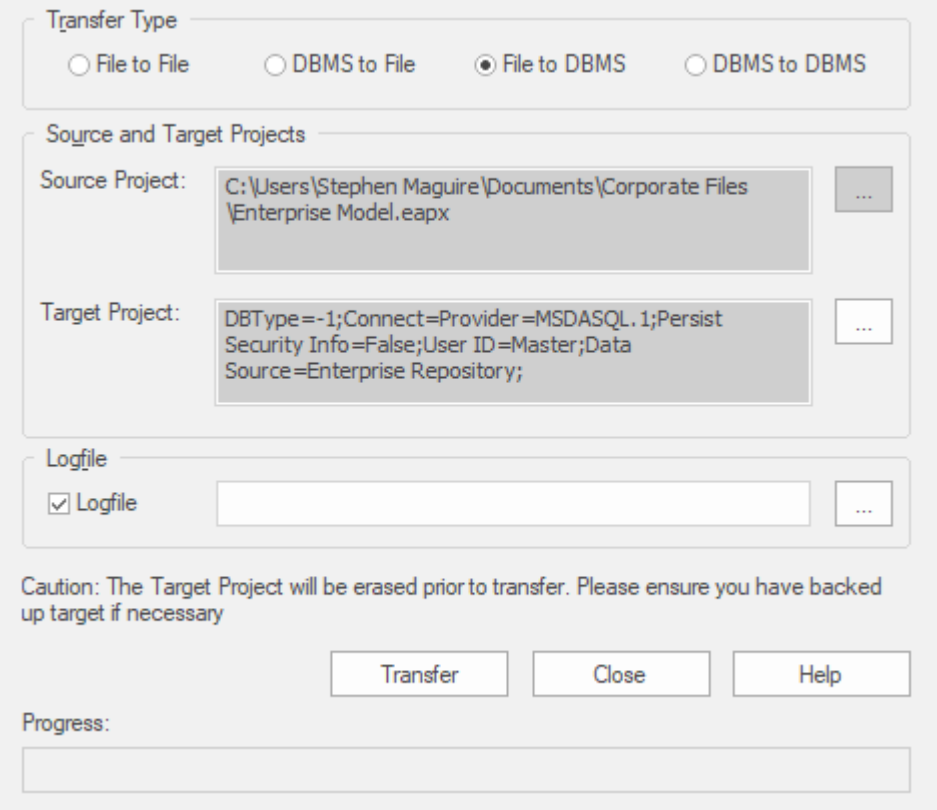
Notes

- Connecting to DBMS based Repositories is available in the Enterprise Architect Corporate, Unified and Ultimate Editions
- If you do not have Access 2007, you can download the Access Database Engine from the Microsoft downloads site

Server Based Repositories



As an Enterprise Architect repository becomes larger, or the number of concurrent users increases, or as 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, Unified or Ultimate editions. If you have started modeling with a file-based repository, Enterprise Architect has a function to transfer your repository from that to a server-based one, helping you to get started quickly. The same user interface is used for both file- and server-based repositories so the applications will, for all intents and purposes, appear the same.



The performance of the repository as experienced by end users will depend very much on the quality, power, size and performance of the server computer 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 each repository is different and although Sparx Systems does its best to maximize performance based on what is expected to be held in a repository, very occasionally this won't be optimal. 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 database.
2. Create the required tables in your database, by running a script supplied on the Sparx Systems website.
3. (For certain DBMS products, and if you are not using the Enterprise Architect Native Connection to access the database) 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 .QEA, .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; we recommend using the Enterprise Architect Native Connection for this.

DBMS Products You can Use

You can set up your project on a repository in:

- Firebird from v2

- MySQL from v5
- MariaDB
- Microsoft SQL Server from 2005, all editions including Express and Azure SQL Database
- Microsoft Access from 2007
- Oracle from 9i (all editions)
- PostgreSQL from v8

For information on creating a project on a specific DBMS from this list, see *Learn more*.

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 MariaDB repository, you work through these stages:

- Stage 1: Create a database and tables
- Stage 2: Set up a connection to the database
- Stage 3: Load initial data

Prerequisites

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

1. Create a MariaDB database and tables

Step	Action
1.1	Create a new empty database and configure it to your preferences, making sure to configure the correct character set and collation. The exact method you use to create the database is not important to this discussion, therefore use the most convenient or familiar method. For example the MariaDB SQL command to create a new database called 'ea1558' is:

	> create schema `ea1558`;
1. 2	Open your SQL management console of choice (such as HeidiSQL) and connect to the new database.
1. 3	Download **EASchema_1558_MySQL.sql** from the Sparx Systems website and open it in your SQL management console of choice.
1. 4	Run EASchema_1558_MySQL.sql to create the required database tables.

2. Set up Connection to MariaDB repository

From Enterprise Architect version 16 there are now two different ways that Enterprise Architect can connect to MariaDB repositories:

- via a Native connection (see Native Connection to MariaDB repository for details), or
- via an ADO/ODBC connection (see ODBC Connection to MariaDB repository for details)

Note: Sparx Systems recommend Native connections as they are much simpler and don't require any other software installed or configured.

3. Load Initial Data

Enterprise Architect repositories require a basic amount of reference data to function correctly.

You can load this initial data in one of two different ways:

- Transfer the entire contents of an existing model into the new database, or
- Run initial data SQL Script

3.1. Load Initial Data - Transfer Existing Model

If you choose to set up your new database with the contents of an existing model, then Sparx Systems recommend that a Project Integrity check is performed on the existing model to ensure it is 'clean' and free from data-related errors before being copied into the new database.

The Enterprise Architect installation provides a model called EABase.qea, which is copied into the installation folder specifically for the purpose of providing basic data for a new model. However, if you have customized the various drop-down lists in another model then you might prefer to use that. Regardless of what model you choose to use, the process is the same and is outlined here.

Step	Action
3.	In Enterprise Architect, open the existing repository

1.1	and select the 'Settings > Model > Integrity > Project Integrity' ribbon option.
3.1.2	In the 'Action' panel select the 'Report Only' radio button, and in the 'Checks to Run' panel select all checkboxes. Click on the Go button.
3.1.3	If Enterprise Architect detects that the existing model contains integrity issues, they will be listed in the main panel at the bottom of the dialog. These issues should be resolved before continuing with the project transfer.
3.1.4	With Enterprise Architect open, ensure that there is no project loaded, by clicking on the  button and selecting 'Close Project'.
3.1.5	Select the 'Settings > Model > Transfer > Full Project Transfer via Connection' ribbon option. The 'Project Transfer' dialog displays.
3.1.6	In the 'Transfer Type' panel, select 'File to DBMS'.
3.1.7	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 .QEA file has Replication enabled, this must

	be removed before performing the transfer.
3.1.8	At the right of the 'Target Project' field, click on the  button and select either: • 'Native Connection' - the 'Open Server Connection' dialog displays (see the Native Connection to MariaDB repository Help topic for details), or • 'ODBC Connection Wizard' - The 'Datalink Properties' dialog displays (see the ODBC Connection to MariaDB repository Help topic for details, in particular step *Connect to the Repository*)
3.1.9	If required, select the 'Logfile' checkbox and type a path and filename for the data transfer log file.
3.1.10	Click on the Transfer button to begin the data transfer process.
3.1.11	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.

3.2. Load Initial Data - SQL Script

The Sparx Systems website provides a SQL Script that contains a series of INSERT statements that will load an empty database with all the data needed by Enterprise Architect.

Step	Action
3.2.1	Open your SQL management console of choice (such as HeidiSQL) and connect to the new database.
3.2.2	Download the EABase_1558_MySQL.sql file from the Sparx Systems website and open it in your SQL management console.
3.2.3	Run EABase_1558_MySQL.sql to load the required initial data.

Native Connection to MariaDB repository

In most installations users will be able to create native connections to MariaDB database without the need to install or configure any other software.

Prerequisites

- A MariaDB database server
- An Enterprise Architect repository defined in the MariaDB database server
- You know the database user credentials that have SELECT, INSERT, UPDATE, DELETE, EXECUTE and SHOW VIEW access permissions to the Enterprise Architect repository

Access

Keyboard Shortcuts	Ctrl+O : Connect to Server
Other	Start Page | Server Connection (direct to 'Open Server Connection' dialog)  | Open Project : Connect to Server Start Page : Manage Projects : Connect to

	Server

Connect to Repository

Step	Action
1	In Enterprise Architect, open the 'Open Server Connection' screen ('Project list | Open Project : Connect to Server' ribbon option).
2	Select the MariaDB option
3	Fill in the details of the database, including Server Name, Port, Database Name, Database User and Password
4	Click the [Test] button. A "Connection Successful" popup message should be displayed, including a summary of the server's version details. If the connection is not successful, the message should include a hint or reason why it failed, the reason(s) listed will need to be corrected before continuing.
5	Once step 4 succeeds the OK button will be enabled,

	click [OK] to open the repository.

Notes

- Connecting to DBMS based Repositories is available in the Corporate, Unified and Ultimate Editions of Enterprise Architect

ODBC Connection to MariaDB repository

In order for an Enterprise Architect client to open a MariaDB repository using an ODBC connection, the client machine must have the MySQL ODBC driver installed and an ODBC DSN defined for the same architecture (32 or 64 bit) that Enterprise Architect uses.

For example, if you are using the 64 bit version of Enterprise Architect, then the client machine will need a 64 bit ODBC driver and 64 bit ODBC DSN, but if you are using the 32 bit version of Enterprise Architect then a 32 ODBC driver and DSN will be required.

Install the ODBC Driver

The ODBC driver will only need to be install once on each client machine, therefore if your machine already has the appropriate MySQL ODBC driver installed you can skip to the next step.

Step	Action
1	Download the appropriate MySQL ODBC driver for your operating system and architecture (32 or 64 bit) of Enterprise Architect from the MySQL website. Note: • Not all versions of MySQL ODBC driver function

	correctly with Enterprise Architect, our experience has determined that versions between 5.2.4 & 5.3.4 and 8.0.16 onwards have been the most stable (see more details in the notes section below). • Sparx Systems recommends the MySQL ODBC driver (instead of the MariaDB driver) because it has better performance.
2	Run the MySQL ODBC driver installer, accepting the default values.

Prerequisites

- A MariaDB database server
- An Enterprise Architect repository defined in the MariaDB database server
- You know the database user credentials that have SELECT, INSERT, UPDATE, DELETE, EXECUTE and SHOW VIEW access permissions to the Enterprise Architect repository
- You know the architecture (32 or 64 bit) of your installed Enterprise Architect application

Install the ODBC Driver

The ODBC driver will only need to be install once on each

client machine, therefore if your machine already has the appropriate MySQL ODBC driver installed you can skip to the next step.

Step	Action
1	Download the appropriate MySQL ODBC driver for your operating system and architecture (32 or 64 bit) of Enterprise Architect from the MySQL website. Note: • Not all versions of MySQL ODBC driver function correctly with Enterprise Architect, see the Driver Issues page for details of the versions that cause problems. • Sparx Systems recommends the MySQL ODBC driver (instead of the MariaDB driver) because it has better performance.
2	Run the MySQL ODBC driver installer, accepting the default values.

Create ODBC DSN

A separate ODBC DSN must be created for each MariaDB repository that the client wants to connect to.

An ODBC DSN is simply the details of a database (server and database names, Port and user credentials), stored in the

registry and given a unique name that allows applications (like Enterprise Architect) to easily connect to databases.

Step	Action
1	The Windows Operating System includes 2 different versions of the ODBC administrative tool, one for 32 bit and the other for 64 bit applications. Therefore you need to open the 'Open Data Source Administrator' that corresponds to the architecture of your Enterprise Architect application. The easiest way to achieve this is to use Enterprise Architect's build-in ribbon option: 'Settings > User Tools > ODBC Data Sources'. The ODBC Data Source Administrator window displays.
2	The DSN can be saved as either a User or System DSN, if you have administrator privileges on the local machine then we would suggest saving the DSN as a System DSN, otherwise your only option is to create a User DSN.
3	Click on the Add button. The 'Create New Data Source' dialog displays, enabling you to add a new DSN.
4	Select the appropriate MySQL ODBC Driver (usually 'MySQL ODBC x.x Unicode Driver') from

	the list.
5	Click on the Finish button. The 'MySQL Connector/ODBC' dialog displays.
6	Enter these configuration details: • **Data Source Name:** (DSN), a unique name for the connection • **Description:** (optional) description of the DSN • **TCP/IP Server**: The host name or IP address of the MariaDB server • **Port:** The Port that the MariaDB server is listening to • **User** and **Password:** database user credentials • **Database:** The database name on the selected server
7	To set the advanced options, click on the Details>> button.
8	Select these checkboxes (where provided): • 'Return matched rows instead of affected rows' ('Connection' or 'Cursors/Results' tab) • 'Allow big result sets' ('Connection' tab)
9	Click on the Test Connection button to confirm that the details are correct.

10	If the test succeeds, click on the OK button to complete the configuration. If the test does not succeed, review your settings.

Connect to the Repository

Once the ODBC driver is installed and ODBC DSN defined on an individual machine the Enterprise Architect client can open the repository using these steps:

Step	Action
1	Open the 'Data Link Properties' dialog using one of these methods: • Keyboard Shortcut: Ctrl+O : Connect to Server • Start Page | Server Connection (direct to 'Open Server Connection' dialog) •  | Open Project : Connect to Server • Start Page : Manage Projects : Connect to Server
2	On the Provider tab select 'Microsoft OLE DB Provider for ODBC Drivers', click [Next >>]
3	Select the ODBC DSN in the '*Use data source name*' combo, click Test Connection.
	Once step 3 is successful, click [OK] to open the

4	repository. If the test does not succeed, review your settings.

Notes

- Connecting to DBMS based Repositories is available in the Enterprise Architect Corporate, Unified and Ultimate Editions

Create a Project in a MySQL Database

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

- Stage 1: Create a database and tables
- Stage 2: Set up a connection to the database
- Stage 3: Load initial data

Prerequisites

- A machine with MySQL server version 5 or higher installed and running

1. Create a MySQL Database and tables

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 we recommend all users to choose InnoDB, as it supports transactions and UTF8.

Note: MySQL has a setting (*max_allowed_packet*) that sets a limit on the amount of data that can be returned by the database server. To avoid issues in the future this value needs to be larger that the largest record in your repository. How large will the largest record be? We can't advise on

that - it depends on how you use Enterprise Architect. Typically the largest element items are embedded documents and images, or baselines of large Packages. It is not unusual for customers to require a *max_allowed_packet* of 128M or, in some environments, even larger than that.

Step	Action
1.1	Create a new empty database and configure it to your preferences, making sure to configure the correct character set and collation. The exact method you use to create the database is not important to this discussion, therefore use the most convenient or familiar method. For example the MySQL SQL command to create a new database called 'ea1558' is: > create schema `ea1558`;
1.2	Open your SQL management console of choice (such as MySQL Workbench) and connect to the new database.
1.3	Download **EASchema_1558_MySQL.sql** from the Sparx Systems website and open it in your SQL management console of choice.
1.4	Run EASchema_1558_MySQL.sql to create the required database tables.

2. Set up Connection to MySQL repository

From Enterprise Architect version 16 there are now two different ways that Enterprise Architect can connect to MySQL repositories:

- via a Native connection (see Native Connection to MySQL repository for details), or
- via an ADO/ODBC connection (see ODBC Connection to MySQL repository for details)

Note: Sparx Systems recommend Native connections as they are much simpler and don't require any other software installed or configured.

3. Set up Initial Data

Enterprise Architect repositories require a basic amount of reference data to function correctly.

You can load this initial data in one of two different ways:

- Transfer the entire contents of an existing model into the new database, or
- Run initial data SQL Script

3.1 Load Initial Data - Transfer Existing Model

If you choose to set up your new database with the contents

of an existing model, then Sparx Systems recommend that a Project Integrity check is performed on the existing model to ensure it is 'clean' and free from data-related errors before being copied into the new database.

The Enterprise Architect installation provides a model called EABase.qea, which is copied into the installation folder specifically for the purpose of providing basic data for a new model. However, if you have customized the various drop-down lists in another model then you might prefer to use that. Regardless of what model you choose to use, the process is the same and is outlined here.

Step	Action
3.1.1	In Enterprise Architect, open the existing repository and select the 'Settings > Model > Integrity > Project Integrity' ribbon option.
3.1.2	In the 'Action' panel, select the 'Report Only' radio button, and in the 'Checks to Run' panel select all checkboxes. Click on the Go button.
3.1.3	If Enterprise Architect detects that the existing model contains integrity issues, they will be listed in the main panel at the bottom of the dialog. These issues should be resolved before continuing with the project transfer.
3.	With Enterprise Architect open ensure that there is

1.4	no project loaded, by clicking on the  button and selecting 'Close Project'.
3.1.5	Select the 'Settings > Model > Transfer > Full Project Transfer via Connection' ribbon option. The 'Project Transfer' dialog displays.
3.1.6	In the 'Transfer Type' panel, select 'File to DBMS'.
3.1.7	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 file has Replication enabled, this must be removed before performing the transfer.
3.1.8	At the right of the 'Target Project' field, click on the  button and select either: • 'Native Connection' - the 'Open Server Connection' dialog displays (see the Native Connection to MySQL repository Help topic for details), or • 'ODBC Connection Wizard' - The 'Datalink Properties' dialog displays (see the ODBC Connection to MySQL repository Help topic for details, in particular step *Connect to the Repository*)

3.1.9	If required, select the 'Logfile' checkbox and type a path and filename for the data transfer log file.
3.1.10	Click on the Transfer button to begin the data transfer process.
3.1.11	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.

3.2 Load Initial Data - SQL Script

The Sparx Systems website provides a SQL Script that contains a series of INSERT statements that will load an empty database with all the data needed by Enterprise Architect.

Step	Action
3.2.1	Open your SQL management console of choice (such as MySQL Workbench), and connect to the new database.
3.2.	Download the EABase_1558_MySQL.sql file from the Sparx Systems website and open it in your SQL

2	management console.
3. 2. 3	Run EABase_1558_MySQL.sql to load the required initial data.

Native Connection to MySQL repository

In most installations users will be able to create native connections to MySQL database without the need to install or configure any other software.

Prerequisites

- A MySQL database server
- An Enterprise Architect repository defined in the MySQL database server
- You know the database user credentials that have SELECT, INSERT, UPDATE, DELETE, EXECUTE and SHOW VIEW access permissions to the Enterprise Architect repository

Access

Keyboard Shortcuts	Ctrl+O : Connect to Server
Other	Start Page | Server Connection (direct to 'Open Server Connection' dialog)  | Open Project : Connect to Server Start Page : Manage Projects : Connect to

	Server

Connect to Repository

1	In Enterprise Architect, open the 'Open Server Connection' screen ('Project list | Open Project : Connect to Server' ribbon option).
2	Select the MySQL option
3	Fill in the details of the database, including Server Name, Port, Database Name, Database User and Password
4	Click the [Test] button. A "Connection Successful" popup message should be displayed, including a summary of the server's version details. If the connection is not successful, the message should include a hint or reason why it failed, the reason(s) listed will need to be corrected before continuing.
5	Once step 4 succeeds the OK button will be enabled, click [OK] to open the repository.

Notes

- Connecting to DBMS based Repositories is available in the Corporate, Unified and Ultimate Editions of Enterprise Architect

ODBC Connection to MySQL repository

In order for an Enterprise Architect client to open a MySQL repository using an ODBC connection, the client machine must have the MySQL ODBC driver installed and an ODBC DSN defined for the same architecture (32 or 64 bit) that Enterprise Architect uses.

For example, if you are using the 64 bit version of Enterprise Architect, then the client machine will need a 64 bit ODBC driver and 64 bit ODBC DSN, but if you are using the 32 bit version of Enterprise Architect then a 32 bit ODBC driver and DSN will be required.

Prerequisites

- A MySQL database server
- An Enterprise Architect repository defined in the MySQL database server
- You know the database user credentials that have SELECT, INSERT, UPDATE, DELETE, EXECUTE and SHOW VIEW access permissions to the Enterprise Architect repository
- You know the architecture (32 or 64 bit) of your installed Enterprise Architect application

Install the ODBC Driver

The ODBC driver will only need to be install once on each

client machine, therefore if your machine already has the appropriate MySQL ODBC driver installed you can skip to the next step.

Step	Action
1	Download the appropriate MySQL ODBC driver for your operating system and architecture (32 or 64 bit) of Enterprise Architect from the MySQL website. Note: • Not all versions of MySQL ODBC driver function correctly with Enterprise Architect, see the Driver Issues page for details of the versions that cause problems.
2	Run the MySQL ODBC driver installer, accepting the default values.

Create ODBC DSN

A separate ODBC DSN must be created for each MySQL repository that the client wants to connect to.

A ODBC DSN is simply the details of the a database (server and database names, Port and user credentials), stored in the registry and given a unique name that allows applications (like Enterprise Architect) to easily connect to databases.

Ste	Action

p	
1	The Windows Operating System includes 2 different versions of the ODBC administrative tool, one for 32 bit and the other for 64 bit applications. Therefore you need to open the 'Open Data Source Administrator' that corresponds to the architecture of your Enterprise Architect application. The easiest way to achieve this is to use Enterprise Architect's build-in ribbon option: 'Settings > User Tools > ODBC Data Sources'. The ODBC Data Source Administrator window displays.
2	The DSN can be saved as either a User or System DSN, if you have administrator privileges on the local machine then we would suggest saving the DSN as a System DSN, otherwise your only option is to create a User DSN.
3	Click on the Add button. The 'Create New Data Source' dialog displays, enabling you to add a new DSN.
4	Select the appropriate MySQL ODBC Driver (usually 'MySQL ODBC x.x Unicode Driver') from the list.
5	Click on the Finish button.

	The 'MySQL Connector/ODBC' dialog displays.
6	Enter these configuration details: • **Data Source Name:** (DSN), a unique name for the connection • **Description:** (optional) description of the DSN • **TCP/IP Server**: The host name or IP address of the MariaDB server • **Port:** The Port that the MariaDB server is listening to • **User** and **Password:** database user credentials • **Database:** The database name on the selected server
7	To set the advanced options, click on the Details>> button.
8	Select these checkboxes (where provided): • 'Return matched rows instead of affected rows' ('Connection' or 'Cursors/Results' tab) • 'Allow big result sets' ('Connection' tab)
9	Click on the Test Connection button to confirm that the details are correct.
10	If the test succeeds, click on the OK button to complete the configuration.

	If the test does not succeed, review your settings.

Connect to the Repository

Once the ODBC driver is installed and ODBC DSN defined on an individual machine, the Enterprise Architect client can open the repository using these steps:

Step	Action
1	Open the 'Data Link Properties' dialog using one of these methods: • Keyboard Shortcut: Ctrl+O : Connect to Server • Start Page | Server Connection (direct to 'Open Server Connection' dialog) •  | Open Project : Connect to Server • Start Page : Manage Projects : Connect to Server
2	On the Provider tab select 'Microsoft OLE DB Provider for ODBC Drivers', click [Next >>]
3	Select the ODBC DSN in the 'Use data source name' combo, click Test Connection.
4	Once step 3 is successful, click [OK] to open the repository.

	If the test does not succeed, review your settings.

Notes

- Connecting to DBMS based Repositories is available in the Enterprise Architect Corporate, Unified and Ultimate Editions

Create a Project in an Oracle Database

To create a project in an Oracle repository, you work through these stages:

- Stage 1: Create a database user to own the Enterprise Architect tables
- Stage 2: Setup up a connection to the repository
- Stage 3: Load initial data

Prerequisites

- A machine with Oracle 9i or higher installed and running
- Oracle Client installed on the client machine; if you are not using the Native Connection, please ensure that the OLE DB drivers are installed (you can download the drivers from the Oracle Technology Network web site, as part of the Oracle Data Access Components (ODAC) package; see *Learn more*)

1. Create Database Repository

Step	Action
1.1	Create an Oracle database user to own the Enterprise Architect table and configure it to your preferences, making sure to configure the correct character set

	and collation. The exact method you use to create the database is not important to this discussion, therefore use the most convenient or familiar method.
1.2	Open your SQL management console of choice (such as Oracle SQL Developer) and connect to the new Enterprise Architect repository.
1.3	Download **EASchema_1558_Oracle.sql** file from the Sparx Systems website and open it in your SQL management console.
1.4	Run EASchema_1558_Oracle.sql to create the required database tables.

2. Set up Connection to Oracle repository

From Enterprise Architect version 16 there is now a third ways that Enterprise Architect can connect to Oracle repositories:

- via a Native connection (see Native Connection to Oracle Repository for details), or
- via an ADO/OLEDB connection (see OLE DB Connection to Oracle Repository for details)
- via an ADO/ODBC connection (see ODBC Connection to Oracle Repository for details)

Note: Sparx Systems recommend Native connections as

they are much simpler and don't require any other software installed or configured.

3. Load Initial Data

Enterprise Architect repositories require a basic amount of reference data to function correctly.

You can set up this initial data in one of two different ways:

- Transfer the entire contents of an existing model into the new database, or
- Run initial data SQL Script

3.1 Load Initial Data - Transfer Existing Model

If you choose to set up your new database with the contents of an existing model then Sparx Systems recommend that a Project Integrity check is performed on the existing model to ensure it is 'clean' and free from data-related errors before being copied into the new database.

The Enterprise Architect installation provides a model called EABase.qea, which is copied into the installation folder specifically for the purpose of providing basic data for a new model. However, if you have customized the various drop-down lists in another model then you might prefer to use that. Regardless of what model you choose to use, the process is the same and is outlined here.

Step	Action
3.1.1	In Enterprise Architect, open the existing repository and select 'Settings > Model > Integrity > Project Integrity' in the ribbon.
3.1.2	Ensure that the 'Action' panel is set to 'Report Only' and that all items in 'Checks to Run' are enabled, then click on the Go button.
3.1.3	If Enterprise Architect detects that the existing model contains integrity issues, they will be listed in the main panel at the bottom of the dialog. These issues should be resolved before continuing with the project transfer.
3.1.4	With Enterprise Architect open, ensure that it there is no project loaded, by selecting 'Close Project' from the  button.
3.1.5	Select the 'Settings > Model > Transfer > Full Project Transfer via Connection' ribbon option. The 'Project Transfer' dialog displays.
3.1.6	In the 'Transfer Type' panel, select 'File to DBMS'.

3.1.17	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 .QEA file has Replication enabled, this must be removed before performing the transfer.
3.1.18	At the right of the 'Target Project' field, click on the  button and select either: • 'Native Connection' - the 'Open Server Connection' dialog displays (see the *Native Connection to Oracle repository* Help topic, then resume at step 3.1.13), or, if using the ODBC DSN • 'ODBC Connection Wizard' - The 'Datalink Properties' dialog displays (see the *OLEDB Connection to Oracle* repository Help topic for details, in particular step Connect to the Repository) • 'ODBC Connection Wizard' - The 'Datalink Properties' dialog displays (see the *ODBC Connection to Oracle* repository Help topic for details, in particular step Connect to the Repository)
3.1.19	If required, select the 'Logfile' checkbox and type a path and filename for the data transfer log file.
3.1.	Click on the Transfer button to begin the data

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

3.2. Load Initial Data - SQL Script

Step	Action
3.2.1	Open your SQL management console of choice (such as SQL Developer) and log in as the owner of the newly created Enterprise Architect repository.
3.2.2	Download the EABase_1558_Oracle.sql file from the Sparx Systems website and open it in your SQL management console.
3.2.3	Run EABase_1558_Oracle.sql to load the required initial data.

Notes

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

Native Connection to Oracle Repository

In most installations users will be able to create native connections to Oracle repositories without the need to install or configure any other software.

Prerequisites

- An Oracle database server
- An Enterprise Architect repository defined in the Oracle database server
- You know the database user credentials that have SELECT, INSERT, UPDATE, DELETE, and EXECUTE access permissions to the Enterprise Architect repository

Access

Keyboard Shortcuts	Ctrl+O : Connect to Server
Other	Start Page | Server Connection (direct to 'Open Server Connection' dialog)  | Open Project : Connect to Server Start Page : Manage Projects : Connect to Server

Connect to Repository

Step	Action
1	In Enterprise Architect, open the 'Open Server Connection' screen ('Project list | Open Project : Connect to Server' ribbon option).
2	Select the Oracle option
3	Fill in the details of the repository, including: • **Server Name**: The host name or IP address of the Oracle server • **Database Name:** The database name on the selected server (in more recent versions of Oracle, call the pluggable database) • **Port:** The Port that the Oracle server is listening to • **User** and **Password:** database user credentials
4	Click the [Test] button. A "Connection Successful" popup message should be displayed, including a summary of the server's version details. If the connection is not successful, the message should include a hint or reason why it failed, the

	reason(s) listed will need to be corrected before continuing. Note - Minimum requirements are: • **Windows 8** (earlier versions doesn't support the underlying database driver, if you require connectivity from Windows 7 then you will need to use OLEDB or ODBC)
5	Once step 4 succeeds the OK button will be enabled, click [OK] to open the repository.

Notes

- Connecting to DBMS based Repositories is available in the Corporate, Unified and Ultimate Editions of Enterprise Architect

OLE DB Connection to Oracle Repository

In order for an Enterprise Architect client to open an Oracle repository using an OLE DB connection, the client machine must have the Oracle client (including the OLE DB driver) installed for the same architecture (32 or 64 bit) that Enterprise Architect uses.

For example, if you are using the 64 bit version of Enterprise Architect, then the client machine will need a 64 bit OLE DB driver, but if you are using the 32 bit version of Enterprise Architect then a 32 OLE DB driver will be required.

Prerequisites

- An Oracle database server
- An Enterprise Architect repository defined in the Oracle database server
- You know the database user credentials that have SELECT, INSERT, UPDATE, DELETE, EXECUTE access permissions to the Enterprise Architect repository
- Oracle Client already installed on the client machine; please ensure that the OLE DB drivers are installed (you can download the drivers from the Oracle Technology Network web site, as part of the Oracle Data Access Components (ODAC) package - see *Learn more*)

Access

Keyboard Shortcuts	Ctrl+O : Connect to Server drop-down arrow : ODBC Connection Wizard
Other	 : Open Project : Connect to Server drop-down arrow : ODBC Connection Wizard Start Page : Manage Projects : Connect to Server drop-down arrow : ODBC Connection Wizard

Install the OLE DB Driver

The OLE DB driver will only need to be install once on each client machine, therefore if your machine already has the appropriate Oracle client (and OLE DB driver) installed you can skip to the next step.

Step	Action
1	Download the appropriate Oracle instant client for your operating system and architecture (32 or 64 bit) of Enterprise Architect from the Oracle website. Note:

	• Installing and configuring the Oracle instant client is not a trivial task, therefore it is best to involve people with previous experience performing this task. Unfortunately Sparx Systems can not provide support for another vendors software.
2	Run the Oracle Instant Client installer, making sure to include the Oracle OLE DB driver component, and perform all client configuration necessary for applications to make use of this driver.

Connect to the Repository

Step	Action
1	In the 'Open Project' dialog, select the 'ODBC Connection Wizard' option available under the Connect to Server drop down. The Data Link Properties screen is shown
2	Select **Oracle Provider for OLE DB** from the list on the Provider tab. Note: • **Do not** select 'Microsoft OLE DB Provider for

	Oracle', it will not function correctly.
3	Click on the Next>> button. The 'Connection' tab displays.
4	In the 'Data source' field, enter Oracle service name (as defined in your TNSNAMES.ORA) of your database.
5	Enter the owner credentials of the Oracle repository (ie. user name and password), check the *Allow saving password* option.
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.
8	Depending on how the connection to Oracle process was started, the 'Connection Name & Type' dialog might be displayed; if so, fill in the Name and - if required - check the 'Encrypt Connection String' option. The Name entered will be the value displayed on the 'Recent Projects' panel on the 'Open Project' dialog, while the 'Encrypt Connection string' option hides the connection details of the database, which is useful for keeping the details secret when you share

	the connection string with other users.

Notes

- Connecting to DBMS based Repositories is available in the Corporate, Unified and Ultimate Editions
- Enterprise Architect only supports the official Oracle OLE DB driver and not the 'Microsoft OLE DB Provider for Oracle' driver that comes with Windows

ODBC Connection to Oracle Repository

In order for an Enterprise Architect client to open an Oracle repository using an ODBC connection, the client machine must have the Oracle client (including the ODBC driver) installed and an ODBC DSN defined for the same architecture (32 or 64 bit) that Enterprise Architect uses.

For example, if you are using the 64 bit version of Enterprise Architect, then the client machine will need a 64 bit ODBC driver and 64 bit ODBC DSN, but if you are using the 32 bit version of Enterprise Architect then a 32 ODBC driver and DSN will be required.

Prerequisites

- An Oracle database server
- An Enterprise Architect repository defined in the Oracle database server
- You know the database user credentials that have SELECT, INSERT, UPDATE, DELETE, EXECUTE access permissions to the Enterprise Architect repository
- Oracle Client already installed on the client machine; please ensure that the OLE DB drivers are installed (you can download the drivers from the Oracle Technology Network web site, as part of the Oracle Data Access Components (ODAC) package - see *Learn more*)

Access

Keyboard Shortcuts	Ctrl+O : Connect to Server drop-down arrow : ODBC Connection Wizard
Other	 : Open Project : Connect to Server drop-down arrow : ODBC Connection Wizard Start Page : Manage Projects : Connect to Server drop-down arrow : ODBC Connection Wizard

Install the ODBC Driver

The ODBC driver will only need to be install once on each client machine, therefore if your machine already has the appropriate Oracle client (and ODBC driver) installed you can skip to the next step.

Step	Action
1	Download the appropriate Oracle instant client for your operating system and architecture (32 or 64 bit) of Enterprise Architect from the Oracle website. Note: • Installing and configuring the Oracle instant client is not a trivial task, therefore it is best to involve

	people with previous experience performing this task. Unfortunately Sparx Systems can not provide support for another vendors software.
2	Run the Oracle Instant Client installer, making sure to include the Oracle ODBC driver component, and perform all client configuration necessary for applications to make use of this driver.

Create ODBC DSN

A separate ODBC DSN must be created for each Oracle repository that the client wants to connect to.

An ODBC DSN is simply the details of the database (server and database names, Port and user credentials), stored in the registry and given a unique name that allows applications (like Enterprise Architect) to easily connect to databases.

Step	Action
1	The Windows Operating System includes 2 different versions of the ODBC administrative tool, one for 32 bit and the other for 64 bit applications. Therefore you need to open the 'Open Data Source Administrator' that corresponds to the architecture of your Enterprise Architect application. The easiest way to achieve this is to use Enterprise Architect's

	build-in ribbon option: 'Settings > User Tools > ODBC Data Sources'. 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 Oracle ODBC Driver from the list.
4	Click on the Finish button. The '*Oracle ODBC Driver Configuration*' dialog is displayed.
5	Enter these configuration details: • **Data Source Name:** (DSN), a unique name for the connection • **Description:** (optional) description of the DSN • **TNS Service Name**: The Oracle service name (as defined in the TNSNAMES.ORA) of the Oracle database • **User ID:** the database owner of repository
6	Click [Test Connection] button, you will then be prompted for the User's password, enter the Password and click [OK] 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 the Repository

Once the ODBC driver is installed and ODBC DSN defined on an individual machine the Enterprise Architect client can open the repository according to these steps:

Step	Action
1	Open the 'Data Link Properties' dialog using one of these methods: Keyboard Shortcut: Ctrl+O : Connect to Server Start Page | Server Connection (direct to 'Open Server Connection' dialog)  | Open Project : Connect to Server Start Page : Manage Projects : Connect to Server
2	On the Provider tab select 'Microsoft OLE DB Provider for ODBC Drivers', click [Next >>]
3	Select the ODBC DSN in the '*Use data source name*'

	combo, click Test Connection.
4	Once step 3 is successful, click [OK] to open the repository. If the test does not succeed, review your settings.

Notes

- Connecting to DBMS based Repositories is available in the Corporate, Unified and Ultimate Editions
- As a general rule Sparx Systems usually recommends the use of the Oracle OLE DB driver over the Oracle ODBC driver due to better performance

Create a Project in a PostgreSQL Database

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

- Stage 1: Create a database and tables
- Stage 2: Set up a connection to the database
- Stage 3: Load initial data

Prerequisites

- A machine with PostgreSQL Server v9 or higher installed and running
- psqlODBC, version 8.03 or higher has been installed (do not use version 8.3.4 or 8.4.1)

1. Create a PostgreSQL database and tables

Step	Action
1.1	Create a new empty database and configure it to your preferences, making sure to configure the correct character set and collation. The exact method you use to create the database is not important to this discussion, therefore use the most convenient or familiar method.

1.2	Open your SQL management console of choice (such as pgAdmin) and connect to the new database.
1.3	Download the **EASchema_1558_PostgreSQL.sql** file from the Sparx Systems website and open it in your SQL management console.
1.4	Run the EASchema_1558_PostgreSQL.sql to create the required database tables.

2. Set up Connection to PostgreSQL repository

From Enterprise Architect version 16 there are now two different ways that Enterprise Architect can connect to MariaDB repositories:

- via a Native connection (see Native Connection to PostgreSQL repository for details), or
- via an ADO/ODBC connection (see ODBC Connection to PostgreSQL repository for details)

Note: Sparx Systems recommend Native connections as they are much simpler and don't require any other software installed or configured.

3. Load Initial Data

Enterprise Architect repositories require a basic amount of reference data to function correctly.

You can set up this initial data in one of two different ways:

- Transfer the entire contents of an existing model into the new database, or
- Run initial data SQL Script

3.1 Load Initial Data - Transfer Existing Model

If you choose to set up your new database with the contents of an existing model then Sparx Systems recommend that a Project Integrity check is performed on the existing model to ensure it is 'clean' and free from data-related errors before being copied into the new database.

The Enterprise Architect installation provides a model called EABase.qea, which is copied into the installation folder specifically for the purpose of providing basic data for a new model. However, if you have customized the various drop-down lists in another model then you might prefer to use that. Regardless of what model you choose to use, the process is the same and is outlined here.

Step	Action
3.1.1	In Enterprise Architect, open the existing repository and select 'Settings > Model > Integrity > Project Integrity' in the ribbon.

3.1.2	Ensure that the 'Action' panel is set to 'Report Only' and that all items in 'Checks to Run' are enabled, then click on the Go button.
3.1.3	If Enterprise Architect detects that the existing model contains integrity issues then it will list them in the main panel at the bottom of the dialog. These issues should be resolved before continuing with the project transfer.
3.1.4	With Enterprise Architect open, ensure that it there is no project loaded, by selecting 'Close Project' from the  button.
3.1.5	Select the 'Settings > Model > Transfer > Full Project Transfer via Connection' ribbon option. The 'Project Transfer' dialog displays.
3.1.6	In the 'Transfer Type' panel, select 'File to DBMS'.
3.1.7	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.

3.1.8	At the right of the 'Target Project' field, click on the  button and select either: • 'Native Connection' - the 'Open Server Connection' dialog displays (see the Native Connection to PostgreSQL repository Help topic for details), or • 'ODBC Connection Wizard' - The 'Datalink Properties' dialog displays (see the ODBC Connection to PostgreSQL repository Help topic for details, in particular step *Connect to the Repository*)
3.1.9	If required, select the 'Logfile' checkbox and type a path and filename for the data transfer log file.
3.1.10	Click on the Transfer button to begin the data transfer process.
3.1.11	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. Note: • During the transfer, if an error message displays reporting '...nonstandard use of \\ in a string literal...', then the server variable (in the postgresql.conf file) will need to get set as shown and the transfer re-run:

	escape_string_warning = off

3.2 Load Initial Data - SQL Script

Step	Action
3.2.1	Open your SQL management console of choice (such as pgAdmin) and connect to the new database.
3.2.2	Download the EABase_1558_PostgreSQL.sql file from the Sparx Systems website and open it in your SQL management console.
3.2.3	Run EABase_1558_PostgreSQL.sql to load the required initial data.

Native Connection to PostgreSQL repository

In most installations users will be able to create native connections to PostgreSQL database without the need to install or configure any other software.

Prerequisites

- A PostgreSQL database server (v9 or later)
- An Enterprise Architect repository defined in the PostgreSQL database server
- You know the database user credentials that have SELECT, INSERT, UPDATE, DELETE, EXECUTE access permissions to the Enterprise Architect repository

Access

Keyboard Shortcuts	Ctrl+O : Connect to Server
Other	Start Page | Server Connection (direct to 'Open Server Connection' dialog)  | Open Project : Connect to Server Start Page : Manage Projects : Connect to Server

Connect to Repository

Step	Action
1	In Enterprise Architect, open the 'Open Server Connection' screen ('Project list | Open Project : Connect to Server' ribbon option).
2	Select the PostgreSQL option
3	Fill in the details of the database, including Server Name, Port, Database Name, Database User and Password
4	Click the [Test] button. A "Connection Successful" popup message should be displayed, including a summary of the server's version details. If the connection is not successful, the message should include a hint or reason why it failed, the reason(s) listed will need to be corrected before continuing.
5	Once step 4 succeeds the OK button will be enabled, click [OK] to open the repository.

Notes

- Connecting to DBMS based Repositories is available in the Corporate, Unified and Ultimate Editions of Enterprise Architect

ODBC Connection to PostgreSQL repository

In order for an Enterprise Architect client to open a PostgreSQL repository using an ODBC connection, the client machine must have the PostgreSQL ODBC driver installed and an ODBC DSN defined for the same architecture (32 or 64 bit) that Enterprise Architect uses.

For example, if you are using the 64 bit version of Enterprise Architect, then the client machine will need a 64 bit ODBC driver and 64 bit ODBC DSN, but if you are using the 32 bit version of Enterprise Architect then a 32 ODBC driver and DSN will be required.

Prerequisites

- A PostgreSQL database server
- An Enterprise Architect repository defined in the PostgreSQL
- You know the database user credentials that have SELECT, INSERT, UPDATE, DELETE, EXECUTE access permissions to the Enterprise Architect repository
- You know the architecture (32 or 64 bit) of your installed Enterprise Architect application

Install the ODBC Driver

Step	Action
1	Download the appropriate PostgreSQL ODBC driver (psqlODBC) for your operating system and architecture (32 or 64 bit) of Enterprise Architect from the PostgreSQL website. Note: • Not all versions of PostgreSQL ODBC driver function correctly with Enterprise Architect, see the Driver Issues page for details of the versions that cause problems.
2	Run the PostgreSQL driver installer, accepting the default values.

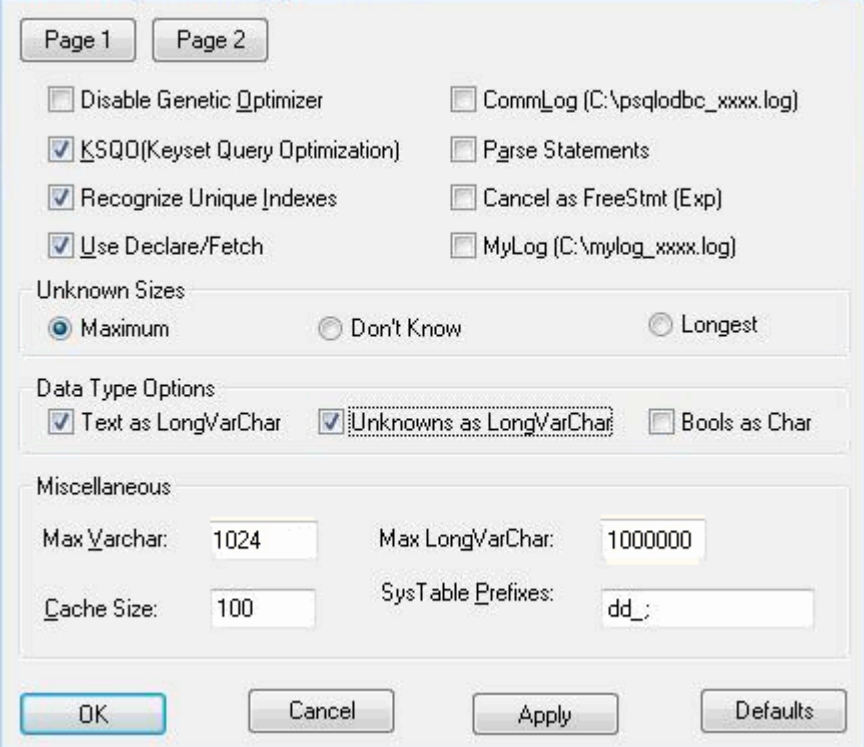
Create ODBC DSN

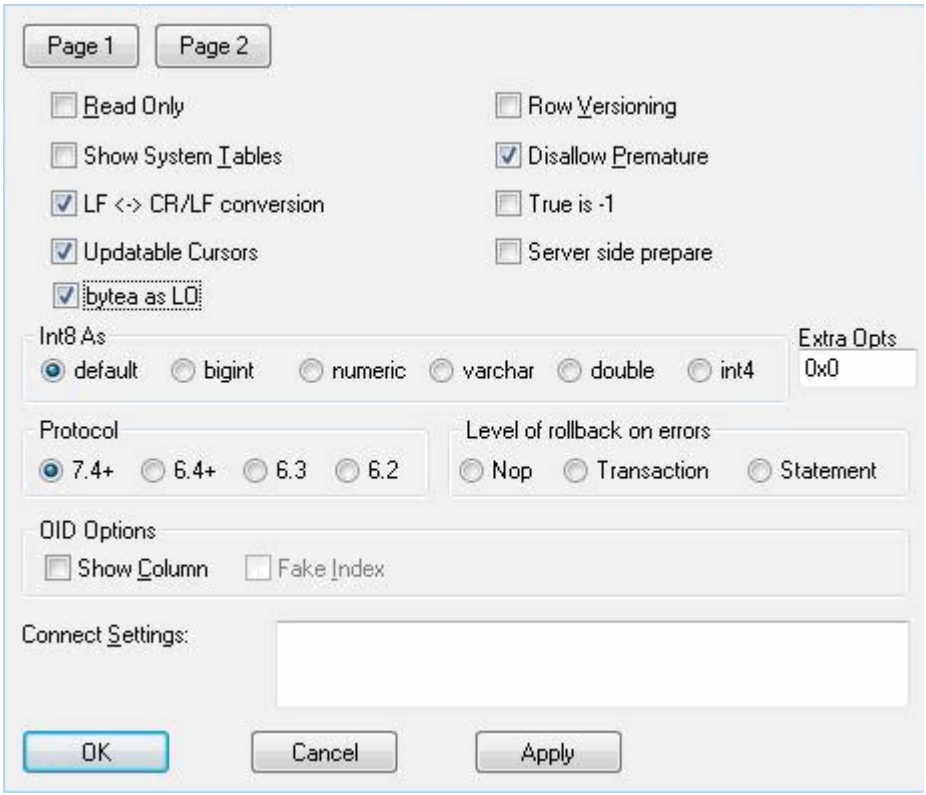
A separate ODBC DSN must be created for each PostgreSQL repository that the client wants to connect to.

A ODBC DSN is simply the details of the a database (server and database names, Port and user credentials), stored in the registry and given a unique name that allows applications (like Enterprise Architect) to easily connect to databases.

Step	Action

1	The Windows Operating System includes 2 different versions of the ODBC administrative tool, one for 32 bit and the other for 64 bit applications. Therefore you need to open the 'Open Data Source Administrator' that corresponds to the architecture of your Enterprise Architect application. The easiest way to achieve this is to use Enterprise Architect's build-in ribbon option: 'Settings > User Tools > ODBC Data Sources'. The ODBC Data Source Administrator window displays.
2	The DSN can be saved as either a User or System DSN, if you have administrator privileges on the local machine then we would suggest saving the DSN as a System DSN, otherwise your only option is to create a User DSN.
3	Click on the Add button. The 'Create New Data Source' dialog displays, enabling you to add a new DSN.
4	Select 'PostgreSQL UNICODE' from the list.
5	Click on the Finish button. The 'Postgre SQL Connector/ODBC' dialog displays.
6	Enter these configuration details:

	• **Data Source:** a unique name for the connection • **Description:** (optional) description of the DSN • **Database:** The database name on the selected server • **Server**: The host name or IP address of the MariaDB server • **Port:** The Port that the PostgreSQL server is listening to • **User** and **Password:** database user credentials
7	To set the advanced options, click on the Datasource button and set the options on 'Page 1' as shown:  Note: From version 9.5.4 onwards some options shown in this image are no longer available and can be ignored.

8	If you are using a version of PostgreSQL between 8 and 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 should be selected Note: From version 9.5.4 onwards the two options are no longer available and can be ignored.
9	Click on the OK button to close the Advanced Options screen.
10	Click Test and if successful, click on the OK button to complete the configuration. If the test does not succeed, review your settings.

Connect to the Repository

Once the ODBC driver is installed and ODBC DSN defined on an individual machine, the Enterprise Architect client can open the repository using these steps:

Step	Action
1	Open the 'Data Link Properties' dialog using one of these methods: • Keyboard Shortcut: Ctrl+O : Connect to Server • Start Page | Server Connection (direct to 'Open Server Connection' dialog) •  | Open Project : Connect to Server • Start Page : Manage Projects : Connect to Server
2	On the Provider tab select 'Microsoft OLE DB Provider for ODBC Drivers', click [Next >>]
3	Select the ODBC DSN in the 'Use data source name' combo, click Test Connection.
4	Once step 3 is successful, click [OK] to open the repository. If the test does not succeed, review your settings.

Notes

- Connecting to DBMS based Repositories is available in the Enterprise Architect Corporate, Unified and Ultimate Editions

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 a database and tables
- Stage 2: Set up a connection to the database
- Stage 3: Set up initial model data

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 (see the *SQL Server Security Permissions* Help topic)
- ALTER permission on the t_image table in Enterprise Architect, to be able to import model images

1. Create a SQL Server database and tables

Step	Action
1.1	Create a new empty database and configure it to your preferences, making sure to configure the

	correct character set and collation. The exact method you use to create the database is not important to this discussion, therefore use the most convenient or familiar method. As an example Microsoft's SQL Server Management Studio could be used: • In SQL Server Management Studio, locate the server on which to create your new database (such as DBSERVER02\SQLEXPRESS) and login with administrator privileges. • On the Databases folder, right-click and choose the 'New Database' option. • Enter a suitable name for the database and set any options as required, including the character set and collation. • Click OK to create the database
1.2	If not already open your SQL management console of choice (such as Microsoft SQL Server Management Studio) and connect to the new database.
1.3	Download the **EASchema_1558_SQLServer.sql** file from the Sparx Systems website and open it in the SQL management console.
1.4	Run EASchema_1558_SQLServer.sql to create the required database schema.

1.5	If your Enterprise Architect user base are all using v14 or greater then we would suggest applying the optional schema update that removes the need for users to have an elevated level of permissions in the database when importing images. To achieve this: Download the **EASchema_1558_SQLServer_Update1.sql** file from the Sparx Systems website and open it in the SQL management console and run it.

2. Set up Connection to the SQL Server repository

From Enterprise Architect version 16 there are now two different ways that Enterprise Architect can connect to SQL Server repositories:

- via a Native connection (see Native Connection to SQL Server Repository for details), or
- via an ADO/OLEDB connection (see OLE DB Connection to SQL Server Repository for details)

Note: Sparx Systems recommend Native connections as they are much simpler and don't require any other software installed or configured.

3. Set up Initial Data

Enterprise Architect repositories require a basic amount of

reference data to function correctly.

You can set up this initial data in one of two different ways:

- Transfer the entire contents of an existing model into the new database, or
- Load initial data using a SQL Script

3.1 Load Initial Data - Transfer existing model

If you choose to set up your new database with the contents of an existing model then Sparx Systems recommend that a Project Integrity check is performed on the existing model to ensure it is 'clean' and free from data-related errors before being copied into the new database.

The Enterprise Architect installation provides a model called EABase.qea, which is copied into the installation folder specifically for the purpose of providing basic data for a new model. However, if you have customized the various drop-down lists in another model then you might prefer to use that. Regardless of what model you choose to use, the process is the same and is outlined here.

Step	Action
3.1.1	In Enterprise Architect, open the existing repository and select 'Settings > Model > Integrity > Project Integrity' in the ribbon.

3.1.2	Ensure that the 'Action' panel is set to 'Report Only' and that all items in 'Checks to Run' are enabled, then click on the Go button.
3.1.3	If Enterprise Architect detects that the existing model contains integrity issues, they will be listed in the main panel at the bottom of the dialog. These issues should be resolved before continuing with the project transfer.
3.1.4	With Enterprise Architect open, ensure that there is no project loaded, by selecting 'Close Project' from the  button.
3.1.5	Select the 'Settings > Model > Transfer > Full Project Transfer via Connection' ribbon option. The 'Project Transfer' dialog displays.
3.1.6	In the 'Transfer Type' panel, select 'File to DBMS'.
3.1.7	In the 'Source Project' field, click on the  button and browse for and select the name of the project file to copy to the repository. If the .QEA file has Replication enabled, this must be removed before performing the transfer.
3.	At the right of the 'Target Project' field, click on the

1.8	 button and select either: • 'Native Connection' - the 'Open Server Connection' dialog displays (see the Native Connection to SQL Server Repository Help topic for details), or • 'ODBC Connection Wizard' - The 'Datalink Properties' dialog displays (see the OLE DB Connection to SQL Server Repository Help topic for details, in particular step Connect to the Repository)
3.1.9	If required, select the 'Logfile' checkbox and type a path and filename for the data transfer log file.
3.1.10	Click on the Transfer button to begin the data transfer process.
3.1.11	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.

3.2 Load Initial Data - SQL Script

The Sparx Systems website provides a SQL Script that contains a series of INSERT statements that will load an empty database with all the data needed by Enterprise

Architect.

Step	Action
3.2.1	Open your SQL management console of choice (such as Microsoft SQL Server Management Studio), and connect to the new database.
3.2.2	Download the EABase_1558_SQLServer.sql file from the Sparx Systems website and open it in your SQL management console.
3.2.3	Run EABase_1558_SQLServer.sql to load the required initial data.

Create a SQL Server Repository

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

There are many ways to create a new SQL Server database. A common method is to use an SQL-based Management Console such as Microsoft SQL Server Management Studio, which provides a convenient graphical user interface to create databases, execute scripts, and perform backups and restores. However, the exact method used is not important to this discussion, and you can use the most convenient or familiar method at your disposal.

Having created your SQL Server database, use the SQL script provided by Sparx Systems to create the required table definitions for Enterprise Architect. The latest EASchema_XXXX_SQLServer.sql script (where XXXX represents the latest schema version) is freely available on the *DBMS Based Repositories* page on the Sparx Systems website.

SQL Server Management Studio example

Step	Action
1	In SQL Server Management Studio, locate the server on which to create your new repository (such as

	DBSERVER02\SQLEXPRESS) and login with administrator privileges.
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 EASchema_1220_SQLServer.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

- 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

Native Connection to SQL Server Repository

In most installations users will be able to create native connections to SQL Server databases without the need to install or configure any other software.

Prerequisites

- A SQL Server database server
- An Enterprise Architect repository defined in the SQL Server database server
- You know the database user credentials that have SELECT, INSERT, UPDATE, DELETE, EXECUTE and SHOW VIEW access permissions to the Enterprise Architect repository

Access

Keyboard Shortcuts	Ctrl+O : Connect to Server
Other	Start Page | Server Connection (direct to 'Open Server Connection' dialog)  | Open Project : Connect to Server Start Page : Manage Projects : Connect to

	Server

Connect to Repository

Step	Action
1	In Enterprise Architect, open the 'Open Server Connection' screen ('Project list | Open Project : Connect to Server' ribbon option).
2	Select the SQL Server option
3	Fill in the details of the database, including Server Name, Port, Database Name and optionally Database User and Password. Note: to use Windows Authentication (or Integrated Authentication) simply leave both the User and Password fields empty. Entering a User and Password will imply that the connection will use SQL Server Authentication.
4	Enterprise Architect Native connection to SQL Server has the ability to make use of either the OLEDB or ODBC drivers, the option you choose will depend on the drivers installed on your machine The *Use strong encryption for data* and *Trust server*

	certificate extended options are only visible for the OLEDB driver type and relate to how the underlying database driver will communication to the database server.
5	Click the [Test] button. A "Connection Successful" popup message should be displayed, including a summary of the server's version details. If the connection is not successful, the message should include a hint or reason why it failed, the reason(s) listed will need to be corrected before continuing. Note: Minimum requirements: • **Windows 7** SP1 • **KB3140245** (for TLS defaults) • **KB4474419** (for SHA-2 support) • **KB4490628** (for SHA-2 support)
6	Once step 5 succeeds the OK button will be enabled, click [OK] to open the repository.

Notes

- Connecting to DBMS based Repositories is available in the Corporate, Unified and Ultimate Editions of Enterprise Architect

- Depending

OLE DB Connection to SQL Server Repository

In order for an Enterprise Architect client to open a SQL Server repository using an OLE DB connection, the client machine must have the SQL Server OLE DB driver installed for the same architecture (32 or 64 bit) that Enterprise Architect uses.

For example, if you are using the 64 bit version of Enterprise Architect, then the client machine will need a 64 bit OLE DB driver, but if you are using the 32 bit version of Enterprise Architect then a 32 OLE DB driver will be required.

Prerequisites

- A Microsoft SQL Server database server
- An Enterprise Architect repository defined in the SQL Server database server
- You know the database user credentials that have SELECT, INSERT, UPDATE, DELETE, EXECUTE access permissions to the Enterprise Architect repository; see the *SQL Server Security Permissions* Help topic
- 'Microsoft OLE DB Provider for SQL Server' or 'Microsoft OLE DB Driver for SQL Server' v18.x should be installed

Access

Keyboard Shortcuts	Ctrl+O : Connect to Server drop-down arrow : ODBC Connection Wizard
Other	 | Open Project : Connect to Server drop-down arrow : ODBC Connection Wizard Start Page : Manage Projects : Connect to Server drop-down arrow : ODBC Connection Wizard

Install the OLEDB Driver

In the majority of modern Windows Operating Systems there is a 'Microsoft OLE DB Provider for SQL Server' driver installed as part of the Operating System, however this driver is quite old and can not connect to more recent versions of SQL Server database server, therefore a manual install of a newer driver is required.

The OLEDB driver will only need to be install once on each client machine, therefore if your machine already has the appropriate SQL Server OLEDB driver installed you can skip to the next step.

Ste	Action

p	
1	Download the appropriate SQL Server OLEDB driver v18.x for your operating system and architecture (32 or 64 bit) of Enterprise Architect, from the Microsoft website. If you are interested, the history of the SQL Server OLEDB driver is discussed on this webpage.
2	Run the SQL Server OLEDB driver installer, accepting the default values.

Connect to the Repository

Step	Action
1	In the 'Open Project' dialog, select the 'ODBC Connection Wizard' option available under the *Connect to Server* drop down. The Data Link Properties screen is shown
2	Select 'Microsoft OLE DB Provider for SQL Server' or 'Microsoft OLE DB Driver for SQL Server' from the list on the Provider tab. Note:

	• 'Microsoft OLE DB Provider for SQL Server' is usually part of the Windows Operating System, otherwise it can be installed as part of MDAC 2.8 • 'Microsoft OLE DB Driver for SQL Server' is a newly released driver that adds support for TLS v1.2 and other more recent features of Microsoft SQL Server (such as multi-subnet failover capabilities); this driver is available from the Microsoft website
3	Click on the Next>> button. The 'Connection' tab displays.
4	Type in the connection details, including Server Name, User Name, Password and check the *Allow saving password* option.
5	From the *Select the database on the server* drop down choose the database name to connect to.
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.
8	Depending on how the connecting to SQL Server process was started, the 'Connection Name & Type' dialog could be displayed; if so fill in the Name and,

	if required, check the 'Encrypt Connection String' option. The Name entered will be the value displayed on the 'Recent Projects' panel on the 'Open Project' dialog, while the Encrypt Connection string option hides the connection details of the database, which is useful for keeping the details secret when you share the connection string with other users.

Notes

- Connecting to DBMS based Repositories is available in the Corporate, Unified and Ultimate Editions

Install the ODBC Driver

The ODBC driver will only need to be install once on each client machine, therefore if your machine already has the appropriate MySQL ODBC driver installed you can skip to the next step.

Step	Action
1	Download the appropriate MySQL ODBC driver for your operating system and architecture (32 or 64 bit) of Enterprise Architect from the MySQL website. Note:

	• Not all versions of MySQL ODBC driver function correctly with Enterprise Architect, our experience has determined that versions between 5.2.4 & 5.3.4 and 8.0.16 onwards have been the most stable (see more details in the notes section below). • Sparx Systems recommends the MySQL ODBC driver (instead of the MariaDB driver) because it has better performance.
2	Run the MySQL ODBC driver installer, accepting the default values.

SQL Server Security Permissions

The security model implemented by Microsoft's SQL Server is tight 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.

Minimum Permissions

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 these database roles to each user:

- db_datareader
- db_datawriter

Additional Permissions for Diagram Images

When an Enterprise Architect repository has not been defined with the optional schema update (EASchema_XXXX_SQLServer_Update1.sql) database users will require additional permissions in order to be able to perform Project Transfers and XML imports that include diagram images.

Why is this the case? In order for Enterprise Architect to be able to store unique images in each repository the value used as the key in T_IMAGE records needs to be set; however, the original schema definition of the T_IMAGE table did not allow this and as a result Enterprise Architect needed to execute SET IDENTITY_INSERT T_IMAGE {ON | OFF} commands, which requires higher permissions; that is, the role of 'db_ddladmin'.

To avoid the need for each user to have an elevated level of permissions, an optional schema update (EASchema_XXXX_SQLServer_Update1.sql) was developed that unfortunately is not backward compatible with earlier versions of Enterprise Architect. Therefore the optional schema update (EASchema_XXXX_SQLServer_Update1.sql) should only be applied if all users are using Enterprise Architect 14 or later. See https://sparxsystems.com/resources/repositories/index.html

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.

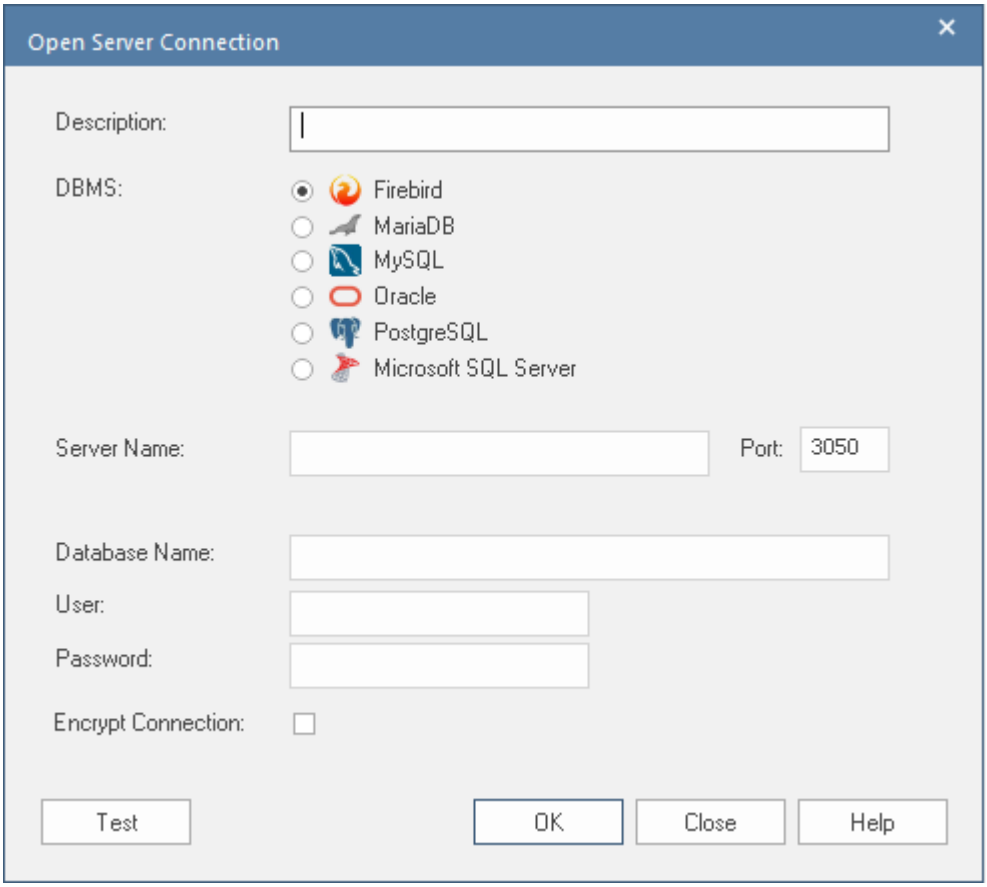
Open Server Connections - Native Connection

The '*Open Server Connection*' dialog helps you to define a new connection to an existing server based repository. It is the simplest way, and the recommended method, to set up such a connection to repositories in:

- Firebird
- MariaDB
- MySQL
- Oracle
- PostgreSQL and
- SQLServer

Using a native connection, in most cases you do not need to install any additional drivers or perform any additional steps outside of Enterprise Architect.

For compatibility reasons the original methods of defining ODBC and OLEDB database connections on the Microsoft 'Data Link Properties' dialog are still available and discussed in their own topics (*ODBC Connection to <dbms name> repository* and *OLE DB Connection to <dbms name> repository*); however with the introduction of a 64 bit version of Enterprise Architect, user machines would most likely require additional ODBC/OLE DB drivers (and/or ODBC DSNs) to continue to function.



Access

Other	Start Page > Open Project | Server Connection Start Page : Manage Projects : Connect to Server (or, on drop down arrow, | Native Connection)  | Open Project : Connect to Server

Define the Connection

Description	Type in the name by which the connection is known.
DBMS	Click on the radio button for the type of DBMS you are connecting to. Depending on which one you select, additional notes and fields might appear on the dialog. For a database on SQLServer that has an ODBC connection, you are prompted to select whether to use that or the OLE DB connection; if there is no SQL Server ODBC or OLE DB driver installed then the relevant radio button is grayed out. By default Enterprise Architect will default to OLE DB. Note: 'Microsoft SQL Server option in Linux/Wine is currently not supported
Server Name	Type in the name or ID of the Server being used to support the database. If you are connecting to: • A Firebird database on the local server, you do not need to complete this field; provide the name if the file is on a remote server • An Oracle database, you can enter either the server name or the name

	defined in TNSNAMES.ORA • A SQL Server database, enter the server and instance name
Port	Type in the number of the Port through which the connection will be made.
Database Name	Type in the name of the database to connect to.
User	In the majority of cases access to a database is limited to authorized users only, enter the user name/ID of the database user.
Password	In the majority of cases access to a database is limited to authorized users only, enter the password of the database user.
Encrypt Connection	If required, select this checkbox to encrypt and hide the connection details of the database from the users who the connection string is given to.
Test	Click on this button to check that a connection can be made to the server-based database, as specified by the entered details.

	Note: that this button is only enabled once all the mandatory fields have a value. If the test does not succeed, revise your settings. If the test succeeds, click on the OK button.
OK	Click on this button to close the dialog and open the repository. Note: this button is only enabled when a test has been successfully completed. By opening the repository an entry will be automatically added to the 'Recent' list on the 'Start Page' and *Recent Projects* list in the *Manage Projects* screen.
Close	Click on this button to cancel any changes on the dialog, and close it.

Connect to a Repository via ODBC

Server based connection via ODBC (and OLE DB) has been supported in Enterprise Architect for many years, and with the release of Enterprise Architect v16 this continues; however, this is no longer the default server based connection since they are complex to define and can involve steps being performed outside Enterprise Architect.

Enterprise Architect supports ODBC based connections from these DBMSs:

- MariaDB
- MySQL
- PostgreSQL
- Oracle

Prerequisites

- A DBMS database server
- A database defined in the database server with the necessary table definitions
- You have the database user credentials that have the correct access permissions, such as SELECT, UPDATE, INSERT, EXECUTE, SHOW VIEW and DELETE
- The appropriate ODBC driver is installed on the local machine

Access

Keyboard Shortcuts	Ctrl+O
Other	 : Open Project Start Page : Manage Projects

Connect to a Data Repository Using an ODBC Driver

Step	Action
1	In the 'Open Project' dialog, click on the Connect to Server button drop-down arrow and select the 'ODBC Connection Wizard' option. The 'Data Link Properties' dialog displays.
2	Select 'Microsoft OLE DB Provider for ODBC Drivers' from the list.
3	Click on the Next>> button. The 'Connection' tab displays.
4	Click on the 'Use data source name' radio button and,

	in the data field, click on the down arrow. From the list, select the ODBC driver you have set up to connect to your database repository.
5	If required, type in a user name and password. If required (for MariaDB, MySQL), in the 'Enter initial catalog to use' field, click on the down arrow and provide an initial catalog name.
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. If appropriate (Oracle) the database prompts you for the password - type in the password.
8	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 hide the connection details from anyone reviewing the connection string, select the 'Encrypt Connection String' checkbox. This option is most useful when sharing a connection string with other users but you don't want them knowing the

	exact details.
11	Click on the OK button to complete the configuration and open the repository. This also adds the Name (defined at step 9) to the 'Recent' list on the Start Page; from now on, you can open the project again just by clicking on this Name.

Notes

- Connecting to DBMS based Repositories is available in the Corporate, Unified and Ultimate Editions of Enterprise Architect

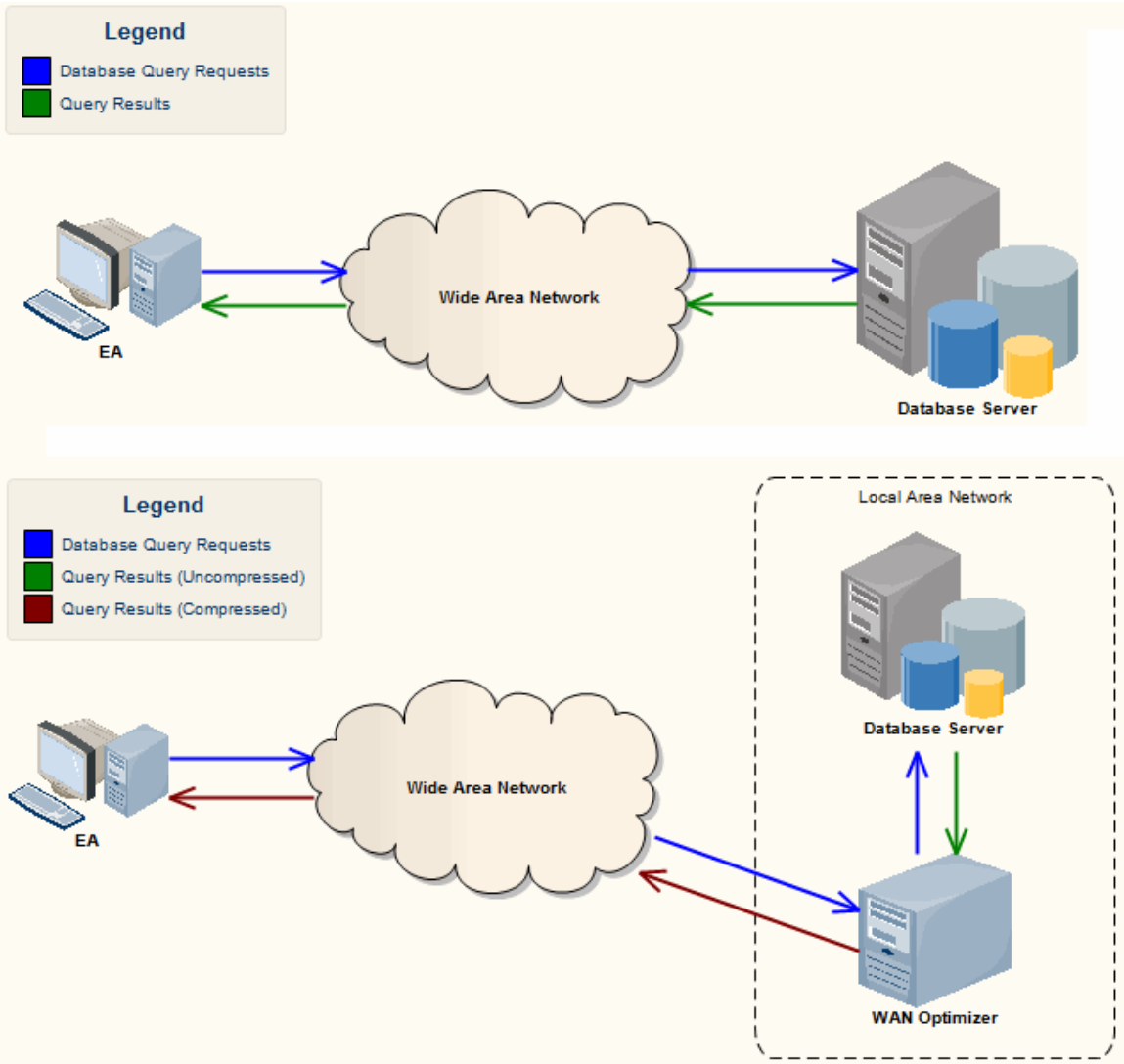
The WAN Optimizer

*"**February 2014,** With the release of Enterprise Architect version 11, the functionality provided by the WAN Optimizer has been replaced by the new product 'Cloud Services', as a result no further development will be made to the WAN Optimizer. The information in this topic remains here for legacy purposes only."*

*"**July 2019,** With the release of Pro Cloud Server v4, the functionality provided by the Cloud Services has been replaced by the unlicensed edition of the Pro Cloud Server, as a result no further development will be made to Cloud Services."*

You can significantly improve Enterprise Architect's performance in a Wide Area Network (WAN) by reducing the amount of data transmitted and 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:

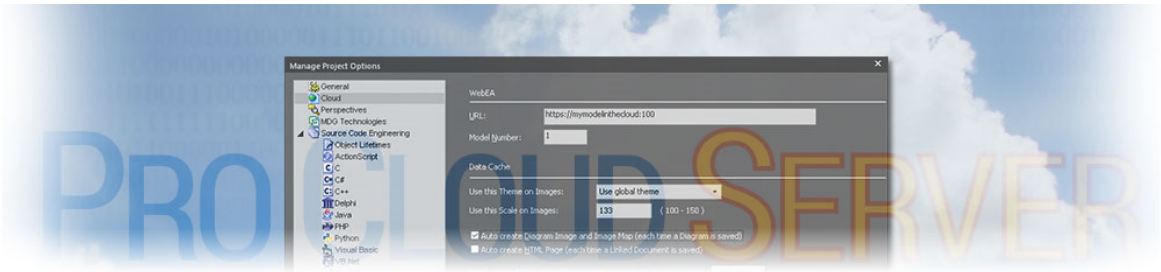
- WAN Optimizer Components
- Installing and Starting the WAN Optimizer Service
- Configuring the Service
- Troubleshooting

Notes

- The WAN optimizer was the first generation product
- The Sparx Systems Cloud Services was the second generation and replaced the WAN optimizer
- The Sparx Systems Pro Cloud Server is the third generation and has replaced the Sparx Systems Cloud Services
- All Cloud products provided similar basic functionality of the WAN Optimizer, but the later generations 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 the Pro Cloud Server should now be used instead of both the WAN Optimizer and Cloud Services products.

Pro Cloud Server Repositories



In terms of Sparx Systems and Enterprise Architect, Cloud-based repositories are repositories that are hosted by the Sparx Systems Pro Cloud Server. The Pro Cloud Server is a supporting product for Enterprise Architect that allows Enterprise Architect repositories to be hosted anywhere in the world, encapsulating the retrieval of all repository data and serving it to Enterprise Architect clients for consumption.

Sparx Systems have designed the Pro Cloud Server to be installed in your company's infrastructure, thereby giving you full control over its configuration and - even more importantly - the data contained in your repositories. We use the term 'Cloud' to indicate that Enterprise Architect users can simply connect to a repository with a single URL without any further configuration or setup.

Notes

- Cloud Repositories are supported in Enterprise Architect Corporate Edition and above
- Enterprise Architect Professional Edition also supports Cloud Repositories, although only those that utilize a

Firebird database

- An unlicensed installation of the Pro Cloud Server provides basic Cloud Repository functionality, allowing you to host Cloud repositories and access them via Enterprise Architect; with a valid license, the Pro Cloud Server provides a range of additional features

Cloud Repositories Overview

The Sparx Systems Pro Cloud Server solution provides seamless access to Enterprise Architect models in repositories held anywhere in the world, providing high performance in 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 of a wealth of development scenarios.

This section will:

- Familiarize you with the concepts of Cloud Repositories
- Discuss considerations of when and where you should use a Pro Cloud Server
- Walk you through the process of setting up a Pro Cloud Server
- Walk you through the process of connecting for the first time
- Highlight some of the additional functionality available through licensing the Pro Cloud Server
- Outline working with Active Directory using IIS

Benefits of Cloud Repositories

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 Pro Cloud Server, the procedure for a user to connect to a model is dramatically simplified. Users connecting to models hosted on a Pro Cloud Server only need a single URL; no special drivers or further configuration are required.

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

1. Improved performance for models used for distributed development. The Pro Cloud Server provides benefits to connections that have 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 Pro Cloud Server can be run from inside the corporate firewall. All model connections are created using HTTP/S, allowing firewalls to completely isolate your database server.

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

Pro Cloud Server Features

Even without a License, the Pro Cloud Server can be used for hosting Cloud Repositories to be accessed by Enterprise Architect, providing all the benefits mentioned previously; however, using a valid Pro Cloud Server license provides a number of additional benefits:

- A web based application, WebEA, that allows users to view and discuss elements in Enterprise Architect models through a web browser, while on portable devices or remote desktop systems
- OSLC Restful API, providing an interface for external applications
- Integration Plug-ins, allowing data from external providers to be displayed within Enterprise Architect

- Prolaborate, a Team Sharing and Collaboration software package
- The Sparx Systems Floating License Server
- Visibility Levels for SQL Server 2016+ and Oracle based databases, so that it is possible to restrict parts of the model to individual users or groups of users

Sparx Systems Pro Cloud licenses are available in three different editions:

- Token Edition is a scalable edition of the Pro Cloud Server; 'Tokens' are allocated to enable specific features of the Pro Cloud Server and they also determine the number of concurrent web users - there are no restrictions on the number of Tokens, you only purchase the number that suits your organization's needs
- Team Server Edition supports up to seven repositories and unlimited web users
- Enterprise Server Edition supports any number of repositories, and unlimited web users

Webinar

Before you go on to set up the Pro Cloud Server, Sparx Systems recommend that you watch the 'Setting Up Pro Cloud Server for Enterprise Architect' webinar.

You can also watch a second webinar that 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.

Pro Cloud Server Setup

This is an overview of the typical process of setting up the Pro Cloud Server.

Typical steps for setting up a Cloud Server

Step	Description
1	Install the Pro Cloud Server.
2	Review the default settings in the 'SSProCloud.config' file; in most environments these will be sufficient. However, if necessary change the SERVER_PORT to suit your environment and restart the Pro Cloud Server service.
3	Open the Configuration Client. By default a shortcut to this client should be created on the Desktop.
4	Configure the Port(s) to be used for accessing your Cloud-based models. If changes were made, restart the Pro Cloud Server service. Note, if you intend to use the Floating License Server component then you will need a Port configured to use HTTPS.

5	Define the connections to your repositories or models.
6	Configure your Firewall.
7	If you intend to use floating licenses for Enterprise Architect, configure the Floating License Server and add your Enterprise Architect keys.
8	Test access to your Cloud-based model(s) from Enterprise Architect.
9	If you are planning to use any 'Pro' features such as WebEA or Integration Plug-ins, your Pro Cloud Server will have to be 'licensed'. Refer to the *Pro Cloud Server Licensing* topic for details.
10	Optionally, if you are using the Token edition of the Pro Cloud Server then consider adjusting the token allocations (for example, enabling specific integration Plug-ins).

Pro Cloud Server Setup for Linux using Wine

Enterprise Architect's Pro Cloud Server can be installed and run on Linux using Wine. In this case pay particular attention to the points regarding privileged ports on Linux (see: Port Configuration) and the commands for running the

Pro Cloud Server in standalone mode (see: Using Multiple Configurations - Standalone mode).

Pro Cloud Server Installation

The basic installation process is quite straightforward. Typically, you can simply download and run the installer on your server, then follow the prompts.

Download the Installer

The Pro Cloud Server installer can be downloaded from the 'Registered User' section of the Sparx Systems website.

Installation Steps

If you already have the 'Sparx Systems WAN optimizer' or 'Sparx Systems Cloud Services' installed on your server we recommend you manually remove them before commencing the Pro Cloud Server Installation.

To install, follow these steps:

1. Run the *ssprocloudserver.msi* installer.
2. On the 'Welcome' page, click on the Next button.
3. Read the End User License Agreement, and accept the terms outlined in it, and click on the Next button.
4. The 'Readme' information page contains the minimum system requirements, so make sure that the host server meets those outlined requirements, and then click on the Next button.
5. The 'Custom Setup' page allows you to choose what components of the Pro Cloud Server are installed; for the

simplest installation (that is, using the defaults), go straight to step 7.

6. If you want to install specific components, for example, to have different components on separate machines, refer to the *Pro Cloud Server Components* table later in this topic, then return here to step 7.
7. The 'Ready to Install' page is in the last step before the installation begins; when ready, click on the Install button.
8. The installation progress will be shown and, once finished, the wizard should state that the process has completed; click on the Finish button.
9. Now refer to the Pro Cloud Server Configuration Help topic

To assist you in performing and understanding the installation process, watch the Sparx Systems webinar *Setting Up Pro Cloud Server for Enterprise Architect.* Click on this link to access the webinar.

Pro Cloud Server Components

The 'Custom Setup' page lists the various components that are available in the installer. Each component can be controlled separately by clicking the small disk image in front of the component name and selecting the option you require.

Component	Description
Pro Cloud	This component is the Windows service

Server service	that will accept connections from Enterprise Architect, the configuration client, and the OSLC RESTful API. For Pro Cloud Server, this includes the Sparx Systems Floating License Server. The default option is to install this component. Generally, the only time you would choose not to install this component is when you have already installed it on one machine and are now installing other components on a different machine.
Management Clients	When the Management Client component is selected the Pro Cloud Server Configuration Client application will be installed, which supports the execution of management tasks including: • Setting which databases a Pro Cloud Server can connect to • Configuring those database connections • Configuring what HTTP/HTTPS ports the Pro Cloud Server will listen on • Configuring various server options The default option is to install this component; Sparx Systems recommends that the Configuration Client should be installed whenever the Pro Cloud Server

	service is installed.
WebConfig	This component installs the application files (PHP, CSS and JavaScript) for the Web Base Configuration Client (**WebConfig)**. The default option is to install this component; you can unselect this component if you do not plan to use the Web Based Configuration Client.
WebEA	This component 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.
Integration Plug-ins	Integration Plug-ins, or the Server Based Plug-in Interface (SBPI); this component installs the SBPI Server and all the SBPI Plug-ins that allow the Pro Cloud Server to read data from external products such as Dropbox, Jira and Azure DevOps.

	There is a separate SBPI Plug-in for each external product. The default option is to install this component; you can unselect the component if you do not plan to link your Enterprise Architect models to external products.
Visibility Levels	Visibility Levels; this component installs a set of SQL scripts to be applied to any Enterprise Architect repository based on MS SQL Server 2016+ or Oracle. The default option is to install this component; you can unselect the 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.
SBPI Examples	An optional component, that installs examples of Custom Integration Plug-ins. The default location for these files is within the 'SBPI Examples' folder. For

	example: C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\SBPI Examples See the Writing a Custom Integration Plugin topic for more information. The default option is to not install this component; if you plan to create Custom Integration Plug-ins then we recommend installing these examples.

Installing New Versions

If you are installing a newer version of the Pro Cloud Server product then the new installer will automatically uninstall the old version. Before installing a new version, typically it is better to manually stop the 'Sparx System Professional Cloud' Windows service because the installer might have problems performing this task and will require the machine to be rebooted after the installation is finished.

Note: The installer has been designed to retain all configuration settings of a previous installation (except Firewall rules, see *Firewall Notes* at the end of this topic)

Installation Steps

1. Run the installer using the earlier installation steps.
2. The installer will have copied the newest version of WebEA to the location specified during installation; if this location is not being used by your web server, you need to manually copy (and overwrite) all WebEA files (except the includes/webea_config.ini file) from the installation folder to the web server folder. See *WebEA Consideration*.
3. As above, if you use the WebConfig client you should copy the contents of the WebConfig folder (specified during installation) to the appropriate location on your Web Server, overwriting the existing WebConfig files (except settings.php).

Pro Cloud Server v5 Considerations

With the release Pro Cloud Server v5 users now have the choice between 32 and 64 bit architecture, earlier versions were all 32 bit. While the majority of functionality and behavior is identical between the two architecture versions of Pro Cloud Server, there are number of minor differences that will particularly effect existing users of the 32 bit making the transition to 64 bit. These include:

- installation folder
- registry hive

Installation Folder - 32 bit uses the default installation folder of: *C:\Program Files (x86)\Sparx Systems\Pro Cloud Server* while the 64 bit uses: *C:\Program Files\Sparx Systems\Pro Cloud Server* . This difference in installation folders will effect the settings that have been saved to the configuration files, since the new installation will install new versions of these files. This can be solved by copying the older configuration files to the new installation folder and restarting the Pro Cloud Server. The configuration files that should be copied are:

- .\Service\ssprocloud.config
- .\Service\ssflsgroups.config
- .\Service\sskeys.dat
- .\Service\server.pem
- .\SBPI*.config

Registry - The Pro Cloud Server uses the registry to store a number of its settings, which include: Database Managers and Integration Plugins (partially). This means that if you have an existing PCS v4 (or earlier) version installed and you choose to install the 64 bit version of PCS v5 then none of your Database Managers or Integration Plugins will be defined in the PCS v5 version, so steps will need to be taken to migrate them over.

Database Managers

In 32 bit Pro Cloud Server, the Database Managers are stored in the registry at: *HKEY_CURRENT_USER\Software\Sparx Systems\SQLBridge\DBManagers* while in 64 bit they are now at: *HKEY_CURRENT_USER\Software\Sparx Systems\PCS\DBManagers*

Therefore while it would be possible to manually export the registry contents from the old hive and import it into the new one, an easier solution is to make use of Pro Cloud Server's Import/Export configuration functionality. For more details see the *Extended Server Properties* Help topic.

Integration Plugins

The Integration Plugins store their settings in both the registry and configuration files within the installation folder, therefore to transition existing Integration Plugin definitions from a 32 to a 64 bit installation both the .\SBPI*.config files and everything in HKEY_CURRENT_USER\Software\Sparx Systems\SQLBridge\SBPI will need to be copied transferred.

Pro Cloud Server v4 Considerations

The release of Pro Cloud Server version 4 made changes to the default installation folder and the main configuration filename. These changes require special considerations depending on your situation and whether you want to retain any of your previous configuration.

Pro Cloud Server 4's default installation path has been changed to:

'C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\'

The configuration filename has been changed to:

'SSProCloud.config'

Installation folder name change

Firebird Database Managers

If you have any Firebird-based Database Managers defined, their connection strings will contain the path of the old installation folder, e.g. 'C:\Program Files (x86)\Sparx Systems\Cloud Services\Service\Models\'; work through these steps:

1. Make a note of each Firebird-based Database Manager, the database name, Alias and extended settings.
2. Delete each of the Firebird-based Database Managers.
3. Shut down the Pro Cloud Server Windows service.
4. Physically move all Firebird database files from the old

installation folder to the new installation folder 'C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service\Models\'.

5. Start the Pro Cloud Server Windows service.
6. Recreate each of the Firebird Database Managers.

Integration Plug-ins

If your environment has any Integration Plug-ins defined and they were created prior to PCS4.1 then the physical path of the SBPI executables will be defined in the definitions; each path will need to be updated to the new installation path. PCS4.1 and later has removed the need to define the complete path to the SBPI executables.

Configuration File

If you wish to retain your previous configuration from version 2 or 3 of the Pro Cloud Server into version 4 or later, you must follow these steps:

1. Shut down the Pro Cloud Server.
2. Back up the default PCS4 configuration file by renaming C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service\SSProCloud.config' to C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service\SSProCloud-default.config'.
3. The old configuration file 'C:\Program Files (x86)\Cloud Services\Service\SSCloudServices.config' will have to be manually copied into the new installation folder

'C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service\' using the new name SSProCloud.config.

4. Start the Pro Cloud Server Windows Service.

WebEA Considerations

In regard to WebEA, consider the situation in which WebEA was installed to the default location and your website is being served by Apache in C:\Xampp\htdocs\webea. This command will perform the required copy and not override your existing configuration file in the live site.

For Pro Cloud Server 32 bit:

robocopy "C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\WebEA" "C:\xampp\htdocs\webea" /MIR /e /DCOPY:T /njh /ndl /nfl /xf webea_config*.ini

For Pro Cloud Server 64 bit:

robocopy "C:\Program Files\Sparx Systems\Pro Cloud Server\WebEA" "C:\xampp\htdocs\webea" /MIR /e /DCOPY:T /njh /ndl /nfl /xf webea_config*.ini

Firewall Notes

Firewall Note: The installer creates a Firewall rule for the Pro Cloud Server. This rule is removed and re-created during an upgrade. If you had previously modified this rule then you will need to apply these changes again after the Pro Cloud Server installation.

Note: To avoid the need to manually adjust the standard firewall rule after each reinstall, you can create a complete new firewall rule using a different name.

Migrating from 32 bit to 64 bit

With the release of Pro Cloud Server v5 users now have the choice between 32 and 64 bit architecture; prior versions (v4.2 and earlier) were all 32 bit. Additional steps are required for existing 32 bit environments wanting to migrate to 64 bit; see these steps:

Migrating from Pro Cloud Server 32 bit to 64 bit

Step	Description
1	Before uninstalling Pro Cloud Server 32 bit, open the the Pro Cloud Server Configuration Client. Select 'Extended Properties', then use the 'Export Config' option to export your configuration to a .pcscfg file. See the 'Extended Server Properties' topic for details.
2	Browse to your Pro Cloud Server 32 bit installation folder (e.g. "C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\").
3	If you have a Pro Cloud Server license make a copy of the license certificate/s and allocations file: .\Service\Licenses*.crt

	.\Service\Licenses\SSAllocations.config
4	Make copies of these configuration files: .\Service\ssprocloud.config .\Service\ssflsgroups.config .\Service\sskeys.dat .\Service\server.pem .\SBPI*.config
5	Make a copy/backup of the 'Models' folder (ie .\Service\Models*). This contains any Firebird (.feap/.fdb) or SQLite (.qea) repositories you have created.
6	Uninstall Pro Cloud Server 32 bit.
7	Install Pro Cloud Server 64 bit as described in the 'Pro Cloud Server Installation' topic.
8	Copy the license and configuration files from the previous installation (mentioned in steps 3 and 4) into the corresponding subfolders within the new installation path (e.g. "C:\Program Files\Sparx Systems\Pro Cloud Server\").
9	Copy the Firebird and SQLite repositories (feap/fdb/qea files) from step 5 to the 'Models' folder in the new installation path.

10	Restart the 'Sparx Systems Professional Cloud' service.
11	Open the Pro Cloud Server configuration client.
12	Import your Pro Cloud Server configuration file, which was exported in step 1. See the 'Extended Server Properties' topic for details.
13	Again, restart the 'Sparx Systems Professional Cloud' service.
14	If you have database managers which use a native database connection (recognizable by the 'ssdb:' prefix) then these do not require any modification.
15	For any Firebird models which were added via the ODBC connection option (recognizable by the 'firebird:' prefix), you will need to create a new model connection. Add a new database manager as described in the 'Add Database Manager - Native' topic. In the filename field enter the name (including extension) which matches the file in the 'Models' folder (mentioned in step 7).
16	For other ODBC model connections (to MySQL, SQL Server, Oracle, etc) you have two options: • (Recommended) Create a new native connection

	to your existing database as described in the 'Add Database Manager - Native' topic. Note, you should ensure the 'alias' of the new connection matches that of the old one, however it is not possible to define two connections with the same alias. Before creating the new connection, modify the alias of the old one (e.g. append '_odbc'). • Or, install the 64 bit version of the ODBC database driver and define a 64 bit ODBC DSN connection to the database. If the name of this DSN connection matches the old 32 bit ODBC DNS then the Pro Cloud Server can make use of the existing database manager (it is not necessary to add a new one).
17	For any newly created database managers update their configuration as required. See the 'Configure Database Manager' topic for details. You can then remove any of the old/unused database managers.
18	Check the Configuration Client's 'Integrations' tab. If you have defined any data providers of type 'Custom Integration', 'Custom Service' or 'Translator' then these will need to be updated. Select the data provider then click Edit. On the 'Edit Data Provider' screen, update the DLL Path so that it refers to the new installation path. Check the bindings for each of your integrations. If

	Database Managers had been removed/added during the migration then you will need to re-enable the bindings. After updating integrations you will also need restart the 'Sparx Systems Professional Cloud' service.
19	You have now fully migrated to Pro Cloud Server 64 bit.

Pro Cloud Server Configuration

This topic discusses the configuration of the Pro Cloud Server. The main settings of the Pro Cloud Server are stored in the configuration file .\Services\SSProCloud.config which will be created and populated with the default values during the initial installation process. By design, subsequent installations/updates of the Pro Cloud Server will not change the contents of this file, so your settings will not be lost.

The Pro Cloud Server Configuration Clients (both Application and Web) are designed to manage all Pro Cloud Server settings, except SERVER_PORT which defines the Port on which the Pro Cloud Server service will listen for requests from the Pro Cloud Server Configuration Client application. By default the SERVER_PORT is set to 1803 (since PCS v5.0, but earlier versions used Port 803), we would expect this value to be acceptable for most environments; however, if this does not suit or you simply want to use a different Port this can be manually changed by editing the SSProCloud.config file using a text editor (*) and restarting the Pro Cloud Server service (**).

(*) Depending on the current user's permissions the SSProCloud.config might have to be edited using the 'Run as administrator' option.

(**) Restarting the Pro Cloud Server after manually changing its configuration file is necessary because it only reads the file as it starts up.

Configuration Summary

Below is a brief summary of the Pro Cloud Server configuration steps. These steps are covered in more detail in the topics below.

Step	Details
1	If necessary, edit the configuration file (.\Services\SSProCloud.config) and change the SERVER_PORT to a suitable Port, then restart the Pro Cloud Server Service.
2	Use Pro Cloud Server Client (either the Windows Client or the Webconfig) to: • Define Ports • Define Database Managers (Pro Cloud Model Connections) • Add your Pro Cloud Server license See the *Pro Cloud Configuration Clients* Help topic for details.
3	Consider configuring additional Pro Cloud Server features, such as: • WebEA • SBPI Integration Plug-ins • Floating License Server

Pro Cloud Configuration Clients

With the release of version 3 of the Pro Cloud Server, Sparx Systems now offers two different Configuration Clients: the original Windows-based application and the new web-based application. Each client is capable of managing the main settings of the Sparx Systems Pro Cloud Server. We expect that most users will prefer to use the web-based application, although it does require one extra step during setup - to install the web pages into a supported web server (such as Apache or IIS).

Windows Application benefits

- No third party products required, you can connect to the Pro Cloud Server immediately after install using the default settings

Web-based Application benefits

- Can be configured to use SSL (HTTPS)
- One interface for all component settings

For details of each of the Configuration Clients, see these Help topics:

- Pro Cloud Server Configuration Client - Application
- Pro Cloud Server Configuration Client - Web

Configuration Client Connection Settings

Normally the first settings you will see in the configuration

file are to control how Pro Cloud Server Configuration Clients will connect to the server. The default values are:

SERVER_PORT=1803

ADMIN_WHITE_LIST=127.0.0.1|::1

Setting	Description
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 *Pro Cloud Server using IIS* Help topic. Note: When changing this setting, 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.
ADMIN_WHITE_LIST	ADMIN_WHITE_LIST defines a separated list of client IP addresses that are allowed to administer the Pro Cloud Server. If a client address attempts to logon to the Pro Cloud Server Configuration Client that is not specified in the White List they will be denied access.

	Default value: 127.0.0.1|::1 Supported Versions: PCS v4.1 and later This setting was added in PCS version 4.1, prior versions did not impose any restrictions on what client addresses could change the Pro Cloud Server settings. The default value is 127.0.0.1|::1, which grants access to the current machine hosting the Pro Cloud Server service, via its IPv4 address (127.0.0.1) and IPv6 address (::1).
SERVER_PASSWORD	**SERVER_PASSWORD** is the password to protect the administration functions of the server. As a security precaution this setting is encrypted and as such can only be changed by using the Configuration Client. This value should not be set or modified by directly editing the configuration file. Default value: pcsadm1n (value is stored encrypted) Supported Versions: All Note: The default value of this setting was changed in PCS version 4.1, prior

	versions did not define a default password.

General Settings

This 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_DEFAULTMAXSIMQUERIES	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** in the *Extended Server Properties* Help topic under *Pro Cloud Server Configuration Client - Application*). There is no explicit limit for this default value.

	Note: If this setting is changed via one of the Configuration Clients, then a server restart is not required.
AUDIT_TIME_PERIOD	The number of seconds between the recording activities by system logs on each database.
TEMP_DIRECTORY	The location to which to write temporary files before they are sent to clients. Generally you should not change this, as there are security implications. For example, the owning user of the Windows service will need full write permissions to this directory.
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** in the *Extended Server Properties* Help topic under *Pro Cloud Configuration Client - Application*). Note: If this setting is changed via one of the Configuration Clients, then a server restart is not required.

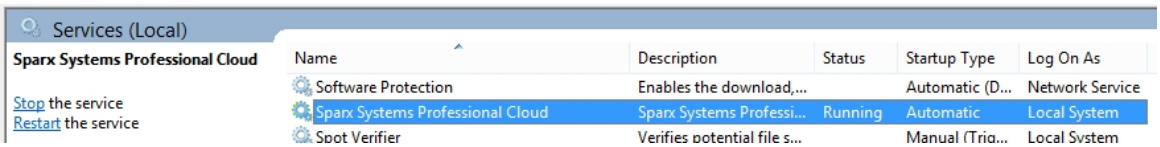
LOGGING_DIRECTORY	Defines where the log files are to be stored. The default is set to: %SERVICE_PATH%\Logs Note: The %SERVICE_PATH% refers to the directory in which the Pro Cloud Server is installed; this is, by default, 'C:\Program Files (x86)\Sparx Systems\Pro Cloud Server'.
LOGGING_FILECOUNT	Determines 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. Whenever the current log file has reached its maximum size, a new log file is created as 'SSProCloudServerLog-1.log' and all older ones have their number incremented. For more details on using the logs see the *Cloud Server Troubleshooting* topic.

Restarting the Sparx Systems Professional Cloud service

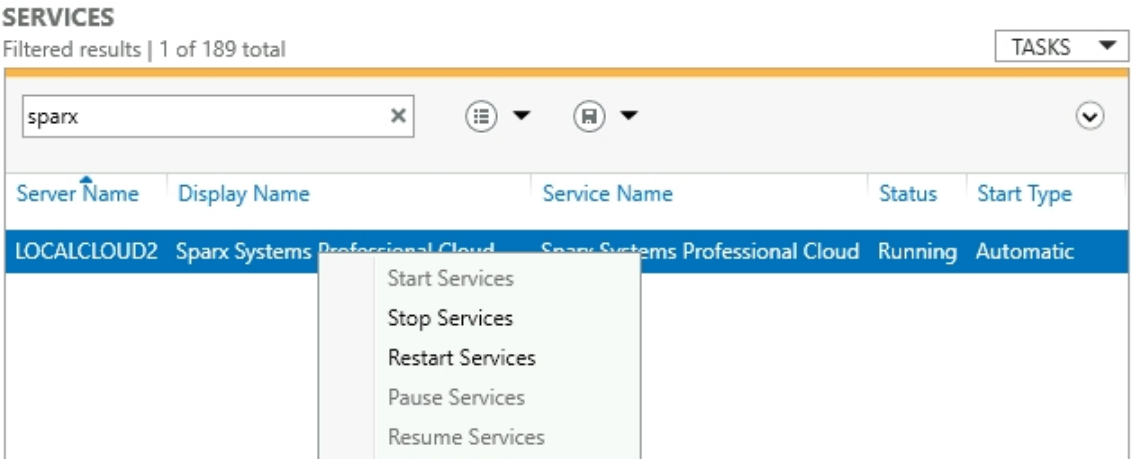
If you make any manual changes to the configuration file you must restart the service for the changes to take effect (unless otherwise stated). A service restart is carried out in the Windows Services application.

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

1) Using Windows Services directly; this is available in all versions of Windows (see 'Control Panel | Administrative Tools | Services').



2) Using the Server Manager on Windows Server 2012.



Pro Cloud Server Configuration Client - Application

At any point after installation you can configure the Pro Cloud Server using the Configuration Client application; the installer will create a **Pro Cloud Config Client** shortcut on the desktop.

By default the Configuration Client executable is accessible from these folders, depending on which architecture of Pro Cloud Server is installed, 32 or 64 bit:

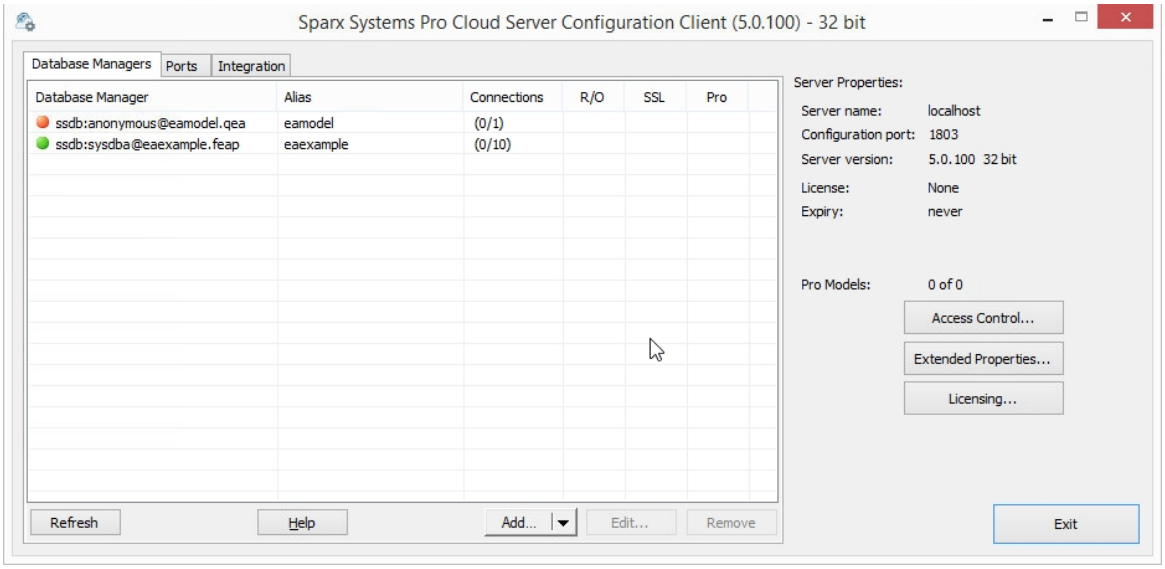
- 32 bit - 'C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Client' or
- 64 bit - 'C:\Program Files\Sparx Systems\Pro Cloud Server\Client'

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

The Configuration Client is used to configure various settings related to the Pro Cloud Server and the repositories it connects to. When the application first starts, a prompt to enter the address of the Pro Cloud Server to connect to is displayed (see the *Pro Cloud Server Logon Screen* Help topic for details). Once you have connected, the main screen displays.

The Main Screen

This screen shows a list of all defined database managers and the details of the connected Pro Cloud Server. It is also the entry point for all management functions, such as configuring your License, setting the ports the Pro Cloud Server listens on, and maintaining database connections.



Sparx Systems Pro Cloud Server supports connections to these DBMS products:

- Firebird
- Microsoft SQL Server
- MySQL
- Oracle
- PostgreSQL

Item	Description
Database Managers	This tab lists the defined repositories that the Pro Cloud Server makes available to clients. Each database manager definition contains the details of one database connection and a number of other

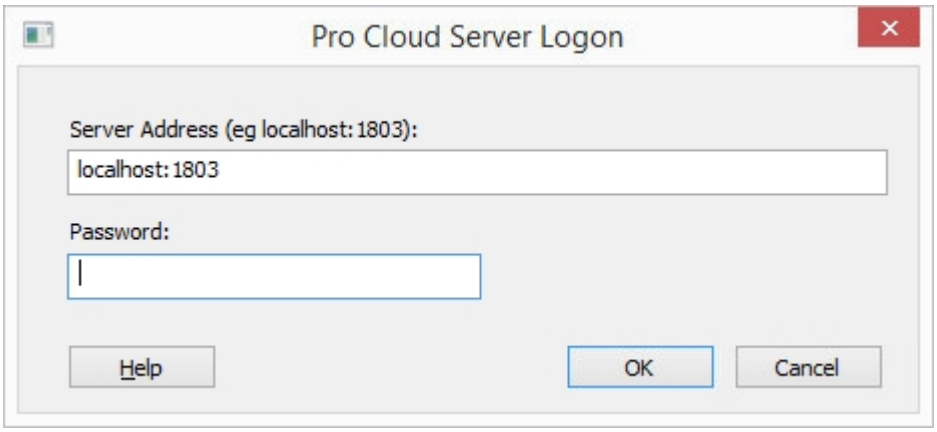
	properties that control how it will function.
Ports	This tab allows you to view and configure Ports that the Pro Cloud Server will allow connections on for accessing models.
Integration	This tab allows you to configure Cloud models to use Integration Plug-ins. Integration Plug-ins require a valid Pro Cloud Server license; see the *Integration Plug-ins* Help topic for details.
Server Properties	The properties on the right hand side specify the details of the connected Pro Cloud Server. **Main Properties** • Server name - the server name hosting the currently connected Pro Cloud Server • Configuration Port - the Port the Cloud Client is using for communication with the Pro Cloud Server • Server version - the major, minor and build numbers and architecture of the connected Pro Cloud Server • License - the edition of the Pro Cloud

	Server • Expiry - the expiry date for the current license • Licensed to - The name of the company that the Pro Cloud Server is licensed to. • Licensee - The email address associated with this license; that is, the email address provided when the License Request was created • Pro Models - the number of models enabled for Pro Features and the number allowed under the current license The Edition, Expiry and Pro Models properties apply to Pro Cloud Server licenses. If no license has been added, or the license is invalid or has expired, then the Edition will display 'None' and the Expiry 'Never'.
Extended Properties	Click on this button to display the extended properties for the connected Pro Cloud Server. A number of these extended properties can also be modified.
Licensing	Click on this button to view and manage the connected Pro Cloud Server's licenses. This button allows you to enter the license for the Pro Cloud Server.

Refresh	Click on this button to reload the contents of the 'Database Managers' list.
Add...	Click on this button to add a new database manager. If the main area of this button is clicked the default action of adding a Database Manager using the Native Connection screen is shown, however if the drop arrow area is clicked the user is presented with two options: Native Connection & ODBC Connection.
Help	Click on this button to show the relevant User Guide page that describes the current screen.
Edit...	Click on this button to view and manage the settings for the currently selected database manager.
Remove	Click on this button to permanently delete the selected database manager from the Cloud Server.

Pro Cloud Server Logon Screen

The Pro Cloud Server Logon screen is the first to be shown when launching the Configuration Client. At a minimum it requires the address and Port of the Pro Cloud Server to connect to.



Pro Cloud Server 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, MyCloudServer. If you are working on the machine with Pro Cloud Server installed then 'localhost' will be sufficient. Note: from PCS v5.0 the default Port was changed to 1803, in earlier versions it is

	was 803.
Password	Enter the password you defined in *Configuration Client Connection Settings*, in the Pro Cloud Server Configuration Help topic. As a security precaution from PCS version 4.1 onwards the default password is: pcsadm1n while PCS version 4 and earlier did not define a password. You can set the password to a new value, as described in the Help topic Extended Server Properties.
Help	Click on this button to display the specific page that describes the current screen in the User Guide.
OK	Click on this button to log in to the Cloud Configuration Client.

Database Managers

The list of Database Managers on the Pro Cloud Server Configuration Client's main screen displays all defined repositories that are accessible by Enterprise Architect clients, and the main properties for each. Database Managers can be maintained (added, modified and deleted) directly from the main screen.

The Database Managers list includes these details for each defined repository:

Columns

Option	Description
Database Manager	This column includes both a colored icon representing the enabled state (green for enabled, red for disabled) of the database manager and its connection string.
Alias	A unique name that identifies the repository. On Enterprise Architect's 'Cloud Connection' screen this is the value entered into the 'Model Name' field. An alias can be assigned when a database manager is defined; if this field is left blank the Pro Cloud Server will determine its value.

Connections	The number of active user connections, and the maximum number of connections possible. Note: Typically the number of active user connections is zero, since they are only considered 'active' while a request is being processed. As soon as the data is returned to the client the connection is set to inactive. The number of connections is only refreshed when either the Refresh button is clicked on or a change is made to a database manager's definition.
R / O	A flag to indicate if the database manager has been configured as Read-Only.
SSL	A flag to indicate if the database manager has been configured to only allow secure connections.
Pro	A flag to indicates if 'Pro features' has been enabled for the repository, this is controlled by the 'Enable Pro Features (OSLC, WebEA and Integration)' option on the Configure Database Manager screen.

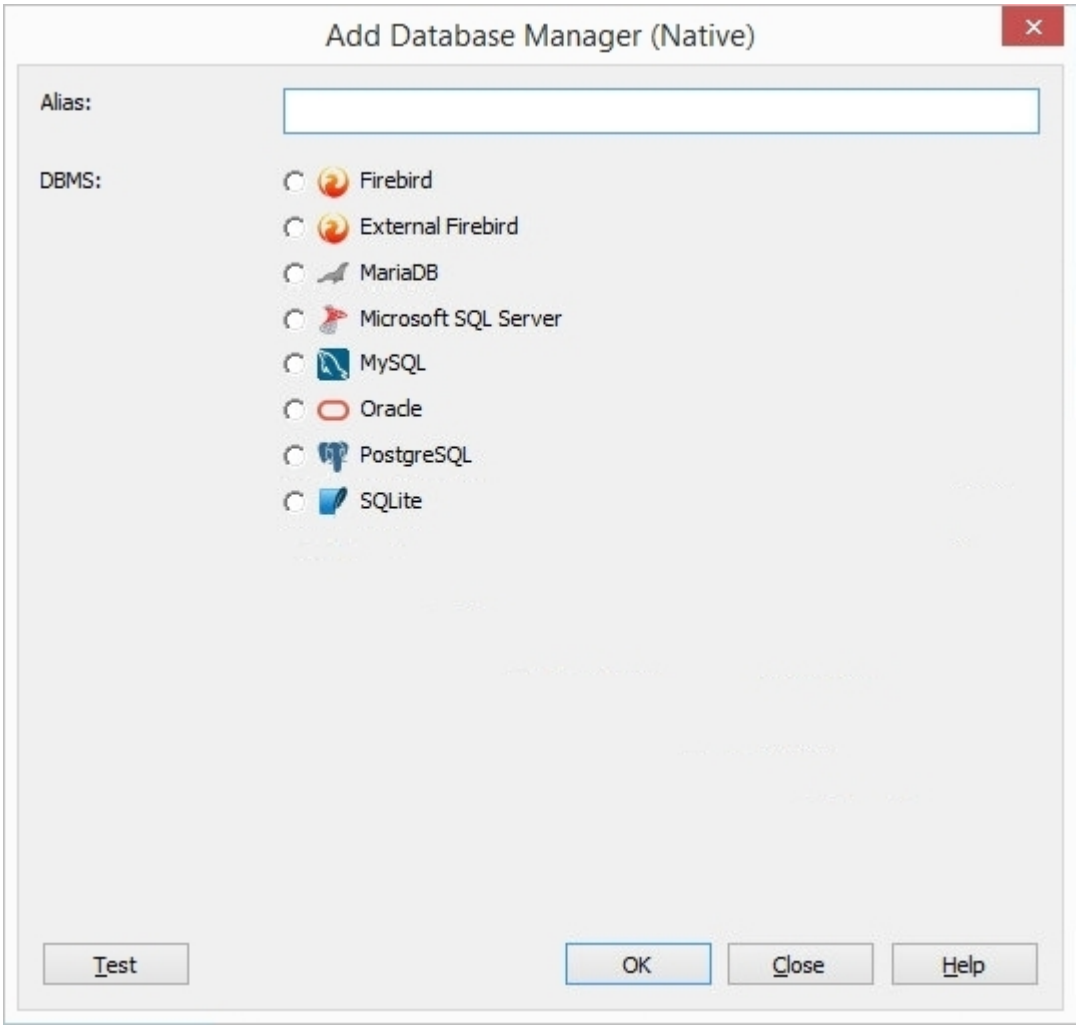
Add Database Manager - Native

The 'Add Database Manager (Native)' dialog helps you to define a new database manager to an existing server based repository. It is the simplest way, and the recommended method, to set up a connection to repositories in:

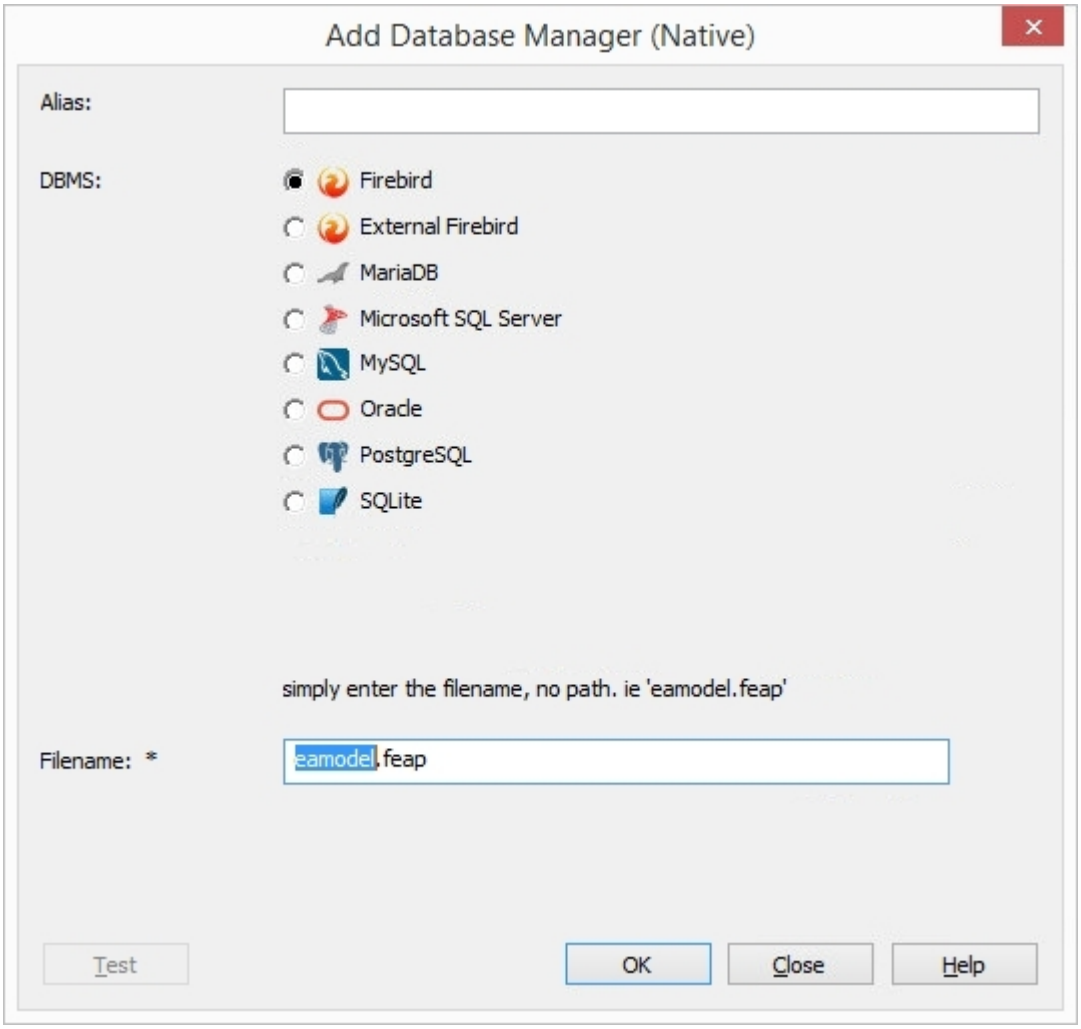
- Firebird
- MariaDB
- Microsoft SQLServer
- MySQL
- Oracle
- PostgreSQL
- SQLite

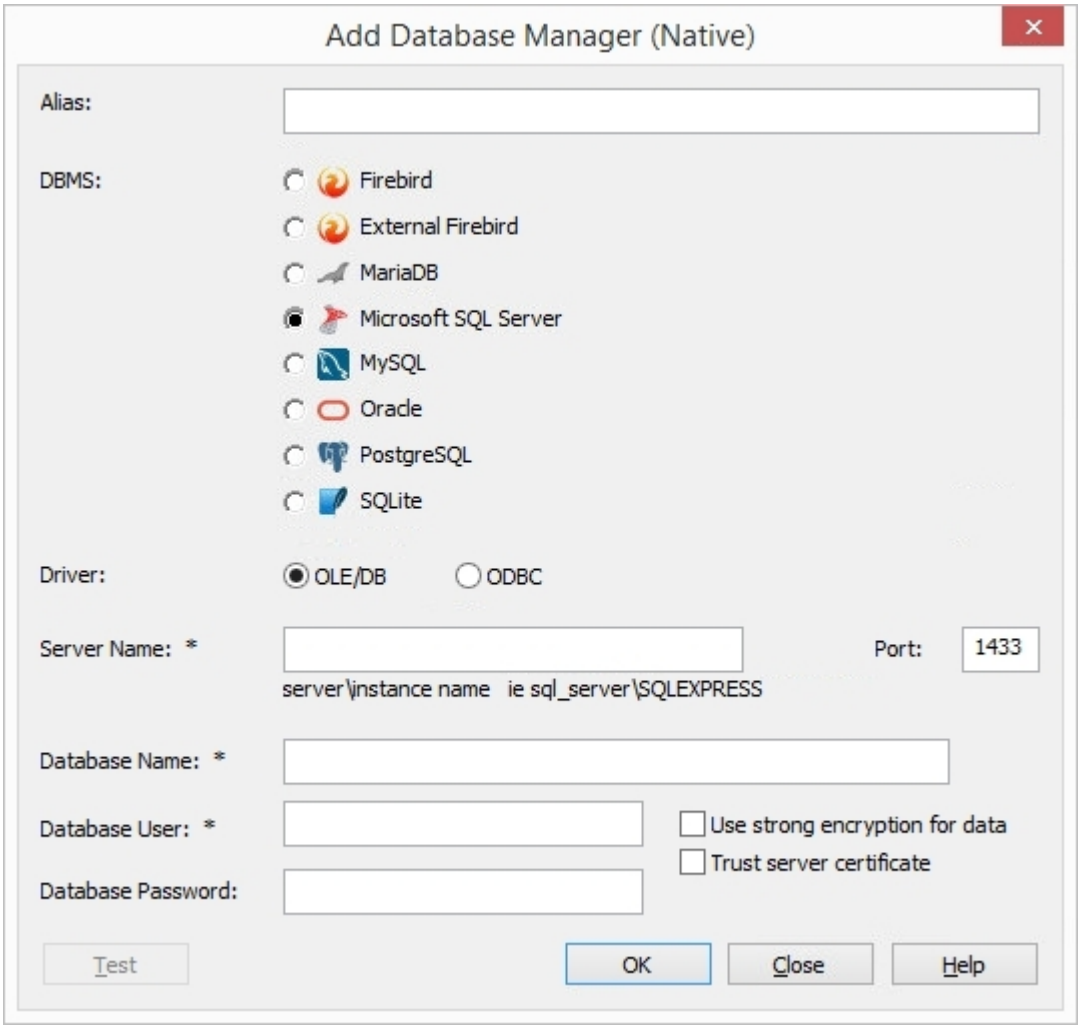
Using a native connection, in most cases you do not need to install any additional drivers or perform any additional steps outside of Enterprise Architect.

When the *Add Database Manager (Native)* screen is first opened its lower section is mostly empty:



However this area will be populated with field(s) based on the selected DBMS. In the case of local Firebird and SQLite databases the only field required is the filename, while the others typically need a Server name, Database name and user credentials. For example, the images here show a sample of the sorts of field displayed:


Add Database Manager (Native)
Alias:
DBMS:
Firebird
External Firebird
MariaDB
Microsoft SQL Server
MySQL
Oracle
PostgreSQL
SQLite
simply enter the filename, no path. ie 'eamodel.feap'
Filename: *
eamodel.feap
Test
OK
Close
Help



You will need to fill in all required fields (as denoted by the '*' character next to their label), and in most cases click on the Test button before the OK button is enabled; this is to ensure that only valid connection details are saved.

The 'Alias'

The one setting that all Database Managers have is: 'Alias'. Its purpose is to uniquely identify a database by a single value. When clients request to connect to a database they specify the 'Alias' (note that in Enterprise Architect this value is referred to as 'Model Name').

When adding a new Database Manager, the 'Alias' can be

specified manually or, if left empty, the Pro Cloud Server will assign an automatic value based on these rules:

- Firebird definitions, the 'Alias' is the filename minus the extension
- SQLite definitions, the 'Alias' is the filename minus the extension
- All other DBMSs, the 'Alias' is a the combination of server and database name (with special characters removed)

Therefore, take care when defining database connections, to ensure that you do not create duplicate database 'Aliases'. In more recent versions of the Pro Cloud Server, additional checks have been put in place to stop duplicate 'Aliases' being defined; however, older versions did not impose these limits. 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'.

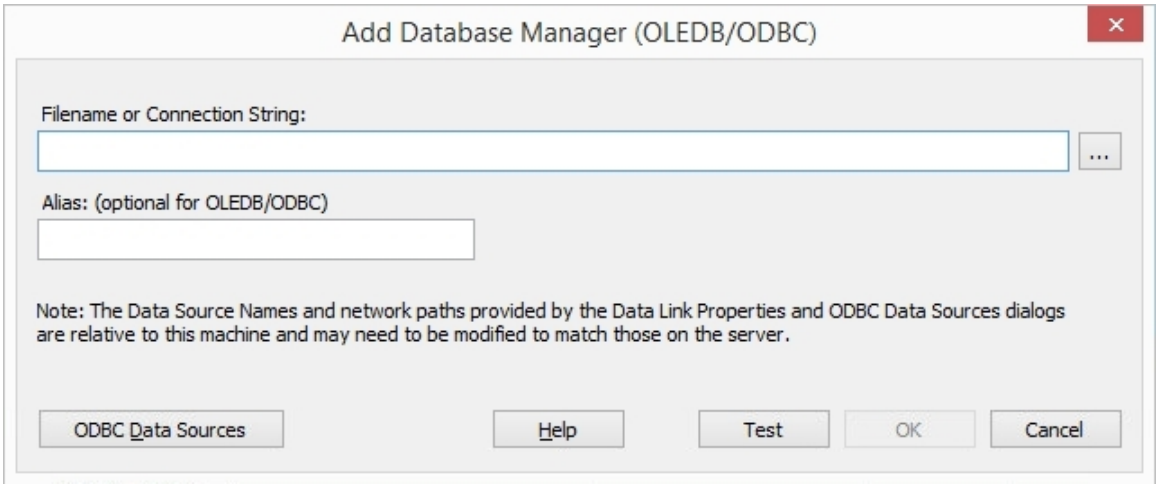
Notes

- For compatibility reasons the original method of defining ODBC and OLE/DB database connections on the Microsoft 'Data Link Properties' dialog is still available and discussed in their own topics (*Add Database Manager (OLEDB/ODBC)*), however with the introduction of a 64 bit version of the Pro Cloud Server, additional ODBC/OLE DB drivers (and/or ODBC DSNs) will most likely be required in order for the Pro Cloud Server to connect to such databases, therefore Sparx

Systems recommend Native connection due to their simplicity.

Add Database Manager - ODBC

The Add Database Manager (OLEDB/ODBC) screen displays when you click on the Add | ODBC Connection.. button on the 'Database Manager' tab of the Sparx Systems Pro Cloud Server Configuration Client screen; it allows you to define a new OLE/DB or ODBC database connection to a supported DBMS.



The process for defining a Database Manager is subtly different for each DBMS, as explained on these pages:

- Add Database Manager - Firebird
- Add Database Manager - ODBC
- Add Database Manager - SQL Server
- Add Database Manager - Oracle

The 'Alias'

It is important to note that the Pro Cloud Server uniquely identifies databases by a single value; this value is referred to as the database 'Alias'.

When adding a new database manager, the Alias can be specified using the 'Alias' field on the 'Add Database Manager (OLEDB?ODBC)' dialog.

If no Alias is specified, then a default is determined using these rules:

- 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 Oracle '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.

Relative path Note

This note is shown on the Add Database Manager (OLEDB/ODBC) 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.'

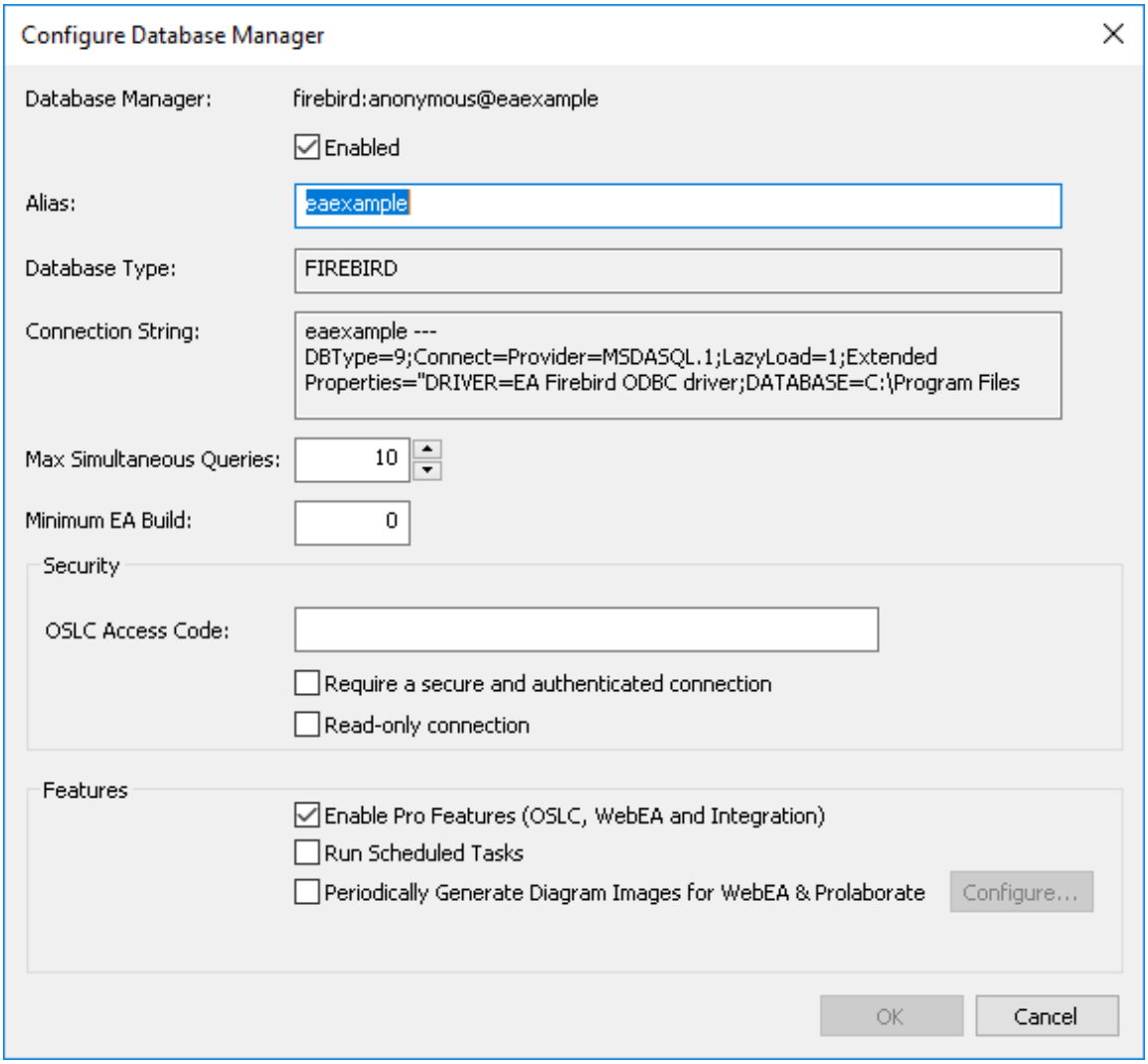
The purpose of this note is to inform the user that ODBC definitions are saved on each machine and that if the Pro

Cloud Server Configuration Client is used to access a Pro Cloud Server from another machine the ODBC DSN definition list is relative to the client machine and not the machine hosting Pro Cloud Server.

This means if the Pro Cloud Server Client application is running on a remote machine (that is, not the machine running the Pro Cloud Server) the definitions that are shown by clicking on the  button and ODBC Data Sources button are relative to the current machine and not the Pro Cloud Server machine, so care should be taken.

Default State of Added Database Managers

As a safety precaution, whenever a new Database Manager is defined it is automatically configured to be disabled. You can enable the database after it has been initially defined by selecting the 'Enabled' checkbox on the Configure Database Manager screen.

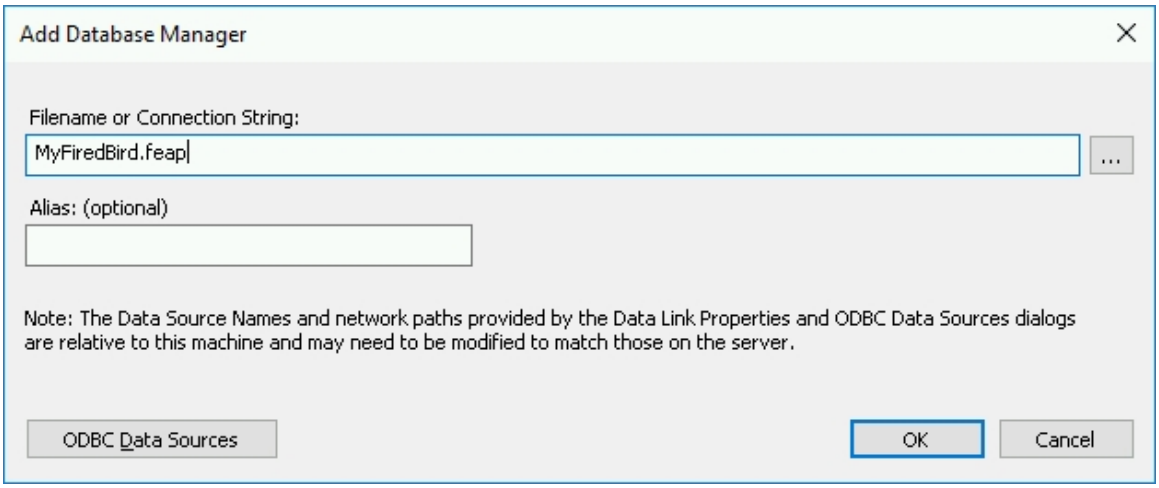

Configure Database Manager
Database Manager:
firebird:anonymous@eaexample
Enabled
Alias:
eaexample
Database Type:
FIREBIRD
Connection String:
eaexample ---
DBType=9;Connect=Provider=MSDASQL.1;LazyLoad=1;Extended
Properties="DRIVER=EA Firebird ODBC driver;DATABASE=C:\Program Files
Max Simultaneous Queries:
10
Minimum EA Build:
0
Security
OSLC Access Code:
Require a secure and authenticated connection
Read-only connection
Features
Enable Pro Features (OSLC, WebEA and Integration)
Run Scheduled Tasks
Periodically Generate Diagram Images for WebEA & Prolaborate
Configure...
OK
Cancel

Add Database Manager - Firebird

Firebird-based repositories are the simplest DBMS to add to the Pro Cloud Server and can be either completely new or an existing repository.

New Firebird Repository

To add a completely new repository, just enter the physical file name of the new model (for example, MyFireBird.feap) and click on the OK button.



The Pro Cloud Server will automatically create a new empty Enterprise Architect repository database and add it to the list of defined Database Managers on the main Pro Cloud Server Configuration Client screen.

Existing Firebird Repository

To add an existing Firebird database to your Pro Cloud Server environment, the physical file must first be copied into the 'Models' folder in the Pro Cloud Server's installation folder (the default being 'C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service\Models'). This ensures the Pro Cloud Server has the necessary security permissions

to the file.

Once the existing file is placed in the 'Models' folder, simply enter the filename of the database in the 'Add Database Manager' screen and click on the OK button. Your existing database will then be added to the list of defined Database Managers on the main Pro Cloud Server Configuration Client screen.

Background Information

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/

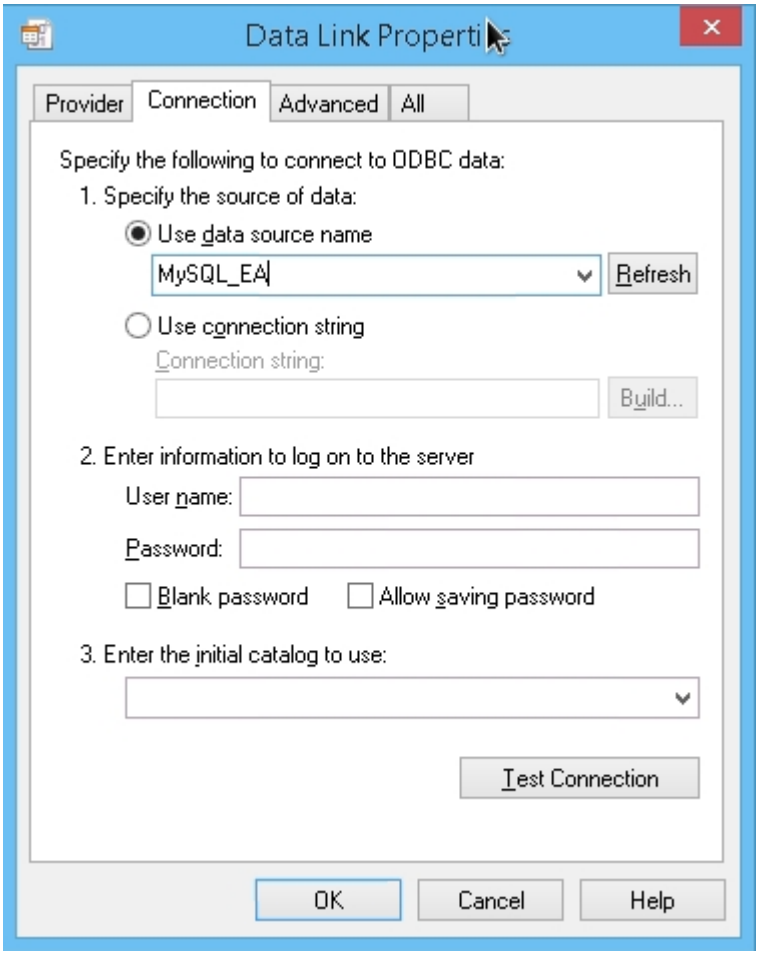
The *Introduction to Pro Cloud Server* webinar shows the use of a Firebird database. You can view the webinar by clicking on this link:

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

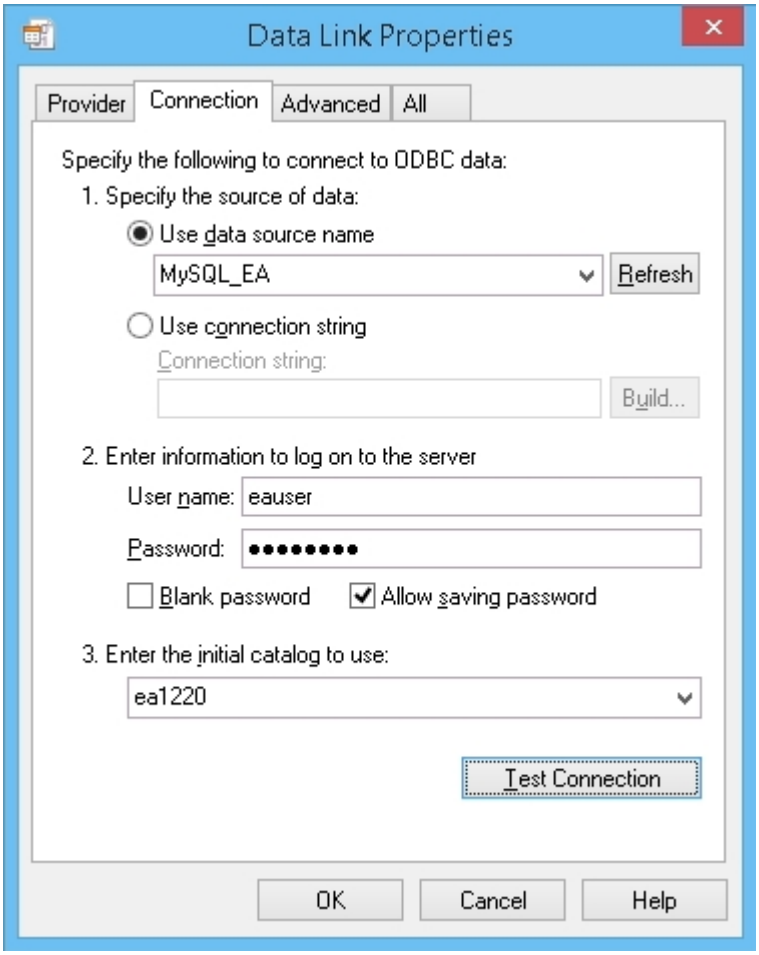
Add Database Manager - ODBC

Pro Cloud Server connections to MySQL, Oracle and PostgreSQL databases require the use of ODBC DSNs, which are defined in the Windows 'ODBC Data Source Administrator (32 bit)' screen. If you are running the Configuration 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. Only fill in the user name, password (and check the 'Allow saving password' checkbox) and initial catalog if values have not already been set within the System DSN. Some ODBC drivers don't function correctly when values are entered in both locations. Normally it is easier to save the user credentials and initial catalog within the System DSN.



When user credentials have not been saved within the System DSN, the 'Data Link Properties' dialog should be filled in like this:



Notes:

- All ODBC DSNs should be defined as SYSTEM DSNs, because the Pro Cloud Server runs as a Windows Service and User DSNs might not be available to the Pro Cloud Server
- If you choose to enter the user name and password on this screen then you must check the 'Allow saving password' option, so that all necessary information is stored within the connection string. Leaving this option uncheckbox will not results in a user prompt since the Pro Cloud Server is a Windows Service and is intended to run without user interaction
- 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 sources

- For **MySQL** - make sure to adjust the MAX_PACKET_SIZE of your MySQL installation to be large enough to store the largest record in your database; this is needed in order for Enterprise Architect to retrieve and manipulate large records
  It is difficult for Sparx Systems to state a recommendation for this setting because it is dependant on the size of records in a user's repository; however, it is not unusual that a value of at least 200M is required
  Typically the largest records are stored in T_DOCUMENT and are the results of embedding a document or storing a baseline within the repository; the MAX_PACKET_SIZE setting is defined in MySQL's configuration file, 'my.ini' for Windows or '~/.my.cnf' for Linux

Add Database Manager - SQL Server

Pro Cloud Server connections to SQL Server-based databases require the use of an OLE/DB driver; currently Sparx Systems Pro Cloud Server supports both the historical 'Microsoft OLE DB Provider for SQL Server' (which has been included with MDAC for many years) and the newly-released 'Microsoft OLE DB Driver for SQL Server' (currently only v18.x), with the latter supporting TLS v1.2 and other new features released with the later versions of SQL Server.

Authentication

The SQL Server OLE/DB drivers support two forms of database authentication, both of which are supported by Sparx Systems Pro Cloud Server:

1. Windows Authentication / Windows NT Integrated security (or SSPI) and

2. SQL Server Authentication / "specific user name and password"

Many Sparx Systems customers expect the 'Windows Authentication' option to use the authentication details of the end user logging into Enterprise Architect; however this is NOT the case. When this option is selected it is actually the credentials of the owning user of the Sparx Systems Professional Cloud Windows Service that are used to authenticate against the database. This, by default, is 'Local System'.

Credentials are needed to establish a database connection;

however, it is a relatively slow process - if the Pro Cloud Server was required to establish a connection to the database using the end user's details for every query, performance would be severely impacted. Instead, the Pro Cloud Server creates a 'pool' of database connections when it starts. Database connections from the pool are allocated to action each request for data as it arrives and, since they have already been established, execution is almost instantaneous.

Therefore, when defining database connections to SQL Server-based repositories, special consideration should be given to determine the best/most appropriate database credentials for your environment. Typically we recommend SQL Server authentication with the use of a predefined SQL Server user; however, for many customers the SQL Server authentication method goes against their company's security policy, so they need to either:

- Change the owning user of the Sparx Systems Professional Cloud Windows Service to one that has the necessary permissions in SQL Server, or
- Grant the Local System account permissions to the SQL Server database

Roles / Permissions

In order for end users of Enterprise Architect/WebEA/Prolaborate to be able to read and update information within the repository, the database users require the relevant permissions; the easiest way to achieve this is to grant these roles to each database user:

db_datareader

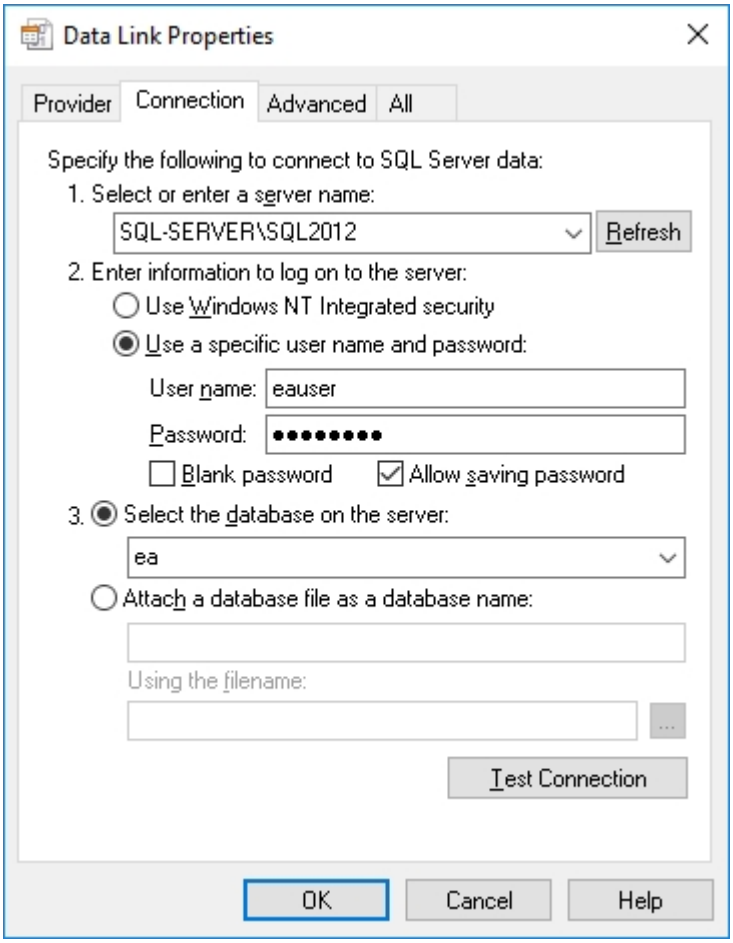
db_datawriter

Add SQL Server database using SQL Server authentication

Prior to defining the Pro Cloud Server connection, define a SQL Server user with roles to the database that will act as the Enterprise Architect repository.

You can define a connection to a SQL Server Database by specifying the connection string to the database. If you are running the Configuration 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' or 'Microsoft OLE DB Driver 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'.

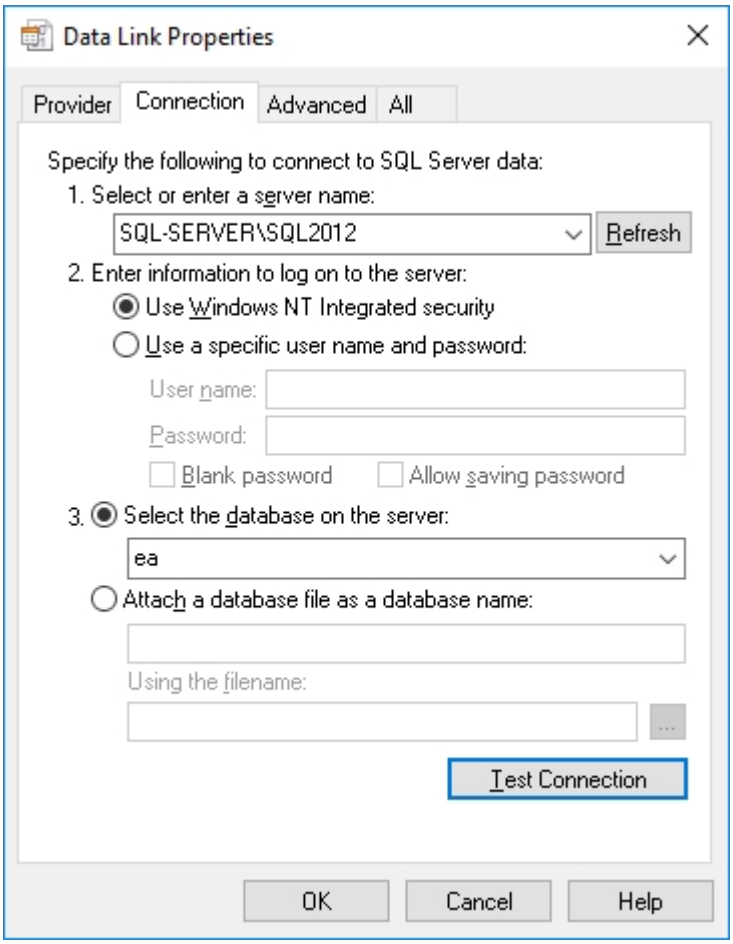


Add SQL Server database using Windows Authentication

You can define a connection to a SQL Server Database by specifying the connection string to the database. If you are running the Configuration 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' or 'Microsoft OLE DB Driver 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:

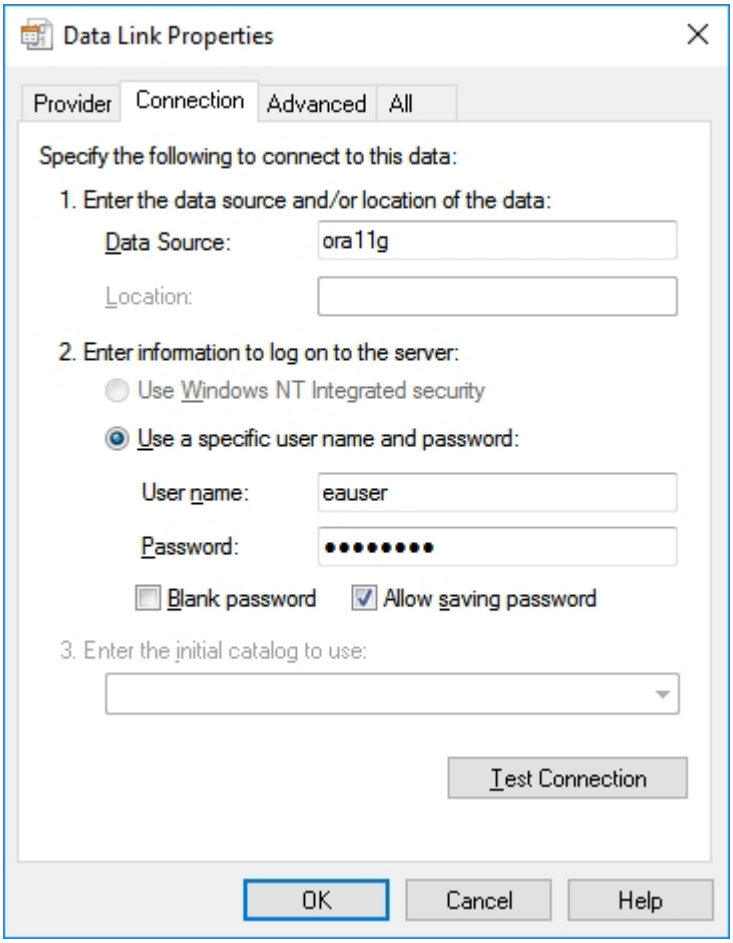
- Because the Sparx Systems Pro Cloud Server is a Windows service and runs under a single account, all database access will be performed using the credentials of the owning account, so it will not be possible to keep track of what records are updated in the database by individual end users; however, internally Enterprise Architect can store audit history if required

Add Database Manager - Oracle

Pro Cloud Server connections to Oracle-based databases require the use of either ODBC or OLE/DB drivers, although Sparx Systems recommend the use of OLE/DB since it provides superior performance compared to ODBC.

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.

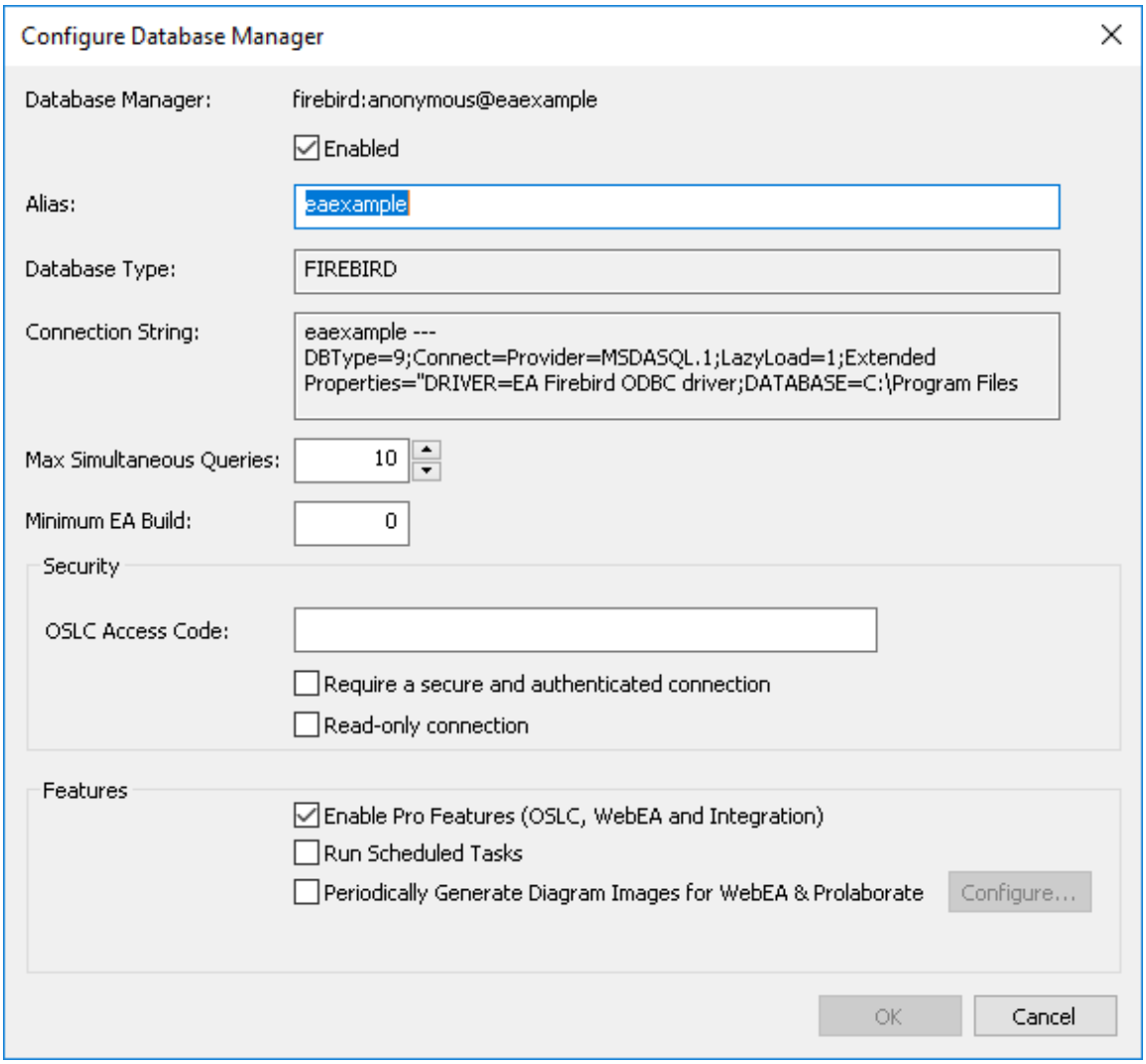


Note:

- The Pro 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

Configure Database Manager

The Configure Database Manager screen is shown when you click on the Edit button on the Sparx Systems Pro Cloud Server Configuration Client screen. It allows you to modify the various options for the selected Database Manager.



Options

Option	Description

Database Manager	Read-only field showing the description of the database as shown in the 'Database Manager' List. The format of the description quickly identifies the database, as illustrated here: {odbc provider}:{user}@{datasource}
Enabled	Select this checkbox to allow the Data Manager to respond to requests for the connection.
Alias	The database Alias. This is the value Enterprise Architect clients will need to enter in the 'Model Name' field of the 'Cloud Connection' screen. Whenever a database manager is created the system will define a value for 'Alias' automatically, but if necessary you can change this value to a more meaningful or appropriate name for your Enterprise Architect and WebEA users.
Database Type	Read-only field showing the DBMS of the defined database, such as Firebird, MySQL, Postgres, SQLSvr or Oracle.
Connection String	Read-only field showing the full connection string used to connect to the database.

Copy Button	A convenient button for copying the connection string to the clipboard.
Max Simultaneous Queries	Type in the maximum number of queries that the Pro Cloud Server can process simultaneously for the selected Database Manager. 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 queries/connections you have defined here, increase it by small increments. At regular intervals the Pro Cloud Server will write out log entries that summarizes each database connection's usage, for example: [AUDIT] sqloledb.1:ea@10.11.12.13,1433.eapos | Free: 10 | Max Active: 12 | Acquisitions:

	125 | The 'Max Active' value indicates that within the audit period there were at most 12 simultaneous queries running, therefore to ensure that queuing does not occur the max simultaneous queries setting should be larger than 12. Perhaps try 15 to start with and afterwards this can be reassessed.
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 that all clients (such as Enterprise Architect, WebEA and Prolaborate) connecting to this Database Manager use SSL authentication. Note: Before setting this option, make sure the repository has been enabled for

	User Security (see Enable Model Security); if the repository is enabled for User Security *after* setting this value, the Pro Cloud Server will not recognize the correct User Security state until after it has been restarted.
Read-only connection	Select this checkbox to prevent users from modifying the database.
Enable Pro Features (OSLC, WebEA and Integration)	This option is only available in Pro Cloud Server editions. A checked state indicates that the current model has 'Pro Features' enabled, which includes OSLC, WebEA and Integration to external systems. If this option is disabled then the reason for disabling it will be shown in the text. For example: **No License** - There is no license defined or the license has expired **Model Limit Reached** - Both the 'Token' and 'Team' editions of the Pro Cloud Server are limited to a certain number of models that can be Pro-enabled; this reason code implies the maximum number of models has already been Pro-enabled
Run	When this option is checked the Pro

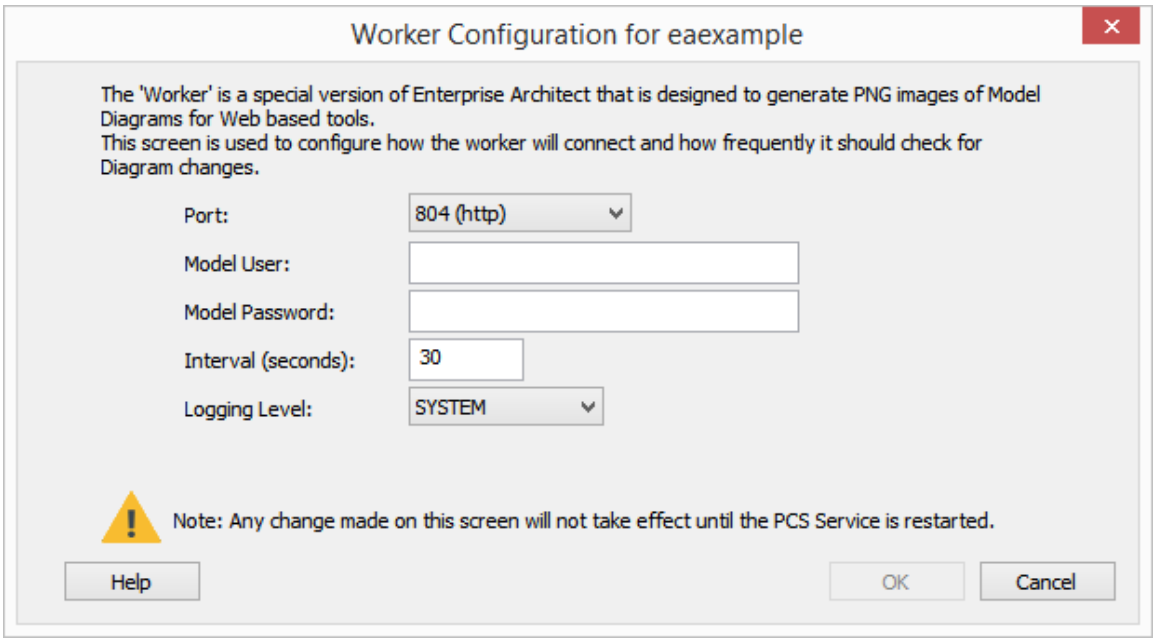
Scheduled Chart Generation	Cloud Server will update Time Series charts in the current model as a schedule task. To configure the Scheduled Chart Generation , select the checkbox, then click the Configure button. • **Start Time**: The time of day that the schedule task should run • **Number of times to retry**: In the event of repeated errors, how many times should the process retry • **Interval between retries**: The number of minutes between retries • **Next Run**: The calculated next run date and time. Note: this value only updates after the various settings has been saved.
Periodically Generate Diagram Images for WebEA & Prolaborate	This option is only available in Pro Cloud Server editions when a Database Manager has Pro Features enabled. In order to generate diagram images and linked documents (which are required by WebEA and Prolaborate) a worker can be configured to run periodically; the Pro Cloud Server is capable of automatically starting these workers whenever it starts. To configure a worker, select the

	checkbox, then click on the Configure button. See the Configure Worker Help topic for more details.
OK	Click on this button to save the settings and close the dialog.

Configure Worker

The Worker Configuration screen is shown when you click on the Configure button on the 'Configure Database Manager' screen. It allows you to easily modify the various options for running a worker instance of Enterprise Architect against the selected database manager.

In order for Web-based tools like WebEA and Prolaborate to view Enterprise Architect diagrams and linked documents, static representations must be generated and stored within the repository whenever a diagram or linked document is updated. The Pro Cloud Server includes a specialized version of Enterprise Architect called the 'EAWorker' that can generate the required information on a regular basis.



Options

Option	Description
Port	Select the desired Port that the EAWorker should use to connect to this repository. This dropdown will only list the ports that are defined within the Pro Cloud Server.
Model User	Enter a valid repository user. This value only requires a value if the current repository has model security enabled
Model Password	Enter the password for the repository user. This value only requires a value if the current repository has model security enabled
Interval	This value defines how often (in seconds) the EAWorker process will check for modified diagrams, and generate where required; the default value is 30 seconds
Logging Level	Select the desired level of logging that the EAWorker should produce. The available logging level options are the same as those for the Pro Cloud Server. The worker log files will be created within a 'Logs' folder under the worker's installation location, ie if PCS is installed into the default location this will be:

	C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service\Worker\logs .

Obtain Database Manager Connection String

Each Cloud Connection/Database Manager definition includes a connection string that the Pro Cloud Server uses to access the Repository/Database.

This connection string can be particularly helpful in diagnosing issues when connecting to the Cloud model. You can obtain the connection string either via the Pro Cloud Configuration Client Application or the Pro Cloud Configuration Web interface (WebConfig).

Cloud Configuration Client

To obtain the connection string via the Pro Cloud Configuration Client Application, follow these steps:

Step	Description
1	Open the Pro Cloud Configuration Client.
2	In the list of defined Database Managers, click on the relevant Database Manager, then click on the Edit button.
3	The Configure Database Manager screen is displayed. This includes the 'Connection String' field.

4	Click on the Copy button, which is located next to the 'Connection String' field.

WebConfig

To obtain the connection string via the Cloud Configuration Web Interface (WebConfig), follow these steps:

Step	Description
1	Log in to the WebConfig home screen.
2	Click on the relevant model in the 'Model Connections' list. Click on the Edit Connection button in the 'Action' column.
3	The 'Edit Model Connection' page is displayed. Click on the 'Advanced' heading to expand its contents.
4	Click on the Copy button, which is located next to the 'Connection String' field.

Migrating OLEDB/ODBC connections to Native

With the introduction of Native connections in Pro Cloud Server v5, Sparx Systems would suggest that all Database Managers defined with either OLE/DB or ODBC drivers be replaced with Native Connections at your earliest convenience.

At this stage the migration process is a manual process.

Prerequisites

- A DBMS database server
- A database defined in the database server with the necessary table definitions
- You know the database user credentials that have the correct access permissions, such as SELECT, UPDATE, INSERT, EXECUTE and DELETE

Native connections offer these advantages

- No need to install any 3rd party drivers (in most case)
- No need to create any ODBC DSN
- PCS connection strings contain all necessary connection details
- Native connection strings work for both 32 and 64 bit architectures

Migrating OLE/DB based connections

An OLE/DB connection string should contain all the necessary details, therefore the migration process consists of these steps:

1. Open the Sparx Systems Pro Cloud Server Configuration Client.
2. Select the OLE/DB based Database Manager within the list.
3. Click on the Edit button, to review the connection details of the Database Manager.
4. Make a note of all settings on the Configure Database Manager screen, as well as the extended ones for *Run Scheduled Chart Generation* and *Periodically Generate Diagram Images for WebEA & Prolaborate.*
5. Make a note of all database settings (as defined in the connection string). That is, 'Server Name', 'Port', 'Database Name' and 'User'. The database user password will be encrypted so it is expected that someone within your organization should know the password for the given database user.
6. Change the 'Alias' value; we suggest adding the text '(oledb)' to the end.
7. Close the Configure Database Manager screen.
8. Click the Add button on the 'Database Managers' tab, this should display the Add Database Manager (Native) screen.

9. Enter the Alias value to be that of the original Database Manager.
10. Enter the database details that were noted in steps 5 and 6.
11. Click on the Test button to confirm that a connection can be made.
12. Click on the OK button to save the connection.
13. Edit the Database Manager you created earlier and adjust all the settings to match those of the original Database Manager.
14. Test that your clients can access the repository (using the original 'Alias').
15. Once everything is confirmed to be working as expected, the renamed database manager can be deleted.

Migrating ODBC based connections

An OLE/DB connection string should contain all the necessary details, therefore the migration process consist of these steps:

1. Open the Sparx Systems Pro Cloud Server Configuration Client.
2. Select the ODBC based Database Manager within the list.
3. Click on the Edit button, to review the connection details of the Database Manager.
4. Make a note of all settings on the Configure Database Manager screen, as well as the extended ones for 'Run

Scheduled Chart Generation' and 'Periodically Generate Diagram Images for WebEA & Prolaborate'.

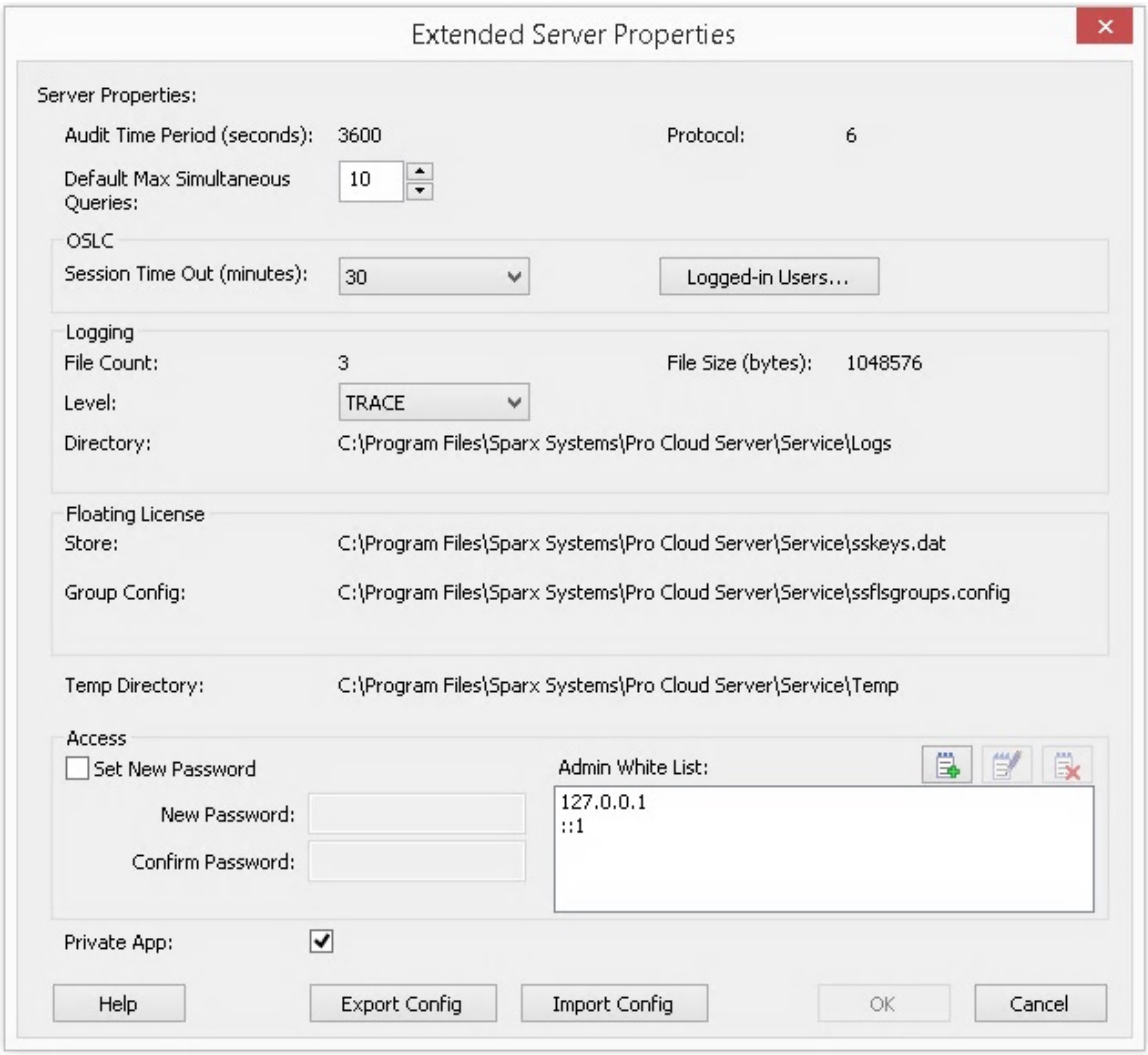
5. Make a note of ODBC DSN.
6. Open the ODBC Data Source Administration screen, locate the ODBC DSN and click on the Configure button.
7. Make a note of all database settings saved within the DSN. That is, Server Name, Port, Database Name and User. The database user password will be hidden so it is expected that someone within your organization would know the password for the given database user.
8. Change the 'Alias' value; we suggest adding the text '(odbc)' to the end, which will rename the original Database Manager to something else.
9. Close the Configure Database Manager screen.
10. Click on the Add button on the 'Database Managers' tab, this should display the Add Database Manager (Native) screen.
11. Type in the 'Alias' value, this being the same as for the original Database Manager.
12. Enter the database details that you noted in Step 7.
13. Click on the Test button to confirm that a connection can be made.
14. Click on the OK button to save the connection.
15. Edit the Database Manager you created in Step 10 and adjust all the settings to match those of the original Database Manager.
16. Test that your clients can access the repository (using

the original Alias).

17. Once everything is confirmed to be working as expected the renamed Database Manager can be deleted.

Extended Server Properties

The Extended Server Properties screen displays when you click on the 'Extended Properties' button on the Main screen. It helps you to configure the global options applied to the Pro 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
Audit Time Period	Read-only value showing the number of seconds between audit reports.
Default Max Simultaneous Queries	Default Value: 10 Provides a default value for the setting of 'Max Simultaneous Queries' when defining new database managers. For more detail see 'Max Simultaneous Queries' on the *Configure Database Manager* Help topic.
OSLC Session Time Out	To use an OSLC session each client must first log into a repository, however due to the stateless nature of HTTP/HTTPS requests the server does not always know when the client has completed their session and as such needs to enforce a timeout period due to inactivity. This option controls that inactivity timeout period. The timeout is important in situations where a PCS "Token" license is used. In this case, a token is considered 'consumed' as soon as the client sends the login request and it is released when the client logs out, however if the client fails to log out the token will stay consumed until the inactivity period is reached.

	Default Value: 30 Click on the drop-down arrow and select the maximum time period, in minutes, after which an inactive OSLC user is logged out of the model.
Logged-in Users	Use this option to open the "Logged-in OSLC Users" dialog and view the users who are currently logged into models using OSLC. Note : • Users logged into non-security enabled models will be represented as {User-<COUNT>} - for example : {User-1}
Logging File Count	Default Value: 3 Read-only value showing the number of log files that should be retained. All logging is written to files named 'SSProCloudServerLog-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 'SSProCloudServerLog-1.log' file is created.

Logging File Size	Default Value: 1048576 Read-only value showing the maximum file size (measured in bytes) 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. The valid options are: • OFF - No logging is performed • FATAL - Only fatal errors are logged • WARNING - Warnings are logged • INFO - Information messages that provide general advice on what actions are occurring • SYSTEM - More detailed messages • DEBUG - (new in PCS v4.1) Normally only used to help debug potential issues • TRACE - (new in PCS v4.1) The most detailed level of logging Log Levels are cumulative; for example, the INFO level includes all messages of type FATAL, WARNING and INFO.
Logging Directory	Read-only value showing the physical directory into which the log files are saved.

Floating License Store	Displays the path of the floating license store file. By default this is: C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service\sskeys.dat.
Floating License Group Config	Displays the path of the floating license group configuration file. By default this is: C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service\ssflsgroups.config.
Temp Directory	Read-only value showing the physical path of the temp directory; by default this is: C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service\Temp.
Set New Password	Select this checkbox to set or change the administrator password for the currently connected Pro Cloud Server. If a password is set then all users attempting to access the Pro Cloud Server Configuration Client will be required to provide the specified password.
Admin White	Added in Pro Cloud Server v4.1.

List	Default Value: 127.0.0.1 and ::1 The Admin Whitelist can be used to restrict access to the Pro Cloud Server configuration: • When using the Configuration Client (Windows Application), users will only be able to connect if their IP address is in the Whitelist. • When using WebConfig, the IP of the web server (hosting WebConfig) will need to be included in the Whitelist, which unfortunately means there is no way to restrict individual users, as all users request will arrive at the PCS with the IP of the web server. Whether the IP address is represented as IPv4 or IPv6 depends on a number of factors (like the connection method, the type of client application, the client's operating system etc) however Sparx Systems would recommend that for each authorized user both their IPv4 and IPv6 numbers should be defined in the white list. Use the Add, Edit and Delete buttons to maintain the client IP numbers.

	Note: • An empty White List will allow all clients without any restrictions. • If WebConfig is hosted on the same machine as the Pro Cloud Server you will typically need to add 127.0.0.1 or ::1 to the admin whitelist.
Private App	Added in Pro Cloud Server v5.0. Default Value: unchecked When this value is checked the Pro Cloud Server does not include its Application Name, Version or Author on the HTML page returned to clients when its URL is opened. ie in a browser open: *http://localhost/:1804*
Help	This button displays the relevant page in the online User Guide for the current screen.
Export Config	Use this option to export the Pro Cloud Server settings to a '.pcscfg' file. This includes the Server Settings, Model Connections, Port and Integration settings. This is particularly useful when moving the Pro Cloud Server Installation to a different server, or setting up multiple servers with similar

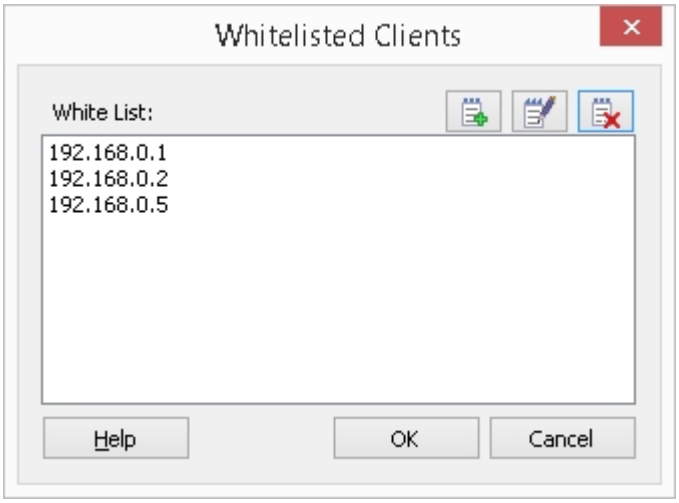
	configurations. An exported '.pcscfg' file can be imported via the 'Import Config' option described below.
Import Config	Warning: Using this function clears and overrides the existing Pro Cloud Server configuration. This includes all Server Settings, Model Connections, Port and Integration settings. Typically this option should only be used on a new Pro Cloud Server installation. Use this option to import Pro Cloud Server settings from a '.pcscfg' file. This is particularly useful when moving the Pro Cloud Server from one server to another, or when setting up a new Pro Cloud Server with a similar configuration to the exported one.

Access Control

The Whitelisted Clients screen displays when you click on the 'Access Control' button on the Main screen. It allows you to configure the what Enterprise Architect clients are authorized to communicate with the Pro Cloud Server. By default all clients are authorized, however if your environment requires a higher level of security then it is possible to limit exactly what clients can access the Pro Cloud Server.

Note, when using WebEA the IP of the web server (hosting WebEA) will need to be included in the whitelist, which unfortunately means there is no way to restrict individual WebEA users, as all users request will arrive at the PCS with the IP of the web server.





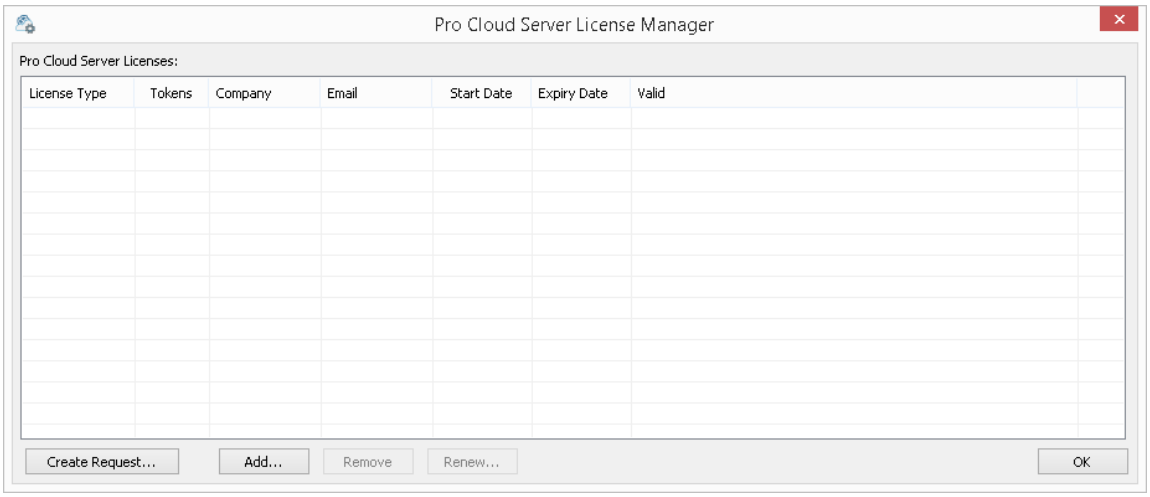
Details

Option	Description
List	Displays a list of whitelisted client addresses. The client address can be either an IPv4 or an IPv6 address; that is, 255.255.255.255 or ABCD:EF01:2345:6789:ABCD:EF01:2345:6789.
Add	Provides the ability to add a single client (either an IPv4 or an IPv6) address to the whitelist.
Edit	Provides the ability to edit an existing client address.
Delete	Provides the ability to remove an existing client address from the white list.

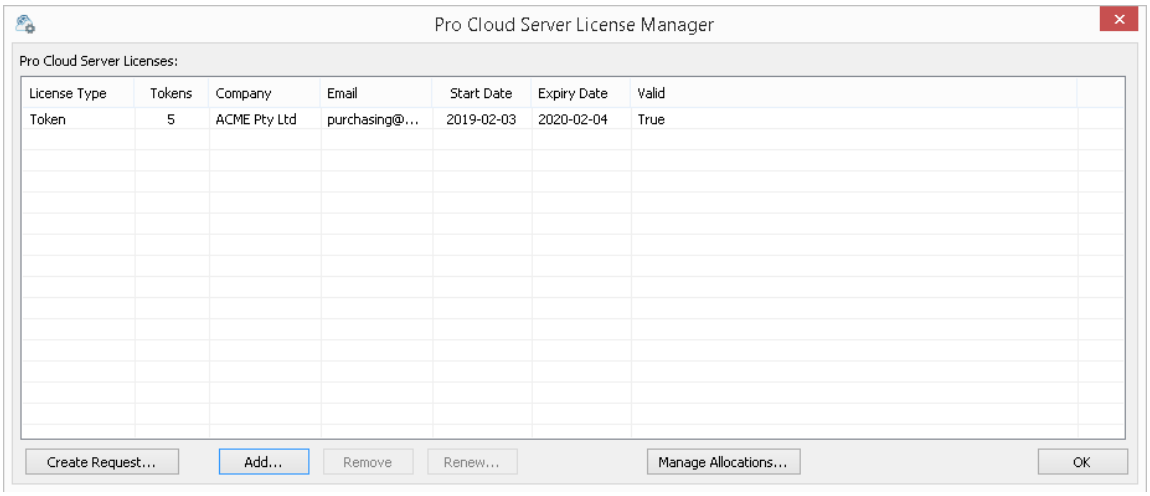
Pro Cloud Server Licensing

The Pro Cloud Server License Manager screen displays whenever you click on the Licensing button on the Main screen. It allows you to quickly and easily see what Pro Cloud Server licenses have been defined, and the complete details of each license.

On a new installation, when the Pro Cloud Server License Manager screen is first shown it resembles this:



However, over time as your Pro Cloud Licenses expire you will be able to see a complete history of all licenses:



In this example, the current Token license is valid from

February 3rd, 2019 to February 4th, 2020 and has 5 tokens.

If there is any kind of problem with the license, the 'Valid' column will show the reason; this message is also written to the current log file.

License Procedure

As of Pro Cloud Server version 4.0.30, the licensing procedure has been enhanced. Pro Cloud Server licenses are now linked to the host machine, which is an enforcement of what was stated in the license agreement. To add a License, the basic procedure to follow is:

1. Purchase the required Pro Cloud Server license.
2. Create a 'License Request' from the Pro Cloud Server machine.
3. Send the 'License Request' file to the Sparx Systems Sales team.
4. The Sparx Systems Sales Team will then return the 'License Certificate'.
5. Add your 'License Certificate' to your Pro Cloud Server installation.
6. Your Pro Cloud Server is now licensed!

Along with the change to Licensing procedures, Pro Cloud Server 4.0.30 introduced the 'Token' Edition, which provides a more flexible upgrade path by making use of tokens that can be purchased separately at any time to enable only the extra functionality your environment requires.

For the premium editions ('Team' and 'Enterprise') the enabled functionality remains unchanged, except that the 'Team' edition now allows 7 Pro-enabled models, 2 more than in earlier versions.

Licensing Screen Options

As shown in the earlier screen shots, the Pro Cloud Server License Manager screen contains a list of Licenses and a selection of action buttons, which are described here:

Option	Description
Pro Cloud Server Licenses	The list of Licenses displays a historical record of all Licenses for the current Pro Cloud Server. Each License will display values in these columns: • License Type - One of 'Token', 'Team' or 'Enterprise' • Tokens - The number of Tokens associated with the License • Company - The name of the company that the Pro Cloud Server is licensed to • Email - The email address associated with this license; that is, the email address that was provided when the License Request was created • Start Date - The date the License is valid from • Expiry Date - The date the License expires on

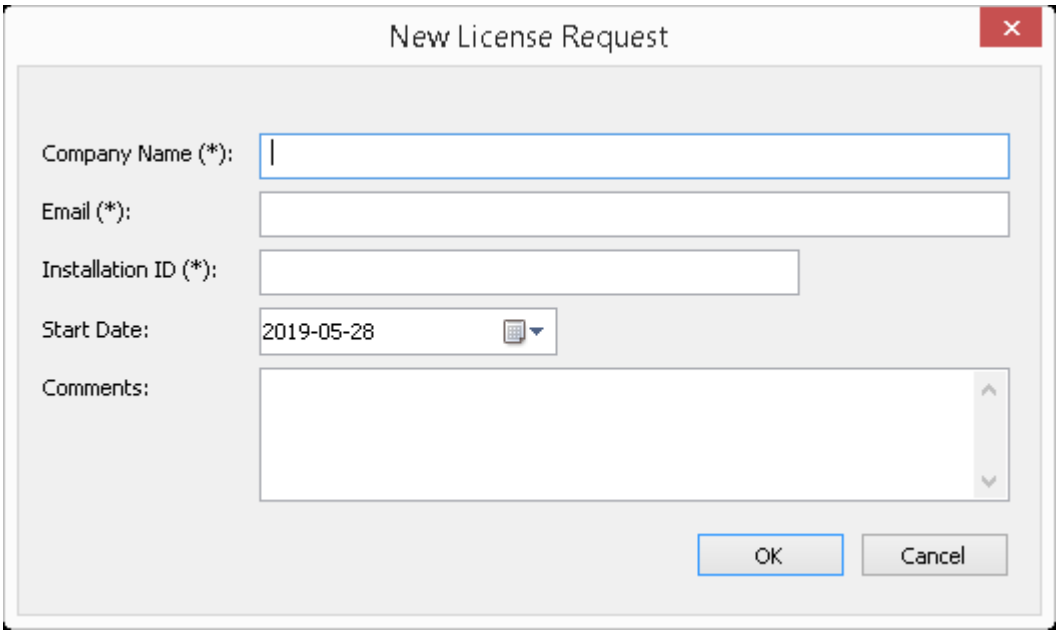
	• Valid - A value of 'True' indicates that the current license is loaded, otherwise an error message displays explaining why it is not valid
Create Request...	The Create Request button is permanently visible on the Pro Cloud Server License Manager screen. Selecting this button will display the New License Request screen, which allows the user the create a new Pro Cloud Server License request.
Add ...	The Add button is permanently visible on the Pro Cloud Server License Manager screen. Selecting it will prompt the user to select a Pro Cloud Server License Certificate file (*.CRT) in the file system.
Remove	The Remove button is permanently visible on the Pro Cloud Server License Manager screen, however it is only enabled when a License is selected in the list. Selecting this button will remove the selected Pro Cloud Server License from list. If the Remove action was performed by mistake, the License Certificate file (.CRT) will still exist in the file system under the, {installation path}\Licenses\deleted\ folder. Therefore to

	reinstate the deleted license, manually copy the {GUID}.CRT from the \deleted\ folder to {installation path}\Licenses\ and restart the Pro Cloud Server.
Renew...	The Renew button is permanently visible on the Pro Cloud Server License Manager screen; however, it is only enabled when a License is selected in the list. Selecting this button will display the 'New License Request' screen, but some of the fields will be pre-populated with the details of the selected License.
Manage Allocations...	The Manage Allocations... button is only visible when the selected Pro Cloud Server License is a Token license. Selecting this button will display the 'Manage License Token Allocations' screen, which allows the user to control the minimum and/or maximum number of Tokens associated with each Pro Feature.

Create Request

The New License Request screen is displayed when the user clicks on the Create Request button on the Pro Cloud Server License Manager screen. The purpose of this screen is to create a license request for the connected Pro Cloud Server installation.

The New License Request screen will initially be displayed as shown here:



The creation of a License Request is required after purchasing a Pro Cloud Server license. The License Request helps Sparx Systems identify and register what hardware is going to host the Pro Cloud Server.

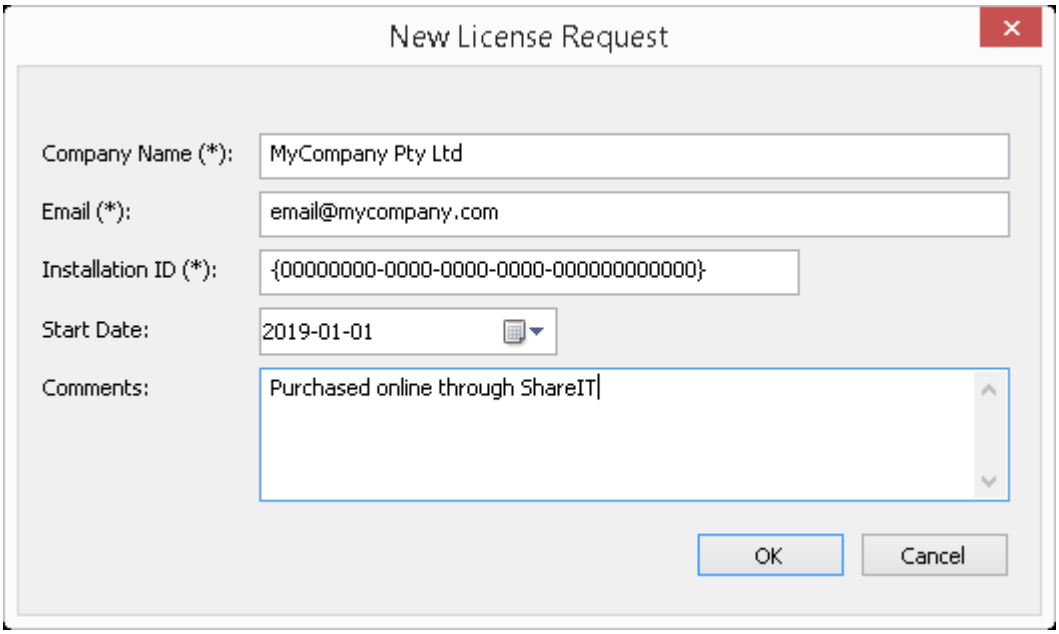
Complete a License Request

The New License Request screen prompts for these input details:

Field	Description
Company Name	Mandatory value. The name of the company that made the Pro Cloud Server purchase.
Email	Mandatory value. The email address associated with the Pro Cloud Server purchase.
Installation ID	Mandatory value. The unique GUID that was supplied in the email to confirm your Pro Cloud Server purchase. If you purchased your Pro Cloud Server license prior to June 2019, use your Pro Cloud Server version 3 License key.
Start Date	Optional value. Use this date to request that your Pro Cloud Server License should start on a future date. If no value is supplied the Sparx Systems Sales Team will assume the current date.
Comments	Optional value. Any additional comments that might be helpful to the Sparx Systems Sales Team.

Submit the Request

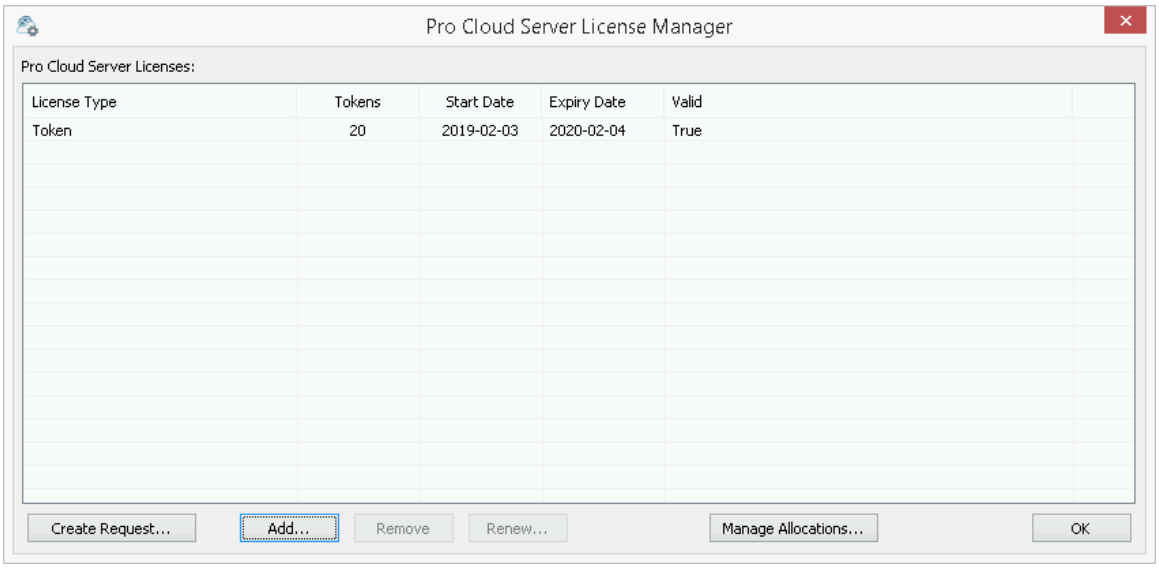
In order to minimize any potential delays please provide the Sales team with as much detail as possible about your Pro Cloud Server License purchase. For example:



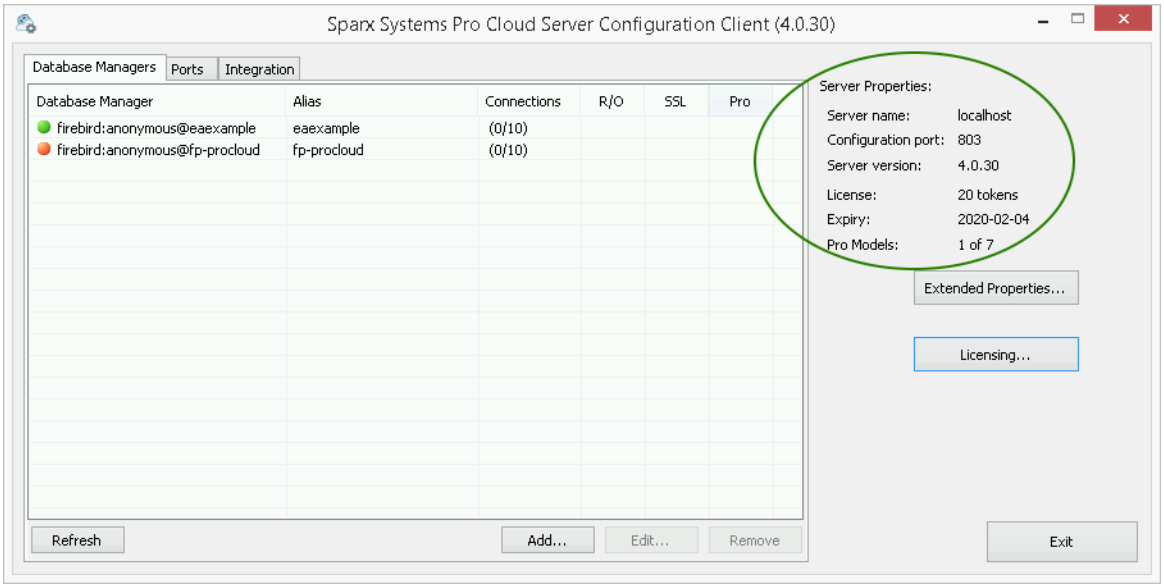
Once you have entered all Pro Cloud Server purchase information, click on the OK button; a prompt displays to save the resulting License Request to the file system. The file should then be emailed to the Sparx Systems Sales team: sales@sparxsystems.com.

Add a New License

After you submit a License Request, the Sparx Systems Sales Team will reply with your License Certificate. When this occurs click on the Add button on the Pro Cloud Server License Manager screen and select the supplied License Certificate file (*.lic or *.crt). The details of the License Certificate will be loaded and shown on the License screen, resembling this:



When the License screen is closed, the Main screen will now reflect the new License in the 'Server Properties' section:


Sparx Systems Pro Cloud Server Configuration Client (4.0.30)
Database Managers
Ports
Integration
Database Manager
Alias
Connections
R/O
SSL
Pro
firebird:anonymous@eaexample
eaexample
(0/10)
firebird:anonymous@fp-procloud
fp-procloud
(0/10)
Server Properties:
Server name: localhost
Configuration port: 803
Server version: 4.0.30
License: 20 tokens
Expiry: 2020-02-04
Pro Models: 1 of 7
Extended Properties...
Licensing...
Refresh
Add...
Edit...
Remove
Exit

Remove a License

It is possible to remove, at any time, an existing Pro Cloud Server License, simply by selecting the license in the list on the Pro Cloud Server License Manager screen and clicking on the Remove button. The list of allocated licenses will be refreshed accordingly. There is no need to restart the Pro Cloud Server, the internal License and all related functionality enabled (or disabled) by the change or removal of a License will take effect immediately.

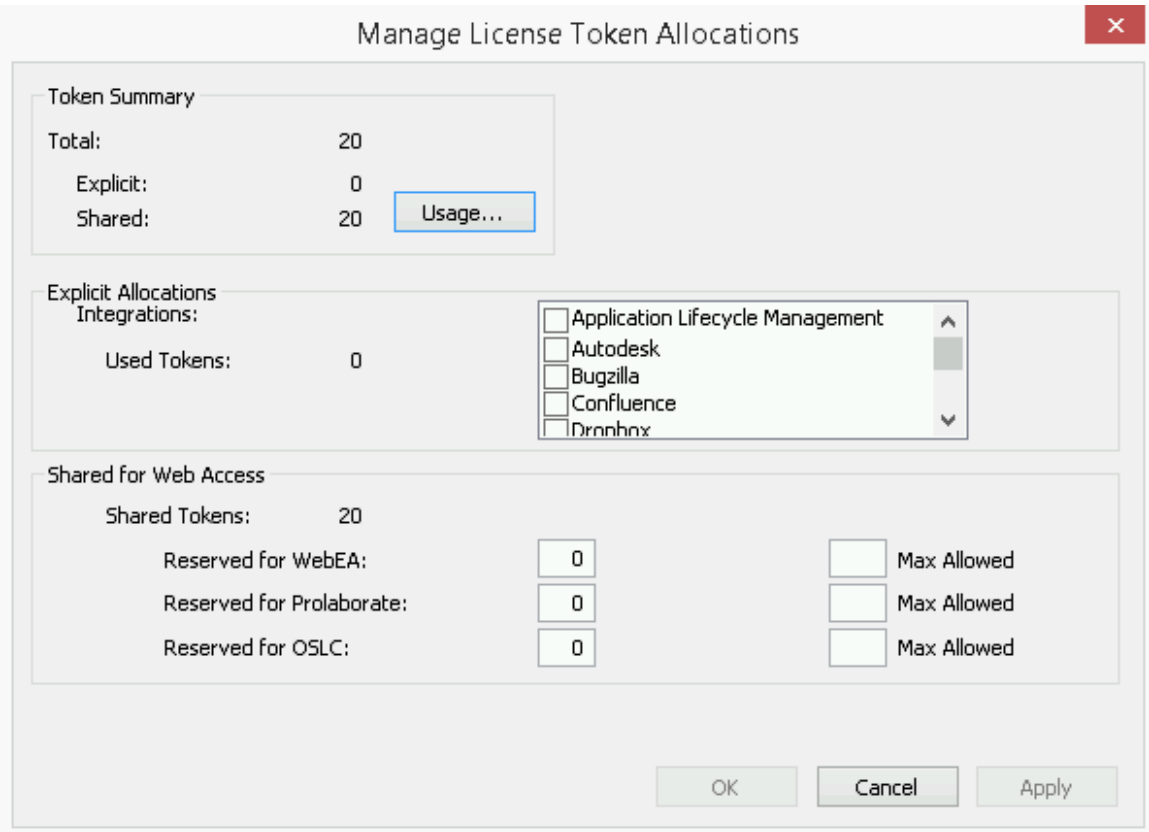
Renew a License

On the Pro Cloud Server License Manager screen, the Renew button will be enabled whenever a License is selected in the list of licenses. It is a convenient function that displays the same New License Request screen as is shown by the Create Request button; however, the difference is that the 'Renew' option will pre-populate the valid details of the selected License, although the remaining fields will still have to be filled in manually. The renewal process is the same as the initial process and will have to be repeated whenever the License expires.

Manage Allocations

The Manage License Token Allocations screen is displayed when you click on the Manage Allocations button on the Pro Cloud Server License Manager screen. This screen allows you to control how tokens are consumed to enable Pro Feature functionality.

Note: The Manage Allocations button is only visible when the Pro Cloud Server has a loaded Token-based license.



The Manage Allocations screen is broken up into three sections: a Summary of token usage, Explicit Allocations, and Allocations of tokens that are shared for Web Access.

Token Summary

The token summary shows:

- Total - The total number of tokens based on the active license(s)
- Explicit - The number of explicit token allocations, which refers to tokens that have been allocated to enable specific features on the server
- Shared - The number of shared tokens that are available; these are consumed when users access features such as WebEA - when logging out, the token is returned to the shared token pool

Click on the Usage button to display a more detailed breakdown of the token allocations, including how many tokens are currently in use.

Explicit Allocations

This section allows you to enable Integration Plug-ins for your Pro Cloud Server. Enabling integration with a specific product (such as Jira) consumes one token.

Shared for Web Access

WebEA, Prolaborate and OSLC users consume a single token with every logon. For most environments these allocations can be left with their default values (0 reserved and no maximum). In this case all users are accessing a single shared pool of tokens, regardless of what type of access (WebEA, Prolaborate or OSLC) they require. Alternatively, each user type can define a number of tokens that are reserved (that is, they can only be used by that type

of access) as well as a maximum number.

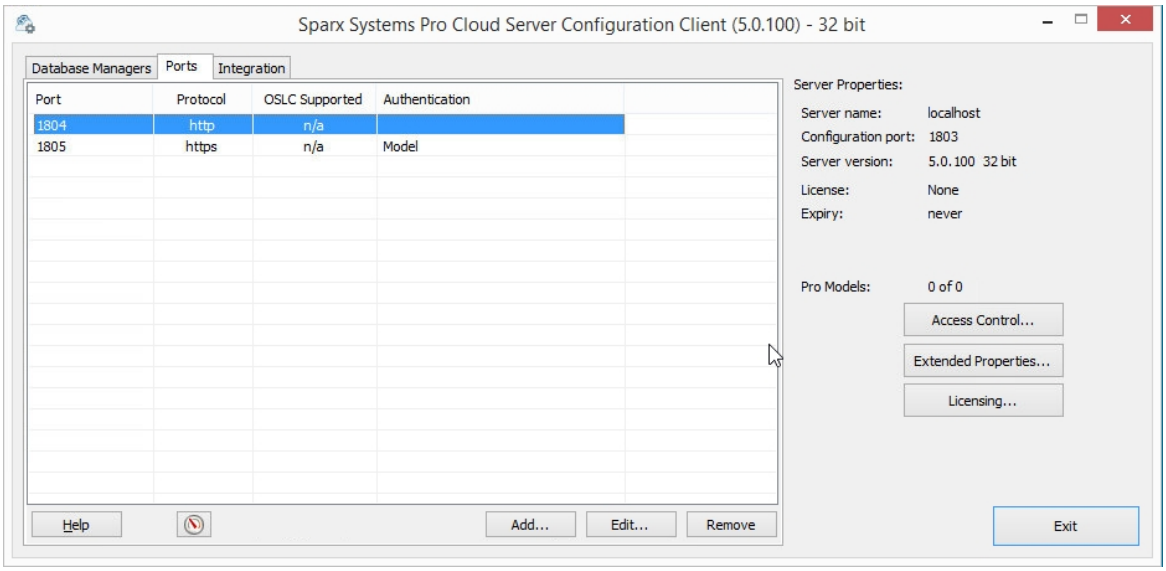
Notes:

- A **Max Allowed** value of 0 implies that users are NOT allowed; that is, if WebEA's **Max Allowed** value was set to 0, the Pro Cloud Server would deny all WebEA user requests to logon
- An empty value for **Max Allowed** implies there is no limit on the number of users, although the total number of users is still limited to the number of available tokens
- Using WebEA's 'Logout' option will immediately return the token; however, if the web browser is closed without using 'Logout' then the token will be returned after 30 minutes of inactivity.

Port Configuration

When setting up the Pro Cloud Server you will need to define at least one Port that will be used for clients communicating to the Pro Cloud Server (this includes Enterprise Architect) to access data stored in Cloud based repositories.

Ports can be configured via the 'Ports' tab in the Pro Cloud Server Configuration client.



The Ports Tab

The 'Ports' tab provides information and a number of options:

Item	Description
Port List	The main area of the 'Ports' tab provides a list of the existing Port definitions. These Port settings are also covered in

	more detail in the Add a Port Definition topic. **Columns** • Port - The Port number • Protocol - Displays the transfer protocol (http or https) that the Port has been configured to communicate on • OSLC Supported - Displays True if OSLC support is enabled for this Port; OSLC Support is required for WebEA access • Authentication - Indicates what type of Authentication is required (if any)
Help	Click on this button to display the online User Guide page describing the current screen.
Resource Monitor	Located in the lower left of the 'Ports' tab, this button can be used to open the Windows Resource Monitor. This allows you to check which Ports are already in use before adding a new Port Definition. Please note: this will open the Windows Resource Monitor for the current machine, which is only relevant if the Configuration Client is running on the machine hosting the Cloud Server.

Add	Click on this button to add a new Port Definition.
Edit	Click on this button to edit the configuration of the selected Port. Editing a Port provides the same options as when adding a new one, except that the Port number cannot be changed.
Remove	Click on this button to permanently delete the selected Port Configuration

Notes:

In order to configure a Port with SSL (HTTPS) the Pro Cloud Server requires a certificate/private file, see *SSL Certificates* section in Add a Port Definition.

Any update made to the Port definitions will not take effect until the Pro Cloud Server is restarted.

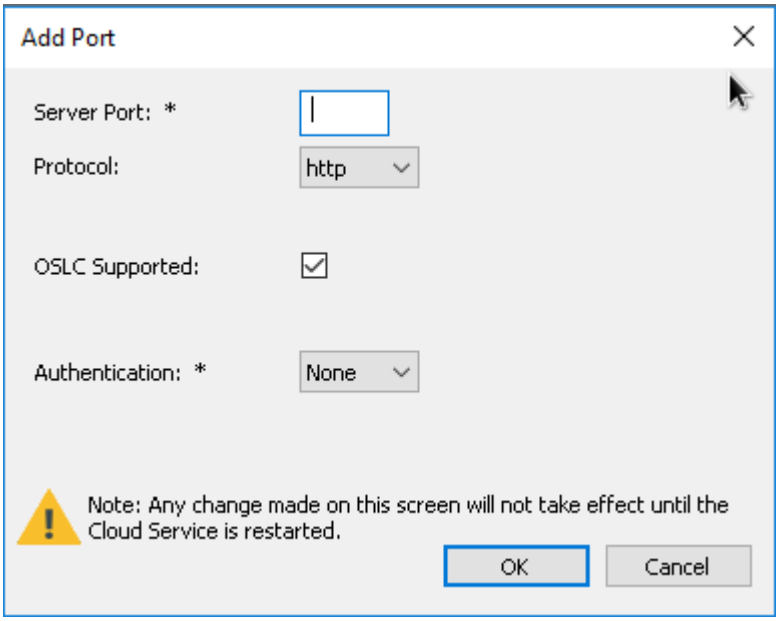
Add a Port Definition

As mentioned in the *Port Configuration* topic, when setting up the Pro Cloud Server you must define at least one Port for the Pro Cloud Server to listen on. Initially the Pro Cloud Server is supplied with one HTTP Port (804) and one HTTPS Port (805).

Note: The default HTTPS Port will not work until a server certificate has been installed as described in the *SSL Certificates* section.

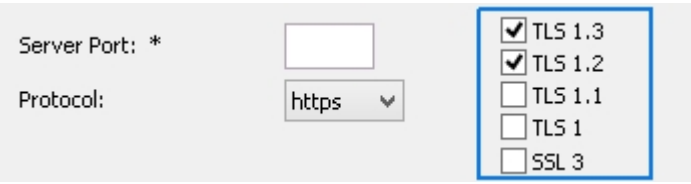
To add a new Port Definition:

1. Open the Pro Cloud Server Configuration Client - Application.
2. Select the 'Ports' tab.
3. Click on the Add button.
4. The 'Add TCP Port' dialog will be displayed. Referring to the *Add TCP Port* table, enter the Port details, then click on the OK button.
5. Confirm that the machine's Firewall Settings will allow clients to connect.
6. Restart the Pro Cloud Server Windows Service so that the changes take effect.



Add TCP Port

Field	Description
Server Port	Enter a valid Port number. Each defined Port must be unique and not used by any other service or application on the current machine. The standard HTTP Port (80) and HTTPS Port (443) are no different; if they are available, the Pro Cloud Server can be configured to use them. You can use the Windows Resource Monitor (which is accessibly via an icon on the Configuration Client 'Ports' tab) to check which Ports are already in use. **Firewall Note**: When changing or adding a new Port, check that incoming requests

	are allowed by the firewall settings at both the local machine and network levels. **Wine Note**: 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.
Protocol	Select the protocol that the Port will be configured as. The options are: • HTTP - The default value • HTTPS - When this option is selected, you will be presented with five additional checkboxes that control which versions of TLS (and SSL) are allowed on the given Port  Note: TLS 1.3 is the latest, it was defined in August 2018. TLS 1.2 was defined in August 2008, TLS 1.1 in April 2006, TLS 1.0 in January 1999 and SSL 3.0 in 1996. SSL 3.0 was deprecated in June 2015 and TLS 1.1 and 1.0 were mostly deprecated since March 2020. Therefore by default PCS will only enable TLS 1.3 and 1.2.

	Important: For HTTPS to be configured correctly, a file named *server.pem* must also exist in the same directory as the configuration file, and it must contain a valid private key/certificate. See the Using HTTPS (SSL) topic for details.
OSLC Supported	Enable this option to allow OSLC access to models. **Note:** WebEA makes use of OSLC; if you intend to access models via WebEA then this option should be enabled
Authentication	Select the type of authentication that will be required for this Port: • **Model** - All model connections that use the current Port will be forced 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 will not work. Windows and OpenID SSO authentication is supported however the option to automatically create or modify users is not supported in this mode. Note: 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 • **Global** - Selecting this option displays an additional 'Global Model' field, in which you enter the DB Alias of a Database Manager that will be used to provide the security profile to all connections opened with the current Port; 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 Global model using the *Maintain Users* procedure; the default administrator password will not work. Note: If you are not using HTTPS as the protocol, the usernames and passwords will be sent in plain text, therefore Sparx Systems recommends that Global authentication should only be used with HTTPS.

Note on SSO authentication (Windows

Active Directory or OpenID)

By default, the HTTPS port has 'Model authentication' enabled. When connecting using SSO authentication (either Windows Active Directory Domain authentication or OpenID) this means that the user is authenticated by the Pro Cloud Server, and not by the Enterprise Architect client.

For this to work the Pro Cloud Server service must have access to the authentication provider:

- Windows Authentication: the service must have access and permissions to the Domain Controller. By default the service runs as Local System, which will inherit the permissions of the logged in user. If the logged in user is a Domain member then it will have access by default, but in more restrictive domain environments it may be necessary to configure the service to run as a Domain Member and not as Local System
- OpenID: For OpenID to work it needs internet access to the OpenID provider's URLs used for authentication. If there is a proxy connection then this needs to be set for account the runs the Pro Cloud Server service. By default this is Local System - so setting the proxy through Internet Options is not sufficient and needs to be set on an elevated command line using commands such as: netsh winhttp set proxy 10.0.0.1:8080

Also note that the option to 'Automatically create or modify Windows or OpenID users' is not supported by Pro Cloud Server when using authenticated ports.

Manual Port Configuration

Optionally, Ports can also be configured by directly editing the Pro Cloud Server's configuration file (SSProCloud.config). This allows some additional settings, which are not available in the Pro Cloud Configuration Client, a manual restart of the Pro Cloud Server Windows Service will still be required before any manually adjusted settings to take effect.

The definition of a Port is denoted in the configuration file, with open and close parentheses, each on their own lines without any leading whitespace. Inside the parentheses are a number of supporting options that control and limit how the Port can be used. All options except SERVER_PORT are optional.

```
    (
    SERVER_PORT=804
    REQUIRE_SSL=0
    OSLC_SUPPORT=1
    MODEL_AUTHENTICATION=
    GLOBAL_AUTHENTICATION=user model
    EA_MINIMUM_BUILD=
    EA_MAXIMUM_BUILD=
    SSL_PROTOCOLS=SSLv3 | TLSv1 | TLSv1_1 |
TLSv1_2 | TLSv1_3
    )
```

Settings Explained

Setting	Description
SERVER_PORT	This option expects a numeric value (whole number) that defines a Port number; there is no default value. For example, SERVER_PORT=804 states that the Pro Cloud Server service will listen to HTTP connections on Port 804. Each defined Port must be unique and not used by any other service or application on the current machine. The standard HTTP Port (80) and HTTPS Port (443) are no different, if they are available the Pro Cloud Server can be configured to use them. **Firewall Note**: When changing or adding a new Port, check that incoming requests are allowed by the firewall settings at both the local machine and network levels **Wine Note:** 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	This option expects a value of either 0 or 1; the default is 0. A value of 1 will enforce the use of SSL on the current Port; that is, all connections must be performed using HTTPS, which is useful when connections need to be exposed on public networks. In order for the Pro Cloud Server to be able to successfully configure Ports as REQUIRE_SSL=1, a file called *server.pem* must exist in the same directory as the configuration file, and contain a valid private key/certificate. For more details on *server.pem* see the earlier *SSL Certificates* section.
SSL_PROTOCOLS	This option is a pipe-delimited list of TLS/SSL protocol versions accepted by the Port. If omitted, the default value of TLSv1_2 | TLSv1_3 is assumed. Valid values are 'SSLv3', 'TLSv1', 'TLSv1_1', 'TLSv1_2', 'TLSv1_3'. For example, if a Port was to accept versions TLS 1.1, 1.2 and 1.3 an entry should be: SSL_PROTOCOLS=TLSv1_1 | TLSv1_2 | TLSv1_3

OSLC_SUPPORT	This option expects a value of either 0 or 1. The default is 1. A value of 1 defines that connections that use the current Port will be able to perform queries using the 'Open Services for Lifecycle Collaboration' standard. This is discussed further in the *OSLC Requirements Management* Help topic.
MODEL_AUTHENTICATION	This option expects a value of either 0 or 1. The default is 0. By defining this option as **1**, all model connections that use the current Port will be forced to request HTTP authorization using the user security defined in the Enterprise Architect repository being connected to. Passwords must be explicitly and individually assigned in that repository using the *Maintain Users* procedure; the default administrator password and any passwords imported from Windows Active Directory will not work. Note that if you are not using HTTPS/SSL to connect, the usernames and passwords will be sent in plain text. If the repository/model does not have security enabled, the user is not prompted for a password. This option is mutually exclusive with

	GLOBAL_AUTHENTICATION.
GLOBAL_AUTHENTICATION	This option expects the alias of a defined database manager, or blank, the default is blank. By defining this option a single Enterprise Architect repository with security enabled can be used to provide the security profile to all connections opened with the current Port. This is helpful if you want to expose multiple repositories/models but only manage one list of users. Passwords must be explicitly and individually assigned in the Global model using the *Maintain Users* procedure; the default administrator password and any passwords imported from Windows Active Directory will not work. This option is mutually exclusive with **MODEL_AUTHENTICATION**.
EA_MINIMUM_BUILD	This option expects a numeric value that represents the minimum Enterprise Architect version that can connect to models using this Port. The default is undefined, which implies all versions are allowed.

EA_MAXIMUM_BUILD	This option expects a numeric value that represents the maximum Enterprise Architect version that can connect to models using this Port. The default is undefined, which implies all versions are allowed.
DEFAULT_MODEL	This option was removed in Pro Cloud Server 4.1.40; earlier versions could have used this value to limit all connections on the current Port to a single repository. However most customers that attempted to use this setting found it confusing and its use tended to cause configuration issues, therefore the decision was made to remove it from the interface.

Using HTTPS (SSL)

In order for the Pro Cloud Server to use an HTTPS (SSL) Port a valid SSL certificate is required. Having obtained a valid certificate, the certificate file and your private key file need to be concatenated to create a **server.pem** file. The **server.pem** file must then be added to the Pro Cloud Server installation's 'Service' folder (e.g. C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service).

Note on SSO (Windows Authentication and OpenID)

By default, the HTTPS port has 'Model authentication' enabled. When connecting using SSO authentication (either Windows Active Directory Domain authentication or OpenID) this means that the user is authenticated by the Pro Cloud Server, and not by the Enterprise Architect client.

For this to work the Pro Cloud Server service must have access to the authentication provider;

- Windows Authentication: the service must have access and permissions to the Domain Controller. By default the service runs as Local System, which will inherit the permissions of the logged in user. If the logged in user is a Domain member then it will have access by default, but in more restrictive domain environments it may be necessary to configure the service to run as a Domain Member and

not as Local System

- OpenID: For OpenID to work it needs internet access to the OpenID provider's URLs used for authentication. If there is a proxy connection then this needs to be set for account the runs the Pro Cloud Server service. By default this is Local System - so setting the proxy through Internet Options is not sufficient and needs to be set on an elevated command line using commands such as: netsh winhttp set proxy 10.0.0.1:8080

Configuring the Pro Cloud Server to use HTTPS

Step	Description
1	Obtain a signed SSL certificate. For the Pro Cloud Server you will require the **certificate file** and your **private key file** (both should be base64 encoded). Note, this is the same process used when obtaining an SSL certificate for a website that uses HTTPS. There are many ways to achieve this, and the specific details are beyond the scope of this documentation. In general, it is preferred to use a certificate that has been signed by a globally trusted root Certificate

	Authority (CA). Some certificate authorities provide a service of signing SSL certificates for a fee; there are also free options available such as 'Let's Encrypt'. In some cases (e.g. for testing purposes, or for setups that are entirely within a closed network) you might consider using a self-signed certificate, or create your own Certificate Authority and use this to sign your certificate/s. See the Self-Signed SSL Certificates and Using Your Own Certificate Authority (CA)Help topics for details.
2	Create a **server.pem** file by concatenating your **certificate file** and **private key file**. For example, use this command: Windows: copy /b cloud.mycompany.com.crt+cloud.mycompany.com.key server.pem Linux: cat cloud.mycompany.com.crt cloud.mycompany.com.key > server.pem
3	Copy the **server.pem** to the Pro Cloud Server installation's 'Service' folder (e.g. C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service).
4	If you have not already, configure a Pro Cloud Server Port that has the protocol set to HTTPS (for a new installation Port 805 is set to use HTTPS by default). See the Add a Port Definition Help topic for details.

5	Restart the Pro Cloud Server service.
6	Check the Pro Cloud Server log file (e.g. C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service\Logs\SSProCloudServerLog-1.log) and confirm that the Port opened successfully. For example: 2022-01-01 15:00:03 [INFO]: Thread X SUCCESS Bound and listening on soap Port 805 (https)

Self-Signed SSL Certificates

Configuring the Pro Cloud Server to use HTTPS requires a signed SSL certificate. In general, it is preferable to use a certificate that has been signed by a globally trusted root Certificate Authority (CA) however in some cases (e.g. when testing) you might consider creating and using your own self-signed certificate.

Creating a Self-Signed certificate

There are many ways to create SSL certificates; however, we have found the OpenSSL toolkit to be the easiest and most comprehensive. As the name suggests, OpenSSL is an open source toolkit for SSL/TLS; see the official website for details.

You will need to download and install the OpenSSL product that suits your environment. For the purposes of this document, OpenSSL was installed into C:\OpenSSL-Win64. If your OpenSSL is installed into a different directory then these steps will have to be adjusted to suit.

- Create a new command/batch file in C:\OpenSSL-Win64\bin\
- Open this file and paste these details into it

```
set SRVNAME=localhost
openssl genrsa -out %SRVNAME%.key 2048
openssl req -new -x509 -key %SRVNAME%.key -out %SRVNAME%.cert -days 3650 -subj /CN=%SRVNAME%
```

copy /b %SRVNAME%.cert+%SRVNAME%.key server.pem

For details on configuring the Pro Cloud Server to use this certificate, refer to the Using HTTPS (SSL) topic.

Using Self-Signed certificates

When configuring and using the Pro Cloud Server with a self-signed certificate, these points should be taken into consideration.

Feature	Description
Enterprise Architect	When accessing a model in Enterprise Architect via cloud connection you will receive a security warning regarding the self-signed certificate.
Integration Plug-Ins	When configuration integration plugins if your are adding a data provider which accesses an Enterprise Architect model then the 'Ignore SSL Error' option will needs to be enabled.
WebEA	If WebEA is configured to access a model via HTTPS (with a self signed certificate) then the WebEA configuration needs the 'Validate SSL Certificates' set to 'No (I.e. the webea_conifg.ini should include

	sscs_enforce_certs="false")
WebConfig	If the WebConfig is configured to use HTTPS (with a self-signed certificate), then the settings.php file must include **$sEnforceCerts = 'false';**

Using Your Own Certificate Authority (CA)

In general, it is preferable to use a certificate that has been signed by a globally trusted root Certificate Authority (CA). There are many CAs that are trusted by all major browser and operating systems and that can be used to sign certificates for use with an https server.

If the system architecture makes using a globally trusted CA impractical, it is still possible to have client browsers and operating systems set up to trust a self-signed certificate and so avoid security errors or showing warnings to end users.

One way is to simply add the certificate to the trust store of the client browser or operating system; however this would need to be done for each certificate generated. A better approach is to create your own root Certificate Authority and use that to sign each server certificate.

To do this, an SSL certificate needs to be signed with your own Certificate Authority (CA) certificate and key, and the clients (browsers, operating systems) need to be told to trust the CA certificate. The instructions for adding a CA to a client vary according to the operating system or browser used.

Create a Certificate Authority

There are many ways to create Certificate Authority certificates; however, the OpenSSL toolkit is one of the

easiest and most comprehensive. As the name suggests, OpenSSL is an open source toolkit for SSL/TLS; see the official website for details. You will need to download and install the OpenSSL product that suits your environment.

To create a CA follow these two steps:

1. Generate a new key and certificate request.
2. Self-sign the request to generate a CA certificate.

1. Generate new key and certificate request

This can be done in multiple steps, or by generating from an existing key file, but for simplicity a new key and new request can be generated in one command. This example shows how to create a request for a company named 'My Company'. Change this to something more appropriate for your organization:

- openssl req -new -sha256 -nodes -newkey rsa:4096 -subj "/O=My Company/CN=My Company Internal CA" -keyout MyCompanyCA.key -out MyCompanyCA.csr

2. Self-sign the request to generate a CA certificate

This step self-signs the CA certificate request, and makes the CA valid for 1 year (-days 365). Change 'MyCompany' to match the key and csr request generated in the previous step:

- openssl x509 -req -sha256 -in MyCompanyCA.csr -signkey MyCompanyCA.key -days 365 -out MyCompanyCA.crt

Create a server certificate and use the CA to sign it

Now that the MyCompanyCA.crt is generated, it can be used to sign your own certificates for a Cloud Server or WebEA.

First, as in the Self-Signed SSL Certificates Help topic, create a new certificate request. This example creates a new key and certificate request for a server named 'cloud.mycompany.com':

- openssl req -new -nodes -newkey rsa:4096 -subj "/CN=cloud.mycompany.com" -keyout cloud.mycompany.com.key -out cloud.mycompany.com.csr

Sign the new certificate request with the CA:

- openssl x509 -req -CA MyCompanyCA.crt -CAkey MyCompanyCA.key -CAcreateserial -sha256 -days 365 -in cloud.mycompany.com.csr -out cloud.mycompany.com.crt

The final step for use with a Pro Cloud Server is to concatenate the key and certificate into a 'server.pem' file:

- Windows: copy /b cloud.mycompany.com.crt+cloud.mycompany.com.key server.pem

- Linux: cat cloud.mycompany.com.crt cloud.mycompany.com.key > server.pem

For details on configuring the Pro Cloud Server to use this certificate, refer to the Using HTTPS (SSL) topic.

Allow clients to trust the root CA

The client operating system or browser now needs to have the CA certificate added to its list of trusted CAs. The instructions vary according to operating system and browser, but instructions for a few major clients are listed here. For all these steps the 'certificate' referred to is the 'MyCompanyCA.crt' generated in Step 2.

Client (Operating System, Browser)	Instructions
Microsoft Windows	Right-click the CA certificate file and select 'Install Certificate'. Follow the prompts to add the certificate to the trust store either for the current user only or for all users of the computer.
Linux - Ubuntu	Copy the CA certificate to /usr/local/share/ca-certificates For example: • sudo cp ~/MyCompanyCA.crt

	/usr/local/share/ca-certificates/ Update certificates with this command: • sudo update-ca-certificates The output should show something similar to 'Adding debian:~/MyCompanyCA.pem'. If using Wine, then close all Wine programs and restart Wine: • wineserver -k See Ubuntu Help for more information.
Firefox	Firefox does not use the operating systems trust store, so the CA has to be added manually. If the certificate has a '.pem' extension, then the simplest way is to drag-and-drop the CA certificate file onto Firefox; a prompt will ask you to trust the certificate. Otherwise, manually add certificates and manage added certificates through Firefox's Privacy & Security preferences. More information can be found on the Firefox wiki.
Chrome/Chromium	Chrome and Chromium do not use the operating system trust store, so the CA needs to be added manually.

	Open Settings > Advanced > Manage Certificates > Authorities select 'Import'
Internet Explorer	Internet Explorer uses the Windows trust store, so adding the certificate to Windows (see the first entry in this table) is sufficient to add trust to the browser as well.
WebEA	WebEA uses PHP/curl to communicate with a Pro Cloud model. If the connection between PHP and the Pro Cloud uses HTTPS, then the CA can be added to PHP's configuration to allow it to trust the certificate.

Pro Cloud Server Configuration Client - Web

As an alternative to the Pro Cloud Server Configuration Application, the Pro Cloud Server also includes a Web-based Configuration Client 'WebConfig'.

Configuration Steps

Step	Description
1	If you have not done so already, install the Pro Cloud Server by working through the steps outlined in the *Pro Cloud Server Installation* Help topic, making sure that the WebConfig component is installed. After installation you should find the WebConfig website files in the Installation directory (e.g. C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\WebConfig). These files make up the 'WebConfig' website/client, and must be hosted on a web server. In the simplest configuration the web server would be installed on the same machine as the Pro Cloud Server.
2	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. Note: The WebConfig 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.
3	Define and configure a website for the WebConfig in your web server. As there are many different web servers and configurations, configuring the web site, 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; the WebConfig does not require these files, so if you have no need for them, move them to a backup location.
4	Copy the WebConfig folder from the Pro Cloud Installation (e.g. C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\WebConfig) to an appropriate location on the web server (e.g.

	C:\xampp\htdocs).
5	Within the WebConfig folder that was copied to your web server, locate the file *settings.php* and open it with a text editor. This file defines these settings: • *$sProtocol = 'http';* The Protocol used to access the Pro Cloud Server • *$sServerName = 'localhost';* The Name/IP of the machine running the Pro Cloud Server • *$sPort = '804';* The Port that should be used to access the Pro Cloud Server • *$sEnforceCerts = 'true';* Applies to https connections only When 'true', self-signed SSL certificates will be denied When 'false', self-signed SSL certificates can be used These settings determine how the WebConfig will access the Pro Cloud Server. By default, the Pro Cloud Server is configured to allow HTTP communication on Port 804; therefore, if the Pro Cloud Server and web server are on the same machine these settings can typically be left as is. Alternatively, update these values to reflect the name/IP of the Cloud Server and the appropriate Protocol and Port based on the definitions in the

	SSProCloud.config file.
6	On a remote machine, open an internet browser and enter the URL http://{yourwebserver}/WebConfig/index.php, where the value '{yourwebserver}' is either the name or IP number of the machine hosting the WebConfig website. This should display the 'PCS Configuration - Login' prompt. Enter the password and click on the Login button to login. As a security precaution from PCS version 4.1 the default password for the Configuration Clients is: pcsadm1n For all versions prior to PCS version 4.1 the default password was empty (i.e. no password), in which case you are logged in automatically. Note, the Pro Cloud Server will only be accessible if the current machine's IP address is on the Pro Cloud Server's WebConfig - Server Settings.

WebConfig Server Requirements

To set up the WebConfig Client you require:

- A web server (Apache or IIS) capable of hosting PHP v7.0 (or later)
- 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 WebConfig website pages

Note: These requirements are the same as for hosting WebEA; the WebConfig could be deployed on the same web server as WebEA

Managing Multiple Pro Cloud Servers

As mentioned above, by default the WebConfig page will connect to the Pro Cloud Server as defined in the settings.php file. Alternatively, an additional 'server' URL parameter can be used to access a different Pro Cloud Server (effectively overriding the settings.php). This allows a single WebConfig to be used for accessing multiple different Pro Cloud Servers.

For example, if your URL for accessing the WebConfig is:

http://sparx-pcs1/webconfig/index.php

You could use this URL to connect to the Pro Cloud Server

on 'sparx-pcsdemo':

http://sparx-pcs1/webconfig/index.php?server=**http**://**sparx-pcsdemo**:**804**

This will then connect to the Pro Cloud Server using:

Protocol: http

Server name/IP: sparx-pcsdemo

Port: 804

Note, for HTTPS connections the $sEnforceCerts setting from the settings.php file will be used.

As usual, the Pro Cloud Server will only be accessible if the current machine's IP address is on the Pro Cloud Server's WebConfig - Server Settings.

WebConfig - Login

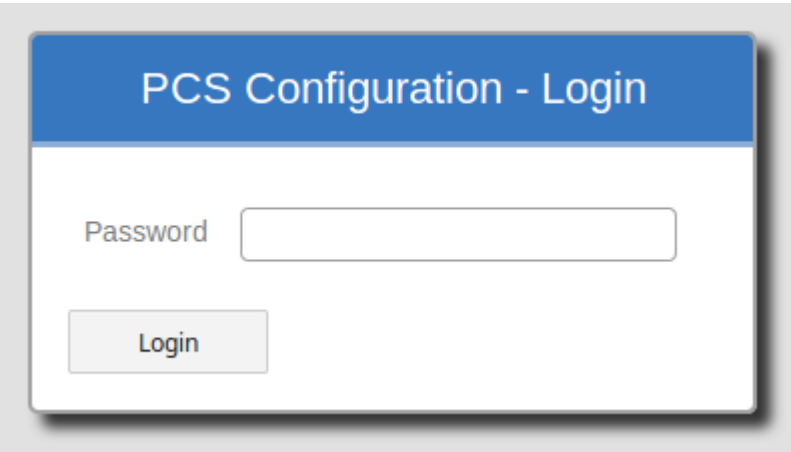
Having setup the WebConfig client/website you can login by entering the relevant URL. For example:

http://{yourwebserver}/WebConfig/

or

http://{yourwebserver}/WebConfig/index.php

If a password has been assigned then the PCS Configuration - Login prompt will be displayed.



Enter the password and click on the Login button to login. As a security precaution from PCS version 4.1 the default password for the Configuration Clients is:

pcsadm1n

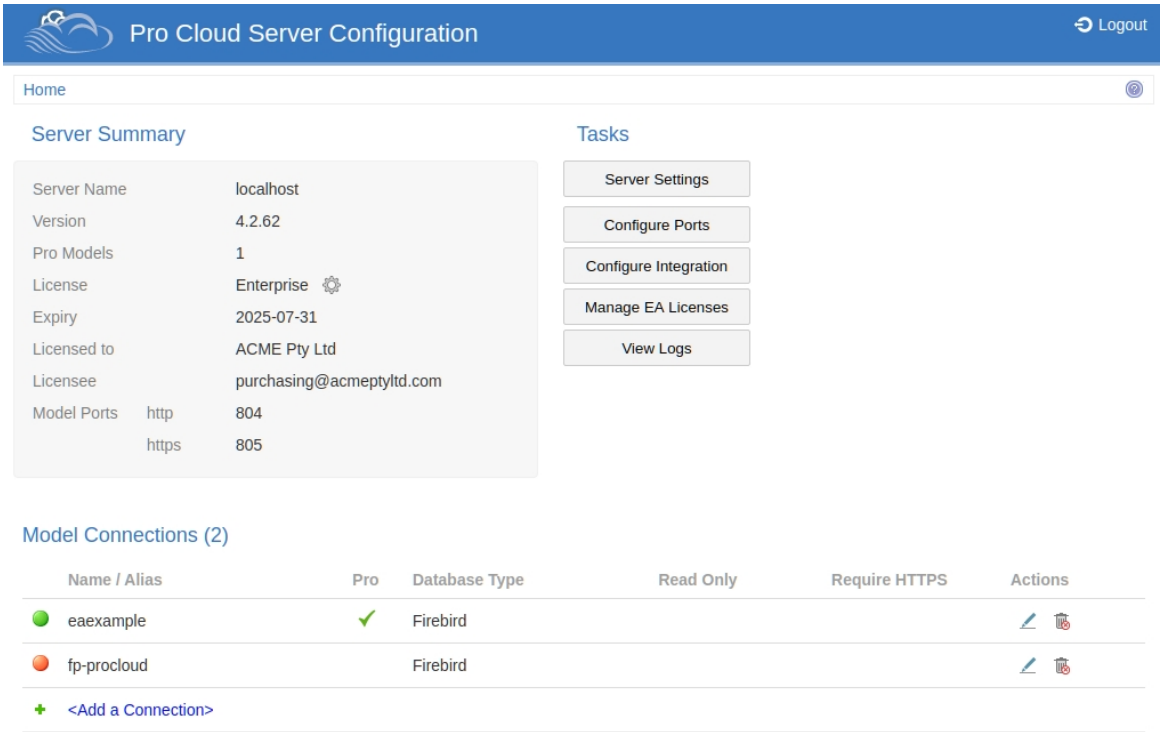
For all versions prior to PCS version 4.1 the default password was empty (i.e. no password), in which case you are logged in automatically.

If necessary, after logging in you can set/reset the password for the Cloud Server via the 'WebConfig - Server Settings |

Change Password' option.

WebConfig - Home Screen

The Pro Cloud Server Configuration 'Home' page has three sections - 'Server Summary', 'Tasks' and 'Model Connections'.



Server Summary

The 'Server Summary' section provides information regarding the current server, including server name, version, license and Port configurations.

Field	Description
Server Name	The Server Name which is being used to access the Pro Cloud Server. Note, in this case 'localhost' would refer to the

	machine which hosts WebConfig.
Version	The version number of the Pro Cloud Server that the WebConfig is accessing.
Pro Models	The number of models that currently have Pro Features enabled.
License	The edition of the Pro Cloud Server (based on the license). If no license has been added then an 'Add' link will be displayed here (which will take you to the 'Pro Cloud Server Licenses' screen').
Expiry	The expiry date of the Pro Cloud Server License.
Licensed to	The name of the company to which the Pro Cloud Server is licensed.
Licensee	The email address associated with this license, this being the email address that was provided when the License Request was created.
Model Ports	Lists Ports that are currently defined for use by Model Connections.

Tasks

The tasks section provides links to perform various tasks. The options available can vary depending on the type of license.

Button	Description
Server Settings	Provides options to configure the Pro Cloud Server (log level, password, and Cloud Server License), and displays additional server information (log settings, and paths to relevant configuration files and directories).
Configure Ports	View and configure Ports that the Pro Cloud Server will use for accessing models.
Configure Integration	Configure the Integration Server and Integration Providers, and bind Integration Providers with Pro Models Connections.
Manage EA Licenses	Manage floating licenses for Enterprise Architect (and Enterprise Architect Add-Ins). Note: This option is only available when using the Team Server or Enterprise

	edition of the Pro Cloud Server; it is not available when using the Token edition.
View Logs	Navigates to the 'View Logs' page, which lists the Cloud Server log files. On the 'View Logs' page, click on the 'View Log' icon in the 'Actions' column to view the contents of the log file.

Model Connections

Lists the model connections defined for the Cloud Server. The list includes this information.

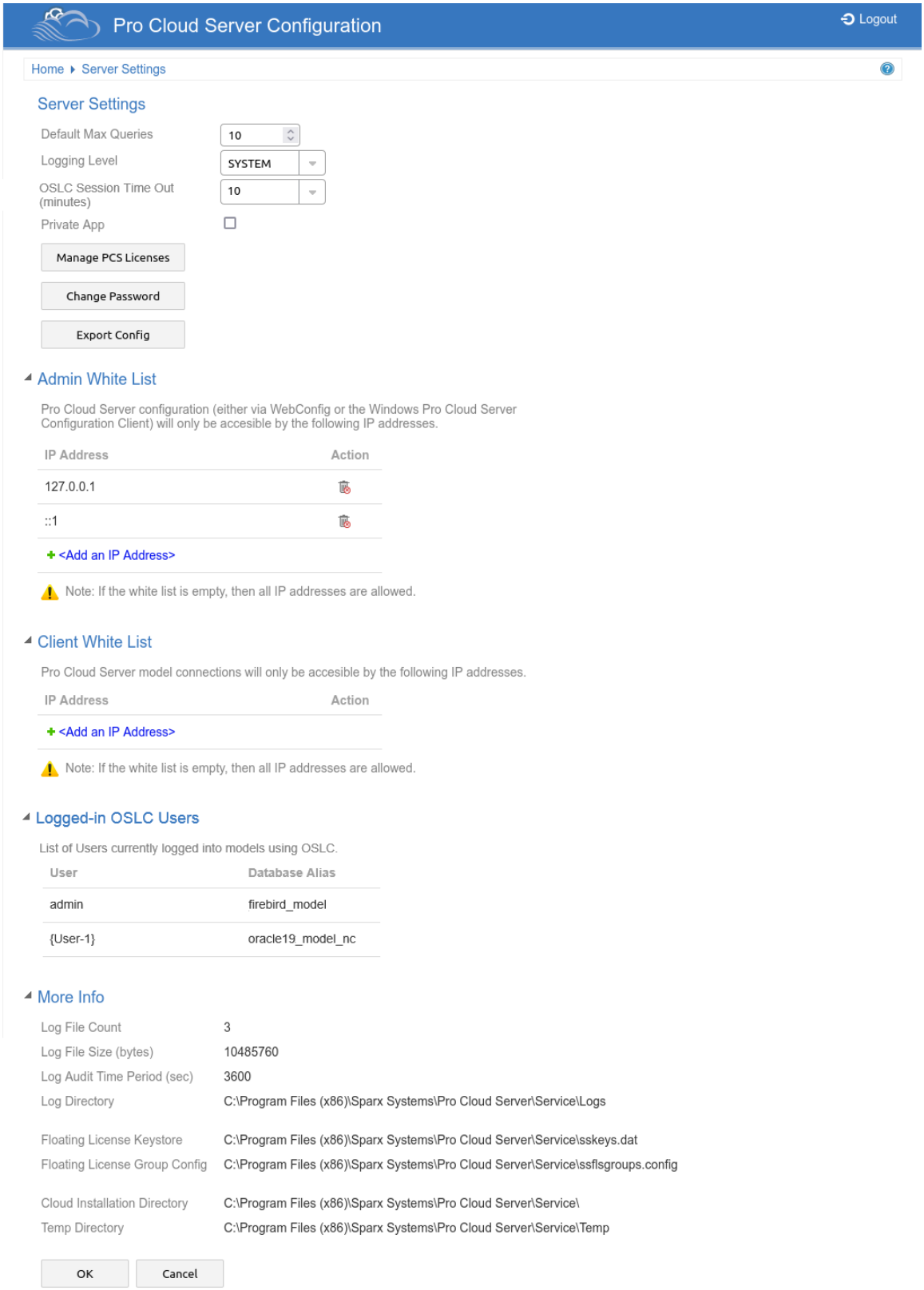
Item	Description
Model Status	To the left of the 'Name / Alias' field is a status icon. Red indicates that the connection has not been enabled (via the 'Enable Access' option), meaning this connection cannot currently be used. Green indicates that the model connection is enabled.
Name / Alias	This field displays the name or alias of the model connection. When connecting to a Cloud model via Enterprise Architect, this name or alias should be

	entered in the 'Model Name' field.
Pro	This field indicates whether Pro features are enabled for this model connection. Pro features include the ability to access the model connection via WebEA, and the ability to use Integration Plug-ins (for Confluence or Jira, for example). A green tick indicates that Pro features are enabled, while a blank field indicates they are not. An alert icon indicates that the 'Pro Features' option was enabled, but Pro Features are not currently enabled/accessible for the model connection. For example, this could occur if Pro Features were enabled but the Pro Cloud License has expired. When the alert icon is displayed, check the icon's tooltip for more information.
Database Type	Displays the database type of the model (such as Firebird or SQL Server).
Read Only	Indicates whether this connection provides read-only access to the model.
Require HTTPS	Indicates that the model connection can only be accessed via the HTTPS protocol

	(via a Port that is also configured to use the HTTPS protocol).
Actions	Provides icons to perform these actions: • Edit - Edit the model connection settings • Delete - Delete the model connection
Add a Connection	The list always contains, at the bottom, an option to add a new connection.

WebConfig - Server Settings

The 'Server Settings' page provides various options and information regarding the Pro Cloud Server Configuration.



Server Settings

Item	Description
Default Max Queries	Model connections can specify the maximum number of requests that can be serviced concurrently. This setting determines the default 'Max Simultaneous Queries' when defining a new model connection. Default: 10
Logging Level	Click on the drop-down arrow and select the level of information to be written to the log file. The valid options are: • OFF - No logging is performed • FATAL - Only fatal errors are logged • WARNING - Warnings are logged • INFO - Information messages that provide general advice on what actions are occurring • SYSTEM - More detailed messages • DEBUG - (new in PCS v4.1) Normally only used to help debug potential issues • TRACE - (new in PCS v4.1) The most detailed level of logging Log Levels are cumulative; for example, the INFO level includes all messages of type FATAL, WARNING and INFO.

OSLC Session Time Out	To use an OSLC session each client must first log into a repository, however due to the stateless nature of HTTP/HTTPS requests the server does not always know when the client has completed their session and as such needs to enforce a timeout period due to inactivity. This option controls that inactivity timeout period. The timeout is important in situations where a PCS "Token" license is used. In this case, a token is considered 'consumed' as soon as the client sends the login request and it is released when the client logs out, however if the client fails to log out the token will stay consumed until the inactivity period is reached. Click on the drop-down arrow and select the maximum time period, in minutes, after which an inactive OSLC user is logged out of the model. Default Value: 30
Private App	Added in Pro Cloud Server v5.0. When this value is checked the Pro Cloud Server does not include its Application Name, Version or Author on the HTML page returned to clients when its URL (e.g. http://localhost/:1804) is opened. Default Value: Unchecked

Manage PCS Licenses	Navigates to the 'Pro Cloud Server Licenses' screen, which allows you to add or remove Pro Cloud Licenses.
Change Password	Allows you to change the password for accessing the Cloud Configuration (or set a password if one has not already been set). Changing or setting a password will also log you out of the Cloud Configuration, returning you to the login screen.
Export Config	Use this option to export the Pro Cloud Server settings to a file ('pcsconfig.pcscfg'). This includes the Server Settings, Model Connections, Port and Integration settings. This is particularly useful when moving the Pro Cloud Server Installation to a different server, or setting up multiple servers with similar configurations. The exported file can be imported via the 'Import Config' option in the Pro Cloud Configuration Client Windows application. See the 'Extended Server Properties' topic for details.
Admin White	Added in Pro Cloud Server v4.1.

List	Default Value: 127.0.0.1 and ::1 The Admin Whitelist can be used to restrict access to the Pro Cloud Server configuration: • When using the Configuration Client (Windows Application), users will only be able to connect if their IP address is on the Whitelist. • When using WebConfig, the IP of the web server (hosting WebConfig) will need to be included in the Whitelist, which unfortunately means there is no way to restrict individual users, as all users request will arrive at the PCS with the IP of the web server. Whether the IP address is represented as IPv4 or IPv6 depends on a number of factors (like the connection method, the type of client application, the client's operating system etc) however Sparx Systems would recommend that for each authorized user both their IPv4 and IPv6 numbers should be defined in the white list. Use the 'Add an IP Address' button, and the Delete button in the Action column to maintain the client IP numbers.

	Note: • An empty White List will allow all clients without any restrictions. • If WebConfig is hosted on the same machine as the Pro Cloud Server you will typically need to add 127.0.0.1 or ::1 to the admin whitelist.
Client White List	Default Value: <empty> The Client Whitelist allows you to configure what Enterprise Architect clients are authorized to communicate with the Pro Cloud Server. By default all clients are authorized, however if your environment requires a higher level of security then it is possible to limit exactly what clients can access the Pro Cloud Server. Note, when using WebEA the IP of the web server (hosting WebEA) will need to be included in the whitelist, which unfortunately means there is no way to restrict individual WebEA users, as all users request will arrive at the PCS with the IP of the web server. Whether the IP address is represented as

	IPv4 or IPv6 depends on a number of factors (like the connection method, the type of client application, the client's operating system etc) however Sparx Systems would recommend that for each authorized user both their IPv4 and IPv6 numbers should be defined in the white list. Use the 'Add an IP Address' button, and the Delete button in the Action column to maintain the client IP numbers. Note: • An empty White List will allow all clients without any restrictions. • If WebEA is hosted on the same machine as the Pro Cloud Server you will typically need to add 127.0.0.1 or ::1 to the admin whitelist.
Logged-in OSLC Users	The Logged-in OSLC Users table shows the list of users who are currently logged into models using OSLC. Note : • Users logged into non-security enabled models will be represented as {User-<COUNT>} - for example : {User-1}

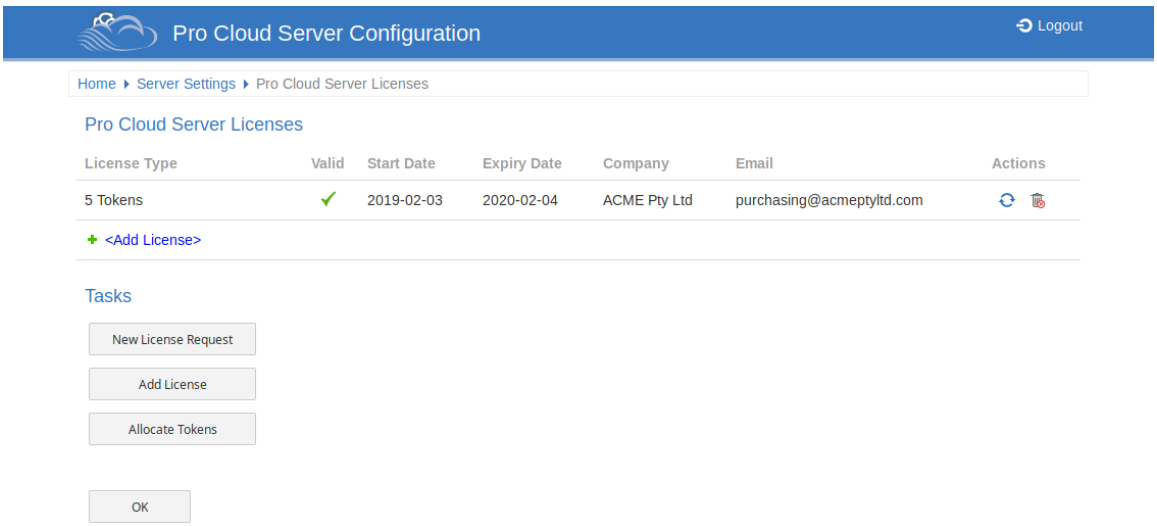
More Info	The 'More Info' section contains a number of read-only fields providing information regarding the Cloud Server Configuration. **Log File Count** The number of log files that should be retained. All logging is written to files named 'SparxCloudServicesLog-X.log' where X represents the file number. File 'SSProCloudServerLog-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 'SSProCloudServerLog-1.log' created. **Log File Size (bytes)** The maximum file size of each log file. When the file reaches this maximum, a new file is created. **Log Audit Time Period (sec)** The number of seconds between audit reports. **Log Directory** The directory into which the log files are saved. **Floating License Keystore** The full path of the Floating License

	Keystore file. **Floating License Group Config** The full path of the Floating License Keystore group configuration file. **Cloud Installation Directory** The directory into which the Cloud Server is installed. **Temp Directory** The Cloud Server Temp directory.

WebConfig - Pro Cloud Server Licenses

The Pro Cloud Server Licenses screen can be accessed from the WebConfig - Home Screen, either by using the 'Add Link' icon found in the 'Server Summary' section, or by clicking on the Server Settings button and then on the Manage PCS Licenses button.

This screen allows you to quickly and easily see what Pro Cloud Server licenses have been defined, and the complete details of each license.



In this example the current Token license is valid from February 3rd, 2019 to February 4th, 2020 and has 5 tokens.

If there is any kind of problem with the license the 'Valid' column will show an alert icon. Mouse-over the icon and check the tooltip for more information on the problem. Details of the issue will also be written to the current log file.

License Procedure

As of Pro Cloud Server version 4.0.30 Pro Cloud Server licenses are linked to the host machine, which is an enforcement of what is stated in the license agreement.

Along with the change to Licensing procedures, Pro Cloud Server 4.0.30 introduced the 'Token' Edition, which provides a more flexible upgrade path by making use of tokens that can be purchased separately at any time to enable only the extra functionality your environment requires.

For the premium editions (Team and Enterprise) the enabled functionality remains unchanged, except that the Team edition now allows seven Pro-enabled models, which is an increase of two compared to earlier versions.

To add a License, the basic procedure is:

Step	Description
1	Purchase the required Pro Cloud Server license (e.g. via the Sparx Systems Website).
2	Create a 'New License Request'.
3	Send the 'License Request' file to the Sparx Systems Sales team with the details of your purchase.
4	The Sparx Systems Sales Team will then return the 'License Certificate'.

5	Add your 'License Certificate' to your Pro Cloud Server installation.
6	Your Pro Cloud Server is now licensed!

Pro Cloud Server Licenses

The Pro Cloud Server Licenses screen displays a table with information and actions for each Pro Cloud Server license that has been added.

Field	Description
License Type	The License Type (either Token, Team or Enterprise). In the case of a Token edition license, this field displays the number of Tokens.
Valid	Displays a tick to indicate that the license is valid and enabled. If the license is not valid an alert icon is displayed. Mouse-over the icon and read the tooltip for details of why the license is considered invalid.
Start Date	The date the License is valid from.
Expiry Date	The date the License expires on.

Company	The name of the purchasing company provided on the License Request.
Email	The contact email address of the person or department that purchased the License.
Actions	Provides buttons to either renew or delete the License.

Tasks

Below the 'Pro Cloud Server Licenses' table is a 'Tasks' section, which provides these options.

Option	Description
New License Request	Create a new license request.
Add License	Add a license to the Pro Cloud Server.
Allocate Tokens	Allocate Tokens to enable specific Pro Cloud Server features. Note: This option appears only when using the Token edition of the Pro Cloud Server.

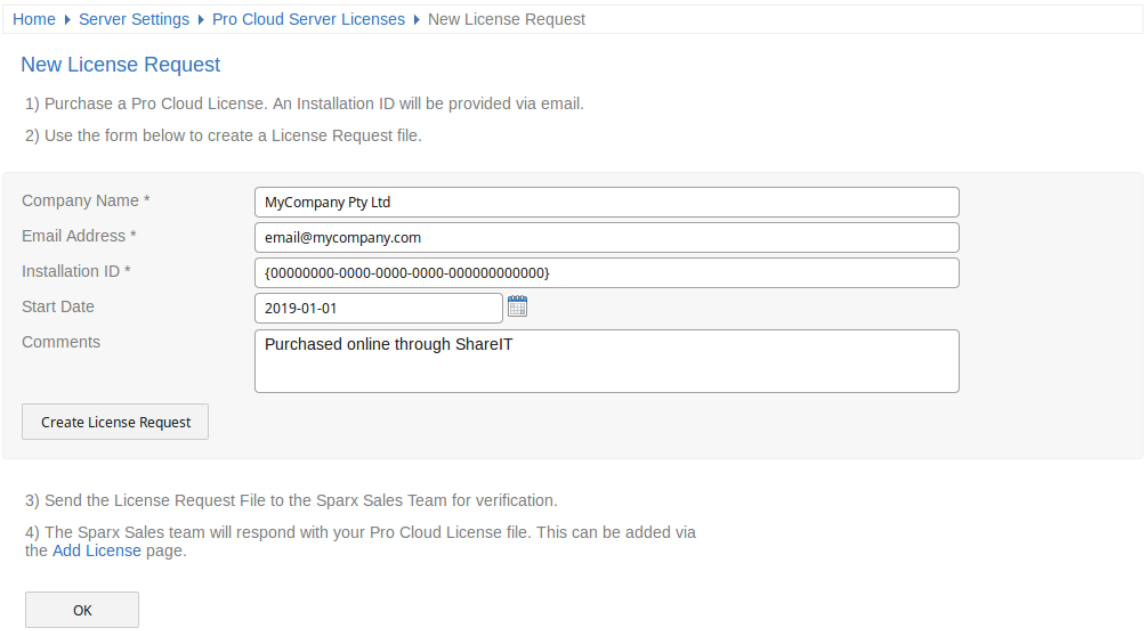
WebConfig - New License Request

Pro Cloud Server licenses are linked to a specific machine. After purchasing the Pro Cloud Server you will be provided with an Installation ID. You will then need to create a License Request file and send this to the Sparx Systems sales team. The Sales team will respond with your License Certificate file.

Note: When creating the License Request it is important that the Pro Cloud Server you are accessing is running on the machine you intend to use. The License Certificate that you are provided with will be valid for this machine only.

To create a new license request, open the WebConfig - Pro Cloud Server Licenses screen, then click on the New License Request button. The New License Request screen is displayed.

Fill in the details of your company and the Pro Cloud Server purchase (refer to the *New License Request Form* table for more information), for example:



Click on the Create License Request button, then email the saved file to the Sparx Systems Sales team on sales@sparxsystems.com.

New License Request Form

The 'New License Request' form contains these fields:

Company Name	Mandatory value. The name of the company that made the Pro Cloud Server purchase.
Email	Mandatory value. The email address associated with the Pro Cloud Server purchase.
Installation ID	Mandatory value. The unique 'Installation ID' that was supplied in the confirmation

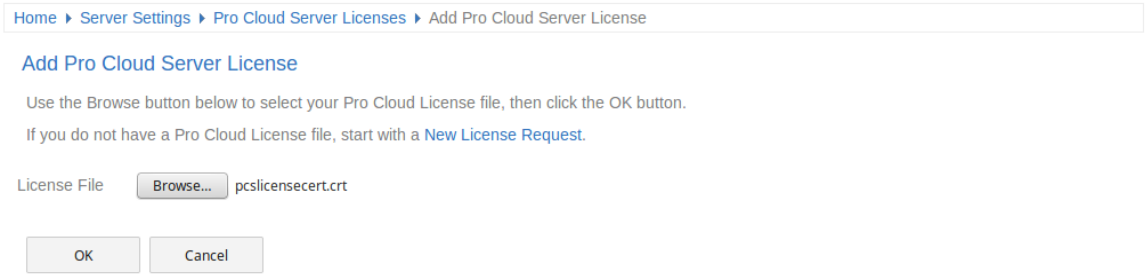
	email for your Pro Cloud Server purchase. If you purchased your Pro Cloud Server license prior to June 2019, use your Pro Cloud Server version 3 License key.
Start Date	Optional value. Use this date to request that your Pro Cloud Server License should start on a future date. If no value is supplied the Sparx System Sales Team will assume the current date.
Comments	Optional value. Any additional comments that might be helpful to the Sparx Systems Sales Team.

WebConfig - Add a License

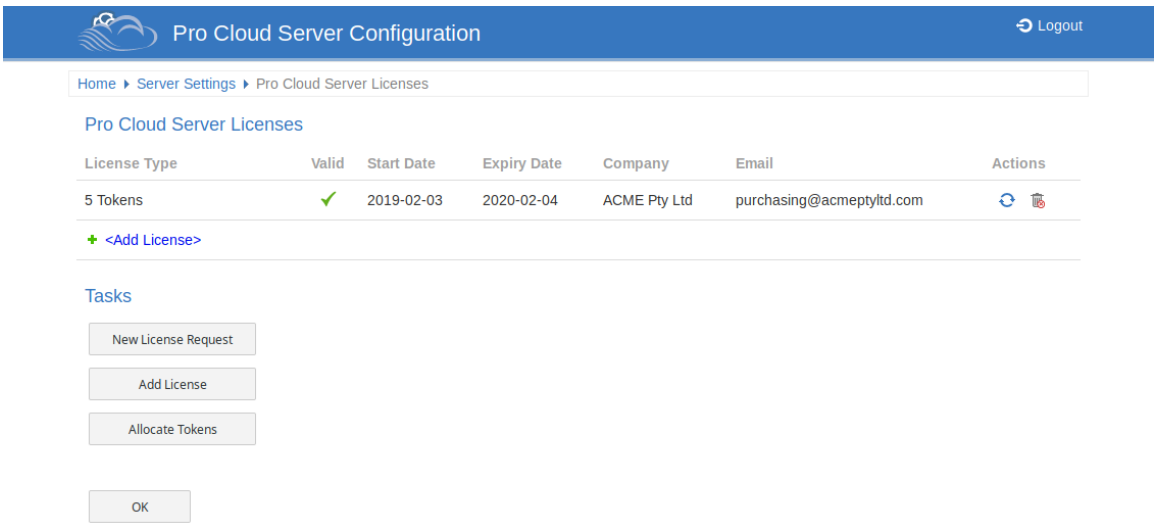
After you send a License Request to the Sparx System's Sales Team, they will reply with your License Certificate file. You can then add your license to the Pro Cloud Server.

On the WebConfig - Pro Cloud Server Licenses screen, click on the Add License button. The 'Add Pro Cloud Server License' screen is then displayed.

Click on the Browse button, select the supplied License Certificate file (*.lic or *.crt), then click on the OK button.



The details of the License Certificate will be loaded and the license should appear in the list on the Pro Cloud Server Licenses screen, similar to this:



When returning to the Pro Cloud Server Configuration

'Home' page, the 'Server Summary' details should now display the appropriate information for your license.

WebConfig - Remove a License

On the WebConfig - Pro Cloud Server Licenses screen, the 'Actions' column for each existing license contains a 'Renew License' icon and a 'Delete License' icon.

Click on the 'Delete License' icon to delete and remove that license. The list of allocated licenses will be refreshed accordingly; there is no need to restart the Pro Cloud Server, the internal License and all related functionality enabled (or disabled) by the change or removal of a License will take effect immediately.

WebConfig - Renew a License

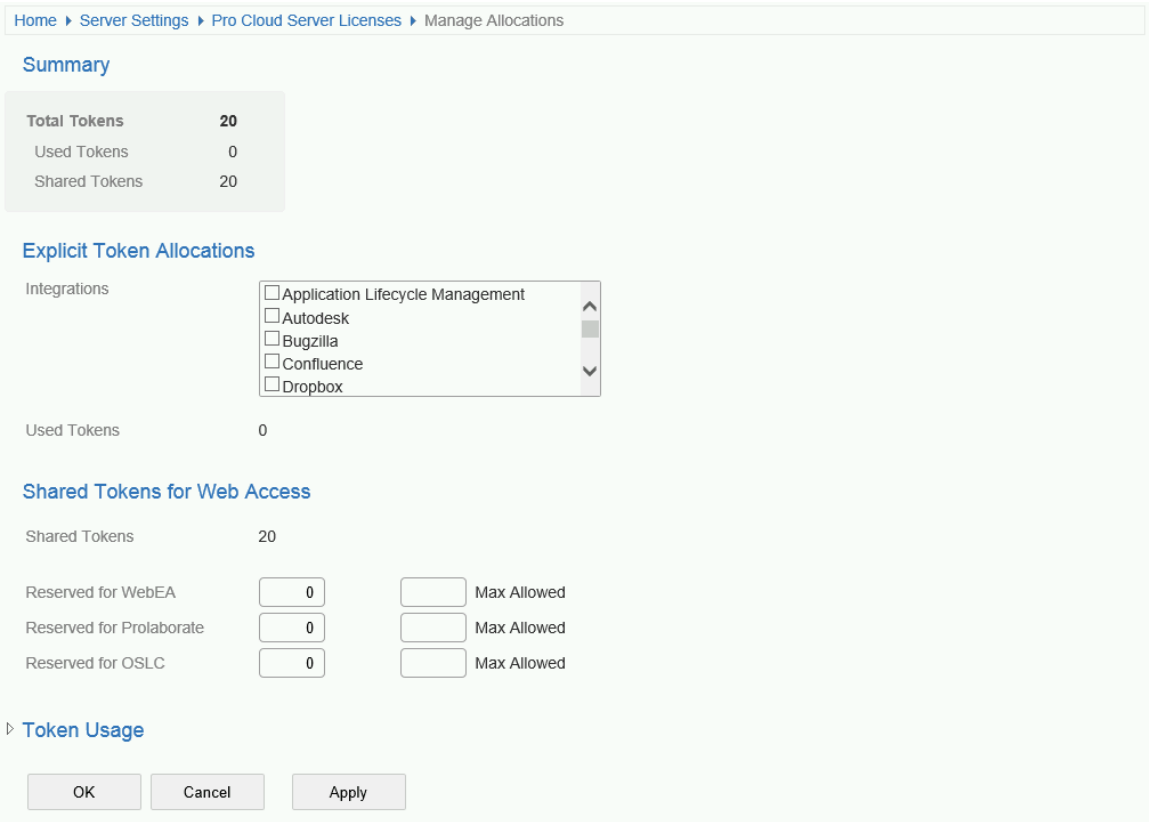
On the WebConfig - Pro Cloud Server Licenses screen, the 'Actions' column for each existing license contains a 'Renew License' icon and a 'Delete License' icon.

The 'Renew License' icon is a convenient function that displays the same License Request form that the Create Request button shows; however, the difference is that the 'Renew' option will pre-populate the valid details of the selected License. The remaining fields will still need to be completed manually. The renewal process is the same as the initial process and will need to be followed whenever the License expires.

WebConfig - Token Allocations

The Manage Allocation button will only be visible when the current license is the Token edition. It allows the user to control how tokens are consumed to enable Pro Feature functionality.

Click on the Allocate Tokens button to access the 'Manage Allocations' screen.



The 'Manage Allocations' screen is broken up into four sections - a Summary of tokens, Explicit Allocations, Web Access Allocations, and Token Usage.

Summary

The token summary lists:

- Total - The total number of tokens based on the active license/s
- Explicit - The number of explicit token allocations; these refer to tokens that have been allocated to enable specific features on the server
- Shared - The number of shared tokens available; shared tokens are consumed when users access features such as WebEA - when logging out the token is returned to the shared token pool

Explicit Allocations

This section allows you to enable Integration Plug-ins for your Pro Cloud Server. Enabling integration with a specific product (such as Jira) consumes one token.

Shared for Web Access

WebEA, Prolaborate and OSLC users consume a single token with every logon. For most environments, these allocations can be left with their default values (0 reserved and no maximum). In this case all users are accessing a single shared pool of tokens, regardless of what type of access (WebEA, Prolaborate or OSLC) they require. Alternatively, each user type can define a number of tokens that are reserved (that is, can only be used by that type of access) as well as a maximum number.

Notes:

- A **Max Allowed** value of 0 implies that users are NOT allowed; that is, if WebEA's **Max Allowed** value was set to 0, the Pro Cloud Server would deny all WebEA user

requests to logon

- An empty value for **Max Allowed** implies there is no limit on the number of users, although the total number of users is still limited to the number of available tokens
- Using WebEA's 'Logout' option will immediately return the token; however, if the web browser is closed without using 'Logout' then the token will be returned after 30 minutes of inactivity

Token Usage

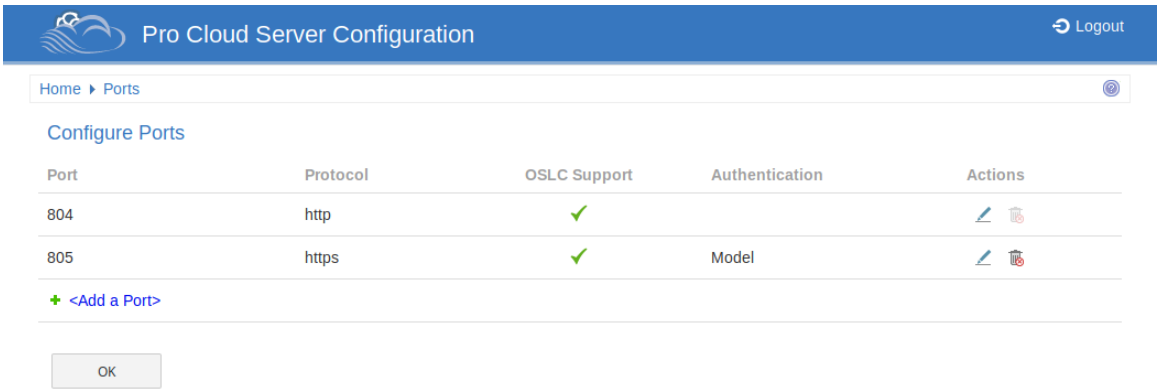
Click the heading to expand this section. The 'Token Usage' section displays a more detailed break down of the token allocations, including how many tokens are currently in use.

WebConfig - Configure Ports

The 'Ports' page allows you to configure Ports that can be used for communication with the Pro Cloud Server. A Port must be configured to allow access to Cloud Models (via Enterprise Architect or WebEA).

By default, Port 804 is defined for http access and Port 805 for https.

The WebConfig client itself also communicates on one of these defined Ports (as specified in the WebConfig 'settings.php' file). The WebConfig's default configuration uses Port 804 (via http).



Configure Ports

The 'Configure Ports' list displays these details and links to corresponding information.

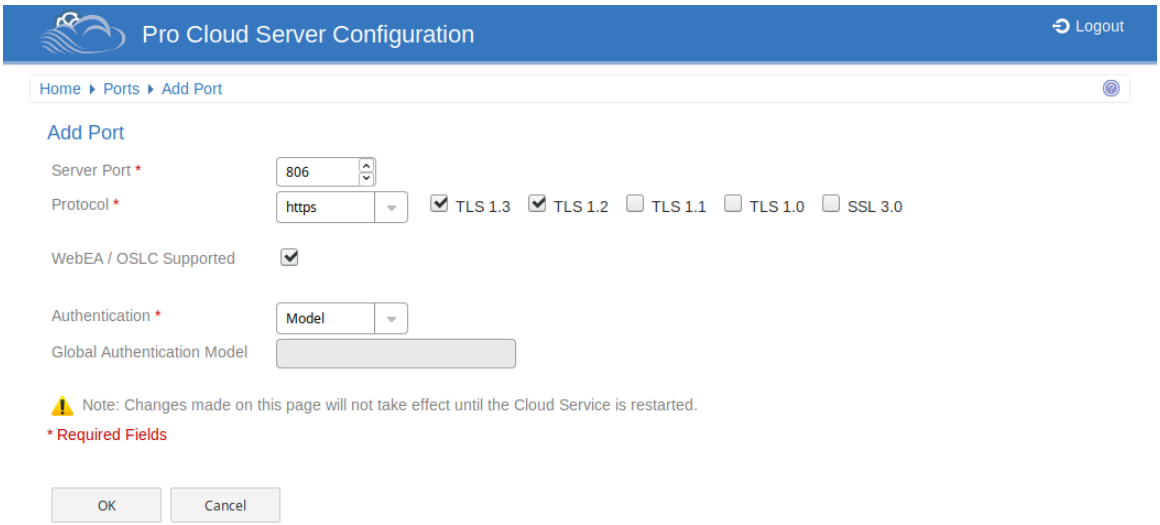
Item	Description
Port	The Port number.

Protocol	The Protocol that can be used on this Port.
OSLC Support	Indicates whether OSLC / WebEA support is enabled.
Authentication	Indicates what type of Authentication is required (if any).
Actions	Provides options to either Edit or Remove the Port definition. Editing a Port provides the same options as adding a new Port, except that the Port number cannot be modified. Note, the WebConfig itself uses one of the defined Ports for communication with the Cloud Server (as defined in the WebConfig 'settings.php' file). It is not possible to delete this Port while it is in use. If necessary you can modify the 'settings.php' file so that the WebConfig uses a different Port. You can then delete the Port that the WebConfig was previously using.
Add a Port	The final item in the Port list is a link to add a Port.

WebConfig - Add or Edit a Port

The 'Add Port' page allows you to define a Port to be used by the Pro Cloud Server.

The 'Edit Port' page provides the same options, except that the Port number cannot be modified.



Add Port / Edit Port

The 'Add Port' and 'Edit Port' pages provide these options.

Item	Description
Server Port	Defines the Port number that the Pro Cloud Server will listen on for HTTP/HTTPS connections. Each defined Port must be unique and not used by any other service or application on the current machine. The standard HTTP Port (80) and HTTPS Port (443) are no different; if they are available, the

	Pro Cloud Server can be configured to use them. You can use the Windows Resource Monitor (which is accessibly via an icon on the Configuration Client 'Ports' tab) to check which Ports are already in use. **Firewall Note**: When changing or adding a new Port, check that incoming requests are allowed by the firewall settings at both the local machine and network levels. **Wine Note**: 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.
Protocol	Select the transfer protocol that the Port communicates on. When selecting HTTPS you can also specify which versions of SSL/TLS are allowed, by selecting the relevant checkboxes. Important: For HTTPS to be configured correctly, a file named *server.pem* must also exist in the same directory as the configuration file, and it must contain a valid private key/certificate. See the Using HTTPS (SSL) Help topic for

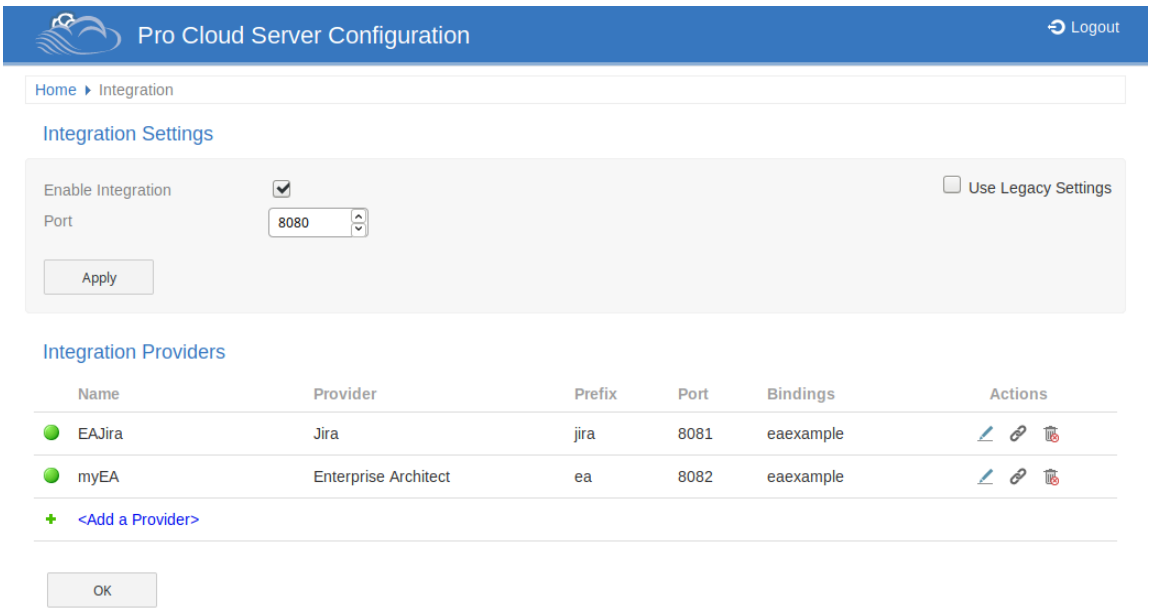
	details.
WebEA / OSLC Supported	If checked, this flag enables OSLC requests on the current Port. Note: this option must be selected for WebEA access.
Authentication	Select the type of authentication that will be required for this Port: **Model** - All model connections that use the current Port will be forced 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: If you are not using HTTPS/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. **Global -** Selecting this option enables the 'Global Authentication Model' field. In this field enter the DB Alias of Data

	Manager that will be used to provide the security profile to all connections opened with the current Port. This is helpful if you want to provide multiple repositories but only manage one list of users. Passwords must be explicitly and individually assigned in the Global model using the *Maintain Users* procedure; the default administrator password and any passwords imported from Windows Active Directory will not work. Note: If you are not using HTTPS/SSL to connect, the usernames and passwords will be sent in plain text. Sparx Systems recommends that Global authentication should only be used with the HTTPS protocol.
Global Authentication Model	Required when the 'Authentication' option is set to 'Global'.

WebConfig - Integration Plug-ins

The 'Integration Settings' page allows you to configure integration Plug-ins to be used with Pro Cloud models. These Plug-ins enable data from external providers to be displayed within Enterprise Architect. This component is only available in the licensed editions of Sparx System Pro Cloud Server; that is, Token, Team and Enterprise editions.

The Integration Plug-ins component is also known as the Server Based Plug-in Interface (SBPI).



Integration Settings

The 'Integration Settings' section refers to the configuration of the Pro Cloud's Integration Server. The Integration Server communicates with one or more Integration Plug-ins, each of which provides the integration with a specific product such as Jira or Confluence.

Typically, if the Integration Server is run on the same machine as the Pro Cloud Server, and Port 8080 is not in use, then the default settings can be used.

To make the Integration Server configuration as easy as possible for the majority of users, version 4.1 of the Pro Cloud Server has introduce a simplified set of options, however if the default configuration does not suit your environment, check the 'Use Legacy Settings' option to show all the available options.

Item	Description
Enable Integration	This checkbox enables (or disables) Integration Plug-ins for the current Pro Cloud Server installation.
Port	Default value: 8080 This option is only shown when the 'Use Legacy Settings' option is unchecked. It defines the Port the Integration Server is listening to for requests from the Pro Cloud Server.
SBPI Server URL	This option is only shown when the 'Use Legacy Settings' option is checked. The SBPI Server URL consists of three components. When combined these settings form the Integration Server's URL that the Pro Cloud Server will send

	SBPI related requests to. Note: The complete URL ({protocol}://{server-name}:{port} needs to be resolvable by the Pro Cloud Server machine. • **Protocol** This field defines the SBPI Server's Protocol. • **Name/IP** This field defines the SBPI Server's machine name or IP. • **Port** This field defines the SBPI Server's Port.
Use Legacy Settings	Default value: Unchecked This checkbox determines if the simplified or complex set of options should be shown. The name refers to the fact that the earlier versions of the Pro Cloud Server required users to fill-in the complex set of options. In later versions of the Pro Cloud Server the set of options needed was greatly reduced by making some assumptions that will be true for the majority of environments.
Ignore SSL Errors	Default value: unchecked This option is only shown when the 'Use

	Legacy Settings' option is checked. This value defines if SSL related errors that occur while communicating with the Integration Server component should be ignored. The most common SSL errors are related to self-signed certificates; by default a client does not treat self-signed certificates as being secure, therefore in order to connect to servers that use self-signed certificates this option should be checked.
Use Proxy	This value defines if the Integration Server is operating behind a proxy, and if Enterprise Architect clients are required to connect to one URL but the Integration Server is physically listening for requests on a completely different URL. In most environments this option should be set to the default value of unchecked.
Attempt URL Auto Discovery	Default value: checked This option is only shown when the 'Use Legacy Settings' option is checked. This option defines if the Pro Cloud Server should automatically attempt to determine the Enterprise Architect client's network address and supply that to the Integration Server.

	This option was introduced in Pro Cloud Server 4.1.40; earlier versions were not capable of discovering the client's network address. When 'Attempt URL Auto Discovery' is checked, the setting is the 'Fallback URL', which will be used only if the Auto Discovery fails. When 'Attempt Auto Discovery' is unchecked, the setting is the 'Absolute URL' which will be used in all cases.
Absolute URL	The Absolute URL consists of three components. When combined these form the resolvable URL that Enterprise Architect clients can communicate to the SBPI server. Note: The complete URL ({protocol}://{server-name}:{port} needs to be resolvable by the Pro Cloud Server machine. In some environments server URLs are redirected to completely different locations, therefore to handle this situation the absolute URL is returned to Enterprise Architect clients so that they can communicate to the SBPI Server. • **Protocol** This field defines the URL's Protocol.

	• **Name/IP** This field defines the URL's machine name or IP. • **Port** This field defines the URL's Port.

Integration Providers

A single Pro Cloud Server can support any number of external Data Providers and, in turn, each of the external Data Providers can be available to as many models as are supported by the installation's license. Please note only Pro-enabled Model Connections can access external data providers. A Pro-enabled Model Connection is one that has the 'Enable Pro Features (OSLC, WebEA and Integration)' option checked.

The 'Integration Providers' section provides this information:

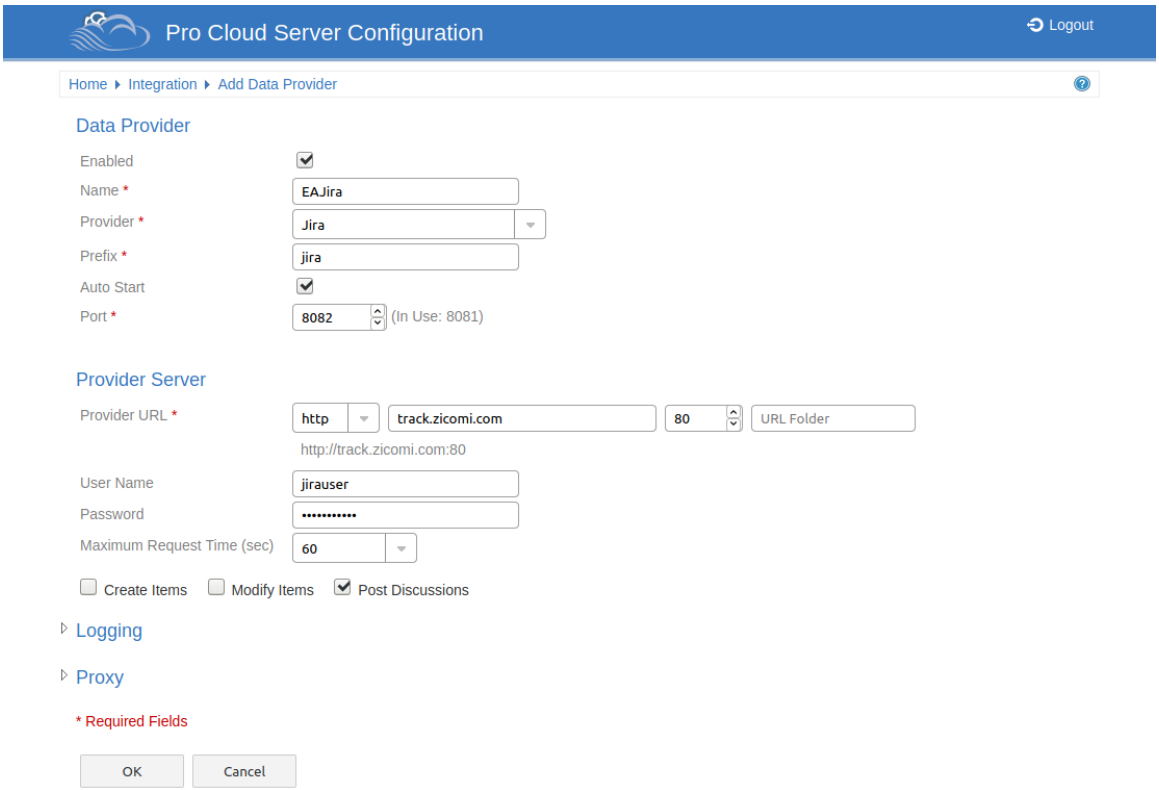
Item	Description
Name	A friendly name that describes the provider; this value will be displayed within Enterprise Architect.
Provider	The type of external provider.

Prefix	A short, unique value for the provider. This value is added to each external link within Enterprise Architect.
Port	The Port the Integration Plug-in is configured to listen to.
Bindings	Bindings refer to the model connections that will have access to this Integration Plug-in. These can be configured via the 'Actions' column.
Actions	Provides options to either: • Edit the Integration Provider's settings • Configure Model Bindings, or • Delete the Integration Provider
Add a Provider	The final item is a link to add a new provider.

WebConfig - Add or Edit an Integration Data Provider

The 'Add Data Provider' and 'Edit Data Provider' screens allow you to configure an Integration Data Provider. The options provided are the same in both cases, except that when editing you cannot modify the Provider and Prefix.

Wherever possible the Add/Edit Data Provider screen will fill each field with a default value in order to make it easier for you; if the default values are not correct, simply overwrite them.



The Integration settings are broken up into a number of groupings.

Data Provider

This group of settings contains the basic information to identify the Data Provider and determine how the Plug-in communicates with the Provider Server.

Item	Description
Enabled	This value controls if the Integration Server (SBPI.exe) should allow communications to be forwarded to this Data Provider.
Name	This value represents the friendly name for the current Data Provider, it will be shown to users of Enterprise Architect.
Provider	This value specifies the Provider type of the current Data Provider. When this value is changed a number of other fields could be automatically updated.
Group	This field is available only when the 'Custom Integration' provider type is selected. Typically in Enterprise Architect, integrations are grouped by the Provider type (such as 'Jira' or 'Dropbox'). For Custom Integrations, the Grouping is applied based on the value entered in this field.

Prefix	This value represents a short unique name to identify the current Data Provider. This value will be saved against every Enterprise Architect element that is linked to an external element.
Auto Start	This value controls if the Integration Server (SBPI.exe) should maintain a running process (*sbpi.exe) for this Data Provider. For standard Integration Plug-ins we would expect this value to be checked; however, for custom ones it might be useful to not auto-start the Plug-in and instead manually control when it is running, particularly while developing it.
Port	This value represents the Port that the Integration Server should use to communicate with the Integration Plug-in. The Integration Server assumes the Port is relative to the current machine (or http://localhost:{port}). Each Integration Plug-in requires a unique Port number, which will be checked when an attempt is made to save the Plug-in's settings. However the Pro Cloud Server can only check what Ports

	are in use by the Pro Cloud Server itself and its components, there is still a possibility of a Port clash with other applications.
DLL Path	This field is only available for 'Custom Service' and 'Custom Integration' provider types. This value represents the physical path to the custom DLL, that will be called by the CustomService.exe or CustomIntegration.exe Data Provider processes.

Custom Properties

The Custom Properties section is only shown for Data Providers of type 'Custom Service'. This table allows you to define Item/Value pairs that can be used by the Custom Service.

Item	Description
Item	The 'name' of the custom property.
Value	The value of the custom property.
Actions	This column provides Action buttons

	which apply to the property in this row. **Accept New Custom Property** - This appears when adding or editing a property. Click this button to save the Item/Value. **Edit Custom Property** - Click this button to allow editing of the property's item/value. **Delete Custom Property -** Click this button to delete the property.

Provider Server

This group of settings relates to the details of the external data provider, such as Dropbox or Jira. The Integration Plug-in will use the defined details to connect to the remote system and retrieve data from it so that it can be sent back to Enterprise Architect. The combination of Protocol, Server Name and Port must be resolvable by the server hosting the Integration Plug-in, which in the simplest case is the Pro Cloud Server.

Item	Description
Provider URL	The Provider URL consists of these four fields, which make up the URL that the Plug-in will use to access the external data.

	• **Protocol** - The Protocol to be used • **Server Name/IP** - The Server Name or IP address • **Port** - The Port to be used; if the data is typically accessed using a URL without a Port, then enter the default Port value (80 for HTTP or 443 for HTTPS) • **URL Folder** - Some external Data Providers allow for the concept of multiple repositories at a single location; for such Data Providers this field acts as a filter For example, it is possible to use another Enterprise Architect model as an external data source; in this case the 'Protocol', 'Server' and 'Port' determine the Pro Cloud Server (for example, http://myEAServer:804) while the 'URL Folder' is the DB Alias of the repository (myModel). Thus: http://myEAServer:804/myModel Below these fields is a label showing the full URL that will be used by the Plug-in (based on the values that have been entered).
Ignore SSL Errors	This option is only displayed if the protocol 'HTTPS' has been selected.

	This checkbox controls if the Integration Plug-in should ignore SSL related errors that occur as a result of communicating with the external system.
Authentication	Default Value: Credentials This value allows the selection of the type of Authentication that should be used for the Provider, the valid options are Credentials and OAuth. When 'Credentials' is selected it would be expected that a User Name and Password would be entered. When 'OAuth' is selected and additional five fields are displayed (Client ID, Client Secret, etc). For details on using these fields, refer to the Data Provider Advanced Authentication topic (this describes the equivalent fields in the PCS Configuration Application). **Note 1**: this option is only available if the selected Provider supports multiple forms of authentication, for example: Dropbox, Jira, Salesforce, ServiceNow, SharePoint and Wrike **Note 2**: this option was introduced in Pro Cloud Server v5.0 build 100, prior releases only support credential based authentication.

Username	This value represents the user name that should be used (in combination with the password) to access the data within the external system. If a value is defined in this field all Enterprise Architect users of the current Data Provider will use the same set of credentials to read the external system. Otherwise if the user name field is left empty, Enterprise Architect will prompt each user independently for their credentials.
Password	This value represents the matching password for the specified user name.
Create Items	This checkbox controls if Enterprise Architect users are able to create new items within the external system.
Modify Items	This checkbox controls if Enterprise Architect users are able to modify the details of external items within the external system.
Post Discussions	This checkbox controls if Enterprise Architect users are able to create discussions against elements within the external system.

Maximum Request Time (sec)	This value represents the maximum amount of time (in seconds) that calls to the Provider system will wait before timing out. Increasing this value is particularly useful when the Provider systems are remote or have slow performance. The default value is 60 seconds. Note: This is a new setting introduced in Pro Cloud Server v4.2 build 65.

Logging

This group of settings relates to how the Integration Plug-in will create entries in its log file(s). Since the Integration Plug-ins run without user intervention, it is important for them to be able to write any messages or potential problems to a log file. However, special consideration must be given to ensuring that performance isn't impacted by writing to such files, which is possible if the log file continually grows without limits. It is also unreasonable to expect that manual intervention should be required to ensure that the size and number of log files don't impact performance. For these reasons each Integration Plug-in has its own set of log files and configuration options to manage its log files.

The current log file will always include a '1' on the end of its filename. A new log file will be created whenever the

Integration Plug-in is started or the physical size of the current log file reaches the Max File Size. When this occurs all existing files will be 'rolled over', meaning their file numbers will be incremented and a new '{filename}1.log' created. If the roll over process causes more files to exist than the 'File Count' setting allows, these additional files will be deleted.

Item	Description
File Count	Default value: 3 This value represents the 'rolling' number of log files that should be retained for the current Integration Plug-in.
Max File Size	Default value: 1048576 This value represents the maximum number of bytes a log file can be before a new one is created.
Level	Default value: WARNING This value represents the maximum level of messages that should be written to the log file. The levels are: OFF, FATAL, WARNING, INFO and SYSTEM. A defined log level is inclusive of all lower levels, therefore if a Plug-in is set to a log level of INFO then all FATAL, WARNING and INFO messages will be

	written to its log file.
Directory	(READ-ONLY) This field displays the physically location of where the log files for the current Data Provider will be saved.

Proxy

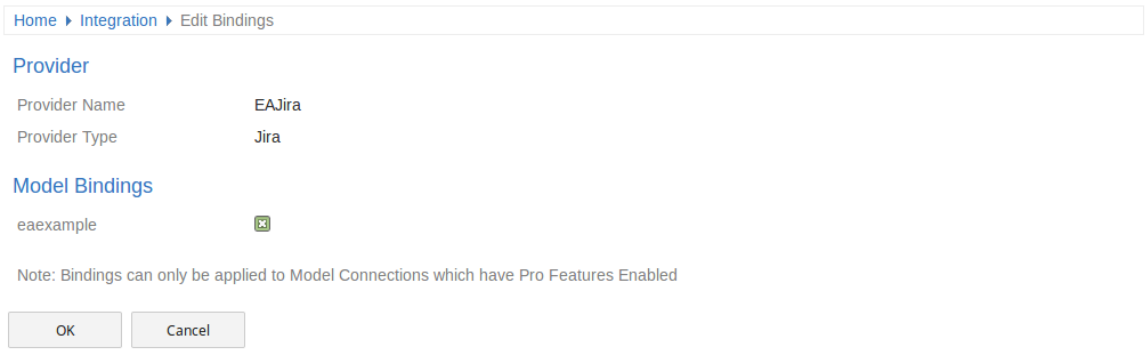
This group of settings relate to how the Integration Plug-in should communicate with the External Data Provider when they are separated by a Proxy Server.

Item	Description
Server Name/IP	The server name (or IP) and Port number of the Proxy Server. For example, proxyserver.com:3131
Bypass	This field provides a mechanism for certain addresses to bypass the proxy; it accepts multiple values separated by semi-colons (;). For example, 192.168.*;*.localdomain.com;www.mydomain.com
User Name	If the Proxy Server requires credentials, then enter the user name into this field.

Password	If the Proxy Server requires credentials, then enter the password into this field.

WebConfig - Edit Model Bindings

After defining an integration provider you must apply some model bindings. The integration will only be available to these 'bound' models. Also note, integrations can only be bound to models that have the 'Enable Pro Features (OSLC, WebEA and Integration)' option enabled.



The 'Edit Bindings' page provides this information and these options.

Provider

Item	Description
Provider Name	(Read Only) The Name of the Provider which you are editing the bindings of.
Provider Type	(Read Only) The Provider type (Enterprise Architect, Jira, etc).

Model Bindings

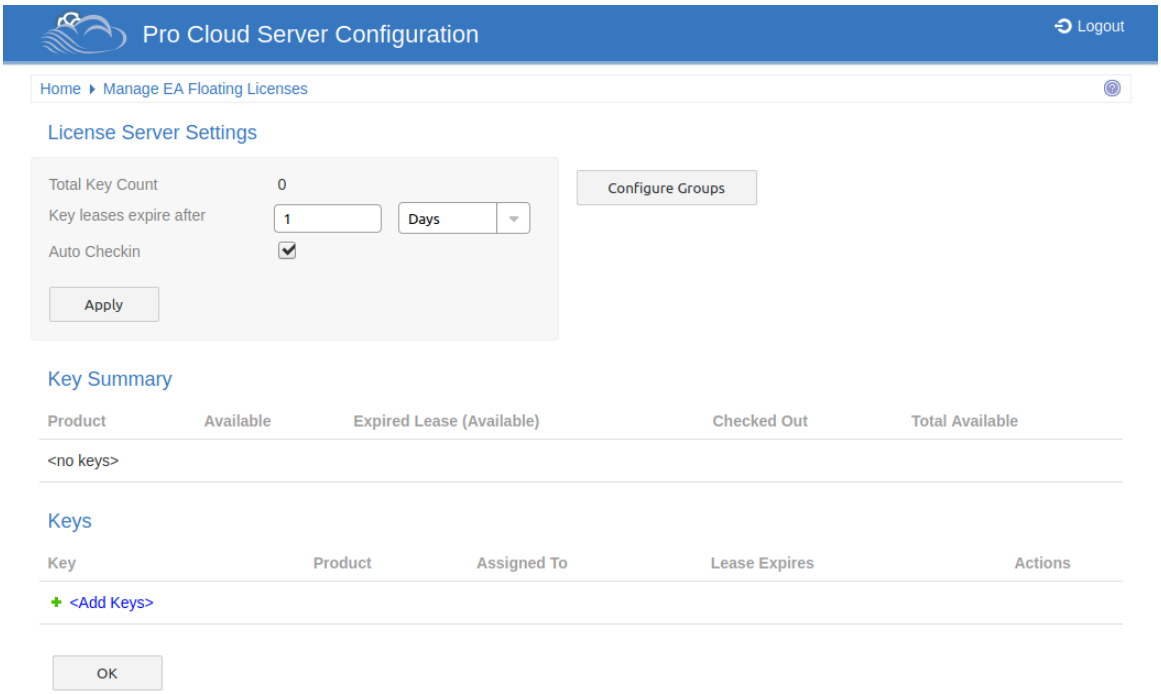
The Model Bindings section lists all model connections which currently have pro features enabled. Next to each model connection name is a checkbox. Check this box to bind the integration provider with that model connection (i.e. making the integration available when using that model).

Note, one Provider can be bound to many model connections.

WebConfig - Manage EA Floating Licenses

The 'Manage EA Floating Licenses' page allows you to manage Enterprise Architect (and Add-in) floating licenses.

Important: All communications with the Pro Cloud's Floating License Server are performed via HTTPS. Therefore, to use the Pro Cloud Server's Floating License Server functionality at least one Port has to be configured to use the HTTPS protocol (with a valid SSL certificate). See the 'Add a Port Definition' topic for details.



The 'Manage EA Floating Licenses' page has three sections, as described in this topic.

License Server Settings

The 'License Server Settings' section allows you to adjust

the settings for handling Floating Licenses.

After adjusting these settings, save and apply them by clicking on the Apply button.

Item	Description
Total Key Count	(Read Only) Displays the total number of keys in the keystore.
Key leases expire after	These two fields determine when a checked out key is automatically returned to the keystore. Enter a numerical value, then select either 'Days' or 'Weeks' from the drop-down list.
Auto Checkin	When enabled, if a user closes Enterprise Architect then their key will be immediately returned to the keystore. When disabled, closing Enterprise Architect will not automatically return the key. Instead, the key remains checked out to the user for the amount of time defined by the 'Key leases expire after' field. Typically the 'Auto Checkin' option would be disabled if users need to work offline for periods of time. Otherwise, the option should be enabled to make the most efficient use of shared keys.

Group Configuration

The Cloud Server allows you to define groups that only have access to a set number of keys for a specific product. To define and modify groups click on the **Configure Groups** button.

See the WebConfig - Floating License Groups Help topic for details.

Key Summary

The 'Key Summary' table on the 'License Server Settings' page provides a quick overview of the keys in the keystore for each product type.

Item	Description
Product	The name of the product.
Available	The number of checked in keys for the product.
Expired Lease (Available)	The number of keys that are checked out but are available because their lease has expired.
Checked Out	The number of keys that are currently checked out.

Total Available	The total number of keys that are currently available for the product (that is, the sum of the 'Available' and 'Expired Lease' values).

Keys

The 'Keys' table lists information on each key held in the keystore. Each item line on the table provides options to force the check-in of the key, delete the key, and add new keys.

Checked out keys are displayed at the top of the list, and keys are grouped by product type.

Item	Description
Key	The product Floating License key.
Product	The name of the product that the key provides access to.
Assigned to	The name of the user who currently has the key checked out. When the field is empty it indicates that the key is checked in.
Lease	The date when the key will be

Expires	automatically returned to the keystore (based on the 'Key leases expire after' field settings).
Actions	Provides the options to manually check in or delete a key: • **Check In Key** Only available if the key is currently checked out; select this option to manually release or check-in the key • **Delete** Deletes the key from the keystore
Add Keys	The final item in the 'Keys' list is the 'Add Keys' option. Click on this option to display the 'Add Floating Licenses' page, which contains a single field for adding keys to the keystore. To add a key, type it into the 'Enter Floating License/s' field, then click on the OK button. You can type multiple keys in the field; each key should be typed on a separate line.

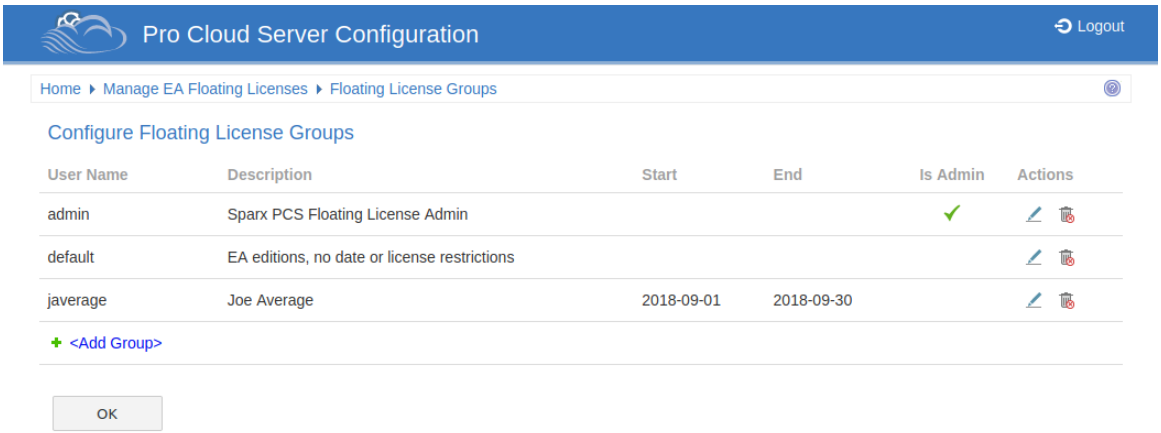
WebConfig - Floating License Groups

In order for users to 'check-out' a floating license they must be a member of a Floating License Server Group. The definition of a Group controls which Product Licenses members of the group can 'check-out' and the period of time that the group is valid.

The Floating License Groups configuration can be accessed via the Configure Groups button on the 'Manage EA Floating Licenses' page.

See the WebConfig - Manage EA Floating Licenses Help topic for details.

The default Group configuration includes the definition of a number of Groups:



The first group shown in the example is 'admin'; each Floating License Server installation will require at least one administration group. For the purpose of allowing connections from the Floating License Server Configuration Client, admin groups cannot be assigned Products and so

therefore can not be used by Enterprise Architect to obtain a license.

'Joe Average' is an example of an expired group that can no longer be used to obtain a license. Any attempt to use this will result in a error message indicating that the specified group is not longer valid.

Note: Floating License Group Configuration was introduced in PCS version 4.2. The WebConfig Client in combination with the Pro Cloud Server can now manage the configuration of Floating License Server Groups; prior to this the Group file (ssflsgroups.config) had to be manually edited using a text editor.

Configure License Groups

The Configure Floating License Groups list provides the information and functions shown here.

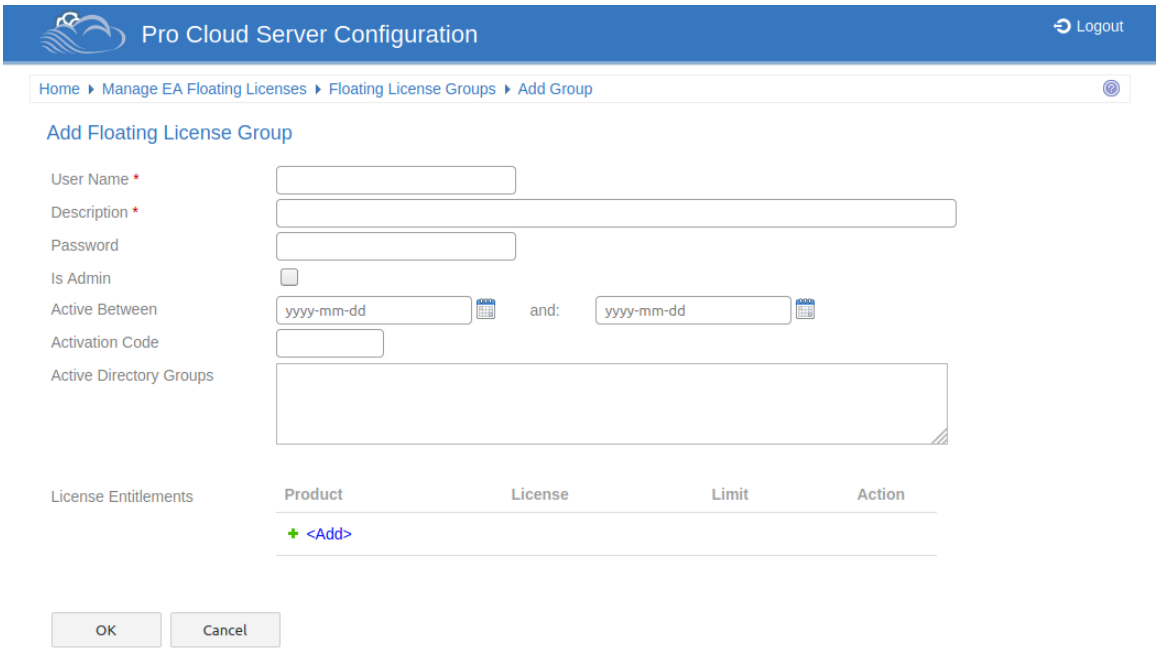
User Name	The unique user name of the group.
Description	A meaningful description for the group.
Start	The start date for when the group will be active. If this column is empty then there are no date limitations on the group.
End	The end date for when the group will be active. If this column is empty then there

	are no date limitations on the group.
Is Admin	A green tick will be displayed if this group is defined as an Administrator.
Actions	Edit - Click on this button to edit the group. Delete - Click on this button to delete the group.
<Add Group>	Click on this link to Add a new group.

WebConfig - Add or Edit a Group

The 'Add Floating License Group' page and 'Edit Floating License Group' page allow you to either add or edit an existing Floating License Group. The options provided are the same in both cases; however, when editing a group the fields will be populated with the group's existing settings.

These pages are accessible via the **Edit** Action and **<Add Group>** link on the 'Configure Floating License Groups' page. See the WebConfig - Floating License Groups Help topic for details.



Add / Edit Floating License Group

The 'Add Floating License Group' page and 'Edit Floating License Group' page provide these fields and options.

Option	Description
User Name	The unique user name a group is known by. When an Enterprise Architect client connects to the Floating License Server, it is this value that must be entered as the 'User Name'.
Description	A meaningful description of the group.
Password	The password for the group. Along with the 'User Name', the password must be entered by Enterprise Architect clients in order to obtain a floating licence. From version 4.2 all groups that are updated will see the password saved to the configuration file as an encrypted value, therefore it is no longer possible to simply view the underlying configuration file to determine a group's password.
Is Admin	When this flag is checked it indicates that the current group is only to be used for administering the Floating License configuration. Note: For most of the fields below 'Is Admin' will be disabled once the flag is checked; for example, 'Activation', 'Active Directory Groups'.

Active Between	These two date fields define the period of time for which the current group is active. Once a group is inactive, users will not be able to connect using this account. Previously it was possible to define a group that was not restricted to a period of time; however, with the use of the configuration client the same is achieved by defining the end date well into the future. An empty start date will be shown as '2010-01-01', while an empty end date will be '2050-12-31'.
Activation Code	When Enterprise Architect is first installed it will prompt the user for an 'Activation Code' that was supplied with the original purchase; this is a simple four-letter code to confirm the user's right to use that version of Enterprise Architect. By assigning the provided Activation code in this field all Enterprise Architect users will derive the Activation Code along with their floating license from the Floating License Server, thereby removing the need for each client machine to be updated whenever new licenses are purchased. Note: The ability for Enterprise Architect

	to obtain an Activation Code from the Floating License Server was added in Enterprise Architect v15.2 in combination with Pro Cloud Server v4.2, therefore earlier versions are not able to take advantage of this feature.
Active Directory Groups	This field is used to associate one or more Active Directory groups to the current group. Each Active Directory group must be specified by its fully qualified name. If you want to define multiple Active Directory groups, simply separate the names by commas; i.e. a CSV of fully qualified Active Directory group names. "WinNT://MYDOMAIN/Sparx Users,WinNT://MYDOMAIN/EA Users,WinNT://CUSTOMERDOMAIN/Sparx Users"
License Entitlements	Displays a list of License Entitlements for the current group. A License Entitlement consists of a Product, License Type and Limit. License Entitlements are uniquely identified by a combination of Product and License Type, which means a Group can be defined with both a Full and Academic license type for a single product.

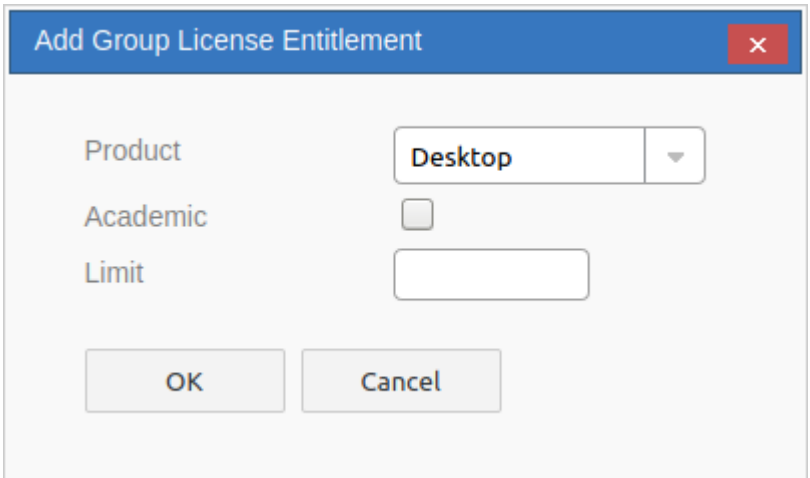
	Click on the **<Add>** link to add a new entitlement to the list. Click on the **Edit** Action button to edit an existing entitlement / row. Click on the **Delete** Action button to delete an existing entitlement / row. See the *WebConfig - Group Entitlements* Help topic for more information.

WebConfig - Group Entitlements

Managing the definition of a License Entitlement is performed using the 'Add Group License Entitlement' / 'Edit Group License Entitlement' dialogs, which will be shown when either the Add or Edit buttons are selected on the WebConfig - Add or Edit a Group.

License Entitlements grants users of the current group the ability to 'check-out' floating licenses for the defined Product and License Type combination.

The 'Add Group License Entitlement' dialog is shown below. The 'Edit Group License Entitlement' dialog provides the same options, however the relevant existing values will be selected.



Note, you can only define one entitlement for each Product / Type (Academic or Full) combination.

E.g. You can add one 'Professional - Full' license entitlement and one 'Professional - Academic' license entitlement.

Add / Edit Group License Entitlement

These options are available for each Group Entitlement.

Product	The name of product (Enterprise Architect edition or Plug-n) that this entitlement gives access to for users of the current group.
Academic	Determines if the current entitlement implies that academic licenses of the selected product are allowed for the current group.
Limit	A optional numeric value representing the maximum of con-current users of the current group that can check-out License keys for the defined Product/License Type. The default is an empty value and represents that the definition imposes no limit to the number of license that can be checked out by the current group.

	This option if particularly useful if departments within a single company purchase their own floating licenses, groups can be setup for each department and a limit applied to restrict the group to the number of licenses that their department purchased.

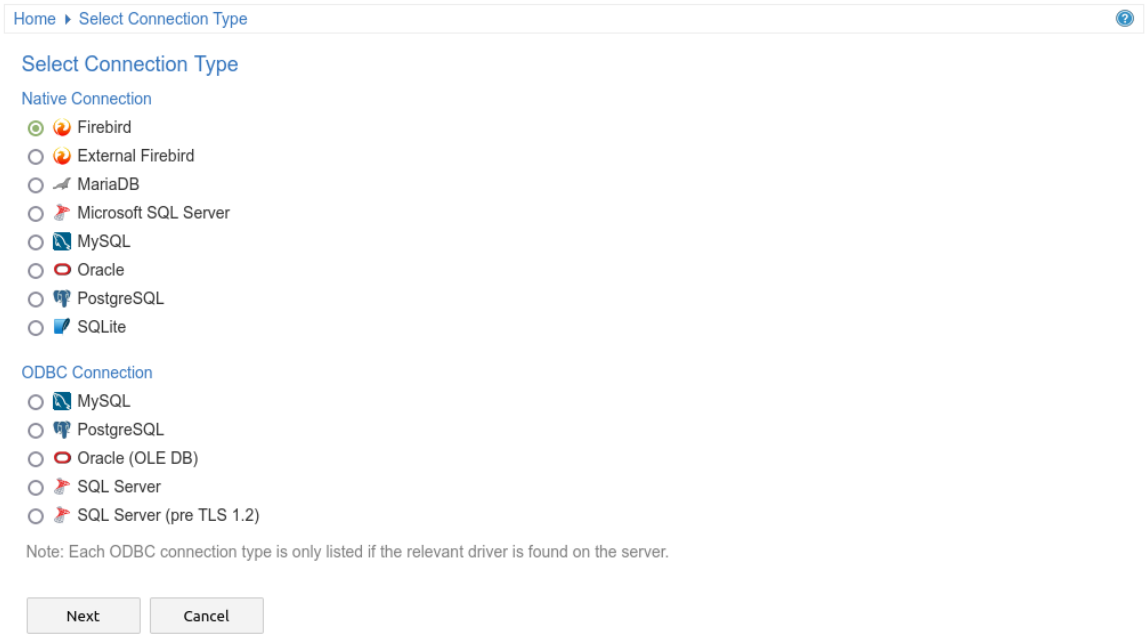
WebConfig - Adding a Model Connection

New model connections can be added using the 'Add a Connection' option at the foot of the WebConfig 'Home' page.

The process of adding a new model connection has two steps - selecting the connection type, then entering the details that are required for the model connection.

Select Connection Type

When you click on the 'Add a Connection' option the 'Select Connection Type' page displays.



This page is broken up into two lists of database connection types:

Native Connection - Refers to database connections using

the Pro Cloud Server's native drivers. This is the simplest and recommended method for connecting to repositories.

ODBC Connection - Refers to the original (pre version 5.0) method of defining ODBC and OLE/DB database connections. With the introduction of a 64 bit version of the Pro Cloud Server, additional ODBC/OLE DB drivers (and/or ODBC DSNs) will most likely be required in order for the Pro Cloud Server to connect to such databases, therefore Sparx Systems recommend Native connection due to their simplicity.

Select the relevant radio button for the connection type, depending on the type of database you are accessing.

In the case of Firebird or SQLite you have the option of creating the model directly from the WebConfig. For other model types (DBMS repositories), the model should have been created and tested (by accessing the model via Enterprise Architect's 'Connect to Server' option) prior to adding the Cloud Connection.

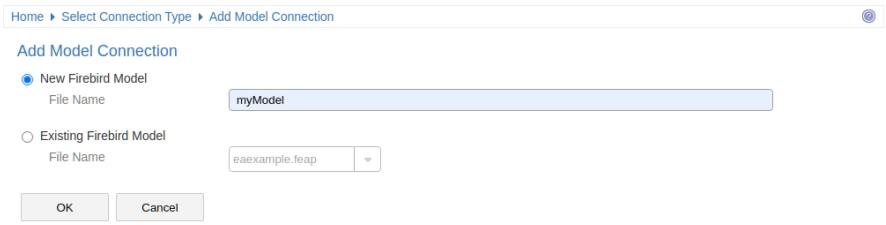
Note:

- The DBMS types under 'ODBC Connection' are only listed if the relevant drivers are found on the server.

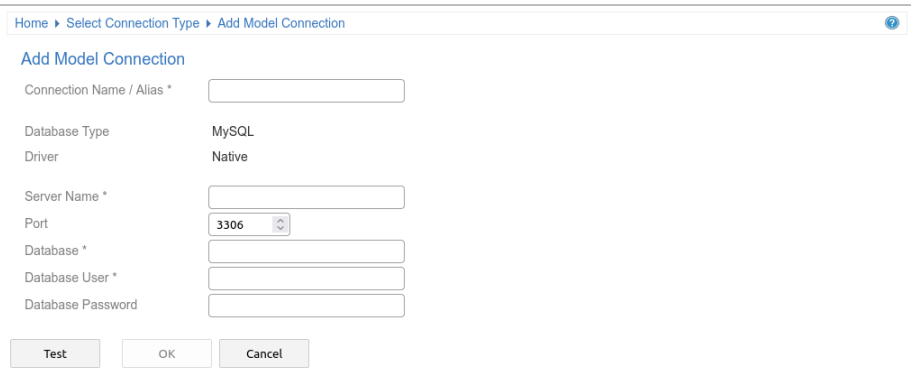
Add Model Connection

After you select the model type, the 'Add Model Connection' page is displayed. The options provided vary depending on the model connection type you selected.

Item	Description

Firebird	Firebird models are stored on the server within the Pro Cloud Server's Models sub-folder (e.g. C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service\Models). When selecting Firebird you are provided with two options.  • **New Firebird Model -** Select this option to create a new Firebird model; enter a name for the Firebird model in the 'File Name' field. No path or file extension are required The new, empty Firebird model will be created on the server (in the 'Models' sub-folder) and the model connection will be associated with this file • **Existing Firebird Model -** Select this option to create a model connection that links to an existing Firebird model (which already exists in the 'Models' sub-folder)
DBMS	

(MariaDB, SQL Server, etc)



Connection Name / Alias

Specify a short, unique name for the connection. This is required when accessing the model via Enterprise Architect or WebEA.

Server Name / Database / etc

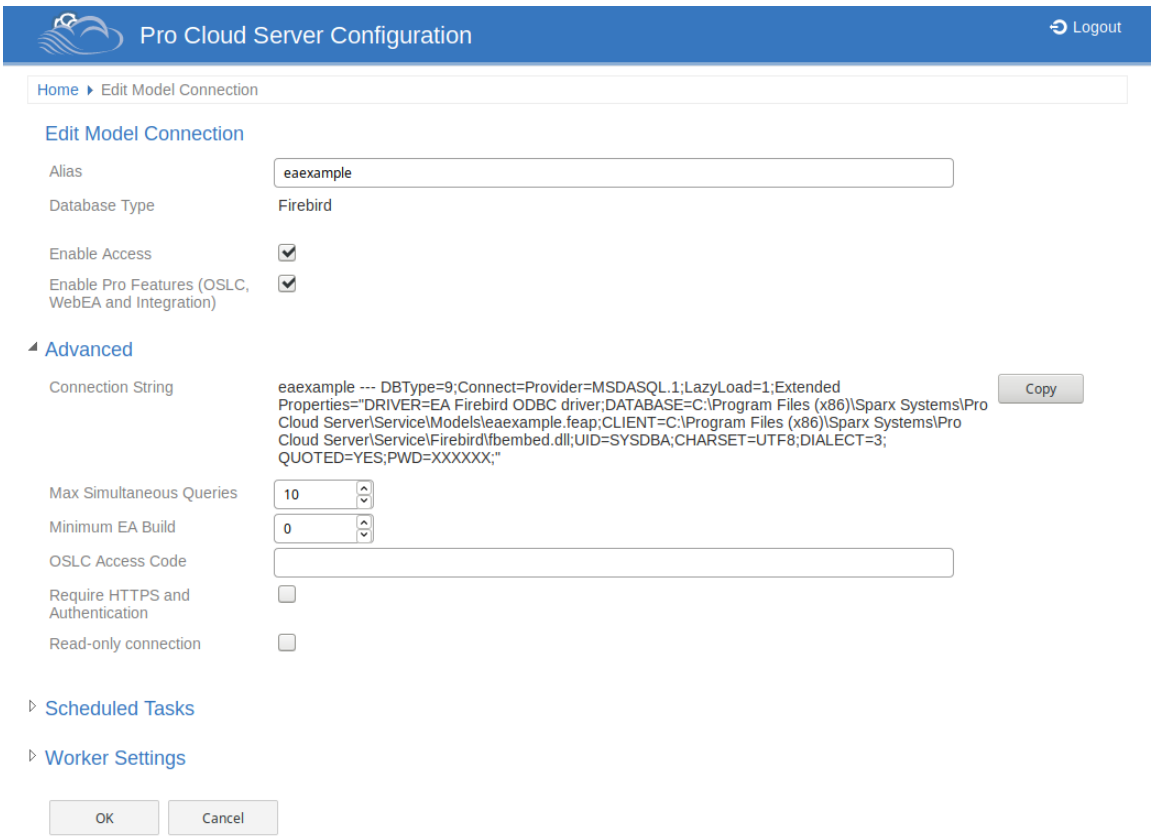
The options provided for DBMS connections (MariaDB, SQL Server, MySQL, etc) vary depending on the model connection type you selected. The available fields are described in the 'Open Server Connections - Native Connection' help topic. Typcially it is best to test access to the database via Enterprise Architect before setting up the Cloud Connection.

Test / OK

When all required fields (*) have been completed, click the Test button. If the connection is successful you will receive a 'Test Successful' message and the OK button will then be enabled. Click OK to add the model connection.

WebConfig - Edit Model Connection

The 'Edit Model Connection' screen displays when you click on the Edit button for an existing model connection on the WebConfig 'Home' page. This page allows you to modify various options for the selected model connection, as described in the tables.



Edit Model Connection

Item	Description
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. Note, this value must be unique. You are not able to use the same alias for multiple model connections.
Database Type	(Read only) Displays the DBMS type of the database, such as Firebird, MySQL, Postgres, SQLServer or Oracle.
Enable Access	By default, new model connections are disabled. Select this checkbox to allow access to the model connection.
Enable Pro Features (OSLC, WebEA and Integration)	This option is only available in Pro Cloud Server editions. Select this checkbox to enable 'Pro Features', such as OSLC, WebEA and Integration with external systems.

Advanced

Connection String	(Read only) Displays the connection string used to connect to the database. The Copy button can be used to copy this connection string to the clipboard.
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	Optional. Type in the access code that all

Access Code	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 HTTPS and Authentication	If this checkbox is selected, this model connection can only be accessed via https, using a Port that is configured to use https with 'Authentication' enabled (that is, set to either 'Model' or 'Global').
Read-only connection	Select this checkbox to prevent users from modifying the database referenced by this connection.

Scheduled Tasks

Start Time (HH:MM)	The time of day that the schedule task should run. This should be entered in 24hr time format (HH:MM).
Number of times to retry	In the event of repeated errors, how many times should the process retry.

Run Scheduled Tasks	When this option is checked the Pro Cloud Server will update Time Series charts in the current model as a scheduled task.
Interval between retries (mins)	Default: 30 Specify how often the 'Worker' process will check for modified diagrams and update their stored images.

Worker Settings

Enable Worker	Select this checkbox to enable the Pro Cloud 'Worker' application. The 'Worker' process runs on the server and periodically generates images for diagrams in the model. These images are required for WebEA and Prolaborate.
Port	Select the desired Port that the Worker should use to connect to this repository. Typically this would match the Port you enter when accessing this Model via Cloud Connection in Enterprise Architect.

Model User	Enter a valid repository user. This value only requires a value if the current repository has model security enabled.
Model Password	Enter the password for the repository user. This value only requires a value if the current repository has model security enabled.
Update Period (seconds)	This value defines how often the Worker process will check for modified diagrams, and generate where required; the default value is 30 seconds.
Logging Level	This value defines the desired level of logging that the Worker should produce. The available logging level options are the same as those for the Pro Cloud Server. The Worker log files will be created within a 'Logs' folder under the Worker's installation location, ie if PCS is installed into the default location this will be: C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service\Worker\logs .

WebConfig - Troubleshooting

If issues occur while using the WebConfig, an error message will be displayed on screen. This topic provides a list of common WebConfig error messages that could occur. Each entry describes why the error occurs and provides guidance on how to correct it.

If you are having issue accessing Pro Cloud Server models from within EA or you are receiving errors in WebEA, refer to the PCS Troubleshooting and Troubleshooting WebEA topics.

WebConfig Error Messages

Issue	Comments
Not authorized to perform admin tasks	*The client is not authorized to perform admin tasks.* This message can appear when attempting to access the WebConfig address in your Web Browser. This indicates that the IP address of the machine that you are using has not been added to the Client White List. To correct this issue, add the IP address to the Client White List. This can be done via WebConfig (on a machine that has

	been white listed), via the Pro Cloud Configuration Client application, or by directly editing the SSProCloud.config file. Refer to the links in the *See also* column for details.
Connection was reset	*Recv failure: Connection was reset* This error can occur when you are attempting to login to the WebConfig. Typically it occurs when you are attempting to use the http protocol to access a Port that has been configured to use https. Check the WebConfig settings.php file. In particular, check that the protocol and Port are correct. These should match one of the Ports defined in the Configuration Client's 'Ports' tab.
Wrong SSL version number	*SSL routines:ssl3_get_record:wrong version number* This error could occur when attempting to login to the WebConfig. It occurs when you are attempting to use the https protocol to access a Port that has been configured to use http. Check the WebConfig settings.php file. In particular, check that the protocol and Port are correct. These should match one

	of the Ports defined in the Configuration Client's 'Ports' tab.
Could not resolve host	*Could not resolve host: <server name>* This error could occur when attempting to login to the WebConfig. Typically this occurs when the Server Name is incorrect. Check the WebConfig settings.php file. In particular, check that the Server Name is correct.
No response from the server	*Request Error: no response from the server* This error could occur when attempting to login to the WebConfig. Typically it occurs if the Pro Cloud Server Service is not running or the Port is incorrect. • Check the Windows Services application on your server and confirm that the 'Sparx Systems Professional Cloud' service is running; see the *Pro Cloud Server Configuration* Help topic for details • Check the WebConfig settings.php file - in particular, check that the Port is correct; it should match one of the Ports defined in the Configuration Client's 'Ports' tab - see the *Pro Cloud*

	Server Configuration Client - Web and *Port Configuration* Help topics for details

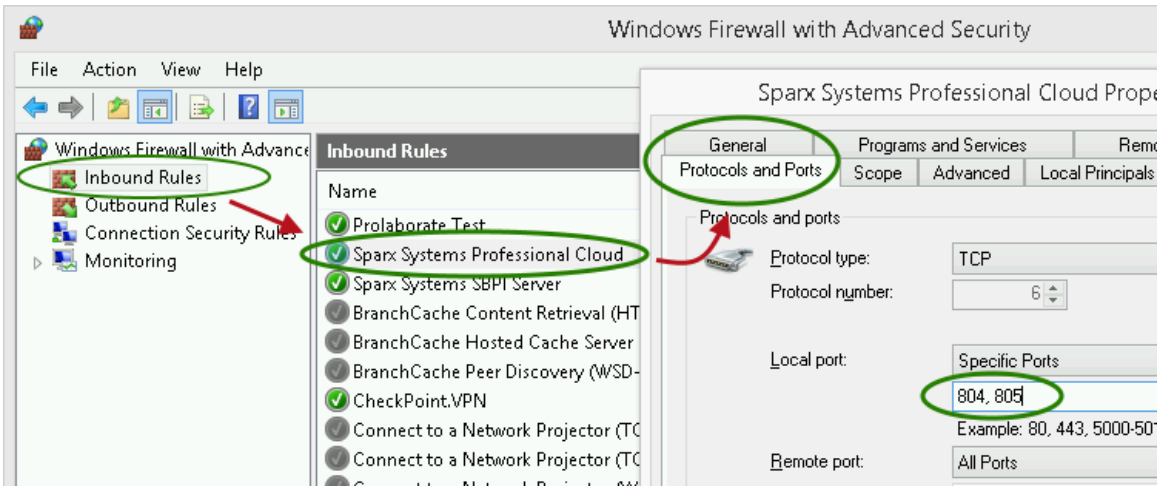
Firewall Settings

When setting up a Pro Cloud Server (PCS), you must check that any Firewall on the server allows for the incoming ports for the database connections that you have created.

For the server's firewall you must set the allocated Ports as enabled for inbound traffic. For example, the default ports for:

- PCS 4.2.65 or earlier, are 804 and 805
- PCS 5.0 or later, are 1804 and 1805.

Below is a Windows firewall example for PCS 4.2.65 or earlier.



The Pro Cloud Server installation will automatically create an inbound Windows firewall rule, similar to the one shown in the image; however, the default rule should be reviewed and updated to suit your environment.

For information on creating the Inbound Rule to identify the Ports you have specified in the config file (the 'Sparx Systems Professional Cloud' rule in the illustration), see the Microsoft Windows online topic Create an Inbound Port

Rule.

Note: A common cause of failure is that other services or programs 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 Pro Cloud Server can be run multiple times with different configurations by passing command line arguments.

The Pro Cloud Server executable can be run as either a standalone executable or a Windows service. By default the Pro Cloud Server installer will create a Windows service that does not specify any arguments (which forces the Pro Cloud Server to uses the default settings).

Command line arguments

Argument	Description
--config / -c	The argument following this will be interpreted as the filename of the configuration file to read on start up. This allows a secondary instance of the Cloud server to run on a different set of Ports, using an altered logging level or different directories. When multiple configurations are being used, you must ensure that each instance reads a configuration file that specifies different paths. Otherwise only the first instance will start successfully. By default this references

	SSProCloud.config in the directory containing the service executable. Note: this parameter should NOT include a path, only the filename. That is: SSProCloud.exe standalone -c mypcsconfig.config
--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_CURRENT_USER, where the models provided by the Cloud server are specified. Changing this parameter allows the use of different Cloud servers to provide access to different models. By default this is: HKCU\Software\Sparx Systems\SQLBridge

standalone	This argument allows starting the SSProCloud.exe as a standalone executable instead of requiring it to be launched as a Windows service. This can be important if the 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 there are any clients connected. Use the command line: wine SSProCloud.exe standalone Or, if you are launching from a terminal and want to close it: wine SSProCloud.exe standalone & disown or nohup wine SSProCloud.exe standalone (The 'disown' parameter and 'nohup' command both close the terminal but ignore the 'hangup' message that would otherwise disconnect Pro Cloud Server again.)

Additional Windows Services

It is possible to create additional Pro Cloud Server Windows services that run independently of one another and use different configurations. Special care needs to be taken to ensure the correct syntax is specified when creating an additional Pro Cloud Server instance.

For example to create a second Windows service that has a '2' appended to the default names/folders, perform these steps.

1.	Copy the entire contents of 'C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\' to a new folder of 'C:\Program Files (x86)\Sparx Systems\Pro Cloud Server2\'
2.	Edit the 'C:\Program Files (x86)\Sparx Systems\Pro Cloud Server2\SSProCloud.config' file and adjust the list of ports that this instance of the PCS should listen to.
3.	Open REGEDIT, and navigate to HKEY_USERS\.DEFAULT\Software\Sparx Systems\SQLBridge
4.	Clone the 'SQLBridge' node (and all children) to SQLBridge2'

6.	To apply a description to the newly created Windows service, execute this command: C:\> sc description "Sparx Systems Professional Cloud 2" "Sparx Systems Professional Cloud 2"
5.	Open a terminal windows and enter this command: C:\> sc create "Sparx Systems Professional Cloud 2" binpath= "\"C:\Program Files (x86)\Sparx Systems\Pro Cloud Server2\Service\SSProCloud.exe\" -p \"C:\Program Files (x86)\Sparx Systems\Pro Cloud Server2\Service\" --registry \"Software\Sparx Systems\SQLBridge2\"" start= auto DisplayName= "Sparx Systems Professional Cloud 2" This will define a new Windows service called **Sparx Systems Professional Cloud 2** and instruct it to use the 'C:\Program Files (x86)\Sparx Systems\Pro Cloud Server2\Service\' folder as the base location and 'HKEY_CURRENT_USER\Software\Spa

	rx Systems\SQLBridge' as the registry hive.

Pro Cloud Server using IIS (Optional)

This topic describes the use of Microsoft's Internet Information Server (IIS) to replace the Pro Cloud Server's in-built webserver. IIS should be used only under certain conditions.

Background

The ability to use IIS was introduced in an early version of Sparx Systems Cloud Services (the product prior to the Pro Cloud Server) to allow Enterprise Architect users to implement a basic form of single sign-on (although the prompt for user credentials was not completely eliminated using this method).

In version 3 of the Pro Cloud Server, native support for Windows Active Directory and OpenID single sign-on was added, providing a more integrated, robust and secure single sign-on methodology.

Limitations

The main limitation of using IIS instead of the in-built PCS webserver is that functionality such as the Floating License Server and SBPI Integrations will not operate when Enterprise Architect users are connecting via IIS. IIS Ports also do not support the use of 'Pro Features' such as OSLC and WebEA access. Basically, the IIS connection only allows Enterprise Architect clients to read data from a repository.

Recommendations

Given the above information, Sparx Systems would now only recommend the use of IIS when users:

- Are using a version of Pro Cloud Server prior to v3
- Are not using the Floating License Server component
- Are not using SBPI Integrations
- Are not using the native PCS single sign-on feature

Configuration

To configure IIS to host the Pro Cloud Server, you must first set the SparxCloudLink.sseap file to refer to the Pro Cloud Server, 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 SSProCloud.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 Pro Cloud Server.

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\Pro Cloud Server\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 Pro Cloud Server. In most circumstances it is best to run the HTTP module and the Pro Cloud Server on the same machine. In this case, the default value of localhost can be used. If the Pro Cloud Server is running on a different machine, use the IP address or server name where the service is running.

port:

The Port the Pro Cloud Server 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 SSProCloud.config file.

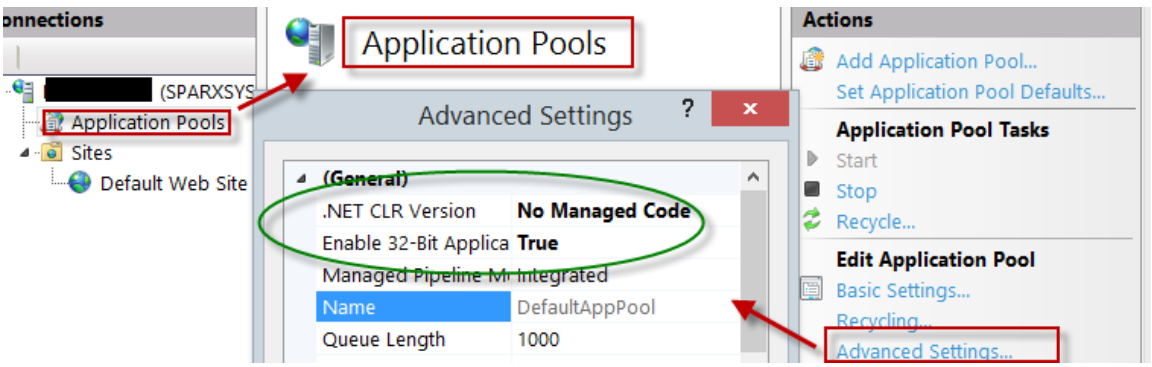
Optional settings

These points are optional. For testing purposes you might prefer to leave these changes until any issues with IIS are resolved.

1. In the SSProCloud.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 SSProCloud.config file.
3. Restart the Service.

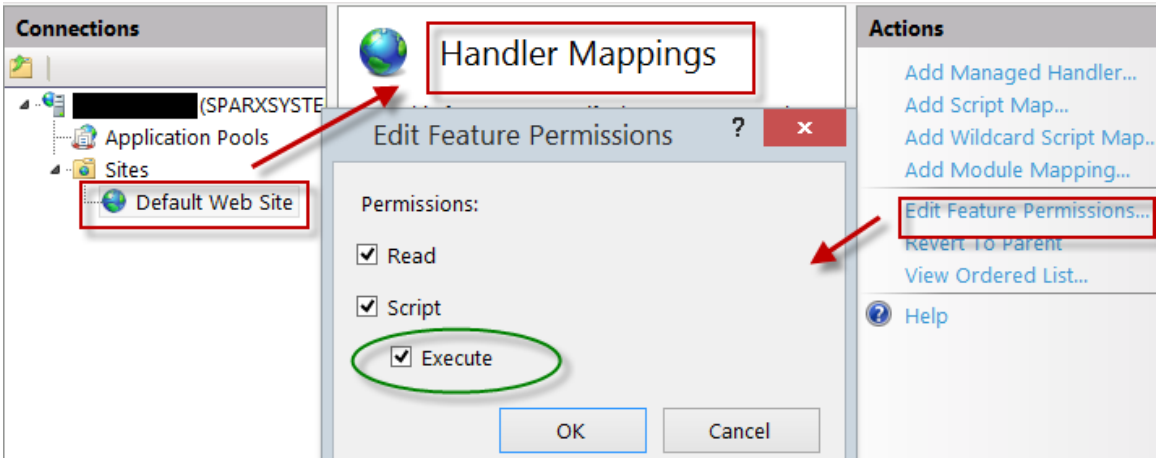
Application Pool Settings

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

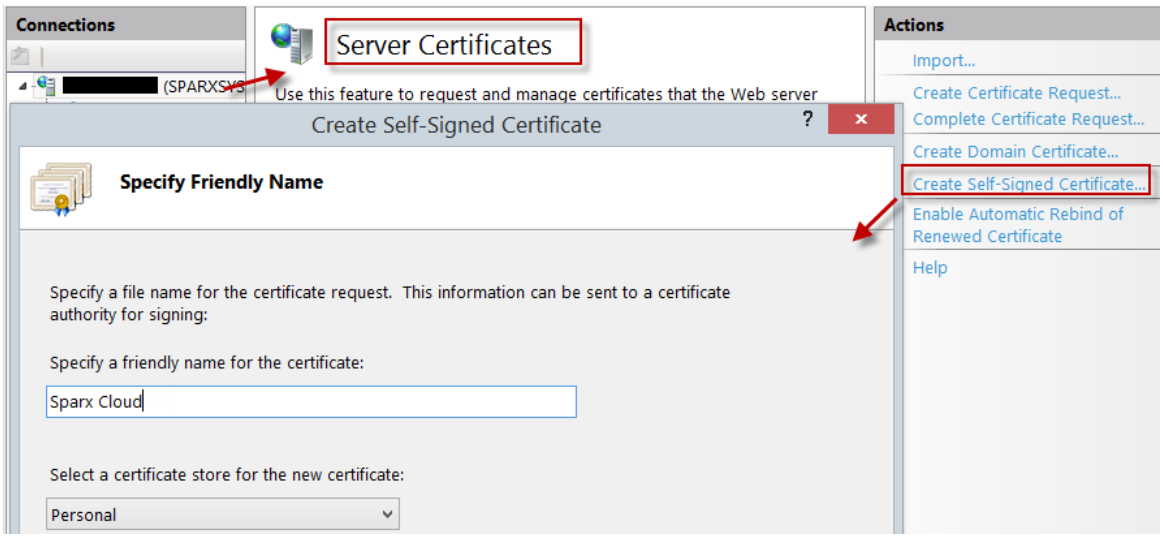
This illustration shows the access path and the 'Default Web Site | Handler' mappings to permit Script Execution.



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.

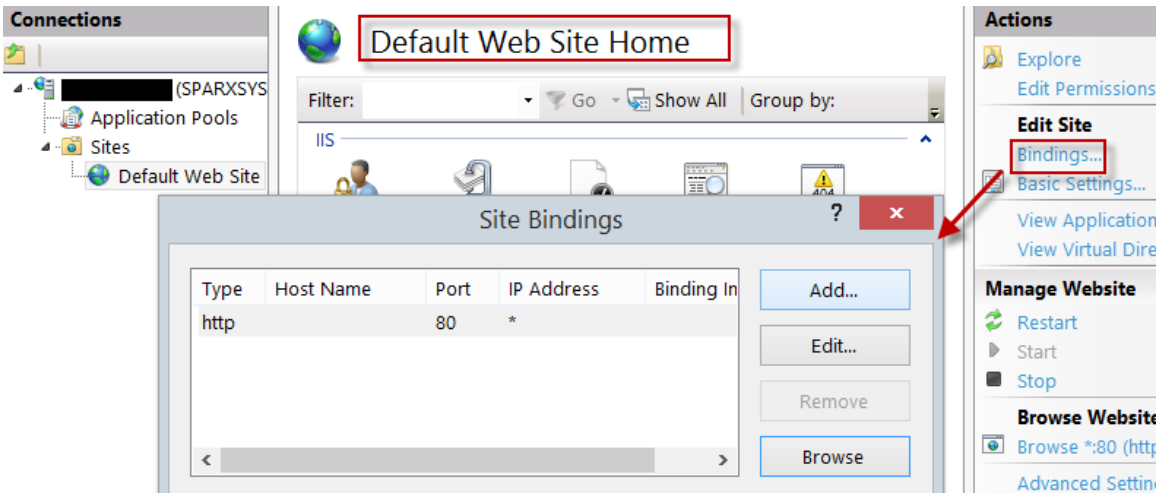


Setting up HTTPS

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:

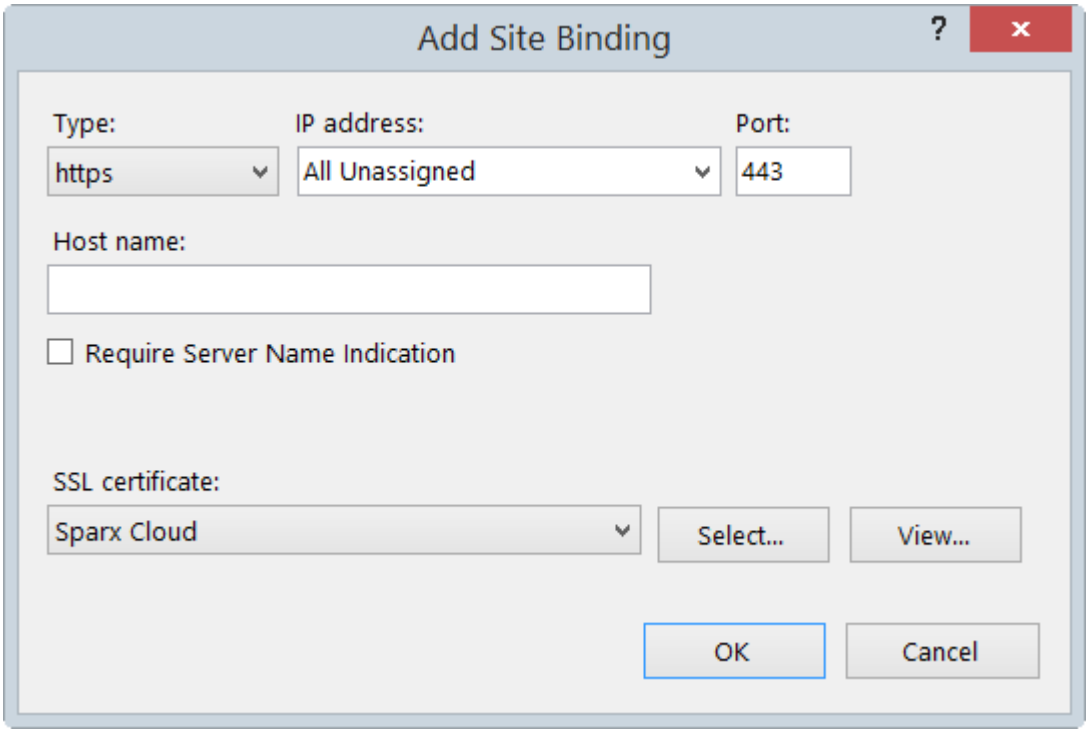
- Under 'Connections', select the 'Default Web Site' option
- Under 'Actions' click on the 'Bindings' option
- In the Site Bindings window click on 'Add'



This will open the Add Site Binding window.

Set these parameters:

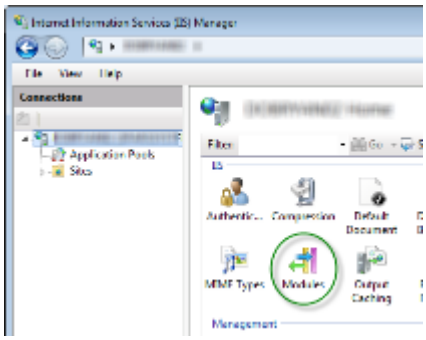
- Type: HTTPS
- Port: 443
- SSL Certificate: select the certificate created in *Setting up a Certificate*



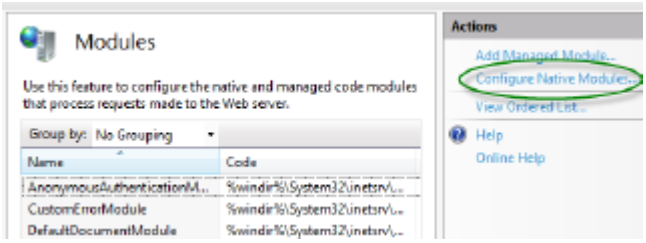
HTTP Module

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.

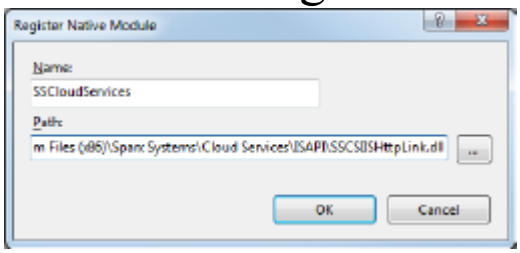


   This returns the 'Modules' list and the 'Actions' view.
3. In the 'Actions' list, click on the 'Configure Native Modules…' option.



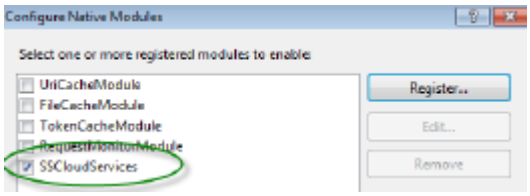
This opens the 'Configure Native Modules' view.

4. Click on the Register button to open the 'Register Native Module' dialog.

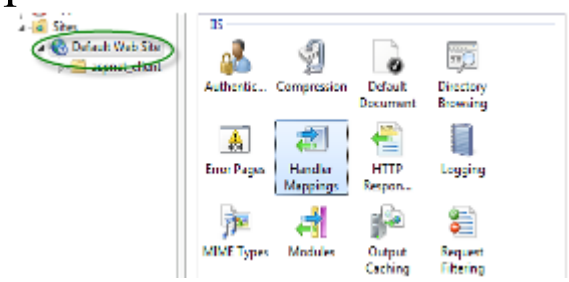


Type in the name and the path of the SSCSIISHttpLink.dll file.

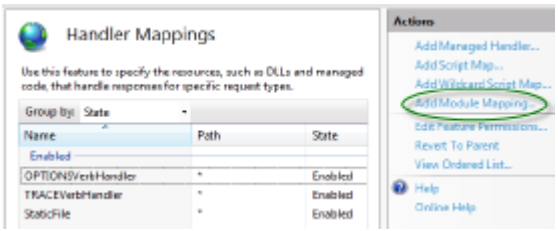
5. Click on the OK button. The 'SSCloudServices' checkbox will now be selected.



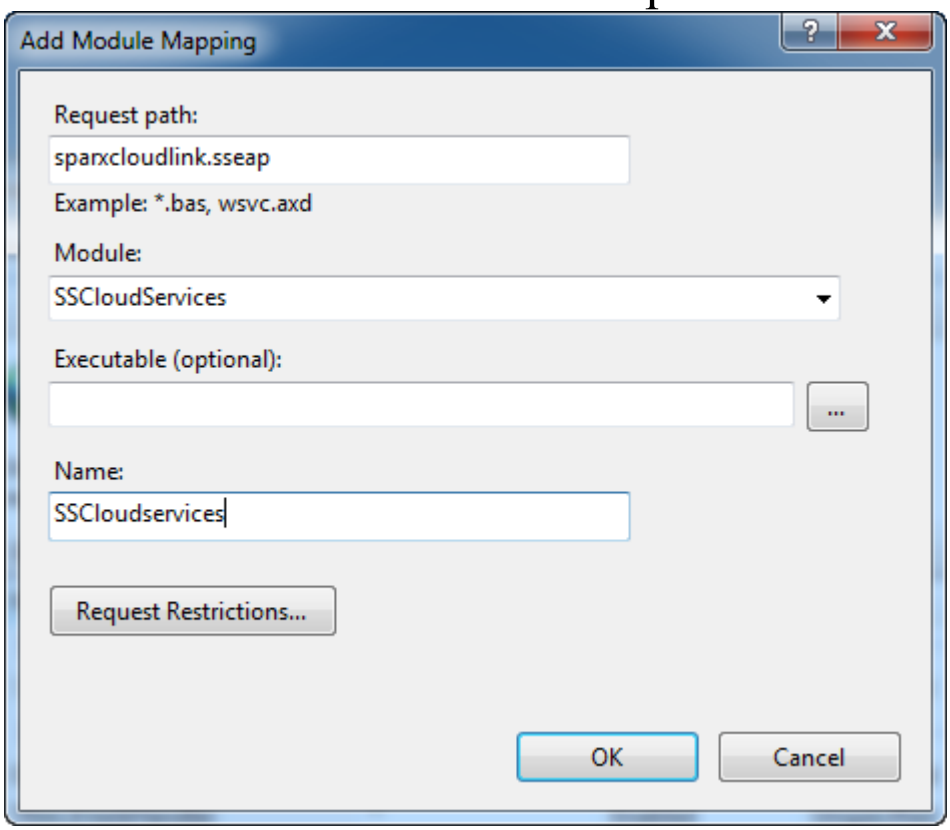
6. Click on the OK button to close the 'Configure Native Modules' dialog.
7. In the 'Connections' panel (see Step 1), click on your web site.
8. Double-click on the 'Handler Mappings' in the middle pane.



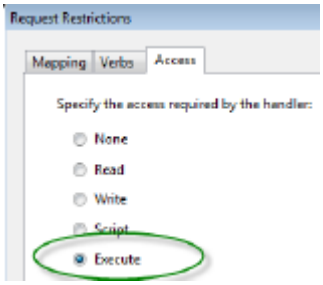
This opens the Handler Mappings view:



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. On the 'Request Restrictions' dialog, select the 'Access' tab. Select the 'Execute' radio button to enable Execute permission.



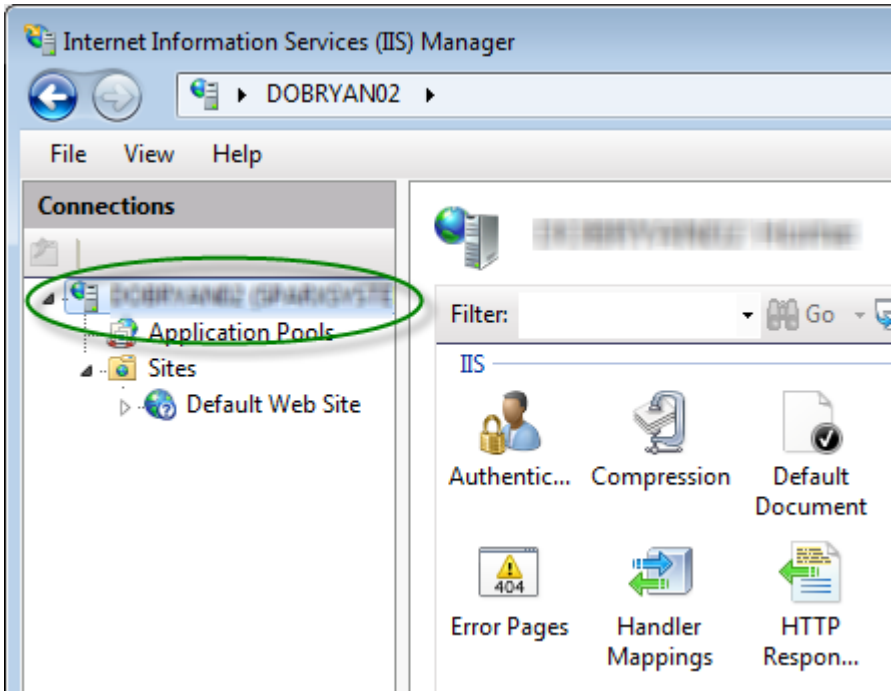
13. Select the 'Mappings' tab and deselect the 'Invoke Handler only if request is mapped to' checkbox.
14. Click on the OK button.
15. Close the 'Add Module Mapping' dialog by clicking on the OK button again.

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 (e.g. via Port 80 or 443).

ISAPI Module

To configure an ISAPI module instead of the HTTP module:

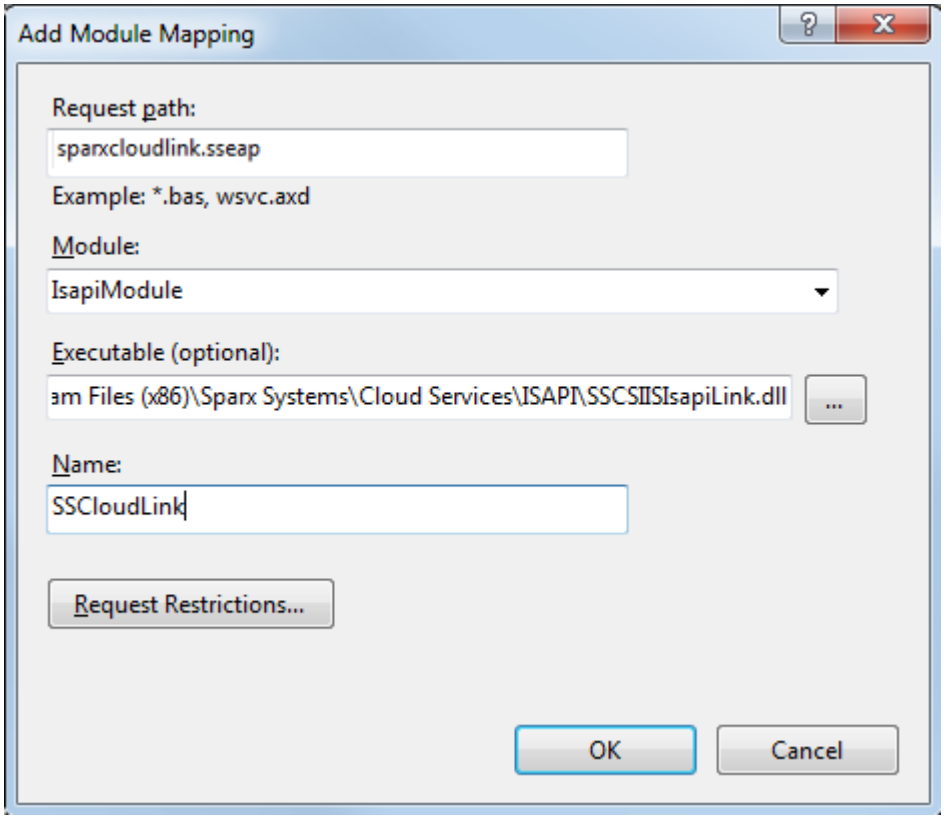
1. In the 'Connections' panel, select the machine properties:



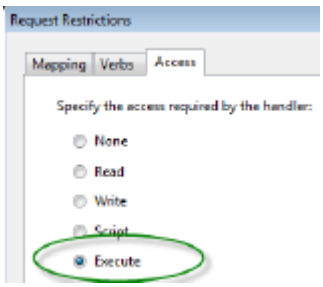
2. Double-click on the 'Handler Mappings' icon:



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 Request Restrictions button. On the 'Request Restrictions' dialog, select the 'Access' tab. Select the 'Execute' radio button to enable Execute permission.



6. Select the 'Mappings' tab and deselect the 'Invoke Handler only if request is mapped to' checkbox, then click on the OK button to close the 'Request Restrictions' dialog.
7. Click on the OK button to close the 'Add Module Mapping' dialog. You should now be able to connect to a model using Enterprise Architect via your IIS server using ISAPI.

Floating License Server

The Floating License Server provides a mechanism to allocate and share Enterprise Architect and other Sparx System Product licenses between multiple users. Licenses are only allocated to users when they open Enterprise Architect, once the application is closed the license is released back into the pool and is available to another user. Therefore this mechanism of sharing licenses can allow an organization to purchase a smaller number of licenses and make more efficient use of the licenses purchased.

The Sparx Systems Key Store server has been available for many years, providing users of Enterprise Architect with the ability to manage their floating licenses. In version 2.1 of Pro Cloud Server the same Floating License functionality offered by the 'Keystore Service' was integrated into the Pro Cloud Server and enhanced to support SSL and IPv6.

Now that floating license allocation is integrated into the Pro Cloud Server, users only have to install, configure and maintain one Microsoft Windows Service for all their Enterprise Architect needs.

This feature required a license for versions 2.1 to 4.1.43; Sparx Systems have relaxed this requirement from version 4.1.44 onwards.

Install and Configure

The Sparx Systems Pro Cloud Server product is installed using a standard Microsoft Windows Installer package that includes a number of components. The Floating License Server is included in the main 'Pro Cloud Server' component. It is possible to change the install location of the Pro Cloud Server; however this document assumes the default install location of C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\ (assuming Pro Cloud Server 32 Bit is being used).

Important: All communications to the Pro Cloud's Floating License Server are performed via HTTPS; therefore, to use the Pro Cloud Server's Floating License Server functionality at least one Port has to be configured to use the HTTPS protocol, and a valid certificate/private key (server.pem) file must be defined in the Pro Cloud Server's Services folder. See the *Add a Port Definition* Help topic for details.

The main Pro Cloud Server's configuration file (SSProCloud.config) defines two settings related to the Floating License Server: FLOATLIC_PATH and FLOATLIC_CONFIG.

The FLOATLIC_PATH option defines the path and file name of the Key Store file; by default this is:

%SERVICE_PATH%\sskeys.dat

"%SERVICE_PATH%" is replaced with the physical path of the 'Services' folder (that is, 'C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service').

The FLOATLIC_CONFIG option defines the path and file name of the Key Store group file; by default this is:

%SERVICE_PATH%\ssflsgroup.config

The Pro Cloud Server's Floating License Server component installs a configuration file called ssflsgroup.config into the Service folder (C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service\ssflsgroup.config). This file defines 'groups' of users that can obtain Licenses from the Floating License Server, and also controls which Enterprise Architect Product Licenses each group can use.

A group is identified by a group name and a set of credentials (user ID and password) that will be used to securely communicate with the Floating License Server over HTTPS.

From Pro Cloud Server v4.2 (build 60) the Floating License Server Group configuration file can be maintained using the Floating License Server Configuration Client (see the Manage Group Configuration Help topic), prior to this editing the file was a manual process.

The Floating License Server's configuration file will typically consist of multiple group definitions; each group definition must be defined as shown:

GROUP

Name=human_resources

```
        UserName={user name}
        Password={password}
        IsManager=false
        ENTITLEMENT
            Product=UltimateSuite
            Academic=false
            Limit=10
        END ENTITLEMENT
        ENTITLEMENT
            :
        END ENTITLEMENT
END GROUP
```

Group Definition Properties

A Group definition begins when the token **GROUP** is encountered on a new line, and extends until the token **END GROUP** is found on a new line.

Property	Description
Name	The common name of the Floating License group.
UserName	Required. Specifies the User ID for the Floating License Group.

Password	Required. Specifies the Password for the Floating License Group.
StartDate	Optional. Used in combination with the EndDate property to define the date range within which the Group can be used.
EndDate	Optional. Used in combination with the StartDate property to define the date range within which the Group can be used.
Entitlement - Section	Each Group can define multiple Entitlements. An Entitlement definition begins when the token **ENTITLEMENT** is encountered on a new line, and extends until the token **END ENTITLEMENT** is found on a new line. Each Entitlement definition accepts each of these sub-properties: • Product={Product Code} - The product that this entitlement gives this group access to • Academic={true or false} - Determines if this entitlement applies to the academic edition of the specified Product • Limit={number} - Optionally defines the maximum number of Product keys

	available to this group The valid Product codes are: • Professional • Corporate • UnifiedSuite • UltimateSuite • BPSimExecution • EclipseIntegration • MDGCodeAnalysis • MDGDDS • MDGDoors • MDGLinkEclipse • MDGLinkVS • MDGRealTime • MDGSysML • MDGTogaf • MDGUPDM • MDGZachman • OfficeIntegration • RaQuest • TCSEIntegration • TCSESuite • VSIntegration
IsManager	Optional. Specifies whether members of this group are permitted to perform

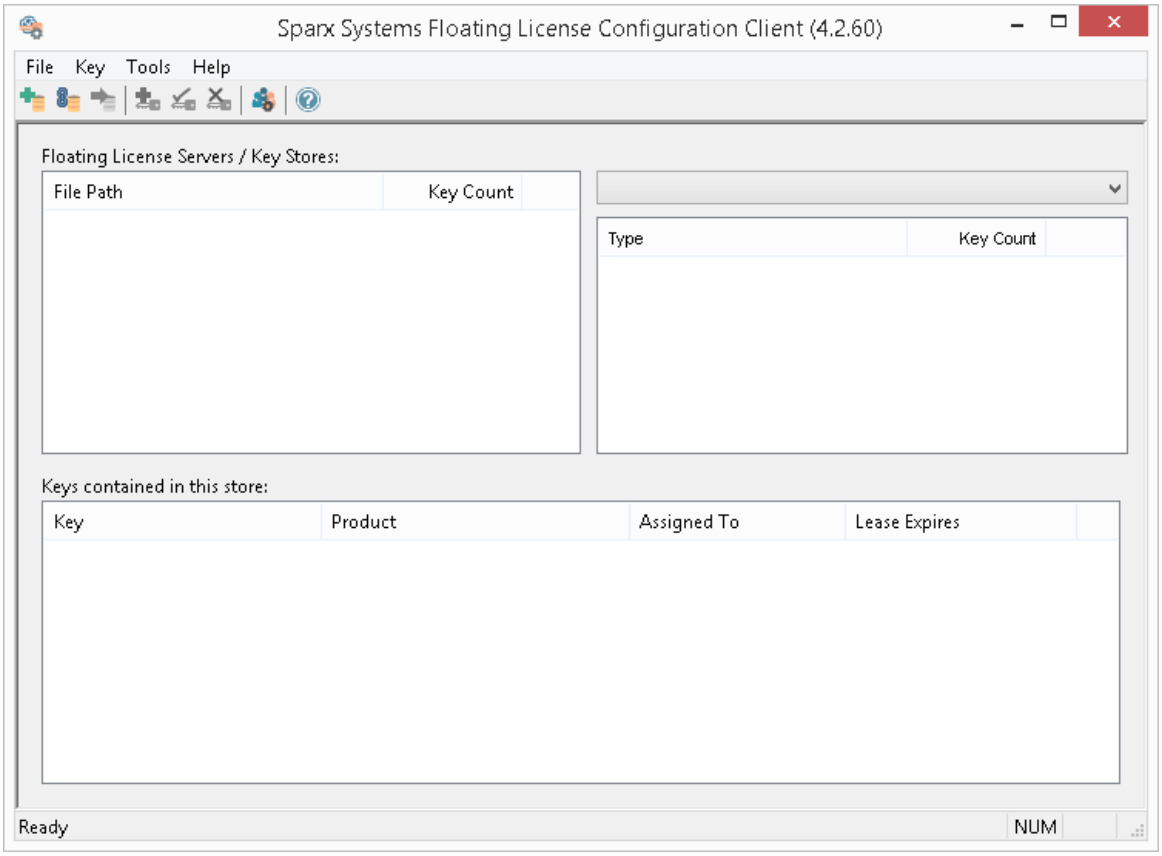
	management operations (such as add/remove keys) on the keystore. Default Value: false

Configuration Changes

Changes to the Pro Cloud Server's Floating License Group file can be made while the Pro Cloud Server is running, without the need to restart it. The Floating License Server component will detects the last modified time, if the configuration file has changed since it was last read the entire content of the file is re-read.

Floating License Configuration Client

Administration of License keys that are stored in the Pro Cloud Server's Floating License Server is managed by a Windows application installed in the Client folder (ie. for 32 bit, C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Client\SSFloatLicClient.exe).



When the Floating License Configuration Client application is first started it will attempt to open a connection to the most recent Floating License Server or keystore. If this is the first time running the Configuration Client, you will be automatically prompted for the connection details of a Floating License Server. See the Floating License Server Connections Help topic for more details

The Floating License Configuration Client is capable of

connecting and administering multiple Floating License Servers and Shared KeyStores at any given time. The Floating License Servers Stores list (in the upper left of the screen) will display each store that the Configuration Client has a connection to.

The Floating License Configuration Client application is capable of performing actions that can be broadly classified into two different areas:

- Floating License Server-based actions, including:
  - Floating License Server Connections
  - Set Shared Key Lease Periods
  - Migrate Shared Keys between Stores
  - Manage Group Configuration
- Key-based actions; selecting a connected store from the Store list will enable you to maintain the keys assigned to that store, using these functions:
  - Add Shared Keys
  - Release Shared Keys
  - Delete Shared Keys

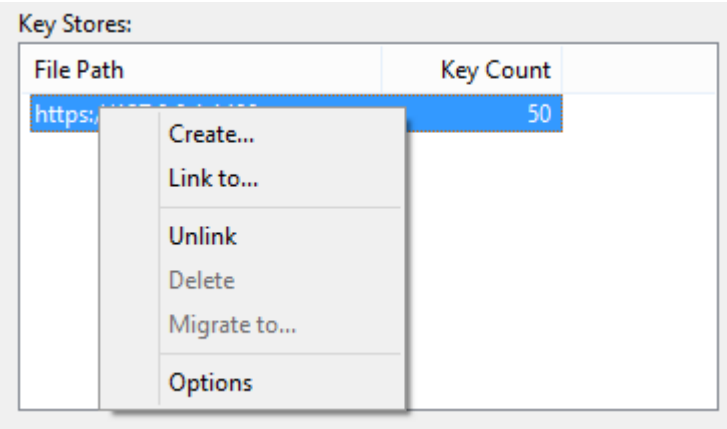
Floating License Server Connections

The top left corner of the 'Floating License Configuration Client' screen displays a list of connected Floating License Servers and Shared Keystores. Each entry in the list can be one of three different types of store:

1. Floating License Server based (prefix: https://)
2. Shared KeyStore or network based (prefix: ssks://)
3. File based

The type of store can be determined by the connection's prefix (or lack of it); the network and file based keystore types are historical, from earlier versions of the Floating License Server (the KeyStore Service or KeyStore).

Floating License Server actions can be performed by selecting a Floating License Server/KeyStore item in the list and then selecting a menu item from the 'File' menu, or by simply right-clicking on an item in the list; the menu options are as shown:

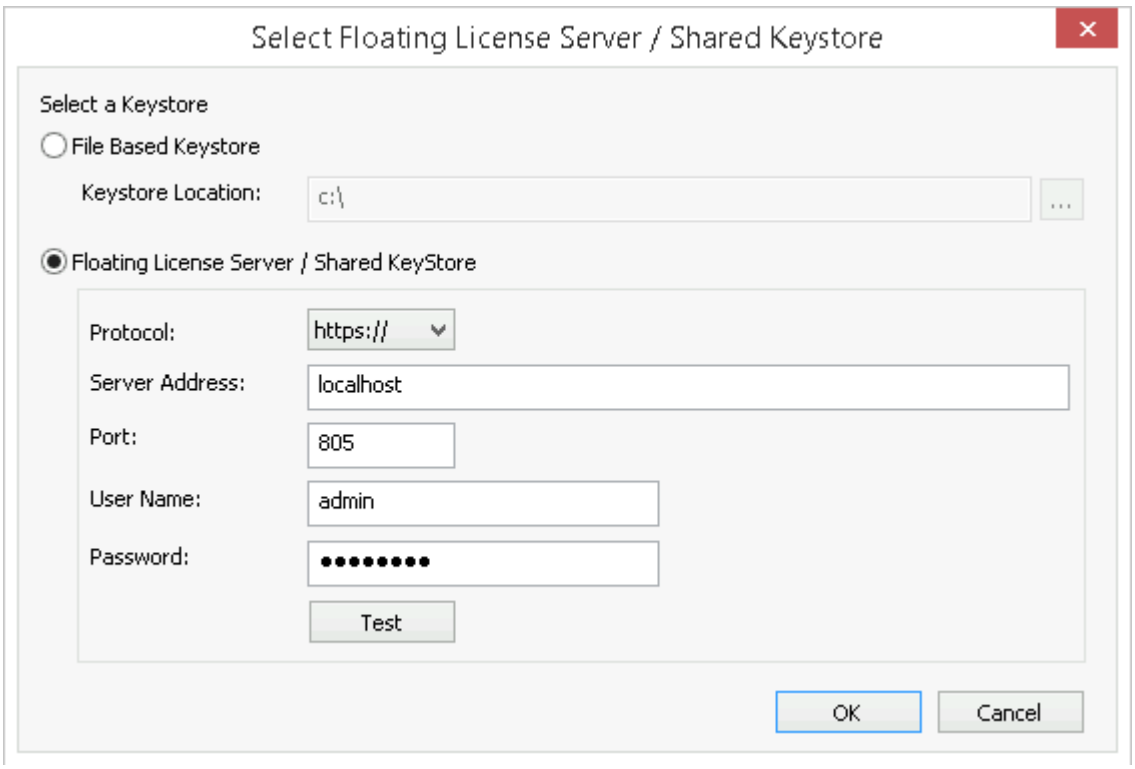


Create...

Selecting the 'Create...' option helps you to create a new file-based keystore.

Link to...

Selecting the 'Link to...' option will display a prompt for the connection details of an existing Floating License Server (or key store) to be added to the list of connected stores.



See the Default Passwords for Floating License Server Help topic for the default passwords for all pre-defined groups.

Unlink

Selecting the 'Unlink' menu option while a Floating License Server is selected in the Stores list will release the Floating

License Configuration Client's connection to that store and remove it from the list.

Delete

Selecting the 'Delete' menu option while a file-based Key Store is selected in the 'Key Store' list will physically and permanently delete the file used to store the keys. This option is not available if the selected Keystore is either a Floating License Server or shared network KeyStore.

Migrate to...

Selecting the 'Migrate to...' menu option while a file-based Key Store is selected in the Floating License Server 'Key Store' list will allow the migration of keys from one store to another. This option is not available if the selected store is either a Floating License Server or shared network KeyStore-based. See the Migrate Shared Keys between Stores Help topic for more details.

Options...

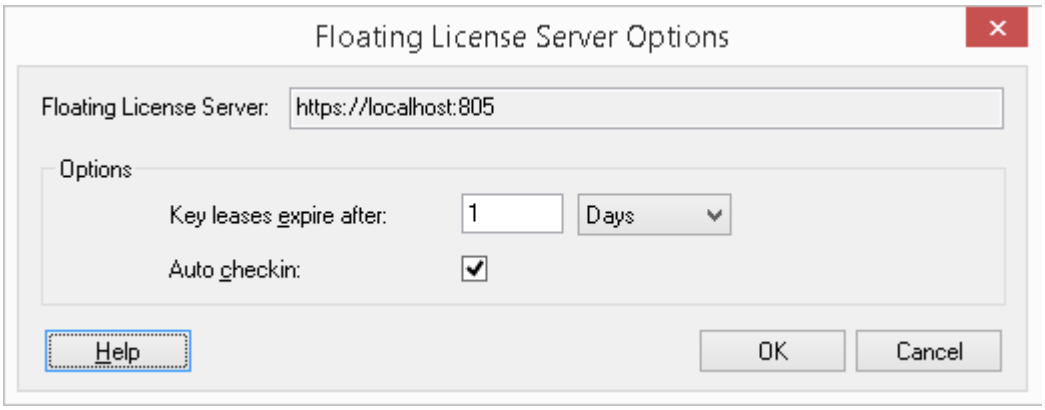
Selecting the 'Options' menu option will display the 'Options' screen for the selected Floating License Server. See the Set Shared Key Lease Periods Help topic for more details

Set Shared Key Lease Periods

Normally, in an environment that requires licenses to be shared, an Enterprise Architect user 'releases' the license they have 'leased' immediate upon closing the application. However, in an environment where Enterprise Architect users work offline for periods of time (for example, using laptops) this type of configuration would not allow an offline user to open Enterprise Architect once they have disconnected from the main network.

For this reason a Floating License Server / Key Store can be configured to have a longer retention period.

To administer the lease period of a Floating License Server / Key Store, select a store from the list and then select the 'Options' menu option. The 'Keystore Options' dialog is then shown.



Key leases expire after

This pair of fields determine how long a user can retain a key even if they log off Enterprise Architect. Typically you would set this for the period that a user is unable to access

the keystore, if they intend to work in Enterprise Architect during that period. The setting is overridden by the 'Auto checkin' field, if it is enabled.

Type in the number of units of time, and then click on the drop-down arrow and select the appropriate unit.

Auto checkin

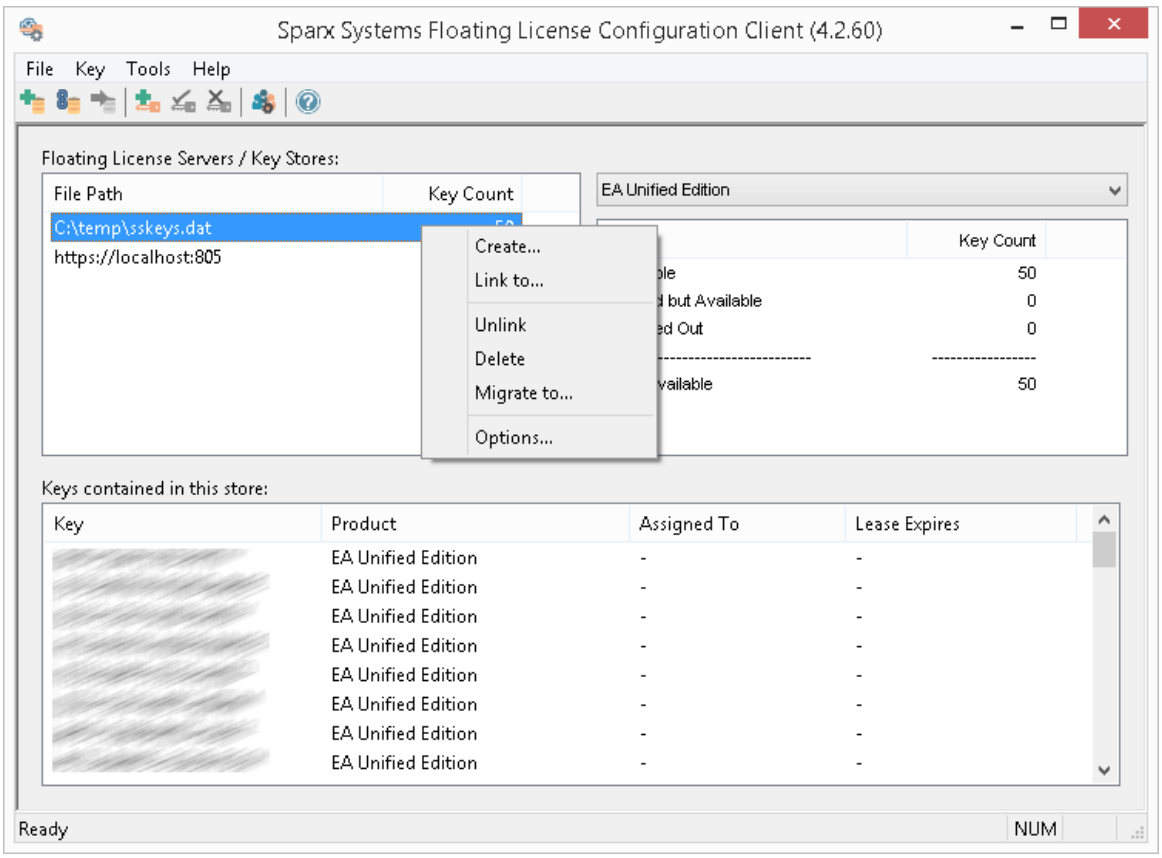
If users need to keep a key checked out when closing Enterprise Architect (for example, if a laptop user needs to keep using the Enterprise Architect license for X days while not having access to the keystore server) then this option should be disabled.

However, for the most efficient sharing of keys this option should normally be left enabled. When enabled, if a user closes Enterprise Architect then their license key is automatically returned to the keystore (regardless of the lease expiry setting).

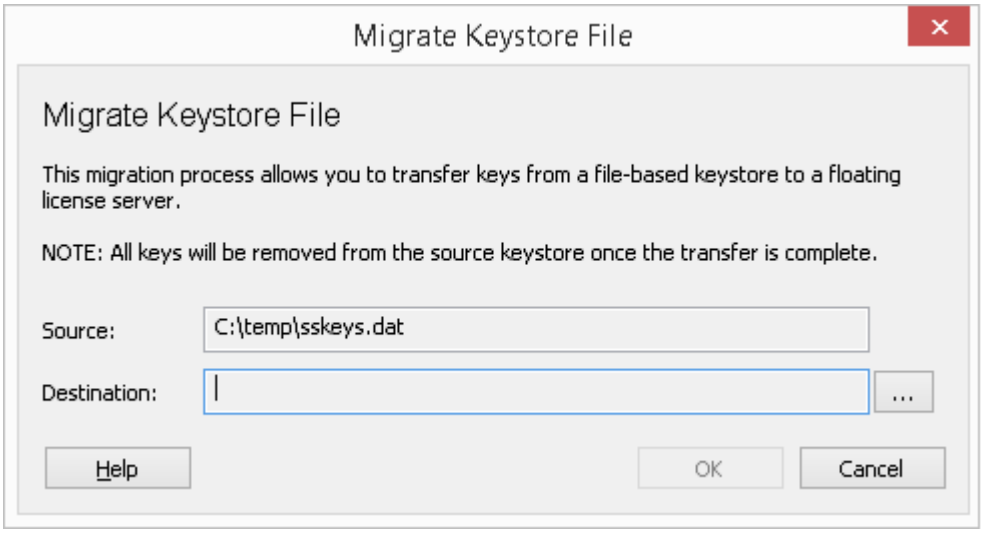
Migrate Shared Keys between Stores

To make best use of your upgraded Pro Cloud Server application, you might want to upgrade from a legacy file-based keystore or network based keystore to the Pro Cloud Server Floating License Server. This process effectively removes all keys from the original keystore, leaving it empty, and places the keys into the Floating License Server.

To migrate shared keys from other keystores, you must first define a connection to each source keystore, using the 'Link to' context menu option in the Stores list. Then, making sure the source keystore is selected in the list, select the 'Migrate to...' context menu option.



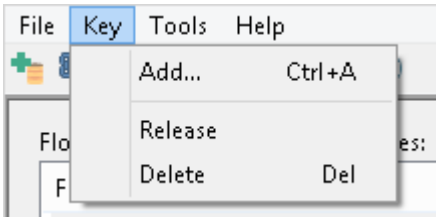
The 'Migrate Keystore File' screen displays:



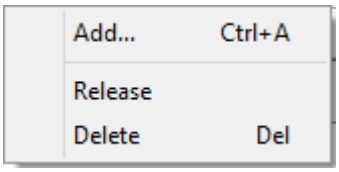
Click on the ... button on the right of the 'Destination' field, and browse for the details of the Pro Cloud Server Floating License Server into which the shared keys are to be transferred. Click on the OK button to migrate the shared keys from the source into the destination Floating License Server.

Add Shared Keys

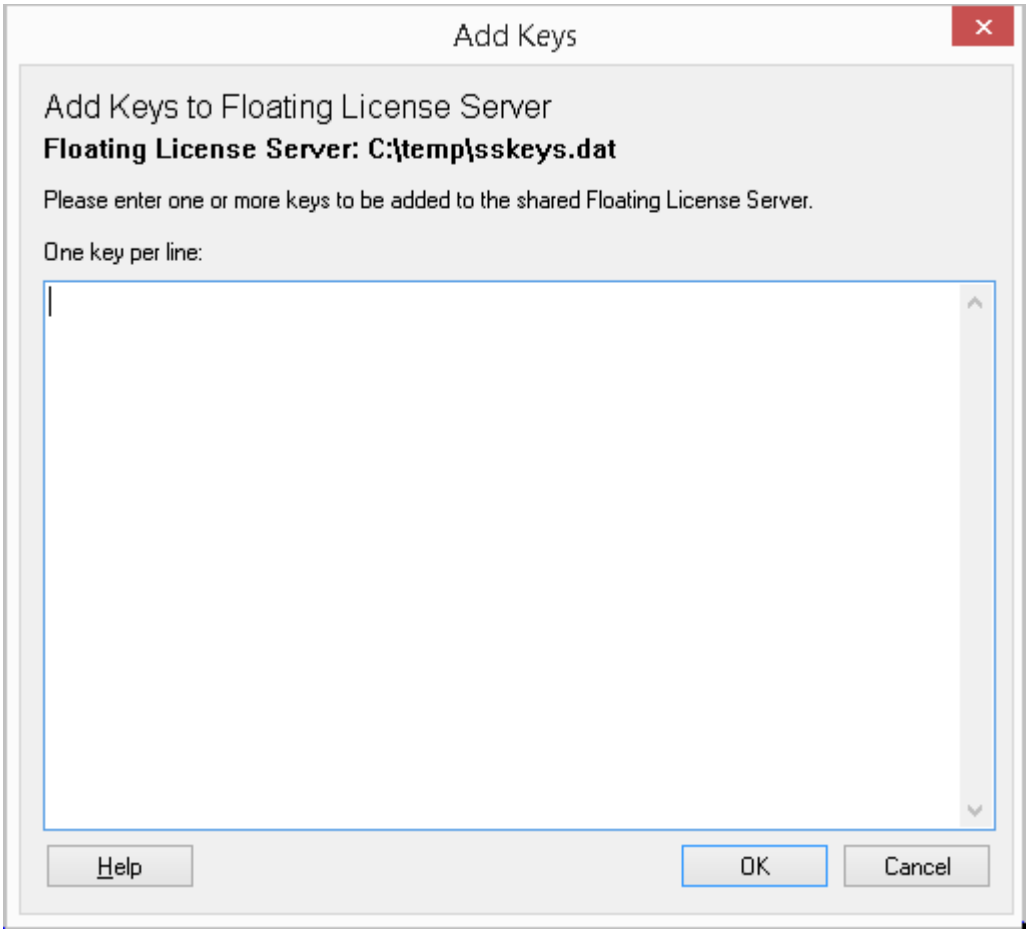
To add shared keys to a Store, select the Floating License Server / Key Store in the 'Store' list, and then select 'Key | Add...' from the header menu.



Alternatively, right-click on the 'Keys contained in this store' list and select 'Add...'.



Both of the above option will open the same Add Keys screen,



simple paste in each License key (making sure to place each key on a separate line) and click on the OK button.

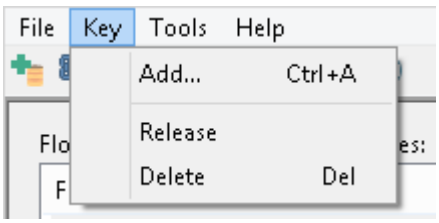
Release Shared Keys

It is usually not necessary to manually release a shared key from a workstation, as this happens automatically if:

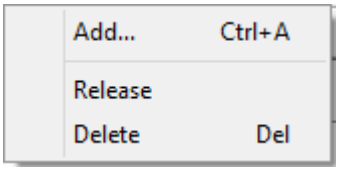
- The Store's 'Auto checkin' option has been enabled and you close all running instances of Enterprise Architect on your workstation
- The lease on the key expires

However, if further users want to access Enterprise Architect and there is some technical anomaly that has prevented the return of a key, you can manually release the key using the Sparx Systems Floating License Configuration Client.

To release a key from its active association with a particular workstation, click on the key in the 'Keys contained in this store' list, and then select the 'Key | Release' header menu option.

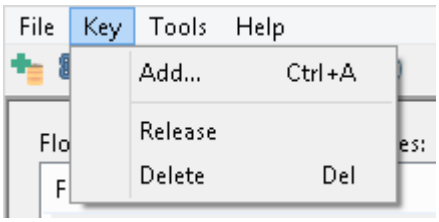


(Alternatively, click on the 'Release Key' icon in the toolbar, or right-click on the key and select the 'Release' context menu option.)

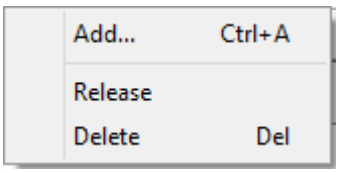


Delete Shared Keys

To delete a key from the selected Floating License Server / Key Store, open the Floating License Configuration Client, click on the key in the 'Keys contained in this store' list, and then select the 'Key | Delete' header menu option.



(Alternatively, click on the Delete Key icon in the toolbar, or right-click on the key and select the 'Delete' context menu option.)



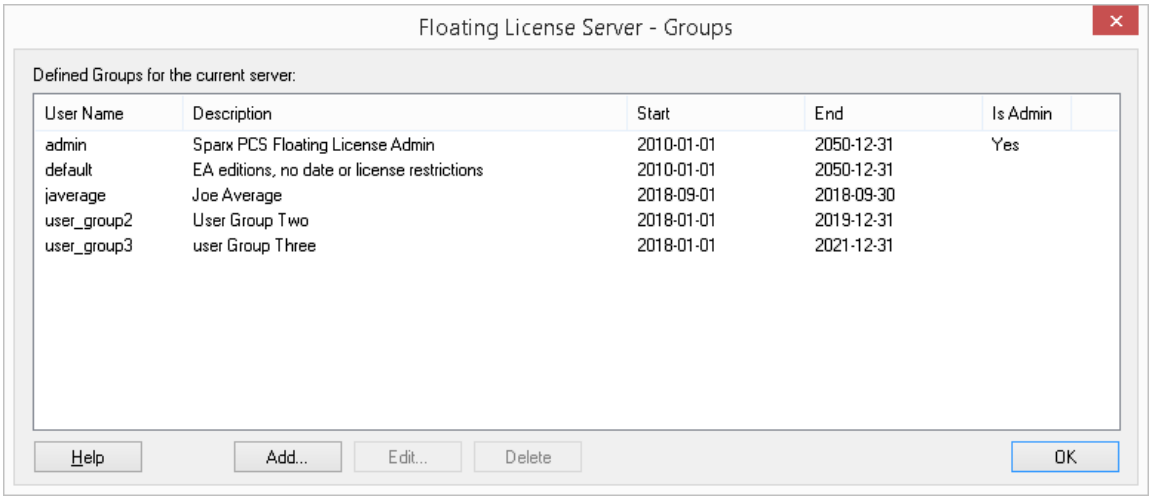
Manage Group Configuration

In order for users to 'check-out' a floating license they must be a member of a Floating License Server Group. The definition of a Group controls which Product Licenses members of the group can 'check-out' and the period of time that the group is valid.

To view the Manage Group Configuration screen click on the Manage Group Configuration button in the main toolbar.



The default Group configuration includes the definition of a number of Groups:



The first group shown in the example above is 'admin', each Floating License Server installation will require at least one administration group, for the purpose of allowing connections from the Floating License Server Configuration Client, admin groups cannot be assigned Products and so therefore can not be used by Enterprise Architect to obtain a

license.

The remaining groups are example of normal (non admin) groups that can be used by Enterprise Architect to obtain a license. 'Joe Average' is an example of an expired group that can no longer be used to obtain a license, any attempt to use this will result in a error message indicating the specified group is not longer valid.

Note: Manage Group Configuration was introduced in PCS version 4.2, the Floating License Server Configuration Client in combination with the Pro Cloud Server can now manage the configuration of Floating License Server Groups, prior to this the Group file (ssflsgroups.config) had to be manually edited using a text editor.

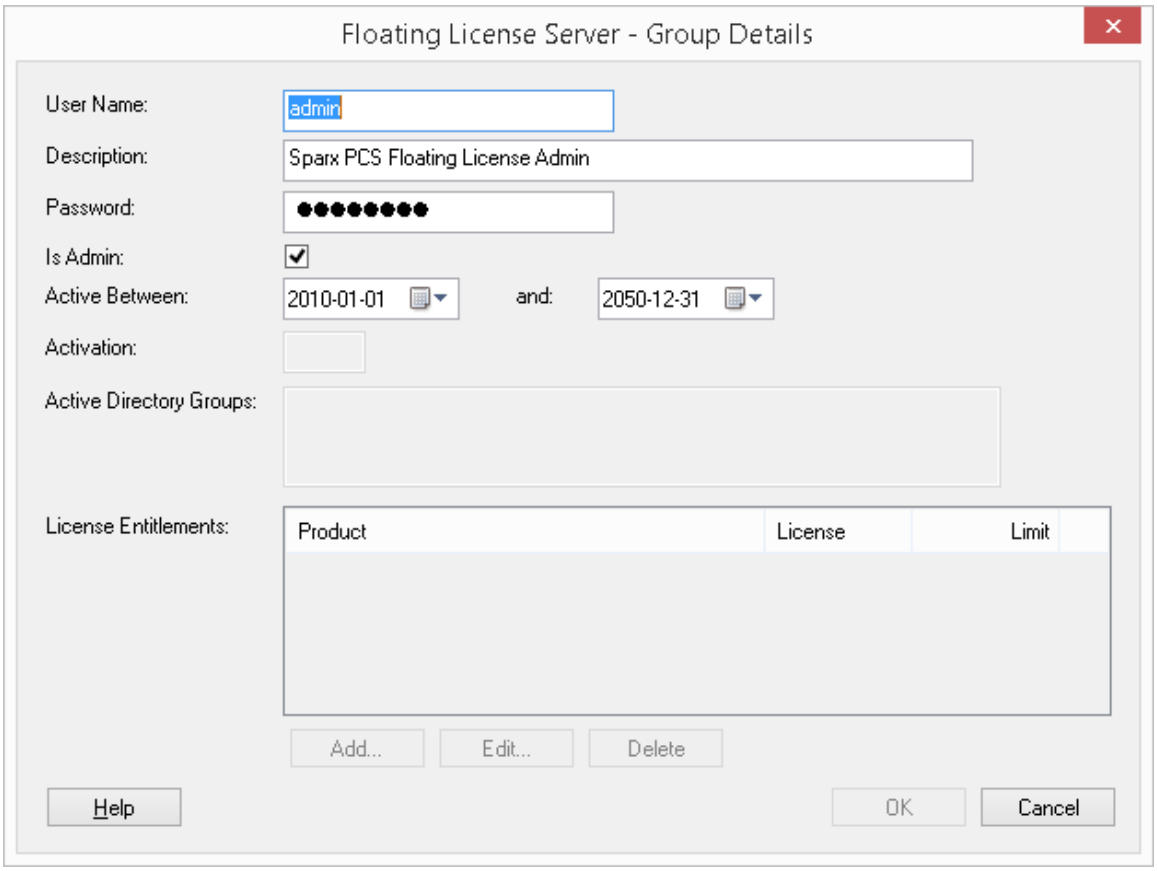
Item	Description
List of Groups	Displays a list of defined Groups in the current Floating License Server. **Note:** Double clicking a row in this control will display the Group Details screen in 'Edit' mode.
Add	Selecting this button will display the Group Details screen in Add mode, and will allow the creation of a new Group

	for the current Floating License Server.
Edit	Selecting this button will display the Group Details screen in 'Edit' mode and load the details of the selected Group. **Note:** This button is only enabled when an existing Group is selected in the List of Groups.
Delete	Selecting this button will delete the entire definition of the currently selected Group from the current Floating License Server. **Note:** This button is only enabled when an existing Group is selected in the List of Groups.

Manage Group Details

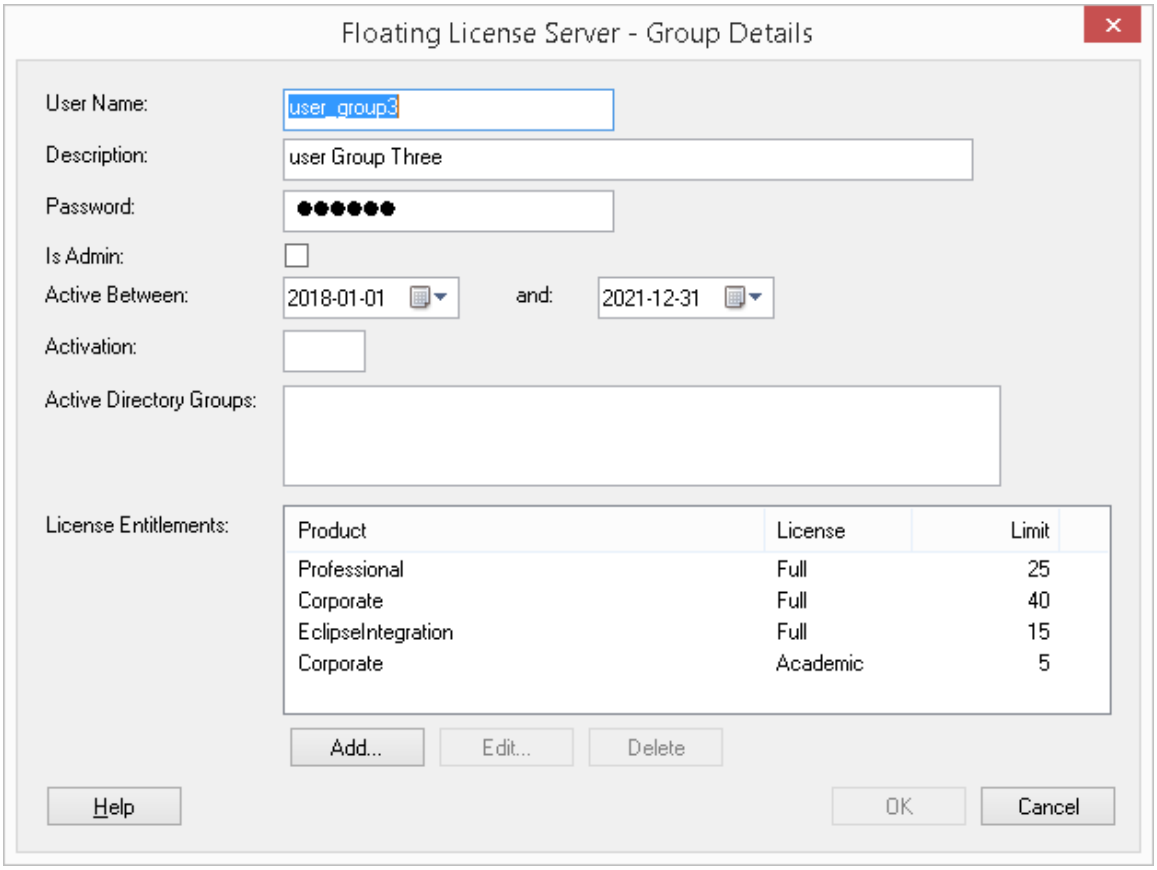
Managing the definition of a Floating License Server Group is performed using the 'Group Details' screen, which will be shown when either the Add or Edit buttons are pressed on the Mange Group Configuration screen.

Below is Group Details screen loaded with the example admin group that is created when the PCS is first installed:



Note: Whenever the 'Is Admin' option is checked the fields below this will be disabled since they are not relevant to administrator groups.

Below is the Group Details screen loaded with the example user_group3 group that is created when the PCS is first installed:



See the Default Passwords for Floating License Server Help topic for the default passwords for all pre-defined groups.

Floating License Server Group Fields

Item	Description
User Name	The unique user name a group is known by. When an Enterprise Architect client

	connects to the Floating License Server, it is this value that must be entered as the 'User Name'.
Description	A meaningful description for the current group.
Password	The password for the group. Along with the 'User Name', the password needs to be entered by Enterprise Architect clients in order to obtain a floating licence. From version 4.2 all groups that are updated will see the Password saved to the configuration file as an encrypted value, therefore it is no longer possible to simply view the underlying configuration file to determine a group's password.
Is Admin	When this flag is checked it indicates that the current group is only to be used for administering the Floating License Server's configuration via the Configuration Client. **Note:** Most of the fields below 'Is Admin' will be disabled once this flag is checked; ie Activation, Active Directory Groups, etc.
Active	These two date fields define the period of

Between	time that the current group is active. Once a group is inactive users will not be able to connect using this account. Previously it was possible to define a group that was not restricted to a period of time, however now with the use of the configuration client the same is achieve by defining the end date well into the future. An empty start date will be shown as '2010-01-01', while an empty end date will be 2050-12-31.
Activation	When Enterprise Architect is first installed it will prompt the user for an 'Activation Code' that was supplied with the original purchase, which is a simple four letter code to confirm the user's right to use that version of Enterprise Architect. By assigning the provided Activation code in this field all Enterprise Architect users will derive the Activation Code along with their floating license from the Floating License Server, thereby removing the need for each client machine to be updated whenever new licenses are purchased. **Note:** The ability for Enterprise Architect to obtain an Activation Code from the Floating License Server was

	added in Enterprise Architect v15.2 in combination with Pro Cloud Server v4.2, therefore earlier versions are not able to take advantage of this feature.
Active Directory Groups	This field is used to associate one or more Active Directory groups to the current group. Each Active Directory groups needs to be specified by its fully qualified name. If you want to define multiple Active Directory groups simply separate each by a comma; that is, a CSV of fully qualified Active Directory group names. "WinNT://MYDOMAIN/Sparx Users,WinNT://MYDOMAIN/EA Users,WinNT://CUSTOMERDOMAIN/Sparx Users" From Pro cloud Server 5.1 b125 onwards multiple naming formats are now supported. The 'WinNT://' is no longer required and group names can be in the following formats: • Canonical (recommended) - eg: my.domain.com/Users/FLSGroup1 This can easily be copied from the 'Active directory Users and Computers' group properties dialog for the Domain Controller. In the 'Object' tab.

	• Distinguised Name - eg: CN=FLS,OU=Sparx,OU=Apps,DC=my,DC=domain,DC=com • Legacy (SAM compatible) - eg: DOMAIN\flsgroup (note: forward slash also accepted) • GUID - eg: {95ee9fff-3436-11d1-b2b0-d15ae3ac8436} (note: with or without {}) • SID - eg: S-1-5-21-1046871492-2828434309-3064382539-2414
License Entitlements	Displays a list of License Entitlements for the current group. A License Entitlement consists of a Product, License Type and Limit. License Entitlements are uniquely identified by a combination of Product and License Type. Which means a Group can be defined with both a Full and Academic license type for a single Product.
Add	Selecting this button will display the Group License Entitlement screen in Add mode, and will allow the creation of a new License Entitlement for the current Group.

Edit	Selecting this button will display the Group License Entitlement screen in Edit mode and load the details of the selected Entitlement. This button is only enabled when an existing Entitlement is selected in the List of Entitlements.
Delete	Selecting this button will delete the entire definition of the currently selected Entitlement from the current Group. This button is only enabled when an existing Entitlement is selected in the List of Entitlements.

Default Passwords for Floating License Server

When the Pro Cloud Server is installed the Floating License Server's configuration will come pre-loaded with a number of sample groups for demonstration purposes.

Sparx Systems recommend that all group passwords should be changed when first configured.

Note: For new installations the **admin** account details will be needed to make the initial connection so that adjustments to the settings can be made to suit your preferences.

Passwords

The initial passwords for the example groups are:

Group Name	Password
admin	password
default	password
javerage	Average
user_group2	group2
user_group3	group3

Manage Group License Entitlement

Managing the definition of a License Entitlement is performed using the 'Group License Entitlement' screen, which will be shown when either the Add button or Edit button is selected on the Manage Group Details screen.

License Entitlements grant users of the current group the ability to 'check-out' floating licenses for the defined Product and License Type combination. Consider the example definition of user_group3. Any user connecting to the Floating License Server using this group will have the ability to 'check-out' licenses for these products:

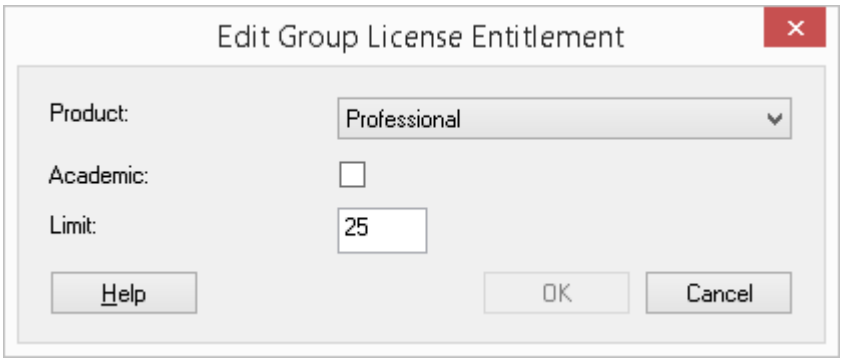
- Enterprise Architect Professional Edition
- Enterprise Architect Corporate Edition
- The Eclipse Integration Plug-in
- Enterprise Architect Corporate Academic Edition

As can be seen in the example, License Entitlements are uniquely identified by a combination of Product and License Type. Notice the two entitlements for the Corporate Edition, both regular and academic.

This is a Group License Entitlement screen in 'Add' mode:



This is a Group License Entitlement screen loaded with the Professional entitlement assigned to the example user_group3 group that is created when the PCS is first installed:



Item	Description
Product	The name of the product (Enterprise Architect edition or Plug-in) that this entitlement gives access to for users of the current group.
Academic	Determines if the current entitlement implies that academic licenses of the selected product are allowed for the current group.
Limit	A optional numeric value representing the maximum of concurrent users of the current group that can check-out License keys for the defined Product/License

	Type. The default is an empty value and represents that the definition imposes no limit to the number of license that can be checked out by the current group. This option if particularly useful if departments within a single company purchase their own floating licenses, groups can be setup for each department and a limit applied to restrict the group to the number of licenses that their department purchased.

Troubleshooting

If users experience a problem with the Pro Cloud Server's Floating License Server, the error message will either be displayed on the screen for errors related to the Floating License Configuration Client, or written to the Pro Cloud Server's log file when an error occurs within the Pro Cloud Server. This topic provides a list of common Floating License-related errors that could occur. Each entry describes the situation that has occurred and provides guidance on how to correct the error.

Configuration-related Floating License Server errors

Issue	Comments
EA Users are not able to connect to a Floating License Group	Enterprise Architect users cannot connect to a defined Floating License Group, even when the entered details exactly match the definition. Possible reasons: • If an entitlement in the Floating License Group file is defined with an invalid Enterprise Architect product, the entire Group will be disabled; check the Pro Cloud Server's log file for error messages - the invalid product

	will be mentioned by name and line number
The Key Store connection screen in EA does not include HTTPS option	In order for Enterprise Architect to connect to the Pro Cloud Server's Key Store, a protocol of HTTPS is required for secure communications. Possible reasons: • A version prior to Enterprise Architect version 14 is in use
The Key Store connection screen in EA cannot perform a successful 'Test'	Your client is not able to communicate with a Pro Cloud Server using the supplied details. Possible reasons: • Pro Cloud Server is not installed • Pro Cloud Server is not running • Pro Cloud Server is not listening on the expected Port using HTTPS • There is no network connectivity between the client and server • A firewall is limiting communications between the client and server

Errors while using Floating License

Configuration Client

Issue	Comments
Could not open keystore at [FilePath]. The file does not exist and could not be created	The service was unable to open the key store file specified by the FLOATLIC_PATH property in the Pro Cloud Server configuration file (SSProCloud.config). Ensure that this path exists, and the user account that the Pro Cloud Server Windows Service is running under has the necessary permissions to read and write to the specified file.
The Floating License Configuration Client is not able to connect	The machine running the Floating License Configuration Client is not able to communicate with a Floating License Server using the supplied details. Possible reasons: • Pro Cloud Server is not installed • Pro Cloud Server is not running • Pro Cloud Server is not listening on the expected Port using HTTPS • There is no network connectivity between the client and server • A firewall is limiting communications

	between the client and server • The Pro Cloud Server configuration file does not contain a valid PCS license key, therefore all Pro Cloud features are disabled, including the Floating License Server • The entered user credentials do not match any of the defined Floating License Groups in 'ssflsgroup.config' • The user credentials for the entered Floating License Group have not been defined as 'IsManager=true' • If an entitlement in the Floating License Group file is defined with an invalid Enterprise Architect product, the entire Group will be disabled; check the Pro Cloud Server's log file for error messages - the invalid product will be mentioned by name and line number
Could not open keystore, no keystore file specified	No file path was specified in the property in the configuration file. Type a file path into the FLOATLIC_PATH configuration property that the user account the Pro Cloud Server Windows Service is running under can read and write to.

Invalid or missing keystore file - Keystore file cannot be opened by this version of the service or the file has been corrupted	The keystore file specified by the configuration property FLOATLIC_PATH is either incompatible with the current version of the Floating License Server, or has somehow been corrupted on the file system. Restore the key store file specified in the configuration property FLOATLIC_PATH from a recent backup, or contact Sparx Systems Support for assistance.
Key file has been moved	The key store file specified by the configuration property FLOATLIC_PATH is locked to the serial number of the hard drive it is created on. If the key store file is moved from this hard drive, then other Floating License Servers are unable to open it. If the key store file has been moved, restore the file to its initial location. Certain RAID configurations can affect how the hard drive's serial number is presented to the Floating License Server. Thus, it is recommended that you house the key store file on a non-RAID drive, wherever possible.

Errors while using Enterprise Architect and the Floating License Server

Issue	Comments
Checkout Failure: There are no more available keys for this product in the key store	Keys for this product exist; however, they are all checked out to other users. If this error is frequently reported, consider limiting the use of the product across your enterprise or purchasing more keys to meet the demand for the product.
Checkout Failure: The key store does not contain any keys for this product	This keystore does not contain keys for the requested product. Keys for the product might have been provided by the keystore in the past, but have since been removed. Uninstall the product and return any shared keys on the client machine.
Checkin Failure: Key not found in keystore.	The key being checked in has either been deleted since it was checked out, or was checked out from another keystore. Ensure that users return any shared keys to the key store they were leased from before swapping Floating License Servers

	/ key stores.

Creating a Cloud Repository

This topic covers the steps for creating a Cloud-based repository. These steps vary depending on:

- Whether you will be creating a new repository or using an existing repository
- The type of repository/database that is to be used (feap or DBMS)

The sections provide only a brief overview of the process in each case, and assume you have first Pro Cloud Server Setup. For more information please refer to the links in the *See Also* column.

Creating a new Firebird Cloud Repository

Step	Action
1	Creating a new Firebird repository is the simplest of all supported repository types, and can be created directly from the Pro Cloud Server Configuration Client (either using the Windows Application or WebConfig).

Creating a new DBMS Cloud Model

Step	Action
1	Create a new Server Based Repository (DBMS) as described in the Server Based Repositories topic. For the best performance, the DBMS repository should be either on the same machine or same Local network as the Pro Cloud Server. The faster the Pro Cloud Server can retrieve data from the database the better the end user's performance will be.
2	Add a Pro Cloud Server Database Manage which connects to the DBMS repository.

Using an existing Firebird Model

Step	Action
1	Copy the feap file to the Models directory in the PCS installation location on your server (e.g. C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service\Models).
2	Add a database manager which connects

	to the model. You can do this using either the Pro Cloud Server Configuration Client application or WebConfig.

Using an existing DBMS repository

Step	Action
1	Simply add a database manager which connects to the DBMS repository. You can do this using either the Pro Cloud Server Configuration Client application or WebConfig.

Using an existing EAP/EAPX file

Step	Action
1	The Pro Cloud server cannot directly access an EAP/EAPX repository, however the content of an EAP/EAPX repository can be transferred to a feap (Firebird), or DBMS repository, using the Project Transfer feature in Enterprise Architect.

2	Add a database manager which connects to the feap or DBMS repository.

Creating a Cloud Repository on AWS

Pro Cloud Server repositories can be hosted on an AWS EC2 server or a similar third-party Cloud hosting service.

- Either feap or DBMS could be used in this case
- For the best performance the Pro Cloud Server and the database/repository should be on the same server in the same network
- When moving an existing local repository to AWS, the project transfer should be performed while running Enterprise Architect on the AWS server

For example:

1. Copy your existing .feap repository to the AWS server.
2. Install Enterprise Architect on the AWS server (if it is not already installed).
3. Initiate an Enterprise Architect transfer from the .feap repository to your target database.

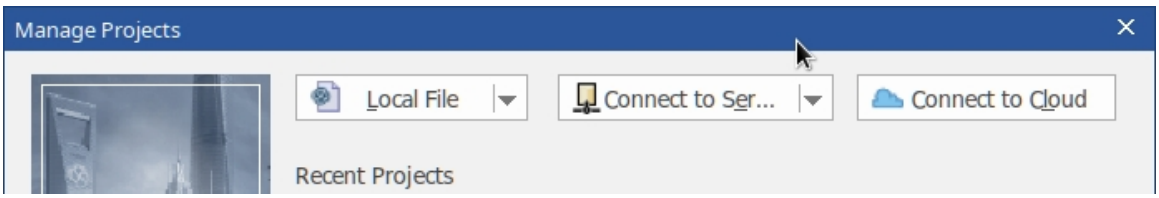
Or if using a DBMS:

1. Perform a complete backup of the original database, using the DBMS's supplied tool.
2. Copy the backup to the AWS server.
3. Restore the backup to the target database.

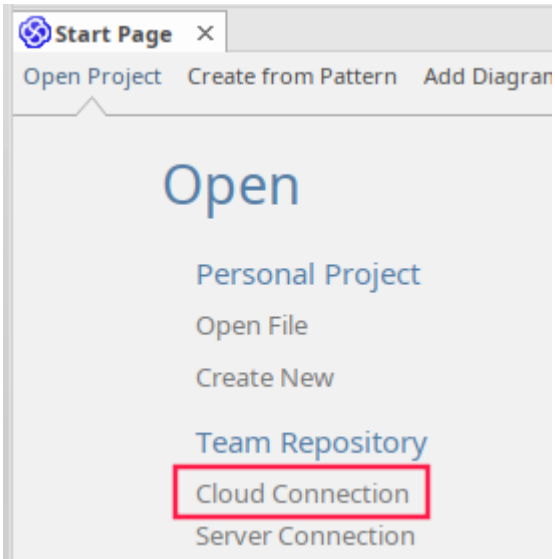
.

Connecting Enterprise Architect to a Cloud Model

Once your Pro Cloud Server has been set up with at least one Port listening for communications and at least one model that is enabled, you can connect to that model from 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.

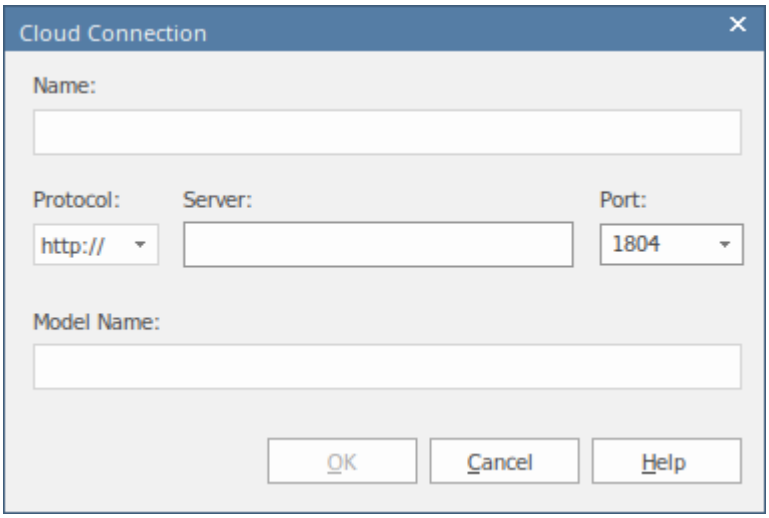


In later work sessions you can access this dialog and button by selecting the 'New | Manage projects' option on the Start Page. However, there is also a Cloud Connection link on the 'Start Page' itself.



Cloud Connection Dialog

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



Option	Description
Name	Type a name for the project/repository that is meaningful to you (or your environment), it does not have to match any values on the server. The name entered will be shown in the list of recent projects.
Protocol	Click on the drop-down arrow and select the protocol to use to connect to the server. When you select the protocol, the Port will be automatically set to match the default for that protocol. • https should be preferred for most users as it allows for both authenticating the server and protecting all communications with the server; this

	requires an SSL certificate • http is not protected; it can be used for model connections, and is best used to test connections before setting up the server certificate • tcp allows connecting directly to the server over its admin Port; using this method eliminates many of the benefits provided by the other two protocols
Server	Type in the machine name (sparxcloud.com) or IP address (192.168.0.20) of the server hosting the Pro Cloud Server.
Port	Enter one of the Port number(s) the Pro Cloud Server service is listening to. Ensure that both the protocol and Port match what is defined in the Pro Cloud Server. Note: • For a PCS 4.x server or earlier, the default settings will use Port 804 for http and Port 805 for https • For a PCS 5.0 server or later, the default settings will use Port 1804 for http and Port 1805 for https
	Type in the Alias for the repository as

Model Name	defined in the Pro Cloud Server Configuration Client. For more details see 'Alias' in the *Database Managers* Help topic. Note: Generally a system administrator will provide you with a list of available models. However, for those with the ability to use the Sparx Systems Pro Cloud Server Configuration Client, the model name to access the given database is the Database Manager's 'Alias' field on the Configure Database Manager screen.

Pro Cloud Server Features

As mentioned in the *Introducing Cloud Repositories* Help topic, even without a License the Pro Cloud Server can be used for hosting Cloud Repositories to be accessed by Enterprise Architect. When using a licensed Pro Cloud Server a number of additional features are available, as outlined here.

Feature	Description
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. WebEA provides 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 action can be performed.

Integration Plug-ins	Also called Server Based Plug-in Interface (SBPI), this is a framework developed to extend the Pro Cloud Server via Plug-ins. The Integration Plug-ins allow you to link Enterprise Architect elements to elements or objects in external products.
Visibility Levels	The 'Visibility Levels' feature within Enterprise Architect gives 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 and Microsoft SQL Server.
Prolaborate	Prolaborate represents the next step in effectively delivering rich models and designs to the broader community and in enabling high level executives, business process managers, process implementers, consultants, customers and other Stakeholders to see their personal and curated view of the model in a way that they are personally comfortable with! Prolaborate is a tailored set of views that reduce complexity, focus attention and

	increase the accessibility of model information for non-modelers and others who are more concerned with consuming the model than directly editing it. Prolaborate provides a social platform 'outside the model' that supports talking about the model from a different viewpoint to those actively engaged 'within the model'. Prolaborate also provides dashboards, impact analysis and other 'outside the model' features that derive information from the model or provide unique windows into the model for a custom audience, such as Chief Executive Officers, Chief Information Officers and customers.
WebConfig	The Pro Cloud Server 'WebConfig' is a web-based configuration client, which can be used as an alternative to the Windows-based 'Configuration Client'. The WebConfig allows you to add and configure Cloud connections, set up integration Plug-ins, manage floating licenses, and view Pro Cloud Server logs, all from a Web Browser.
OSLC RESTful API	The OSLC RESTful API (OSLC Architecture Management v2.0) provides

	direct access to a wide range information from your Enterprise Architect Cloud Repositories.
OSLC Requirements Management	OSLC is an initiative to provide easier integration between Requirement Management tools. It uses HTTP to list, add, modify and delete requirements.

WebEA

Sparx System's WebEA is an application designed to display 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 or IIS) 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 one of the *WebEA Quick Start Guides* for the necessary steps.

Requirements

- Users of WebEA require an 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 the 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 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
- WebEA will access one brand new Firebird model (because it is the simplest, as it does not require any third party ODBC drivers)
- The Enterprise Architect user base is using Enterprise Architect 13.5 or later

1. Download the Pro Cloud Server

Step	Description
1.1	The Pro Cloud Server installer can be downloaded from the 'Registered User' section of the Sparx Systems website.

	https://sparxsystems.com/registered/reg_ea_down.html Note: You will need valid credentials to access this area of the website.

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 your 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 Server 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\Pro Cloud Server\' folder and added a shortcut to the desktop.

Ste	Description

p	
5.1	Run the 'Pro Cloud Config Client' shortcut that was added to your desktop.
5.2	Enter the Server address: localhost:1803 Enter the default password: pcsadm1n Click on the OK button. Note, if you are using PCS version 4 or earlier, the default password is blank (leave the password field empty).
5.3	Click on the Licensing button, then on the Create Request button, fill in your company and purchase details, save the License Request and email it to the Sparx Systems Sales team (sales@sparxsystems.com). When the Sparx Systems Sales team return your License Certificate, click on the Add button and select the supplied certificate file, the Pro Cloud Server should now have a valid License loaded.
5.	Select the 'Ports' tab.

4	Select Port '1804' from the list and click on the Edit button. Tick the 'OSLC Supported' checkbox, then click on the OK button.
5.5	Select the 'Database Managers' tab, then click on the Add button. Type the name 'fb-procloud.feap' in the 'Connection String' field, then click on the OK button. Select the new entry in the list and click on the Edit button. On the Configure Database Manager screen, select the 'Enabled' and 'Enable Pro Features (OSLC, WebEA and Integration)' checkboxes. Click on the OK button. For a more in-depth explanation of how to add Database Managers, see the Add Database Manager - ODBC Help topic.
5.6	Open the standard Windows Services console (command 'services.msc'), locate and select the service 'Sparx Systems Professional Cloud', then restart it.

6. Copy WebEA files to Website

Step	Description
6.1	Copy the folder C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\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_db_alias = "fb-procloud" • sscs_port = "1804" • sscs_model_user = "" • sscs_model_pwd = "" • auth_code = "password01"
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

Ste	Description

p	
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 1804 and 1805.

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 v13.5 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: 1804 • Model Name: fb-procloud Click on the OK button.
9.5	Open the 'Settings > Model > Options > Cloud' options and tick these checkboxes: • Auto create Diagram Image and Image Map (each time a Diagram is saved) • Auto create HTML Page (each time a Linked Document is saved) These options ensure diagrams and Linked Documents are saved in a format that allows them to be viewed in WebEA.
9.6	Create a sample set of Model Views, Packages, diagrams and elements, so that they can be viewed in WebEA later.

10. Test WebEA access

Ste	

p	Description
10.1	On a client machine, open a web 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
10.2	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.
10.3	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.
10.4	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.

WebEA Quick Start Guide - IIS

The purpose of the *WebEA Quick Start Guide - IIS* is to provide the complete process for quickly installing and configuring WebEA in a new IIS environment using the most common options.

The process is based on these assumptions:

- A clean Windows Server 2012 (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
- IIS will be used as the Web Server
- WebEA will access one brand-new Firebird model (because it is the simplest, as it does not require any third party ODBC drivers)
- The Enterprise Architect user base is using Enterprise Architect 13.5 or later

1. Download the Pro Cloud Server

Step	Description
1.1	The Pro Cloud Server installer can be downloaded from the registered user section of the Sparx Systems website. https://sparxsystems.com/registered/reg_ea_down.ht

	ml Note: You will need valid credentials to access this area of the website.

2. Install the Web Server

Step	Description
2.1	For details on IIS and PHP installation, refer to the following link: https://learn.microsoft.com/en-us/iis/application-frameworks/scenario-build-a-php-website-on-iis/configuring-step-1-install-iis-and-php
2.2	Install IIS as described in the link above. **Note**: As mentioned in the link, ensure you enable CGI during the installation.
2.3	Install PHP as described in the "Download and Install PHP Manually" section; however, skip the download/extraction of the WinCache extension (Steps 2 - 4).
2.4	Check that PHP was installed successfully by entering the following at the Windows Command

	Prompt: *php -info* If this returns an error you may need to install the latest VS C++ redistributable (See: https://answers.microsoft.com/en-us/windows/forum/all/vcruntime140dll-was-not-found/f43d5afd-9239-4913-8f32-20e44a959e13).
2.5	Test PHP as described in the "To test your PHP installation" section.
2.6	Now enable the PHP extensions which are required for WebEA. Open the folder where you installed php (e.g. C:\PHP\) and edit the php.ini file. Find each of the following lines and remove the comment character (;). ;extension=curl ;extension=gettext ;extension=mbstring ;extension=openssl **Note:** • With a new installation you may find you have php.ini-development and php.ini-production files, but no php.ini file. In this case copy the php.ini-production file and rename it to php.ini. • You may also need to uncomment the line ";extension_dir=" and adjust the path to the extensions folder.

3. Configure the Web Server

With the default IIS configuration, the default website will use Port 80. Its base folder is %SystemDrive%\inetpub\wwwroot, which is normally C:\inetpub\wwwroot.

Step	Description
3.1	No additional configuration is required. Test the connection by browsing to the server address from another computer. The 'Internet Information Services' landing page should display. **Note:** If you wish to use Windows Authentication for WebEA then IIS should be configured as described in the 'Configuring IIS for WebEA Windows Authentication' topic.

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 Server Setup' dialog displays. Click on the Next button.
4.3	Read the license agreement and select the checkbox to indicate 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\Pro Cloud Server\'

folder and added a shortcut to the desktop.

Step	Description
5.1	Double-click on the 'Pro Cloud Config Client' shortcut that was added to your desktop.
5.2	Enter the Server address: localhost:1803 Enter the default password: pcsadm1n Click on the OK button. Note, if you are using PCS version 4 or earlier, the default password is blank (leave the password field empty).
5.3	Click on the Licensing button and then on the Create Request button; fill in your company and purchase details, save the License Request and email it to the Sparx Systems Sales team (sales@sparxsystems.com). When the Sparx Systems Sales team return your License Certificate, click on the Add button and select the supplied certificate file; the Pro Cloud

	Server should now have a valid License loaded.
5.4	Select the 'Ports' tab Select Port '1804' from the list and click on the Edit button. Tick the 'OSLC Supported' checkbox then click on the OK button.
5.5	Select the 'Database Managers' tab, then click on the Add button. Type the name 'fb-procloud.feap' in the connection string field, then click on the OK button. Select the new entry in the list and click on the Edit button. On the Configure Database Manager screen, select the 'Enabled' and 'Enable Pro Features (OSLC, WebEA and Integration)' checkboxes. Click on the OK button. For a more in-depth explanation of how to add database managers, see the Help topic Add Database Manager - ODBC.
5.6	Open the standard Windows Services console (command 'services.msc'), locate and select the service 'Sparx Systems Professional Cloud', then restart it.

6. Copy WebEA files to Website

Step	Description
6.1	Delete the default IIS files from C:\inetpub\wwwroot
6.2	Copy the folder C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\WebEA\ to C:\inetpub\wwwroot.
6.3	Restart IIS (in IIS Manager, select the PC name from the connection list, then within the "Actions" panel click on the "Restart" option).

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:\inetpub\wwwroot\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_db_alias = "fb-procloud" sscs_port = "1804" sscs_model_user = "" sscs_model_pwd = "" auth_code = "password01"

7.6	Save the changes to the file webea_config.ini in C:\inetpub\wwwroot\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 1804 and 1805.

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 v13.5 or later.
9.2	From the Start Page select 'Manage Projects' 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: 1804 Model Name: fb-procloud Click on the OK button.
9.5	Open the 'Settings > Model > Options > Cloud' page and tick these checkboxes: • Auto create Diagram Image and Image Map (each time a diagram is saved) • Auto create HTML Page (each time a Linked Document is saved) These options ensure diagrams and Linked Documents are saved in a format that allows them to

	be viewed in WebEA.
9.6	Create a sample set of Model Views, Packages, diagrams and elements, so that they can be viewed in WebEA later.

10. Test WebEA access

Step	Description
10.1	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
10.2	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.
10.3	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.
10.4	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 on IIS. 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.

WebEA Installation and Configuration

The Sparx Systems Pro Cloud Server product is installed using a standard Microsoft 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 default settings are used WebEA will be installed into the C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\WebEA\ folder (assuming a 64 bit machine is being used).

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

If you have already have setup WebEA and are wanting to update to a newer versions, refer to the Updating WebEA to a New Version topic.

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 (Apache or IIS) capable of hosting PHP v8.0 (or later)
- 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
- Connectivity 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 *Pro Cloud Server Installation* Help topic, making sure that the WebEA component was installed.

In addition 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 ID 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 element tests? • Should WebEA users be able to edit element resource allocations? • What is the default list of watchlist items for the model?
2	Use the Cloud Configuration Client and define a Port to be used for access to your Cloud Models. Ensure that the 'OSLC Supported' option has been set to true. After adding or modifying Port configurations, ensure the Windows service 'Sparx Systems Professional Cloud' is restarted.

3	Ensure a database manager is defined for each model decided on in step 1, in the Pro Cloud Server's database manager list. The process to do this is as outlined in Sparx Systems Pro Cloud Server Configuration Client, in *Add Database Manager*.
4	For each of the models decided on in step 1, select the model on the 'Sparx Systems Pro Cloud Server Configuration Client', click on the Configure button and select the 'Enable Pro Features (OSLC, WebEA and Integration)' checkbox.
5	Ensure that your Pro Cloud Server has a valid License, loaded by completing these steps: 1. Click on the Licensing button. 2. Click on the Create Request button. 3. Fill in your company and purchase details. 4. Save the License Request and email it to the Sparx Systems Sales team (sales@sparxsystems.com). When the Sparx Systems Sales team return your License Certificate: 1. Click on the Add button. 2. Select the supplied certificate file. The Pro Cloud Server should now have a valid License loaded.
6	Check that you can access the model(s) in Enterprise

	Architect using the 'Connect to Cloud' option. After opening each model in Enterprise Architect, enable the 'Data Cache' options and click on the Create button to generate the initial set of viewable diagrams and Linked Documents. See the *Model Configuration* topic for details.
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 v8.1.5 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,

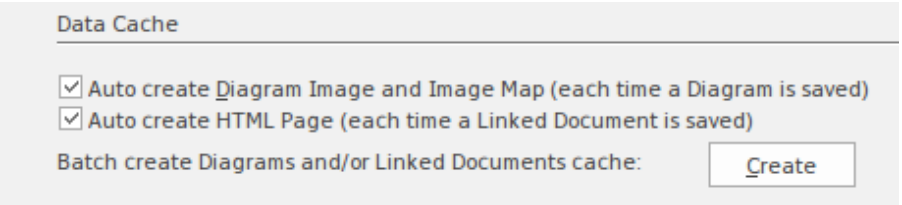
	deciding what Port the web server will listen to, and making sure the website is secure from external attacks, are all outside the scope of this documentation. **Tip:** Using the default XAMPP configuration, the Port for the default website is 80 and the base folder is at 'C:\xampp\htdocs'. XAMPP will 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\Pro Cloud Server\WebEA\') to the base folder of the website. **Tip:** Copy the WebEA folder from 'C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\WebEA\' to 'C:\xampp\htdocs'. **Special Note regarding reinstalls:** Before updating the WebEA files, take a backup of the files you have modified; that is, \includes\webea_config.ini and \includes\webea_strings.ini.
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, start or restart 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 the '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. **Special Note regarding reinstalls:** If you a have a previous backup of the \includes\webea_config.ini file and wish to use the same configuration, you can simply restore the backup file.
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 models defined in the webea_config.ini file.
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	By default, diagrams and Linked Documents are not saved in a format that

Architect model options	is viewable by web 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 'Settings > Model > Options' ribbon option. On the 'Manage Model Options' dialog, refer to the 'Data Cache' options on the 'Cloud' page.  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 are 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 component*s 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 (using the sscs_model_user and sscs_model_pwd defined in the 'webea_config.ini' file)
Single Sign-On (SSO)	WebEA supports these Single Sign-On Systems: • Windows authentication with Active Directory • OpenID Provided the Enterprise Architect model has been configured for Single Sign-On, users will have the option to use Single Sign-On when accessing the model via WebEA. **Note:** The IIS Windows Authentication setting can cause a conflict with WebEA's Windows Authentication option. If you are using IIS and wish to use WebEA's 'Login with Windows ID' then the WebEA site itself must have Windows Authentication set to Disabled.

Configuring your WebEA site for HTTPS (SSL)

As there are many different web servers and configurations,

configuring your WebEA site for HTTPS access is outside the scope of this documentation.

Please refer to the relevant documentation for your web server. For example, the Apache "Apache SSL/TLS Encryption" documentation or the Microsoft "How To Set Up an HTTPS Service in IIS" help topic.

Configuring the Pro Cloud Server to use HTTPS (SSL)

As with Enterprise Architect, WebEA can access models either via HTTP or HTTPS. Using HTTPS requires you to have a Port configured for HTTPS access as described in the Add a Port Definition topic.

In addition, if you are using your own CA to sign certificates, then also see the Configuring WebEA to Trust Your Own Certificate Authority (CA) Help topic.

Acknowledgements

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

- jQuery v3.5.1 - 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

To view diagrams and linked documents in WebEA, they must first be saved in an appropriate format. There are two ways to achieve this:

1) In Enterprise Architect, open the model which will be accessed via WebEA and enable these 'Data Cache' options:

- Auto create Diagram Image and Image Map (each time a Diagram is saved)
- Auto create HTML Page (each time a Linked Document is saved)

Refer to the WebEA Installation and Configuration topic for details.

And/Or...

2) Enable the Pro Cloud Server's 'EA Worker' for the model connection. The EA Worker is a custom version of Enterprise Architect which runs on the server and automatically generates WebEA viewable diagrams and linked documents when necessary.

Typically it is recommended to enable the 'Data Cache' options for all models which will be accessed via WebEA. Optionally you could also choose to run the EA Worker application. This Help topic covers the EA Worker in more

detail.

Benefits of the EA Worker

The Data Cache options for saving diagrams and linked documents in WebEA format were introduced in Enterprise Architect 13.5 (Build 1350). If users will be updating your model using an earlier build of Enterprise Architect, then the diagrams and linked documents will not be saved/updated for viewing in WebEA. In this case you can (and should) run the EA Worker on your server to ensure that all updates are reflected in WebEA.

Enterprise Architect's option for saving diagram images ('Auto create Diagram Image and Image Map') saves the diagram image only when that diagram is modified directly within Enterprise Architect. However, in the case of the EA Worker, if you modify a diagram indirectly (e.g. you rename an element that appears on multiple diagrams), then the Worker will update/save the image for all the relevant diagrams. For this reason you might choose to enable the EA Worker, even when you know users will be modifying the model using a recent (post 13.5) version of Enterprise Architect.

Performance Considerations

A separate instance of the EA Worker application (EAWorker.exe) will be running on your server for each model that has the EA Worker enabled. For the best

performance on your server, the EA Worker should only be enabled where it is necessary.

If you have multiple large/high traffic models that require the EA Worker then it might be necessary to host models in a separate Pro Cloud Server instance on a different server (reducing the number of EA Worker processes running on each).

How to run EA Worker

The EA Worker can be enabled either via the Configure Worker screen in Pro Cloud Server's Configuration Client or the WebConfig - Edit Model Connection page in WebConfig. Once the EA Worker has been enabled, the Worker process (EAWorker.exe) will be started the next time the Pro Cloud Server service is started (or restarted).

In early versions of PCS (prior to v4.1) the only way to run the EA Worker was with a Windows Batch/command line, using either a manually entered connection string or a connection string saved as a .EAP file. See the Manual EA Worker Configuration topic for details.

Using MDG Technologies with the EA Worker

If you make use of MDG Technologies in your model then the EA Worker might need access to them so that the

generated diagrams appear as expected (e.g. making use of shape scripts).

There are two options for enabling an MDG Technology for the EA Worker:

1) Using Enterprise Architect, import the MDG Technology into the model. See the Import MDG Technologies to Model Help topic for details.

2) Copy the MDG Technology file into the 'MDGTechnologies' folder within the PCS installation. E.g. "C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service\Worker\MDGTechnologies". In this case the MDG Technlology will be enabled for all models which are using the EA Worker on this machine/server.

After either of these changes the 'Sparx Systems Professional Cloud' service should be restarted for the change to take effect.

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.

EA Worker Troubleshooting

The EA Worker application runs as a background task, without any user interface. If you have issues with the EA Worker (e.g. diagram images are not being generated), then you can troubleshoot this by referring to the EA Worker logs as described below.

Steps	Details
1	If you have not already, enable logging for the EA Worker. E.g. Set the Logging Level to 'System'.
2	On the server, check the Windows Task Manager ('Details' tab) and confirm that the EAWorker.exe process is running.
3	End the EAWorker.exe task/s. In some cases there could be multiple EAWorker.exe tasks running for a single model connection.
4	Restart the "Sparx Systems Professional Cloud" service.
5	Referring to the Task Manager, you should see that one or more EAWorker.exe tasks is started. There should be one for each Model Connection which has the EA Worker enabled.
6	Restarting the Service will have created a new log file on the server. The Worker log files can be found

	within the Pro Cloud Server's installation directory. E.g. C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service\Worker\Logs Find the log file that corresponds with the model having issues. E.g. EAWorker-myModel-1.log Open the log and check for errors. The error message itself might provide enough details to identify and resolve the issue. The error might also correspond to one of the error messages covered in the *'Pro Cloud Server Troubleshooting'* Help topic.

Manual EA Worker Configuration

If you are using the Pro Cloud Server version 4.1 or later then the EA Worker should be configured and run via the Worker Configuration screen in Pro Cloud Server's Configuration Client or the Edit Model Connection page in WebConfig. Refer to the How to configure automatic viewable components topic for details.

In early versions of PCS (prior to v4.1) the only way to achieve this was with a Windows Batch/command line, using either a manually entered connection string or a connection string saved as a .EAP file.

For historical purposes the manual approach is described here:

How to start the EA Worker manually

All EA Worker command lines require at least 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:\> EAWorker.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.

This is 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 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 Step 1: "C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Worker\EA.exe" /cloudupdate:{seconds} "{filepath}"
3	Run the batch/command file. Double-click on the file created in Step 2. Sparx System recommends that this batch/command file is added to Startup so that it is automatically run

	when the server starts.

How to configure WebEA models

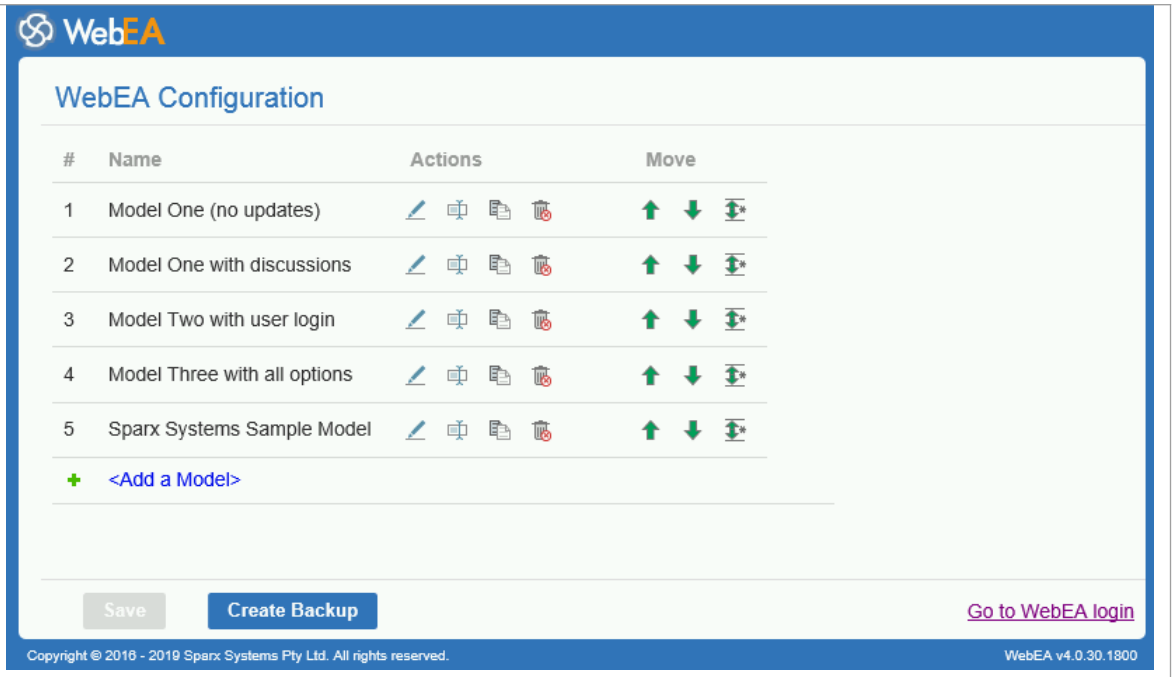
All WebEA configuration is controlled by a single file 'webea_config.ini', which must reside in the WebEA '\includes\' folder.

This configuration file defines the model connections that are available on WebEA's login page. For each WebEA model connection you define how WebEA will access the model (which server, which Port and so on), and optionally apply settings to adjust WebEA's functionality (such as showing or hiding specific features).

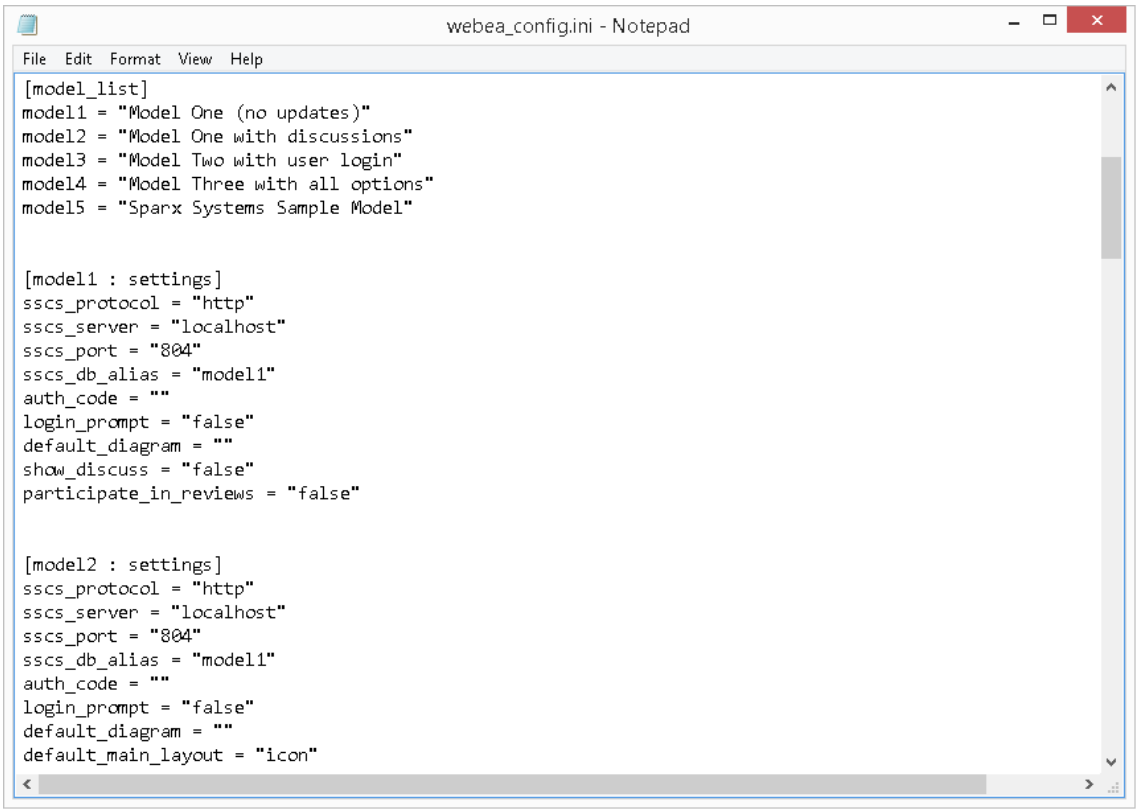
Configuration Options

There are two different methods available for configuring WebEA models:

Step	Description
1	(Recommended) Use the web based configuration page (config.php).



2 Directly modify the 'webea_config.ini' file using a text editor.



Security Considerations

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.

For example, in Apache this could be achieved using .htaccess and .htpasswd files in the '\includes\' folder.

Configure WebEA models - via Web Browser

After copying the WebEA files from the Pro Cloud Installation to your website, you can configure WebEA via a web browser.

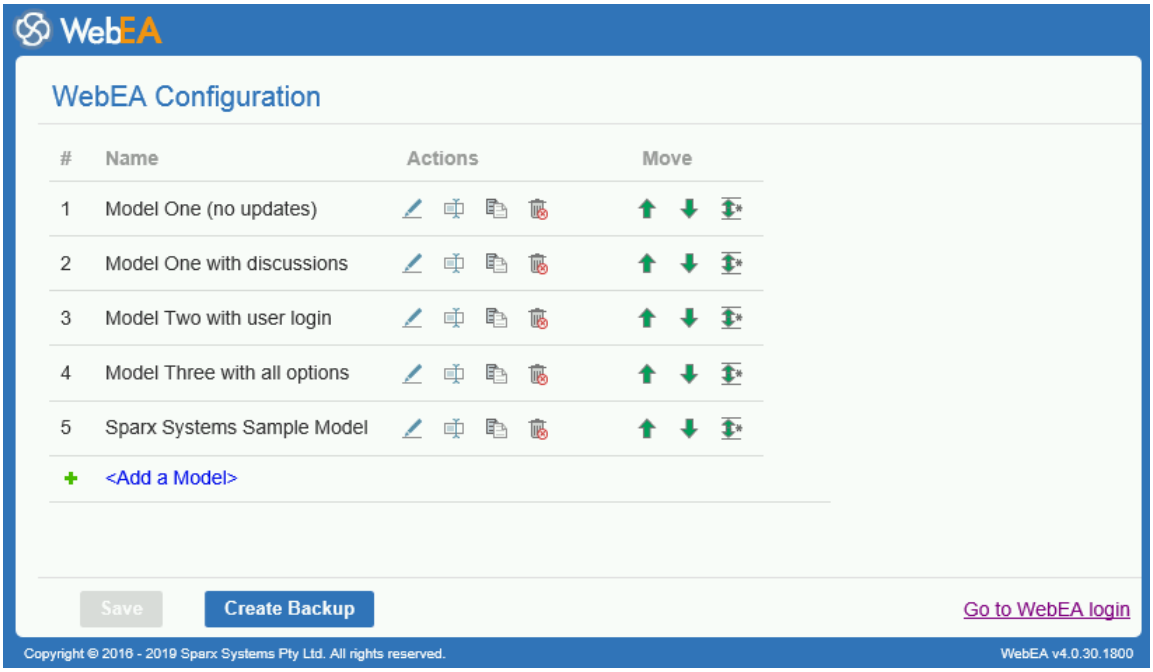
Using the web browser, enter the URL for WebEA's login page and, at the end of the URL, append '?config' after 'login.php'.

For example:

- http://{yourwebserver}/login.php?config or
- http://{yourwebserver}/webea/login.php?config

The value '{yourwebserver}' is either the name or the IP number of the machine hosting the WebEA website.

This should display the 'WebEA Configuration' page, which includes a list of the WebEA connections that are currently defined in the webea_config.ini file.



WebEA Configuration

The 'WebEA Configuration' model list provides these fields.

Field	Description
Model #	This field displays the model number for this WebEA model configuration. The model number is used in hyperlinks to specific elements within this model. If necessary the model number can be changed by using the 'Change Model Number' button in this dialog; however, typically this should not be necessary. Changing the model number could cause existing hyperlinks in the model to stop working as expected, therefore take care when using this option.

Name	The display name for the WebEA Model connection. This appears in the WebEA login page, and in the WebEA header after opening the model.
Actions	The 'Actions' column contains a number of buttons for modifying the model connection. • **Edit -** Allows you to edit/configure the options for this WebEA model connection • **Rename -** Allows you to change the WebEA display name • **Copy -** Creates a duplicate of this WebEA model connection; the duplicate will be added to the list of models with the name: *{original name} - Copy* • **Delete -** Delete this WebEA model connection
Move	The 'Move' column contains options for moving the model connection. • **Move Up -** Move the model connection up one position in the list • **Move Down -** Move the model connection down one position in the

	list • **Move to... -** Move the model connection to a specific location in the list
Add a Model	The final list item is the option to add a new model connection. When selecting this option you will be prompted to enter the connection name or WebEA display name. After you enter the name, the page to edit the Model Connection settings is displayed.

Saving the WebEA configuration

Changes made via the 'WebEA Configuration' page (including changes when editing a specific model connection) are not saved automatically.

To save changes, click on the **Save** button at the bottom of the 'WebEA Configuration' page.

Note, the Save button is only enabled when there are unsaved changes.

Creating a backup of the webea_config.ini file

The Create Backup button at the bottom of the 'WebEA Configuration' page will create a backup of the current webea_config.ini file.

Backups are created with the name:

webea_config_bak_{date}_{time}.ini

Note: A backup cannot be created while there are unsaved changes (if necessary click on the Save button, then on the Create Backup button).

Edit WebEA Model Connection settings

After you select the option to edit a Model Connection, the configuration options for that connection are displayed.

In the simplest case, the connection can be defined by only updating the 'Protocol', 'Server', 'Port' and 'Model Name/Alias' fields. A number of additional options are provided for configuring settings such as authentication, available functions and display options.

Connection Options

Item	Description
Protocol	Identifies the protocol to be used to connect to the Sparx Systems Pro Cloud Server.
Server	Identifies the fully qualified machine name or IP address of the server hosting the Pro Cloud Server.
Port	Defines the TCP Port used to connect to the Sparx Systems Cloud Service. This Port number must match one of the Ports that the Pro Cloud Server Windows Service is listening to or, in the case of an

	IIS setup, the IIS Port servicing this repository.
Model Name/Alias	Defines the name or alias of the model connection. This corresponds to the Database Manager's 'Alias' in the Pro Cloud Server Configuration Client or the 'Model Name' on the Enterprise Architect's 'Cloud Connection' screen. The value is case sensitive.
Validate SSL Certificates	Defines whether self signed SSL certificates can be used. If you are using HTTPS and a self signed certificate, then this should be set to 'No'. Note: This option is only applicable when the Protocol is set to HTTPS.
OSLC Access Code	When configuring a Pro Cloud Server model connection / database manager an optional 'OSLC Access Code' can be defined. If the model connection has an 'OSLC Access Code' assigned then it will need to be entered in this field to allow WebEA access.

Authentication Options

Item	Description
Authentication 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 is only used for basic username/password authentication and is not used for Windows Authentication or OpenID. It is possible to set this option and also set a Model User. In this case the user will be required to enter the access code only and will not be prompted for a username or password.
Model User	Defines the model user login (as defined in Enterprise Architect User Security) that will be used in combination with the 'User Password' option to log in to a security-enabled model. • When set, all other authentication schemes are bypassed, including Windows Authentication, OpenID and the basic username/password login

	prompt • Sparx Systems would recommend the use of this option only on a model that has been configured as read only, since all updates made by WebEA users in this configuration will be identified as the same user
User Password	Defines the password (as defined in Enterprise Architect User Security) that will be used in combination with the 'Model User' option to login to a security enabled model. • When set, all other authentication schemes are bypassed, including Windows Authentication, OpenID and the basic username/password login prompt • Sparx Systems recommend the use of this option only on a model that has been configured as read only, since all updates made by WebEA users in this configuration will be identified as the same user
Allow blank passwords	Defines whether or not WebEA will attempt to send blank passwords to the Pro Cloud Server for validation.

Automatic Windows Login	When set to Yes, WebEA will automatically attempt to login using Windows ID (bypassing the 'Login' screen). If the Windows login fails then the 'Login' screen will be displayed, allowing the user to manually enter credentials. Note: This option only applies if the model has the user security 'Accept Windows Authentication' option enabled. See the 'Maintain Users' help topic for details.

General Options

Item	Description
Default Package Layout	Defines the default layout when viewing the contents of a Package (or an element with children) in WebEA's main view. WebEA can display child items in one of three layouts, either Icon view (the default), List view, or Notes view (resembling the Enterprise Architect Specification Manager view).
Object Order	Defines the order of objects in the

	Browser and Object List views. There are three different ordering behaviors: • **Alphabetical -** Objects are sorted alphabetically (by name) • **Free Sort -** Objects are sorted based on their 'Set Position' (manual sorting, which is applied in Enterprise Architect), then alphabetically; this reflects Enterprise Architect's sorting when the 'Allow Free Sorting' option is enabled • **Object Type -** Objects are sorted by type, then set position, then alphabetically; this reflects Enterprise Architect's sorting when the 'Allow Free Sorting' option is disabled Note: Regardless of the 'Object Order' setting, WebEA displays diagrams, then Packages, then elements. The 'Object Order' applies to the items within each of these groupings.
Favorites as Home	When set to 'Yes', the WebEA 'Home' page will display the user's 'Favorites' Packages in the 'Browser' or 'Package' view. This is loaded when first accessing the model, and also when using WebEA's Home button. Notes:

	When set to Yes, this overrides the 'default_diagram' setting Favorites are defined using Enterprise Architect; refer to the link in the See Also column for details If a user has no Favorites defined then the Model Root or Default diagram (if one is defined) will be loaded instead
Default Diagram	Defines the GUID of the diagram that should be automatically loaded when users log onto the model. The GUID of any diagram can be obtained from WebEA using the 'Obtain direct link to the current page' icon in the Navigation bar. If this option is not set then WebEA will load either: • The user default diagram, as defined in Enterprise Architect • The model default diagram, as defined in Enterprise Architect (if no user default is set) or • The model root (if no user/model defaults are set). Note: The entire GUID (including {}) must be used.

Mini Properties Navigates	When set to 'Yes', selecting certain 'navigable' elements will cause WebEA to navigate to the target of these elements. Examples of such 'navigable' elements are Hyperlinks and Navigation Cells. For all other elements, selecting them will display the properties of the element in the Properties View. This option has no effect when the Properties View is not shown.
Diagram Elements Navigate	From WebEA version 3.01.23.1690 (in Pro Cloud Server 3.0.23) onwards, the default selection of some objects on diagrams matches Enterprise Architect's behavior. This applies to i) Composite elements and ii) Package elements that contain a diagram. • Composite elements - A Composite element on a diagram will navigate directly to its Composite diagram • Package elements - A Package element on a diagram will navigate to the first child diagram contained within the selected Package; if the selected Package does not contain any diagrams, then the Package's properties are displayed When 'Diagram Elements Navigate' is set

	to 'No', selecting either a Composite element or Package element on a diagram will display the selected element's properties.
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.
Show Browser	Defines the default visibility of the Browser. Notes: • If a user selects the 'Mobile' option when logging in to WebEA, this overrides the 'Show Browser' option and the Browser will be hidden • This behavior could be overridden depending on the screen resolution; for example, on a low resolution screen such as a mobile phone, the Browser will be hidden regardless of the 'Show Browser' option
Show Diagram	Select Yes to show diagram-only objects such as Notes, Text and Boundaries in the

Objects	Browser. These objects are grouped in an collapsible [...] node.
Show Properties View	Defines the default visibility of the Properties View. Notes: • If a user selects the 'Mobile' or 'Tablet' option when logging in to WebEA, this overrides the 'Show Properties View' option and the 'Properties View' will be hidden • This behavior could be overridden depending on the screen resolution; for example, on a low resolution screen such as a mobile phone, the 'Properties View' will be hidden regardless of the 'Show Properties View' option
Show Path Button	Select 'Yes' to show the Path button in the navbar. The Path button is used to display a list of all Packages/objects from the current object up to the model root.
Show Breadcrumb	Select Yes to show the full path to the current object in WebEA's navbar. When set to No, the navbar will display the current object name only.

Collaboration Options

Item	Description
Show Journal Entries	Defines whether Journal entries are accessible in WebEA. When set to 'Yes', Journal entries will be accessible via the Discuss & Review - History window's 'Journals' tab. Journal entries will also be accessible via the Object Properties View, if the object has any Journal entries defined.
Show Model Chat	Defines whether Model Chat is accessible in WebEA. When set to 'Yes', Model Chat will be accessible via the Discuss & Review View's 'Chat' tab. Note: As with Enterprise Architect, Model Chat is not available in models which do not have user security enabled.
Chat Notification Frequency	Defines how frequently (in seconds) WebEA will check for new Chat messages and display/update the Chat Notification which appears in the WebEA status bar. By default this value is set to 30

	(seconds).
Show Model Mail	Defines whether Model Mail is accessible in WebEA. When set to 'Yes', Model Mail will be accessible via the Collaboration View's 'Mail' tab. Note: As with Enterprise Architect, Model Mail is not available in models which do not have user security enabled.
Mail Time Period	Defines how many days worth of messages are displayed in the Model Mail Inbox and Sent Mail lists. By default this is set to 90 (days). Reducing this number can improve performance when loading the Inbox/Sent Mail.
Show Discussions	Defines whether Discussions are accessible in WebEA. When set to 'Yes', Discussions will be accessible via the Collaboration View's 'Reviews' and 'Discussions' tabs. Discussions and Reviews will also be accessible via the Main View - Object Properties, and the Properties View.
Add Discussions	Defines whether or not new discussions on Packages and elements can be added

	to the model. Note: This option is disabled when Show Discussions is set to 'No'.
Participate in Reviews	When set to 'Yes', users are able to Join discussion reviews. In order to make full use of this option the 'Show Discussions' and 'Add Discussions' options would also have to be set to 'Yes'.
Use Avatars	When set to 'Yes', all discussion and chat messages are displayed with the creating user's avatar. When set to 'No', a set of generic images representing discussions and replies is used instead.

Object Update Options

Item	Description
Can add Objects	Defines whether or not new objects (elements, diagrams, or Packages) can be added to the model.
Edit Object Notes	Defines whether an object's notes can be edited.

Can add <type>	The remaining options in this section define whether or not elements of the given object type can be added to the model. Note: If the 'Can add Objects' option is set to 'No', then this overrides the individual 'Can add <type>' options.

Object Feature Update Options

Item	Description
Can add Object Features	Defines whether or not element features (such as Changes and Tests) can be added to the model.
Can edit Tests	Defines whether or not the details of element tests can be edited in the model.
Can edit Resources	Defines whether or not the details of element resource allocations can be edited in the model.
Can add <feature type>	The remaining options in this section define whether or not specific feature types (such as Changes or Tests) can be added to elements in the model.

	Note: If the 'Can add Object Features' option is set to 'No', then this overrides the individual 'Can add <feature type>' options.

Property Section Options

The Property Section Options define whether specific properties are available and/or visible when viewing the properties of an element in WebEA. By default, all property sections are set to be visible ('Yes').

Note: When viewing the Properties of an element in WebEA's Main View, the feature buttons to display specific properties are only displayed if the element has features or properties of that type.

For example, if an element has no Tagged Values, then the 'Tagged Values' feature button will not be displayed (regardless of Property Section Options).

Default Watchlist Options

Item	Description
Days to Watch	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.
Cookie Retention	Defines the period over which a WebEA user's set of Watchlist options is stored. The value is the number of days that the cookies will be retained for.
Recent Discussions	Define whether or not the watchlist should display a count of elements that have had discussions during the activity period.
Recent Reviews	Define whether or not the watchlist should display a count of review elements that have been added or modified during the activity period.
Recent Diagrams	Define whether or not the watchlist should display a count of added/modified diagrams during the activity period.
Recent Elements	Define whether or not the watchlist should display a count of added/modified elements during the activity period.
Active resource tasks	Define whether or not the watchlist should display a count of elements with active resource allocations during the activity period.

Ending resource tasks	Define whether or not the watchlist should display a count of elements with resource allocations ending during the activity period.
Overdue resource tasks	Define whether or not the watchlist should display a count of elements with resource allocations that fall overdue during the activity period.
Recently passed tests	Define whether or not the watchlist should display a count of elements with tests that passed during the activity period.
Recently failed tests	Define whether or not the watchlist should display a count of elements with tests that failed during the activity period.
Recently deferred tests	Define whether or not the watchlist should display a count of elements with tests that were deferred during the activity period.
Recent tests not checked	Define whether or not the watchlist should display a count of elements with tests that were not checked during the activity period.

Tests not run	Define whether or not the watchlist should display a count of elements with tests that were not run during the activity period.
Verified changes	Define whether or not the watchlist should display a count of elements with verified changes during the activity period.
Recently requested changes	Define whether or not the watchlist should display a count of elements with requested changes during the activity period.
Recently completed changes	Define whether or not the watchlist should display a count of elements with changes completed during the activity period.
New changes	Define whether or not the watchlist should display a count of elements with new changes added during the activity period.
Incomplete changes	Define whether or not the watchlist should display a count of elements with incomplete changes during the activity

	period.
High Priority events	Define whether or not the watchlist should display a count of elements with events set to high during the activity period.
Verified defects	Define whether or not the watchlist should display a count of elements with verified defects during the activity period.
Recently reported defects	Define whether or not the watchlist should display a count of elements with defects reported or 'requested' during the activity period.
Recently resolved defects	Define whether or not the watchlist should display a count of elements with defects completed during the activity period.
New defects	Define whether or not the watchlist should display a count of elements with defects added during the activity period.
Incomplete defects	Define whether or not the watchlist should display a count of elements with incomplete defects during the activity period.

Verified issues	Define whether or not the watchlist should display a count of elements with verified issues during the activity period.
Recently reported issues	Define whether or not the watchlist should display a count of elements with issues reported or 'requested' during the activity period.
Recently resolved issues	Define whether or not the watchlist should display a count of elements with issues completed during the activity period.
New issues	Define whether or not the watchlist should display a count of elements with issues added during the activity period.
Incomplete issues	Define whether or not the watchlist should display a count of elements with incomplete issues during the activity period.
Verified tasks	Define whether or not the watchlist should display a count of elements with verified tasks during the activity period.
Recently	Define whether or not the watchlist

reported tasks	should display a count of elements with requested tasks during the activity period.
Recently resolved tasks	Define whether or not the watchlist should display a count of elements with tasks completed during the activity period.
New tasks	Define whether or not the watchlist should display a count of elements with tasks added during the activity period.
Incomplete tasks	Define whether or not the watchlist should display a count of elements with incomplete tasks during the activity period.
Recently reported events	Define whether or not the watchlist should display a count of elements with requested events during the activity period.
Incomplete events	Define whether or not the watchlist should display a count of elements with incomplete events.
Verified decisions	Define whether or not the watchlist should display a count of elements with verified decisions during the activity

	period.
Recently reported decisions	Define whether or not the watchlist should display a count of elements with requested decisions during the activity period.
Recently resolved decisions	Define whether or not the watchlist should display a count of elements with decisions resolved during the activity period.
New decisions	Define whether or not the watchlist should display a count of elements with decisions added during the activity period.
Incomplete decisions	Define whether or not the watchlist should display a count of elements with incomplete decisions during the activity period.

Configure WebEA models - via Text Editor

WebEA models can be configured by directly editing WebEA's \includes\webea_config.ini file. For example:

C:\xampp\htdocs\webea\includes\webea_config.ini

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"
```

Ordering: As of version 4.1.44 it is possible to change the display order of the models shown on the Login page; items within the *model_list* can be reordered as required and the Login page will display them as they are defined.

Additionally, the existence of consecutive model numbers is no longer a requirement; that is, there can be missing model

numbers. For example:

```
[model_list]
model5 = "Model One (no updates)"
model1 = "Model One with discussions"
model21 = "Model Two with user login"
model32 = "Model Three with all options"
model4 = "Sparx Systems Sample Model"
```

Sparx Systems recommends that once a model has been assigned a number it should retain this number so that any saved URLs will continue to work, which is now achievable with the changes introduced in v4.1.44.

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 cannot 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 Server'). This group of settings defines how WebEA will connect and communicate with the Pro Cloud Server for data retrieval.

All communication between WebEA and the Pro Cloud Server 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 Pro Cloud Server. **Options:** http or https **Default:** http **Example:** sscs_protocol = "http" Note: Using https requires you to have a Port configured for https access and an SSL certificate as described in the 'Add a Port Definition' topic. If you are using a self-signed certificate then

	'sscs_enforce_certs = "false"' should also be included.
sscs_port	Defines the TCP Port used to connect to the Sparx Systems Pro Cloud Server. This Port number must match one of the Ports the Sparx Systems Pro Cloud Server 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 Pro Cloud Server. **Default:** localhost **Example:** sscs_server = "sparxpublic.com" or sscs_server = "197.168.0.10"
sscs_use_ssl	*As of WebEA version 3.01.23.1690 (included in Pro Cloud Server 3.0.23) this option is no longer used. Instead WebEA determines this value dynamically based on the protocol in use:* *HTTP - sscs_use_ssl="false" and* *HTTPS - sscs_use_ssl="true"* WebEA versions prior to 3.0.23.1690 use

	this value to define if all communications between WebEA and Pro Cloud Server 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_use_ssl = "false"
sscs_db_alias	Defines the database alias name of the model repository hosted on the Pro Cloud Server. This corresponds to the 'Database Alias' identified in the Pro Cloud Server Configuration Client or the 'Model Name' on the Enterprise Architect 'Cloud Connection' screen. The value is case sensitive. **Default:** ea **Example:** sscs_db_alias = "RepositoryName"
sscs_enforce_certs	When HTTPS is the protocol in use, this option defines whether or not cURL will enforce that the certificate installed in Pro Cloud Server is correctly signed by a Certificate Authority. Note: If you are using a self-signed certificate for Pro Cloud Server, this

	option must be set to "false". **Options:** true or false **Default:** true **Example:** sscs_enforce_certs = "false"
sscs_access_code	When configuring a Pro Cloud Server model connection / database manager an optional 'OSLC Access Code' can be defined. If the model connection has an 'OSLC Access Code' assigned then it will need to be included in the WebEA configuration using the 'sscs_access_code' setting. **Example:** sscs_access_code = "myoslcaccesscode"
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 log in to a security enabled model. **Note:** • *Starting in Pro Cloud Server version 3.0.23* - When set, all other authentication schemes are bypassed, including Windows Authentication, OpenID and basic username/password

	login • Sparx Systems recommend the use of this option only on a model that has been configured as read-only, since all updates made by WebEA users in this configuration will be identified as the same user. **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 login to a security enabled model. **Note:** • *Starting in Pro Cloud Server version 3.0.23* - When set, all other authentication schemes are bypassed, including Windows Authentication, OpenID and basic username/password login • Sparx Systems recommend the use of this option only on a model that has been configured as read-only, since all updates made by WebEA users in this configuration will be identified as the

	same user **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 is only used for basic username/password authentication and is not used for Windows Authentication or OpenID. It is possible to set this option and also set 'sscs_model_user'. In this case the user will be required to enter the access code only and will not be prompted for a username or password. **Options:** text **Default:** blank

	Example: auth_code = "JWG478"
login_prompt	*This option is no longer used as of WebEA version 3.0.23.1690 (included in Pro Cloud Server 3.0.23). Basic authentication username and password fields will show for models that allow basic authentication (that is, are not restricted to SSO logins).* The basic authentication (username/password) will show for security enabled models that have not set *sscs_model_user*. Windows Authentication, if enabled for the model, will be attempted automatically when selecting the model. OpenID authentication, if enabled in the model, will show above any basic authentication prompt. For older versions of WebEA, the option is used to define whether or not all WebEA users must enter a model user/password before they can access the model. Note: When this option is set to "false" and the model has security enabled, the 'sscs_model_user' and 'sscs_model_pwd' options must also be used to specify the

	logon credentials. **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 Pro Cloud Server for validation. **Options:** true or false **Default:** false **Example:** login_allow_blank_pwd = "false"
auto_login_windows_auth	When set to "true", WebEA will automatically attempt to login using Windows authentication (bypassing the 'Login' screen). If the Windows login fails then the 'Login' screen will be displayed, allowing the user to manually enter credentials. **Options:** true or false **Default:** false **Example:** auto_login_windows_auth = "true" Note: This option only applies if the model has the user security 'Accept

	Windows Authentication' option enabled. See the 'Maintain Users' help topic for details.
miniprops_navigates	When the 'miniprops_navigates' option is set to "true", selecting certain 'navigable' elements will cause WebEA to navigate to the target of these elements. Examples of such 'navigable' elements are Hyperlinks and Navigation Cells. For all other elements, selecting them will display the properties of the element in the Properties View. This option has no effect when the Properties View is not shown. **Options:** true or false **Default:** true **Example:** miniprops_navigates = "false"
navigate_to_diagram	As of WebEA version 3.01.23.1690 (included in Pro Cloud Server 3.0.23) the default selection of some objects on diagrams matches Enterprise Architect behavior. This applies to i) Composite elements and ii) Package elements that contain a diagram. • Composite Elements - A composite element on a diagram will navigate

	directly to its composite diagram. • Package Elements - A Package element on a diagram will navigate to the first child diagram contained within the selected Package; if the selected Package does not contain any diagrams then the Package's properties are displayed When 'navigate_to_diagram' is set to "false", selecting either a composite element or Package element on a diagram will display the selected element's properties. **Options:** true or false **Default:** true **Example:** navigate_to_diagram = "false"
show_search	Defines whether the 'Search' menu is visible and available. **Options:** true or false **Default:** true **Example:** show_search = "false"
show_watchlist	Defines whether the 'Watchlist' menu item is visible and available. **Options:** true or false **Default:** true

	Example: show_watchlist = "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"
prop_sec_<section>_visible	Defines whether or not a property section (or feature) is accessible. If a feature is not visible then the button to display that feature will not be available in the 'Main View', and the Feature will not be listed in the 'Select Feature' menu in the Properties View. <section> can be one of: • location • instances • relationships • taggedvalues • testing • resourcealloc

	• attributes • operations • requirements • constraints • files • runstates • features • changes • documents • defects • issues • tasks • events • decisions • efforts • risks • metrics **Options:** true or false **Default:** true **Example:** prop_sec_testing_visible = "true"
prop_sec_<section>_expanded	*This option is no longer used as of WebEA version 3.0.23.1690 (included in Pro Cloud Server 3.0.23). Property sections are no longer displayed as*

	expandable/collapsible panels. Defines whether or not a property section is expanded by default when the model is 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 • instances • 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. The GUID of any diagram can be obtained from WebEA by loading the diagram properties. If this option is not set then WebEA will load either: • The user default diagram, as defined in Enterprise Architect • The model default diagram, as defined in Enterprise Architect (if no user default is set), or • The model root (if no user/model defaults are set) 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 three layouts. Either "icon" view (the default), "notes" view (resembling the Enterprise Architect Specification Manager view) or "list" view. **Options:** icon, notes or list **Default:** icon **Example:** default_main_layout = "icon"
object_order	Defines the order of objects in the 'Browser' and 'Object List' views. There are three different ordering behaviors, represented by values 1,2, and 3. • "1" = Alphabetical - Objects are sorted alphabetically (by name) • "2" = Free Sort - Objects are sorted based on their 'Set Position' (manual sorting which is applied in Enterprise Architect), then alphabetically; this reflects Enterprise Architect's sorting when the 'Allow Free Sorting' option is enabled • "3" = Object Type - Objects are sorted by type, then set position, then alphabetically; this reflects Enterprise Architect's sorting when the 'Allow

	Free Sorting' option is disabled **Options:** 1, 2 or 3 **Default:** 3 **Example:** object_order = "2" Note: Regardless of the 'object_order' setting, WebEA displays diagrams, then Packages, then elements; the 'object_order' applies to the items within each of these groupings
favorites_as_home	When set to "true", the WebEA 'Home' page will display the user's 'Favorites' Packages in the 'Browser' or 'Package' view. This is loaded when first accessing the model, and also when using WebEA's Home button. The Browser's Up button is also hidden, helping to keep navigation within the Favorites Package(s). **Options:** true or false **Default:** true **Example:** favorites_as_home = "false" Notes: • When set to true, this overrides the 'default_diagram' setting • Favorites are defined using Enterprise Architect; refer to the link in the *See Also* column for details

	• If a user has no Favorites defined then the Model Root or Default diagram (if one is defined) will be loaded instead
show_browser	Defines the default visibility of the 'Browser'. **Options:** true or false **Default:** true **Example:** show_browser = "false" Notes: • If a user selects the 'Mobile' option when logging in to WebEA then this overrides the 'show_browser' option and the 'Browser' will be hidden • This behavior can be overridden, depending on the screen resolution; for example, on a low resolution screen such as a mobile phone, the 'Browser' will be hidden regardless of the 'show_browser' option setting
show_diagram_objs	Defines the visibility of diagram-only objects such as Notes, Text and Boundaries in the Browser. Wehen displayed these objects are grouped in an collapsible [...] node. Options: true or false Default: false

	Example: show_diagram_objs = "true"
show_propertiesview	Defines the default visibility of the 'Properties View'. **Options:** true or false **Default:** true **Example:** show_propertiesview = "false" Notes: • If a user selects the 'Mobile' or 'Tablet' option when logging in to WebEA, this overrides the 'show_propertiesview' option and the Properties View will be hidden • This behavior could be overridden depending on the screen resolution; for example, on a low resolution screen such as a mobile phone, the Properties View will be hidden regardless of the 'show_propertiesview' option setting
show_path_button	Defines the visibility of the Object Path button. When set to true, the button is available in WebEA's Navigation Bar. The Object Path button is used to display a list of Packages and Objects from the current Object up to the Model Root. **Options:** true or false

	Default: true **Example:** show_path_button = "false" Setting this option to "false" can help in keeping WebEA focused on a specific area of your model, especially when used in conjunction with 'favorites_as_home = "true"'.
show_breadcrumb	Defines the display of the full path (or 'breadcrumb') to the current object in WebEA's navbar. When set to false, the navbar will display the current object name only. Options: true or false Default: true Example: show_diagram_objs = "true"

Model collaboration settings

show_comments	Defines whether comments are accessible in WebEA. When set to 'true', comments will be accessible via the Discuss & Review window's 'Journal' tab. Comments will also be accessible via the

	Object Properties View, if the object has any comments defined. **Options:** true or false **Default:** false **Example:** show_comments = "true"
show_chat	Defines whether Model Chat is accessible in WebEA. When set to 'true', Model Chat will be accessible via the Chat & Mail View's 'Chat' tab. **Options:** true or false **Default:** false **Example:** show_chat = "true" Note: As with Enterprise Architect, Model Chat is not available in models which do not have user security enabled.
chat_notify_sec	Defines how frequently (in seconds) WebEA will check for new Chat messages and display/update the Chat Notification which appears in the WebEA status bar. **Default:** 30 **Example:** chat_notify_sec = "30"

show_mail	Defines whether Model Mail is accessible in WebEA. When set to 'true', Model Mail will be accessible via the Collaboration View's 'Mail' tab. **Options:** true or false **Default:** false **Example:** show_mail = "true" Note: As with Enterprise Architect, Model Mail is not available in models which do not have user security enabled.
mail_days	Defines how many days worth of messages are displayed in the Model Mail Inbox and Sent Mail lists. **Default:** 90 **Example:** mail_days = "90" Note: Reducing this number can improve performance when loading the Inbox/Sent Mail.
show_discuss	Defines whether Discussions are accessible in WebEA. When set to 'true', Discussions will be accessible via the

	Collaboration View's 'Reviews' and 'Discussions' tabs. Discussions and Reviews will also be accessible via the Main View - Object Properties, and the Properties View. **Options:** true or false **Default:** false **Example:** show_discuss = "true"
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"
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"
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"

Model update settings

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

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

Option	Description
add_objects	Defines whether or not new objects can be added to the model.

	Options: true or false **Default:** false **Example:** add_objects = "true"
edit_object_notes	Defines whether or not the notes of elements can be edited. **Options:** true or false **Default:** false **Example:** edit_object_notes = "true"
edit_object_tags	Defines whether or not the tagged values of elements can be edited. **Options:** true or false **Default:** false **Example:** edit_object_tags = "true"
add_diagrams	Defines whether or not new diagrams can be added to the model. **Options:** true or false **Default:** false **Example:** add_diagrams = "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". The value of <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_objectfe	Defines whether or not element features

ature_<type>	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". The value of <type> can be one of: • tests • resources • features • changes • documents • 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 watchlist 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 watchlist 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 watchlist 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 watchlist should display a count of the diagrams that have been added/modified during the activity period. **Options:** true or false **Default:** false
wl_recent_element	Define whether or not the watchlist should display a count of the elements that have been added/modified during the activity period. **Options:** true or false **Default:** false
wl_resalloc_active	Define whether or not the watchlist 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 watchlist 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 watchlist 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 watchlist 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 watchlist 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 watchlist should display a count of elements with tests that were deferred during the activity period. **Options:** true or false **Default:** false
wl_test_recen	Define whether or not the watchlist

tnotchk	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 watchlist should display a count of elements with tests that were not run during the activity period. **Options:** true or false **Default:** false
wl_feature_verified	Define whether or not the watchlist should display a count of elements with verified features during the activity period. **Options:** true or false **Default:** false
wl_feature_requested	Define whether or not the watchlist should display a count of elements with requested features during the activity period. **Options:** true or false **Default:** false
wl_feature_c	Define whether or not the watchlist

ompleted	should display a count of elements with features completed during the activity period. **Options:** true or false **Default:** false
wl_feature_new	Define whether or not the watchlist should display a count of elements with new features added during the activity period. **Options:** true or false **Default:** false
wl_feature_incomplete	Define whether or not the watchlist should display a count of elements with incomplete features during the activity period. **Options:** true or false **Default:** false
wl_change_verified	Define whether or not the watchlist should display a count of elements with verified changes during the activity period. **Options:** true or false **Default:** false
wl_change_r	Define whether or not the watchlist

equested	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 watchlist 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 watchlist 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 watchlist should display a count of elements with incomplete changes during the activity period. **Options:** true or false **Default:** false
wl_document	Define whether or not the watchlist

_verified	should display a count of elements with verified documents during the activity period. **Options:** true or false **Default:** false
wl_document_requested	Define whether or not the watchlist should display a count of elements with requested documents during the activity period. **Options:** true or false **Default:** false
wl_document_completed	Define whether or not the watchlist should display a count of elements with documents completed during the activity period. **Options:** true or false **Default:** false
wl_document_new	Define whether or not the watchlist should display a count of elements with new documents added during the activity period. **Options:** true or false **Default:** false
wl_document	Define whether or not the watchlist

_incomplete	should display a count of elements with incomplete documents during the activity period. **Options:** true or false **Default:** false
wl_defect_verified	Define whether or not the watchlist 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 watchlist 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 watchlist 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 watchlist 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 watchlist 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 watchlist 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 watchlist 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 watchlist 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 watchlist 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 watchlist 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 Watchlist 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 watchlist 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 watchlist 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 watchlist 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 watchlist 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 watchlist should display a count of elements with requested events during the activity period. **Options:** true or false **Default:** false
wl_event_hig	Define whether or not the watchlist

h	should display a count of elements with events set to high during the activity period. **Options:** true or false **Default:** false
wl_event_incomplete	Define whether or not the watchlist should display a count of elements with incomplete events. **Options:** true or false **Default:** false
wl_decision_verified	Define whether or not the watchlist 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 watchlist 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 watchlist 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 watchlist 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 watchlist 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 our example, when a single element has been modified recently the message displayed on the watchlist page is:

1 recently modified element

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

5 recently modified elements

Configuring WebEA to Trust Your Own Certificate Authority (CA)

An SSL certificate can be signed with your own Certificate Authority (CA). This allows trust to be added to clients to avoid security warnings. The instructions for adding a CA to a client vary by operating system or browser used.

See Using Your Own Certificate Authority (CA) for instructions on generating a CA and using it to sign server certificates.

WebEA uses PHP and cURL to communicate with a Pro Cloud Server. To enable PHP/curl to trust a new CA:

- Add/edit 'curl.cainfo' in php.ini to point to either a single root CA certificate or a CA bundle file (this is multiple concatenated crt files)

  ```
  [curl]
  ; A default value for the CURLOPT_CAINFO option.
  This is required to be an
  ; absolute path.
  curl.cainfo="C:\xampp\php\MyCompanyCA.crt"
  ```

- Restart Apache/PHP

Notes

- This does not configure the website itself for HTTPS access. For details on doing this, please refer to the documentation for your web server.

Single Sign-On (SSO) with WebEA

As with Enterprise Architect, WebEA can make use of these Single Sign-On systems:

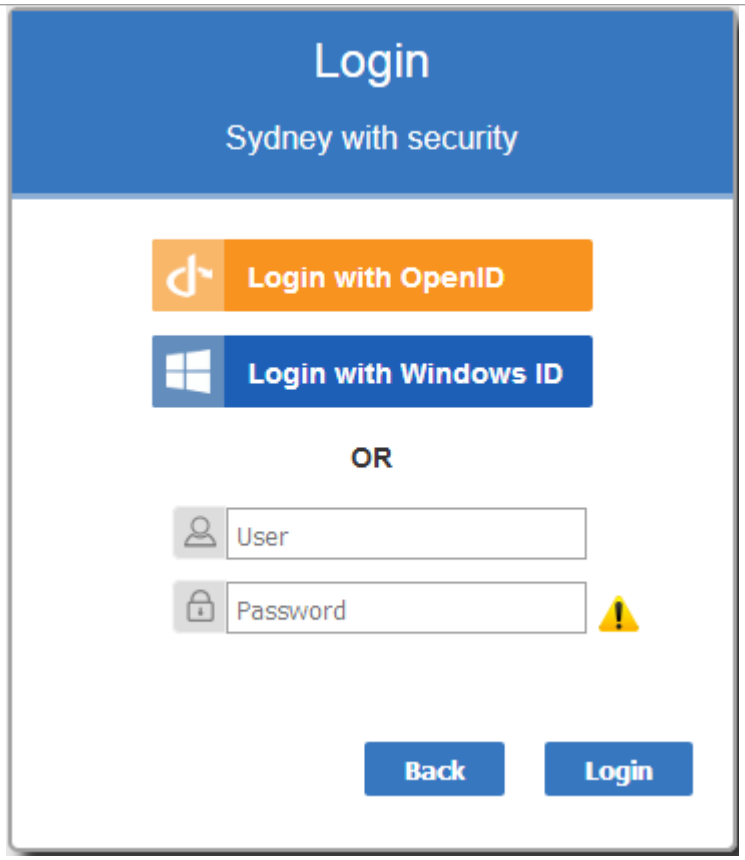
- Windows authentication with Active Directory
- OpenID

The ability to use Single Sign-On for WebEA is based on the configuration of the Enterprise Architect model itself.

Steps

Step	Description
1	Using Enterprise Architect, configure the model to use Single Sign-On.
2	Using Enterprise Architect, confirm that the model can be accessed via Single Sign-On.
3	If you intend to use OpenID, ensure that your OpenID server configuration includes the WebEA login_sso.php page as a valid redirect/callback URI. For example: *http://myserver/webea/login_sso.php* Or (if the OpenID provider supports wildcards): *http://myserver/webea/**

4	If you intend to use Windows Authentication and WebEA is deployed using IIS then configure IIS as described in the 'Configuring IIS for WebEA Windows Authentication' topic.
5	If you have not already done so, configure WebEA to access the model. No WebEA configuration options are required specifically for Single Sign-On (WebEA will detect that Single Sign-On is enabled in the model).
6	Login to the model via WebEA. The WebEA login screen will provide Login with OpenID and/or Login with Windows ID buttons, allowing you to log in via Single Sign-On.



Web Browser Considerations

Single Sign-On support can vary depending on the Web Browser being used to access WebEA.

WebEA's Windows Authentication makes use of the NTML authentication protocol. Web Browsers typically require some configuration to make use of NTLM authentication. For example, see:

https://www.websense.com/content/support/library/web/hosted/iseries_deploy/ntlm_firefox.aspx

As with Enterprise Architect, for OpenID support an OpenID server must be configured and accessible by your

Web Browser.

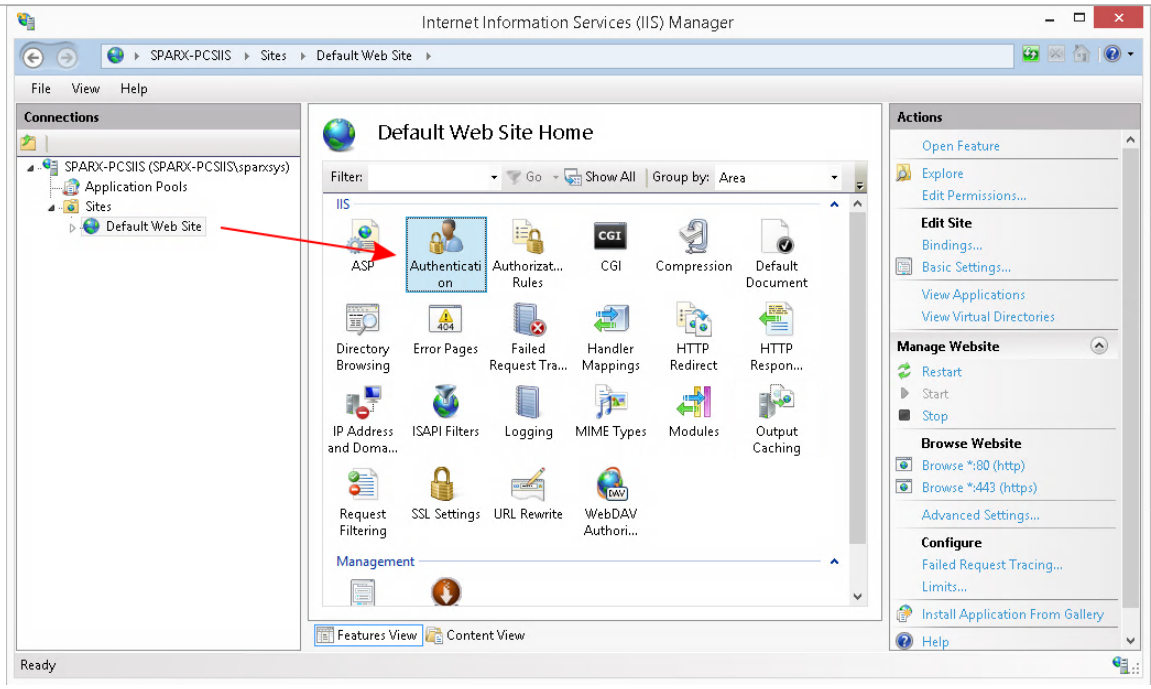
The OpenID server must support the 'OpenID Connect' standard and should be able to return a unique user identifier in the 'user_info' request. This user identifier will be matched to a local model user.

Configuring IIS for WebEA Windows Authentication

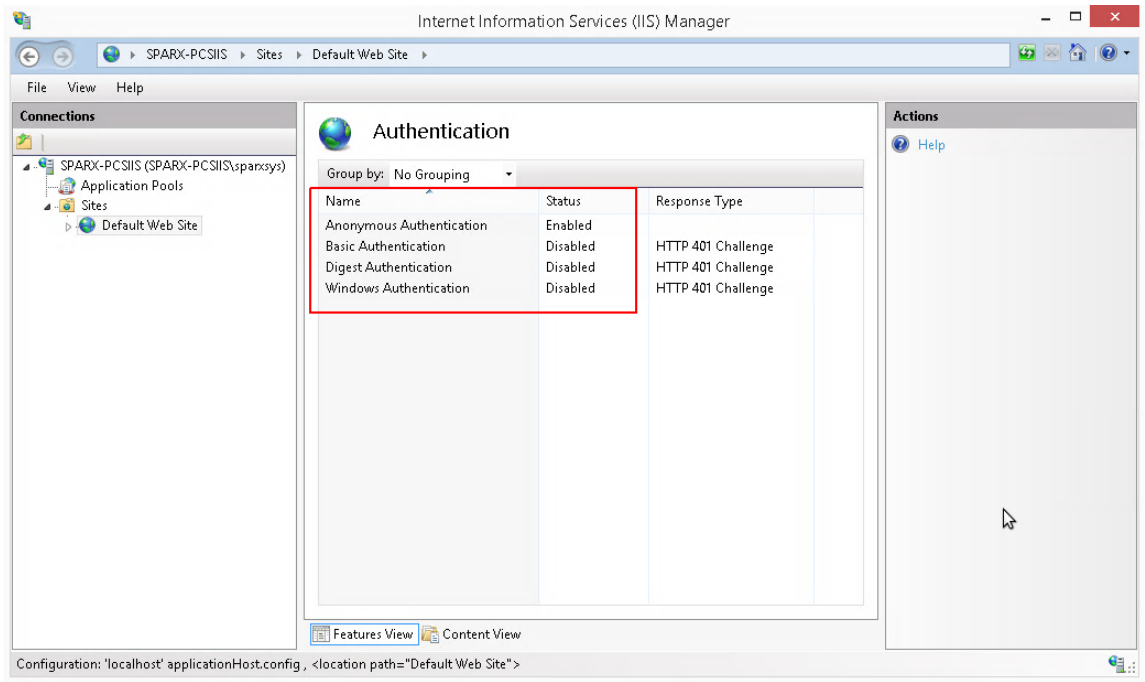
IIS Windows Authentication can cause a conflict with WebEA's Windows Authentication option. If WebEA is deployed using IIS and you wish to use WebEA's 'Login with Windows ID' then the WebEA site itself must have Windows Authentication disabled as described below.

Steps

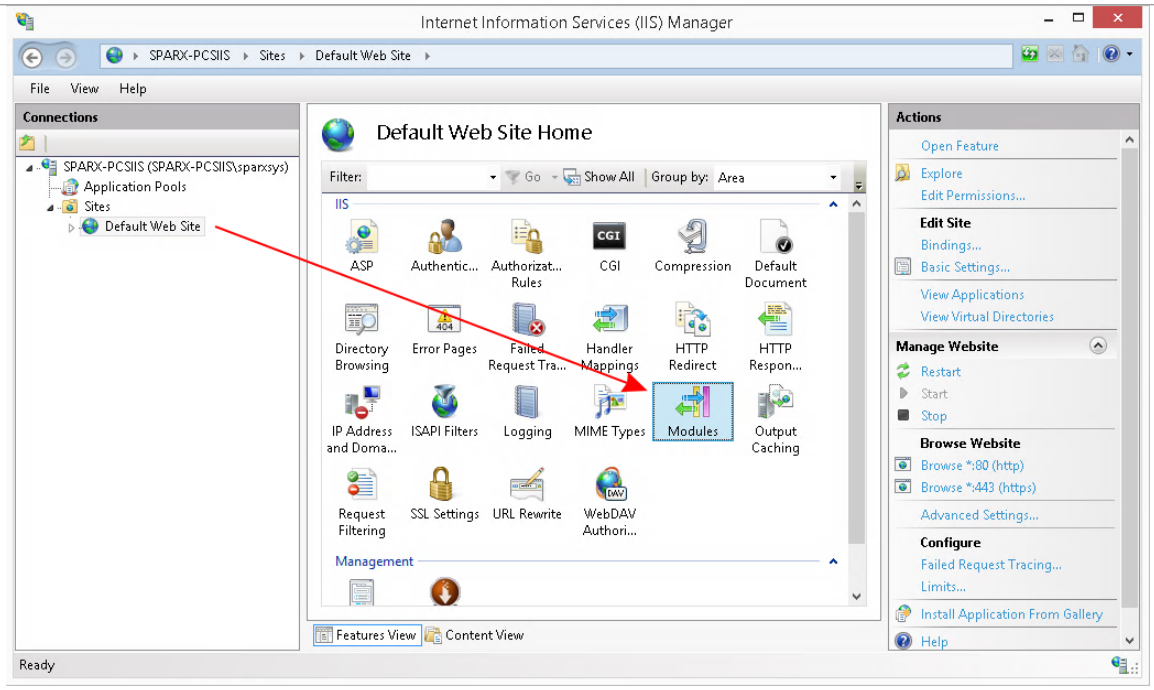
Step	Action
1	Open the Windows 'Internet Information Services (IIS) Manager' application.
2	Select your WebEA site from the 'Sites' list, then open the 'Authentication' configuration.



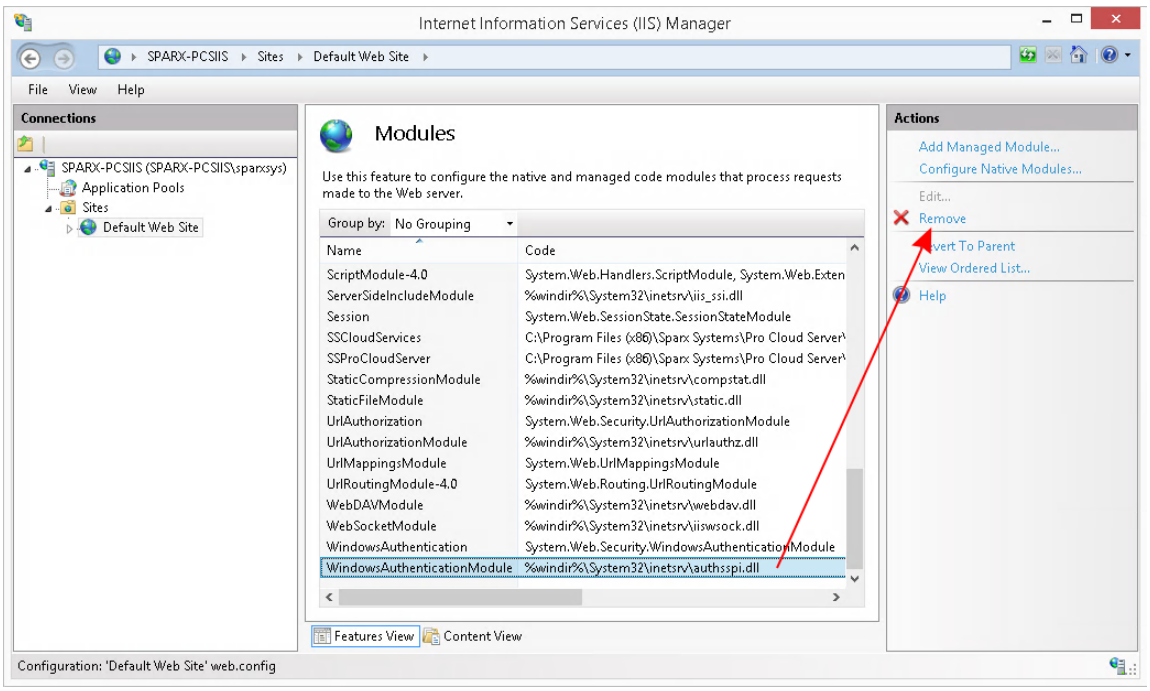
3 Enable Anonymous Authentication and disable Windows Authentication.



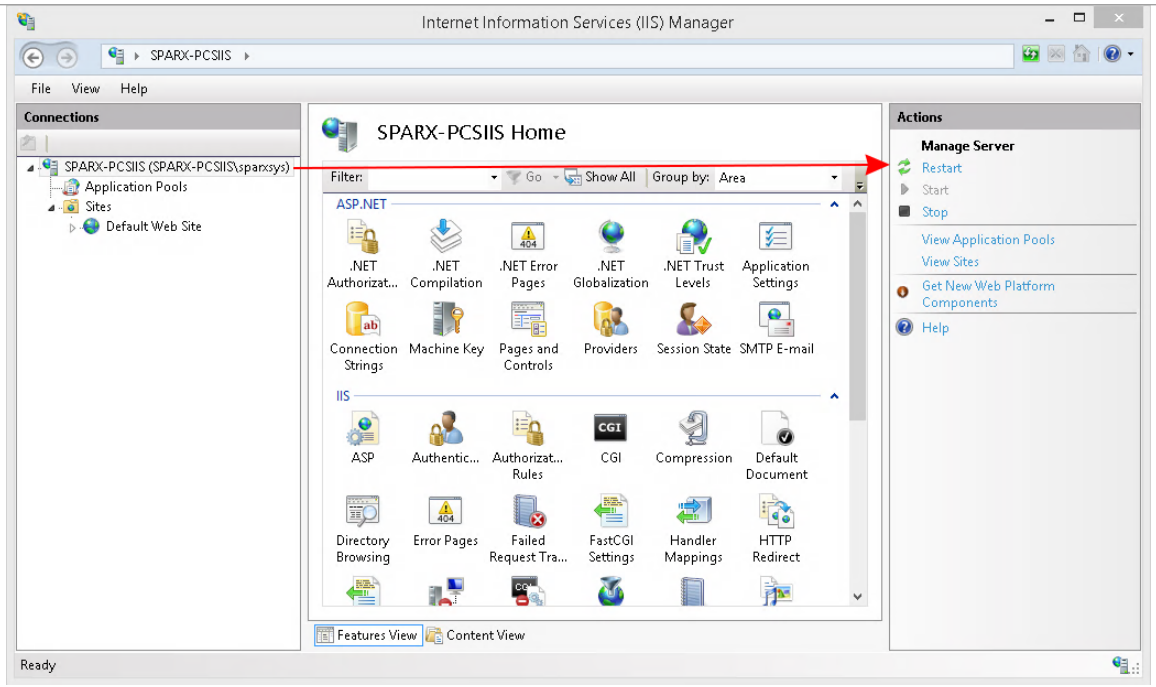
4 Again, select your WebEA site from the 'Sites' list, then open the 'Modules' configuration.



5 Select the 'WindowsAuthenticationModule', then click the 'Remove' action.



6 Now select the IIS server/home, then click the 'Restart' action.

	
7	IIS is now configured to allow use of WebEA's 'Login with Windows ID' option. See the Single Sign-On (SSO) with WebEA topic for details.

Updating WebEA to a New Version

WebEA is typically updated with each new release of the Pro Cloud Server, and is designed to work with that version of the Pro Cloud Server specifically. If you update your Pro Cloud Server to a new release you will also need to update the WebEA files on your web server as described below.

Note, if you have updated the Pro Cloud Server but have not updated WebEA then you will receive a warning after the WebEA login ('Warning - WebEA / OSLC Version mismatch').

Option 1 - Updating WebEA manually

Step	Details
1	If you have not already, install the desired version of the Pro Cloud Server (typically the latest release).
2	On your web server (often the same machine as the Pro Cloud Server), make a backup of the folder for your existing WebEA site. If you use 'WebConfig' then backup its folder also. E.g. C:\xampp\htdocs\webea C:\xampp\htdocs\webconfig
3	Browse to the location where you installed the Pro

	Cloud Server and open the WebEA subfolder. For example: C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\WebEA Copy the content of the WebEA folder, **EXCEPT** for the .\WebEA\includes\webea_config.ini file. You might prefer to rename the webea_config.ini file before copying the files (for example,. webea_config_defaults.ini), in which case it can be copied.
4	Paste the WebEA files into the folder for your WebEA site (e.g. C:\xampp\htdocs\webea), overwriting the existing files.
5	If you make use of WebConfig (otherwise skip to step 7), browse to the location where you installed the Pro Cloud Server and open the WebConfig subfolder. For example: C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\WebConfig Copy the content of the WebConfig folder, **EXCEPT** for the .\WebConfig\settings.php file. You might prefer to rename the settings.php file before copying the files (for example, settings_default.php), in which case it can be copied.
6	Paste the WebConfig files into the folder for your WebConfig site (e.g. C:\xampp\htdocs\webconfig),

	overwriting the existing files.
7	Your WebEA (and optionally WebConfig) site has now been updated to match the Pro Cloud Server version.

Option 2 - Update WebEA via command line

As an alternative to manually copying the WebEA/WebConfig files, this could be done via the command line. For example, consider the situation in which the Pro Cloud Server was installed to the default location and your website is being served by Apache in C:\Xampp\htdocs\webea. These commands will perform the required copy and not override your existing configuration file in the live site.

For Pro Cloud Server 32 bit:

robocopy "C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\WebEA" "C:\xampp\htdocs\webea" /MIR /e /DCOPY:T /njh /ndl /nfl /xf webea_config*.ini

robocopy "C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\WebConfig" "C:\xampp\htdocs\webconfig" /MIR /e /DCOPY:T /njh /ndl /nfl /xf settings.php

For Pro Cloud Server 64 bit:

robocopy "C:\Program Files\Sparx Systems\Pro Cloud Server\WebEA" "C:\xampp\htdocs\webea" /MIR /e /DCOPY:T /njh /ndl /nfl /xf webea_config*.ini

robocopy "C:\Program Files\Sparx Systems\Pro Cloud Server\WebConfig" "C:\xampp\htdocs\webconfig" /MIR /e /DCOPY:T /njh /ndl /nfl /xf settings.php

WebEA User Guide

Sparx System's WebEA is a web-based application that allowsyou to review and comment on Enterprise Architect repositories, in real time, on a range of mobile devices or on a remote work station, without needing to install Enterprise Architect. You can be anywhere in the world and do not need to be in an office, just somewhere where internet access is available(#). WebEA 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 you to update element notes and Tagged Values, 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.

The data shown by WebEA is the same as is shown by Enterprise Architect, even though you are using your web browser. 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.

Specific configuration is required to enable WebEA access to your Enterprise Architect repositories via the internet.

How to Open WebEA

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.

https://demo.sparxpublic.com

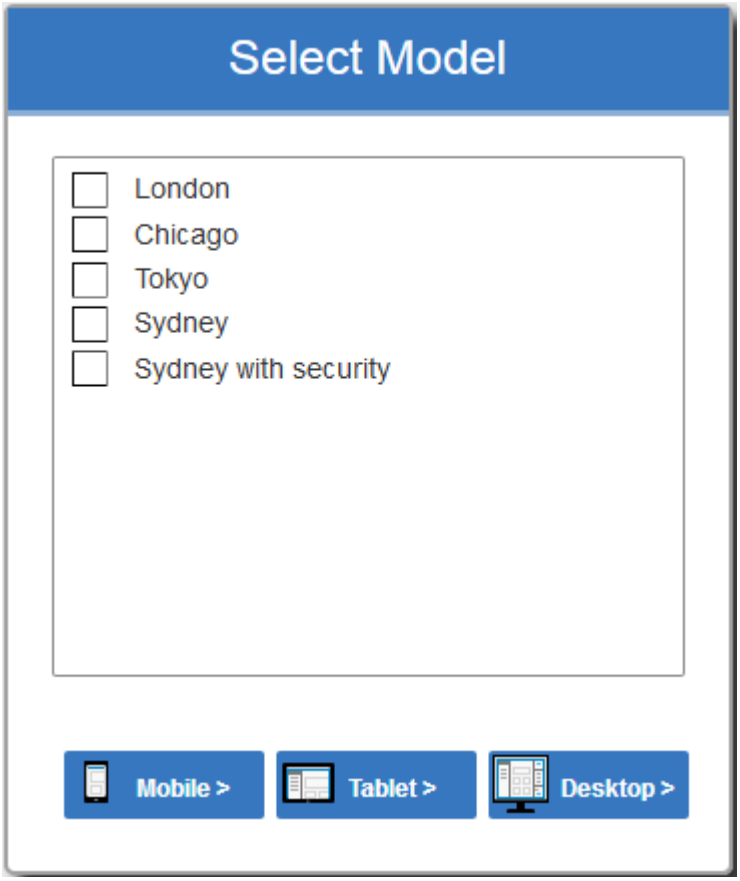
In the case where the web server has the IP of 11.22.33.44 and has been configured to use the non-standard Port of 800, the complete URL would be:

https://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 displays, 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
- HTML Purifier v4.10.0, licensed under GPL v2.1

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 and 2) standard Enterprise Architect model security. These two validation methods can be used independently or in combination in each model.

Prerequisites

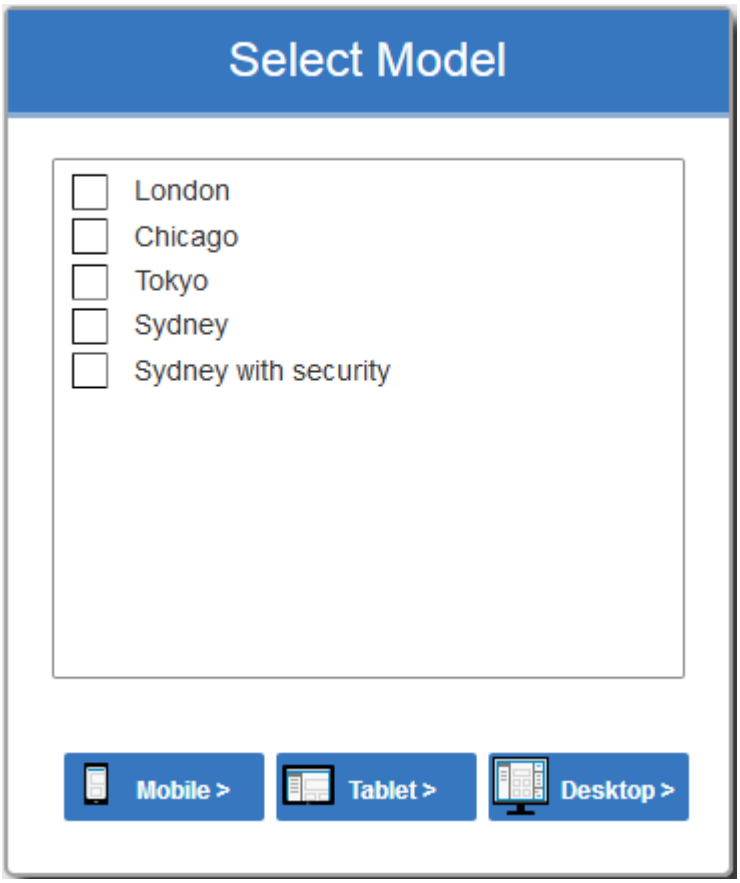
Users of WebEA will need to be advised of their login credentials for each model, which will include:

- 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.

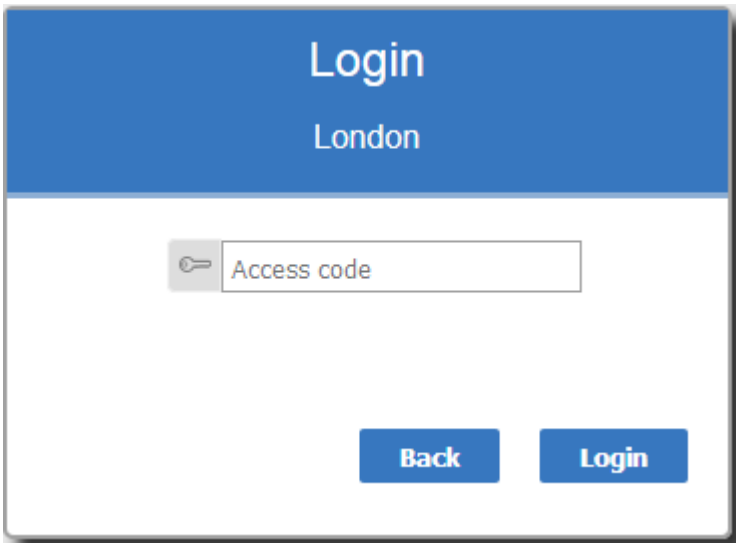


Select the model you want to open, then click or tap on either the Mobile button, Tablet button or Desktop button. The button selected determines whether the Browser View and Properties View will be enabled by default.

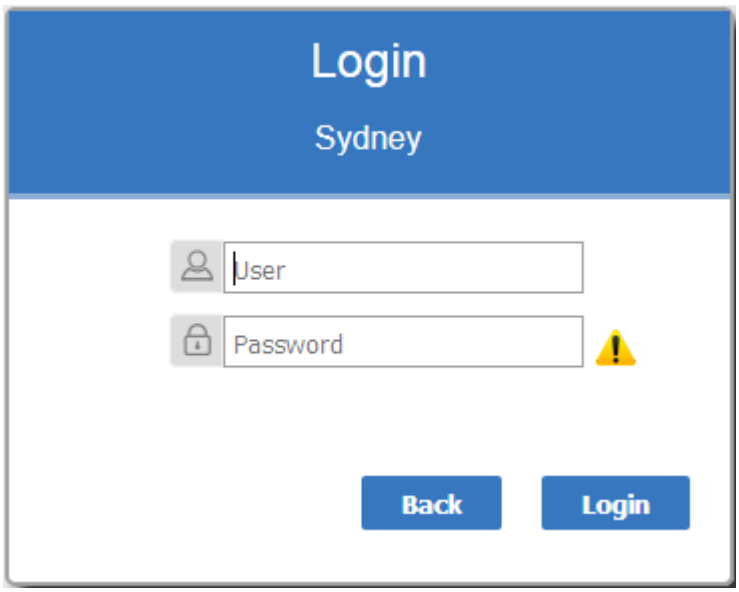
If access to your model requires only an access code, then the 'Login' page prompts you for this.

Note:

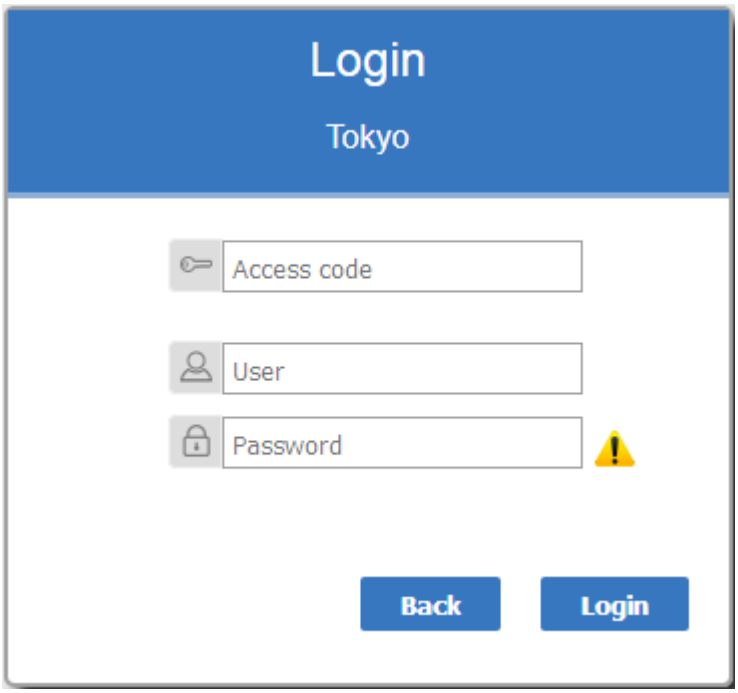
- The red asterisk (*) denotes a mandatory field
- Access Codes are not required for Windows Authentication or OpenID authentication



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



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



Single Sign On

This feature is available from Enterprise Architect Release 14.1 and Pro Cloud Server Release 3.0 build 23.

If the Enterprise Architect model has been configured for Windows and/or OpenID authentication then the 'Login' page will show the relevant single sign-on Login with OpenID button or Login with Windows ID button. Clicking the Login with OpenID button will redirect the user to the 'OpenID Server Authentication' page. After authenticating with OpenID the browser will automatically return to WebEA and log the user in.

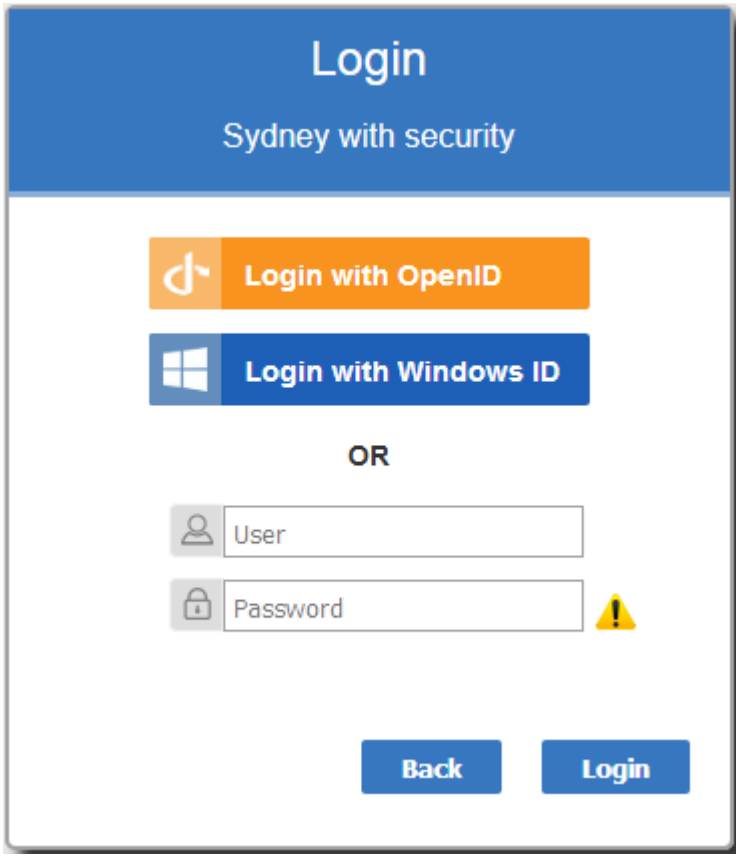
If the user is already authenticated with OpenID, the OpenID website might not be displayed, and the user will simply be logged in.

If the model is configured to also allow manual login then both the single sign on buttons and manual login fields will

be visible on the 'Login' page. If the option 'Restrict access to Windows & OpenID users only' is checked then the manual login fields will not be available.

See:

- Single Sign-On (SSO) Options



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

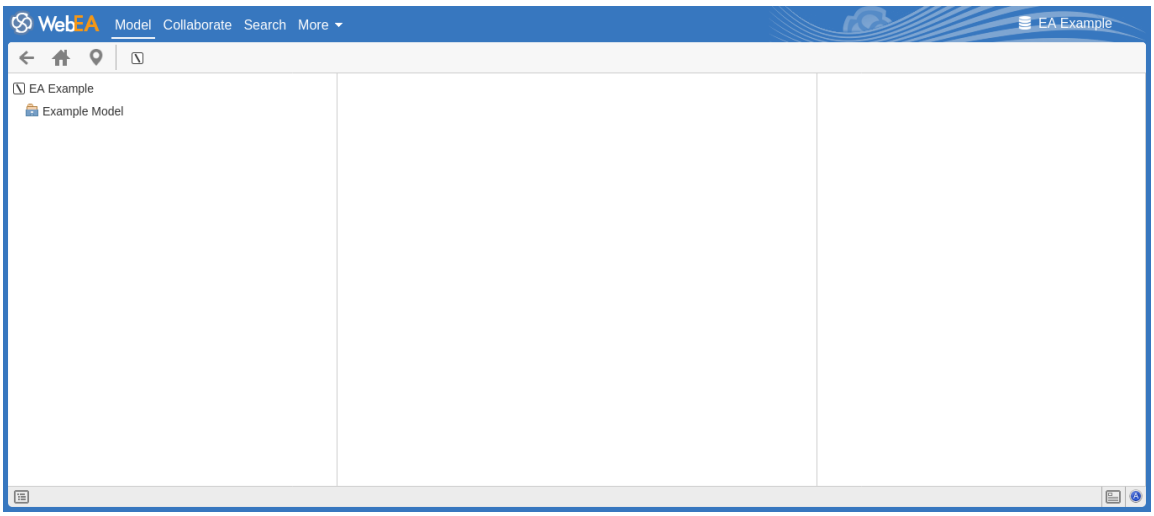
Having logged on to a model in WebEA, depending on how the model has been configured and the resolution of the Browser/Device, WebEA will typically display the model root node/s in either the Browser (on the left), or in the main content area.

Alternatively, if the WebEA model connection has been configured with a default diagram or the model is configured with User and Model Default Diagrams, then the diagram will be displayed.

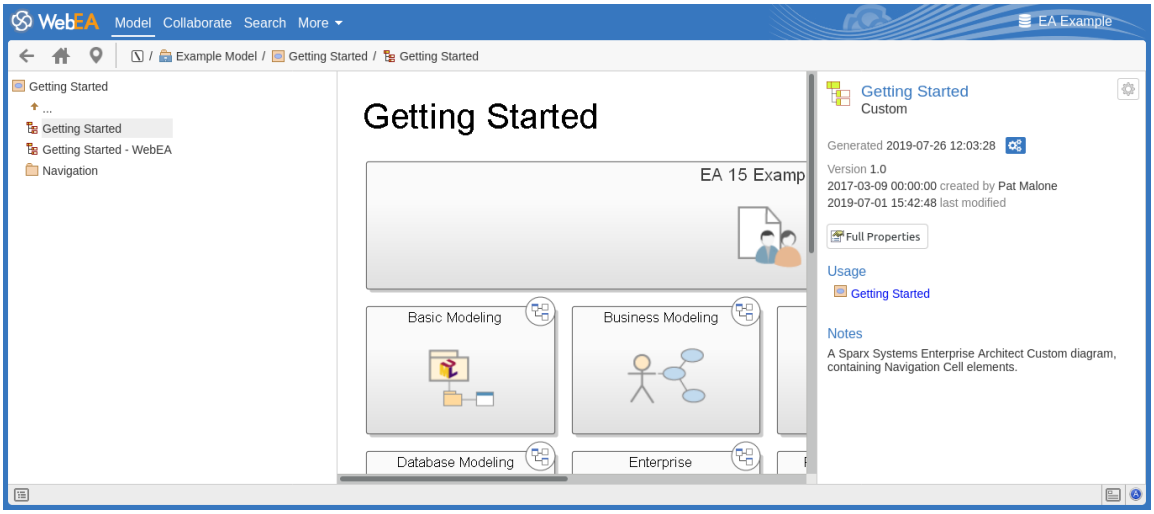
From this point you can start navigating the model by selecting objects or links, review the organization and contents of the model elements, and read and add to discussions or reviews of particular structures.

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

This image shows the WebEA screen after loading the example model, when no default diagram has been configured.



The next image shows the WebEA screen after loading the example model, when a default diagram has been configured.



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.

Main Tabs / Options



WebEA's heading area displays the application logo in the top left corner, followed by tabs for each of WebEA's main views, 'Model', 'Collaboration' (if enabled), 'Search', and a 'More' dropdown menu. By default, the Model tab is selected.

For more information on each of these tabs/options refer to the links in the table below.

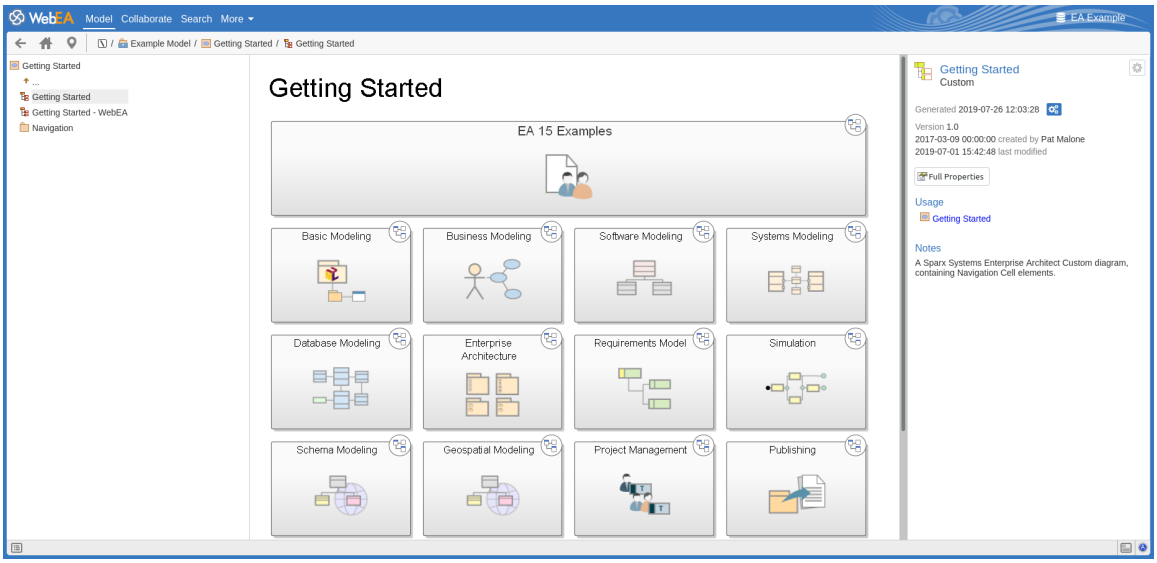
Tab / Option	Description

Model	The Model tab is used for browsing and viewing your model content.
Collaborate	The Collaborate tab provides access to collaboration features, such as Discussions, Journals, Chat and Model Mail.
Search	The Search tab provides access to WebEA's search and watchlist features.
More	The More button provides access to WebEA's main menu, This menu provides access to the relationship matrix view, as well as providing options for sharing links, accessing help, and logging out of WebEA.

WebEA - Model View

The Model View is used for browsing and viewing the content of your model. This view consists of five main components:

- The **Navigation Bar** at the top.
- The **Browser** on the left, used to assist in navigating the model.
- The **Main View** in the center, used for displaying diagrams and element properties.
- The **Properties View** on the right, which allows you to view properties of an element while simultaneously viewing something else (such as a diagram) in the Main View
- The **Status Bar** at the bottom.



Navigation Bar



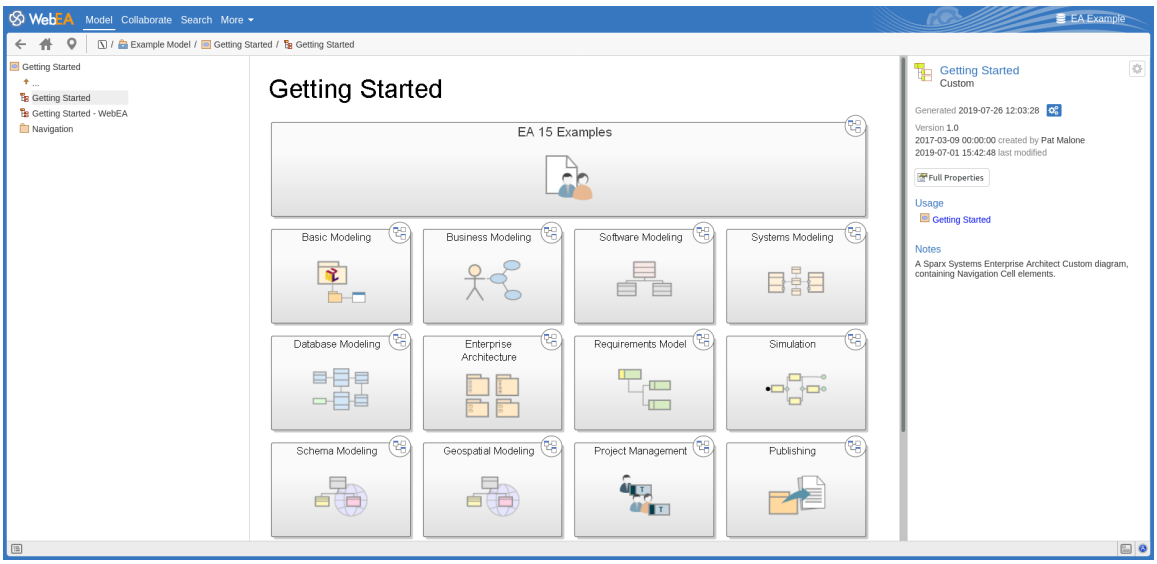
The Navigation bar provides information and buttons that assist in navigating an Enterprise Architect model.

Icon	Description
Home 	Click on this button to return focus to the 'home' object for the current model. The home object is either the WebEA default diagram (defined in the configuration file), the User/Model default diagram as set in Enterprise Architect or the model root Package.
Object Path 	Click on 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 • Optionally, this button can be hidden using the 'Show Path Button' configuration option
Back	

	Click on this button to navigate to the previous object.
Object Name / Breadcrumb	Displays the path (or 'breadcrumb') to the current object, and the object's name. You can click on an item in the path to navigate to that Package or element. Alternatively you can configure WebEA to only display the object name, by setting the 'Show Breadcrumb' option to 'No'.
Context Menu 	This button is only accessible when the screen is less than 800 pixels in width. This button provides a context menu relating to the current object. For Diagrams this menu allows you to switch between the diagram image or the diagram's properties. It also provides the option to adjust the diagram zoom level for the diagram image. Note, adjusting the zoom level will apply to all diagrams being viewed in WebEA during the current session. This allows you to find a comfortable zoom level when using WebEA on a mobile device. Adjusting the zoom level will not impact how other users see the diagram.

	For Packages and Element this menu allows you to switch between the Package/Element properties, or the list of child Objects.

Browser



The browser assists with navigation, and provides context for the object that is currently displayed in the Main View. See the WebEA - Browser topic for more information.

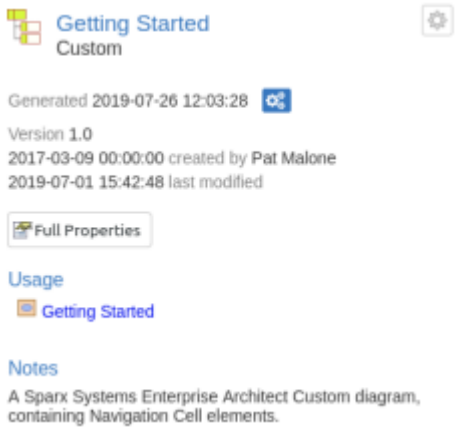
Main View



The main (center) view typically displays these types of content:

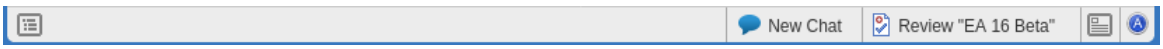
- A diagram image (see: WebEA Main View - DiagramWebEA Main View - Object Properties)
- The properties of an element (see: WebEA Main View - Object Properties)
- A list of objects that are contained in a specific Package (or objects that are children of a specific element); this object list is only accessible when the 'Browser' is not displayed, as the 'Browser' functions as an alternative. (see: WebEA Main View - Object List)

Properties View



The properties view displays properties of the selected object. When viewing a diagram, it allows you to view some properties of an element without navigating away from the diagram. See the WebEA - Properties View topic for more information.

Status bar



The WebEA Status bar is displayed at the very bottom of the page; it provides these options and information.

Icon	Description
Browser	Click this button to hide/show the 'Browser' view on the left side of the main content area. This area will contain the currently selected object and a list of all of its child objects. Between the object name and the name of the first child object is a ... icon; clicking on this icon refreshes the display to show the

	next-highest level of the hierarchy. Note: This option is disabled when the screen is less than 800 pixels in width.
New Chat	If WebEA has Chat enabled (in the WebEA configuration) and you have received one or more new chat message then "New Chat" is displayed in the status bar. Clicking on this text will provide a menu for accessing the Chat.
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 Properties button. Click on the name to display the Review element in WebEA's Main View.
Properties 	Click this button to hide/show the 'Properties View' on the right side of the main content area. When viewing a diagram the Properties View allows you to access properties of an element without navigating away from the diagram. When viewing an element, the Properties View allows you to have a split view of the element's properties, displaying one feature in the Main View while also

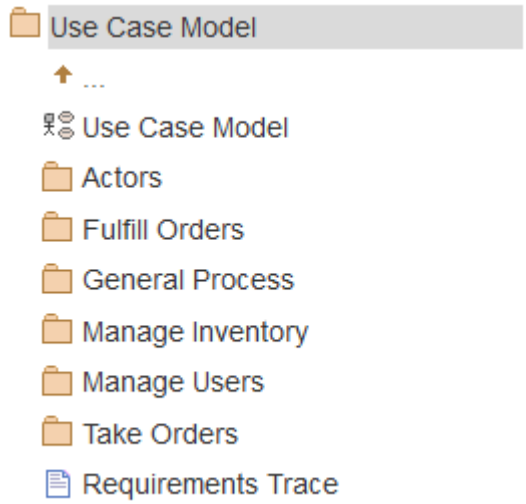
	showing another in the Properties View (for example, you could add Discussions in the Main View while referring to the element's notes in the Properties View). Note: This option is only accessible when the screen is more than 800 pixels in width.
About 	Click on this button to display the 'About WebEA' screen. The 'About...' screen provides information regarding the version of WebEA, Pro Cloud Server, and OSLC, as well as license and model information.
iOS Scrolling	When accessing WebEA via iPad or iPhone, the status bar will display 'iPad Diagram Scrolling: Off' or 'iPhone Diagram Scrolling: Off' respectively. Touch this text to toggle the option on and off. Typically, images can be scrolled on mobile devices by touching/dragging on the image; however, iOS browsers will not scroll if you are touching a clickable area (such as an element on a diagram). This could make scrolling difficult on diagrams that are densely populated with clickable elements.

	• iPad/iPhone Diagram Scrolling: On - Diagram elements will not be selectable, allowing you to scroll by touching/dragging anywhere on the diagram • iPad/iPhone Diagram Scrolling: Off - Diagram elements will be selectable; however, to drag the diagram image you will need to ensure you are not touching a selectable element

WebEA - Browser

The WebEA Browser displays a context sensitive view of a Package or element and its children. It can be toggled on or off via the 'Browser' button, which is located in the status bar at the bottom of WebEA. When enabled, the Browser displays on the left side of WebEA's Main View.

Show/Hide Browser button



Optionally, the browser can be configured to include diagram objects such as notes and text elements. See the 'Show Diagram Objects' setting in the Edit WebEA Model Connection settings topic for details.

If the currently-selected object is a Package, or an element with children, then the Browser displays the object at the parent level, and the children underneath it.

If the currently-selected object is a diagram or an element without children, then the Browser displays the object's parent, then its children (which include the current object and its siblings).

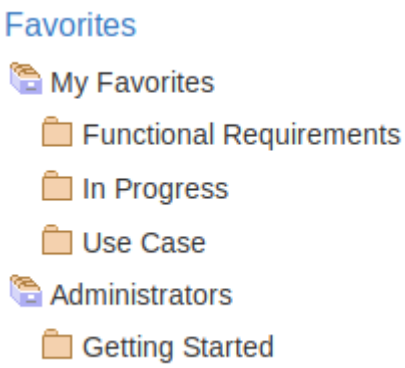
If the currently selected object is not the root, then the second item in the Browser is an up arrow, allowing navigation to the next level up. Selecting an object in the Browser will cause WebEA to navigate to that object.

Favorites in the WebEA Browser

WebEA Model connections can be configured to treat the user's 'Favorites' as the Home page (via the 'favorites_as_home' configuration option); see the Refine Browser Content and How to configure WebEA models Help topics for details.

In this case the Browser behavior is changed to help focus on Favorites Packages.

- When initially opening the model, the Browser will display your Favorites
- Selecting a Package from the Favorites list will update the Browser to display the contents of that Package; to help keep the focus on the contents of the Favorites, the Browser will not include the Up arrow to go to the Package's parent - when navigating down through nested Packages you can simply use the Web Browser's Back function to return to the previous page (i.e. the parent)
- You can return to the list of Favorites using WebEA's Home button

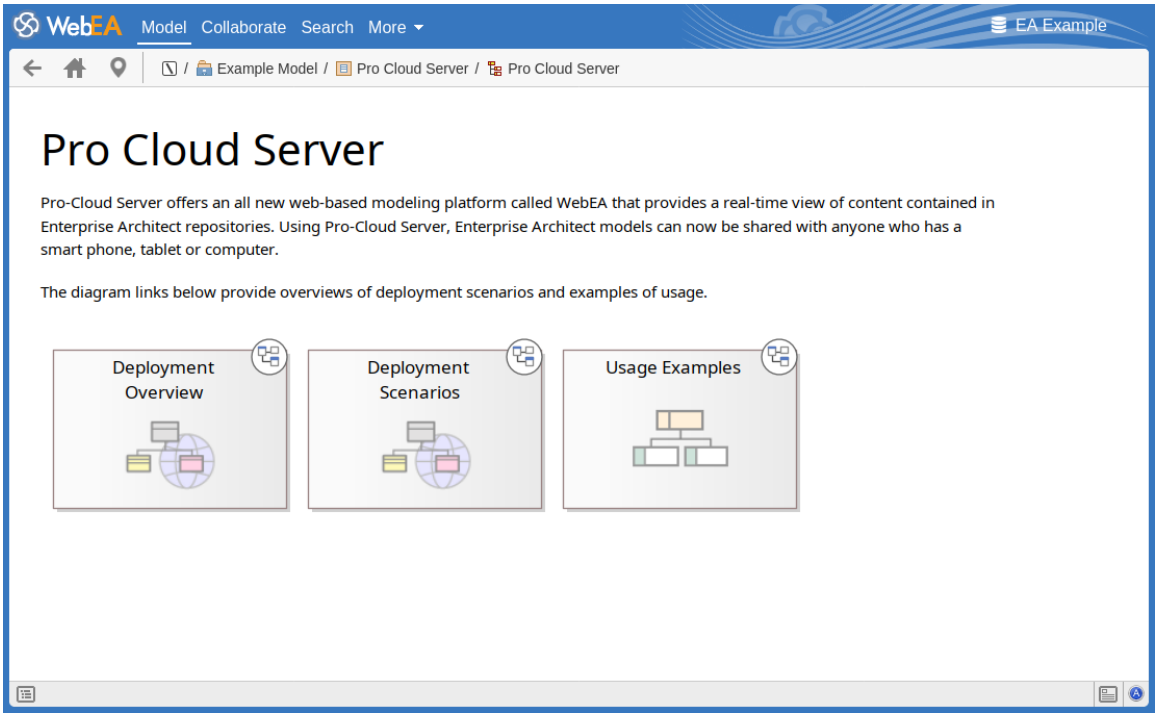


Notes

- The Browser panel is only visible if the screen is greater than 800 pixels wide
- The Browser acts as an alternative to viewing the object list in WebEA's Main View; when the Browser is enabled the Main View Object List is not accessible

WebEA Main View - Diagram

When a diagram is selected, the Main View content area displays the diagram image. As an example, this is the Pro Cloud Server diagram in the EAExample model:



Objects on the diagram are selectable and, based on the object type and if the 'Properties View' is visible, will behave slightly differently.

The behavior when the 'Properties View' is not visible:

- Packages: opens the selected Package in the Object List mode
- Diagrams: opens the selected diagram image
- Elements: opens the selected element in the Main View
- Hyperlinks to external URLs: navigates away from WebEA and loads the external URL

The behavior when the 'Properties View' is visible:

- Packages: main properties are displayed in the 'Properties View'
- Diagrams: main properties are displayed in the 'Properties View'
- Elements: main properties are displayed in the 'Properties View'
- Hyperlinks: navigate to link target

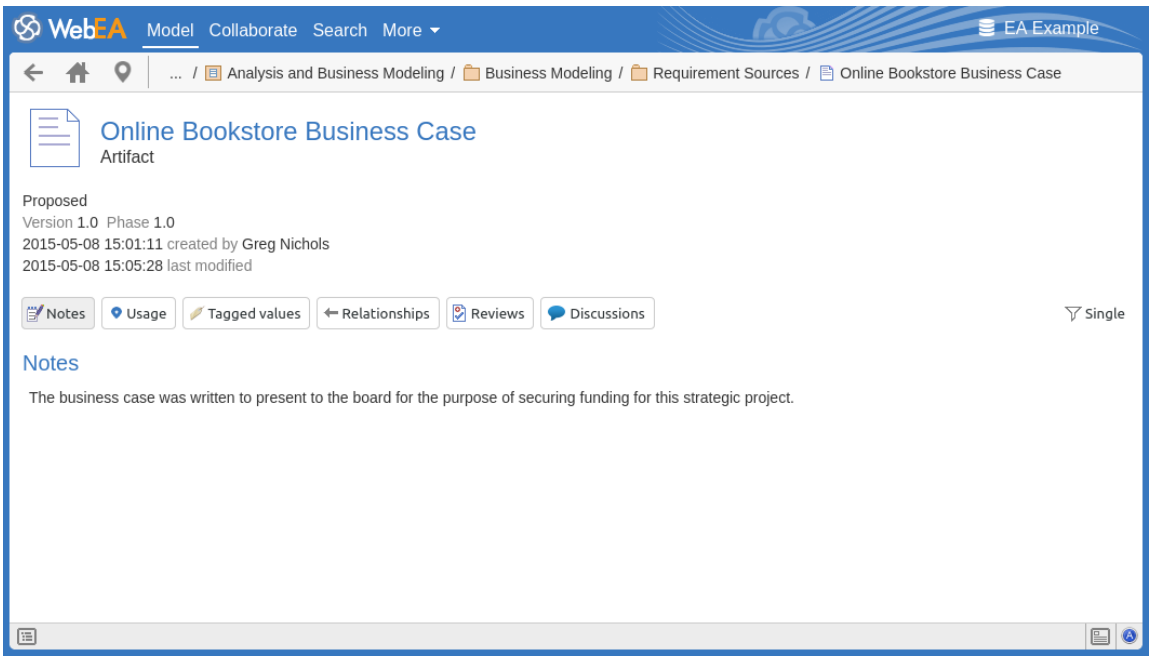
Notes

- To view the properties of a diagram either display the Properties View, or (when using a screen width less than 800 pixels) use the  button in the Navigation bar

WebEA Main View - Object Properties

When you select an element in WebEA, its properties are displayed either in the Main View or in the Properties View (if it is currently open). If the properties are displayed in the Properties View (to the right side of WebEA's Main View) then you can click on the Full Properties button to start to explore more of the element's properties in the WebEA Main View.

The property display in the Main View consists of a section at the top with common element properties, followed by a row of feature buttons (such as **Notes** and **Relationships**) which can be used to access specific features. By default the Notes are already displayed.

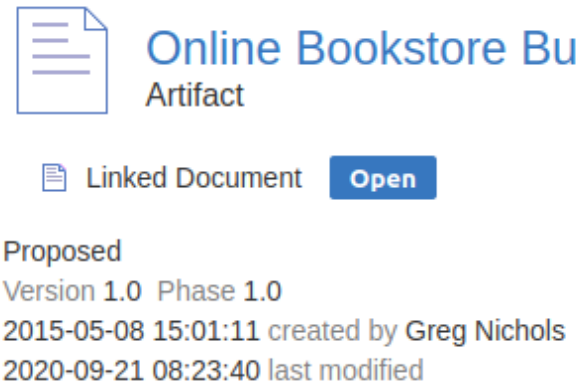


Common Properties

The common properties section displays an image

representing the object's type, and its name, type and stereotype. This is followed by general information about the object, such as the status, version and phase, the time and date created and last modified, and the author.

Note: The Properties View can also be enabled and displayed alongside the property display in the Main View. In this case the author, version, and other general characteristics will only be displayed in the Properties View.



- If the model has been configured to allow it, there will be an element 'Hamburger' icon that allows the creation of object features; see the *Object Creation in WebEA* Help topic for further details
- 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 text box 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 your machine where you can use the appropriate application to view it
- If the element has an 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 your machine where you can use the appropriate application to view it
- If you are viewing the properties of a diagram then a 'Generated' field will be displayed. This shows the date/time when the diagram image was saved, followed by a button which can be used to flag the diagram to be 're-generated' by the EA Worker application. After clicking this button, the 'Image pending regeneration' message will be displayed and the diagram image will be re-generated by the EA Worker application. See the *How to configure automatic viewable components* Help topic for details.

Selectable Features

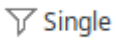
Below the common properties a number of buttons are displayed that allow you to view other existing features of the object, including location, Tagged Values, attributes and operations. By default, the element Notes will be selected and displayed.

If WebEA is configured to allow creation of objects or

object features then an Add New button will also be included. See the 'Object Creation in WebEA' topic for more information.



To the right of the feature buttons is a filter button.



Click this button to toggle between the display modes:

- **Single** - In this mode, clicking on a feature button will display only that feature type, such as Notes or Tagged Values
- **All** - In this mode, all the element's features are displayed on the page, one after the other; clicking a feature button will scroll down (if necessary) so that the section for that feature is visible

These buttons are available to display the corresponding features:

- Notes: the notes of the current object
- Location: information relating to the current object's parent, any diagrams that the element appears on, and a list of instances that the element is a classifier of
- Tagged Values: a list of defined Tagged Values for the current object
- Relationships: a list of incoming and outgoing connections for the current object; clicking on the connector type will display the connector's properties

- Attributes: a list of all defined attributes for the current object
- Operations: a list of all defined operations for the current object
- Files: a list of Associated Files that have been defined for the current object
- Requirements: a list of all defined internal requirements (responsibilities) for the current object
- Constraints: a list of all defined constraints for the current object
- Scenarios: a list of all defined scenarios for the current object
- Tests: a list of all defined tests for the current object; selecting a test from the list will display more information - 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
- Resources: a list of all defined Resource Allocations for the current object; selecting a resource from the list will display more information - 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
- Features: a list of all defined 'feature' maintenance items for the current object; selecting a change from the list will display more information
- Changes: a list of all defined 'change' maintenance items for the current object; selecting a change from the list will

display more information

- Documents: a list of all defined 'document' maintenance items for the current object; selecting a document from the list will display more information
- Defects: a list of all defined 'defect' maintenance items for the current object; selecting a defect from the list will display more information
- Issues: a list of all defined 'issue' maintenance items for the current object; selecting an issue from the list will display more information
- Tasks: a list of all defined 'task' maintenance items for the current object; selecting a task from the list will display more information
- Events: a list of all defined events for the current object; selecting an event from the list will display more information
- Decisions: a list of all defined decisions for the current object; selecting a decision from the list will display more information
- 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
- Reviews: a list of all formal review discussions; if the current model has been configured to allow it, you can contribute to Review Discussions. See the *Reviews in WebEA* Help topic for details.

- Discussions: a list of discussions on the element; if the current model has been configured to allow it, you can contribute to discussions. See the *Discussions in WebEA* Help topic for details.
- Comments: the Comments on the current element. Initially only one Comment is displayed (to improve performance). You can load all comments by clicking on the 'Show All' button.

Opening Associated Files from WebEA

If an element has Associated Files defined, the file paths will be listed as hyperlinks in the 'Files' section.

For security reasons, web browsers typically will not allow local files to be opened directly via a hyperlink on a web page. In some cases the browser can be configured to allow this.

For example, for FireFox, see:

http://kb.mozillazine.org/Firefox_:_Issues_:_Links_to_Local_Pages_Don%27t_Work

At the end of each file path is a 'Copy to Clipboard' icon. This can be used to easily copy and paste the path (for example, into Windows Explorer) and open the file.

WebEA Main View - Object List

If the WebEA browser is not currently displayed, then selecting a Package will display the Package contents in the central Main View of WebEA. This is referred to as the 'Object List'. The Object List can be displayed in one of three viewing styles: Icon, List or Notes view.

It is possible to define a default viewing style for a model in the configuration file. However, you can also change the style via the icons in the upper right of the Object List.



Icon View

In Icon view, each object is represented by an image and an object name. As an example, a root node with the name 'ASA Finance' would be represented by:



A View object with the name 'Class Model' would be represented by:



List View

In List view, each object is represented as a row in a table,

which includes columns for name, type, author and modified date. The two examples would be represented in List view as shown:

Name	Type	Author	Modified
ASA Finance	ModelRoot		2017-01-20 17:21:03

Name	Type	Author	Modified
Class Model	Package	admin	2017-11-15 10:40:54

Notes View

In Notes view each object is represented by the object name and any text from the object's 'Notes' field.

Display Order

In the Object List, 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 alphabetical order. This is different to Enterprise Architect's behavior, which can allow elements to be manually ordered.

As an example:



Additional Symbols

While navigating the model structure with WebEA you might see a number of symbols applied to the object images. Each symbol has a meaning; for example, 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 Browser window 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 for the selected element

To view the properties of a Package while viewing it in the Object List, click on the  button in the status bar to display the Properties View. Alternatively, on lower resolution screens (less than 800 pixels wide), where the Properties View is not available, you can use the  button to switch to the Package's properties.

If you navigate 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 icon will be shown as the last element of a Package's Object List. For more details see the *Object Creation in WebEA* Help topic.

Notes

- WebEA's WebEA - Browser acts as an alternative to the Object List; if the WebEA Browser is displayed then the Object List is not accessible

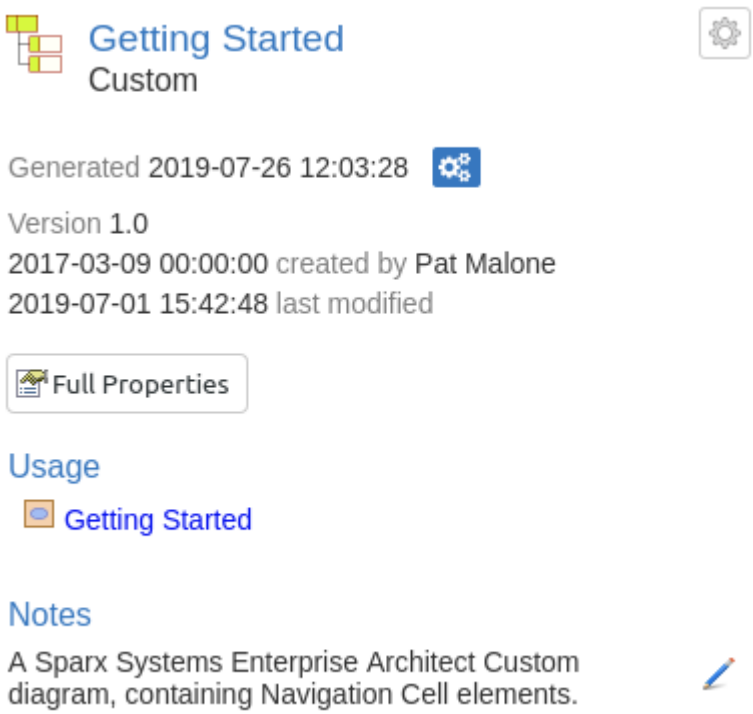
WebEA - Properties View

The WebEA **Properties View** displays a context-sensitive view of a diagram, Package or element's properties. It can be toggled on or off via the Properties button, which is located in the status bar at the bottom of WebEA.

 Show/Hide Properties View button

The Properties View displays the principle property details of the object currently selected in the main content area, these being the name, type, notes, creation and modification dates, version and location details.

By default, the Properties View will also display the element Notes; however, you can select additional features to display by using the **Feature Visibility** button at the top right of the Properties View.



The main purpose of the Properties View is to allow users to quickly and easily see details of elements without navigating away from a diagram.

The Properties View is also displayed while viewing the Full Properties of an object in WebEA's Main View. This allows you to access one feature of the element (such as Notes) in the Properties View while also accessing another (such as Discussions) in the Main View.

Configure Properties View

Click on the Feature Visibility button at the top right of the Properties View to configure which features/sections are displayed in the properties view.



Clicking this button will display the 'Configure Properties View' dialog, which provides these options.

Option	Description
Feature List	The feature list on the left side of the dialog includes buttons to toggle on/off each features. Items within this list can also be dragged up or down to change their order. Note, WebEA model connections can be configured to hide/disable specific features. If a feature is disabled via the WebEA configuration then it will not be available in this list.
Hide Empty Feature Sections	When enabled this option will hide sections which do not have any content. When disabled, headings for the enabled/visible sections will be displayed even when they have not content. Note, if a section includes the option to add content then if will still be displayed even when Hide Empty Feature Sections is enabled.
Set All Visible	Turns on the Visible setting for all features in the Feature List.
Sets All Hidden	Turns off the Visible setting for all features in the Feature List.

Restore Default Visibility	Restores the default setting (Visible On or Off) for all items in the Feature List.
Restore Default Order	Restores the default order for items in the Feature Visibility list.

Navigation Buttons

When appropriate, the Properties View will also include navigation buttons that allow you to either display the object's full properties in WebEA's Main View, or view the element/Package's children.

Button	Description
	This can be used to navigate to the object, showing its full properties in the Main View.
	If you have selected a Package or an element with children, and the 'Browser' is not currently enabled, then an additional icon is displayed. This allows you to view the object's children in the Object List/Icon view.

Navigation Element Behavior

Within Enterprise Architect it is possible to define a number of different element types that make it easier to navigate your model (such as Hyperlinks or Navigation Cells). By default, selecting one of these objects in WebEA will navigate directly to the target (instead of displaying its properties in the Properties View).

If necessary, you can change this behavior by setting the 'miniprops_navigates' option in the 'Configure WebEA models - via Text Editor' file to "false".

Notes

- The Properties View is only visible if the screen is greater than 800 pixels wide
- Selecting elements on a diagram while the Properties View is shown will not affect the browsing history

Object Creation in WebEA

Depending on how the current model has been configured in the WebEA configuration file, and on 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 Packages and diagrams; 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 Objects

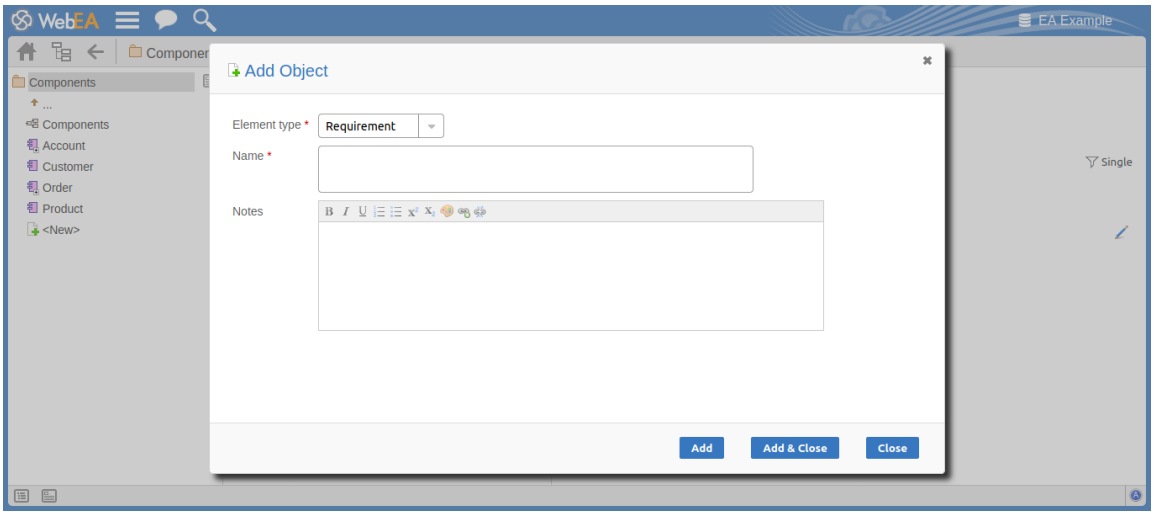
You can quickly establish if you have the ability to create objects in WebEA by checking the WebEA - Browser or the WebEA Main View - Object List.

If you are able to create objects then the final item in the list will be a '<new>' button.

For example:



Click on this button to display the 'Add Object' page.



Field	Description
Element type	Click on the drop-down arrow and select the type of object to create. Depending on your security permissions and the model's options, you can create some or all of these objects: • Package • Diagram • Review • Actor • Change • Component • Feature • Issue

	• Node • Requirement • Task • Use Case
Technology	This is an optional field that is only visible when 'Diagram' is selected as the element type. It allows the selection of a Technology that provides an additional set of diagram types.
Diagram Type	This is an optional field that is only visible when 'Diagram' is selected as the element type. Click on the drop-down arrow and select the type of diagram to create.
Name	Type in an appropriate name for the object, 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 • Color of selected text
Add	Click on this button to add the new object. This message displays: Object added Either: • Work through the fields again, defining another new object in this Package, or • Click on the Close button to close the dialog.
Add & Close	Click on this button to add the object and also close the dialog.
Close	Click on this button to close the dialog.

Adding Element Features

You can, again, quickly establish if you have the ability to

create element features in WebEA, by displaying the Properties of an element in WebEA's WebEA Main View - Object Properties; if the 'Add New' button is displayed before the elment feature buttons ('Notes', 'Usage', etc)



Display the Properties of the element to which you want to add a feature, and click on the 'Add New' button. A menu displays offering options to add to the element each feature that you have the access permissions to create. These features might include:

- Other Objects
- Tests
- Resources
- Features
- Changes
- Documents
- 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 a Feature*
- *Add Change Requests*
- *Add a Document*
- *Add a Defect*
- *Add an Issue Notification*
- *Add a Task*
- *Add a Risk*

Notes

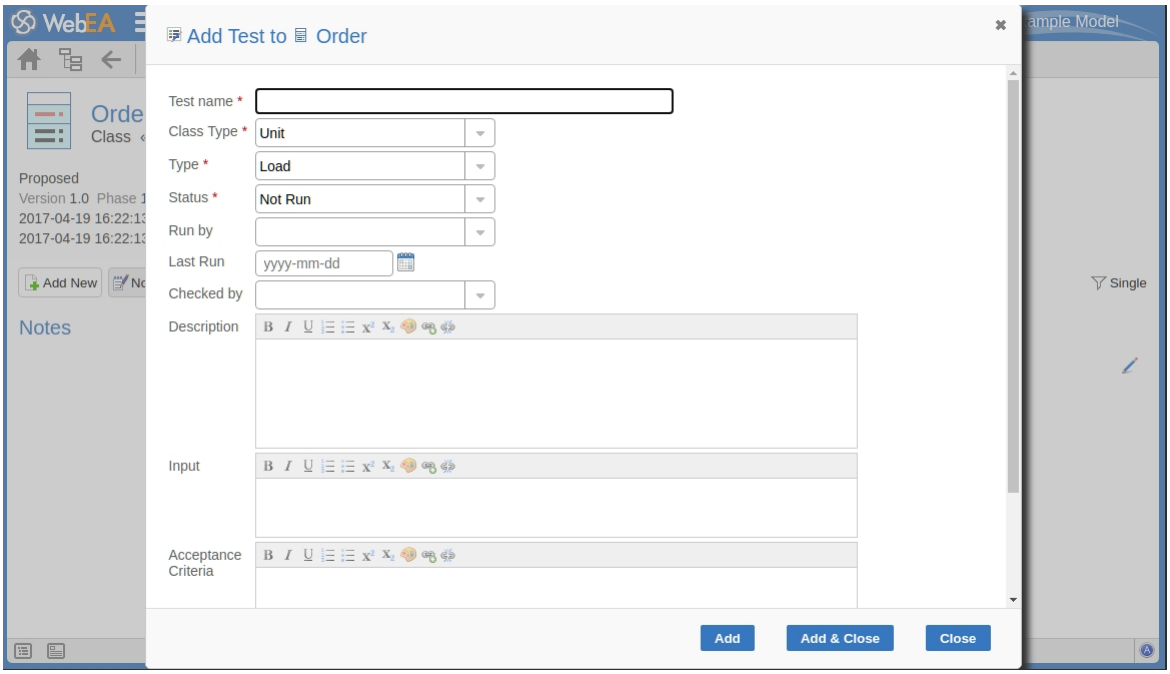
- As suggested in the *Adding Element Features* section, you can also add an object to a Package or to another object within the Main View's 'Object Properties' page; click on the 'Hamburger' icon and select the 'Add Object' option, and complete the 'Add Object' page as described at the start of this topic
- When using the security 'Lock to Edit' mode, features and child elements can only be created if the element has been locked by the current user (via Enterprise Architect); users require 'Lock Elements' permission to release or set locks on a diagram element

Add Test Records

Within an element, you can create a record for each Test that you want to add to the element.

Add a Test

Display the properties of the element to which you want to add the Test record, and click on the 'Add New' button. A menu displays offering options to add various features to the element. 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
	Type an appropriate and meaningful

Test name	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 Close button to close the dialog
Add & Close	Click on this button to add the Test item

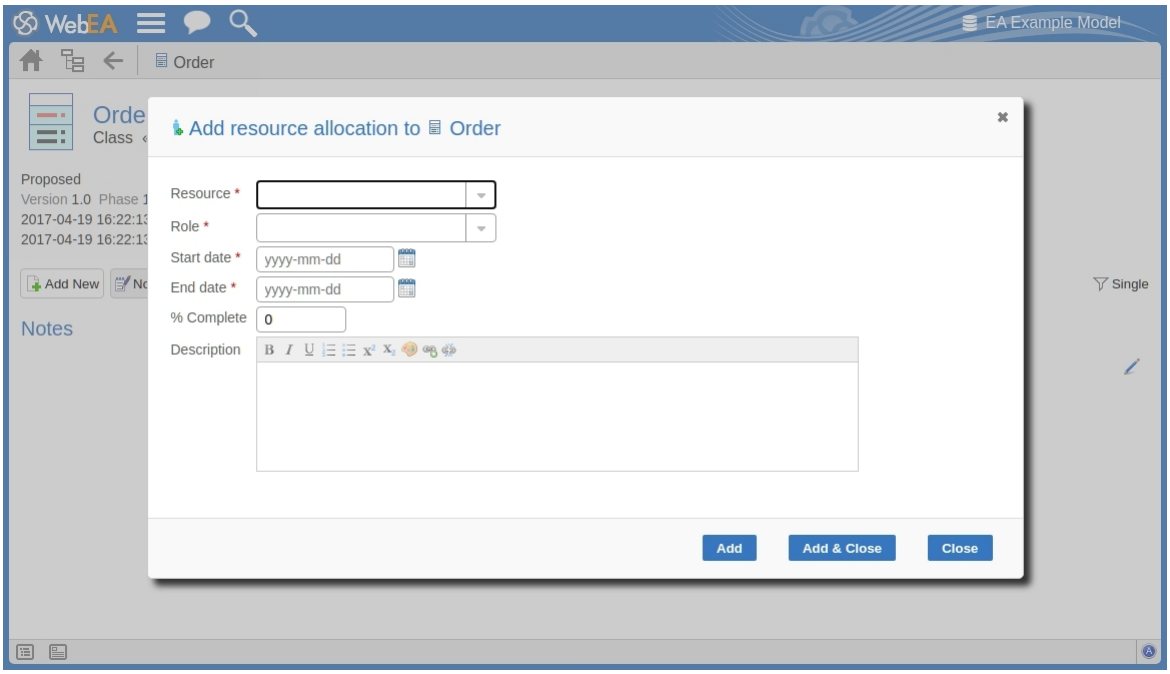
	(as described above) and also Close the dialog.
Close	Click on this button to close the dialog.

Add Resource Records

Within an element, you can create a record for each Resource that you want to add to the element.

Add a Resource

Display the properties of the element to which you want to add the Resource record, and click on the 'Add New' button. A menu displays offering options to add various features to the element. 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
	Click on the drop-down arrow and select

Resource	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 or • 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) resource 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 Close button to close the dialog
Add & Close	Click on this button to add the Resource item (as described above) and also Close

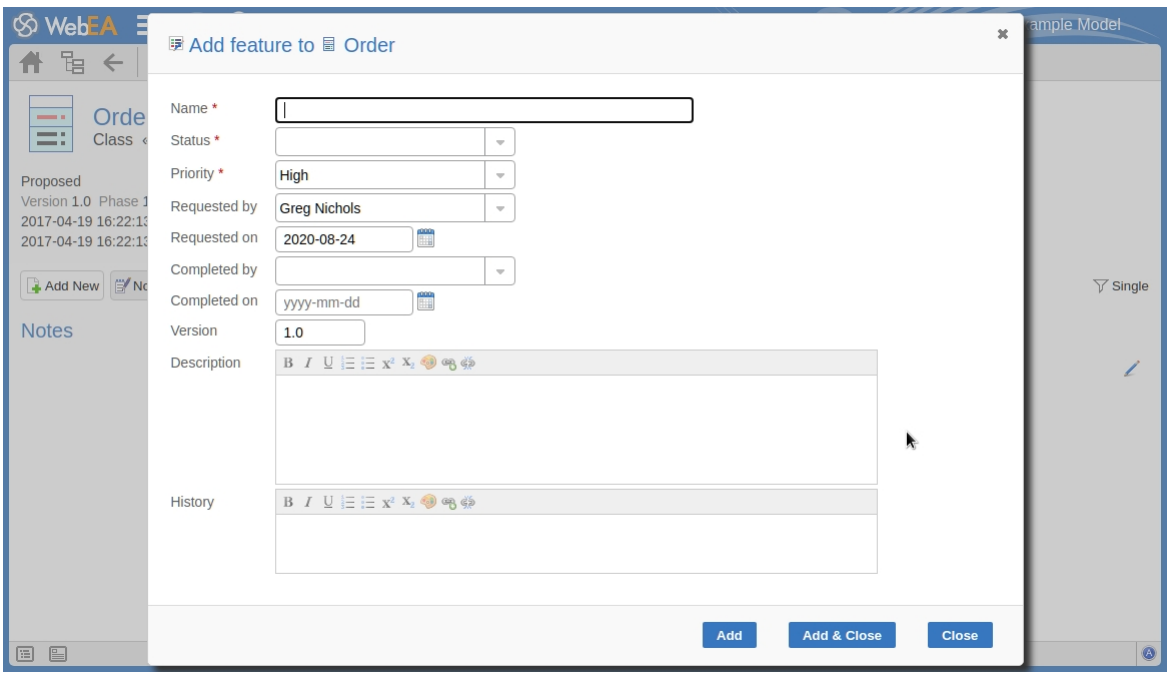
	the dialog.
Close	Click on this button to close the dialog.

Add a Feature

Within an element, you can create a record for each 'feature' maintenance item that you want to add to the element.

Add a Feature

Display the properties of the element to which you want to add the feature, and click on the 'Add New' button. Select the 'Add feature' option. The 'Add feature 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 feature.
Status	Click on the drop-down arrow and select the appropriate record 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 feature.
Requested on	Click on the calendar icon and select the date on which the record was made.
Completed by	If the feature has been completed/added, click on the drop-down arrow and select the name of the person who complete it.
Completed on	Click on the calendar icon and select the date on which the feature was completed/added.

Version	Type in a version number or letter to indicate any evolution of the same feature. Generally, '1.0' indicates the first instance of a feature.
Description	Type in a description of the feature. 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	Over time, enter any notes on the actions concerning this feature; 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 feature record on the selected element. This message is displayed: Feature <name> was successfully

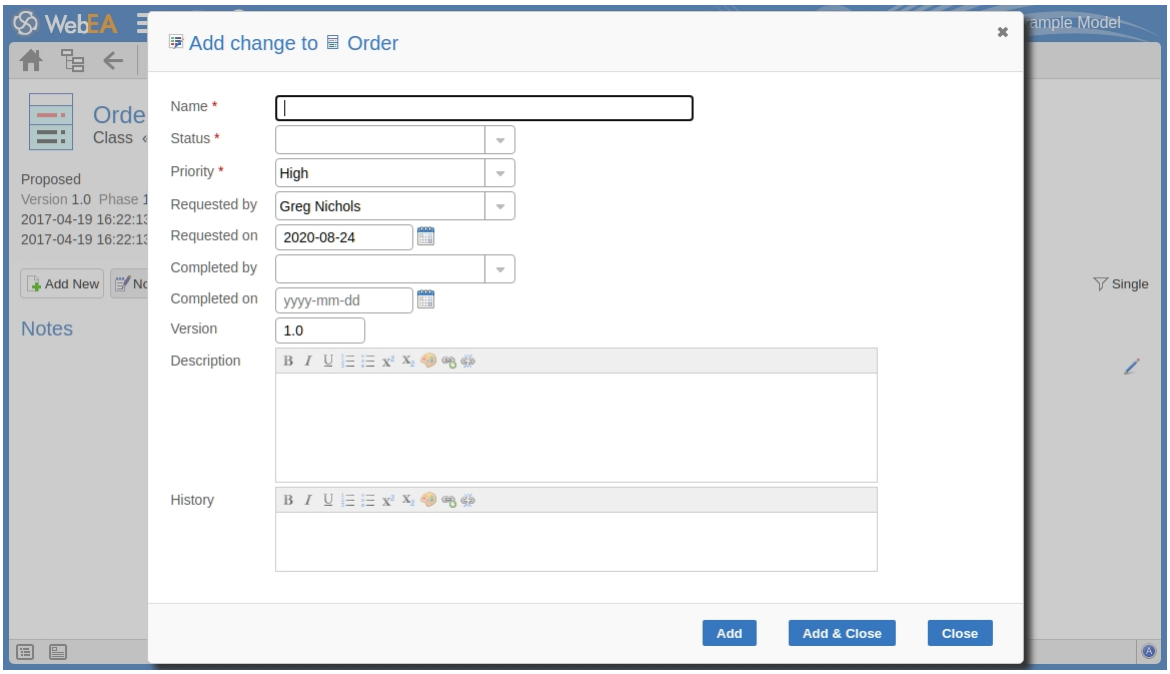
	added to <element name>. Either: • Work through the fields again, defining another feature on this element, or • Click on the Close button to close the dialog
Add & Close	Click on this button to add the Feature item (as described above) and also Close the dialog.
Close	Click on this button to close the dialog.

Add Change Requests

Within an element, you can create a record for each Change Request that you want to add to the element.

Add a Change Request

Display the properties of the element to which you want to add the Change Request, and click on the 'Add New' button. A menu displays offering options to add various features to the element. 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
	Type an appropriate and meaningful

Name	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.

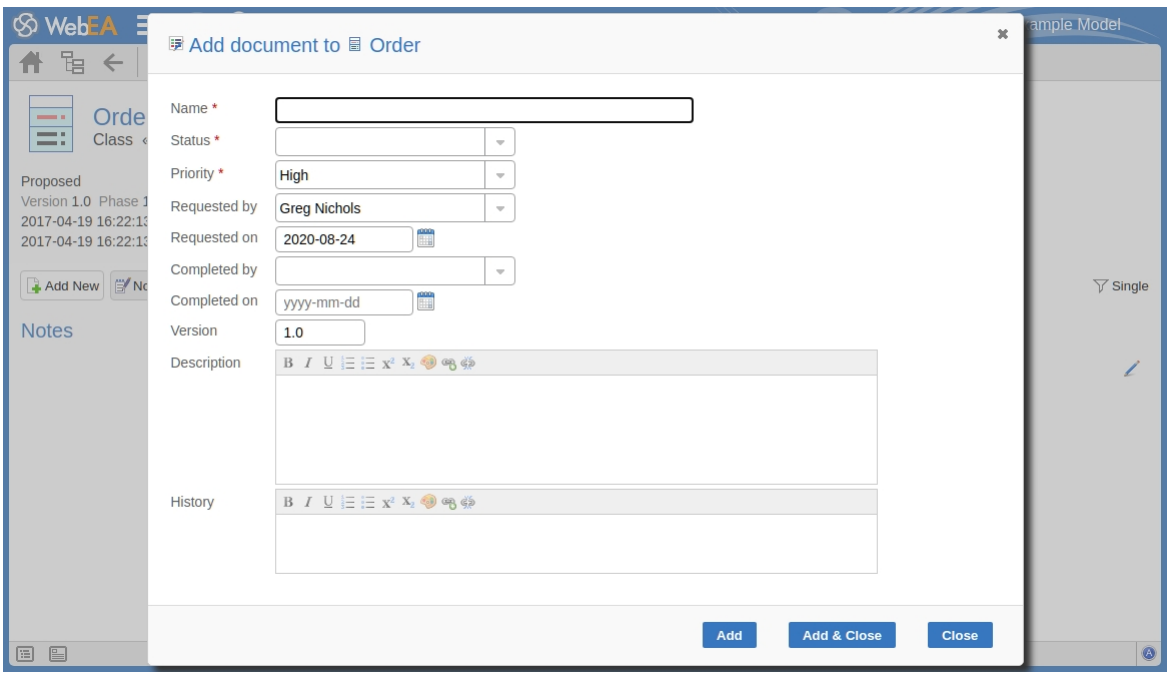
	You can format this text using the facilities provided in the Notes toolbar.
Add	Click on this button to submit the details and create the change request on the selected element. This message is displayed: Change <name> was successfully added to <element name>. Either: • Work through the fields again, defining another new change request on this element, or • Click on the Close button to close the dialog
Add & Close	Click on this button to add the Change item (as described above) and also Close the dialog.
Close	Click on this button to close the dialog.

Add a Document

Within an element, you can create a record for each document maintenance item that you want to add to the element. A document could be used to record supporting documentation such as Help, release notes or user comments.

Add a Document

Display the properties of the element to which you want to add the document, and click on the 'Add New' button. Select the 'Add document' option. The 'Add document 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 document.
Status	Click on the drop-down arrow and select the appropriate record 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 who requested the document.
Requested on	Click on the calendar icon and select the date on which the record was made.
Completed by	If the document is complete, 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 document was completed.
Version	Type in a version number or letter to indicate any evolution of the same document. Generally, '1.0' indicates the first instance of a document.
Description	Type in the details/contents of the document. 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	Over time, enter any notes on the actions concerning this document; you can format this text using the facilities provided in the Notes toolbar.
Add	Click on this button to submit the details

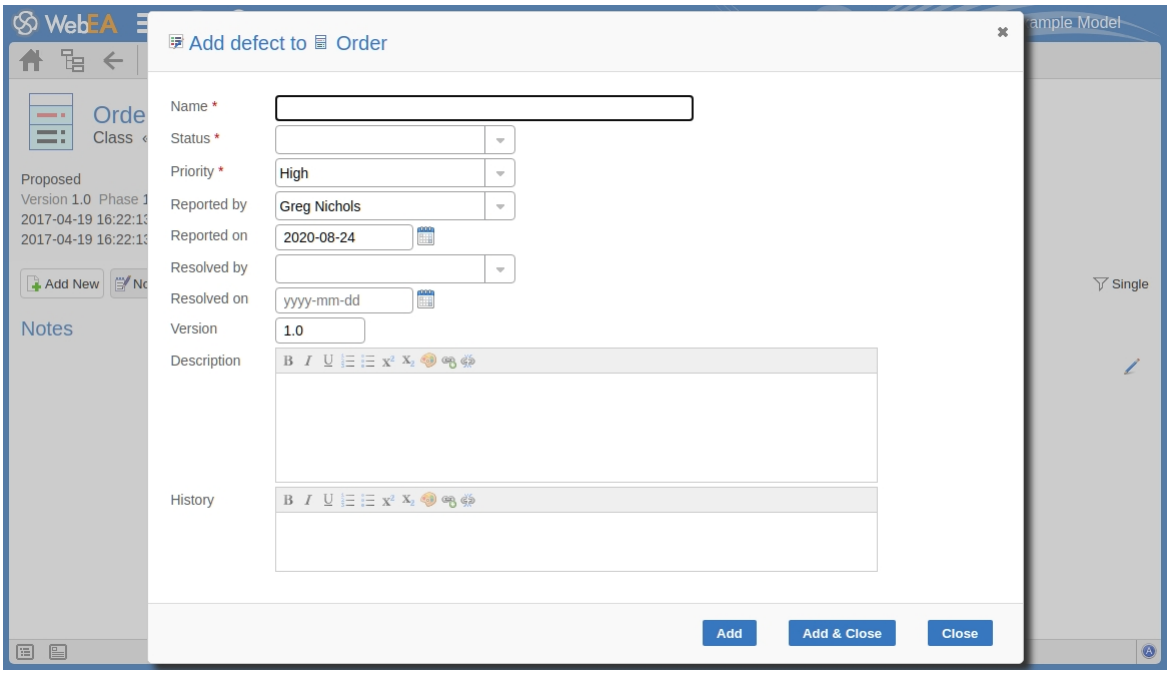
	and create the document record on the selected element. This message is displayed: Document <name> was successfully added to <element name>. Either: • Work through the fields again, defining another document on this element, or • Click on the Close button to close the dialog
Add & Close	Click on this button to add the Document item (as described above) and also Close the dialog.
Close	Click on this button to close the dialog.

Add a Defect

Within an element, you can create a record for each defect that you want to report on the element.

Add a Defect Record

Display the properties of the element to which you want to add the defect record, and click on the 'Add New' button. A menu displays offering options to add various features to the element. 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
	Type an appropriate and meaningful

Name	name for the defect.
Status	Click on the drop-down arrow and select the appropriate record 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 record 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 recorded 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 occurrences, document them here (you might copy the 'Description' from the last version or iteration of this defect record, for example). It is recommended that you include a date stamp and user initials in

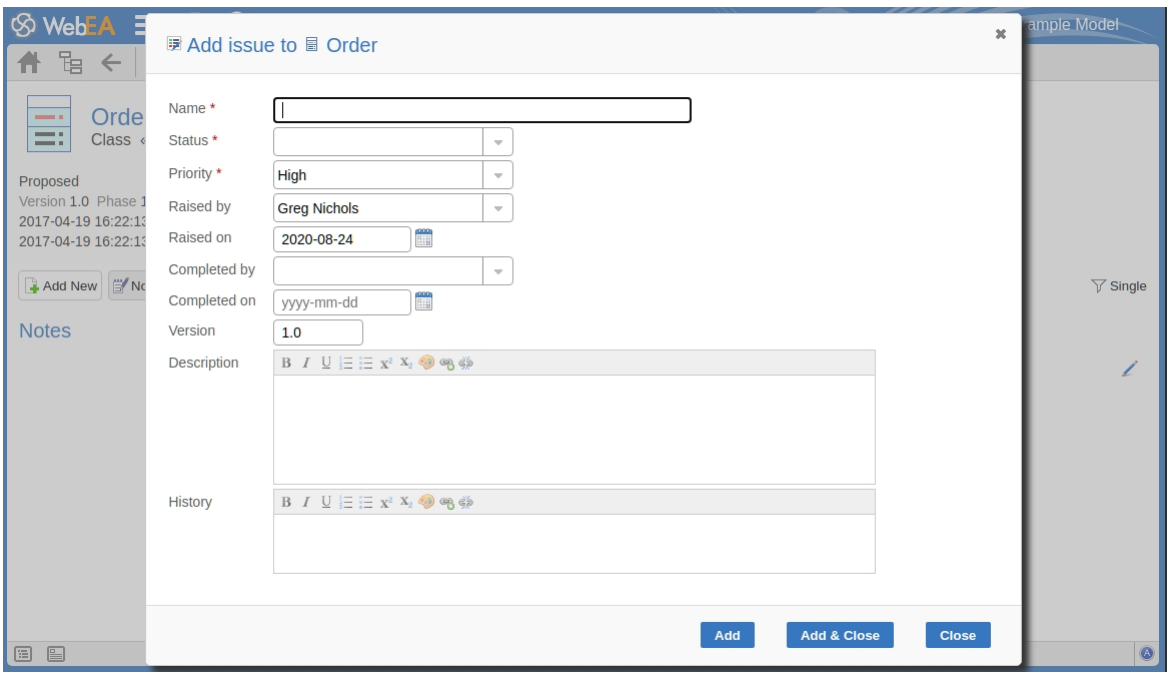
	the entry. You can format this text using the facilities provided in the Notes toolbar.
Add	Click on this button to submit the details and create the defect record on the selected element. This message is displayed: Defect <name> was successfully added to <element name>. Either: • Work through the fields again, defining another new defect record on this element, or • Click on the Close button to close the dialog
Add & Close	Click on this button to add the Defect (as described above) and also Close the dialog.
Close	Click on this button to close the dialog.

Add an Issue Notification

Within an element, you can create a record for each issue notification that you want to add to the element.

Add an Issue

Display the properties of the element to which you want to add the issue notification, and click on the 'Add New' button. A menu displays offering options to add various features to the element. 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
	Type an appropriate and meaningful

Name	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	Click on the calendar icon and select the

on	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 occurrences, 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

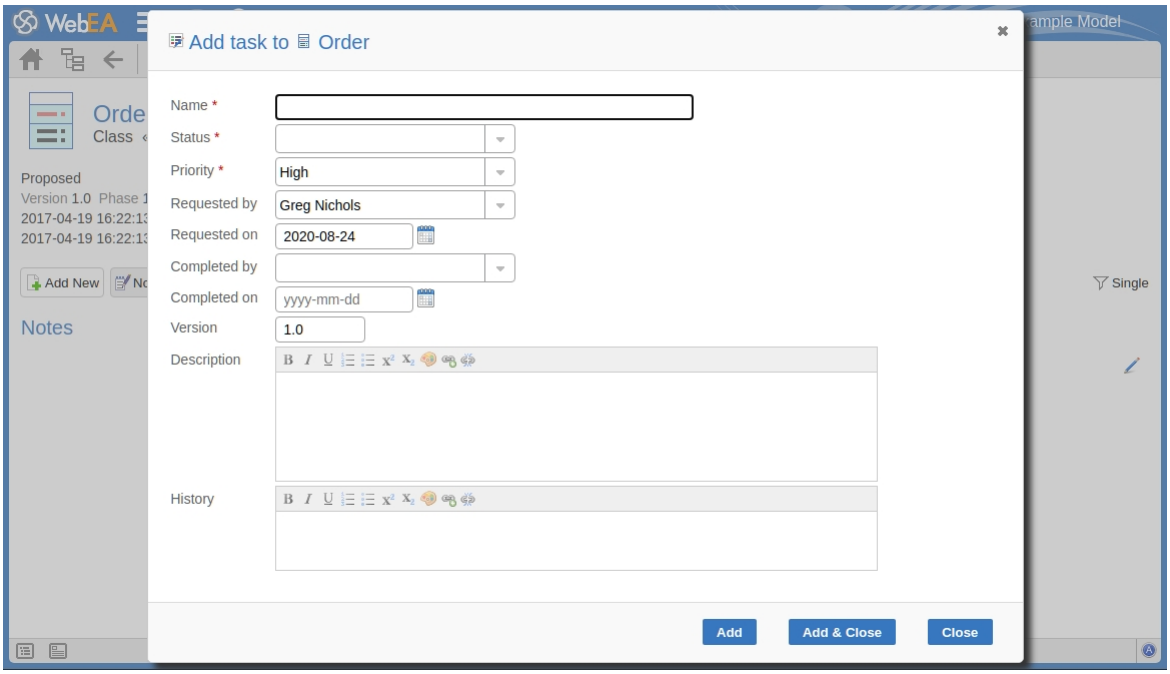
	stamp and user initials in the entry. You can format this text using the facilities provided in the Notes toolbar.
Add	Click on this button to submit the details and create the issue report on the selected element. This message is displayed: Issue <name> was successfully added to <element name>. Either: • Work through the fields again, defining another new issue on this element, or • Click on the Close button to close the dialog
Add & Close	Click on this button to add the Issue (as described above) and also Close the dialog.
Close	Click on this button to close the dialog.

Add a Task

Within an element, you can create a record for each task that you want to add to the element.

Add a Task Record

Display the properties of the element to which you want to add the task record, and click on the 'Add New' button. A menu displays offering options to add various features to the element. 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
	Type an appropriate and meaningful

Name	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.
Requested 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 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 occurrences, 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 Close button to close the dialog
Add & Close	Click on this button to add the Task item (as described above) and also Close the dialog.
Close	Click on this button to close the dialog.

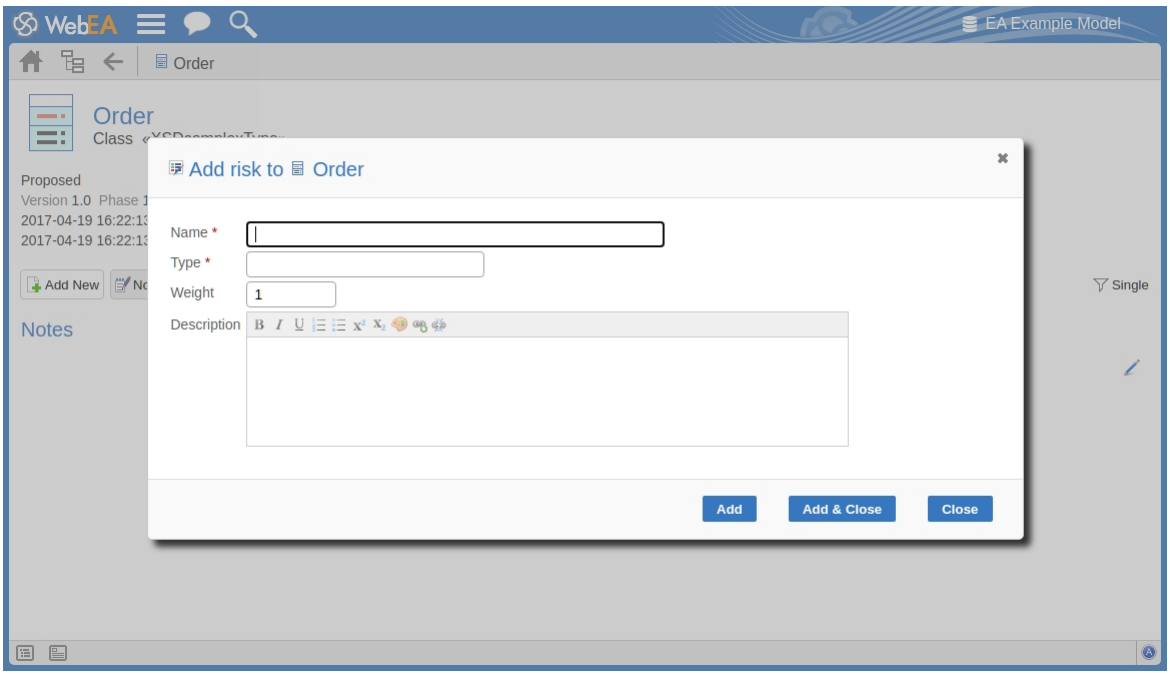
Add a Risk

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.

Within an element, you can create a record for each risk that you want to record against the element.

Add a Risk Record

Display the properties of the element to which you want to add the risk record, and click on the 'Add New' button. A menu displays offering options to add various features to the element. 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 Close button to close the dialog
Add & Close	Click on this button to add the Risk item (as described above) and also Close the dialog.
Close	Click on this button to close the dialog.

Edit Objects

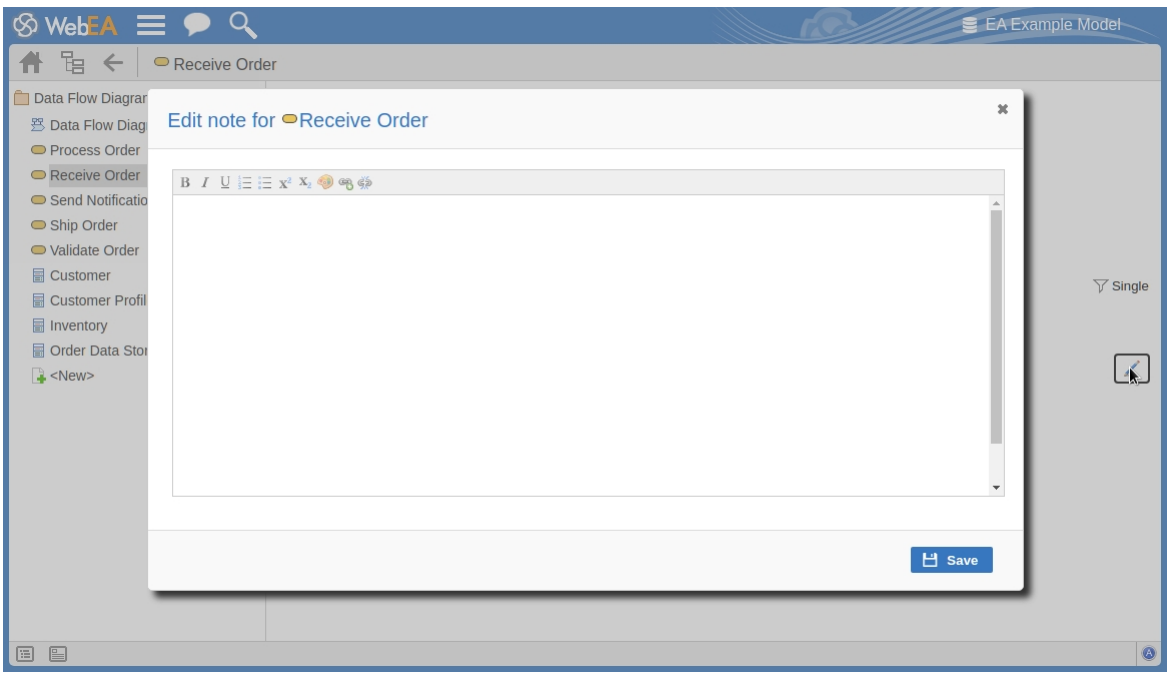
Within any model in which you have the security permissions to create elements, you can also edit the Notes, Tagged Values, Resource Allocation and Testing details of any object in which they occur.

Firstly, display the property (Notes, Tagged Values, etc) that you want to edit in WebEA's WebEA Main View - Object Properties, or in the WebEA - Properties View.

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>' dialog 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

The final two toolbar buttons allow you to add or remove a hyperlink from some selected text.

To add a hyperlink:

1. Select the text in the editor.
2. Click on the Add Hyperlink button.
3. Enter the URL for the hyperlink.
4. Click on the Apply button.

To remove a hyperlink:

1. Select the text.
2. Click on the Remove Hyperlink button.

Note: This function supports hyperlinks to Web URLs only.

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

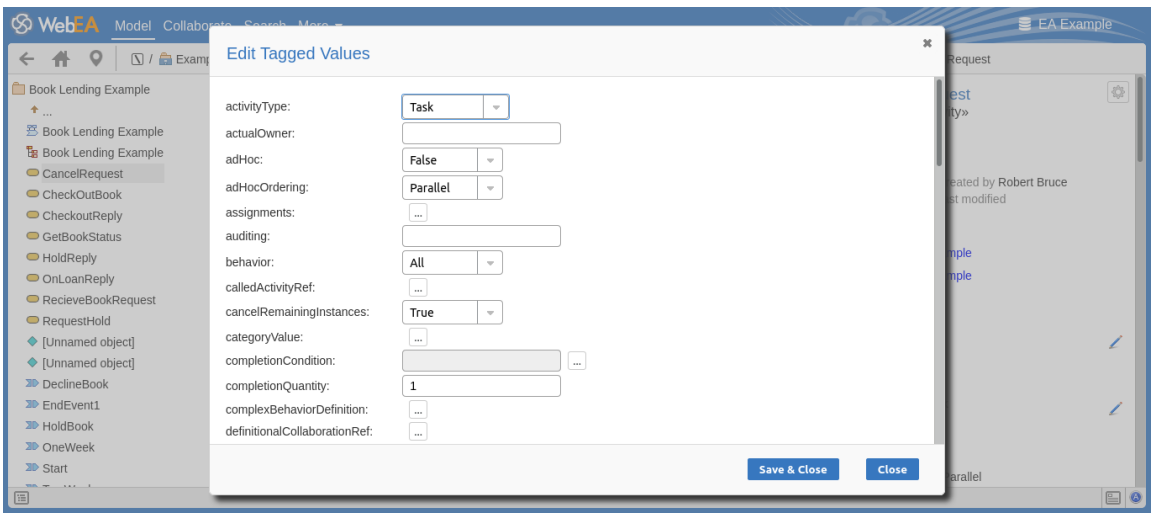
- Undo the last change
- Move (cut) the text string into the buffer for pasting

- 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
- 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)

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 Tagged Values

To edit the Tagged Values of an object, click on the icon to the right of the Tagged Values list. The 'Edit Tagged Values' dialog displays.



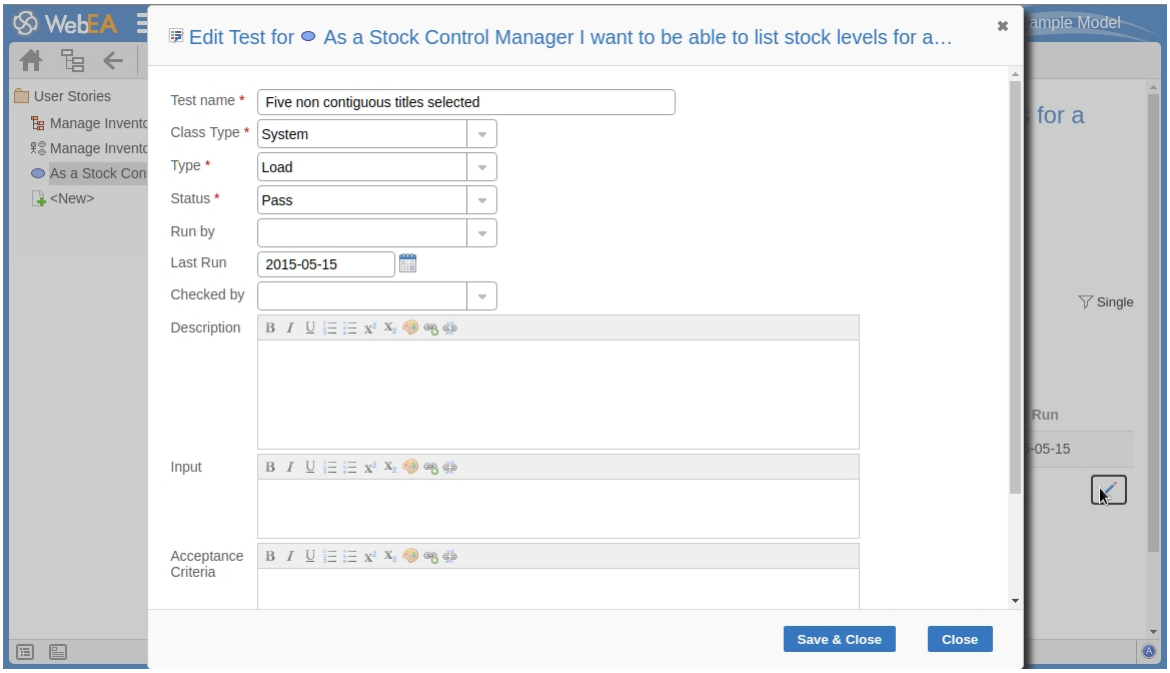
Each tag provides an editable field. Structured Tagged Value types such as 'Memo' and 'Enumeration' tags provide the appropriate controls for entering a value. After modifying one or more tags use the 'Save & Close' button to save the new value/s or select 'Close' to discard the changes.

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 WebEA's Main View, click on the Tests button and select the appropriate test from the list.

Once the test is displayed click on the  icon. The 'Edit Test for <element name>' dialog 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

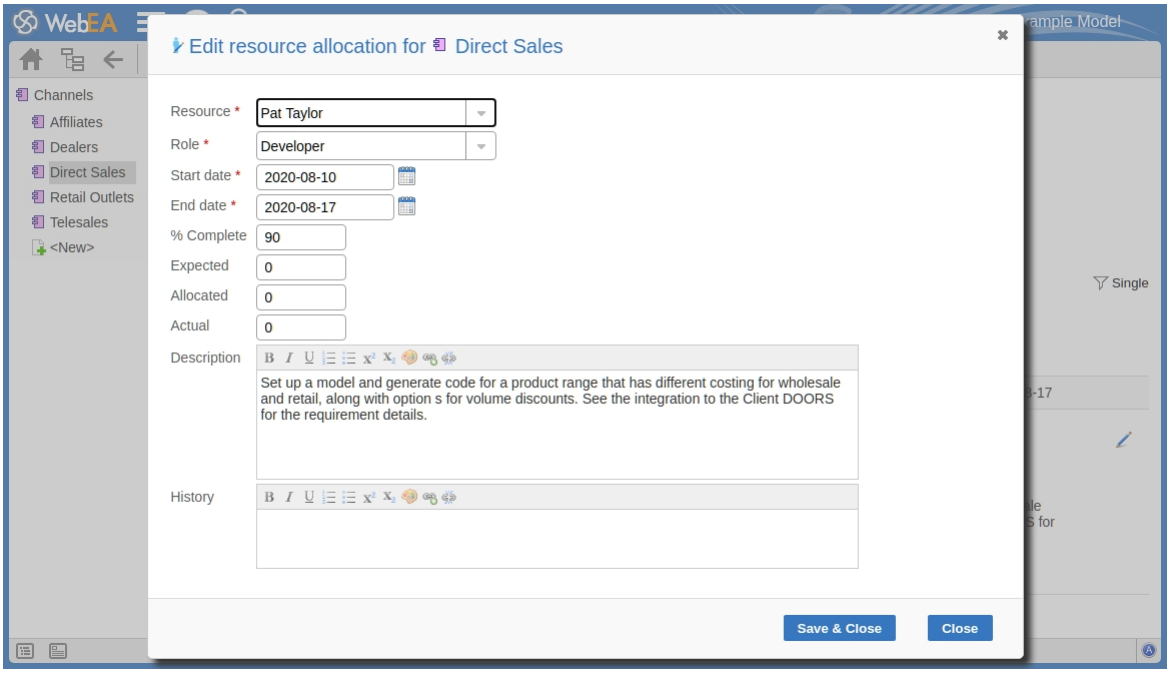
	facilities provided in the Notes toolbar.
Save	Click on this button to submit the details and update the Test record on the selected element. A confirmation message briefly displays and you are returned to the object 'Properties' view

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 WebEA's Main View, click on the Resources button and select the appropriate resource from the list.

Once the resource is displayed, click on the  icon. The 'Edit resource allocation for <element name>' dialog 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.

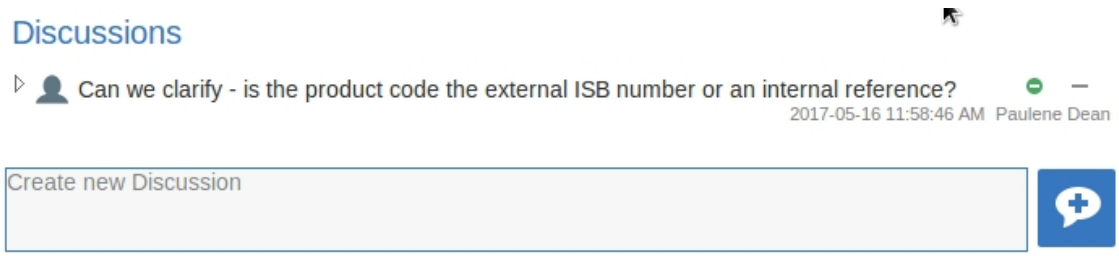
Save	Click on this button to submit the details and update the Resource Allocation record on the selected element. A confirmation message briefly displays and you are returned to the object 'Properties' view.

Discussions in WebEA

Enterprise Architect has a Discussion facility through which users can post short messages in 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.

To access Discussions, when viewing the properties of an object in WebEA's Main View, select the Discussions feature button.

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.

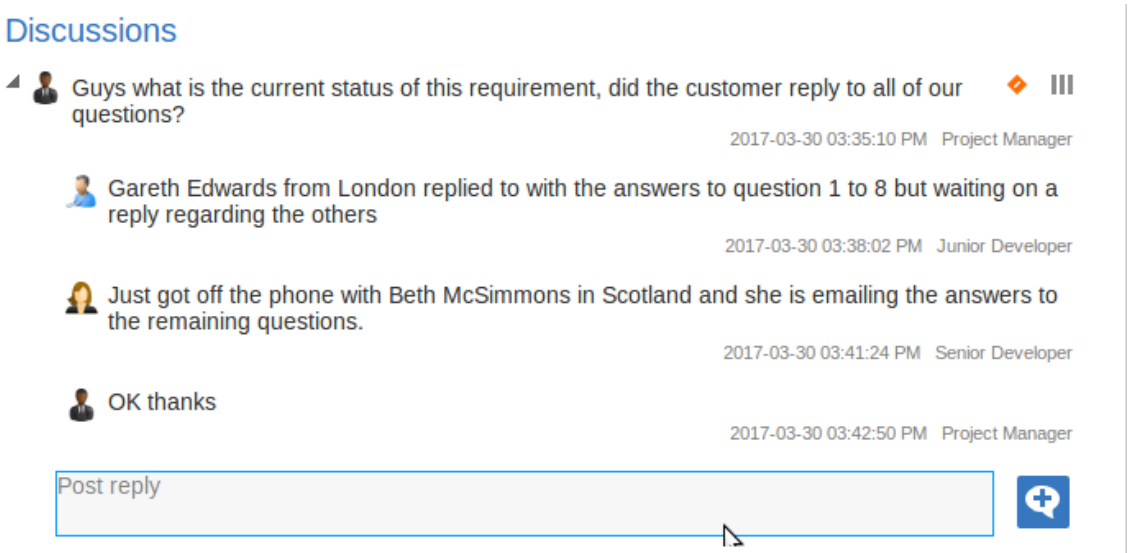


There are two types of discussion message - an initial message (usually called a Discussion) and replies. Replies are responses to an initial message, and are displayed as child items.

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.

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



Note: WebEA does not support editing (or deleting)

Discussions, therefore you should take care to type messages 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.

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 model when the page is loaded, which means if other users make additional comments they will not be displayed until the page is reloaded or refreshed.

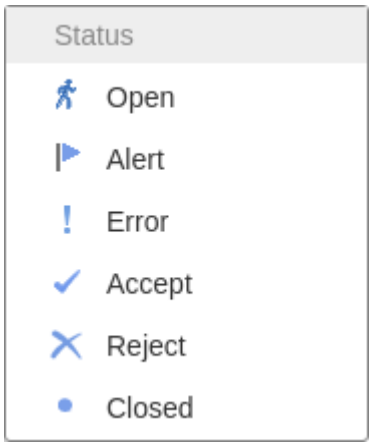
Status and Priority

If you are adding comments to a Discussion, at any point you can return to a posted topic or reply and add or change

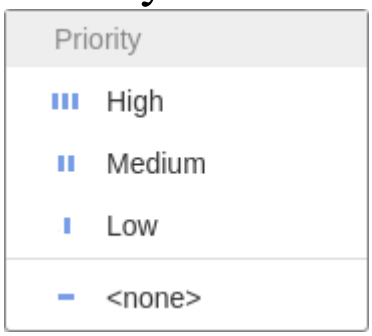
its status and priority.

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



- Priority



Notes

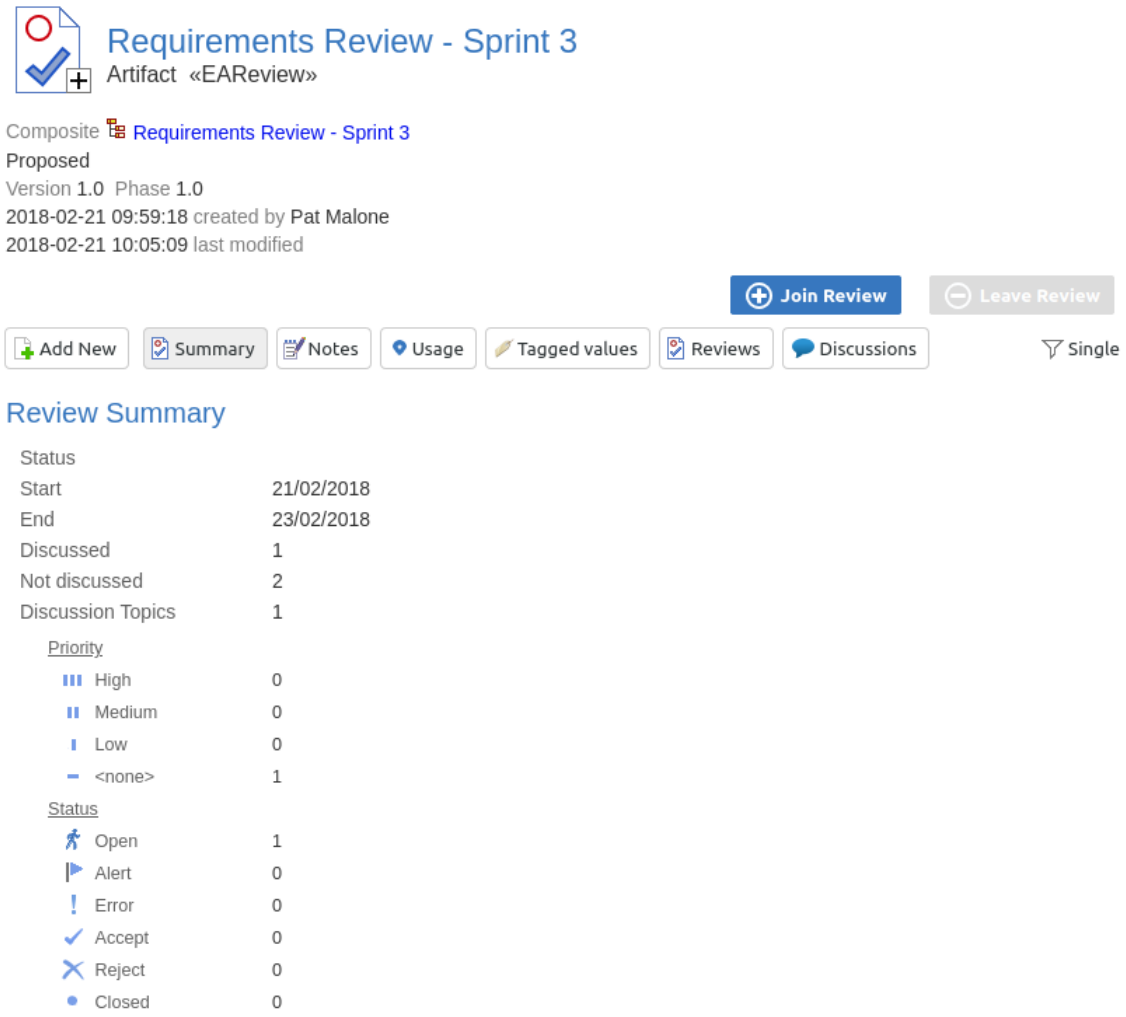
- As a WebEA user you cannot delete Discussions or replies; however they can be deleted using Enterprise Architect

Reviews in WebEA

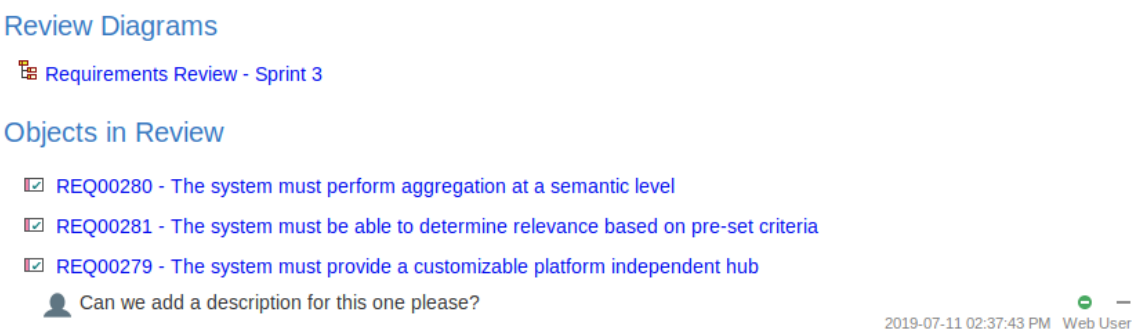
Review elements are a convenient way for groups of users to collaboratively discuss one or more model objects and capture those discussions as a formal review. A Review element is a custom element that captures all discussions that you participate in over a period of time. That period begins when you 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's Main View, the default feature that is displayed is a 'Review Summary'. This includes the status, start and end dates, number of elements discussed and yet to be discussed, and a breakdown of review topics based on priority and status, as illustrated here:



The 'Review Summary' is followed by the 'Review Diagrams' and 'Objects in Review' sections.



The 'Review Diagrams' section provides links to the diagrams that are associated with the review (i.e. those that are children of the Review element).

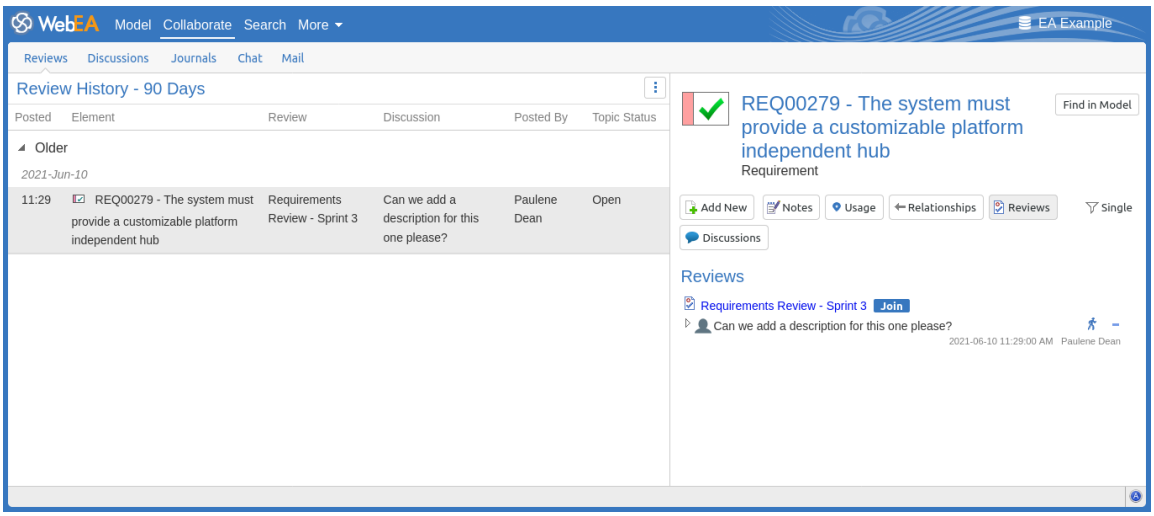
The 'Objects in Review' section displays a list of elements to be discussed during the review. Elements that have not yet

been discussed are presented first in the list, followed by elements that have been commented on, showing each review topic. The name of the object acts as a hyperlink to that object.

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 discussions, click on the Collaboration button in the WebEA header, then select the 'Reviews' tab. This view provides a history of recent Review discussions. Selecting an item will display the Review discussions for that element in the right panel.

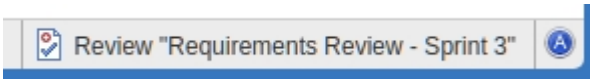


From here you can Join the review and add a topic/reply, or click on the review element name to access the review element's properties.



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.



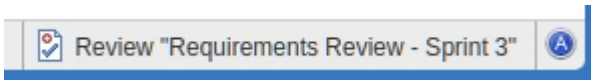
Join a Review

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' section filtering elements to the current Review element only
- The Join Review button being disabled, and the Leave Review button being enabled

The Join Review button might be disabled, for these reasons:

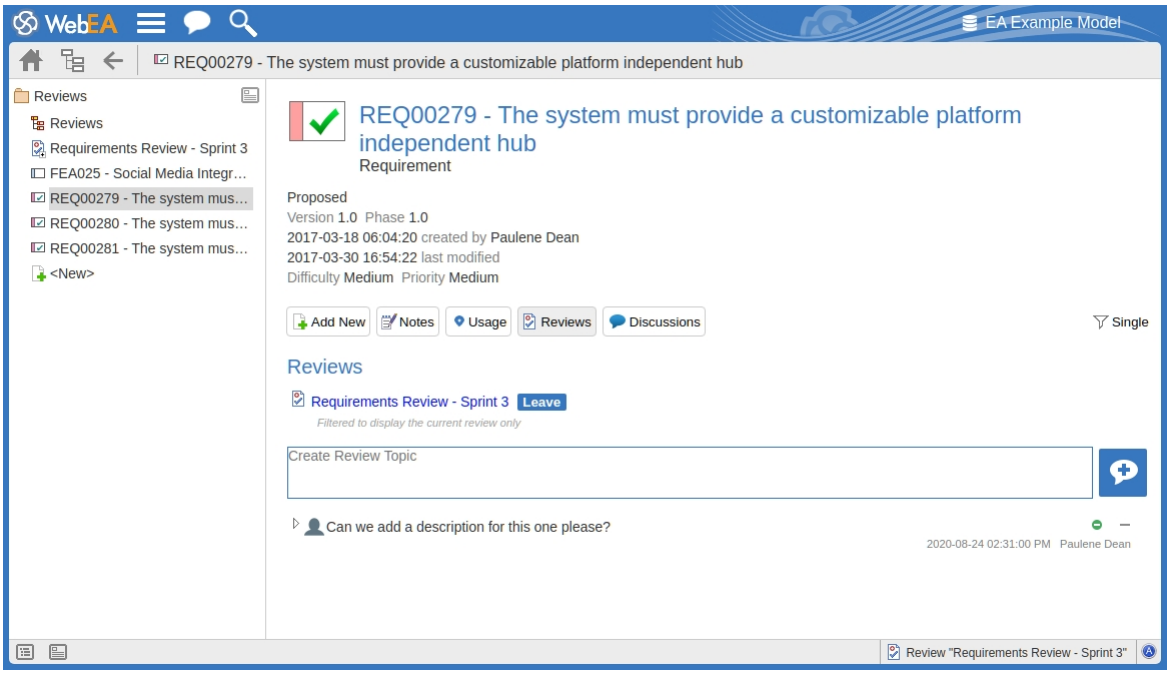
- You are already joined to a review on a different Review element
- The loaded Review is closed

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 object you want to comment on, then select the 'Discussions' feature button.



Note: 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

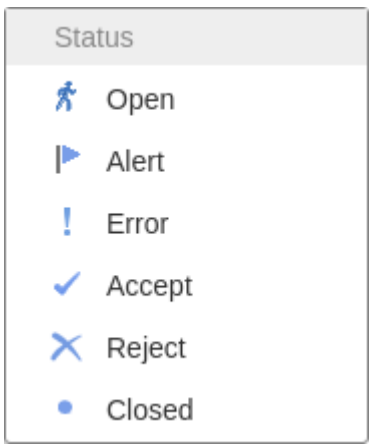
To create a review topic on a new point, simply type into the 'Create Review Topic' field. When you have finished typing your message, submit it by clicking on this button:

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

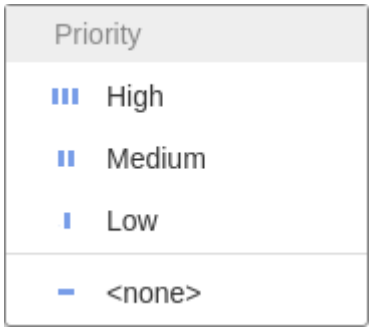
Depending on how the WebEA model is configured, you might connect to it using your personal user ID, or via a single generic User that is shared by multiple people. In the second case, Sparx Systems recommends that each user identifies themselves in each message, either at the beginning or at the end.

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



- Priority



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 model when the page is loaded, which means if other users make additional comments they will not 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.



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 *Object Creation 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.

WebEA - Collaborate

The 'Collaborate' tab is used for accessing WebEA's collaboration features such as Chat, Mail and Discussions. It can be accessed by using the 'Collaborate' button/link in WebEA's main header.



Note: The Collaborate tab is only available in the WebEA header if one or more of the Collaboration features (e.g. Discussions or Journal Items) is enabled in the WebEA configuration.

Collaboration Views

This Collaborate tab consists of these child tabs / views.



Click on these links for more information:

Tab / View	Description
Reviews	Provides a history of recent review discussions.
Discussions	Provides a history of recent discussions.
Journals	Provides a history of recently added/modified Journal entries.

Chat	Provides a list of recent Chats, and provides access to the Chat view (for chatting with other model users and groups).
Mail	Provides access to the mail inbox and outbox, and allows you to send mail to other users or groups.

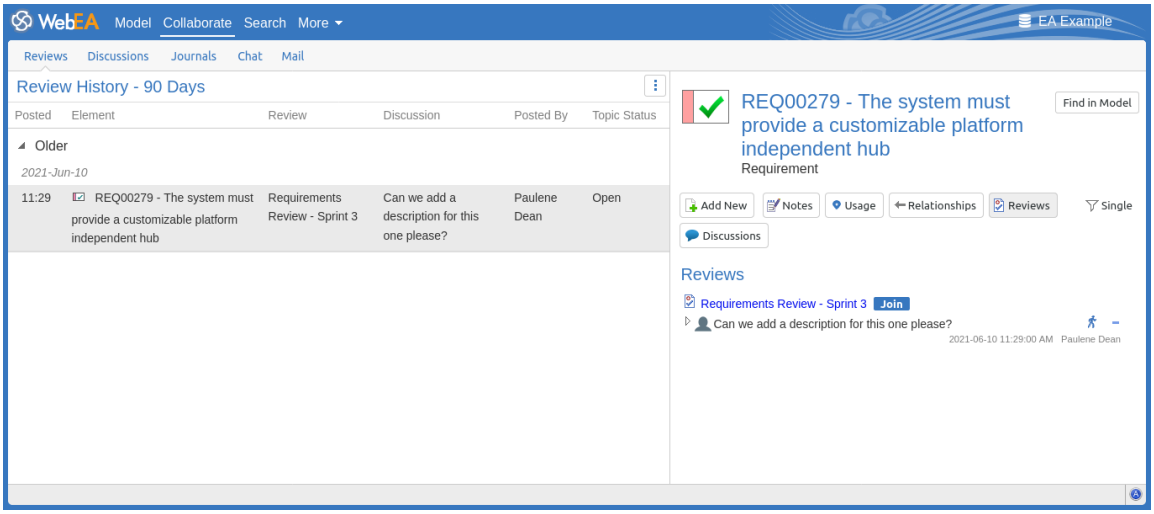
Reviews

A summary of Enterprise Architect's Reviews can be accessed within WebEA, allowing you to quickly view reviews that have occurred recently, all from the convenience of your web browser.

When enabled, the Review History can be accessed by selecting the main 'Collaborate' tab, then the 'Reviews' tab within the collaborate view.

Review Summary

The 'Review' tab of the Discuss & Review window provides these features and options.



Feature	Description
Review History	The left hand side of the Reviews window displays a summary of reviews made within the model, organized by date. The details displayed will included

	the date/time of the review, element's name, review name, the discussion text, who posted the discussion and its status. Selecting an object in the list will display its details in the properties area (on the right side).
Review Timeframe	Click on the Review Timeframe button to select the timeframe for the recent items in the Review History list.
Properties and Reviews	The right hand side of the Reviews window displays the main properties of the selected object along with the reviews that have been added against this object.

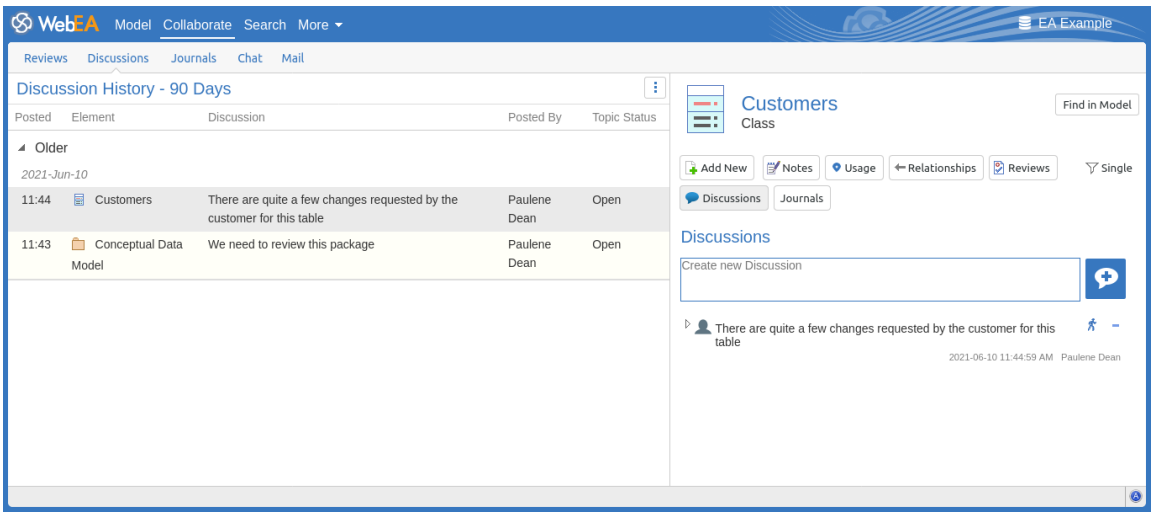
Discussions

A summary of Enterprise Architect's Discussions can be accessed within WebEA, allowing you to quickly view discussions that have occurred recently, all from the convenience of your web browser.

When enabled, the Discussion History can be accessed by selecting the main 'Collaborate' tab, then the 'Discussions' tab within the collaborate view.

Discussion Summary

The 'Discussions' tab of the Discuss & Review window provides these features and options.



Feature	Description
Discussion History	The left hand side of the Discussions window displays a summary of discussions made on elements within the model, organized by date. The details

	displayed will included the date/time of the discussion, element's name, the discussion text, who posted the discussion and its status. The summary does not list discussions concerning diagrams. Selecting an object in the list will display its details in the properties area (on the right side).
Discussion Timeframe	Click on the Discussion Timeframe button to select the timeframe for the recent items in the Discussion History list.
Properties and Discussions	The right hand side of the Discussions window displays the main properties of the selected object along with the discussions that have been added against this object.

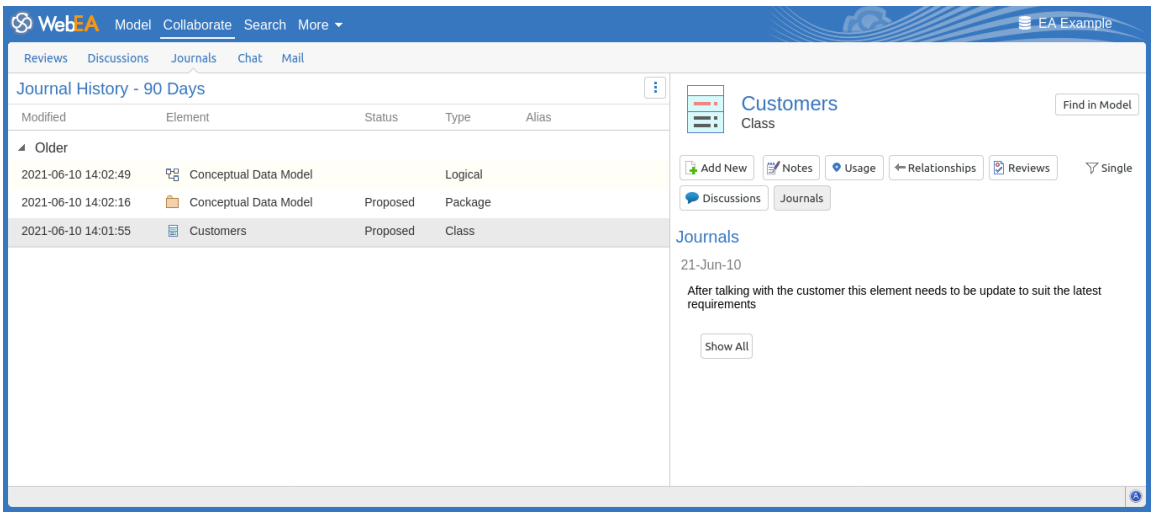
Journals

A summary of Journal entries made against Enterprise Architect's objects can be accessed within WebEA, allowing you to view any ad-hoc comments, notes or reminders, all from the convenience of your web browser.

When enabled, the Journal History can be accessed by selecting the main 'Collaborate' tab, then the 'Journals' tab within the collaborate view.

Journal Features

The 'Journals' tab provides these features and options.



Feature	Description
Journal History	The left hand side of the Journals window displays a summary of journal entries made within the model, organized by date. The details displayed will included the date/time of the journal entry,

	element's name, status, type and alias. Selecting an object in the list will display its details in the properties area (on the right hand side).
Journal Timeframe	Click on the Journal Timeframe button to select the timeframe for the recent items in the Comment History list.
Properties and Journals	The right hand side of the Journals window displays the main properties of the selected object along with the journal entries that have been added against this object.

Chat

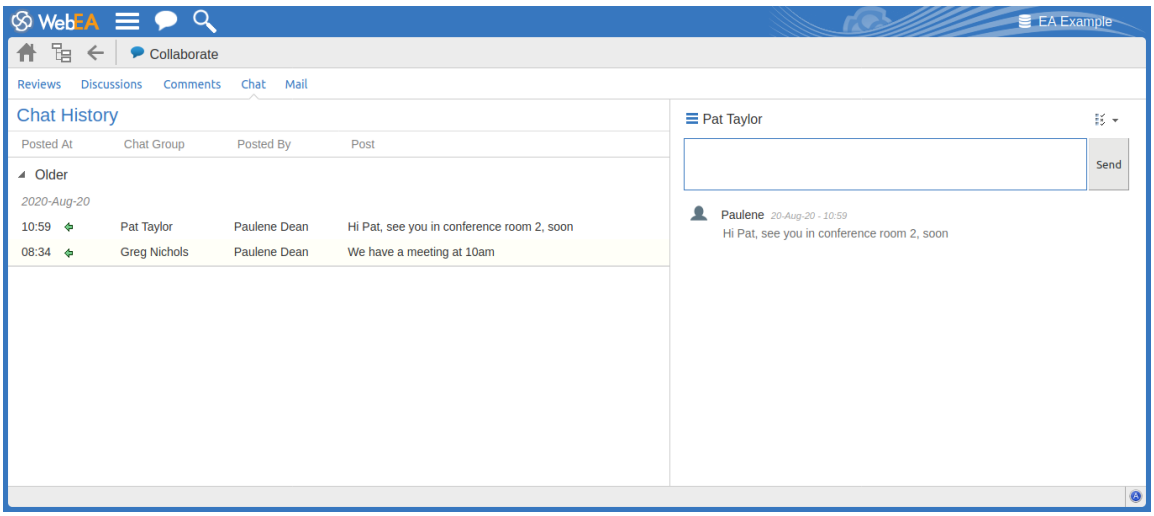
Enterprise Architect's internal Model Mail system can be accessed within WebEA, allowing you to view and send instant chat messages to other model users and groups from your web browser.

When enabled, the Chat view can be accessed by selecting the main 'Collaborate' tab, then the 'Chat' tab within the collaborate view.

Note, to use Model Chat the model must have User Security enabled, and the WebEA configuration should have the 'Show Model Chat' option set to 'Yes'. This is set to 'No' by default. See the How to configure WebEA models topic for details.

Model Chat Features

The 'Chat' tab of the Chat & Mail window provides these features and options.



Chat History	On the left hand side of the Chat & Mail window is a summary of the latest chats to all people (or groups) that you have made organized by date. Simply select any person (or group) in the summary to make them the active chat session.
Chat Messages	The right hand side of the Chat window displays the currently selected user or group that you are 'chatting' with, along with the previous chat messages that have been sent between you and the other person/group. The small hamburger icon will display a drop-down of available users and groups that you can chat with. Selecting one will refresh the lower section with the previous chat messages. Recent users are displayed at the top of the drop-down and the two lower options present a submenu of all available Users and Groups. The options icon will display a drop-down of available options that will control how many days of chat messages will be displayed. Note: keeping this number low will improve performance.
Send	You can only be actively chatting to one person or group at a time, simply select

	the person (or group), type in your chat message and click the Send button

Mail

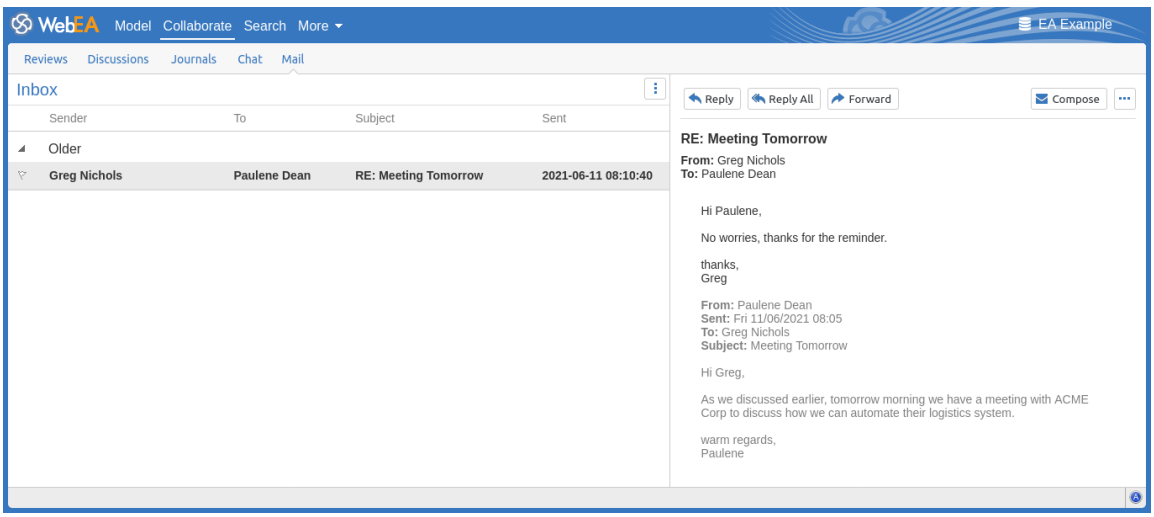
Enterprise Architect's internal Model Mail system can be accessed within WebEA, allowing you to view and send messages from your web browser.

When enabled, the Mail view can be accessed by selecting the main 'Collaborate' tab, then the 'Mail' tab within the collaborate view.

Note, to use Model Mail the model must have User Security enabled, and the WebEA configuration should have the 'Show Model Mail' option set to 'Yes'. This is set to 'No' by default. See the How to configure WebEA models topic for details.

Model Mail Features

The 'Mail' tab provides these features and options.



Feature	Desciption
Inbox / Sent	Displays the list of Messages in your

Mail list	Inbox or Sent Mail. Selecting an item from the list will display the Message contents in the panel to the left.
Message View	Located to the left of the Inbox/Sent items list, this panel displays the contents of the selected message. See the Viewing Messages topic below for details.
Mail Options	Click on the Mail Options button to display the Mail Options menu. This menu provides the options to **Compose** a new message, or switch between the **Inbox** and **Sent Items**

Viewing Messages

When selecting a message in the Mail Inbox list, the message's contents are displayed in the panel to the right. Within this panel, the options discussed here are available.

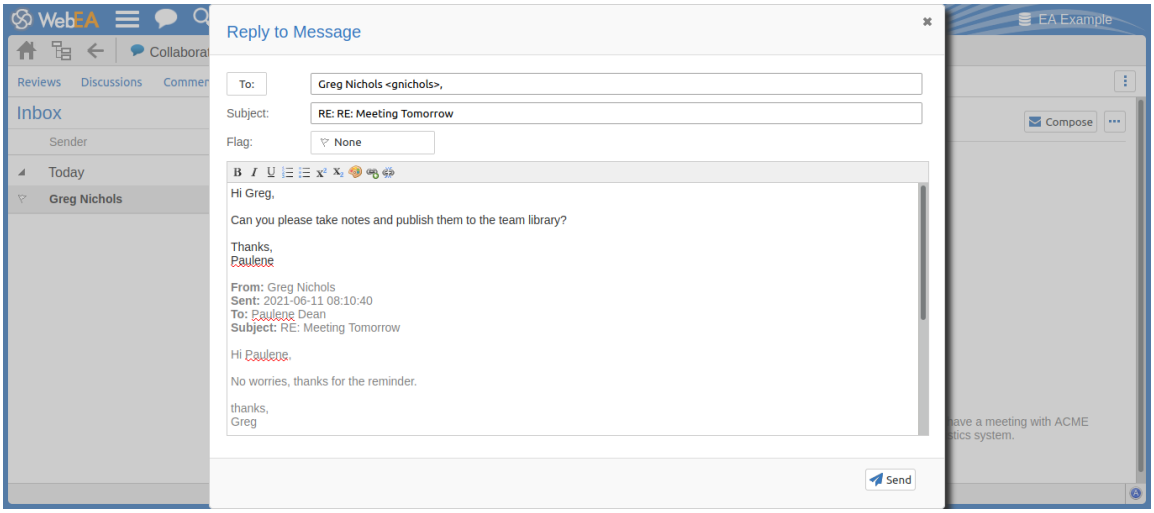
Note: selecting a Sent Mail item will also display the message contents, but in this case only the Forward option is available.

Option	Description

Reply	Displays the 'Reply to Message' dialog. The 'To' field will be populated with the User Name of the Sender from the message which you are replying to.
Reply All	Displays the 'Reply to Message' dialog. The 'To' field will be populated with the User Name of the Sender and Recipients from the message which you are replying to.
Forward	Displays the 'Forward Message' dialog.
Set Message State / Flag	Click on this button to set the Message state (either Read or Unread) or Set a Flag (None, Complete, Red, etc).

Composing Messages

Model Mail messages can be composed via the 'New Message', 'Reply to Message', and 'Forward Message' dialogs. These dialogs are accessed via the Compose button, Reply/Reply All buttons and Forward button respectively.



The dialogs all provide these fields and functions.

Feature	Description
To	The To field should contain the name and username of the intended recipients (as defined in Enterprise Architect's user security settings). To add a recipient, click on the To button/label and select the user from the list.
Subject	Enter the subject for the message.
Flag	Set to None by default. To change the Flag for the message click on the Flag field/button and select an item from the list.
Message Content	Enter the message contents. The toolbar at the top of this text area provides the

	same formatting options and features as WebEA's Edit Note fields. See the Edit Objects topic for details.
Send	Click on the Send button to sent the message.

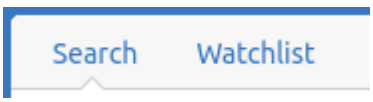
WebEA - Search

The 'Search' tab is used for accessing WebEA's two search functions; the model search, and watchlists. It can be accessed by using the Search button/link in WebEA's main header.



Search Views

The main 'Search' tab consists of two child tabs / views.



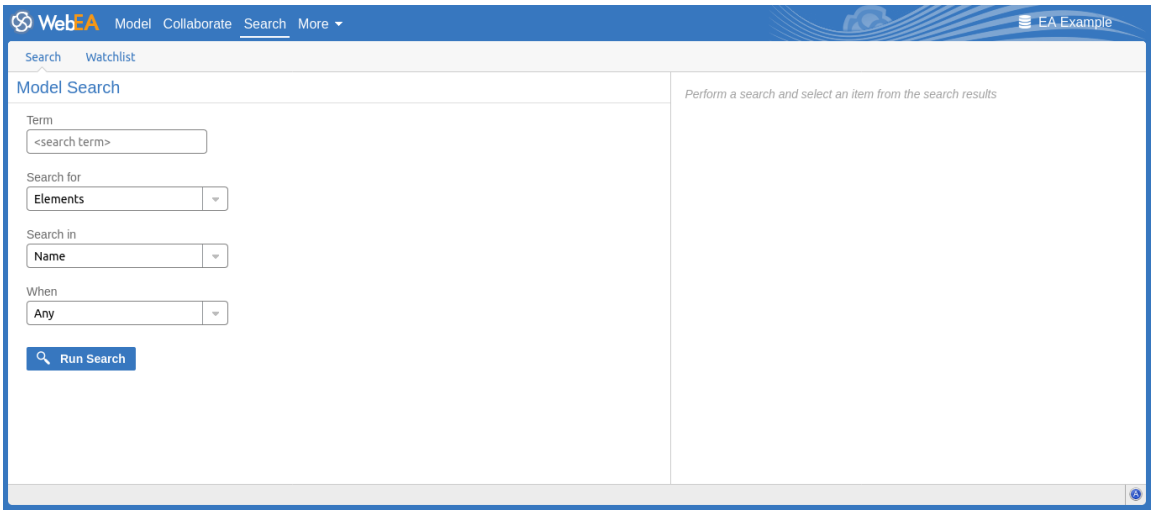
Refer to the links below for more information:

Tab / View	Description
Search	Allows you to perform custom searches on your model, specifying a search term, selecting which fields to search in, etc.
Watchlist	Provides a configurable list of predefined 'Watchlist' searches. E.g. Recently modified elements, or recent discussions.

Search

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).

The search can be accessed by selecting the main 'Search' tab, then the 'Search' tab within the search view.



Search Specification

Define your search using these options.

Option	Description
Term	Type or paste in 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.

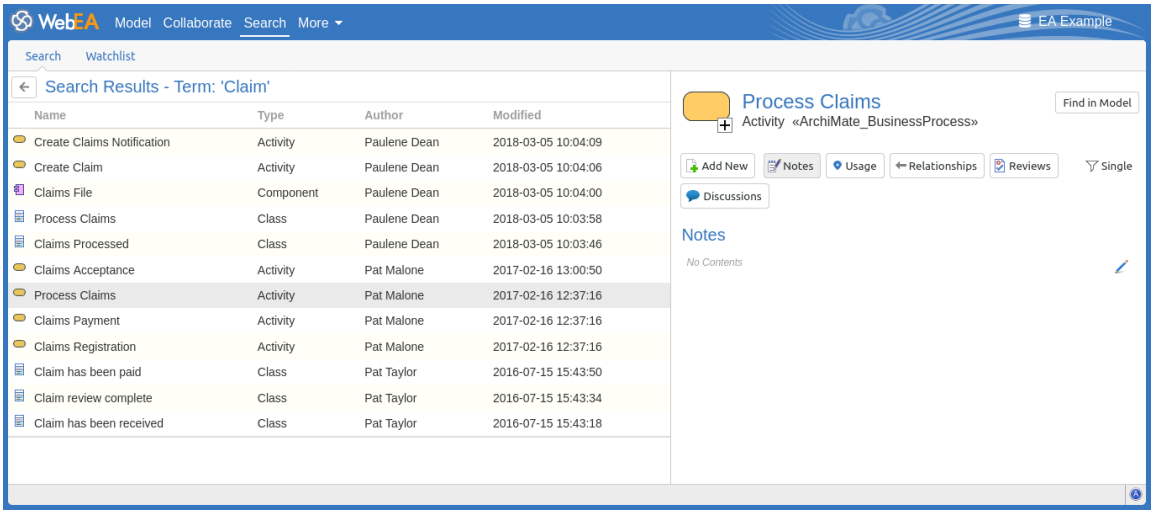
Search for	Select one of these object types to search: • Diagrams • Elements • Packages • Discussions, or • Reviews If you select 'Discussions', the 'Search in' and 'Term' options are not relevant and so are grayed out and unavailable. The search is for elements having active discussions within the specified time period. Note: the 'Discussions' option is only available in models that have the *show_discuss* parameter set to "true" in the WebEA configuration file. Similarly, the 'Reviews' option is only available when the *participate_in_reviews* parameter is set to "true" in the configuration file.
Search in	Select from the 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	Select 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 on the left side of the search view. 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, then its

properties will be displayed in the panel to the right.



Within the properties panel you can use the 'Find in Model' button (shown below) to locate the object in your model (I.e. switching to 'Model' tab).



If you would like to perform a new search click on the back arrow (shown below) in the Search Results header to return to the Search input fields.



Watchlist

The Watchlist facility in WebEA helps you to quickly identify recent activity in the model. Each user can tailor their Watchlist to monitor activity in any of a wide range of objects and features with various characteristics. Whenever the Watchlist Summary is generated, all object types of interest 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.

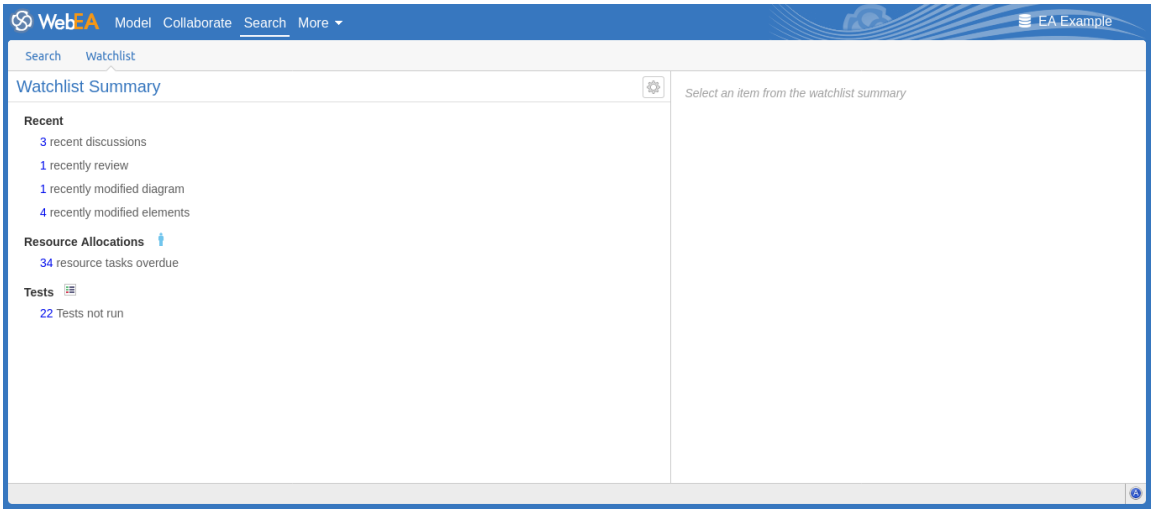
Note: Each model has a default set of activity types that have been enabled in the WebEA configuration file; each user can override the model defaults and save their customized settings.

Watchlist Summary

The watchlist can be accessed by selecting the main 'Search' tab, then the 'Watchlist' tab within the search view. The left side of the watchlist shows:

- Any activity that has recently been detected, for
- All objects and features that are to be monitored

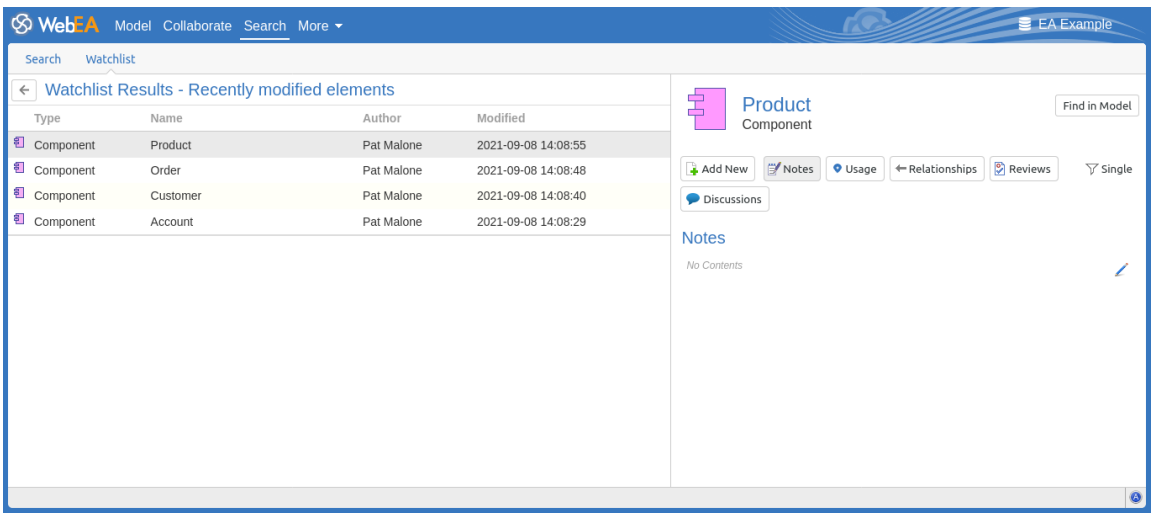
If there is no activity for a configured activity type, that activity type is omitted from the 'Watchlist Summary'. The display might look like this:



Every type of object on the report is represented by a line providing the number of objects that match the given criteria.

Watchlist Results

Clicking on an item in the Watchlist Summary list will display the Watchlist results for that item (I.e. a list of all items matching the criteria). Selecting an object from the results list will display its properties in the panel to the right.



Within the properties panel you can use the 'Find in Model' button (shown below) to locate the object in your model (I.e.

switching to 'Model' tab).

Find in Model

If you would like to return to the Watchlist Summary click on the back arrow (shown below) in the Watchlist Results header to return to the Search input fields.

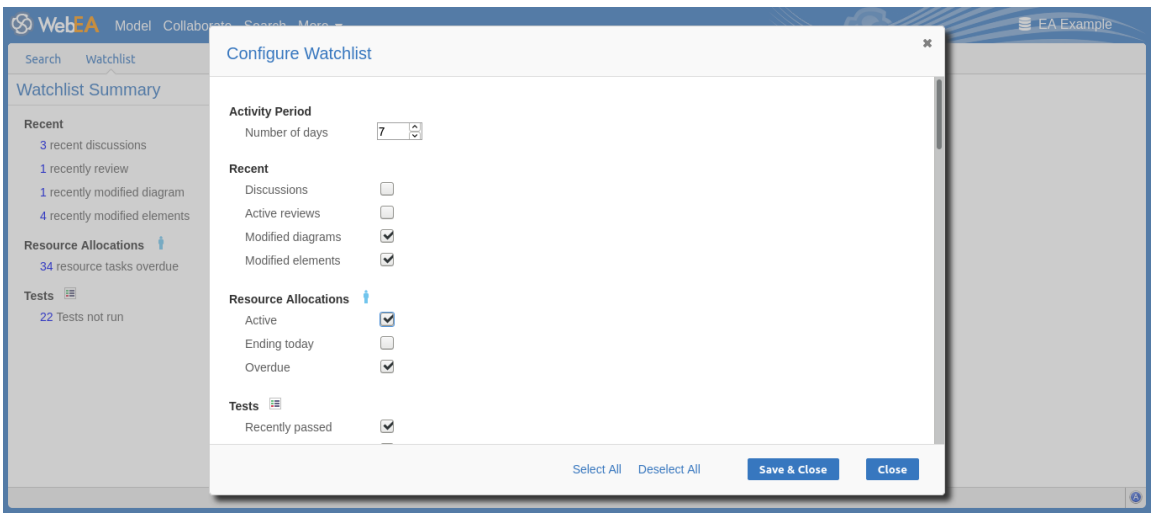


Configure the Watchlist

If you want to customize the activity types monitored in your Watchlist summary, click on the configure button located in the Watchlist summary header (shown below).



The 'Configure Watchlist' 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 Watchlist. 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:
  - Verified
  - Recently requested
  - Recently completed
  - New
  - Incomplete
- Events:
  - Recently requested
  - High Priority
  - Incomplete

The 'Select All' and 'Deselect All' options can be used to

select or deselect every checkbox.

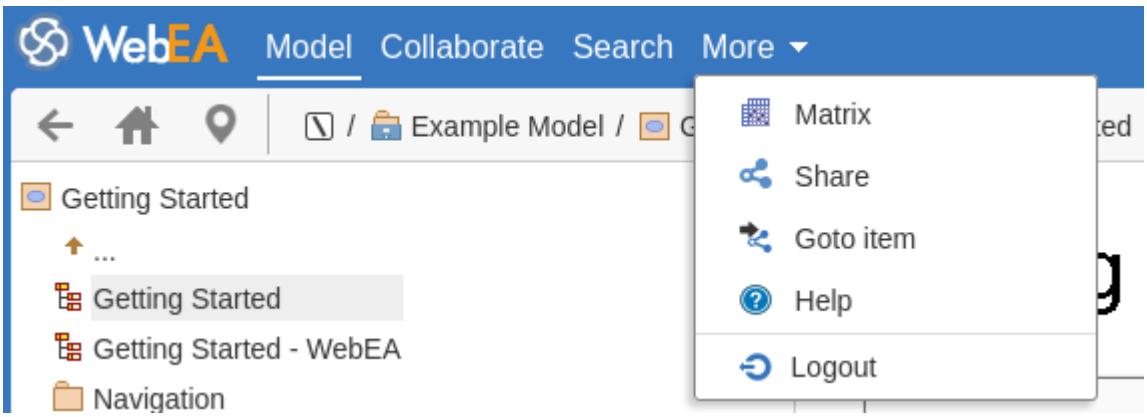
Once you have finished choosing which items to monitor, click on the Save & Close button to save the configuration and refresh the Watchlist summary. Alternatively you can use the Close button to discard any changes.

Note:

- All Watchlist 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 Watchlist setting will only last as long as the current session

WebEA's Main Menu

WebEA's main menu is accessed by clicking on the 'More' button in WebEA's header (shown below).



Menu Options

The 'More' dropdown menu provides these options.

Option	Description
Matrix	Select to display the relationship matrix. See the Relationship Matrix topic for details.
Share	Select to display the 'Link to WebEA item' dialog, which shows the current object's unique ID (GUID) and full URL. You can copy this ID, then at a later point use it in the 'Goto WebEA item' dialog to return directly to that page (see below). Alternatively, you can copy the full URL and send it to another user if you want

	them look at a particular item. Note: This option is only supported when displaying a specific object (diagram, element or Package). It is not supported for the Watchlist, Matrix, or similar facilities.
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 'Share' option.
Help 	Select this option to open WebEA's online help.
Logout 	Choose this option if you want to disconnect from the current model, returning to the 'Select Model' screen.

Relationship Matrix

The Matrix facility in WebEA enables you to view the Relationship Matrix generated from any Relationship Matrix Profile that has been defined in the model.

The Relationship Matrix is a spreadsheet display of relationships between model elements within a Package, or between elements in two different Packages. It is a convenient and simple tool for reporting and working on all the relationships in a selected structure. The relationships shown can be filtered according to:

- The element type of the source and target elements in each relationship
- The relationship type and direction
- The Package(s) in which the source elements and target elements are held

Those parameters can be defined in a Relationship Matrix profile; to display a matrix in WebEA you select one of the available profiles.

The matrix shows all the relationships of the specified type between source and target elements by:

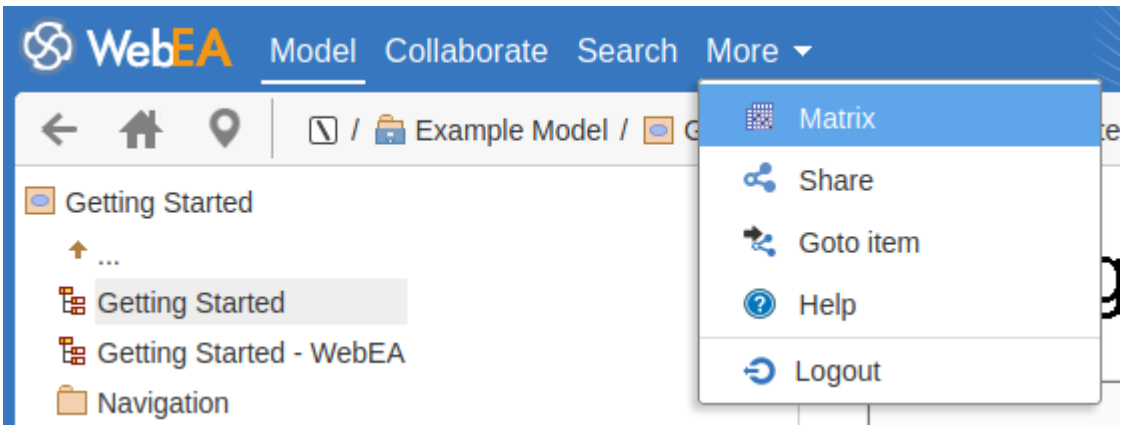
- Listing the source Package elements down the side of the matrix, as row titles
- Listing the target Package elements across the top of the matrix, as column titles, and
- If a relationship exists between a source and target element, displaying an arrow indicating the direction of

the relationship, at the intersection of the appropriate row and column

Clicking on a square containing an arrow displays the 'Relationship Properties' screen for the relationship represented by that arrow. On that screen you can click on the source or target element names and display the 'Element Properties' screen for those elements.

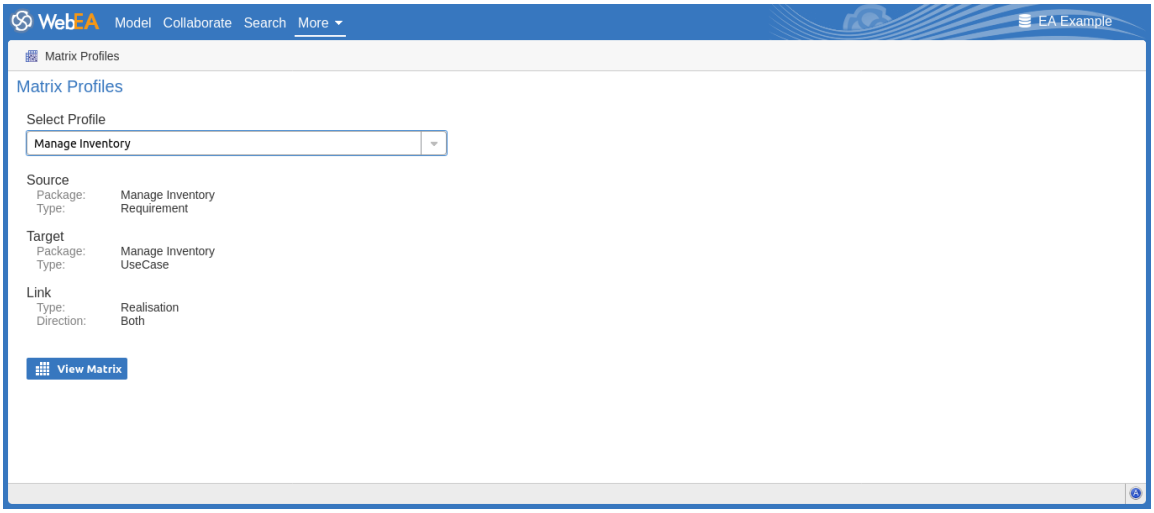
Access

In WebEA's main header, click on the 'More' button, then 'Matrix'.



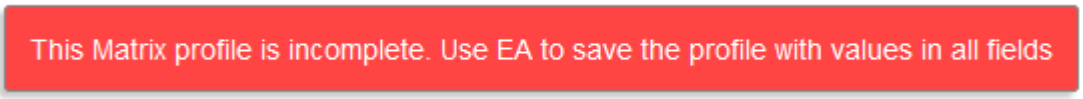
Matrix Profiles

The 'Matrix Profiles' page provides the 'Select Profile' drop-down field that you use to list and select from the matrix profiles defined within Enterprise Architect. You cannot add to or edit these profiles.



When you select a matrix profile, the rest of the fields on the page will be populated with the values defined in the profile. Click on the View Matrix button to load the relationships selected under the profile, in a Relationship Matrix grid.

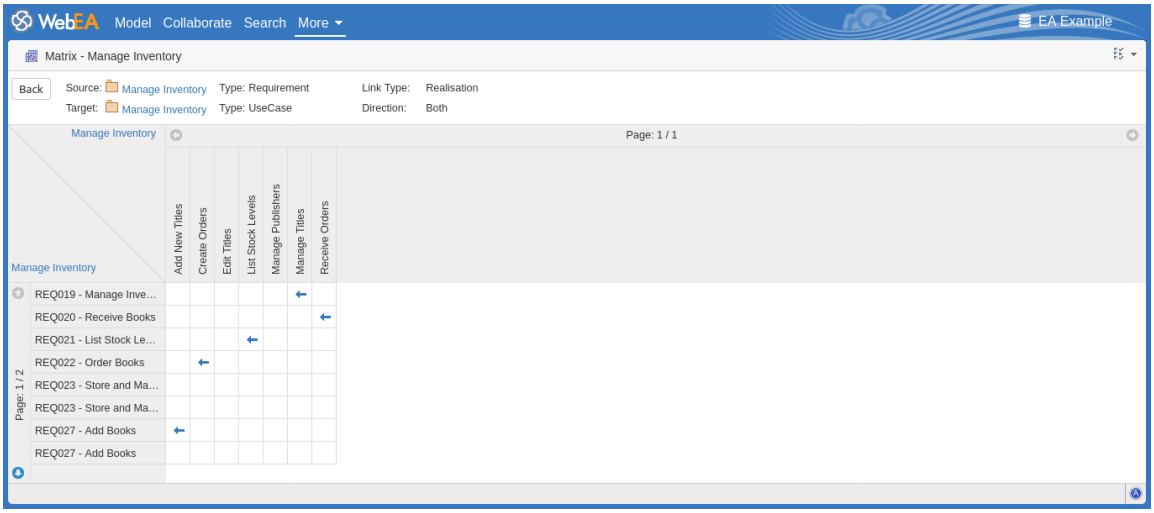
WebEA is only able to load complete matrix profiles; if an incomplete matrix profile is requested, this error message will be displayed:



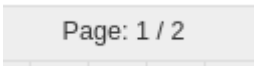
If this message is displayed, ask the Project Manager to update the Matrix Profile.

Matrix

This is a sample Relationship Matrix as displayed by WebEA. It is almost identical to what would be displayed in Enterprise Architect.



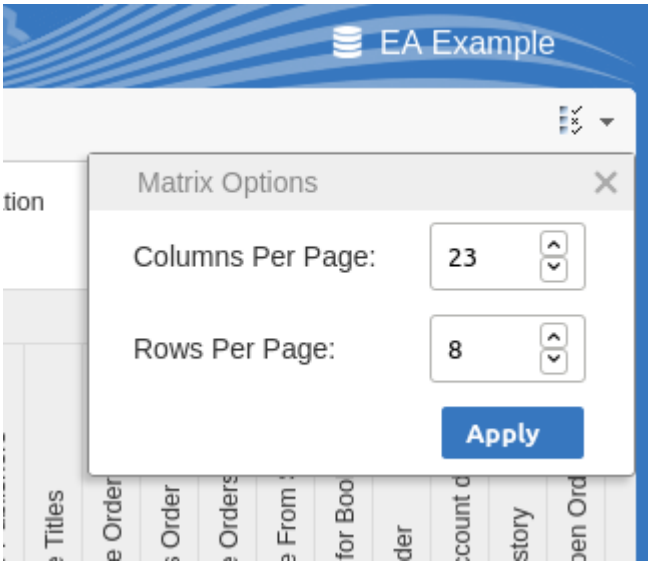
If the matrix content (rows/columns) will not fit onto a single screen then the matrix is automatically divided into multiple pages.



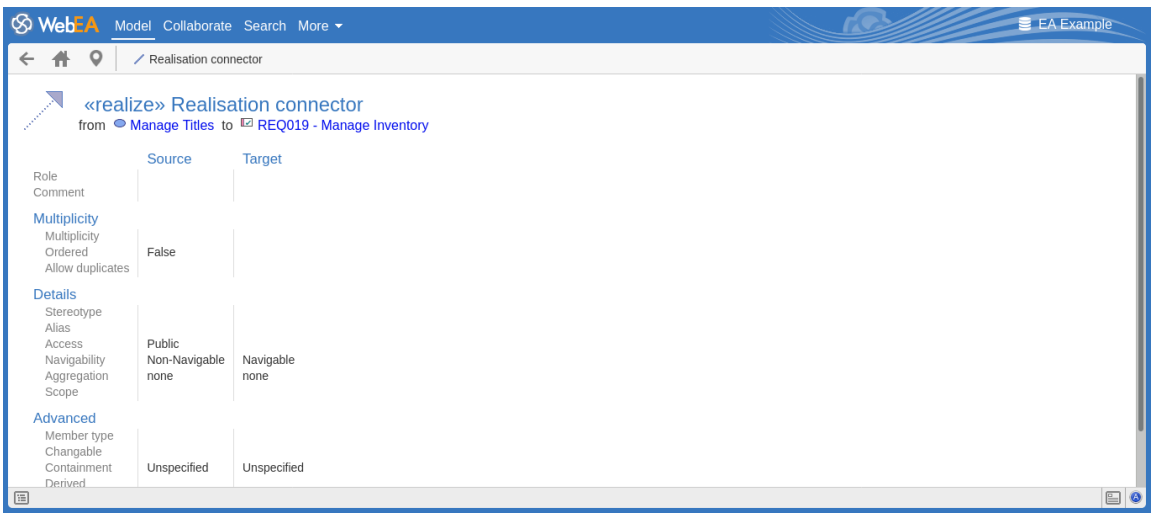
You can access the previous/next page using the arrows in the source/target headers.



If necessary you can also increase number of columns/rows which are displayed per page by using the Matrix Options button. Note, the number of cells per page will have an impact on the web browser performance, so avoid using pages with a large number of cells (e.g. more than 200 x 200).



When you click on a relationship arrow within the matrix - such as, in the example, the 'Manage Titles' to 'REQ019 - Manage Inventory' arrow - the 'Relationship Properties' screen displays.



Notes

- A Matrix Profile can also be accessed via a Hyperlink or a Matrix Specification Artifact on a diagram
- Matrix Profiles that use a Model Search to populate the source or target elements are not supported in WebEA

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 Configuration Test

On initial set up, these steps can be performed to help identify configuration issues:

Step	Action
1	On the Web Server machine, open a web browser and test access to the WebEA login page. For example, type: http://localhost/webea/login.php If this fails, check that the Web Server has been set up as described in the *WebEA Installation and Configuration* Help topic.
2	On a client machine test access to the WebEA login page. For example, type: http://<server name>/webea/login.php If this fails, there is a network/communication issue between the client and the web server.

3	Before attempting to access a model via WebEA, confirm you can successfully open the model using Enterprise Architect's 'Connect to Cloud' option. If this fails, refer to the *Pro Cloud Server Troubleshooting* Help topic.
4	In the webea_config.ini file, add a model connection with the minimum required options, as outlined here. Ensure these match the settings used when connecting via Enterprise Architect. Note, the 'sscs_db_alias' should match the value entered in the 'Model Name' field. sscs_protocol = "<Protocol>" sscs_server = "<Server Name or IP address>" sscs_port = "<Port>" sscs_db_alias = "<Model Name>" auth_code = "" ;leave this empty If the Project has Enterprise Architect User Security enabled then also include: login_prompt = "true" If the Protocol being used is HTTPS then also include: sscs_use_ssl = "true"

	If you are using HTTPS with a Self-signed certificate then also include: sscs_enforce_certs = "false"
5	Try accessing the model in WebEA. If this fails, the error returned by the WebEA login screen might help identify the cause of the issue.

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 Server is running!*

	Possible reasons: • Pro Cloud Server is not installed • Pro Cloud Server is not running • Pro Cloud Server is not listening on the expected Port • The Pro Cloud Server has IP addresses defined in the Client Whitelist but the WebEA server IP is not included; see the *Access Control* Help topic for details
OSLC not supported	*The server port is not configured to support OSLC.* Possible reasons: • WebEA communicates with Sparx Systems Pro Cloud Server via OSLC, but the defined Sparx Systems Pro Cloud Server has been configured to not allow OSLC requests for this Port Using the Pro Cloud Server Configuration Client, check the configuration for the Port that you are using to access this model. Ensure the 'OSLC Supported' option is enabled.
Protocol Mismatch	*Protocol and Port mismatch. Check 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
Connection timed out	*Request Error: Connection timed out after 5001 milliseconds.* Possible reasons: • Invalid Pro Cloud Server settings (sscs_*), WebEA cannot connect to the defined server/Port • Apache/PHP web server requires a proxy server to reach the Pro Cloud Server machine and the proxy settings defined in the .config file are incorrect or 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 Server configuration file (SSProCloud.config); make them the same

	Note that, if the Pro Cloud Server is being re-directed via IIS, you must ensure this matches the IIS Port used for the Pro Cloud Server.
The selected model is not defined	*The selected model is not defined in the Pro Cloud Server.* 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 • The SSProCloud.config file has a DEFAULT_MODEL defined; we recommend removing this line from the SSProCloud.config file - see the *Add a Port Definition* Help topic for details
Model is Shutdown	*The selected model is shutdown.* Possible reasons: • The Cloud Server Model connection that you are attempting to access is not enabled Using the Pro Cloud Server Configuration Client, check that the Database Manager is set to 'Enabled'.

Secure connection is required	*A secure connection is required for the selected model.* This message indicates that the 'Require a secure and authenticated connection' option is enabled for this Model Connection/Database Manager. To access this model via WebEA you must use the HTTPS protocol. You must also use a Port that has 'Require SSL' enabled, and has the 'Authentication' option set to either 'Model' or 'Global'.
SSL Error	*Error: SSL certificate problem: unable to get local issuer certificate* This error suggests you using an SSL certificate which was signed with your own Certificate Authority (CA), however PHP/cURL have not been configured to trust the CA. Update your PHP/cURL configuration to trust your CA. See the *Configuring WebEA to Trust Your Own Certificate Authority* Help topic for details.
Self signed certificate error	*Request Error: SSL certificate problem: self signed certificate* This error suggests you are attempting an HTTPS connection with SSL using a

	self-signed certificate; however, WebEA has not been configured to allow self-signed certificates. Possible Reasons: • The webea_config.ini file does not include 'sscs_enforce_certs = "false"' for this model; try adding this, then attempt to access the model again
Unable to access secure session cookie	*Unable to access secure session cookie* This error indicates that PHP is configured to use secure cookies only (session.cookie_secure = true). Check the URL you are using to access WebEA. In this case the WebEA site should be accessed via HTTPS.
Model has not been specified	This error can occur if WebEA is unable to set or read PHP session variables. Check you PHP configuration and confirm that session variables are supported.
Missing User Authenticatio n Identifier	*40x - Missing User Authentication Identifier* There was a problem authenticating the user in the repository. Possible Reasons:

	• Incorrect user authentication configuration in the repository • Using an old or mismatched version of WebEA with Pro Cloud Server
No credentials provided	*400 - User Security is enabled, however no credentials were provided.* Possible Reasons: • The Project has Enterprise Architect user security enabled; however, WebEA is not configured to either show a login prompt or automatically login with specific credentials Check that the webea_config.ini contains: login_prompt = "true" Or, all of these: login_prompt = "false" sscs_model_user = "<user id>" sscs_model_pwd = "<password>"
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
Denied user	*Pro Cloud Server denied user login:*

login: User Limit Reached	*WebEA user Limit Reached or* *Pro Cloud Server denied user login: Prolaborate user Limit Reached or* *Pro Cloud Server denied user login: 3rd Party user Limit Reached* When using a ProCloud Server with token license(s) once all tokens have been consumed, any further user request to logon will receive the 'user limit reached' error message. Possible Reasons: • There are more users than available Pro Cloud Server Token Licenses. • Users that have previously logged in have not logged out and their sessions are yet to expire, so their token(s) are consumed but not currently used. • A single physical user has multiple sessions (multiple browsers) open, each consuming a token. Note: due to the stateless natural of web based requests, once a user 'logs in' a token is allocated for 30 minutes or until the user logs out. When using token licensing it is important to stress to your user base that they need to log out of WebEA once they are finished so other users are not

	impacted by a lack of licenses.
Incorrect Access Code	*The entered access code is incorrect!* Possible reasons: • This model has been configured (in the webea_config.ini file) to require an access code for WebEA access; the access code that was entered is incorrect Check the webea_config.ini file for the correct access code / "auth_code".
Problem reading the model root	*Problem reading the model root.* Possible reasons: • The WebEA options for the current model do not match a Pro Cloud Server configuration • You are attempting to access a DBMS repository that has not had a Project transferred into it; see Step 4 in the *Server Based Repositories* Help topic
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: • Your 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 Pro Cloud Server configuration!* This error normally occurs while attempting to log in or connect to a model from the initial page. The error message will display whenever the response from the configured Pro Cloud Server does not return the expected XML message. Check that the defined Sparx Systems Pro Cloud Server details, such as server name and Port, are correct and not those of a web server.
Diagram image not found	*Diagram image not found.* This error indicates that the diagram exists in the model, but the image of the diagram has not been generated. Possible reasons: • The model does not have the 'Auto create Diagram Image and Image Map'

	option enabled in model-specific options. See the *Cloud Page* Help topic. • The diagram was created in Enterprise Architect, but has not yet been saved • The user is on an old version (prior to build 1350) of Enterprise Architect and the Worker application is not running for this model (it has not been configured, it is not currently running, or has not run since the diagram was saved) Open the model in Enterprise Architect (build 1350 or later), enabled the 'Data Cache' options, and click on the 'Create' button as described in the 'Model Configuration' section of the 'WebEA Installation and Configuration' topic. Alternatively, ensure that the Worker application is running for this model. See the *How to configure automatic viewable components* Help topic for details.
The selected database does not have Pro Features	*The selected database does not have Pro Features enabled.* This error normally occurs if the selected

enabled	model does not have 'Pro Features' enabled. Possible reasons: • In the Pro Cloud Server Configuration Client, the Database Manager for this model does not have the 'Enable Pro Features (OSLC, WebEA and Integration)' option enabled • The Pro Cloud Server's license has expired and the server is now operating as the free, core edition of the Pro Cloud Server • The License key entered is invalid and the Pro Cloud Server is operating as the core edition • The database manager has an 'OSLC Access Code' assigned, but this has not been provided in the webea_config.ini file Using the Sparx Systems Pro Cloud Server Configuration Client, check the Database Manager configuration. Ensure that the 'Enable Pro Features (OSLC, WebEA and Integration)' option is enabled. If an 'OSLC Access Code' has been assigned, ensure the 'sscs_access_code' option has been set in the

	webea_config.ini file.
Resource not found	*404 - Resource with identifier '{XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}' could not be found* The item (element, Package or diagram) with this GUID could not be found in the current model. Possible Reasons: • The item has been deleted from the model To confirm that the item has been deleted, check the parent Package (or parent element) in WebEA. If the error occurred when selecting an item on a diagram, the diagram image might have to be regenerated.
Version Mismatch	*Warning - WebEA / OSLC Version mismatch* Each release of the Pro Cloud Server includes an OSLC component and the WebEA component (php files). If you are using the OSLC component from one release and the WebEA component from another, then WebEA might not function as intended. Possible Reasons:

	• A new version of the Pro Cloud Server was installed; however, the new version of WebEA was not copied to the Web Server When updating the Pro Cloud Server to a newer version, ensure that WebEA is also updated. That is, after performing the installation of the Pro Cloud Server, copy the new WebEA files from the Pro Cloud Server installation to the relevant location on your web server.
Basic login fields not shown	The basic login fields will not be visible when attempting to log in to a model under these circumstances: • The webea_config.ini has 'sscs_model_user' defined • The model has the user option 'Restrict access to Windows & OpenID users only' enabled
Failed to start PHP session	*Error: Failed to start PHP session* WebEA makes use of PHP Sessions. This error indicates that PHP is not able to write to the session 'save_path'. This could be because the path does not exist, or the user which runs PHP does not have write permissions for the folder. This error message also includes

	information regarding the save path location, whether the path is writable, and the user name/id which PHP is running as. Ensure that the folder exists, and that the PHP user has write permissions. If necessary the session.save_path setting can be changed via the php.ini file (then restarting your web server).

Problems loading pages

Issue	Comments
No response when clicking on diagram elements	When clicking on an element on a diagram nothing happens. Possible reason: • The element properties are being loaded in the Properties View; however, the Properties View is automatically hidden due to a narrow screen/browser width (less than 800px) From the hamburger menu at the top right of WebEA, select 'Properties' to turn off the Properties View.
Pages don't	After the server has been updated to a new version of WebEA, clients

load correctly	experience problems with pages loading incorrectly. Possible reason: • 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

Prolaborate

The Enterprise Architect set of tools to deliver world-class modeling and design solutions to business, industry and companies around the world has been significantly expanded with the inclusion of Prolaborate in the Pro Cloud Server offering.

Whilst Enterprise Architect has become the pre-eminent modeling and design solution globally for modelers and designers wanting to visualize and understand complex systems, software and architecture, Prolaborate is the next step in delivering the richness of those models and designs to the broader and greatly expanded consumer community, enabling high-level executives, business process managers, process implementers, consultants, customers and other stakeholders to see their personal and curated view of the model in a way that they are personally comfortable with!

Prolaborate is a tailored set of views that reduce complexity, focus attention and increase the accessibility of model information for non-modelers and others who are more concerned with consuming the model than directly editing it (although this can occur). Prolaborate provides a social platform 'outside the model' that supports talking about the model from a different viewpoint to those actively engaged 'within the model'. Prolaborate also provides dashboards, impact analysis and other 'outside the model' features that derive information from the model or provide unique windows into the model for a custom audience, such as Chief Executive Officers, Chief Information Officers and

customers.

Benefits:

- Takes the next step beyond modeling
- A social and collaborative tool for everyone
- Highly focused and customized model views
- New ways of seeing and interacting with Enterprise Architect models
- New tools for decision and impact analysis based on model content
- A platform for 'out of model' reviews and discussions
- For C-level executives looking for critical and strategic information
- For model consumers reviewing or implementing process flows
- For customers and other stakeholders who want to review and discuss the model
- For developers and software teams
- For those unfamiliar with modeling who want to understand and share their viewpoint
- A rich and evocative environment tailored for individual domains
- Reveals the hidden value in your Enterprise Architect models
- Engages with a broader audience - presentation and discussion at the 'right level'

For further information

Full documentation on Prolaborate is provided on the website. You can access this documentation by clicking on this link (JavaScript must be enabled on your web browser).

https://prolaborate.sparxsystems.com/resources/documentation

Integrate Data from External Providers

The Pro Cloud Server helps you to integrate the data from external providers into an Enterprise Architect model. Enterprise Architect is a team player and through its server-based integration capability helps you to create models that relate elements from a wide range of disparate tools, each of which might contain entities such as strategic objectives and Requirements through to implementation statements, work packages and configuration items. Enterprise Architect does not manage the master records for these items, but rather acts as an accumulator, bringing content into a single repository and allowing the items to be related. A variety of third-party providers can be integrated with Enterprise Architect, including:

-  Application Lifecycle Management (formerly HP Quality Center)
-  Jama Integration
-  Jazz (interacts with:
  - IBM Rational DOORS Next Generation's requirements management tool
  - Rational Rhapsody Design Management (DM)
  - Rational Team Concert Change and Configuration Management (CCM)
  - Rational Quality Manager (QM))
-  Jira and  Confluence
-  Polarion

-  Azure DevOps / Team Foundation Server
-  Wrike
-  ServiceNow
-  Autodesk
-  Bugzilla
-  Salesforce
-  SharePoint
-  Dropbox and
-  Other Enterprise Architect models

See the Install and Configure Help topic for information on how to configure each provider. Walkthroughs are also available for Walkthrough: Jira Integration and Walkthrough: Polarion Integration integration.

When an item from an external provider is selected in the list the meta-data for the item will be displayed in the appropriate Enterprise Architect window. So any property-and-value type of information will be displayed in the Properties window, descriptions and comments will be displayed in the Notes window and discussions or posts will be displayed in the Discuss & Review window. For example, if Jira was the External Data source and the integration was listing User Stories, a Jira User Story property such as *Priority: Medium* would appear in the Properties window, the *Story description* would appear in the Notes window and the *comments* would appear in the

Discuss & Review window.

Some meta-data such as collaboration information might not be available for all items and integrations, but where it is available the facility provides a uniform, cross-integration view of the meta-data, making it easy for Enterprise Architect users to understand the data from multiple providers and integrations without the need to leave the tool or grapple with vendor specific terminology. Enterprise Architect is performing the role of an accumulator, allowing information from a wide range of disparate sources to be related to the already rich set of architectural models in the tool, creating a view of how the information in these otherwise unrelated tools can be visualized. This removes the need for the bundle of static spreadsheets that organizations have traditionally used to relate pairs of items such as Test-Cases to Business Drivers, or Stakeholder Concerns to application services, and much more.

All integrations offer support for linking objects and elements from the external system into an Enterprise Architect client. The External Data window supports browsing the external provider's items and retrieving lists of elements and objects based on the provider's queries. Capabilities include:

- Link an Enterprise Architect element to an external object
- View external element properties
- View and in some cases add to, external object discussions
- Export links to WebEA URL's that correspond to the current model

- Open external items in a web browser
- Import elements
- Export elements

From Enterprise Architect Release 14.1 onwards it is possible to link a non-Cloud model to Integration Plug-ins configured on a Pro Cloud Server.

See the *Cloud Page* Help topic for configuration options.

Access

Ribbon	Specialize > Tools > System Integration > Open External Data To view a list of all elements that are linked to external items: Specialize > Tools > System Integration > Show All Linked Items
Context Menu	In a diagram or the Browser window, for elements that are already linked to an external item: Right-click on element > Specialize > External Properties
Keyboard Shortcuts	Alt+1 > System Integration

Features

Feature	Description
Pro Cloud Server Configuration	Each external provider must be configured on the Pro Cloud Server to enable connection. Multiple configurations can be made for each provider (such as connecting to two separate Jira servers).
Authorization	If the Integration Provider requires authorization you are prompted to enter your credentials. If the provider supports it, a new internet browser window will open and prompt you to log in to the Integration Provider and allow Enterprise Architect access to its resources. Alternatively, a simple dialog will pop up asking for your credentials, with the option to securely store them in the current model. If stored in the model the credentials will only be used for the current user.
Navigate External Provider	Find external items to link to by navigating the external provider.

External Item Details	Select an external item from the External Data window to see its properties, notes and discussions in the Properties window, Notes window and 'Discuss' tab of the Discuss & Review window.
Link External Items	External items can be linked to Enterprise Architect as a generic «ExternalReference» stereotype or as another element type.
Add WebEA Link	Hyperlinks to WebEA can be added to the external item so that you can quickly open the WebEA element from the external source. Right-click on a local linked element in the list and select 'Add WebEA Hyperlink to External Object'. This will update the external item with a link to the WebEA element that is linked to it in Enterprise Architect. Note that not all provider types have a 'link' mechanism. Where none exists, some providers might allow adding the link as a comment on the item. Ensure that the model has a valid WebEA address set in the model options.
Configuratio	Each Integration Provider comes with a

n	set of default mapping values which determines what type of local element is created in Enterprise Architect, as well as which fields are copied to the new element. These mappings are configurable via the 'Configure' option on the External Data window toolbar menu. You must have the 'Configure External Data Sources' permission to access this functionality.
Troubleshooting	• The System Output window will show any errors while attempting to retrieve data from the external providers • The Pro Cloud Server outputs log files for each external provider

Notes

- 'Integration' requires a Pro Cloud Server and is only available to Cloud models
- 'Integration' is currently provided with the Corporate, Unified and Ultimate Editions of Enterprise Architect
- If you select two or more elements at the same time, you can perform an operation on all of the selected elements at once
- For WebEA links to work a valid URL must be set for the

WebEA address (see the *Cloud Page* Help topic)

Install and Configure

The Sparx Systems Pro Cloud Server product is installed using a standard Microsoft Windows Installer package that includes a number of optional components. One of these components is Integration Plug-ins, which is installed by default into the C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\SBPI\ folder (assuming a 64 bit machine is being used).

Requirements

In order to install a particular Integration Plug-in you must have:

- A licensed Pro Cloud Server
- Physical network access to a server hosting the external data
- User credentials to access the external data
- Enterprise Architect Corporate, Unified or Ultimate Edition, v14 or later
  Nb: Enterprise Architect Trial edition provides read-only access to Integrations

What is in the Package?

The Integration Plug-ins folder initially consists of an Integration Server executable (SBPI.exe), a separate executable (*SBPI.exe) for each supported external product,

a sample configuration file and a text file describing the manual installation and configuration steps.

Plug-In	Description
Integration (SBPI) Server	The SBPI.exe application acts as the interface between the Pro Cloud Server and each of the plug-ins, by translating Enterprise Architect requests, forwarding them to the appropriate plug-in and then returning the generated response to Enterprise Architect.
Application Lifecycle Manager Plug-in	The ALMSbpi.exe plug-in interacts with the MicroFocus Application Lifecycle Manager product (previously known as HP Quality Center).
Autodesk Plug-in	The AutodeskSbpi.exe Plug-in interacts with AutoCAD's file and management component AutoDesk.
Bugzilla Plug-in	The BugzillaSbpi.exe Plug-in interacts with the web based defect/bug tracking system that was originally developed and used by the Mozilla project and is now licensed under the Mozilla Public License agreement.
Confluence	The ConfluenceSbpi.exe plug-in interacts

Plug-in	with Atlassian's Team Collaboration Software. It is able to list *spaces* and link to *pages*. Note that the content of the Confluence HTML pages is not synchronized.
Dropbox Plug-in	The DropboxSbpi.exe plug-in interacts with Dropbox's web based file hosting service. It is able to list folders within Dropbox and link to individual files.
EA Plug-in	The EASbpi.exe Plug-in interacts with external Sparx Systems's Enterprise Architect Cloud-based repositories. It is able to browse the Package hierarchy or perform search based queries.
Jazz Plug-in	The JazzSbpi.exe Plug-in interacts with: • IBM Rational DOORS Next Generation's Requirement management tool • Rational Rhapsody Design Management (DM) • Rational Team Concert Change and Configuration Management (CCM) • Rational Quality Manager (QM)
Jira Plug-in	The JiraSbpi.exe plug-in interacts with

	Atlassian's issue tracking system. It is able to list a user's favorite filters (also known as *starred* filters). Each filter will then list all the Jira items returned by the filter.
Salesforce	The SalesforceSbpi.exe plug-in interacts with Salesforce's Customer Relationship Management system.
ServiceNow Plug-in	The ServiceNowSbpi.exe plug-in interacts with ServiceNow's asset management component of its Cloud-based enterprise management system.
SharePoint Plug-in	The SharePointSbpi.exe Plug-in interacts with Microsoft's web-based collaborative platform, SharePoint.
Azure DevOps / TFS Plug-in	The TFSSbpi.exe Plug-in interacts with Microsoft's Azure DevOps / Team Foundation Server (TFS) work items
Wrike Plug-in	The WrikeSbpi.exe Plug-in interacts with Wrike's project management system.

How to Set Up

The Integration framework consists of an Integration server (SBPI.EXE) application that starts one or more Plug-ins (such as DropboxSbpi.exe and JiraSbpi.exe). The Integration Server and each Integration Plug-in can be configured to run either on the same machine as the Pro Cloud Server or on completely different machines. In the simplest configuration the Integration server and all Integration Plug-ins are installed on a single server. There are two main advantages with this configuration:

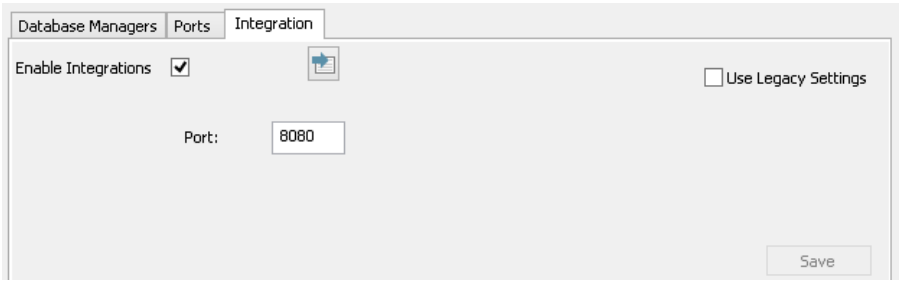
1. The Pro Cloud Server will automatically start (and stop) all configured Plug-ins whenever its Windows service is started (or stopped).
2. The Integration configuration GUI inbuilt into the Cloud Configuration client can be used to completely manage all aspects of the Integration configuration; see the *Steps - Simple* table.

However, if you elect to run the Integration Server or Integration Plug-ins on different machine(s) to the Pro Cloud Server, each of the individual Plug-ins must be manually configured as well as configured to start as Windows services with the correct parameters. See the *Steps - Manual* table.

Steps - Simple

The Integration configuration GUI included in the Cloud Configuration client removes most of the complexity

involved in configuring the Integration Server and Data Provider(s), therefore this is the recommended method for most users. However, this configuration method is restricted to running all Integration components (*SBPI.exe) on the same server as the Pro Cloud Server.

Step	Description
Configuring Pro Cloud Server for Integration	Each installation of Sparx Systems Pro Cloud Server can be configured to communicate with a single Integration Server; the configuration options of this definition are defined as a series of registry settings, however the 'Integration' tab in the Pro Cloud Server Configuration Client allows the definition and maintenance of the Integration Server options without the need for you to manually manipulate the registry or configuration files. This image shows the definition of an Integration Server with the default settings:  Alternatively, the Integration Server and Plug-ins can be configured via the

	WebConfig interface. See the WebConfig - Integration Plug-ins topic for details.
Configuring Integration Data Providers	Each Data Provider (or Plug-in) is defined both as a series of registry entries in [HKEY_USERS\.DEFAULT\Software\Sparx Systems\SQLBridge\SBPI\Plugins\{unique} and as settings within a configuration file. Again, the 'Integration' tab in the Cloud Configuration Client allows the definition and maintenance of Integration Data Provider details without the need for you to manually manipulate the registry and configuration files.
Configuring the Firewall	In an effort to minimize Firewall rules needed to configure the Pro Cloud Server and its features, PCS version 4.1 now routes all requests for the Integration server (SBPI.exe) via the normal PCS ports, therefore if you are using PCS 4.1 or later there are no additional Firewall rules needed other than the ones for Enterprise Architect client to communicate to the Pro Cloud Server. For versions 3 and 4 of the PCS, the Integration server (SBPI.exe) typically

	must be granted access through any local firewall so that Enterprise Architect clients can connect to it. The Pro Cloud Server installer will automatically create a Firewall exception that allows any incoming requests to be passed through to SBPI.exe; however, the default settings should be reviewed and adjusted to suit your environment.

Steps - Manual

Important: these steps are only needed if the Integration components will run on different machine(s) to the Pro Cloud Server, otherwise the *Steps - Simple* table should be used.

Step	Description
Configuring Pro Cloud Server for Integration	Each installation of Sparx Systems Pro Cloud Server can be configured to communicate with a single Integration (or SBPI) server. These configuration options are defined as a series of registry settings. This is an example of all valid options for the Integration server: [HKEY_USERS\.DEFAULT\Software\Sparx Systems\SQLBridge\SBPI\Server]

	"Enabled"="true" "LocalPort"=dword:00001f90 "UseLegacy"="false" "Arguments"="-port 8080 -protocol http" "Protocol"="https" "Server"="localhost" "Port"=dword:00001f90 "IgnoreSSLErrors"="true" "AttemptAutoDiscovery"="true" "ClientProtocol"="http" "ClientServer"="alternativeservername" "ClientPort"=dword:00001f90 • **Enabled** - true or false, representing the Port number that the SBPI server should be listening on, which value should match the value specified in the arguments; for example, dword:00001f90 (decimal 8080) • **LocalPort** - a hexadecimal value, representing the Port number that the Integration Server is listening on when the Use Legacy option is FALSE; for example, dword:00001f90 (decimal 8080) • **UseLegacy** - true or false, controls if the simple (false) or complex (true) set

	of configuration options should be used by the Integration Server • **Arguments** - not used in Pro Cloud Server 4.1 or later versions; in earlier versions this represents the arguments that are used to start the Integration server, which include the Port and Protocol the server should listen on - for example, "-port 8080 -protocol http" • **Protocol** - http or https, the protocol that should be used to communicate with the machine hosting the Integration server when the Use Legacy option is TRUE; this field is combined with the 'Server' and 'Port' to form the Integration Server's URL, which the Pro Cloud Server will send SBPI related requests to Note: The complete URL ({protocol}://{server-name}:{port} must be resolvable by the Pro Cloud Server machine • **Server** - the name (or IP number) of the machine hosting the Integration server when the Use Legacy option is TRUE (for example, yourdomain.com); this field is combined with the 'Protocol' and 'Port' to form the

	Integration Server's URL, which the Pro Cloud Server will send SBPI related requests to Note: The complete URL ({protocol}://{server-name}:{port} must be resolvable by the Pro Cloud Server machine • **Port** - a hexadecimal value, representing the Port number that the Integration Server is listening on when the Use Legacy option is TRUE - for example, dword:00001f90 (decimal 8080); this field is combined with the 'Protocol' and 'Server' to form the Integration Server's URL, which the Pro Cloud Server will send SBPI related requests to Note: The complete URL ({protocol}://{server-name}:{port} must be resolvable by the Pro Cloud Server machine • **IgnoreSSLErrors** - true or false, defines whether the SSL related errors that occur while communicating with the Integration Server component should be ignored when the Use Legacy option is TRUE • **AttemptAutoDiscovery** - true or false, defines if the Pro Cloud Server should

	automatically attempt to determine the Enterprise Architect client's network address and supply it to the Integration Server when the Use Legacy option is TRUE • **ClientProtocol** - http or https, defines the protocol that, when combined with the 'ClientServer' and 'ClientPort', forms the resolvable URL that Enterprise Architect clients can communicate to the Integration Server when the Use Legacy option is TRUE Note: The complete URL ({protocol}://{server-name}:{port} must be resolvable by the Enterprise Architect client machine • **ClientServer** - defines the server name (or IP number) that, when combined with the 'ClientProtocol' and 'ClientPort', forms the resolvable URL that Enterprise Architect clients can communicate to the Integration Server when the Use Legacy option is TRUE Note: The complete URL ({protocol}://{server-name}:{port} must be resolvable by the Enterprise Architect client machine • **ClientPort** - a hexadecimal value defining the Port number that, when

	combined with the 'ClientProtocol' and 'ClientServer', forms the resolvable URL that Enterprise Architect clients can communicate to the Integration Server when the Use Legacy option is TRUE; for example, dword:00001f90 (decimal 8080) Note: The complete URL ({protocol}://{server-name}:{port} must be resolvable by the Enterprise Architect client machine **Note:** From version 4.1 onwards of the Pro Cloud Server, the Integration Server does not need to have Firewall rules of its own so that Enterprise Architect clients can communicate with it. For PCS versions 3 and 4 the Integration Server (SBPI.exe) typically needs to be granted access through any local firewall so that clients can connect to it. The Pro Cloud Server installer will automatically create a Firewall exception that allows any incoming request to be passed through; however, the default settings should be reviewed and adjusted to suit your environment.
Configuring Data	Each Data Provider is defined as a series of registry entries in:

Providers	[HKEY_USERS\.DEFAULT\Software\Sparx Systems\SQLBridge\SBPI\Plugins\{unique} where {unique} is a unique UUID for the Data Provider. This is an example of a complete External Data Provider definition: [HKEY_USERS\.DEFAULT\Software\Sparx Systems\SQLBridge\SBPI\Plugins\{853489C1-4C22-4bad-9A8E-3098D07A3FC1}] "AutoStart"="true" "Enabled"="true" "Group"="" "Name"="Sparx Systems Sample account" "Port"=dword:00001f91 "Prefix"="jr1" "TypeKey"="jira" "Arguments"="-port 8081 -config jr.config" "Config"="jr1.config" • **AutoStart** - true or false, defines if the Integration Server (SBPI.exe) should maintain a running process (*sbpi.exe)

	for this Data Provider • **Enabled** - true or false, defines if the Integration Server (SBPI.exe) should allow communications to be forwarded to this Data Provider • **Group** - an optional value that can be used to 'sort' Providers into groups when displayed in Enterprise Architect • **Name** - a 'friendly' project name to describe the external Data Provider, which is displayed to all Enterprise Architect users; for example 'Sparx Systems Sample account' • **Port** - a hexadecimal value representing the Port number that the external data source expects to receive requests on, which value should match the value specified in the arguments; for example, dword:00001f91 (decimal 8081) • **Prefix** - a short unique name that is meaningful to the users and that prefixes each link stored within the Enterprise Architect model; for example, jr1::10001 (where 10001 is the Jira object ID) • **TypeKey** - defines the Provider type of the current Data Provider; only these supported values can be used: cint,

	csvc, alm, ad, bug, cflu, drop, ea, jazz, jira, sf, now, sp, tfs, wrike • **Arguments** - deprecated from PCS 4.1 onwards, a dynamic arguments list is built from the individual settings; earlier versions used this field to define the arguments that should be used to start the Integration Plug-in, including the Port, Protocol and Config filename • **Config** - deprecated from PCS 4.1 onwards, the config filename is now the prefix with a '.config'; in versions PCS 3 and PCS 4 this field was given an independent value that had to be unique Each External Data Provider requires its own set of options to define the details of how the configured Plug-in connects to the External Data Source. These settings are stored in a .config file that resides in the same location as the Plug-in's .exe file. For example, using the above definition a 'jr1.config' would need to be created, and would contain information similar to this: PROTOCOL=https SERVER=example.com PORT=443

	BASEURL=myproject USERNAME= PASSWORD= CREATEITEMS=false MODIFYITEMS=false POSTDISCUSSIONS=true PROXY=10.0.0.0:3128 PROXYBYPASS=<local>;10.* IGNORESSLERRORS=true • **PROTOCOL** - the communication protocol, http or https • **SERVER** - the name (or IP number) of the external data source's server, such as example.com • **PORT** - the Port the external data source is configured to listen on, such as 443 • **BASEURL** - when the external data source supports multiple 'projects', the BaseURL property identifies which should be used; for example, for Enterprise Architect SBPI the BaseURL is the DB Alias as defined in the PCS • **USERNAME** - optional - see Note1 • **PASSWORD** - optional - see Note1 • **CREATEITEMS** - defines if

	Enterprise Architect users can create items in the External Data Source - see Note2; default value: false • **MODIFYITEMS** - defines if Enterprise Architect users can modify existing items in the External Data Source - see Note2; default value: false • **POSTDISCUSSIONS** - defines if Enterprise Architect users can add Discussions to items within the External Data Source - see Note2; default value: true • **PROXY** - optional - the server name or IP number and Port of the proxy server, such as 10.0.0.0:3128 • **PROXYBYPASS** - optional - a semi colon separated list of IP numbers that should not be sent through the proxy; for example, <local>;10.* • **IGNORESSLERRORS** - optional - a boolean value to ignore any SSL certificate errors **Note1** • If the external server supports OAuth 2 authentication (Autodesk, Dropbox and Wrike), when accessing from within Enterprise Architect, a browser

	window will open and prompt you to log in to the external account and allow Enterprise Architect to access your account • Enterprise Architect never sees your credentials in this process; instead it is provided with a unique token to access the external account • If the external server allows basic authentication, then the username and password can be optionally set in the configuration file • If the username and password are not specified in the configuration file, Enterprise Architect will prompt you for credentials to access the external data **Note2** • Not all external products support the creation and modification of their objects; for example, Dropbox does not
Configuring Models for External Data Providers	In order for users of an Enterprise Architect model to connect to a given External Data Provider, a 'binding' between the Enterprise Architect model and the External Data Provider must be configured. This is performed by a series of registry settings in

	[HKEY_USERS\.DEFAULT\Software\Sparx Systems\SQLBridge\SBPI\Bindings\{unique}] for each model/External Data Provider combination. In this string, {unique} is an 8-digit hexadecimal number uniquely identifying the binding. For example: [HKEY_USERS\.DEFAULT\Software\Sparx Systems\SQLBridge\SBPI\Bindings\B6EE6851] "LocalModel"="eaexample" "Plugin"="{853489C1-4C22-4bad-9A8E-3098D07A3FC1}" • LocalModel - the DB Alias of the model, such as 'eaexample' • Plugin - the UUID of the external data source; for example, {853489C1-4C22-4bad-9A8E-3098D07A3FC1}

Special Notes

Using SSL at the Integration Server or Provider Level

If you choose to implement the HTTPS protocol at either the

Integration Server level or Integration Provider level, the Integration Executables (*SBPI.EXE) will require a 'server.pem' file in the same folder as themselves; therefore, if using the default installation path this would mean 'server.pem' should be placed into the C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\SBPI\ folder.

See the section *SSL Certificates* in the Add a Port Definition Help topic and the Self-Signed SSL Certificates Help topic for more information on how to create a valid 'server.pem' file.

The Integration Plug-in for Enterprise Architect Requirements:

- The BaseURL as the model's DB Alias
- The defined model configured as 'Enabled' and 'Enable Pro Features (OSLC, WebEA and Integration)' in the Pro Cloud Server

Integration Plug-ins

The Integration Plug-in components represent a set of Plug-ins (or extensions) for the Pro Cloud Server that enable data from external providers to be displayed within Enterprise Architect. These components are also known as the Server Based Plug-in Interface (SBPI).

The Integration Plug-in components of the Pro Cloud Server run as a series of executables (.exe files) that do not have a graphical user interface (GUI) themselves; however, the topics of this section describe the configuration editor screen(s) that are included in the Pro Cloud Server's Configuration Client that can be used to configure the Integration Plug-ins.

The main screen of the Configuration Client (installed as part of the Pro Cloud Server) includes an 'Integration' tab that is divided into two main sections. The top half defines the details of the Integration Server (or SBPI Server) with the lower half displaying the details of the various 'Data Providers'; each of these sections is described in greater detail in this topic.

Note 1: The Integration feature is only available in licensed editions of the Pro Cloud Server: Team, Enterprise or Token.

Note 2: If you are using the Token edition of the Pro Cloud Server, before configuring Integrations ensure you have enabled the required Integration Provider/s via the Token Allocation options; see the Manage Allocations Help topic.

Note 3: If you are using the Enterprise Architect Trial edition, the Integration Plug-ins are read-only and will not update any stored data

Note 4: Any changes to the Integration Server or Plug-ins (made either through the Configuration Client or manually) will require a restart of the Pro Cloud Server

Alternatively, the Integration Server and Plug-ins can be configured via the WebConfig interface. See the WebConfig - Integration Plug-ins Help topic for details.

Detailed Walkthroughs are also available for Walkthrough: Jira Integration and Walkthrough: Polarion Integration integration specifically.

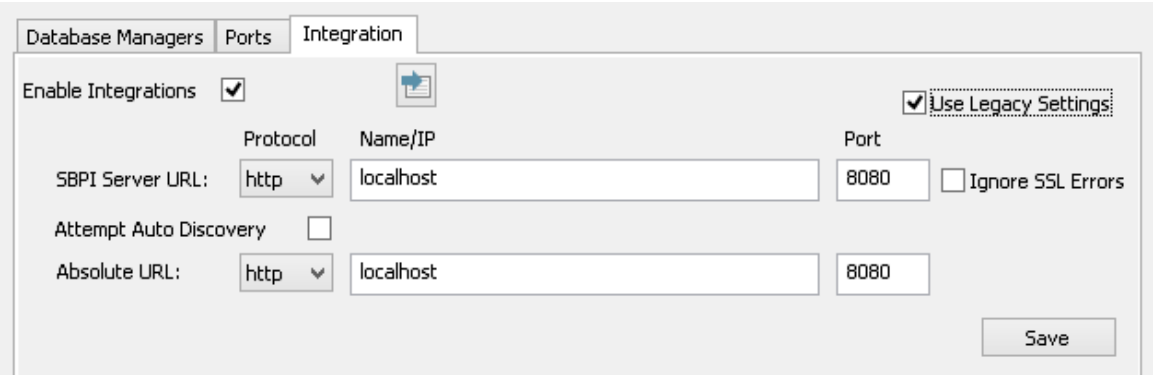
Integration Server



All requests for external data from Enterprise Architect are sent to the Integration Server (via the Pro Cloud Server), which decides which Data Provider (or SBPI Plug-in) will action the request.

To make the Integration Server configuration as easy as possible for the majority of users, version 4.1 of the Pro Cloud Server has introduced a simplified set of options;

however, if the default configuration does not suit your environment, select the 'Use Legacy Settings' option to show all the available options.



Note: The Integration Server settings are only mandatory when the 'Enable Integrations' flag is checked. The combination of Protocol, Name and Port must match the machine hosting the Integration Server (in this case the Pro Cloud Server) in order for Enterprise Architect clients to be able to communicate with it.

Item	Description
Enable Integrations	Default value: Unchecked This checkbox enables (or disables) Integrations for the current Pro Cloud Server installation. When this value is selected it implies that the other fields ('Protocol', 'Name', 'Port' and 'Path') are now mandatory and the Integration Server definition can not be saved without a value in each field.
Load	This button loads the default values for

Defaults	the Integration Server settings into any empty field. **Note:** If a field already has a value, then the Load Defaults button will not change or update this existing value
Use Legacy Settings	Default value: Unchecked This checkbox determines if the simplified or complex set of options should be shown. The name refers to the fact that the earlier versions of the Pro Cloud Server required users to fill in the complex set of options. In later versions of the Pro Cloud Server the set of options needed was greatly reduced by making some assumptions that will be true for the majority of environments. When this value is selected it implies that the other fields (SBPI Server URL: 'Protocol', 'Name/IP' and 'Port') are now mandatory and the Integration Server definition can not be saved without a value in each field.
Port	Default value: 8080 This option is only shown when the 'Use Legacy Settings' option is not selected. It defines the Port that the Integration Server is listening to for requests from the

	Pro Cloud Server. Ensure that there is no other application or service already using the chosen Port.
SBPI Server URL: Protocol	Default value: HTTP This option is only shown when the 'Use Legacy Settings' option is checked. This field defines the SBPI Server's Protocol, combined with the 'SBPI Server Name/IP' and 'SBPI Server Port'; these settings form the Integration Server's URL that the Pro Cloud Server will send SBPI related requests to. Note: The complete URL ({protocol}://{server-name}:{port} must be resolvable by the Pro Cloud Server machine.
SBPI Server URL: Name/IP	Default value: localhost This option is only shown when the 'Use Legacy Settings' option is checked. This field defines the SBPI Server's machine name or IP, combined with the 'SBPI Server Protocol' and 'SBPI Server Port' these settings form the Integration Server's URL that the Pro Cloud Server will send SBPI related requests to. Note: The complete URL

	({protocol}://{server-name}:{port} needs to be resolvable by the Pro Cloud Server machine.
SBPI Server URL: Port	Default value: 8080 This option is only shown when the 'Use Legacy Settings' option is checked. This field defines the SBPI Server's Port, combined with the 'SBPI Server Protocol' and 'SBPI Server Port'; these settings form the Integration Server's URL that the Pro Cloud Server will send SBPI related requests to. Note: The complete URL ({protocol}://{server-name}:{port} needs to be resolvable by the Pro Cloud Server machine.
Ignore SSL Errors	Default value: unchecked This option is only shown when the 'Use Legacy Settings' option is checked. This value defines if SSL related errors that occur while communicating with the Integration Server component should be ignored. The most common SSL errors are related to self-signed certificates; by default a client does not treat self-signed certificates as being secure, therefore in order to connect to servers that use

	self-signed certificates this option should be checked.
Attempt Auto Discovery	Default value: checked This option is only shown when the 'Use Legacy Settings' option is checked. This option defines if the Pro Cloud Server should automatically attempt to determine the Enterprise Architect client's network address and supply that to the Integration Server. This option was introduced in Pro Cloud Server 4.1.40; earlier versions were not capable of discovering the client's network address. When 'Attempt Auto Discovery' is checked, the setting is the 'Fallback URL', which will be used only if the Auto Discovery fails. When 'Attempt Auto Discovery' is unchecked, the setting is the 'Absolute URL' which will be used in all cases.
Absolute URL: Protocol	Default value: HTTP This option is only shown when the 'Use Legacy Settings' option is checked. This field defines the Protocol that when used in combination with the Absolute Server Name and Port form the

	resolvable URL that Enterprise Architect clients can communicate to the Integration server. Note: The complete URL {protocol}://{server-name}:{port} needs to be resolvable by Enterprise Architect client machines. In some environments server URLs are redirected to completely different locations, therefore to handle this situation the absolute URL is returned to Enterprise Architect clients so that they can communicate to the Integration Server.
Absolute URL: Name/IP	Default value: {empty} This option is only shown when the 'Use Legacy Settings' option is checked. This field defines the Server name/IP that when used in combination with the Absolute Protocol and Port form the resolvable URL that Enterprise Architect clients can communicate to the SBPI server. Note: The complete URL {protocol}://{server-name}:{port} needs to be resolvable by Enterprise Architect client machines. In some environments server URLs are

	redirected to completely different locations, therefore to handle this situation the absolute URL is returned to Enterprise Architect clients so that they can communicate to the SBPI Server
Absolute URL: Port	Default value: 8080 This option is only shown when the 'Use Legacy Settings' option is checked. This field defines the Port that when used in combination with the Absolute Protocol and Server Name form the resolvable URL that Enterprise Architect clients can communicate to the SBPI server. Note: The complete URL {protocol}://{server-name}:{port} needs to be resolvable by Enterprise Architect client machines. In some environments server URLs are redirected to completely different locations, therefore to handle this situation the absolute URL is returned to Enterprise Architect clients so that they can communicate to the SBPI Server
Save	This button saves any pending changes to the Integration Server settings; it is only enabled when there are unsaved changes.

Data Providers

A single Pro Cloud Server can support any number of external Data Providers, and in turn each of the external Data Providers can be available to as many models as are supported by the installation's license. Note that only Pro-enabled repositories (or database managers) can access external data providers. A Pro-enabled database manager is one that has the 'Enable Pro Features (OSLC, WebEA and Integration)' option checked.

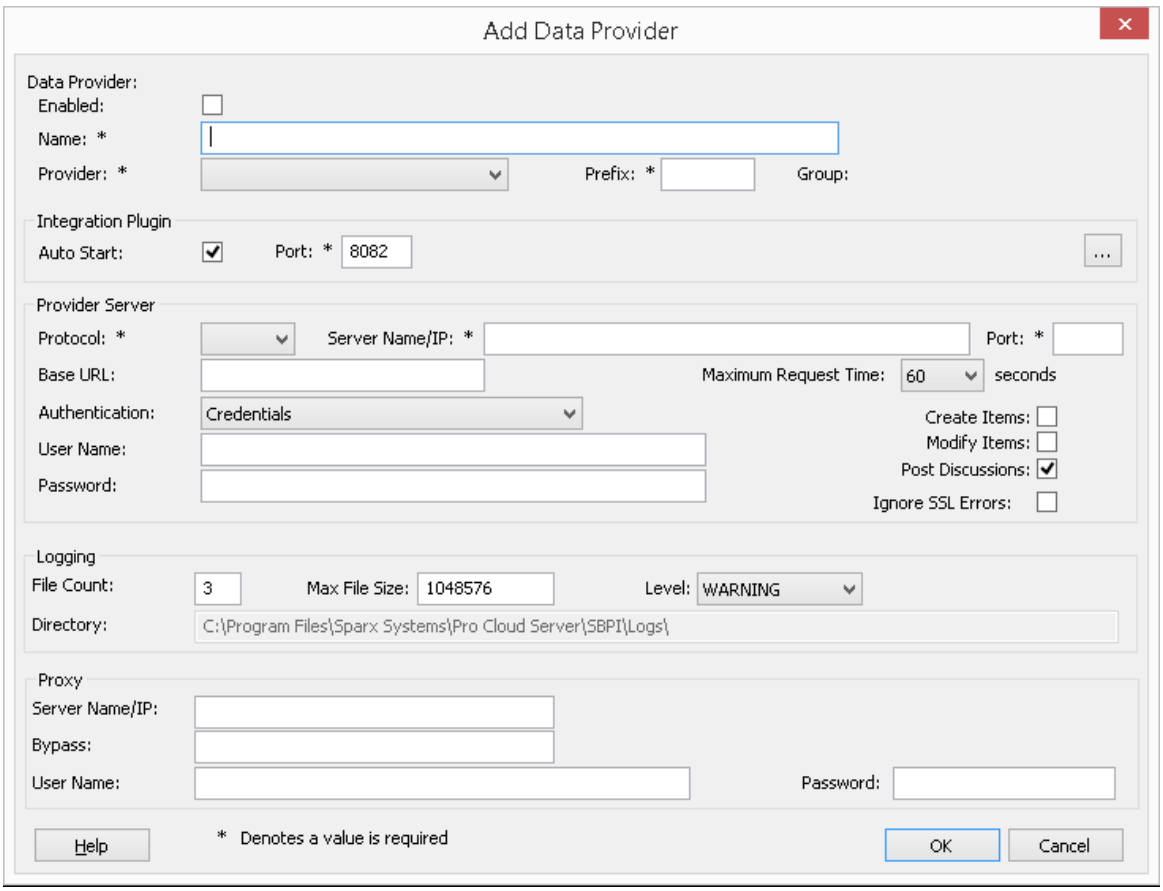
Option	Description
Defined Providers	This controls lists a summary of all defined external Data Providers. Double-clicking an existing item will display the Edit Data Provider screen.
Add...	This button displays the Add Data Provider screen to allow the entry of a new external Data Provider.
Edit...	This button displays the Edit Data Provider screen to allow the modification an existing external Data Provider definition. **Note:** This button is only enabled when

	there is a Data Provider selected.
Remove	This button permanently deletes the selected Data Provider and all of its Bindings. **Note:** This button is only enabled when there is a Data Provider selected.
<'plug-in name'> is bound to:	This control displays a check list of all Database Managers that are Pro-enabled (that is, the 'Enable Pro Features (OSLC, WebEA and Integration)' option is selected). A selected Database Manager allocates the current Data Provider to the given repository.
Check All	This button is a quick and easy way to allocate all defined Pro-enabled Database Managers to the currently selected Data Provider.
Uncheck All	This button is a quick and easy way to de-allocate all defined Pro-enabled Database Managers from the currently selected Data Provider.

Add/Edit Data Provider

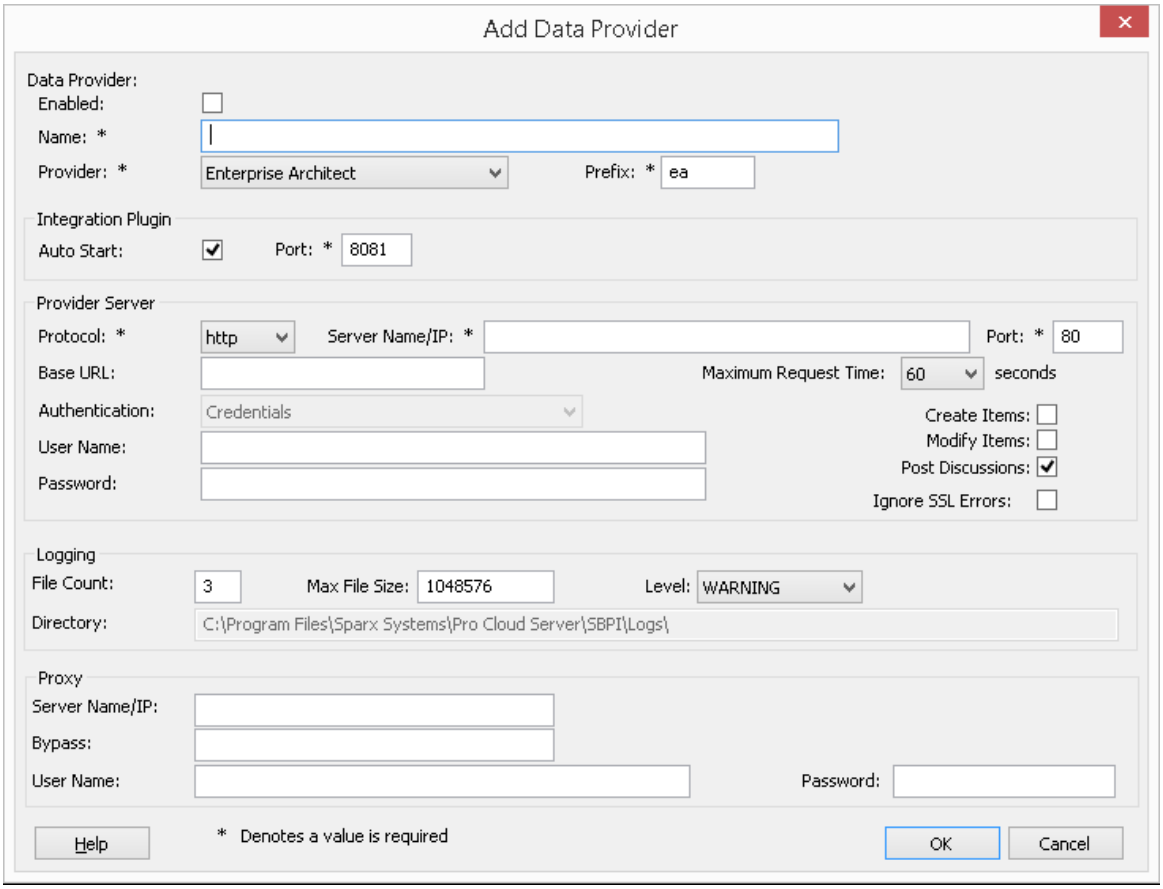
The Add/Edit Data Provider screen will be shown whenever you request to create a new Data Provider or edit an existing one. The screen's behavior will be identical in either mode, the one difference being that the details of the selected Data Provider will be displayed when the screen loads in 'Edit' mode.

When you choose to add a new Data Provider, the screen will initially display as shown:



Wherever possible the Add/Edit Data Provider screen will fill each field with a default value in order to make it easier for you; if the default values are not correct, simply overwrite them. To this end, whenever the 'Provider' value is

changed, a number of other fields will be set to the default values for the new Provider, such as the Data Provider's Prefix, as shown here when Enterprise Architect is selected:



Note: When defining a Data Provider there are a number of mandatory fields (denoted with a '*') that require a value before the Data Provider can be saved.

Data Provider settings

Item	Description
Enabled	Default Value: unchecked This value controls if the Integration Server (SBPI.exe) should allow

	communications to be forwarded to this Data Provider.
Name	This value represents the friendly name for the current Data Provider. It will be shown to users of Enterprise Architect.
Provider	This value specifies the Provider type of the current Data Provider; only supported values can be used. As mentioned, whenever this value is changed a number of other fields will be automatically updated to make data entry easier for the user.
Prefix	This value represents a short unique name to identify the current Data Provider. This value will be saved against every Enterprise Architect element that is linked to an external element.
Group	This field is available only when the 'Custom Integration' provider type is selected. Typically in Enterprise Architect, integrations are grouped by the Provider type (such as 'Jira' or 'Dropbox'). For Custom Integrations, the Grouping is applied based on the value entered in this

	field.
Max Request Time	This field is available only when the 'Custom Service' provider type is selected. This value represents the maximum amount of time (in seconds) that calls to the Provider system will wait before timing out. Increasing this value is particularly useful when the Provider systems are remote or have slow performance. The default value is 60 seconds. Note: This is a new setting introduced in Pro Cloud Server v4.2 build 65.

Integration Plug-in settings

This group of settings defines the relationship between the Integration Plug-in and the Integration Server and how the two interact.

Item	Description
Auto Start	Default Value: Checked This value controls if the Integration Server (SBPI.exe) should maintain a running process (*sbpi.exe) for this Data

	Provider. For standard Integration Plug-ins we would expect this value to be checked; however, for custom ones it might be useful to not auto start the Plug-in and instead manually control when it is running, particularly while developing it.
Port	Default Value: 8081 This value represents the Port that the Integration Server should use to communicate with the Integration Plug-in. The Integration Server assumes the Port is relative to the current machine (or http://localhost:{port}). Each Integration Plug-in requires a unique Port number, which will be checked when an attempt is made to save the Plug-in's settings. However, the Pro Cloud Server can only check what Ports are in use by the Pro Cloud Server itself and its components, there is still a possibility of a Port clash with other applications.
Protocol	This value represents the protocol that the Integration Server should use to communicate with the Integration Plug-in; in the simplest configuration this

	will be 'HTTP'.
DLL Path	This field is only available for 'Custom Service', 'Custom Integration' and 'Translator' provider types. This value represents the physical path to the custom DLL, that will be called by the CustomService.exe or CustomIntegration.exe Data Provider processes. As a convenience, the application populates the drop list with all .DLL files that are present in the .\SBPI\Custom\ folder (C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\SBPI\Custom\).

Custom Properties

The Custom Properties section is only shown for Data Providers of type 'Custom Service'.

Item	Description
List of Custom Properties	Lists the current properties of the Custom Data Provider.

Add	This button shows the 'Add SBPI Provider Custom Property' screen and allows entry of a new custom property, defined as an Item/Value pair.
Edit	This button shows the 'Edit SBPI Provider Custom Property' screen and allows modification of an existing custom property definition.
Delete	This button permanently removes the selected Custom Property from the current Data Provider.

Provider Server settings

This group of settings defines the details of the external data provider, such as Dropbox, Jira or TFS. The Integration Plug-in will use the defined details to connect to the remote system and retrieve data from it so that it can be sent back to Enterprise Architect. The combination of 'Protocol', 'Server Name 'and 'Port' must be resolvable by the server hosting the Integration Plug-in, which in the simplest case is the Pro Cloud Server.

Item	Description
Protocol	This value represents the protocol that the

	Integration Plug-in should use to communicate with the external Data Provider. Note: Most online Cloud-hosted providers require https and Port 443.
Server Name/IP	This value represents the Server Name or IP address that the Integration Plug-in should use to communicate with the external Data Provider.
Port	This value represents the Port that the Integration Plug-in should use to communicate with the external Data Provider. Note: Most online hosted accounts operate via https, which requires a Port of 443. Self-hosted servers will depend on the individual server configuration. • Jazz: server defaults to using http Port 9080 or https 9443 • Jira: self-hosted server default Port is http Port 8080 • Confluence: self-hosted server default Port is http Port 8090
Base URL	Some external Data Providers allow for the concept of multiple repositories at a single location; for such Data Providers

	this field acts as a filter. For example, it is possible to use another Enterprise Architect model as an external data source; in this case the 'Protocol', 'Server' and 'Port' determine the Pro Cloud Server (for example, http://myEAServer:804) while the 'Base URL' is the DB Alias of the repository (myModel). Thus: http://myEAServer:804/myModel Details for specific providers: • Confluence: online Cloud hosted accounts require this to be 'wiki'; self-hosted servers will be dependent on the individual server set-up • Jira: for online Cloud hosted accounts this should be blank; self-hosted servers will be dependent on the individual server set-up • Dropbox: this should be left blank • Enterprise Architect: this should be the 'model alias' of the target Enterprise Architect model
Maximum Request Time	This value represents the maximum amount of time (in seconds) that calls to the Provider system will wait before

	timing out. Increasing this value is particularly useful when the Provider systems are remote or have slow performance. The default value is 60 seconds. Note: This is a new setting introduced in Pro Cloud Server v4.2 build 65.
Authentication	Default Value: Credentials This value allows the selection of the type of Authentication that should be used for the Provider, the valid options are Credentials and OAuth. When 'Credentials' is selected it would be expected that a User Name and Password would be entered. When 'OAuth' is selected it would be expected that the Advanced Authentication options would be filled in on the Data Provider Advanced Authentication screen. **Note 1**: this option is only enabled if the selected Provider supports multiple forms of authentication, for example: Dropbox, Jira, Salesforce, ServiceNow, SharePoint and Wrike **Note 2**: this option was introduced in Pro

	Cloud Server v5.0 build 100, prior releases only support credential based authentication.
User Name	This value represents the user name that should be used (in combination with the password) to access the data within the external system. If a value is defined in this field all Enterprise Architect users of the current Data Provider will use the same set of credentials to read the external system. Otherwise, if the user name field is left empty, Enterprise Architect will prompt each user independently for their credentials. This value is only shown when the Authentication of 'Credentials' is selected. Note: For many online Cloud accounts, the username can be an email address for the account.
Password	This value represents the matching password for the user name specified in the 'User Name' field. This value is only shown when the Authentication of 'Credentials' is selected. **Note:** For Atlassian Cloud hosted accounts (Jira, Confluence) the use of regular passwords has been replaced with

	an API Key. See the Atlassian documentation on how to obtain an API key and paste the API key into the password field.
Advanced...	This button is used to show the Data Provider Advanced Authentication screen and is only visible when the Authentication of OAuth is selected.
Create Items	Default value: unchecked This checkbox controls if Enterprise Architect users are able to create new items within the external system.
Modify Items	Default value: unchecked This checkbox controls if Enterprise Architect users are able to modify the details of external items within the external system.
Post Discussions	Default value: checked This checkbox controls if Enterprise Architect users are able to create discussions against elements within the external system.
Ignore SSL Errors	Default value: unchecked This checkbox controls if the Integration

	Plug-in should ignore SSL related errors that occur as a result of communicating with the external system.

Logging settings

This group of settings define how the Integration Plug-in will create entries in its log file(s). Since the Integration Plug-ins run without user intervention, it is important for them to be able to write any messages or reports of potential problems to a log file. However, special consideration must be given to ensuring that performance isn't impacted by writing to such files, which is possible if the log file continually grows without limits. It is also unreasonable to expect that manual intervention should be required to ensure that the size and number of log files don't impact performance. For these reasons each Integration Plug-in has its own set of log files and configuration options to manage those log files.

The current log file will always include a '1' on the end of its filename. A new log file will be created whenever the Integration Plug-in is started or the physical size of the current log file reaches the Max File Size. When this occurs all existing files will be 'rolled over', meaning their file numbers will be incremented and a new '{filename}1.log' created. If the roll over process causes more files to exist than the 'File Count' setting allows, these additional files will be deleted.

Item	Description
File Count	Default value: 3 This value represents the 'rolling' number of log files that should be retained for the current Integration Plug-in.
Max File Size	Default value: 1048576 This value represents the maximum size, in number of bytes, that a log file can reach before a new log file is created.
Level	Default value: WARNING This value represents the maximum level of messages that should be written to the log file. The levels are: OFF, FATAL, WARNING, INFO and SYSTEM. A defined log level is inclusive of all lower levels; therefore if a Plug-in is set to a log level of INFO then all FATAL, WARNING and INFO messages will be written to its log file.
Directory	(READ-ONLY) This field displays the physically location of the directory into which the log files for the current Data Provider will be saved.

Proxy settings

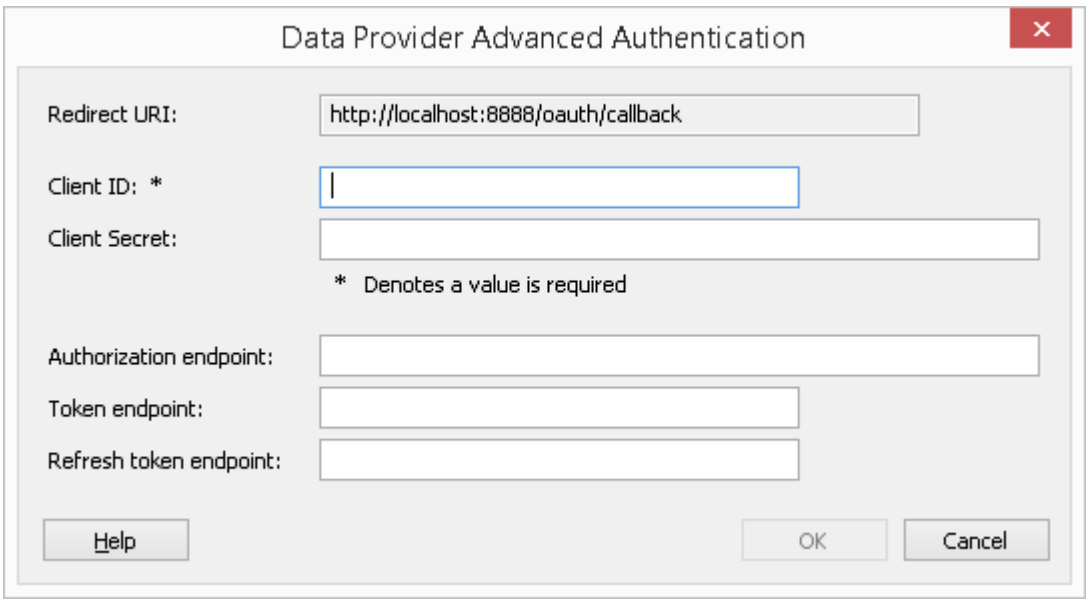
This group of settings defines how the Integration Plug-in should communicate with the External Data Provider when they are separated by a Proxy Server.

Item	Description
Server Name/IP	The server name (or IP) and Port number of the Proxy Server; for example, proxyserver.com:3131
Bypass	This field provides a mechanism for certain addresses to bypass the proxy; it accepts multiple values separated by semi-colons (;). For example, 192.168.*;*.localdomain.com;www.mydomain.com
User Name	If the Proxy Server requires credentials, this field provides the user name.
Password	If the Proxy Server requires credentials, this field provides the user password.

Data Provider Advanced Authentication

This screen allows the entry of all settings used for OAuth authentication for the current Integration Provider.

Note: OAuth is only supported by some Integration Providers.



Integration Server

Item	Description
Redirect URI	This read-only value displays the Pro Cloud Server's URI that OAuth systems will make callbacks to. Important: When defining the Data Provider in OAuth the Redirect URI will need to be saved so that OAuth knows

	what URI it should communicate to.
Client ID	This value represents the unique identifier of the Data Provider in OAuth and is sometimes referred to as the "App ID".
Client Secret	This value represents a secret that only the Data Provider should know, essential like a password. When defining the Data Provider in the OAuth system the Client/App ID would need to have a secret defined.
Authorization endpoint	The authorization endpoint is a complete URL of the OAuth system that Pro Cloud Server communicates with to determine if the current user is authorized to access the current Data Provider. All necessary communications between the user and OAuth occurs in the user's browser and Pro Cloud Server is notified of the outcome. **Note**: A value for authorization endpoint is not mandatory as the Pro Cloud Server has a set of known addresses that will be used. An example is: https://login.microsoft.com/oauth/authori

	ze
Token endpoint	The token endpoint is used by Pro Cloud Server to obtain access tokens, after the user has been granted access to the system. **Note**: The Token endpoint can be either a relative or full URL.
Refresh token endpoint	The refresh token endpoint is usually the same as token endpoint, but some providers use a different URL for obtaining refresh tokens. If left blank, then the token endpoint will be used.
OK	This button saves any pending changes and closes current screen. It is only enabled when all mandatory fields have a value and there are unsaved changes.

Walkthrough: ServiceNow Integration

This walkthrough helps you to set up and use the integration capability of Pro Cloud Server to integrate your Enterprise Architect models with the corresponding ServiceNow tables.

Using the integration with ServiceNow you can:

- Link Enterprise Architect elements with their counterparts in ServiceNow
- Create Enterprise Architect elements based on ServiceNow items, and vice-versa
- Perform a traceability analysis

Prerequisites

In order to set up integration between ServiceNow and Enterprise Architect, you must first:

- Be using a licensed installation of Pro Cloud Server with the Integration Plug-ins component
- Select the 'Enable Pro Features (OSLC, WebEA and Integration)' option in the Pro Cloud Server configuration for each Database Management System hosting your Enterprise Architect models
- Have network access between the PCS server and ServiceNow
- Have user credentials to access the external data
- Be using Enterprise Architect Corporate, Unified or Ultimate edition v14 or later

We will assume that the Integration Server and the ServiceNow Integration Plug-in are to run on the same machine as the Pro Cloud Server.

Minimum required permissions

Read access is required on these tables:

- **sys_filter** - lists the custom filters that determine what is displayed in Enterprise Architect
- **sys_db_object** - list of tables to group the filters by the main base table used
- **sys_dictionary** - for the display name of table fields
- Any table that the custom filter uses

To enable 'push' synchronization, the user will require write access to the tables as well.

Read access can be enabled by adding an Access Control (ACL) rule on the table.

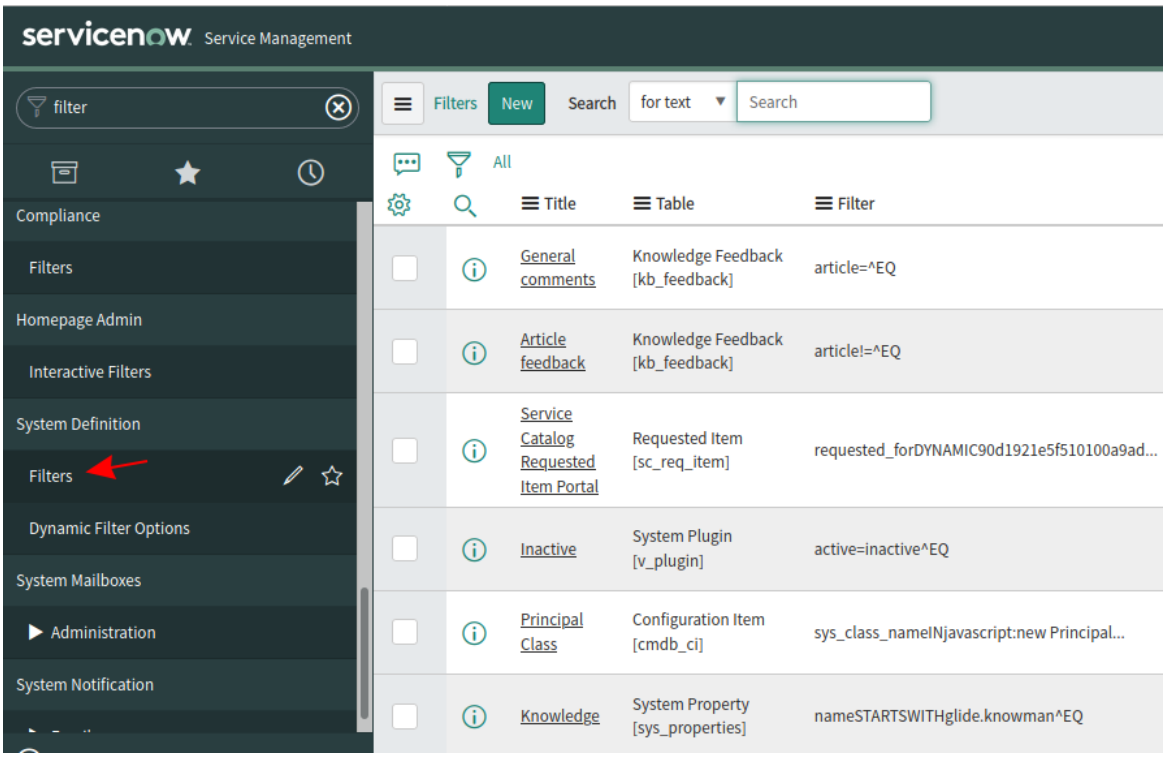
A second rule will be required to enable read access on all fields in the table (e.g. **table.***).

The Access Control rules should have a required Role assigned, and then that role can be given to a group or individual user.

Note: On some tables it is required to 'Elevate role' from a System Administrator to a 'Security Administrator - security_admin' to be able to edit Access Controls.

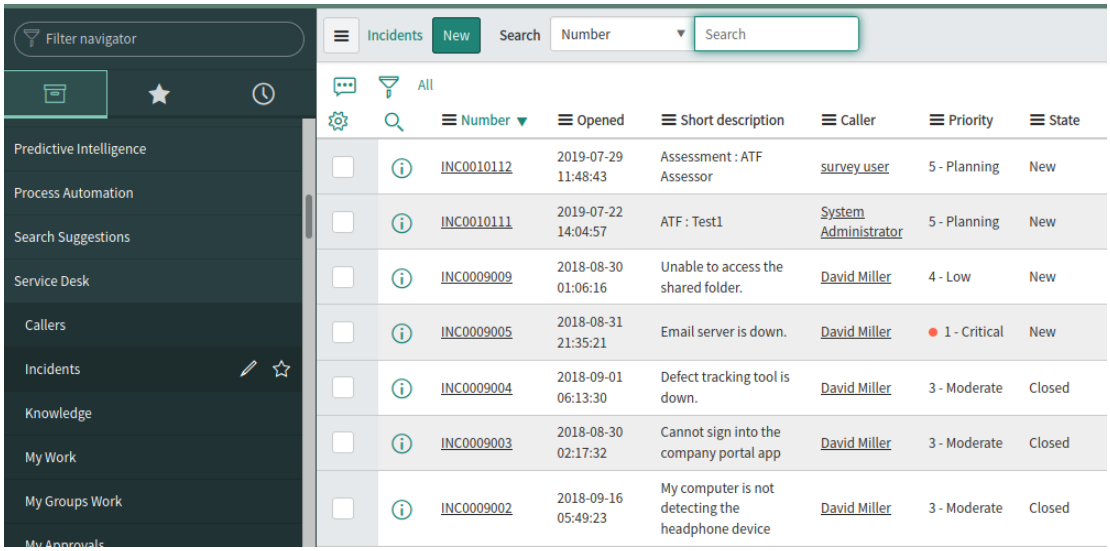
Define ServiceNow table filters

The ServiceNow integration allows access to items via saved table filters. By default ServiceNow typically includes a number of predefined global filters. You can browse all your defined filters in ServiceNow using the 'System Definition | Filters' option.

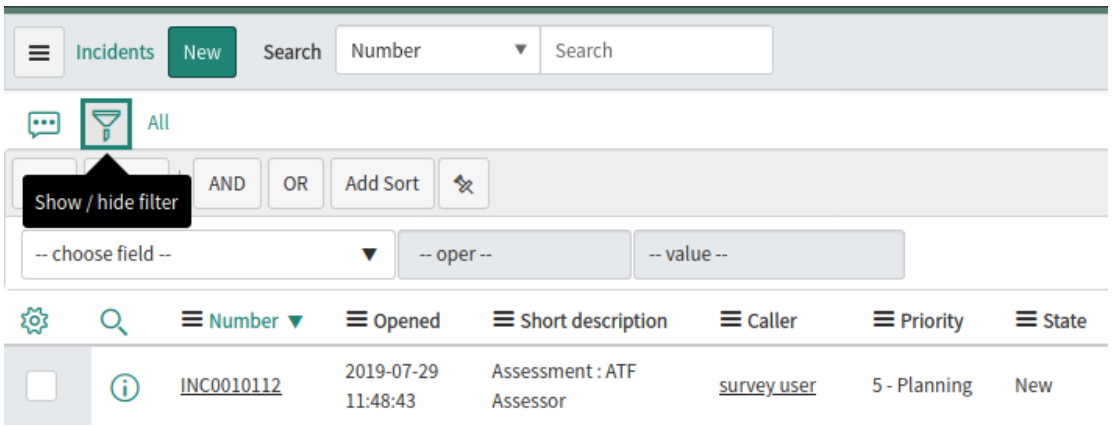


You can create new filters in ServiceNow by following these steps:

1. Locate the relevant data within ServiceNow (e.g. Incidents)

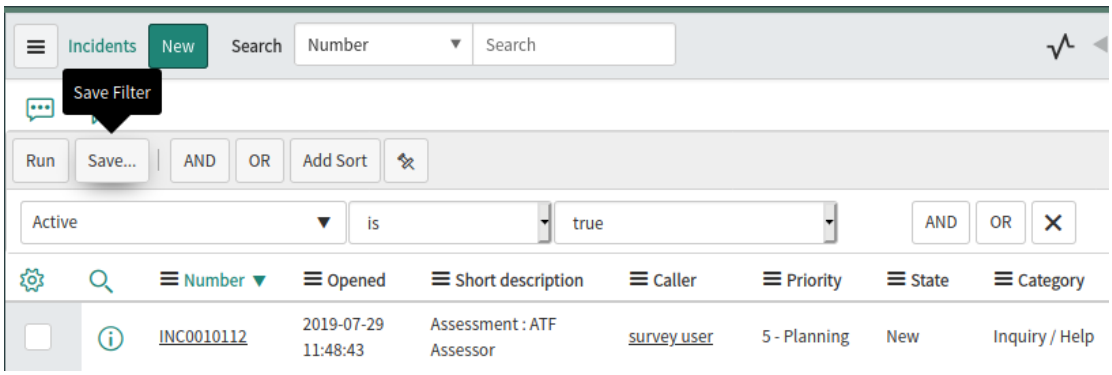


2. Show the filter bar.



3. Adjust the filter options.

4. Save the filter.



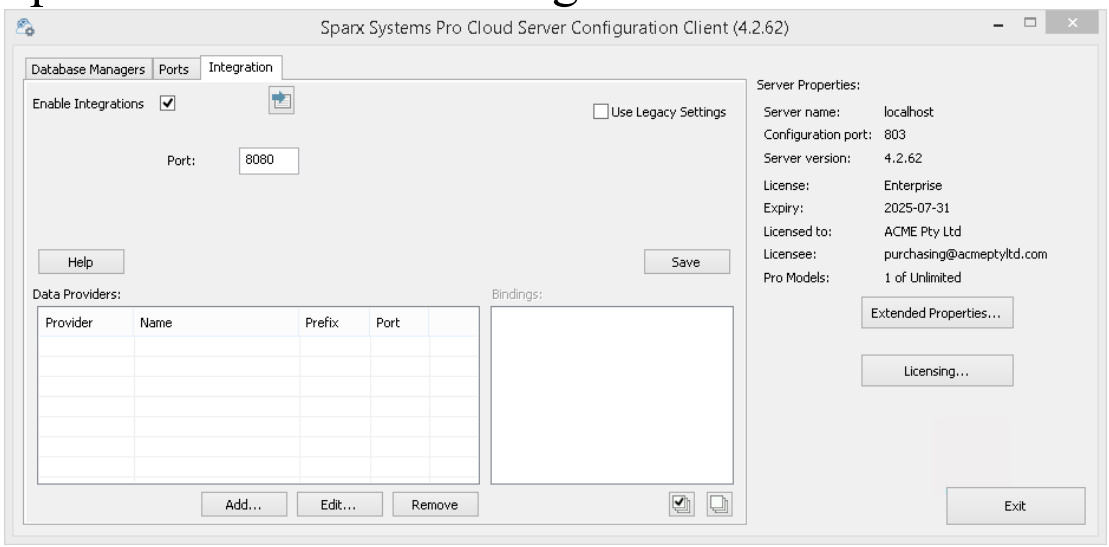
5. This filter (and the items it returns) will now be accessible to the Pro Cloud Server's ServiceNow integration.

Identify ServiceNow to Pro Cloud Server and

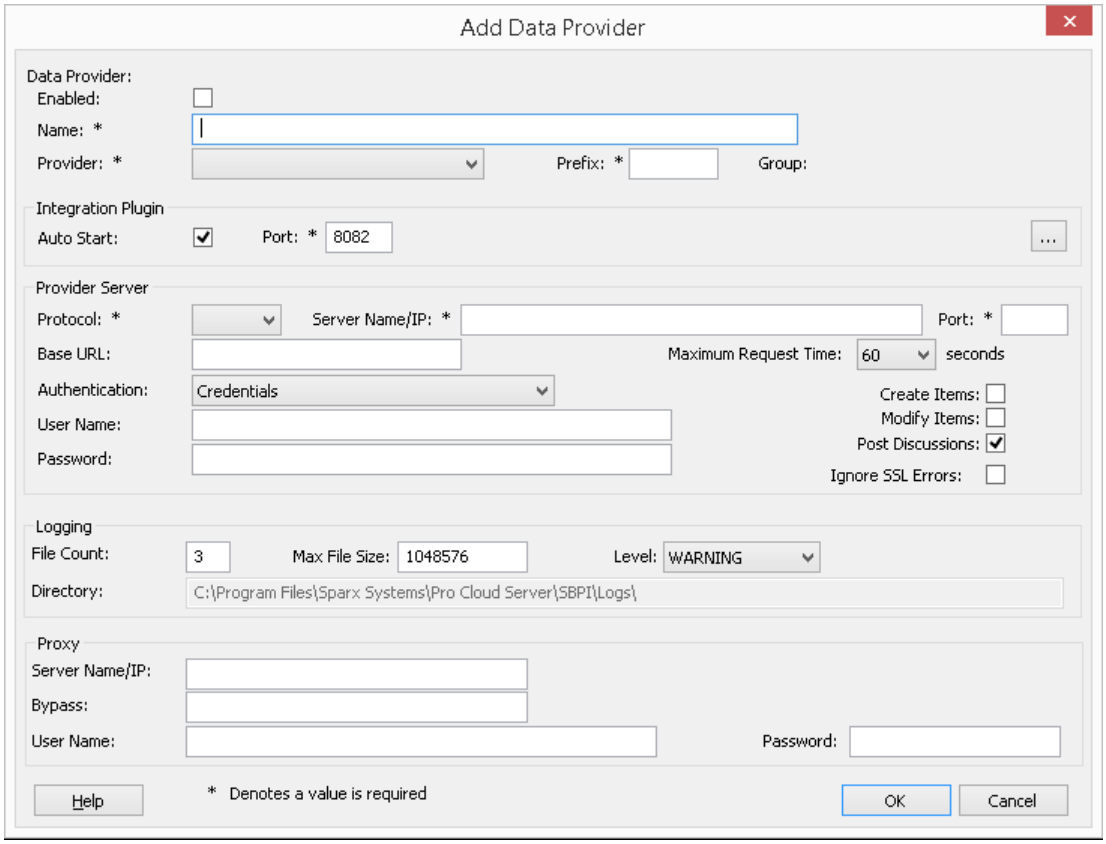
Enterprise Architect

Work through these steps:

1. Double-click on the 'Pro Cloud Config' Desktop icon to open the Pro Cloud Config Client window.

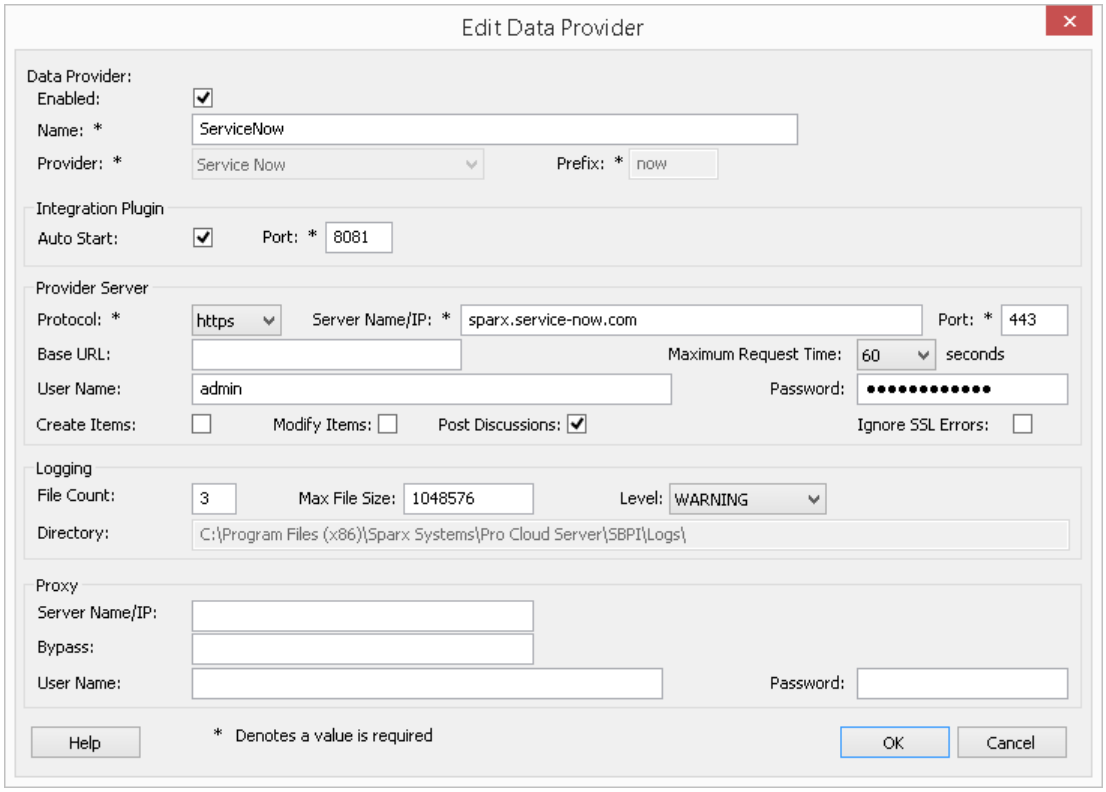


2. Select the 'Integration' tab and select the 'Enable Integrations' checkbox. In the 'Port' field type the number of the Port the Integration Server is listening on for requests from the Pro Cloud Server.
3. Click on the Add button. The 'Add Data Provider' dialog displays.



4. Select the 'Enabled' checkbox.
5. In the 'Name' field, type an appropriate connection name, such as 'ServiceNow'.
6. In the 'Provider' field, click on the drop-down arrow and select 'Service Now'; this automatically adds 'now' to the 'Prefix' field as well.
7. In the 'Integration Plugin' panel, in the 'Port' field, type the number of the Port that the Integration Server will use to communicate with the ServiceNow Plug-in.
8. In the 'Protocol' field, click on the drop-down arrow and select 'https'.
9. In the 'Server Name/IP' field, type the server name or IP address that the ServiceNow Plug-in will use to communicate with ServiceNow.
10. In the 'Port' field enter '443'.

11. Leave the 'Base URL' field blank for a default ServiceNow configuration. The field might be required for non-default ServiceNow Server configurations. Note that the fields in steps 8 to 11 are concatenated to make a web address; that is: *<protocol>://<server>:<port>/baseURL* (*baseURL* included if the field is not blank).
12. If you leave the 'User' and 'Password' fields blank, then each Enterprise Architect user will be prompted for their personal ServiceNow credentials, which can give a better usability.
13. If you prefer to set values in the 'User Name' and 'Password' fields, the values will be used in combination to access the data within ServiceNow. All current Enterprise Architect users of the ServiceNow installation will use the same set of credentials to read the external data.
14. Select the 'Create Items' and/or 'Modify Items' checkboxes as necessary, to allow users of the ServiceNow Plug-in to create and/or update items in ServiceNow using Enterprise Architect.
15. Leave the 'Ignore SSL Errors' option unchecked.



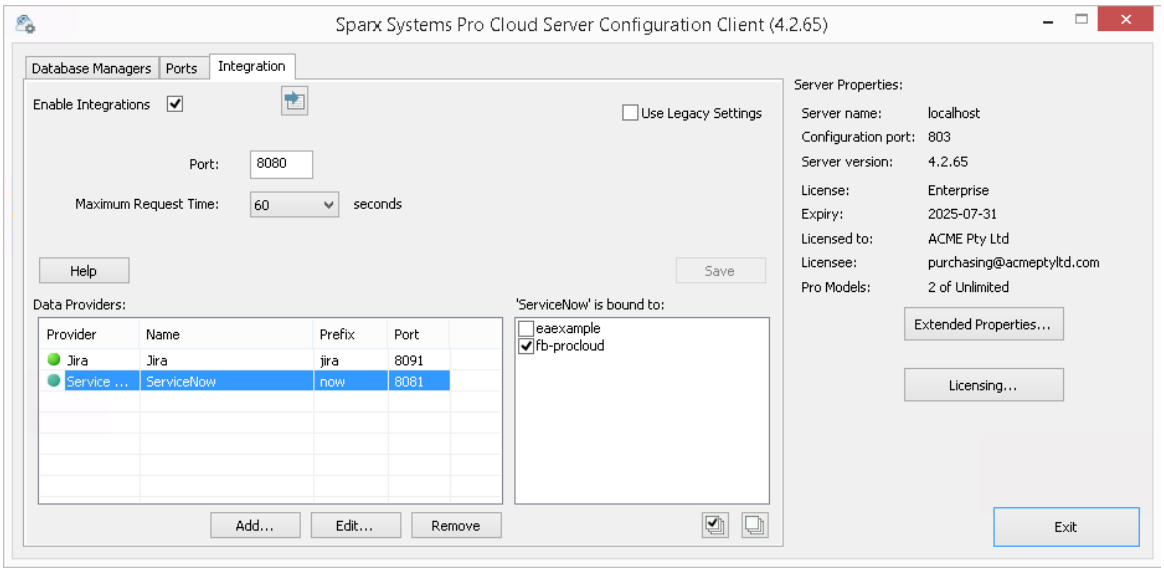
16. You can leave other fields blank or set to their default values. Click on the OK button to complete the configuration. This returns you to the 'Integration' tab of the Pro Cloud Server Configuration Client window.

Identify Enterprise Architect models to ServiceNow

On the 'Integration' tab of the Pro Cloud Server Configuration Client window, you will now see:

- On the lower left of the screen, the 'Data Providers' panel listing ServiceNow as a Data Provider
- On the lower right of the screen, the "ServiceNow' is bound to:' panel displaying a check list of the Enterprise Architect models from your Pro-enabled Database Manager(s)

You can either click on the checkbox against each Enterprise Architect model you want to bind to ServiceNow, or click on the  to select all of them at once.



Click on the Exit button.

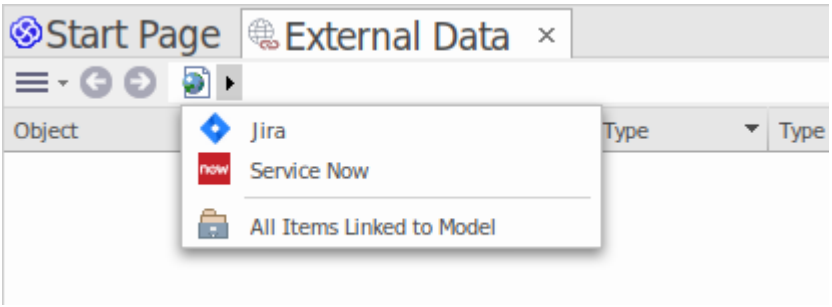
Restart the PCS Service to apply the changes you have made.

You can now go into one of your Enterprise Architect models and check the integration with ServiceNow.

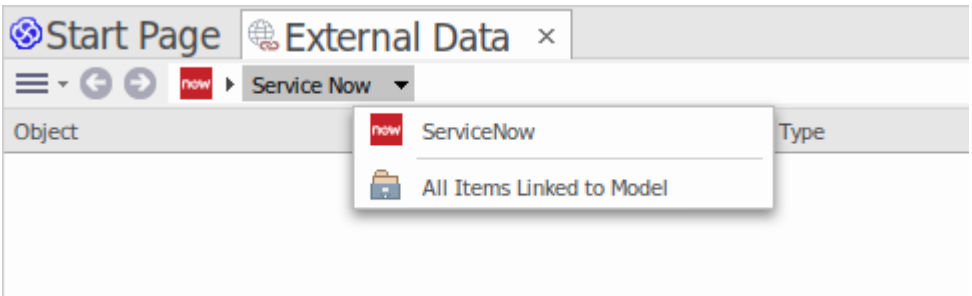
Test the Integration

In Enterprise Architect open one of the models you have identified as being bound to ServiceNow.

1. Select the ribbon option 'Specialize > Tools > System Integration'.
2. In the toolbar of the External Data window, click on the drop-down arrow to the right of the 'globe' icon.



3. Select 'Service Now' from the list; this adds the provider to the breadcrumb trail in the toolbar.
4. Click on 'ServiceNow' in the breadcrumb trail to display a list of ServiceNow providers that have been configured and bound to this model. If nothing is listed then no provider has been bound to this model.
5. Click the Provider that was set up and bound to this model, to display a list of ServiceNow tables that are now available for selection to work with in this Enterprise Architect model.
6. Select the ServiceNow table to display its list of User filters.



If nothing shows here then you might not have sufficient privileges to view the tables. However, if you have not had any User filters created for you the list just contains '[ ]' or an information message.

(If you cannot see ServiceNow or a list of ServiceNow tables, check through the procedures in this topic, or the broader and more detailed topics in the Integration Plug-ins

section to see if there is a step you have missed or a prerequisite that is not satisfied. You can also check the Troubleshooting Help topic.)

Data Mapping

In order for ServiceNow and Enterprise Architect to smoothly exchange data items, you have to review and - where necessary - edit:

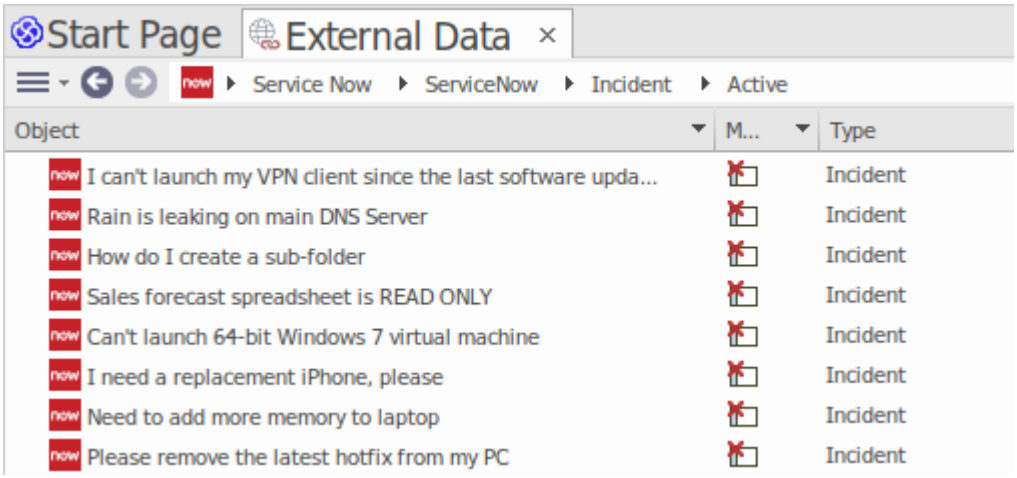
- What type of ServiceNow item corresponds to which type of Enterprise Architect element, and
- What property of a ServiceNow item corresponds to which property of an Enterprise Architect element

These tasks are fully explained in the Configuration Help topic.

Review ServiceNow Data

When you tested the communication between Enterprise Architect and ServiceNow, on the External Data window, you opened the breadcrumb trail to show a list of ServiceNow projects.

When you select one of the projects, you again click on the drop-down arrow and select one of the filters from the list, which then lists the items from that filter in the body of the window.



If this list is too long, you can click on the 'Hamburger' icon in the window toolbar and select options such as:

- 'Linked Items' to show only those ServiceNow items that are linked to Enterprise Architect elements
- 'Items Linked to Current Context' to show only the ServiceNow items that are linked to the currently selected Enterprise Architect element or
- 'Unlinked items' to show only those ServiceNow items that have not yet been linked to Enterprise Architect elements

Select the appropriate option, if necessary, and then explore the content of selected ServiceNow items. Each of the Properties window, Notes window and Inspector window have separate tabs or versions of the window to display any properties, notes and features (respectively) of the selected item. These are illustrated in the External Item Details Help topic.

Working with ServiceNow items and Enterprise Architect elements

Having created a working communication between Enterprise Architect and ServiceNow, you can:

- Create a new element in Enterprise Architect linked to a ServiceNow item
- Create a new ServiceNow item linked to an element in Enterprise Architect
- Link an existing element in Enterprise Architect to a ServiceNow item
- Update the ServiceNow item with changes to the Enterprise Architect element
- Update the Enterprise Architect element with changes to the ServiceNow item
- Update all linked Enterprise Architect elements with any changes to the ServiceNow items

These actions are all explained in the Linking Items Help topic.

Walkthrough: Jira Integration

This walkthrough helps you to set up and use the integration capability of Pro Cloud Server to integrate your Enterprise Architect models with the corresponding Jira Software projects, via the Cloud installation option.

Using the integration with Jira you can:

- Link Enterprise Architect elements with their counterparts in Jira
- Create Enterprise Architect elements based on Jira items, and vice-versa
- Synchronize changes between Enterprise Architect and Jira, as and when needed
- Perform a traceability analysis
- Add comments to Jira items from Enterprise Architect

When Pro Cloud Server is installed on your system, it provides a prompt to also install the optional Integration Plug-ins component; by default, the component is installed into the C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\SBPI\ folder (assuming a 64 bit machine is being used). The Integration Plug-ins component includes the:

- Integration (SBPI) Server (sbpi.exe)
- The plug-in .exe file for each of the many integrated tools, including the one for Jira (JiraSbpi.exe)

The JiraSbpi.exe plug-in interacts with Atlassian's issue tracking system. It is able to list a user's favorite filters (also known as *starred* filters). Each filter will then list all the Jira

items returned by the filter. These filters are configured in Jira in a Favorite folder for each user logging in, prior to Enterprise Architect requesting that the Integration Plug-in should connect to them. You can view only those Jira items that are part of a filter, so create appropriate filters in Jira. For example, if you want to link Enterprise Architect Requirements to Jira User Stories, create a filter that will show the relevant User Stories.

Jira itself has two installation options, Jira Cloud and Jira Server. The Jira Server option is being phased out, so these procedures assume that you are working with Jira Cloud. However, if you are working with Jira Server, there are some comments in the procedures to accommodate that.

Access Tokens

Where Jira is hosted on an Atlassian server and not hosted locally, it requires using a TokenID that is created on the Atlassian site, for a single predefined user set for logging into Atlassian. See the Atlassian web site pages on Access Tokens - a general review (https://www.atlassian.com/software/access/guide/elements/api-token-controls#what-are-api-token-controls) and a discussion on creating tokens (https://confluence.atlassian.com/cloud/api-tokens-938839638.html). See the links under *Learn More* below.

Prerequisites

In order to set up integration between Jira and Enterprise Architect, you must first:

- Be using a licensed installation of Pro Cloud Server with the Integration Plug-ins component, as just discussed
- Select the 'Enable Pro Features (OSLC, WebEA and Integration)' option in the Pro Cloud Server configuration for each Database Manager System hosting your Enterprise Architect models
- Have physical network access between the PCS server and the Jira server hosting the external Jira installation and its data
- Have user credentials to access the external data
- Have a 'Favorite' folder of appropriate filters set up in Jira, as just discussed
- Be using Enterprise Architect Corporate, Unified or Ultimate edition v14 or later

We will assume that:

- The Integration Server and the Jira Integration Plug-in are to run on the same machine as the Pro Cloud Server
- You are defining the Integration Server options using the 'Integration' tab in the Pro Cloud Server Configuration Client rather than by manually manipulating the registry or configuration files or by using the WebConfig interface

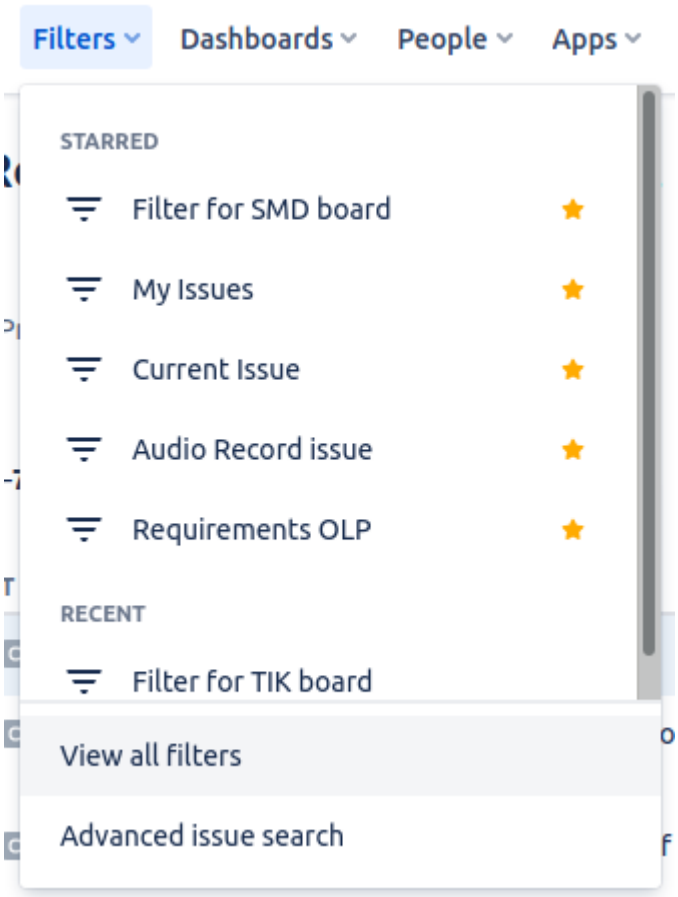
Define starred filters in Jira

The Pro Cloud Server's Jira integration allows access to

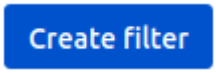
items which are returned by Jira's starred filters. Before using the integration you should ensure some starred filters have been set up to return the items that you want to be accessible in Enterprise Architect.

To define starred filters in Jira, follow these steps:

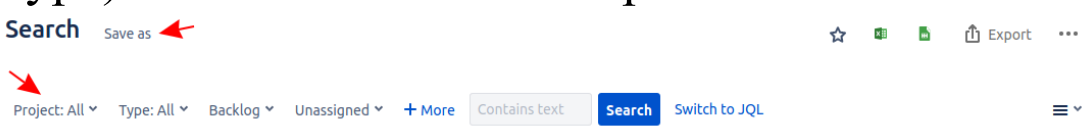
1. Within Jira, select the 'Filters' dropdown menu, then 'View All Filters'.



2. At the top right of the filters list, click on the 'Create Filter' button.



3. Adjust the filter/search settings (such as Project and Type) then use the 'Save as' option to save a new filter.



4. Now return to the list of all filters ('Filters | View All Filters'). In the filter list you can use the star icon to add the filter to your starred filters.

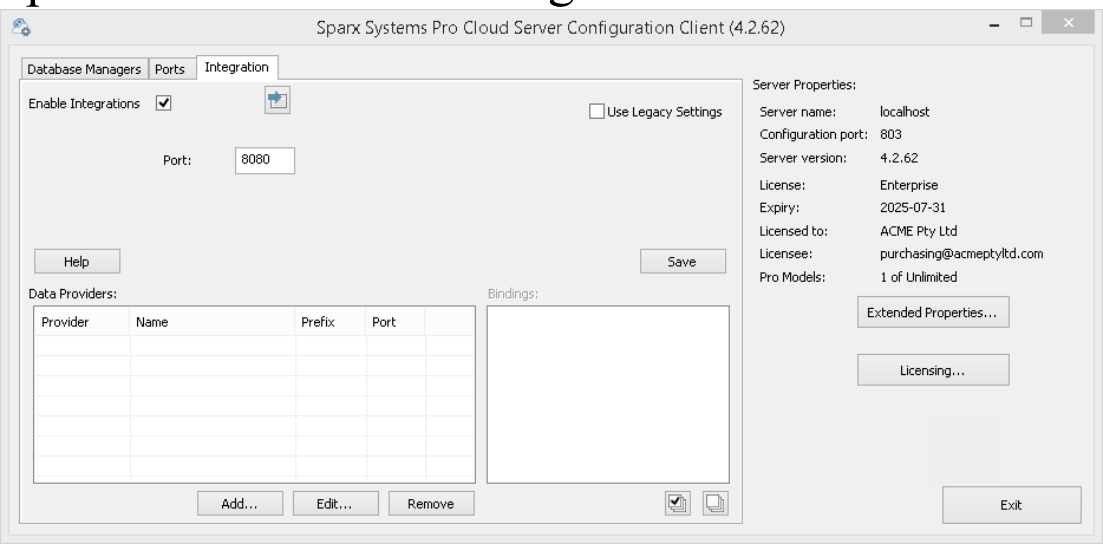


5. This filter (and items it returns) should now be accessible to the Pro Cloud Server's Jira integration

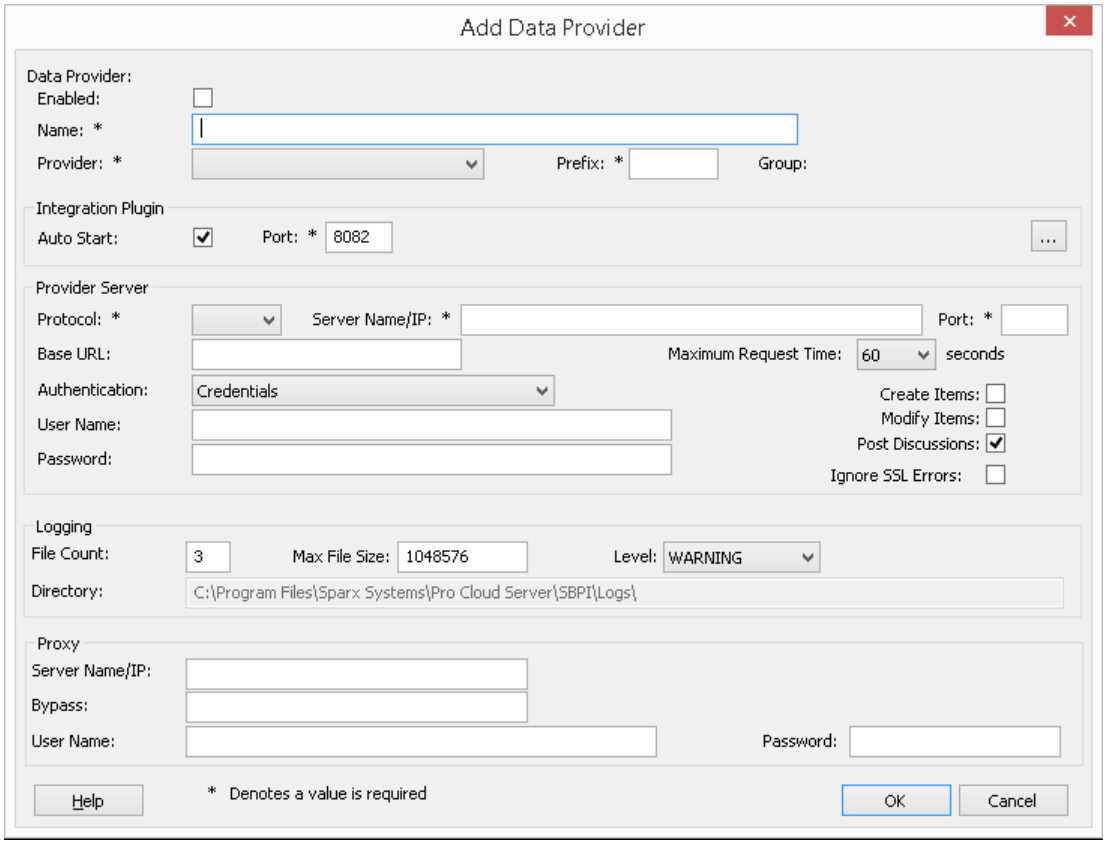
Identify Jira to Pro Cloud Server and Enterprise Architect

Work through these steps:

1. Double-click on the 'Pro Cloud Config' Desktop icon to open the Pro Cloud Config Client window.



2. Select the 'Integration' tab and select the 'Enable Integrations' checkbox. In the 'Port' field type the number of the Port the Integration Server is listening on for requests from the Pro Cloud Server.
3. Click on the Add button. The 'Add Data Provider' dialog displays.

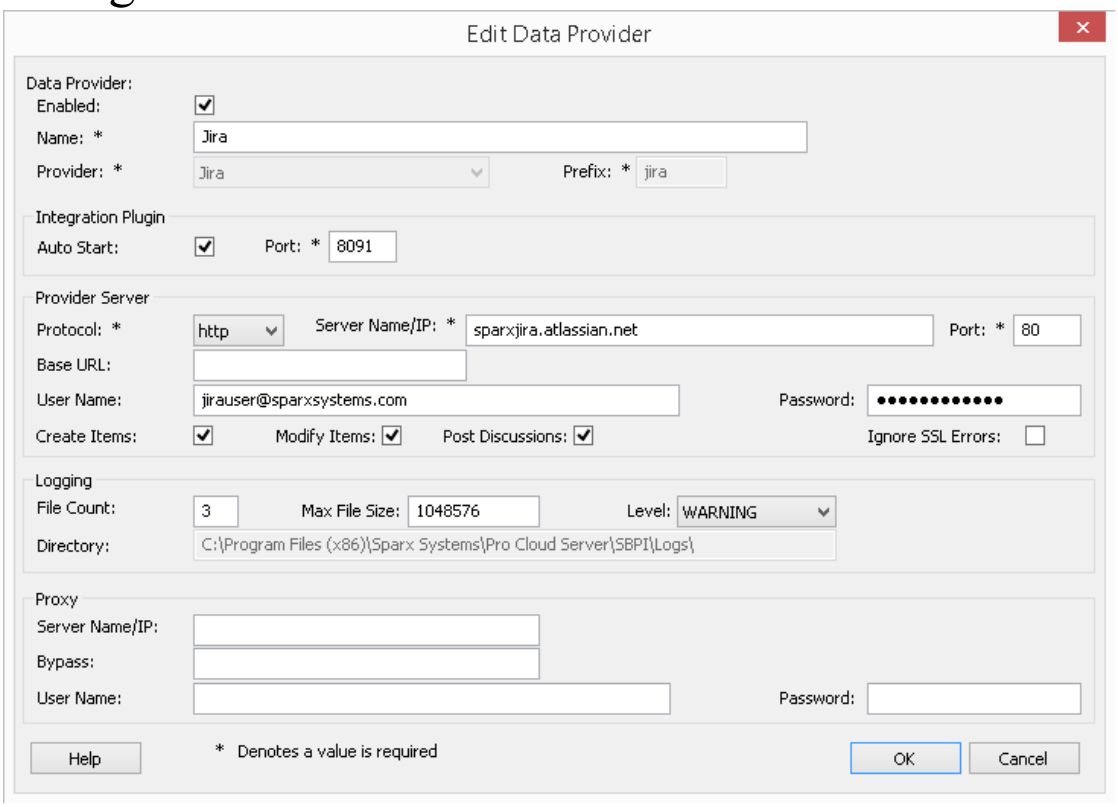


4. Select the 'Enabled' checkbox.
5. In the 'Name' field, type an appropriate connection name, such as 'Jira'.
6. In the 'Provider' field, click on the drop-down arrow and select 'Jira'; this automatically adds 'Jira' to the 'Prefix' field as well.
7. In the 'Integration Plugin' panel, in the 'Port' field, type the number of the Port that the Integration Server will use to communicate with the Jira Plug-in.
8. In the 'Protocol' field, click on the drop-down arrow and select the protocol that the Jira Plug-in will use to communicate with Jira.
    - For a Jira Cloud default installation, this must be 'HTTPS'
    - For a Jira Server installation,the default is 'HTTP'

9. In the 'Server Name/IP' field, type the server name or IP address that the Jira Plug-in will use to communicate with Jira.
   - For a Jira Cloud default installation, this must be <account>.atlassian.net
   - For a Jira Server installation, the default is <name of server or host>
10. In the 'Port' field type the number of the Port that the Jira Plug-in will use to communicate with Jira.
   - For a Jira Cloud default installation, this must be '443'
   - For a Jira Server installation,the default is '8080'
11. Leave the 'Base URL' field blank for Jira Cloud, and for a default Jira Server installation. The field might be required for non-default Jira Server configurations. Note that the fields in steps 8 to 11 are concatenated to make a web address; that is: *<protocol>://<server>:<port>/baseURL* (*baseURL* included if the field is not blank).
12. If you leave the 'User' and 'Password' fields blank, then each Enterprise Architect user will be prompted for their personal Jira credentials, which can give a better usability.
13. If you prefer to set values in the 'User Name' and 'Password' fields, the values will be used in combination to access the data within Jira. All current Enterprise Architect users of the Jira installation will use the same set of credentials to read the external data.
14. Select the 'Create Items', 'Modify Items' and/or 'Post Discussions' checkboxes as necessary, to allow users of

the Jira Plug-in to create and/or update items and/or create Discussion posts in Jira using Enterprise Architect.

15. If you want the Jira Plug-in to ignore SSL-related errors that occur as a result of communicating with Jira, select the 'Ignore SSL Errors' checkbox.



16. You can leave other fields blank or set to their default values. Click on the OK button to complete the configuration. This returns you to the 'Integration' tab of the Pro Cloud Server Configuration Client window.

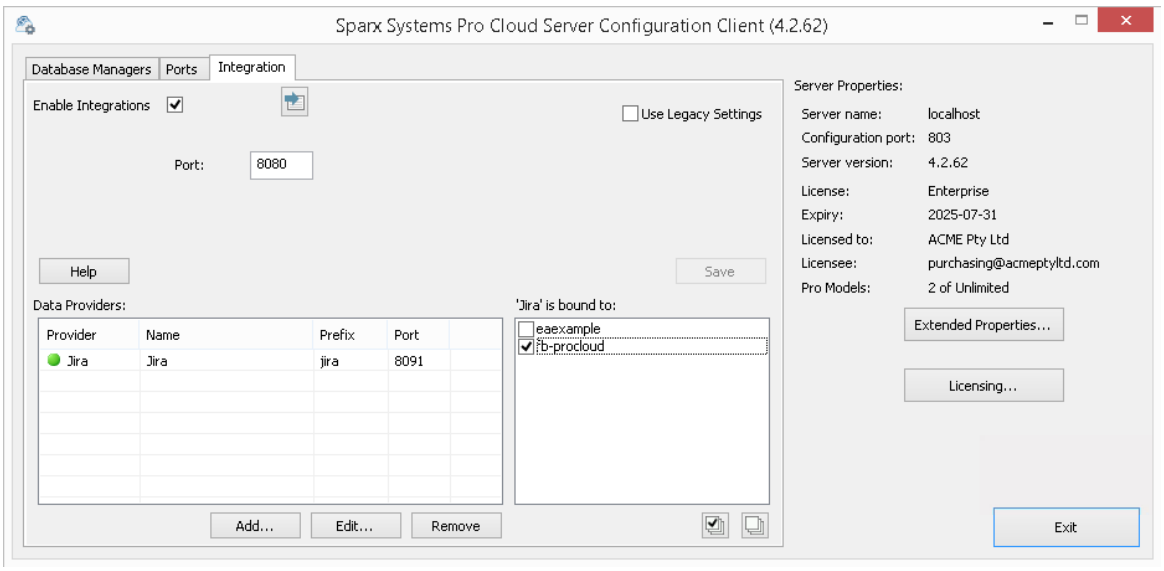
Identify Enterprise Architect models to Jira

On the 'Integration' tab of the Pro Cloud Server Configuration Client window, you will now see:

- On the lower left of the screen, the 'Data Providers' panel listing Jira as a data Provider

- On the lower right of the screen, the "Jira' is bound to:' panel displaying a check list of the Enterprise Architect models from your Pro-enabled Database Manager(s)

You can either click on the checkbox against each Enterprise Architect model you want to bind to Jira, or click on the  to select all of them at once.



Click on the Exit button.

Restart the PCS Service to apply the changes you have made.

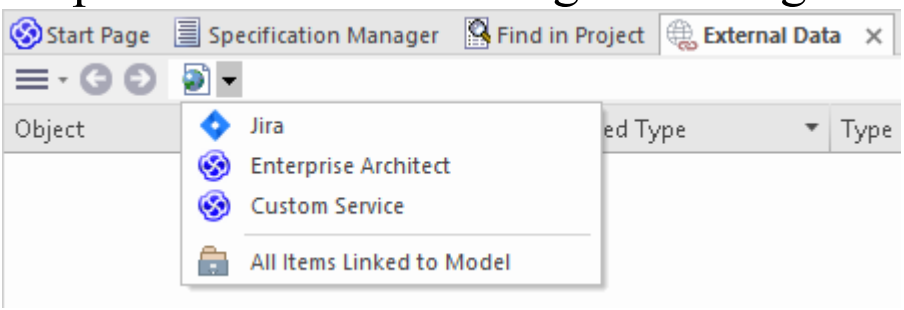
You can now go into one of your Enterprise Architect models and check the integration with Jira.

Test the Integration

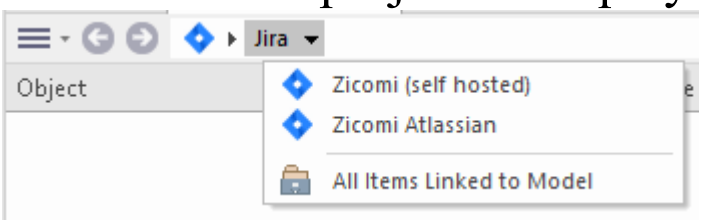
In Enterprise Architect open one of the models you have selected in the list: *'Jira' is bound to. S*ee the image above.

1. Select the ribbon option Specialize > Tools > System Integration.
2. In the toolbar of the External Data window, click on the

drop-down arrow to the right of the 'globe' icon.



3. Select 'Jira' from the list; this adds the provider to the breadcrumb trail in the toolbar.
4. Click on 'Jira' in the breadcrumb trail to display a list of Jira providers that have been configured and bound to this model. If nothing is listed then no provider has been bound to this model.
5. Click the Provider that was set up and bound to this model, to display a list of Jira projects that are now available for selection to work with in this Enterprise Architect model.
6. Select the Jira project to display a list of User filters.



If nothing shows here then you might not have sufficient privileges to view the projects. However, if you have not had any User filters created for you the list just contains '[ ]' or an information message.

(If you cannot see Jira or a list of Jira projects, check through the procedures in this topic, or the broader and more detailed topics in the Integration Plug-ins section to see if there is a step you have missed or a prerequisite that is not

satisfied. You can also check the Troubleshooting Help topic.)

Data Mapping

In order for Jira and Enterprise Architect to smoothly exchange data items, you have to review and - where necessary - edit:

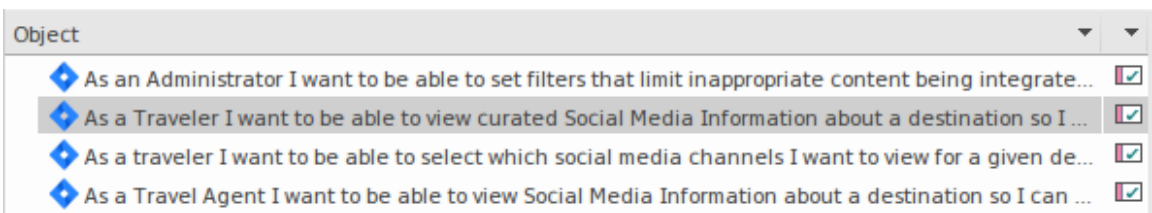
- What type of Jira item corresponds to which type of Enterprise Architect element, and
- What property of a Jira item corresponds to which property of an Enterprise Architect element.

These tasks are fully explained in the Configuration Help topic, which uses Jira as the example Data Provider.

Review Jira Data

When you tested the communication between Enterprise Architect and Jira, on the External Data window, you opened the breadcrumb trail to show a list of Jira projects.

When you select one of the projects, you again click on the drop-down arrow and select one of the filters from the list, which then lists the items from that filter in the body of the window.



If this list is too long, you can click on the 'Hamburger' icon in the window toolbar and select options such as:

- 'Linked Items' to show only those Jira items that are linked to Enterprise Architect elements
- 'Items Linked to Current Context' to show only the Jira items that are linked to the currently selected Enterprise Architect element or
- 'Unlinked items' to show only those Jira items that have not yet been linked to Enterprise Architect elements

Select the appropriate option, if necessary, and then explore the content of selected Jira items. Each of the Properties window, Notes window, Inspector window and Discuss tab of the Discuss & Review window have separate tabs or versions of the window to display any properties, notes, features and discussions (respectively) of the selected item. These are illustrated in the External Item Details Help topic.

Working with Jira items and Enterprise Architect elements

Having created a working communication between Enterprise Architect and Jira, you can:

- Create a new element in Enterprise Architect linked to a Jira item
- Create a new Jira item linked to an element in Enterprise Architect
- Link an existing element in Enterprise Architect to a Jira

item

- Update the Jira item with changes to the Enterprise Architect element
- Update the Enterprise Architect element with changes to the Jira item
- Update all linked Enterprise Architect elements with any changes to the Jira items

These actions are all explained in the Linking Items Help topic, which uses Jira as the example Data Provider.

Walkthrough: DevOps Integration

This walkthrough helps you to set up and use the Pro Cloud Server's DevOps/Team Foundation Server integration.

Note, in 2019 Microsoft's Team Foundation Server was renamed to Azure DevOps. More specifically, the cloud hosted Visual Studio Team Services (VSTS) was renamed to Azure DevOps Services, and on-premises Visual Studio Team Foundation Server (TFS) was renamed to Azure DevOps Server.

The Pro Cloud Server integration is compatible with Azure DevOps (both cloud hosted and on-premises) and Team Foundation Server (TFS). This walkthrough will primarily use the current name 'DevOps', however within the Pro Cloud Server and Enterprise Architect, the plugin will often be referred to as Team Foundation Server or TFS.

Using the integration with DevOps you can:

- Link Enterprise Architect elements with their counterparts in DevOps
- Create Enterprise Architect elements based on DevOps items
- Synchronize changes between Enterprise Architect and DevOps, as and when needed
- Perform a traceability analysis

Prerequisites

In order to set up integration between DevOps and

Enterprise Architect, you must first:

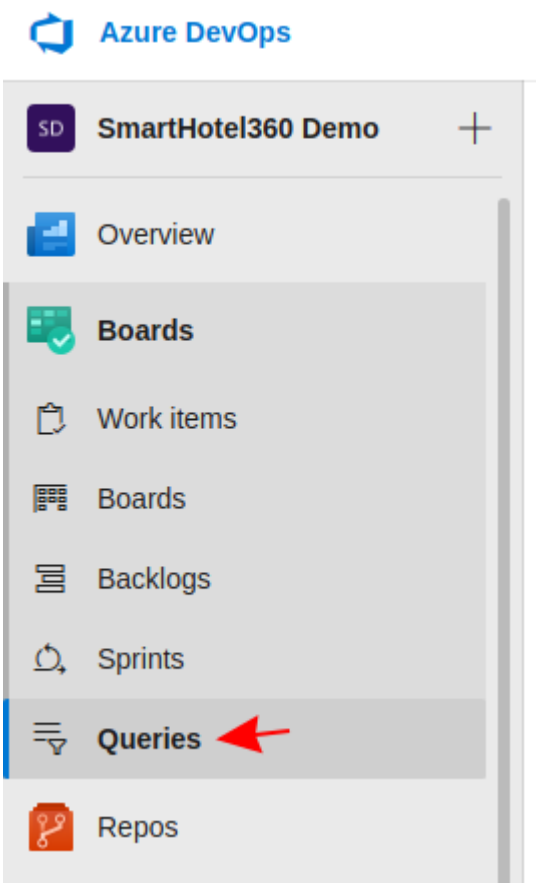
- Be using a licensed installation of Pro Cloud Server with the Integration Plug-ins component
- Select the 'Enable Pro Features (OSLC, WebEA and Integration)' option in the Pro Cloud Server configuration for each Database Manager System hosting your Enterprise Architect models
- Have network access between the PCS server and DevOps
- Have user credentials to access the external data
- Be using Enterprise Architect Corporate, Unified or Ultimate edition v14 or later

We will assume that the Integration Server and the DevOps Integration Plug-in are to run on the same machine as the Pro Cloud Server

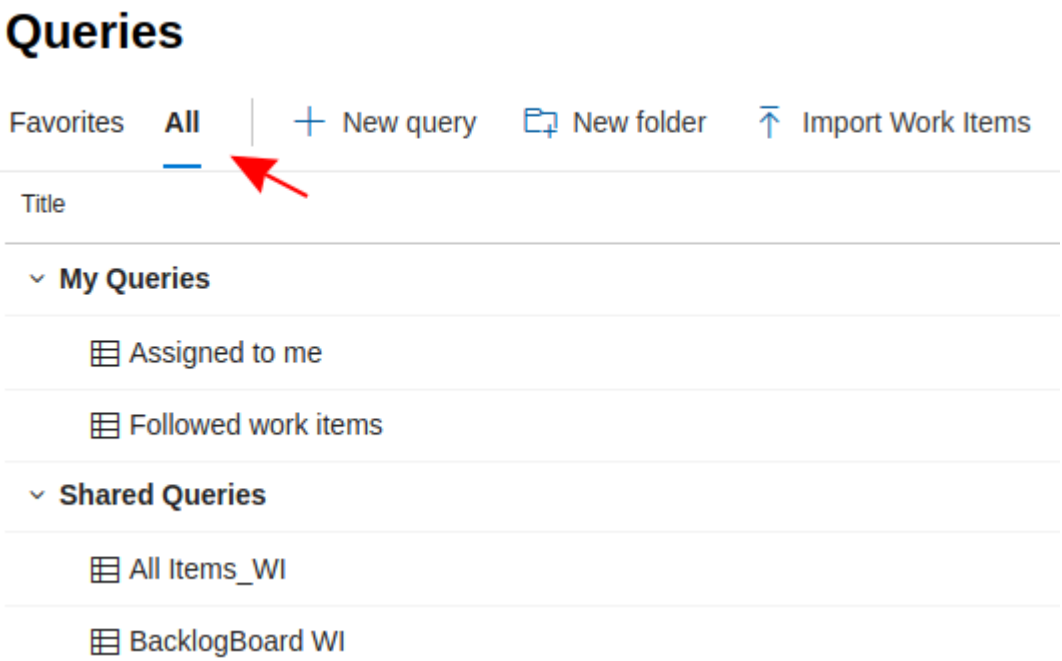
Define DevOps Queries

The DevOps integration allows access to work items via queries which have been defined in DevOps. You can view your existing queries and create new ones as described below.

1. Login to Azure DevOps and select your Project.
2. From the left panel select 'Boards', then 'Queries'.



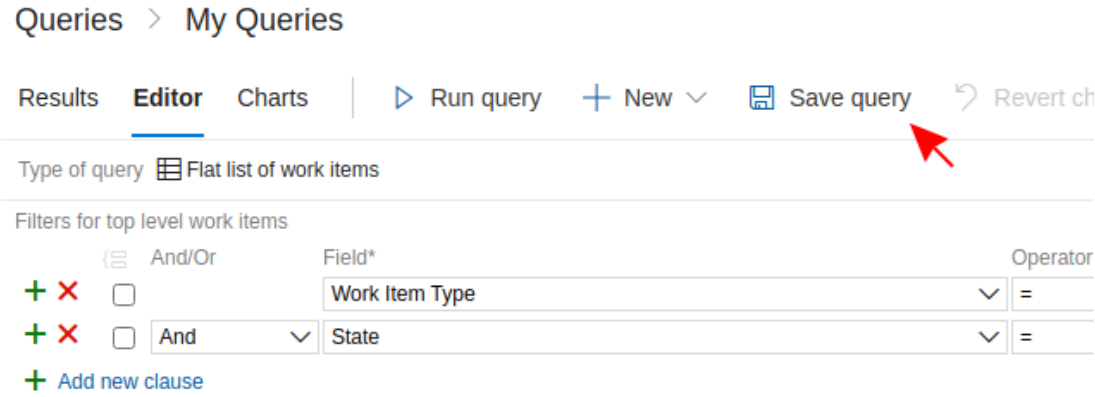
3. To view all queries which will be accessible to the DevOps integration click on 'All'. The Queries will be grouped into 'My Queries' and 'Shared Queries'.



4. A new query can be created using the 'New query' button.



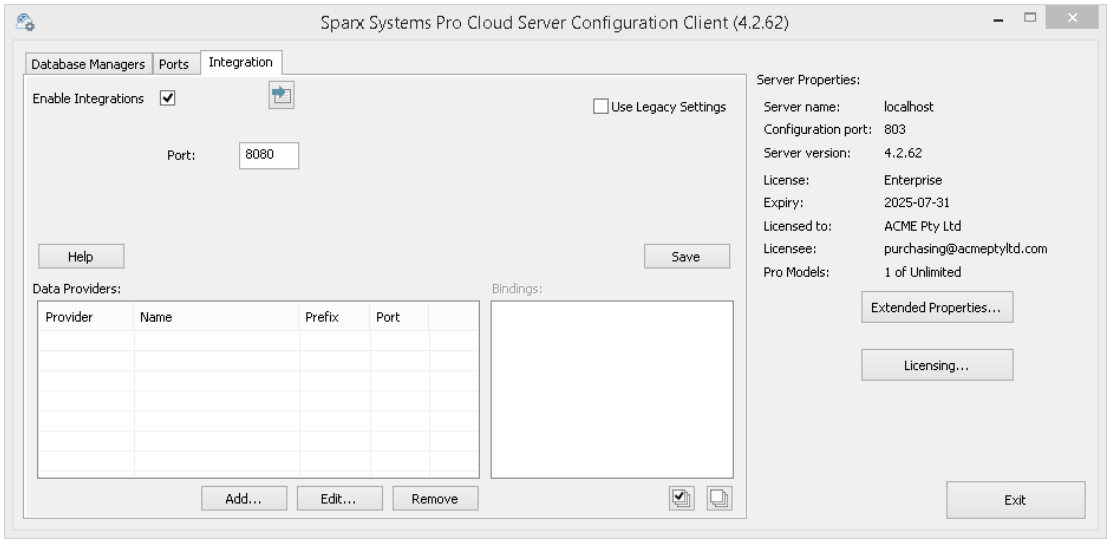
5. After adjusting the query's filter options, click on the 'Save query' button to save the query.



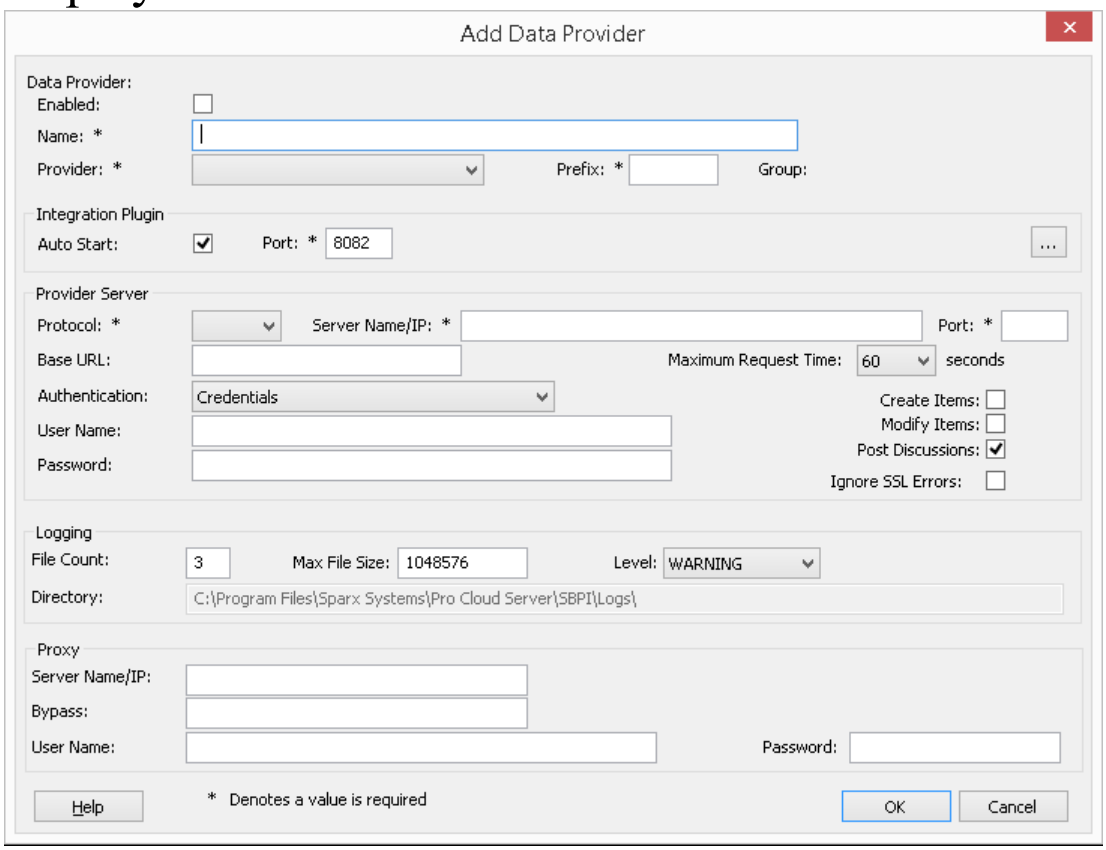
Identify DevOps to Pro Cloud Server and Enterprise Architect

Work through these steps:

1. Double-click on the 'Pro Cloud Config' Desktop icon to open the Pro Cloud Config Client window.
2. Select the 'Integration' tab and select the 'Enable Integrations' checkbox. In the 'Port' field type the number of the Port the Integration Server is listening on for requests from the Pro Cloud Server.



3. Click on the Add button. The 'Add Data Provider' dialog displays.

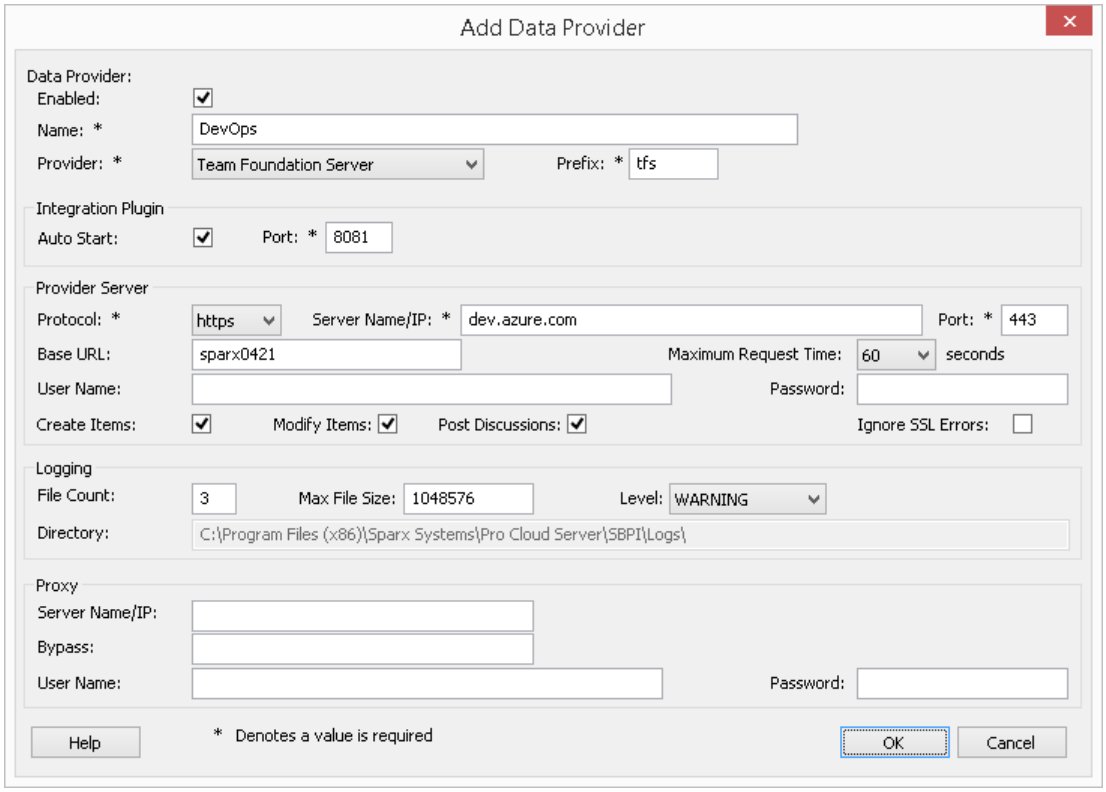


4. Select the 'Enabled' checkbox.
5. In the 'Name' field, type an appropriate connection name, such as 'DevOps'.
6. In the 'Provider' field, click on the drop-down arrow and select 'Team Foundation Server'; this automatically adds

'tfs' to the 'Prefix' field as well.

7. In the 'Integration Plugin' panel, in the 'Port' field, type the number of the Port that the Integration Server will use to communicate with the DevOps Plug-in.
8. If you are using the **Cloud-hosted 'Azure DevOps Services'**, fill in the Provider Server fields as shown:

   Protocol: https
   Server Name/IP: dev.azure.com
   Port: 443
   Base URL: Enter the DevOps organization name. This can be found at the end of your DevOps URL. For example, If your url is https://dev.azure.com/org1/, then in the 'Base URL' field you would enter 'org1'.
   User Name: Leave this field empty. Users will be prompted for credentials when using the integration within Enterprise Architect.
   Password: Leave this field empty.
   Create Items / Modify Items / Post Discussions: Enable these checkboxes as necessary, to allow users of the DevOps Plug-in to create and/or update items and/or create Discussion posts in DevOps using Enterprise Architect.
   Ignore SSL Errors: Leave this option unchecked.



If you are using an **on-premises 'Azure DevOps Server' / TFS**, fill in the Provider Server fields as shown:

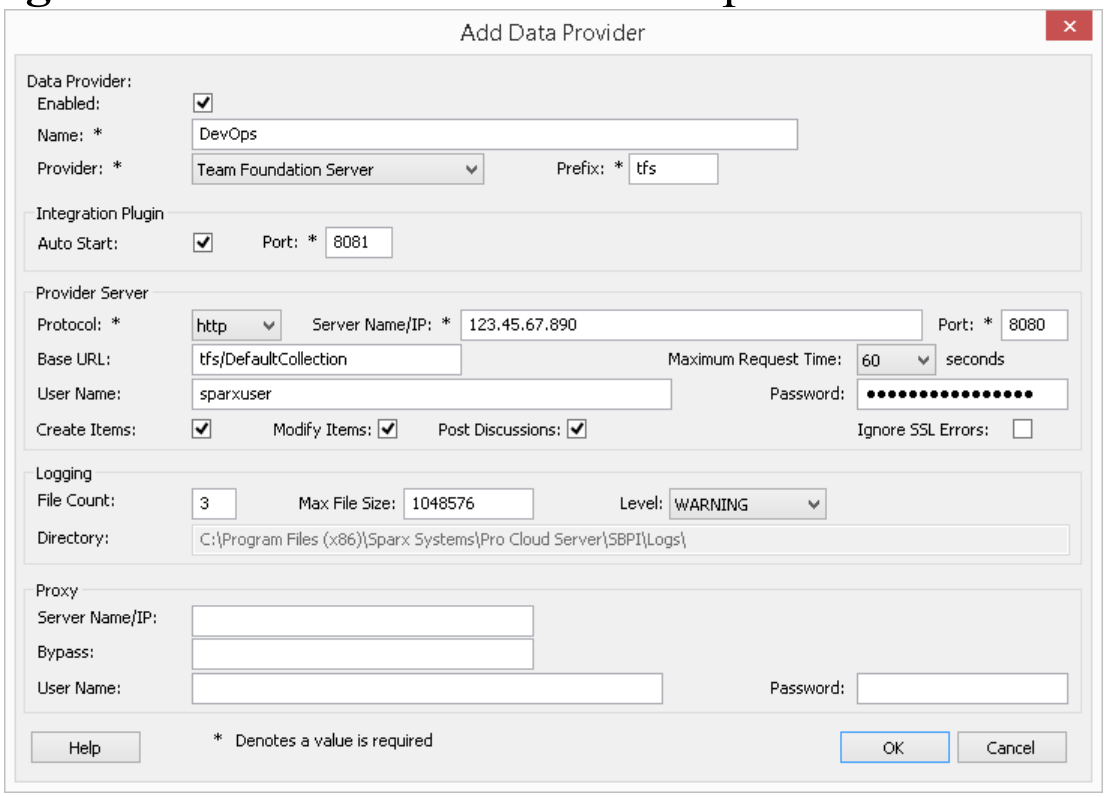
Protocol: Select http or https
Server Name/IP: Enter your server name or IP address
Port: Enter the Port number that DevOps/TFS is configured to use.
Base URL: Enter the final component of your DevOps/TFS url (everything after the server name/Port). E.g. tfs/DefaultCollection
User Name: Leave this field empty. Users will be prompted for their user name when using the integration within Enterprise Architect.
Password: Leave this field empty. Users will be prompted for a password when using the integration

within Enterprise Architect. In this case the 'password' refers a personal access token (PAT). Personal access tokens are created from within DevOps via the 'Settings | Personal access tokens' option. Refer to the DevOps documentation for details.

If you prefer to set values in the 'User Name' and 'Password' fields, the values will be used in combination to access the data within DevOps; however, all current Enterprise Architect users of the DevOps installation will use the same set of credentials to read the external data.

Create Items / Modify Items / Post Discussions: Enable these checkboxes as necessary, to allow users of the DevOps Plug-in to create and/or update items and/or create Discussion posts in DevOps using Enterprise Architect.

Ignore SSL Errors: Leave this option unchecked.



Note, the Protocol, Server, Port and Base URL fields are concatenated to make a web address; that is: *<protocol>://<server>:<port>/baseURL* (*baseURL* included if the field is not blank).

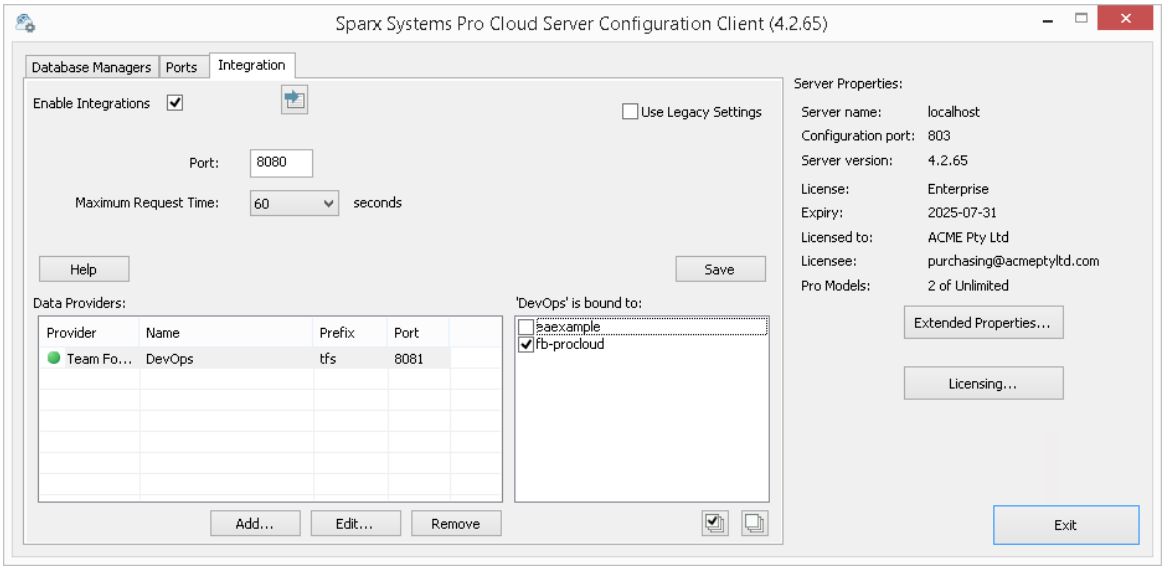
9. You can leave other fields blank or set to their default values. Click on the OK button to complete the configuration. This returns you to the 'Integration' tab of the Pro Cloud Server Configuration Client window.

Identify Enterprise Architect models to DevOps

On the 'Integration' tab of the Pro Cloud Server Configuration Client window, you will now see:

- On the lower left of the screen, the 'Data Providers' panel listing Team Foundation Server/DevOps as a data Provider
- On the lower right of the screen, the "DevOps' is bound to:' panel displaying a check list of the Enterprise Architect models from your Pro-enabled Database Manager(s)

You can either click on the checkbox against each Enterprise Architect model you want to bind to DevOps, or click on the  to select all of them at once.



Click on the Exit button.

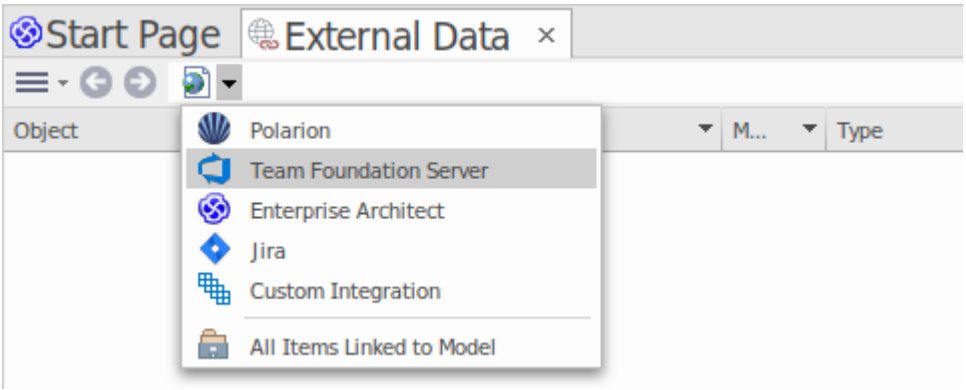
Restart the PCS Service to apply the changes you have made.

You can now go into one of your Enterprise Architect models and check the integration with DevOps.

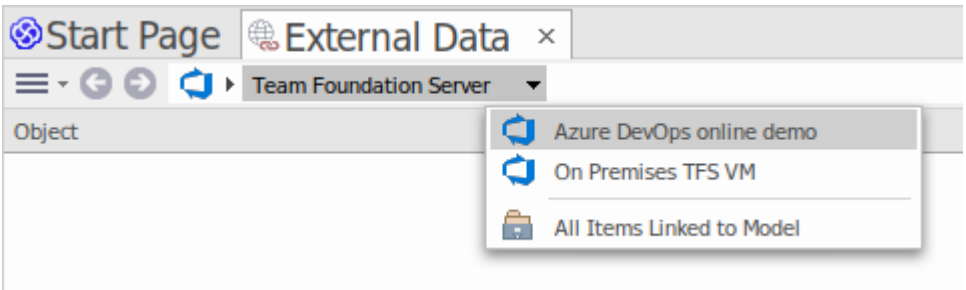
Test the Integration

In Enterprise Architect open one of the models you have identified as being bound to DevOps.

1. Select the ribbon option Specialize > Tools > System Integration.
2. In the toolbar of the External Data window, click on the drop-down arrow to the right of the 'globe' icon.
3. Select 'Team Foundation Server' from the list; this adds the provider to the breadcrumb trail in the toolbar.



4. Click on 'DevOps' in the breadcrumb trail to display a list of DevOps providers that have been configured and bound to this model. If nothing is listed then no provider has been bound to this model.



5. Click on one of the Projects which are defined in DevOps.
6. Select either 'Shared Queries' or 'My queries'
7. Select one of the defined queries to display all the items returned by that query in the list below.

(If you cannot see Team Foundation Server, the DevOps project/s or any defined queries, check through the procedures in this topic, or the broader and more detailed topics in the Integration Plug-ins section to see if there is a step you have missed or a prerequisite that is not satisfied. You can also check the Troubleshooting Help topic.)

Data Mapping

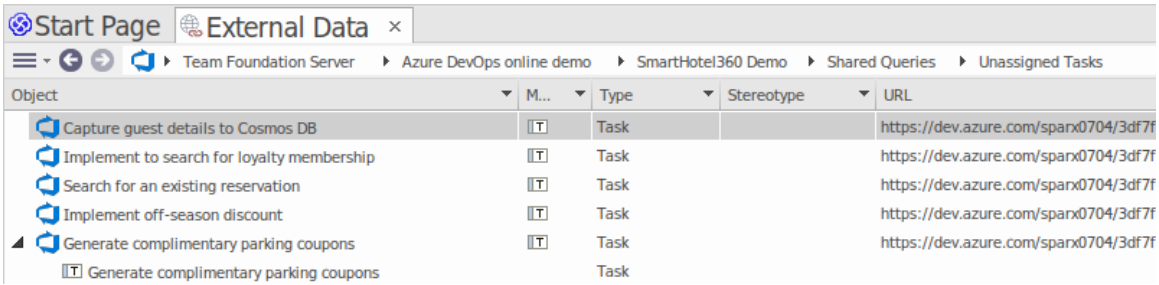
In order for DevOps and Enterprise Architect to smoothly exchange data items, you have to review and - where necessary - edit:

- What type of DevOps item corresponds to which type of Enterprise Architect element, and
- What property of a DevOps item corresponds to which property of an Enterprise Architect element.

These tasks are fully explained in the Configuration Help topic.

Review DevOps Data

When you tested the communication between Enterprise Architect and DevOps, on the External Data window, you opened the breadcrumb trail to show a list of DevOps projects, then selected either 'My Queries' or 'Shared Queries' and selected a query from the list. The work items are then listed in the body of the window.



If this list is too long, you can click on the 'Hamburger' icon in the window toolbar and select options such as:

- 'Linked Items' to show only those DevOps items that are linked to Enterprise Architect elements
- 'Items Linked to Current Context' to show only the

DevOps items that are linked to the currently selected Enterprise Architect element or

- 'Unlinked items' to show only those DevOps items that have not yet been linked to Enterprise Architect elements

Select the appropriate option, if necessary, and then explore the content of selected DevOps items. Each of the Properties window, Notes window and Inspector window have separate tabs or versions of the window to display any properties, notes and features (respectively) of the selected item. These are illustrated in the External Item Details Help topic.

Working with DevOps items and Enterprise Architect elements

Having created a working communication between Enterprise Architect and DevOps, you can:

- Create a new element in Enterprise Architect linked to a DevOps item
- Create a new DevOps item linked to an element in Enterprise Architect
- Link an existing element in Enterprise Architect to a DevOps item
- Update the DevOps item with changes to the Enterprise Architect element
- Update the Enterprise Architect element with changes to the DevOps item
- Update all linked Enterprise Architect elements with any

changes to the DevOps items

These actions are all explained in the Linking Items Help topic.

Walkthrough: Jama Integration

Prerequisites

In order to set up integration between Jama and Enterprise Architect, you must first:

- Be using a licensed installation of Pro Cloud Server (Version 5.0 or later) with the Integration Plug-ins component
- Select the 'Enable Pro Features (OSLC, WebEA and Integration)' option in the Pro Cloud Server configuration for each Database Manager System hosting your Enterprise Architect models
- Have physical network access between the PCS server and the Jama server
- Have user credentials to access the external data
- Be using Enterprise Architect Corporate, Unified or Ultimate edition v16.0 or later

We will assume that the Integration Server and the Jama Integration Plug-in are to run on the same machine as the Pro Cloud Server

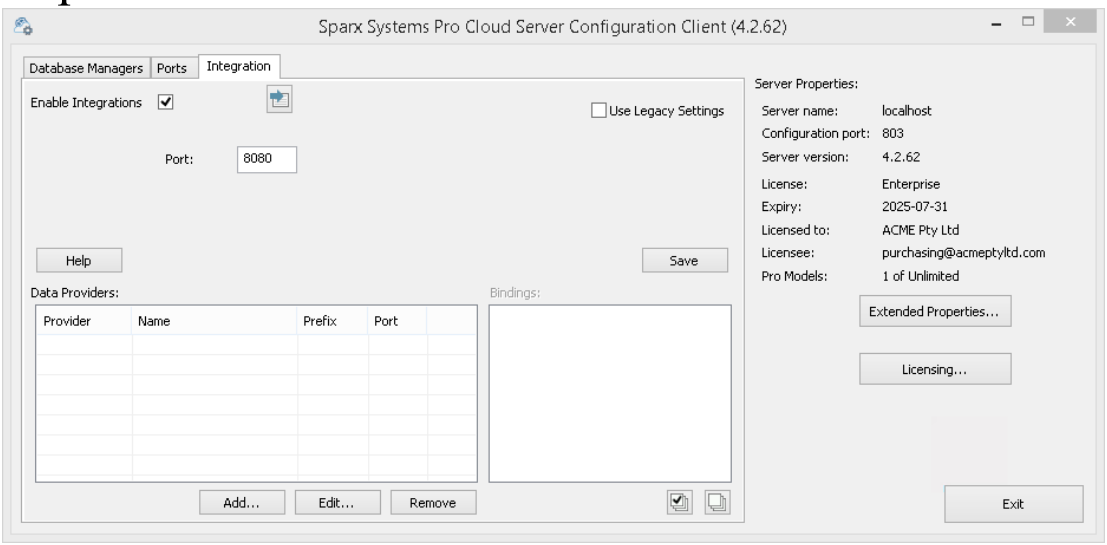
Identify Jama to Pro Cloud Server and Enterprise Architect

Work through these steps:

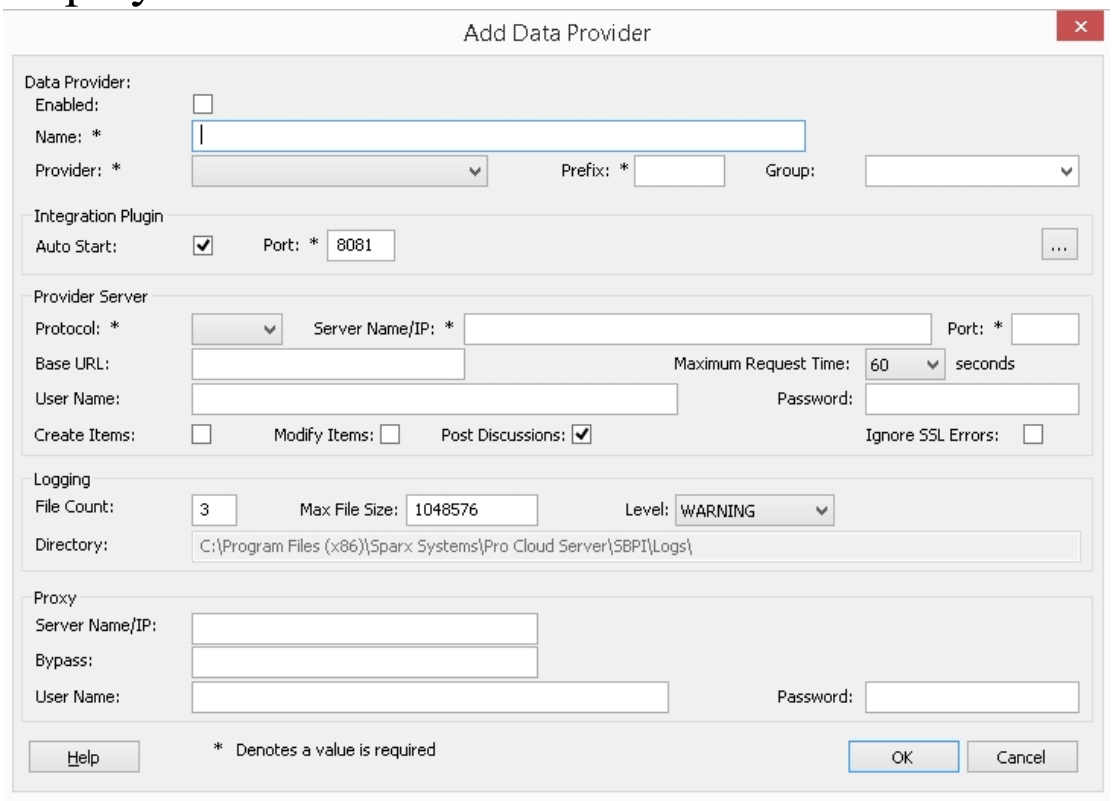
1. Double-click on the 'Pro Cloud Config Client' Desktop

icon to open the Pro Cloud Config Client window.

2. Select the 'Integration' tab and select the 'Enable Integrations' checkbox. In the 'Port' field type the number of the Port the Integration Server will listen on for requests from the Pro Cloud Server.



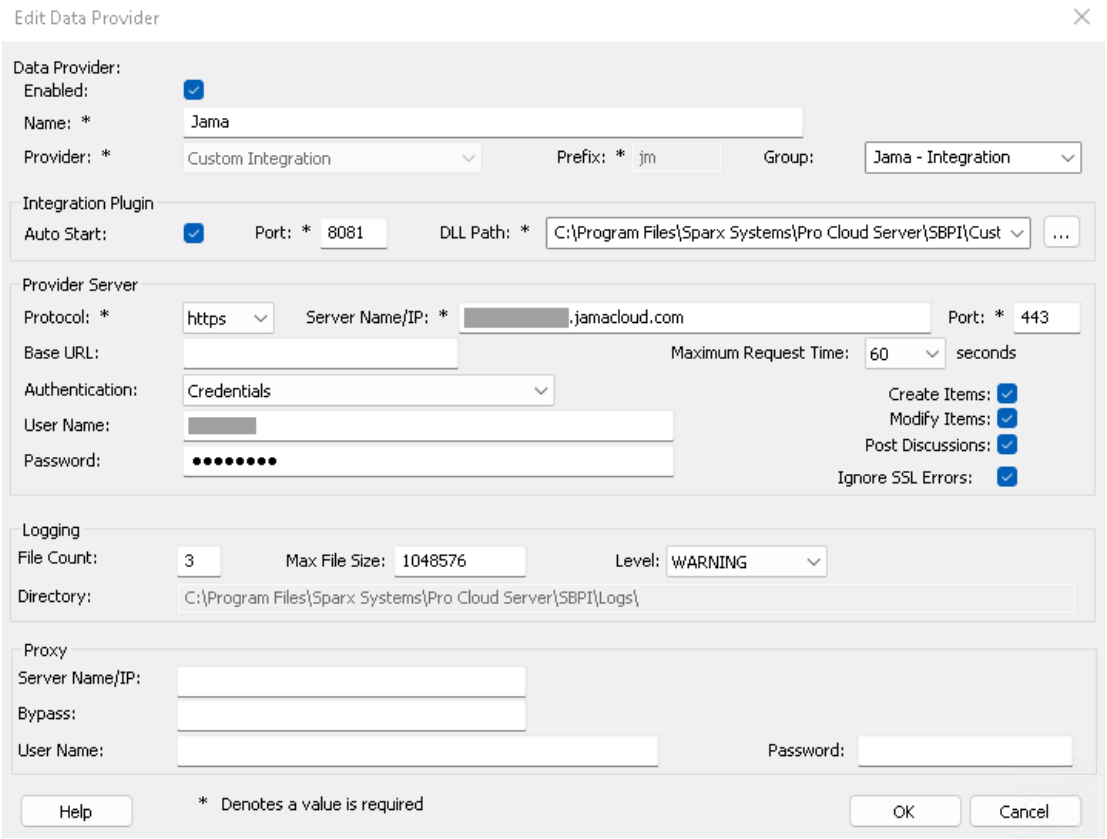
3. Click on the Add button. The 'Add Data Provider' dialog displays.



4. Select the 'Enabled' checkbox.

5. In the 'Name' field, type an appropriate connection name, such as 'Jama'.
6. In the 'Provider' field, click on the drop-down arrow and select 'Custom Integration'.
7. In the 'Prefix' field, enter an appropriate prefix, such as 'Jama'.
8. In the 'Group' field enter a name for the root level when browsing the Jama integration (e.g. 'Jama - Integration').
9. In the 'Integration Plugin' panel, in the 'Port' field, type the number of the Port that the Integration Server will use to communicate with the Jama Plug-in.
10. Click on the ellipsis button next to the 'DLL Path' field. Browse and select the 'JamaIntegrationPlugin.dll' file. This is located within the Pro Cloud Server Installation folder. E.g. C:\Program Files\Sparx Systems\Pro Cloud Server\SBPI\JamaIntegrationPlugin.dll
11. The fields in steps 11 to 14 make up the components of your Jama web address; that is:
<protocol>://<server>:<port>/baseURL
In the 'Protocol' field, click on the drop-down arrow and select the protocol that the Jama Plug-in will use to communicate with Jama.
12. In the 'Server Name/IP' field, type the server name or IP address that the Jama Plug-in will use to communicate with Jama.
13. In the 'Port' field type the number of the Port that the Jama Plug-in will use to communicate with Jama.
14. The 'Base URL' field can be left empty.

15. Set values in the 'User Name' and 'Password' fields, the values will be used in combination to access the data within Jama. All current Enterprise Architect users of the Jama installation will use the same set of credentials to read the external data.
16. Select the 'Create Items', 'Modify Items' and/or 'Post Discussions' checkboxes as necessary, to allow users of the Jama Plug-in to create and/or update items and/or create Discussion posts in Jama using Enterprise Architect.
17. If you want the Jama Plug-in to ignore SSL-related errors that occur as a result of communicating with Jama (e.g. if you are using a self-signed SSL certificate), select the 'Ignore SSL Errors' checkbox.
18. If your access to Jama uses a Proxy server enter the relevant details in the Proxy section:
    Server Name/IP - The server name (or IP) and Port number of the Proxy Server; for example, proxyserver.com:3131
    Bypass - This field provides a mechanism for certain addresses to bypass the proxy; it accepts multiple values separated by semi-colons (;). For example, 192.168.*;*.localdomain.com;www.mydomain.com
    User Name - If the Proxy Server requires credentials, this field provides the user name.
    Password - If the Proxy Server requires credentials, this field provides the user password.



19. You can leave other fields blank or set to their default values. Click on the OK button to complete the configuration. This returns you to the 'Integration' tab of the Pro Cloud Server Configuration Client window.

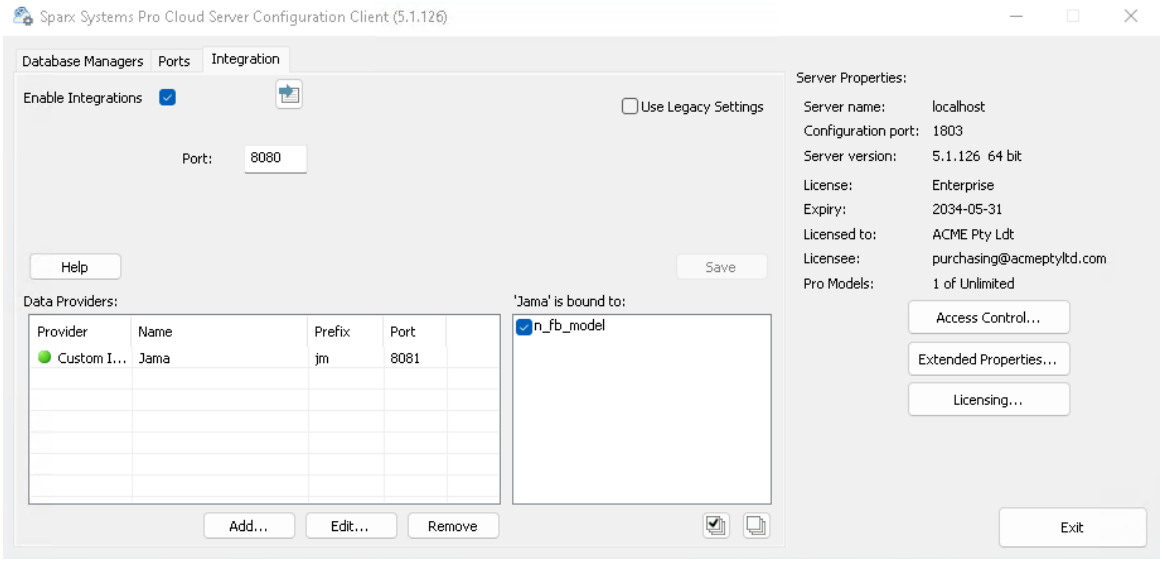
Identify Enterprise Architect models to Jama

On the 'Integration' tab of the Pro Cloud Server Configuration Client window, you will now see:

- On the lower left of the screen, the 'Data Providers' panel listing the Custom Integration for Jama as a data Provider
- On the lower right of the screen, the 'Jama' is bound to:' panel displaying a check list of the Enterprise Architect models from your Pro-enabled Database Manager(s)

You can either click on the checkbox against each

Enterprise Architect model you want to bind to Jama, or click on the  to select all of them at once.



Click on the Exit button.

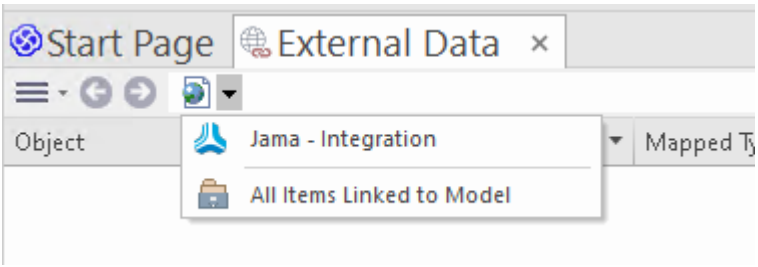
Restart the PCS Service to apply the changes you have made.

You can now go into one of your Enterprise Architect models and check the integration with Jama.

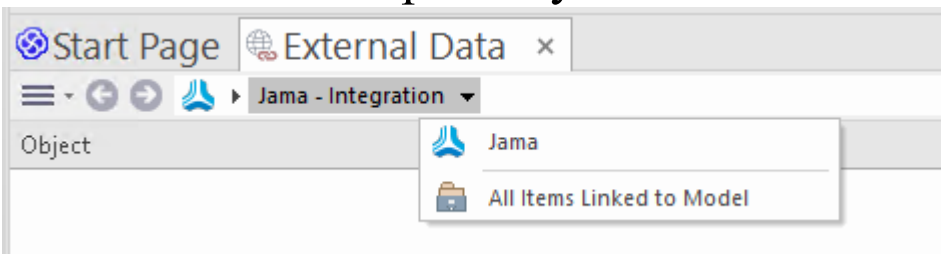
Test the Integration

In Enterprise Architect open one of the models you have identified as being bound to Jama.

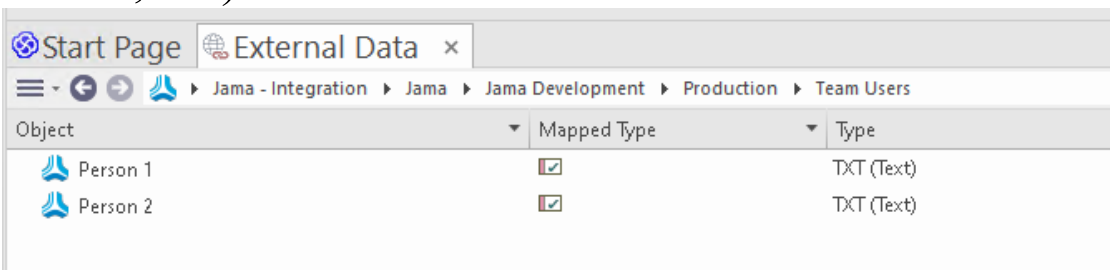
1. Select the ribbon option Specialize > Tools > System Integration.
2. In the toolbar of the External Data window, click on the drop-down arrow to the right of the 'globe' icon.
3. Select the integration group (e.g. 'Jama - Integration' or 'Custom Integration') from the list. This is the 'Group' which you had entered in the 'Add Data Provider' dialog.



4. Click on 'Jama' in the breadcrumb trail, this represents the root of the Jama repository.



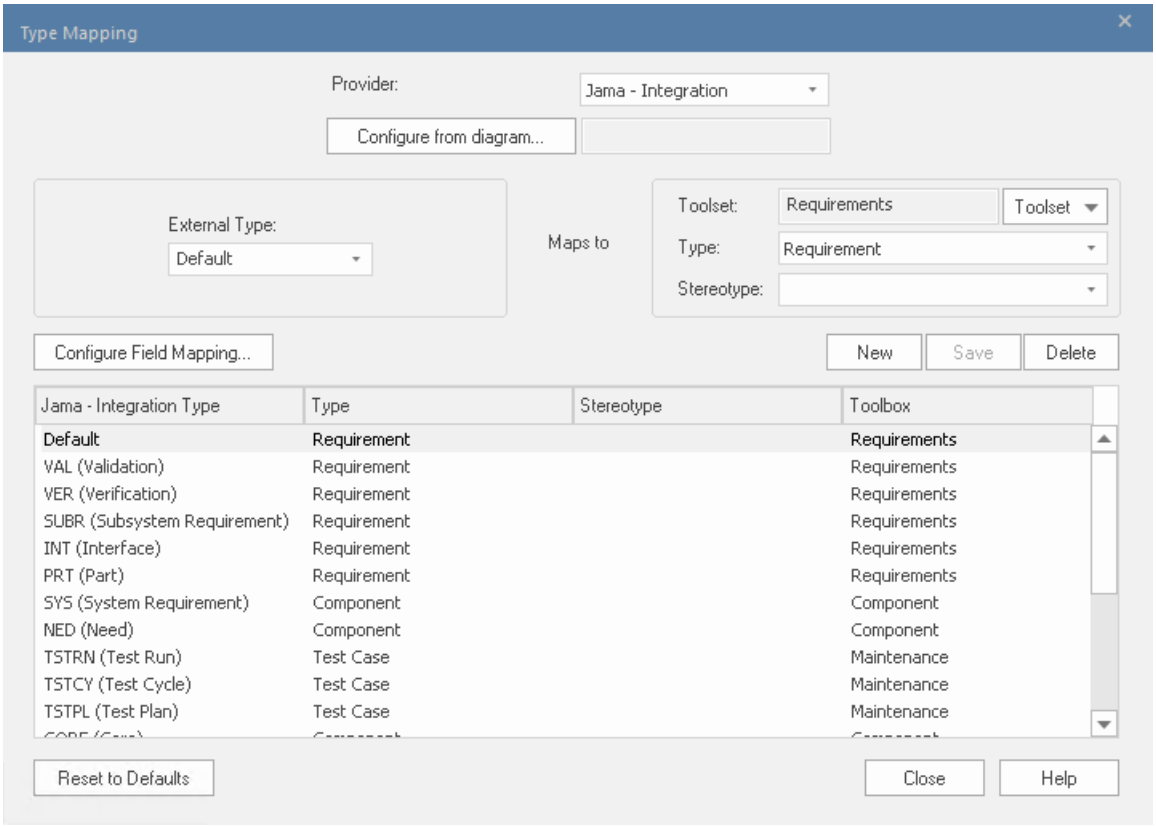
5. Continue using the breadcrumb options to drill down into the Jama projects and access Work Items (Requirements, Issues, etc).



Data Mapping

In order for Jama and Enterprise Architect to smoothly exchange data items, you should review and - if necessary - edit:

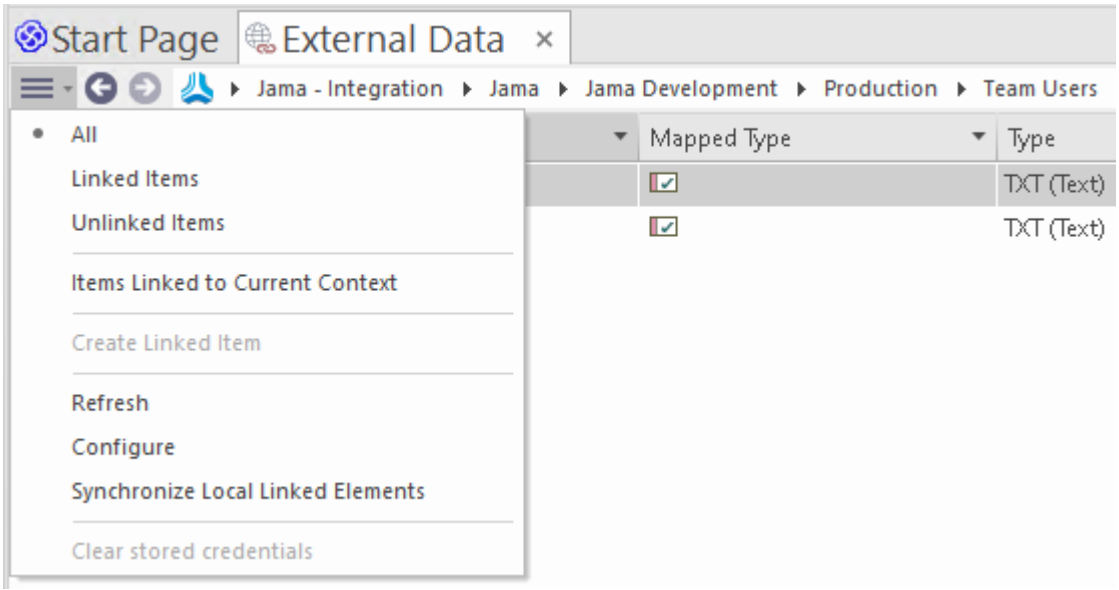
- What type of Jama item corresponds to which type of Enterprise Architect element, and
- What property of a Jama item corresponds to which property of an Enterprise Architect element.



These tasks are fully explained in the Configuration Help topic.

Review Jama Data

When you tested the communication between Enterprise Architect and Jama, on the External Data window, you used the breadcrumb trail to view lists of work items (e.g. Tasks). If this list is too long, you can click on the 'Hamburger' icon in the window toolbar.



This menu provides options such as:

- 'Linked Items' to show only those Jama items that are linked to Enterprise Architect elements
- 'Unlinked items' to show only those Jama items that have not yet been linked to Enterprise Architect elements, or
- 'Items Linked to Current Context' to show only the Jama items that are linked to the currently selected Enterprise Architect element

It's also possible to apply text based filters to the External Data table contents.

1. Right click the column header and select 'Toggle Filter Bar' (if it's not displayed already).
2. Type some text into the filter/search field just below the column name,

When selecting an item from the list each of the Properties window, Notes window, Inspector window and 'Discuss' tab of the Discuss & Review window have separate tabs or

versions of the window to display any properties, notes, features and discussions (respectively) of the selected item. These are illustrated in the External Item Details Help topic.

Working with Jama items and Enterprise Architect elements

Having created a working communication between Enterprise Architect and Jama, you can:

- Create a new element in Enterprise Architect linked to a Jama item
- Create a new Jama item linked to an element in Enterprise Architect
- Link an existing element in Enterprise Architect to a Jama item
- Update the Jama item with changes to the Enterprise Architect element ('Push' changes)
- Update the Enterprise Architect element with changes to the Jama item ('Pull' changes)
- Update all linked Enterprise Architect elements with any changes to the Jama items (Synchronize All)

These actions are all explained in the Linking Items Help topic.

Walkthrough: Polarion Integration

Prerequisites

In order to set up integration between Polarion and Enterprise Architect, you must first:

- Be using a licensed installation of Pro Cloud Server (Version 4.2.62 or later) with the Integration Plug-ins component
- Select the 'Enable Pro Features (OSLC, WebEA and Integration)' option in the Pro Cloud Server configuration for each Database Manager System hosting your Enterprise Architect models
- Have physical network access between the PCS server and the Polarion server
- Have user credentials to access the external data
- Be using Enterprise Architect Corporate, Unified or Ultimate edition v15.2 or later

We will assume that the Integration Server and the Polarion Integration Plug-in are to run on the same machine as the Pro Cloud Server

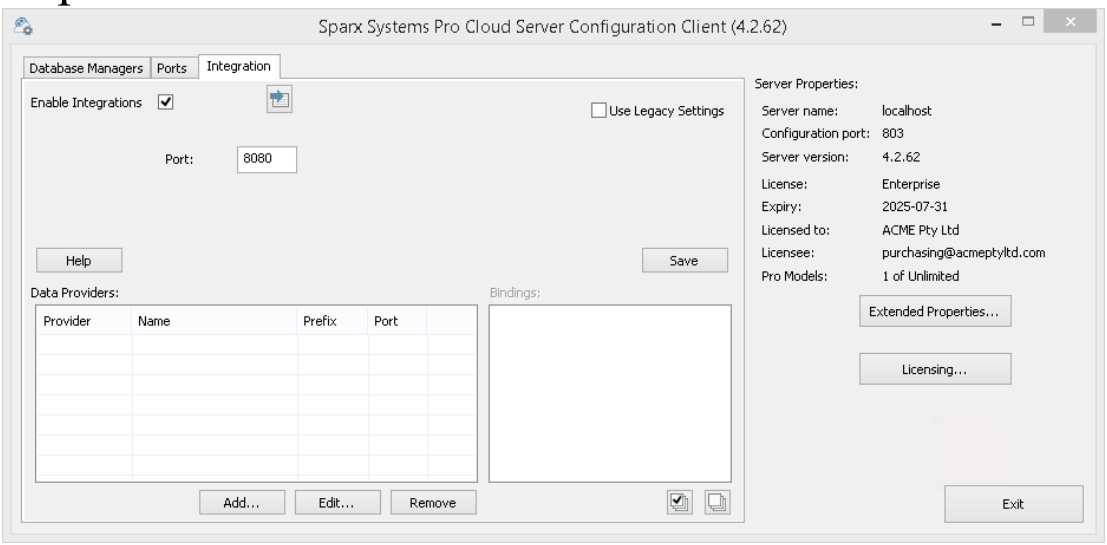
Identify Polarion to Pro Cloud Server and Enterprise Architect

Work through these steps:

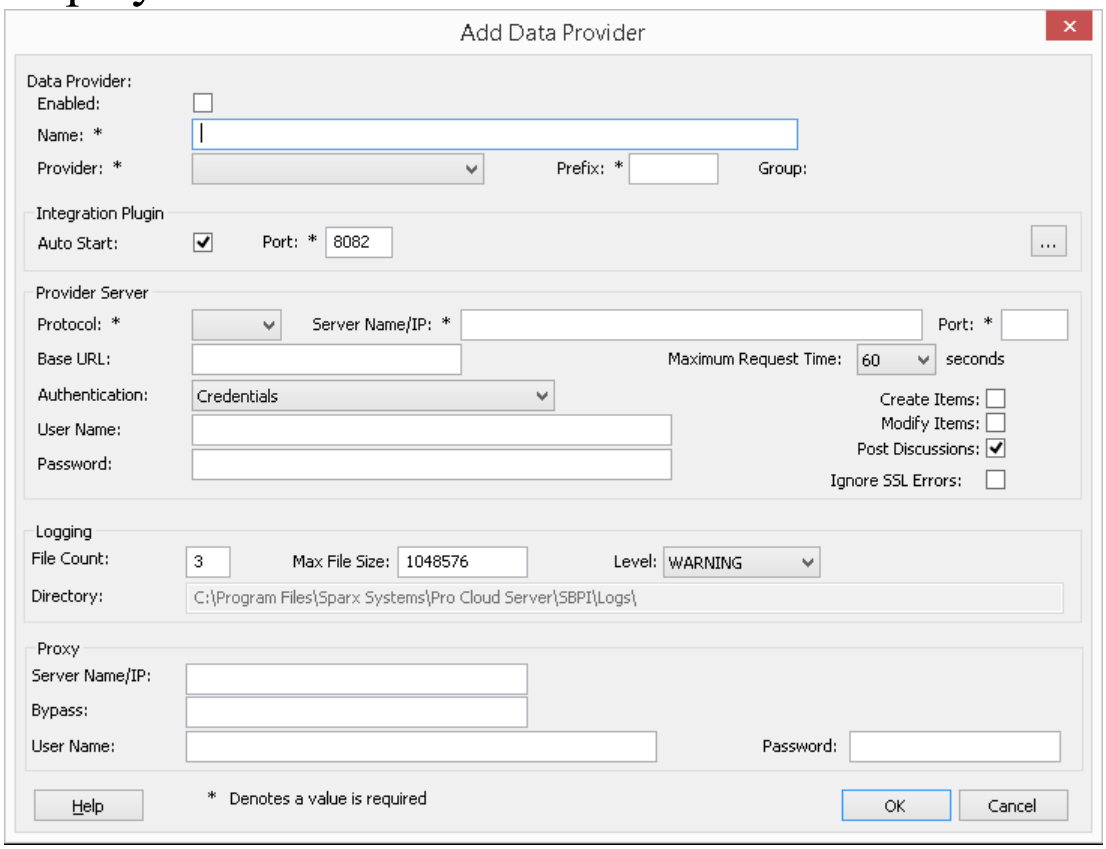
1. Double-click on the 'Pro Cloud Config Client' Desktop

icon to open the Pro Cloud Config Client window.

2. Select the 'Integration' tab and select the 'Enable Integrations' checkbox. In the 'Port' field type the number of the Port the Integration Server will listen on for requests from the Pro Cloud Server.

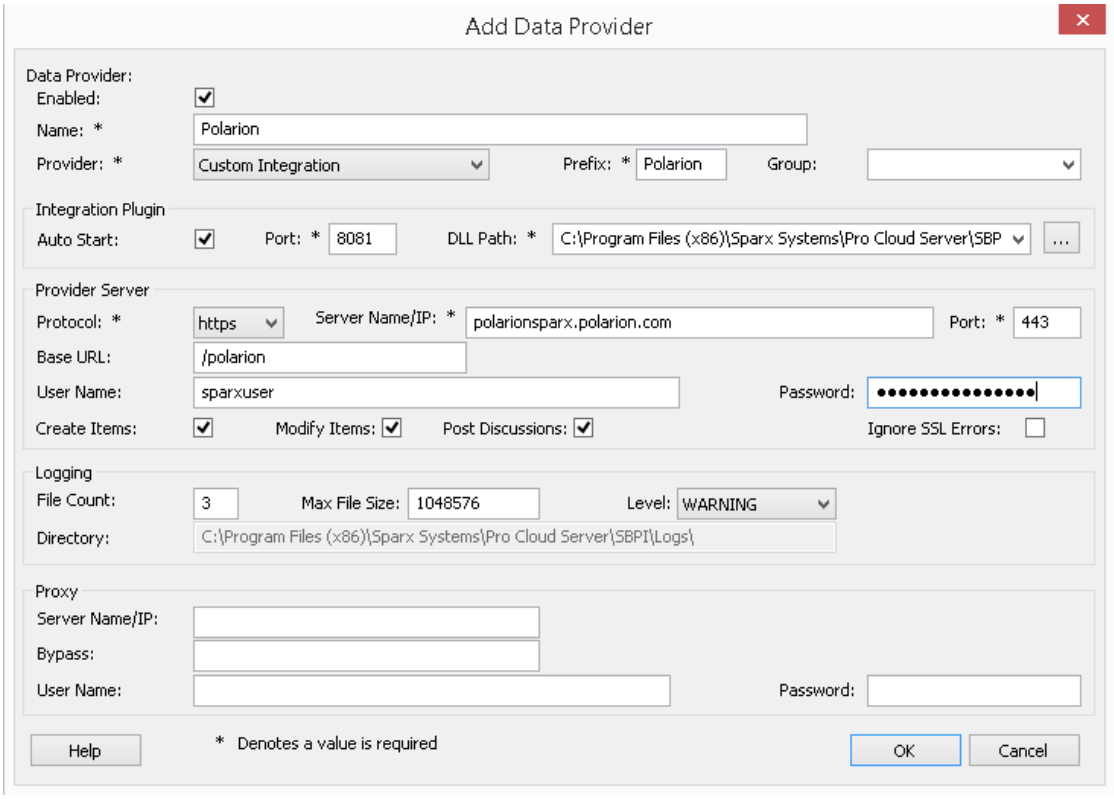


3. Click on the Add button. The 'Add Data Provider' dialog displays.



4. Select the 'Enabled' checkbox.
5. In the 'Name' field, type an appropriate connection name, such as 'Polarion'.
6. In the 'Provider' field, click on the drop-down arrow and select 'Custom Integration'.
7. In the 'Prefix' field, enter an appropriate prefix, such as 'Polarion'.
8. In the 'Integration Plugin' panel, in the 'Port' field, type the number of the Port that the Integration Server will use to communicate with the Polarion Plug-in.
9. Click on the ellipsis button next to the 'DLL Path' field. Browse and select the 'PolarionSbpi.dll' file. This is located within the Pro Cloud Server Installation folder. E.g. C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\SBPI\PolarionSbpi.dll
10. The fields in steps 10 to 13 make up the components of your Polarion web address; that is:
<protocol>://<server>:<port>/baseURL
In the 'Protocol' field, click on the drop-down arrow and select the protocol that the Polarion Plug-in will use to communicate with Polarion.
11. In the 'Server Name/IP' field, type the server name or IP address that the Polarion Plug-in will use to communicate with Polarion.
12. In the 'Port' field type the number of the Port that the Polarion Plug-in will use to communicate with Polarion.
13. In the 'Base URL' field enter the final portion of your Polarion URL (anything after the server name and port)

14. If you leave the 'User' and 'Password' fields blank, then each Enterprise Architect user will be prompted for their personal Polarion credentials, which can give a better usability.
15. If you prefer to set values in the 'User Name' and 'Password' fields, the values will be used in combination to access the data within Polarion. All current Enterprise Architect users of the Polarion installation will use the same set of credentials to read the external data.
16. Select the 'Create Items', 'Modify Items' and/or 'Post Discussions' checkboxes as necessary, to allow users of the Polarion Plug-in to create and/or update items and/or create Discussion posts in Polarion using Enterprise Architect.
17. If you want the Polarion Plug-in to ignore SSL-related errors that occur as a result of communicating with Polarion (e.g. if you are using a self-signed SSL certificate), select the 'Ignore SSL Errors' checkbox.



18. You can leave other fields blank or set to their default values. Click on the OK button to complete the configuration. This returns you to the 'Integration' tab of the Pro Cloud Server Configuration Client window.

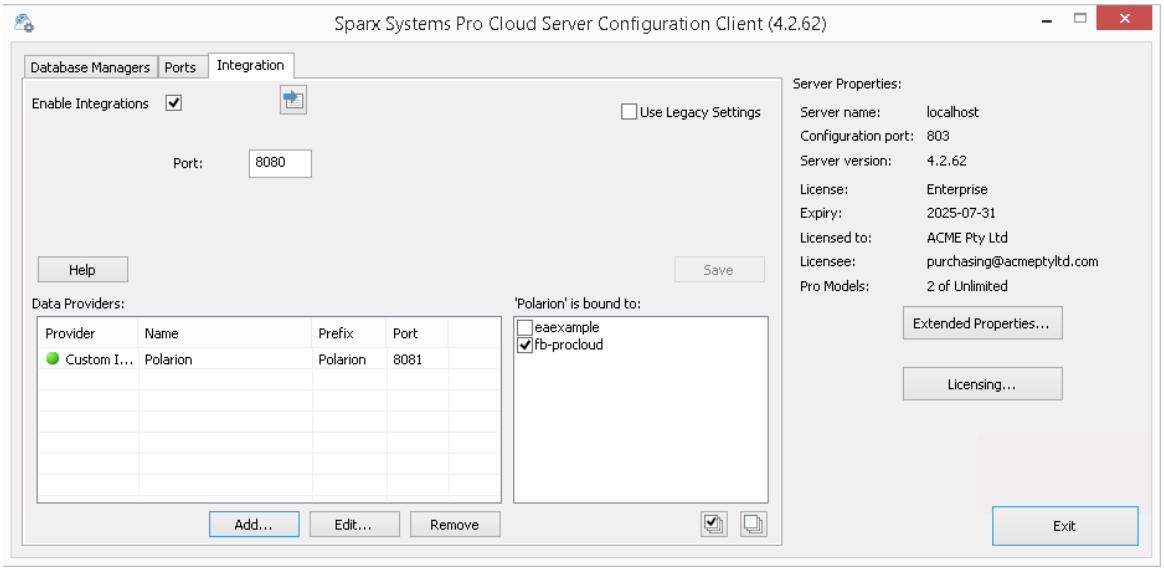
Identify Enterprise Architect models to Polarion

On the 'Integration' tab of the Pro Cloud Server Configuration Client window, you will now see:

- On the lower left of the screen, the 'Data Providers' panel listing the Custom Integration for Polarion as a data Provider
- On the lower right of the screen, the "Polarion' is bound to:' panel displaying a check list of the Enterprise

Architect models from your Pro-enabled Database Manager(s)

You can either click on the checkbox against each Enterprise Architect model you want to bind to Polarion, or click on the  to select all of them at once.



Click on the Exit button.

Restart the PCS Service to apply the changes you have made.

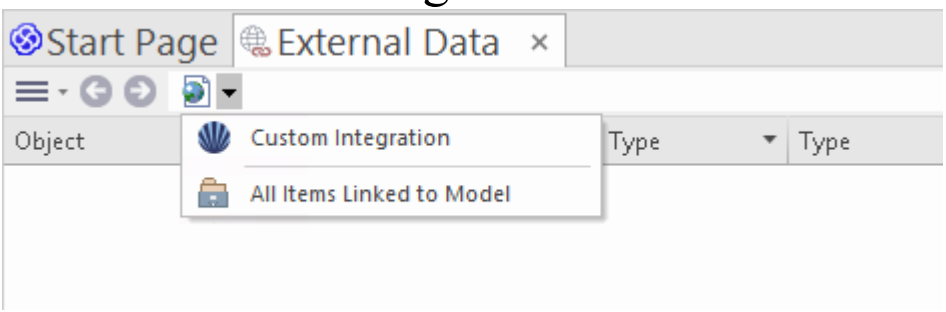
You can now go into one of your Enterprise Architect models and check the integration with Polarion.

Test the Integration

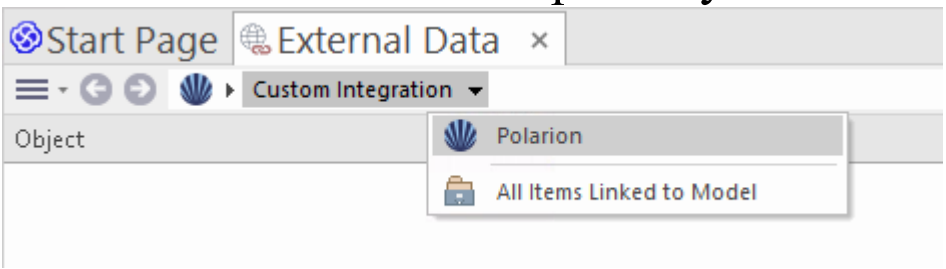
In Enterprise Architect open one of the models you have identified as being bound to Polarion.

1. Select the ribbon option Specialize > Tools > System Integration.
2. In the toolbar of the External Data window, click on the drop-down arrow to the right of the 'globe' icon.

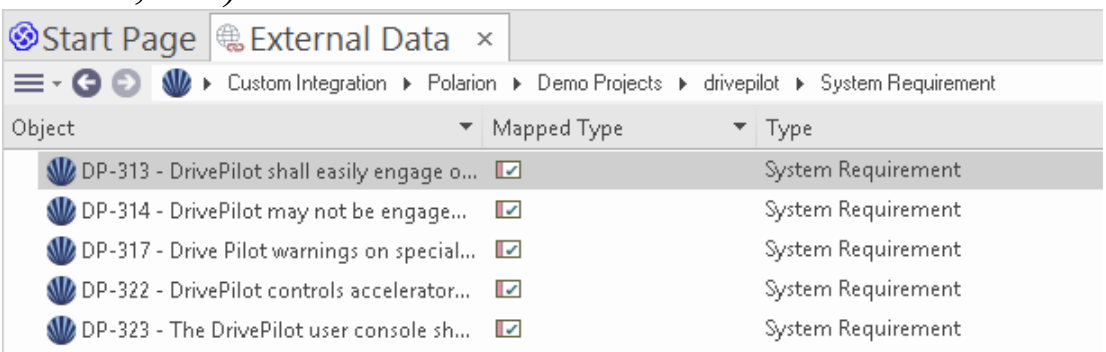
3. Select 'Custom Integration' from the list.



4. Click on 'Polarion' in the breadcrumb trail, this represents the root of the Polarion repository.



5. Continue using the breadcrumb options to drill down into the Polarion projects and access Work Items (Tasks, Issues, etc).



Data Mapping

In order for Polarion and Enterprise Architect to smoothly exchange data items, you have to review and - where necessary - edit:

- What type of Polarion item corresponds to which type of Enterprise Architect element, and

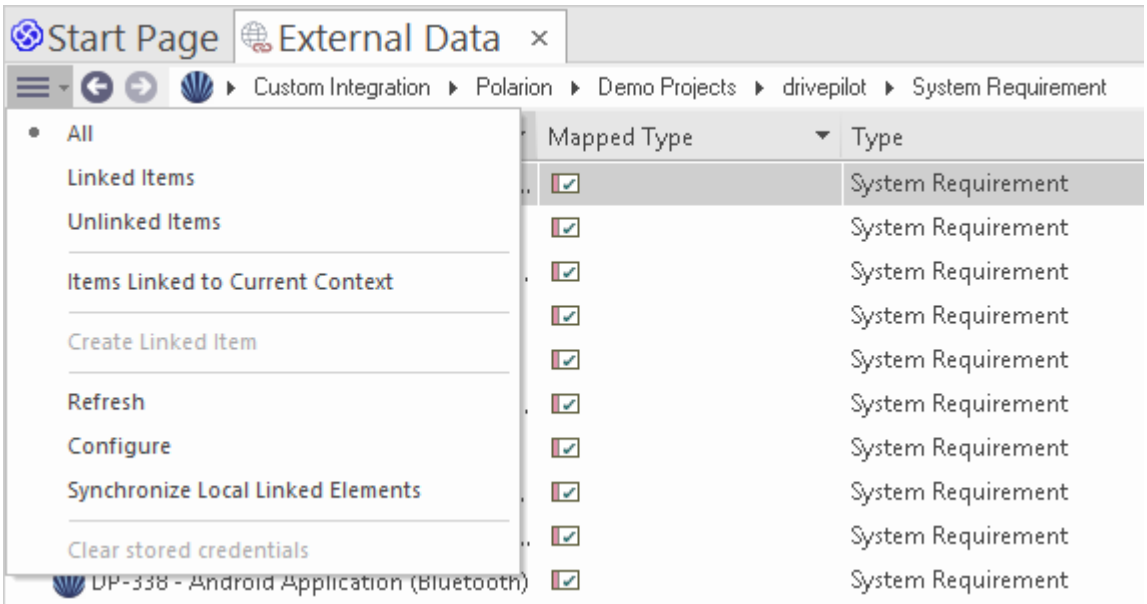
- What property of a Polarion item corresponds to which property of an Enterprise Architect element.

These tasks are fully explained in the Configuration Help topic.

Review Polarion Data

When you tested the communication between Enterprise Architect and Polarion, on the External Data window, you used the breadcrumb trail to view lists of work items (e.g. Tasks).

If this list is too long, you can click on the 'Hamburger' icon in the window toolbar.



This menu provides options such as:

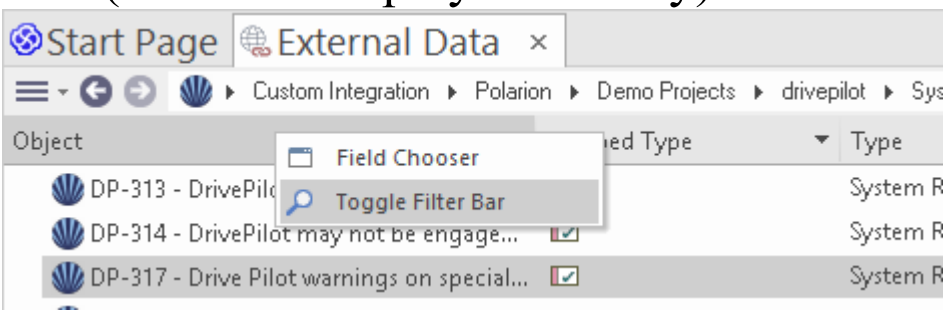
- 'Linked Items' to show only those Polarion items that are linked to Enterprise Architect elements
- 'Unlinked items' to show only those Polarion items that have not yet been linked to Enterprise Architect elements,

or

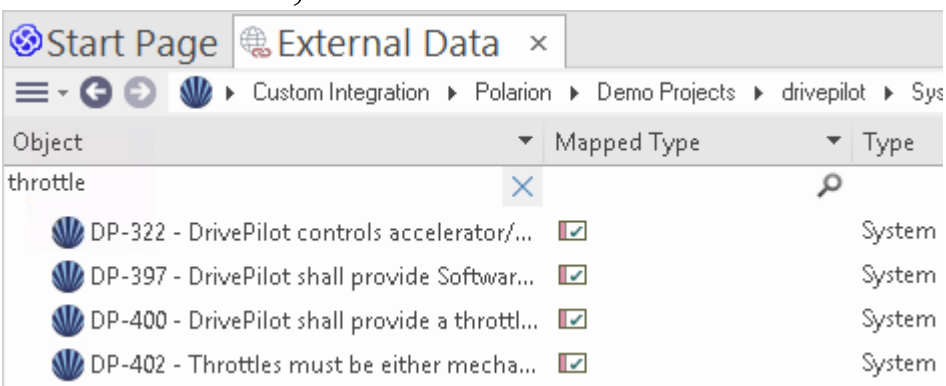
- 'Items Linked to Current Context' to show only the Polarion items that are linked to the currently selected Enterprise Architect element

It's also possible to apply text based filters to the External Data table contents.

1. Right click the column header and select 'Toggle Filter Bar' (if it's not displayed already).



2. Type some text into the filter/search field just below the column name,



When selecting an item from the list each of the Properties window, Notes window, Inspector window and 'Discuss' tab of the Discuss & Review window have separate tabs or versions of the window to display any properties, notes, features and discussions (respectively) of the selected item. These are illustrated in the External Item Details Help topic.

Working with Polarion items and Enterprise Architect elements

Having created a working communication between Enterprise Architect and Polarion, you can:

- Create a new element in Enterprise Architect linked to a Polarion item
- Create a new Polarion item linked to an element in Enterprise Architect
- Link an existing element in Enterprise Architect to a Polarion item
- Update the Polarion item with changes to the Enterprise Architect element
- Update the Enterprise Architect element with changes to the Polarion item
- Update all linked Enterprise Architect elements with any changes to the Polarion items

These actions are all explained in the Linking Items Help topic.

Note, when creating a Polarion item from Enterprise Architect, it is placed in the Polarion Work Items group (as a new job to be done).

Walkthrough: Enterprise Architect Integration

This walkthrough helps you to set up and use the integration capability of Pro Cloud Server to integrate one Enterprise Architect model with another, via the Cloud installation option.

Using integration with Enterprise Architect you can:

- Link Enterprise Architect elements with their counterparts in an external Enterprise Architect model
- Create local elements based on the external elements, and vice-versa
- Synchronize changes between local and the external elements, as and when needed
- Perform a traceability analysis
- Add discussions to elements from the external model

Prerequisites

In order to set up the integration Enterprise Architect, you must first:

- Be using a licensed installation of Pro Cloud Server with the Integration Plug-ins component, as just discussed
- The source and target models should both be accessible via the Pro Cloud Server
- Select the 'Enable Pro Features (OSLC, WebEA and Integration)' option in the Pro Cloud Server configuration

for each model/database manager.

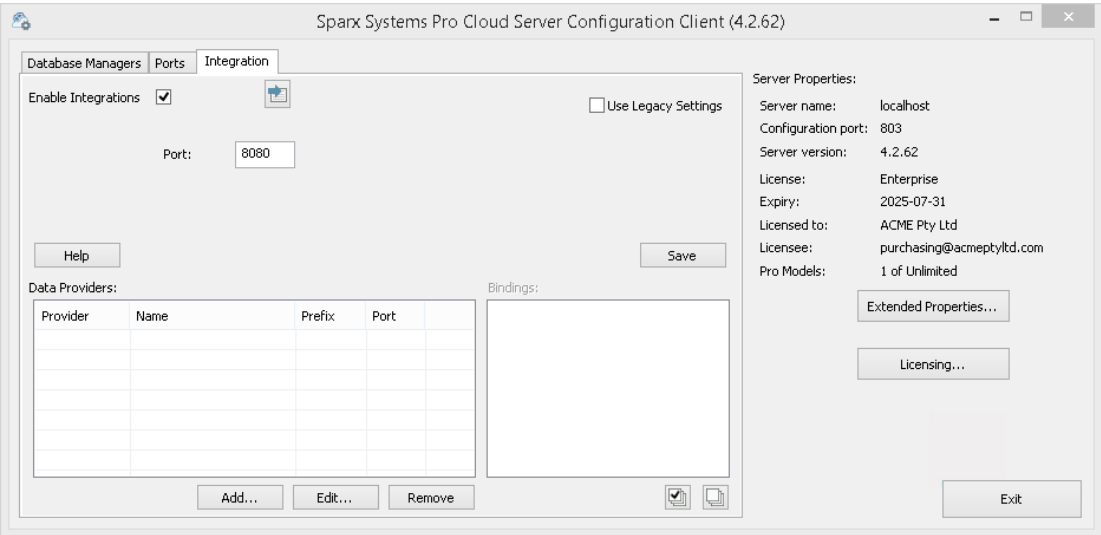
- Have physical network access between the PCS server and the PCS server hosting the external EA model (this is only applicable if the target model is hosted using a different/external PCS)
- Have user credentials to access the target model
- Be using Enterprise Architect Corporate, Unified or Ultimate edition v14 or later

For this walkthrough we will assume that you are defining the Integration Server options using the 'Integration' tab in the Pro Cloud Server Configuration Client rather than by manually manipulating the registry or configuration files or by using the WebConfig interface.

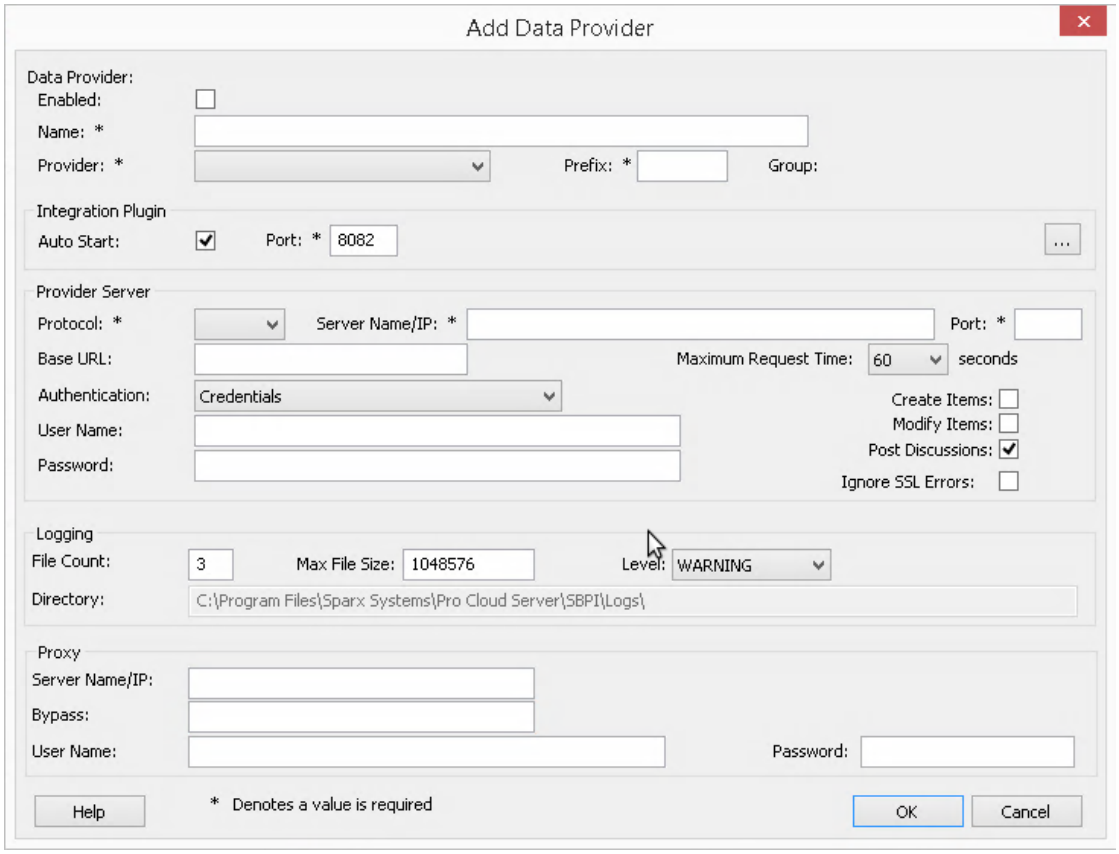
Configure the Enterprise Architect integration to access the target model

Work through these steps:

1. Double-click on the 'Pro Cloud Config' Desktop icon to open the Pro Cloud Config Client window.



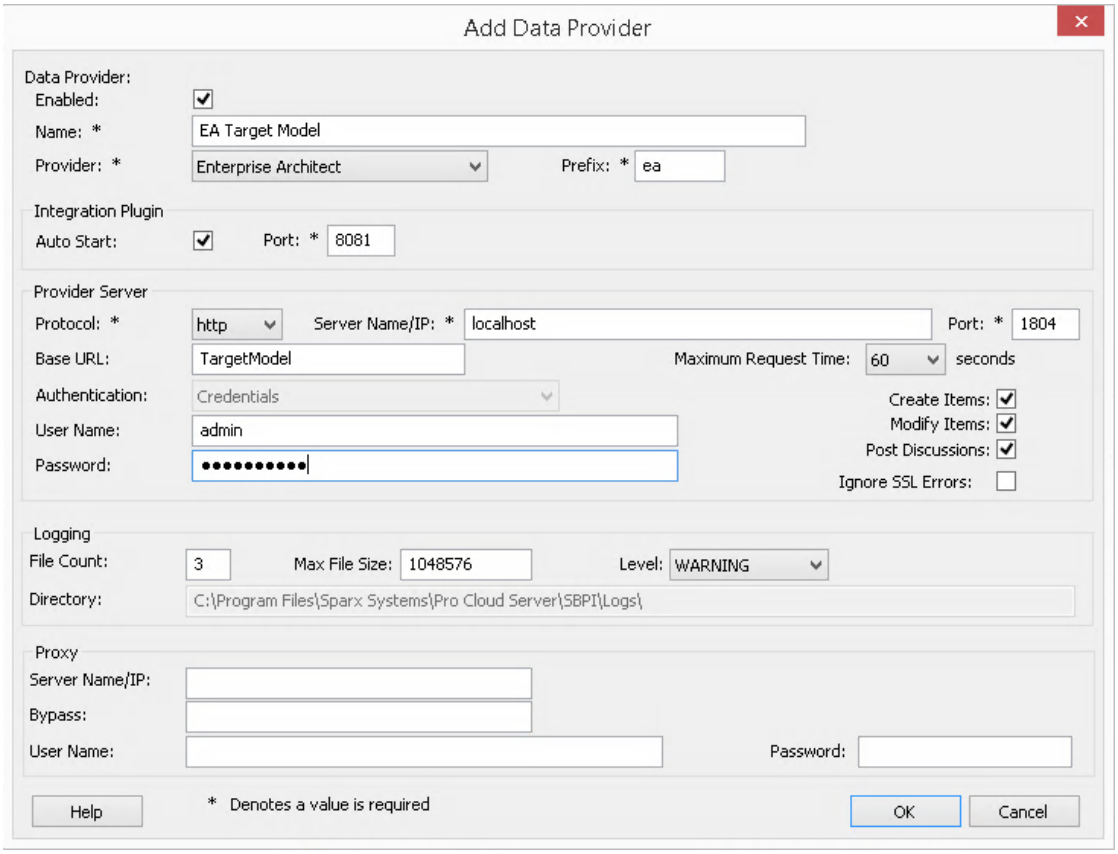
2. Select the 'Integration' tab and select the 'Enable Integrations' checkbox. In the 'Port' field type the number of the Port the Integration Server will listen on for requests from the Pro Cloud Server. This can be any valid/available port number of your choosing. In most cases the default value 8080 can be used.
   Note, you can use the Windows Resource Monitor (resmon.exe) utility to help identify which "Listening Ports" are already in use by other applications and services. There is a button to open the Resource Monitor on the "Ports" tab of the Pro Cloud Configuration Client.
3. Click on the Add button. The 'Add Data Provider' dialog displays.



4. Select the 'Enabled' checkbox.
5. In the 'Name' field, type an appropriate connection name. This integration will link to a specific target model, so it's a good idea to use a name which helps identify that model. For this example we will use a generic name, 'EA Target Model'.
6. In the 'Provider' field, click on the drop-down arrow and select 'Enterprise Architect'; this automatically adds 'EA' to the 'Prefix' field as well.
7. In the 'Integration Plugin' panel, in the 'Port' field, type the number of the Port that the Integration Server will use to communicate with the Enterprise Architect Plug-in. This can be any valid/available port number of your choosing. Typically the default value can be used (this dialog assigns a number above 8080 which is not already

assigned to another integration provider).

8. Note, the following four fields correspond to the Protocol, Server, Port and Model Name which you would enter when accessing the target model via the Cloud Connection option in Enterprise Architect.
9. In the 'Protocol' field, click on the drop-down arrow and select 'http' or 'https'.
10. In the 'Server Name/IP' field, type the server name or IP address of the machine running the Pro Cloud Server which is configured to access the target model.
    Often the source and target model are both defined on the current machine's Pro Cloud Server (where you are configuring the integration). In this case you can enter 'localhost' in this field.
11. In the 'Port' field enter the Port number used to access the target model.
12. In the 'Base URL' field enter the target model name, as you would when connecting to it via Enterprise Architect.
13. If the target model has user security enabled, enter some valid credentials in the 'User' and 'Password' fields.
14. Select the 'Create Items', 'Modify Items', and/or 'Post Discussions' checkboxes as necessary. These options allow users to create or modify elements, or post discussions in the target model.
15. If necessary (e.g. you are using HTTPS with a self signed certificate), check the 'Ignore SSL Errors' option.



16. You can leave other fields blank or set to their default values. Click on the OK button to complete the configuration. This returns you to the 'Integration' tab of the Pro Cloud Server Configuration Client window.
17. If you want to integrate with multiple different target models you can achieve this be repeating the steps above (starting at step 3), defining a provider for each target model.

Configure which source models can access the Enterprise Architect Integration

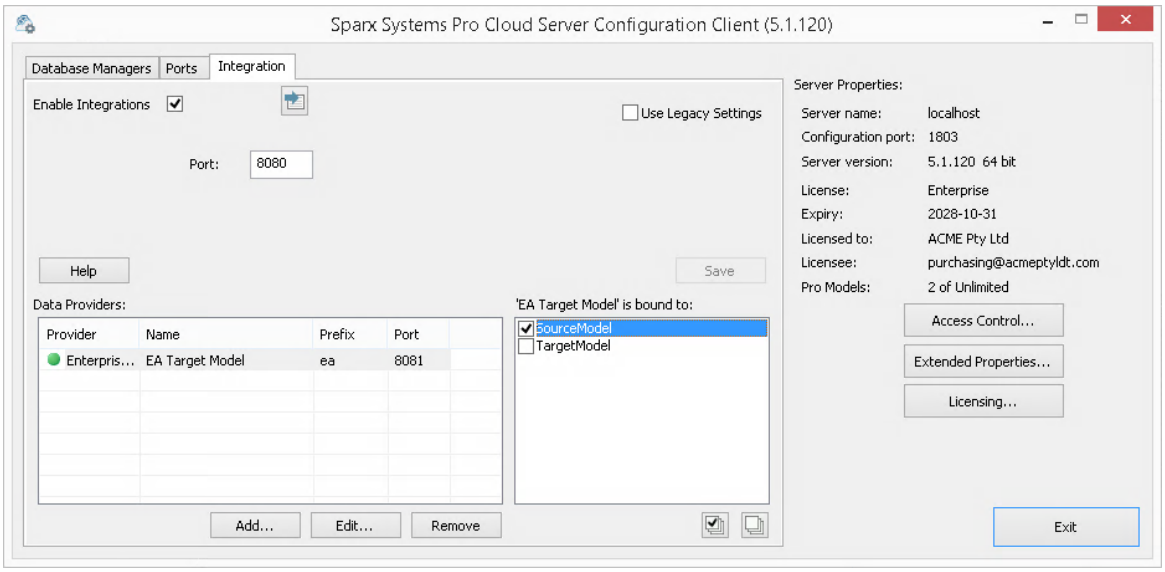
On the 'Integration' tab of the Pro Cloud Server Configuration Client window, you will now see:

- On the lower left of the screen, the 'Data Providers' panel

listing Enterprise Architect as a Data Provider

- On the lower right of the screen, the "Enterprise Architect' is bound to:' panel displaying a check list of the Enterprise Architect models from your Pro-enabled Database Manager(s)

You can either click on the checkbox against each Enterprise Architect model you want to bind to Enterprise Architect Integration (giving access to teh target model), or click on the  to select all of them at once.



Click on the Exit button.

Restart the PCS Service to apply the changes you have made.

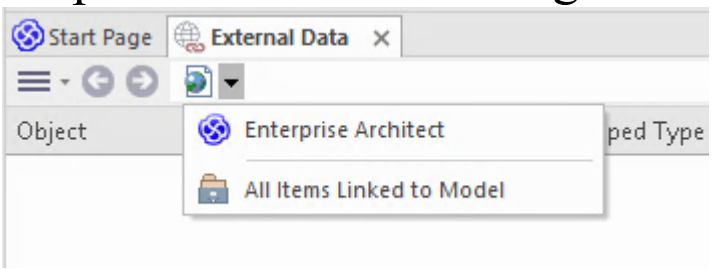
You can now go into one of your Enterprise Architect "source" models and check the integration with the external "target" model.

Test the integration

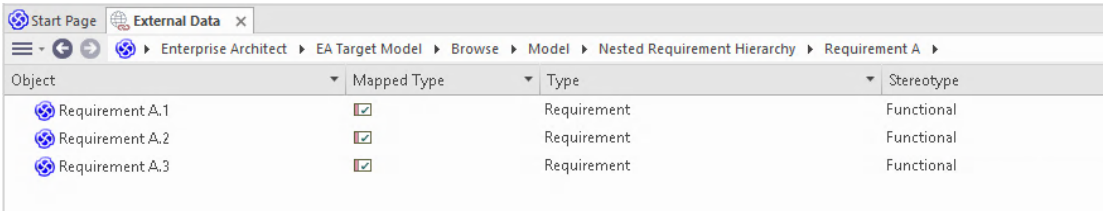
In Enterprise Architect open one of the "Source" models you

have bound to the external "Target" model.

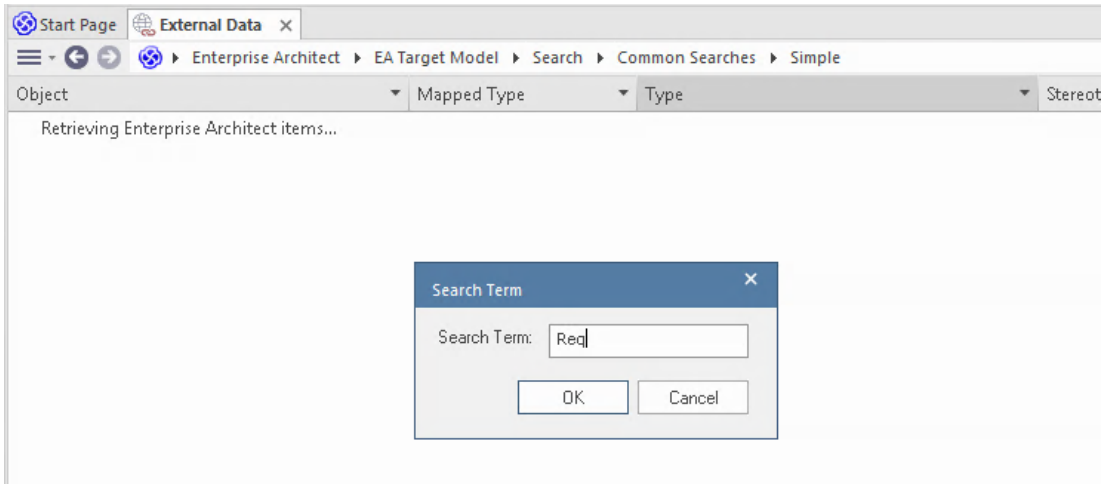
1. Select the ribbon option 'Specialize > Tools > System Integration'.
2. In the toolbar of the External Data window, click on the drop-down arrow to the right of the 'globe' icon.



3. Select 'Enterprise Architect' from the list; this adds the provider to the breadcrumb trail in the toolbar.
4. Click on the integration name (e.g. 'EA Target Model') in the breadcrumb trail, then either 'Browse' or ' Search'.
5. If you selected 'Browse', continue using the bread crumb to drill down into the model, displaying elements in the list below.

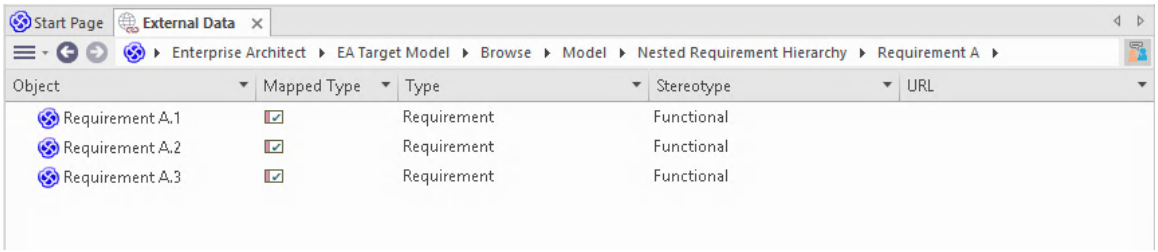


6. If you selected Search, select a search group (e.g. 'Common Searches'), a search definition (e.g. 'Simple'), then enter a Search Term in the prompt. All elements from the external model, which match the search terms, will be listed below.



Review the external model data

When you tested the communication between Enterprise Architect and the external model, on the External Data window, you either browsed, or search for a list of elements.



If this list is too long, you can click on the 'Hamburger' icon in the window toolbar and select options such as:

- 'Linked Items' to show only those elements in the external model, which are linked to elements in the current model.
- 'Items Linked to Current Context' to show only the elements from the external model, which are linked to the currently selected Enterprise Architect element or
- 'Unlinked items' to show only element from the external model, which have not yet been linked to elements in the current model.

Select the appropriate option, if necessary, and then explore the content of selected elements from the external model. Each of the Properties window, Notes window and Inspector window have separate tabs or versions of the window to display any properties, notes and features (respectively) of the selected item. These are illustrated in the External Item Details Help topic.

Working with Enterprise Architect elements and elements from the external model

Having created a working communication between the currently open model and an external model, you can:

- Create a new element in Enterprise Architect linked to an element from the external model
- Create a new element in the external model, which is linked to an element in the current model
- Link an existing element in Enterprise Architect to an element from the external model
- Update elements in the external model with changes to an element in the current model
- Update the elements in the current model with changes from an element in the external model
- Update all linked elements in the current model with any changes to elements in the external model

These actions are all explained in the Linking Items Help topic.

Troubleshooting

Integration Server messages

There are certain error messages that could display during the definition of the Integration Server settings; most are self explanatory. This table describes the most common error messages.

Error Message	Description
Please enter a value for [field_names]. The Integration Server can only be saved (while enabled) when all mandatory fields have a value	Reason: When the 'Enabled' flag is checked, all remaining Integration Server fields are mandatory. This message occurs when one or more Integration Server fields are empty and the 'Enabled' flag has been checked. [field_names] represents a placeholder for a comma separated list of field names that are empty.
The executable name of SBPI.EXE	Reason: The 'Executable Path' field has a value but it does not include the filename "...\SBPI.exe", which is the only valid

was expected	filename that can be used.
The specified Integration Server path of [full_path] does not exist or is invalid	Reason: The 'Executable Path' field has a value but the path portion of the specified value is not valid or does not exist on the Pro Cloud Server machine.
The specified Port is a duplicate of [duplicate_plugin_name]	Reason: The 'Port' field has been assigned a Port number, but it is the same as a Port being used by another Integration Data Provider identified by the name [duplicate_plugin_name].

Integration Data Provider messages

There are certain error messages that could display during the definition of an Integration Data Provider; most are self explanatory. This table describes the most common error messages.

Error Message	Description
Please enter a value for	Reason: To minimize the chance of defining an invalid Data Provider, the

[field_names]. An Integration Provider cannot be saved without all mandatory fields being assigned a value.	application ensures that all mandatory fields are given a value. [field_names] represents a placeholder for a comma separated list of field names that are empty.
Ports need to be unique across all Providers, the port value of [new_port] has already been used by [other_provider]	Reason: A Port number has been defined for the current Data Provider; however, the value is a duplicate of another provider's Port.
The specified execution path for the provider does not exist or is invalid,	Reason: The 'Execution Path' for the data provider has been assigned a value, but either the path or filename, or both, cannot be found on the local machine.

[full_path]	
The specified Port is a duplicate of the Integration Server	Reason: A Port number has been defined for the current Data Provider, but the value is a duplicate of the Port used by the Integration Server.
Failed to rename config file [old_filename] to [new_filename]	Reason: When the user changes the config filename of an existing Data Provider, the Cloud Server must rename the old filename to the new one. This message will be shown if the rename task is not successful (which can occur if the file is in use by another process) and in this case manual intervention is the only way to resolve this issue.
Prefixes need to be unique across all Providers, the prefix value of [new_prefix] has already been used by [other_provid	Reason: A Prefix has been defined for the current Data Provider, but the value is a duplicate of the prefix used by another provider.

er]	
Config filenames need to be unique across all Providers, the config filename of [new_filename] has already been used by [other_provider]	Reason: A Config filename has been defined for the current Data Provider, but the value is a duplicate of the config filename used by another provider.

General Troubleshooting

This table provides general advice to help identify and resolve common issues with the Integration component of the Pro Cloud Server.

Problem Description
Problem: After making changes to the Integration definitions of the Server or Data Providers, the Enterprise Architect users do not notice any differences. Solution: After changing the Integration configuration, was the Pro Cloud Server restarted? If not, restart the Pro

Cloud Server.
Problem: A newly configured Integration Data Provider is not listed in the navigation breadcrumbs of Enterprise Architect (after the Pro Cloud Server has been restarted). Solution: Confirm that the currently open model has a Binding entry for the Data Provider. For details, see the 'Bindings' field in the *Data Providers* table of the Integration Plug-ins Help topic.
Problem: Selecting items within the Integration navigation breadcrumb of Enterprise Architect seem to have no effect. Solution: It is possible that errors (or warnings) are occurring that are logged to the System Output window, therefore make sure the System Output window is displayed while using the External Data window.
Problem: General errors seem to be occurring; however, no useful information is shown in the System Output window. Solution: Each Integration Data Provider has the capability to write detailed entries to a log file; to ensure that the most detailed information is written, change the log level for the Data Provider to SYSTEM (restart the Pro Cloud Server) and then retry the same action in Enterprise Architect. Then review the Data Provider's log file; that is, in C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\SBPI\Logs, on the Pro Cloud Server.

Problem: Generic or blank errors received from the Plug-in.

Solution: Check that there are no other applications of services using the same Port as the SBPI.exe as defined in the main Integration configuration tab. If another program is using the same Port then the error messages will come from that application and not Pro Cloud Server or SBPI.

A list of open Ports and the applications using them can be found in Windows 'Resource Monitor'.

Information Accessed

Each of the Integration Plug-ins returns information based on a 'Filter' or position within the external product's data. Some products, such as Enterprise Architect, Jira and DevOps, provide a mechanism to customize the data returned, whilst others simply return all information at a particular position within the external provider's application.

Information Accessed from each Provider

Provider	Information returned
Application Lifecycle Manager	Information returned based on the internal list for Defects, Requirements and Tests.
AutoDesk	Information returned based on the contents of Hubs | Projects | Folders.
Bugzilla	Information returned based on the contents of Product | Component | <all items in component>.
Dropbox	Information returned based on the contents of Folders.
Enterprise	When you connect into an external

Architect	Enterprise Architect repository via an Integration, in the 'External Data' bread-crumb, you will be offered a menu with 'Browse' or 'Search'. On selecting 'Search' it returns a list of the searches defined in the local repository. On selecting a specific search, the Item List will display the results from the external repository's data.
Jazz	Information returned based on the contents of (DoorsNG) - Folders.
Jira	Presents a list of 'Favorite Filters'. See the menu option 'Issues | Manage Filters'.
Salesforce	Presents all item types that have a 'List View'. In a default install these include: Accounts, Assets, Campaigns, Cases, Groups, Contacts, Contracts, Leads, and Opportunities.
ServiceNow	Presents a list of user-defined filters, grouped by the table they are based on.
DevOps / Team Foundation Server	Presents a list of DevOps / TFS global queries and 'My ...' queries.

Wrike	Information returned based on the contents of Accounts | Folders.

Notes

For Enterprise Architect, Jira or DevOps the filters must be configured prior to Enterprise Architect requesting that an Integration Plug-in should connect to them.

Navigate External Data

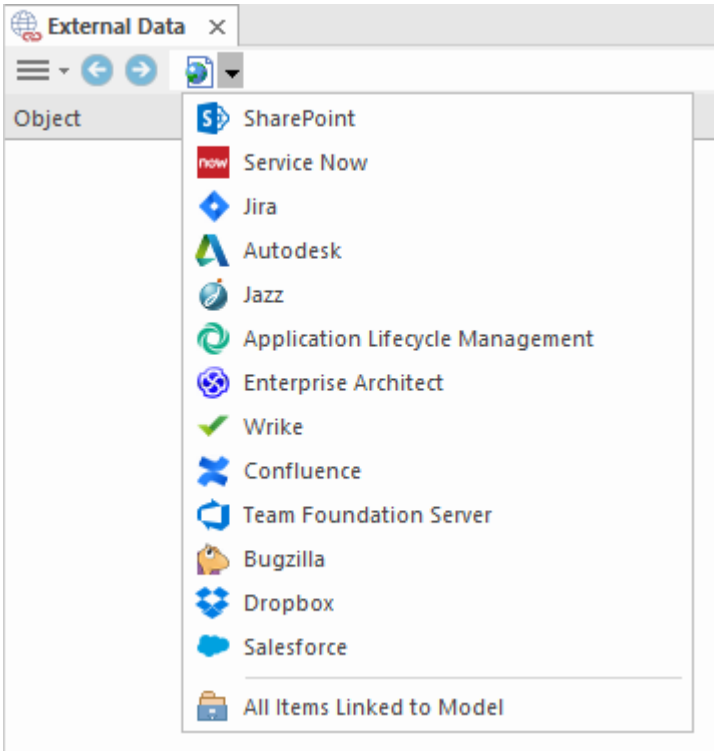
Each Integration Provider can be navigated to show lists of external items that can be linked to Enterprise Architect. Each provider might provide a slightly different mechanism for navigation, based on how it stores its data. Some provide a simple folder hierarchy, whilst others provide for user-defined filters. See the *What data is returned by Integration Plug-ins* table in the Install and Configure Help topic for details of each provider.

Access

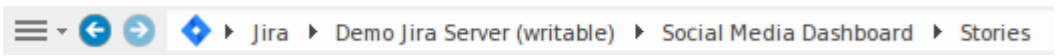
Ribbon	Specialize > Tools > System Integration > Open External Data

Navigate the Hierarchy

Begin navigation by selecting the provider type from the right-hand drop-down menu.

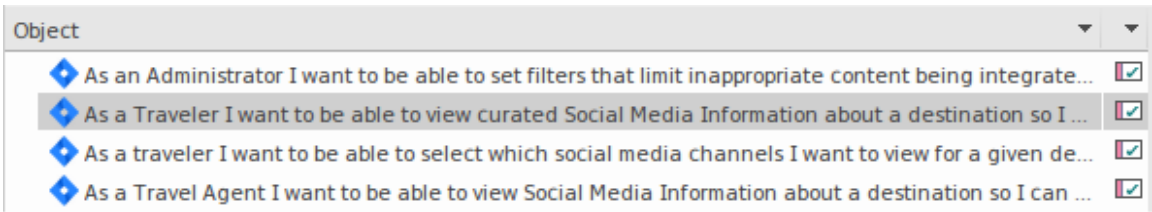


Next, each provider type offers a slightly different navigation system; for example, Dropbox allows for browsing the folder structure, whilst Jira gives a list of projects followed by a list of user queries.



Item List

At each navigation level, if available, a list of items corresponding to the navigation level will populate the left-hand panel.



Any local Enterprise Architect elements that are linked to the external item will be shown as a child of the external

item.

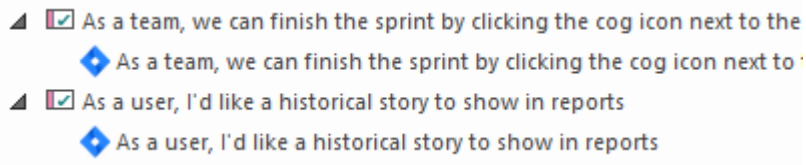


All Items Linked to Model

At each navigation level there is an option to show all local Enterprise Architect elements that are linked to the selected external data source. Select 'All Items Linked to Model' and choose the number of days prior to today (7, 30 or 90 days, or 'All') from which to collect the information.

Choose 'All Items Linked to Model' on the root navigation level to see the linked items for all external providers.

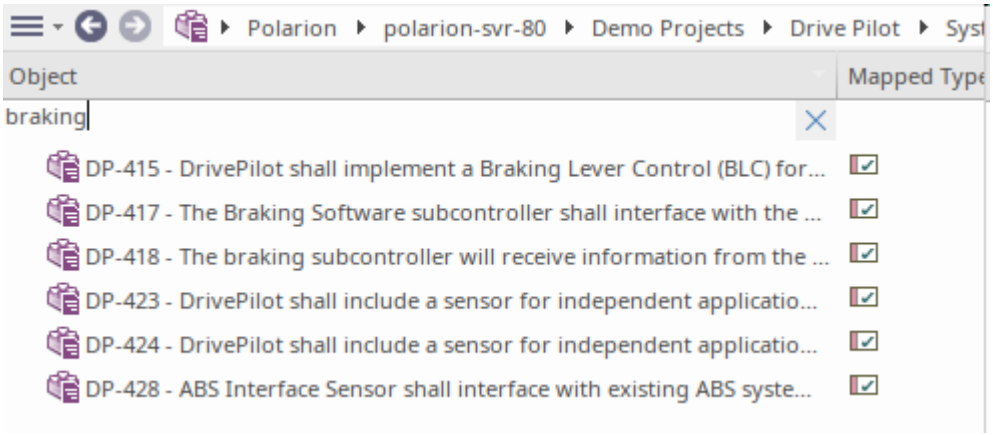
This view differs from the regular list as it shows the local Enterprise Architect element on top, and the external linked item as a child of the local element. All the same context menu items are available in the view.



Filter the List

The list of external items can be filtered using the Filter Bar. To activate the Filter Bar, right-click on the header of the list and select 'Toggle Filter Bar'.

Columns can be filtered by typing text into the Filter Bar. The list will be filtered to show only those items with text containing the filter text.



Show Items Linked to Current Context

In the integrations menu, select 'Items Linked to Current Context' to show only the external items that are linked to the currently selected local element. That is, select an item in the Browser window and see the external items that are linked to it.

This view is the same as for 'All Items Linked to Model'.

External Item Details

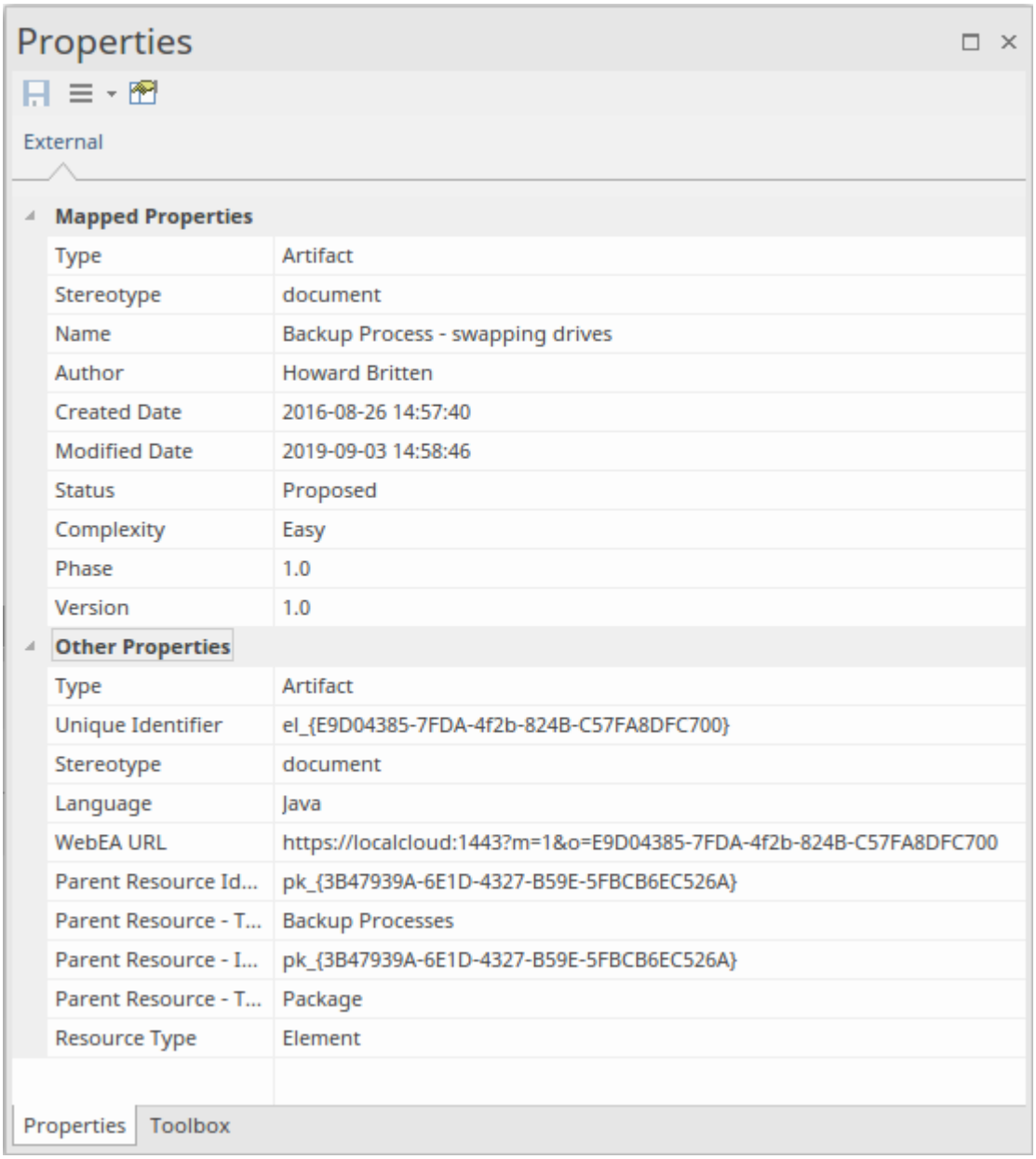
When an external item is selected in the External Data window, the item's details are retrieved and displayed in an 'External' tab of the Inspector window, and in External versions of the appropriate Properties, Notes and Discuss & Review windows.

Inspector Window

The 'External' tab of the Inspector window displays only when you are reviewing external items. It behaves in much the same way as the 'Details' tab of the Inspector window, revealing the external element's relationships, features, requirements, Tagged Values, project maintenance items and related files. However, the 'External' tab shows only categories for which items exist, and does not list all the possible but empty categories.

Properties

Click on an external item in the list to view its properties in the 'External' version of the Properties window. This window shows all available properties of the external item, as defined by the external provider.

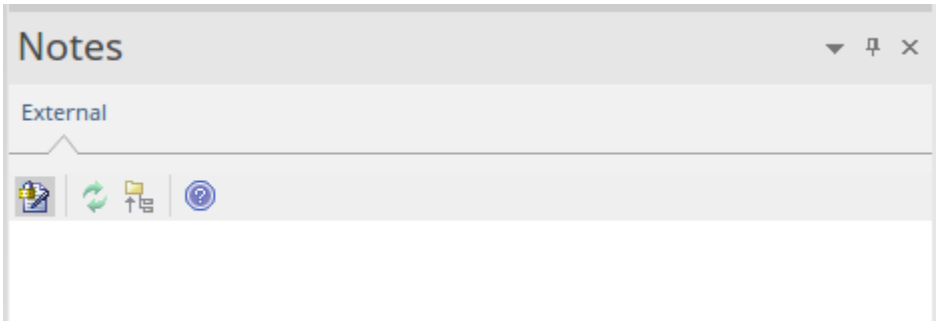


Properties that will be used when creating a linked local element are shown in the top group 'Mapped Properties', while all other properties are grouped in 'Other Properties'. The property mapping can be configured, as described in the *Configuration* Help topic.

Notes

The external item's 'Notes' or 'Description' text is shown in

the 'External' version of the Notes window.



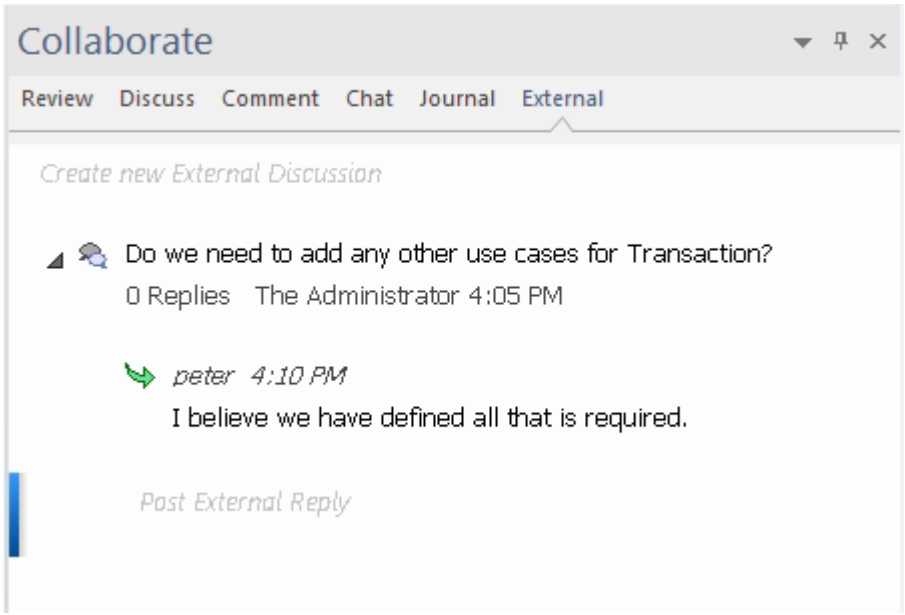
Note that not all external providers have 'Notes' or 'Description' fields.

This version of the Notes window has a different toolbar, containing icons to:

- Toggle the editing lock on the notes
- Synchronize the notes in the Enterprise Architect view and the external source
- Push the edited note text to the external source
- Display the online Enterprise Architect User Guide

Discussion

Select an external item from the list to view and participate in its discussions in the 'External' tab of the Discuss & Review window. The 'External' tab resembles the 'Discuss' tab and behaves in the same way.



New discussions can be posted by double-clicking on the *Create new External Discussion* text, or by right-clicking in the tab and selecting the 'Create New External Discussion' menu option.

Note:

- To post an external discussion the Integration Provider must be configured to allow this on the Pro Cloud Server (allowed by default)
- Not all external providers have discussions or comments

Configuration

Each Integration Provider comes with a set of default mapping values that determines firstly what type of local element is created in Enterprise Architect, and secondly which fields are copied to the new element. These mappings are configurable for each client model.

Permission

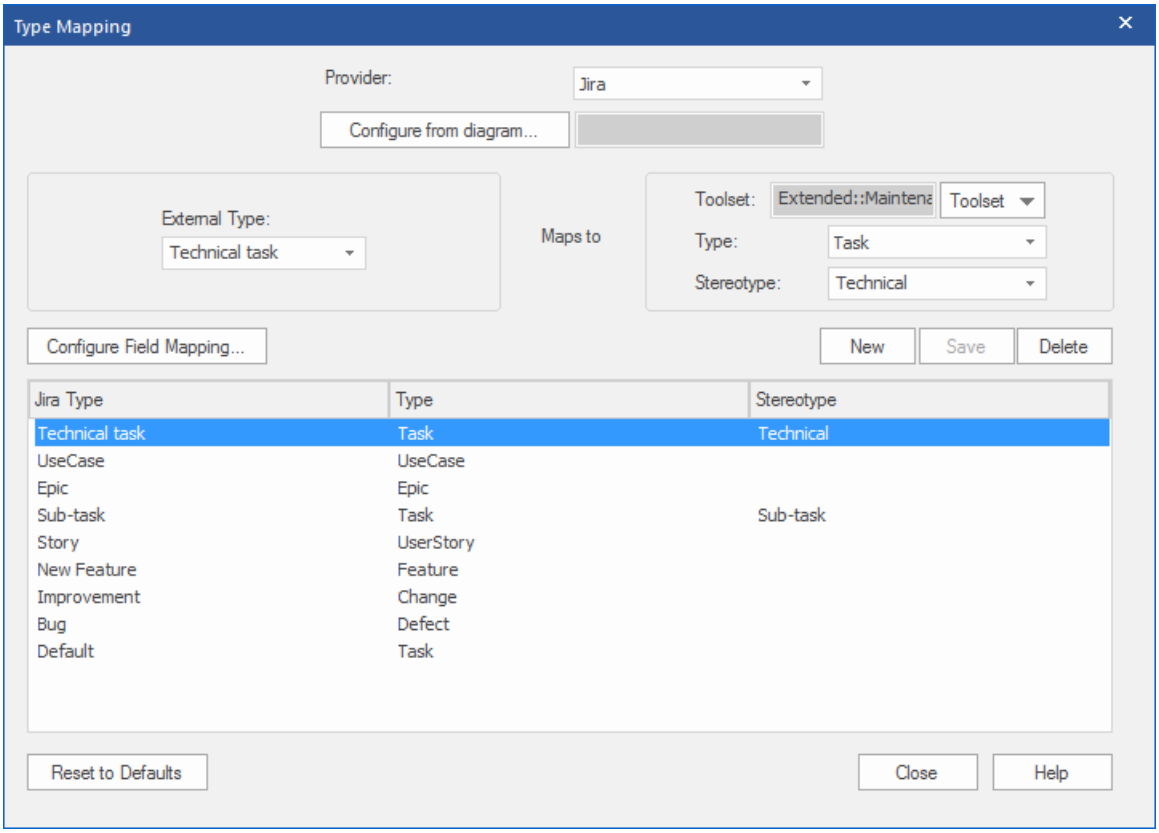
You must have 'Configure External Data Sources' permission to access this functionality.

Access

External Data window toolbar	'Hamburger' Menu > Configure

Type Mapping

The 'Type Mapping' dialog defines the element type that will be created when users select the 'Create Local Element' option for an external element.



To create a new mapping:

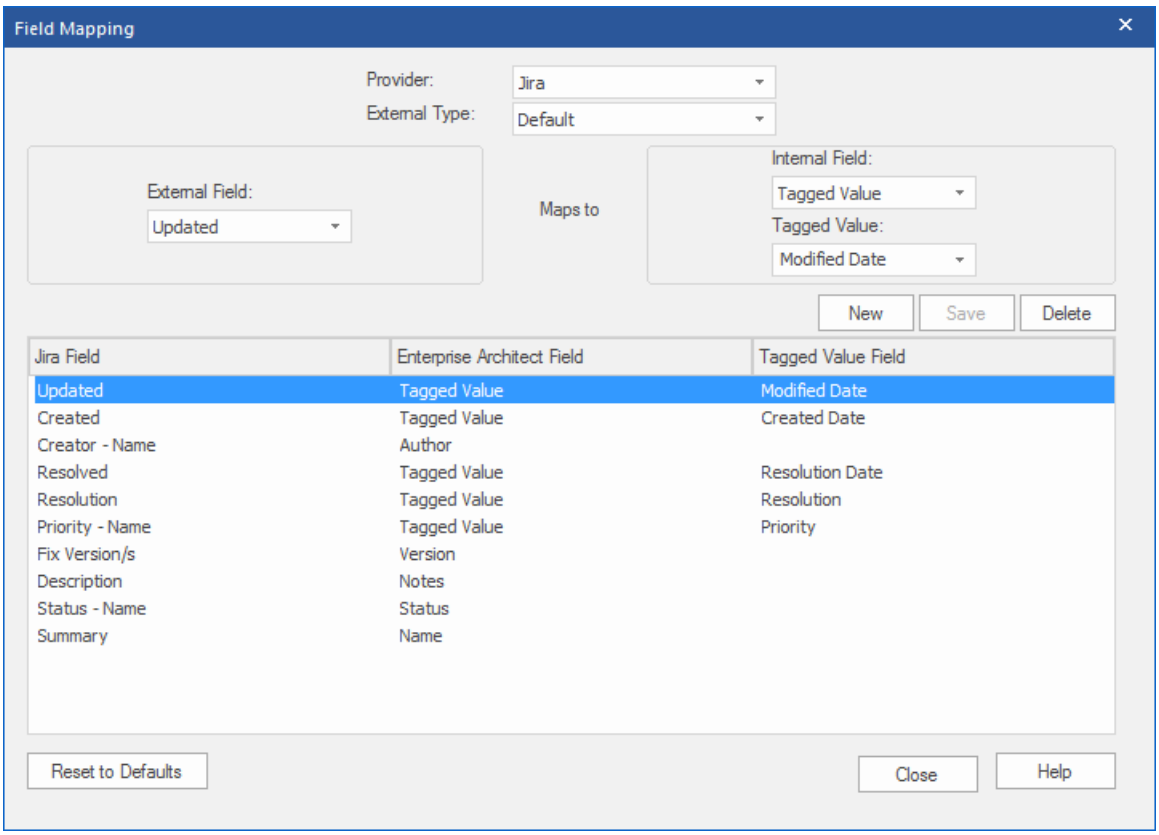
1. Click on the New button, then click on the 'External Type' drop-down arrow and select the type of the element to map from the external source.
2. Click on the 'Toolset' drop-down arrow and select the appropriate Enterprise Architect Toolset, then click on the 'Type' drop-down arrow and select the Enterprise Architect element type that the external element type maps to.
3. If appropriate, also click on the 'Stereotype' drop-down arrow and select the stereotype for the extended Enterprise Architect element type that the external element type maps to.
4. Click on the Save button.

Note that if the external element type has been linked as an

<<ExternalReference>> element, any Type Mapping settings are ignored; see the Linking Items Help topic.

Field Mapping

The 'Field Mapping' dialog defines the element fields that will be updated for a mapped element when users select the 'Create Local Element' option. You display this dialog by clicking on the Configure Field Mapping button on the 'Type Mapping' dialog, having mapped an external element type to an Enterprise Architect element (or selected an existing mapped element type).



The process generally maps the fields for a specific type of element; however, if you want to map certain fields for all types of element, click on the 'External Type' drop-down arrow on the 'Field Mapping' dialog, and select the value

'Default'.

To map the fields:

1. Click on the New button, then click on the 'External Field' drop-down arrow and select the name of the field to map from the external source.
2. Click on the 'Internal Field' drop-down arrow and select the appropriate Enterprise Architect field name that the external element field maps to.
3. If the internal field is a Tagged Value field, also click on the 'Tagged Value' drop-down arrow and select the Tagged Value that the external element field maps to.
4. Click on the Save button.

Notes

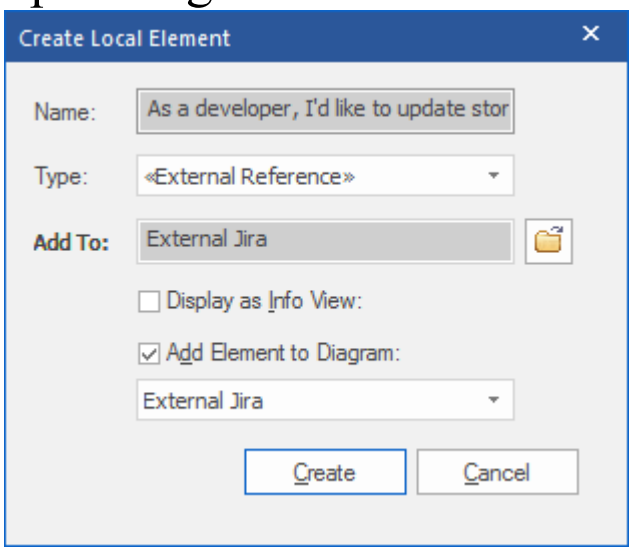
- Each mapping can be reset to default values by clicking on the Reset to Defaults button
- If no type mapping is defined for a given element type, then the mapping for type 'Default' will be used

Linking Items

Create Linked Element in Enterprise Architect

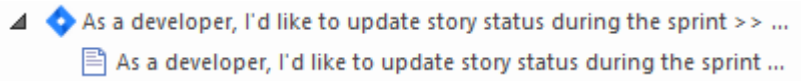
To create a new element in the Enterprise Architect model linked to the external item:

1. Right-click on an external item in the list in the External Data window.
2. Select 'Create Local Element'.
3. Select either the default type or «ExternalReference» (which ignores any Type Mapping; see the Configuration Help topic).
4. Select which Package the element will be created in.
5. Optionally add the new element to the currently-active open diagram.



The local element properties will be created from the values defined in the 'Field Mapping' dialog, described in the Configuration Help topic.

Once linked, the local element will show under the external item:



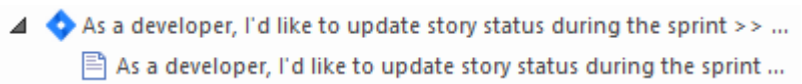
Link Multiple Items

Select multiple items to create multiple elements at the same time. All elements will be created with either the default type or «ExternalReference».

- To select a group of multiple items, select the first item, hold shift and select the last item.
- To select (or deselect) an individual item, hold Control and click the item.

Link an Existing Enterprise Architect Element

An existing element in Enterprise Architect can be linked to an existing item in the External Provider by dragging the local element from the Browser window and dropping it onto the external item. So in our example:



The Enterprise Architect element 'As a developer, I'd like to update story status during the sprint' was dragged from the Browser window and dropped onto the Jira item of the same name, creating a link between the two.

Create a Linked Item in the External Provider

Existing local elements in Enterprise Architect can be added to the External Provider in this way:

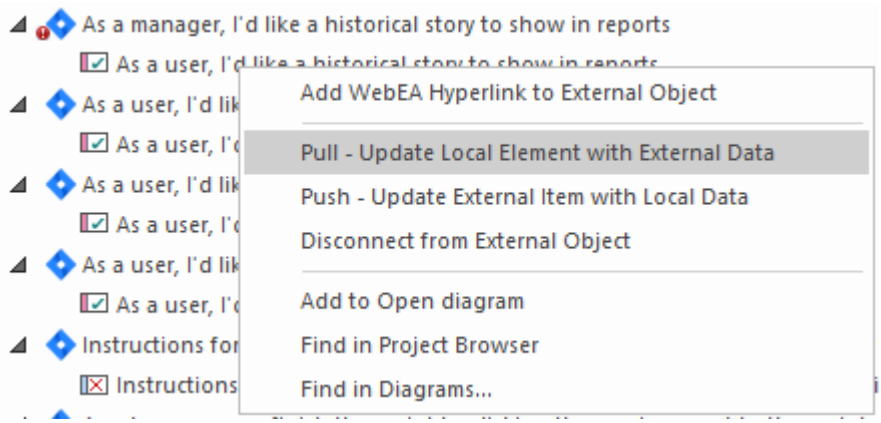
1. Select the local Enterprise Architect element (in the Browser window).
2. Browse the External Provider and select the required destination location.
3. In the 'External Data' menu, select 'Create Linked Item'.

Note:

- To create a new external item, the Integration Provider must be configured to allow this on the Pro Cloud Server (it is disallowed by default)
- Not all providers allow for creation of new items

Update Local Element ('Pull' changes)

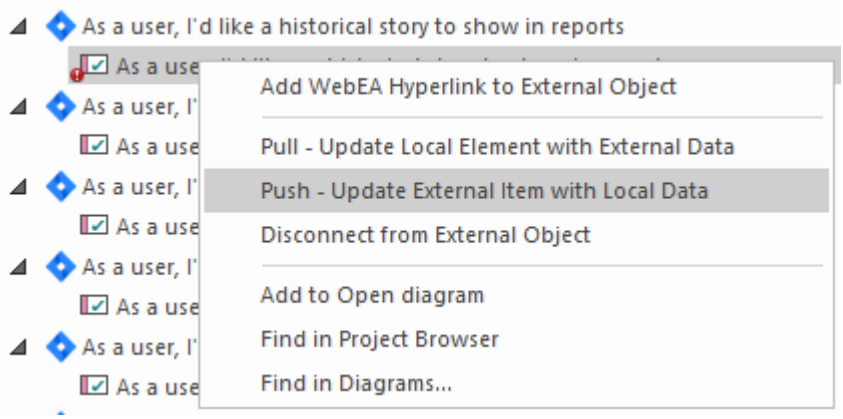
If either the local element or the external data item have been modified since the items were created or linked, an 'exclamation mark' indicator will display on top of the icon next to the name of the changed item. Right-click on the linked Enterprise Architect element and select the 'Pull - Update Local Element with External Data' option.



Note: If the local element has been modified since the link was created then its changes will be overwritten with the external data.

Update External Item ('Push' changes)

If either the local element or the external data item have been modified since the items were created or linked, an 'exclamation mark' indicator will display on top of the icon next to the name of the changed item. Right-click on the Enterprise Architect element and select the 'Push - Update External Item with Local Data' option.



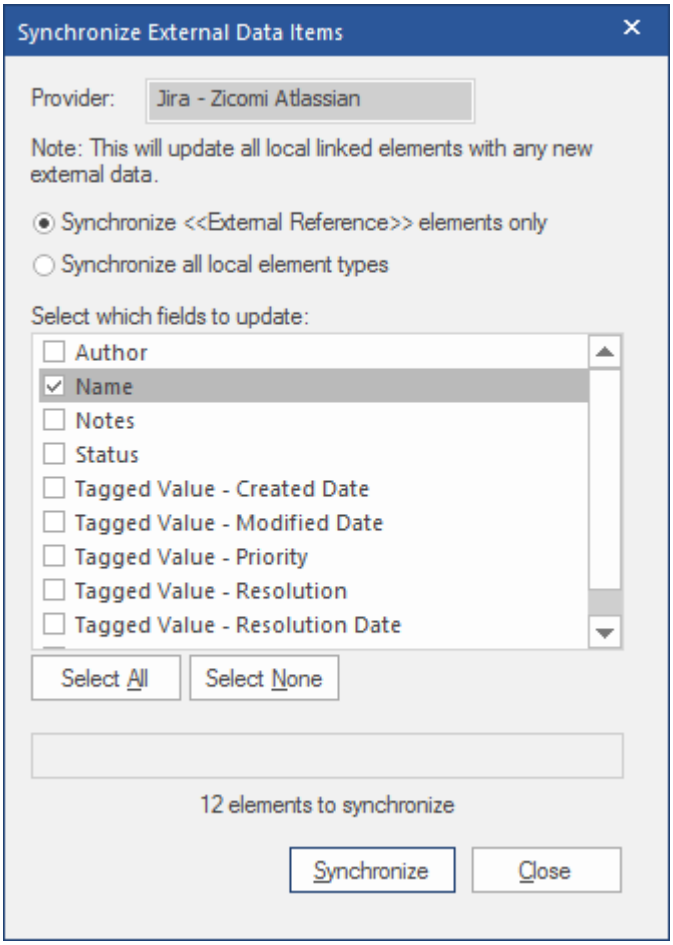
Note:

- If the external element has been modified since the link was created then its changes will be overwritten with the local data
- To update an external item the Integration Provider must be configured to allow this on the Pro Cloud Server (disallowed by default)

Synchronize All Local Elements

All local linked elements in Enterprise Architect can be quickly updated with new data from the linked items in the External Data Provider.

1. In the External Data window, display the data from the External Data Provider to be synchronized.
2. Click on the 'Hamburger' icon in the toolbar, and select the 'Synchronize Local Linked Elements' option.
3. On the 'Synchronize External Data Items' dialog, select the checkbox against each property field that should be updated with new data from the external item. Note that any local changes in the Enterprise Architect element in these fields will be overwritten with the external data.
4. The bottom of the window indicates how many local elements will be updated (note that it is unknown at this point if there have been any changes to the external item).
5. Click on the Synchronize button.

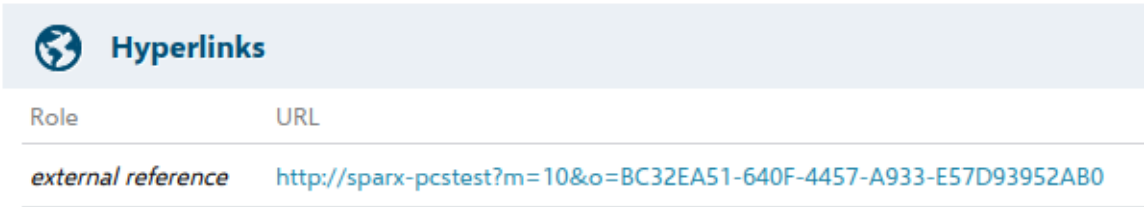

Synchronize External Data Items
Provider:
Jira - Zicomi Atlassian
Note: This will update all local linked elements with any new external data.
Synchronize <<External Reference>> elements only
Synchronize all local element types
Select which fields to update:
Author
Name
Notes
Status
Tagged Value - Created Date
Tagged Value - Modified Date
Tagged Value - Priority
Tagged Value - Resolution
Tagged Value - Resolution Date
Select All
Select None
12 elements to synchronize
Synchronize
Close

Item Hyperlinks to WebEA

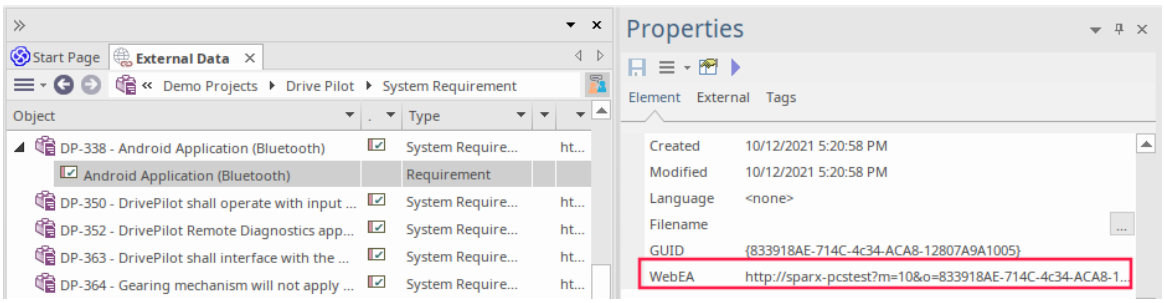
The elements created from external items can be referenced from the external application via a hyperlink back to the element details hosted on WebEA. The benefits of using WebEA as the reference include:

- Easy access to the details via a web- browser
- Accesses the most up-to-date data for that element
- No need to install a local copy of Enterprise Architect

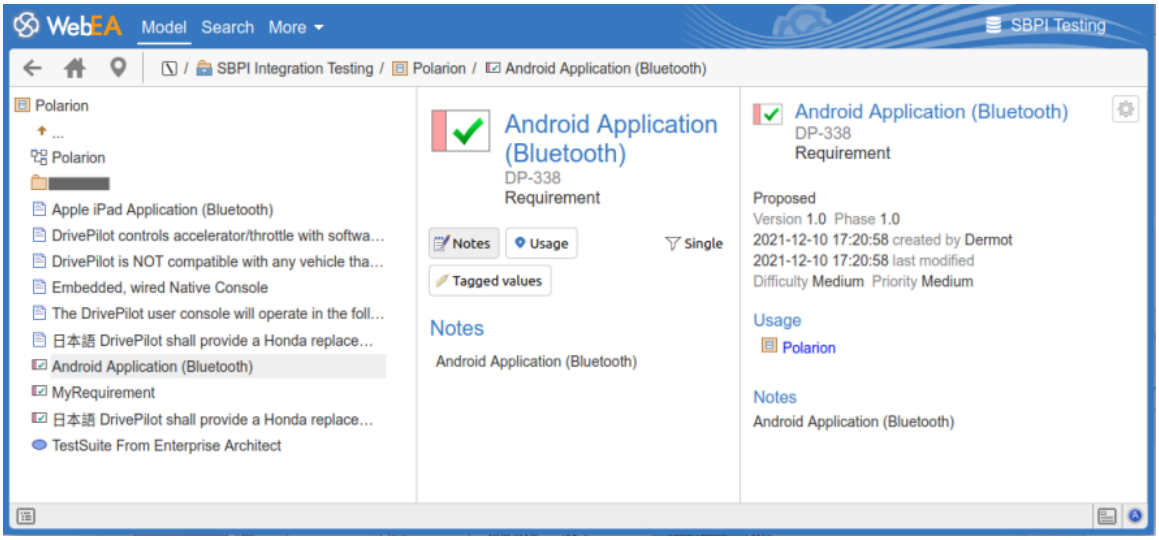
The hyperlink reference is set in the external item when a new element is created in Enterprise Architect. For example, here we have a hyperlink, inside an item in an external application, to a WebEA page for an element:



The hyperlink reference is system-generated and shown in the Properties window for that related element:



Here is the WebEA page referenced from the external hyperlink:



Supported

The Integrations that support hyperlink references to WebEA include:

- ALM
- Azure Dev Ops (TFS)
- Confluence (appended as 'info' field)
- Jira
- Jazz (including Doors NG)
- Polarion
- Wrike (as a comment on the item)

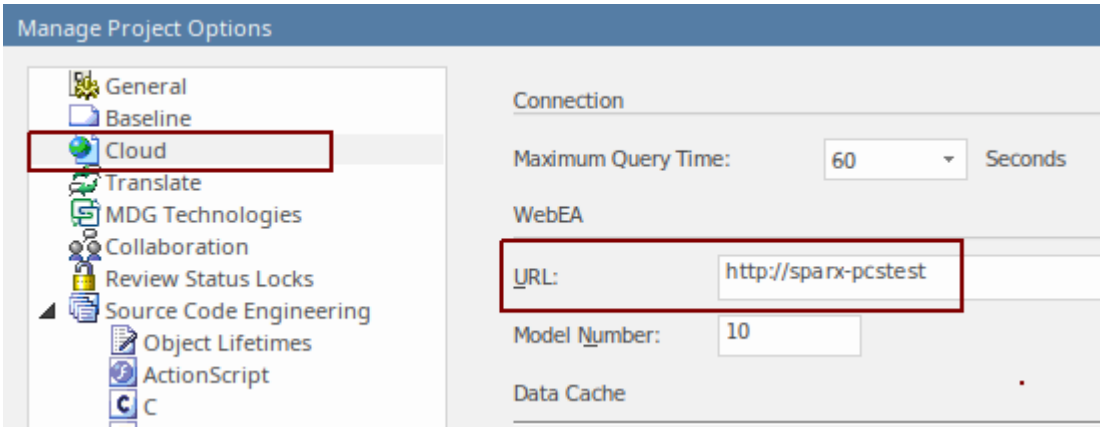
The external applications where this is not supported are:

- Dropbox
- Remote Enterprise Architect
- ServiceNow

Configuration

The configuration requires that WebEA is operating for that repository. To enable the external hyperlinks you must set the WebEA URL reference for that repository. The base reference is defined in:

- Settings > Model > Options > Cloud > URL



For more details see the *Cloud Page* Help topic.

Writing a Custom Integration Plugin

We live in a highly connected world, and while there is a formidable list of out-of-the-box integrations provided with the Pro Cloud Server, custom (proprietary) integrations can be created with any product that has a standard web service interface. This facility will both open up the contents of the Enterprise Architect repository to an external tool and make the information in the external tool available within Enterprise Architect. For example, a Project Management tool might define work pages that would be useful to visualize in Enterprise Architect, or an automated testing tool could define test cases and test procedures that could be related to implementation and specification elements in Enterprise Architect. This will require some technical expertise to create an integration using one of a number of programming languages such as C++ or C# , but this only needs to be done once and can be used across any number of repositories.

The Pro Cloud Server and Enterprise Architect will do the heavy lifting, and there is no need for an administrator to change any of the base security settings for the Pro Cloud Server as the new integration will operate through the existing Ports and firewalls. There is also no need for the developer to write http listening code, allowing them to focus on determining and configuring the RESTful API calls to pass the external items' information into and out of the server.

When installing the Pro Cloud Server, enable the 'SBPI

Examples' component to include the custom integration examples. When enabled, the default location for the example files is within the 'SBPI Examples\ExampleIntegrationPlugins' folder. For example:

C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\SBPI Examples\ExampleIntegrationPlugins

See the Pro Cloud Server Installation Help topic for more information.

Note, the 'SBPI Examples' installation option is not enabled by default. If you have already installed the Pro Cloud Server without the 'SBPI Examples', you can either perform a full re-install (enabling the 'SBPI Examples'), or use the installer's 'Change' option to add just the 'SBPI Examples' component.

To write your own Custom Integration Plug-in you can either start from scratch or make a copy of one of the examples and modify it. The Plug-ins can be written in either C++ or C#.

The examples are written using Visual Studio 2017 but this is not a pre-requisite.

The Custom Integration Plug-in must implement the interface defined in the ISBPIIntegrationPlugin, which is included in ISBPIIntegrationPlugin.h (for C++) or ISBPIIntegrationPlugin.cs (for C#).

The general flow of the program is:

- The user performs an action within Enterprise Architect that needs information from the Integration Plug-in
- The Plug-in will receive a call (or multiple calls) to the

appropriate interface method

- The Plug-in parses the request and, if required, makes its own request to the actual data provider
- The Plug-in receives the result from the actual provider, and parses the data
- The Plug-in sends the response to Enterprise Architect via the provided callback functions; this can either be the actual data requested or an error value
- Enterprise Architect receives the callback data and displays it to the user

Function/Class	Details
CheckVersion	(Not required in C#.) input: unsigned int version Returns true if your Plug-in supports the requested version. *Version 2 adds notifications when elements in Enterprise Architect are linked or unlinked to the external item, or when they are modified.* *Version 2 extends version 1, so returns true for versions up to and including the version supported.* *e.g. return (version <= 2);*
Create	(Not required in C#.)

Plug-in	The Plug-in must implement this export function:
	extern "C" SBPI_API SBPI_INTEGRATION_PLUGIN CreatePlugin();
	It must return a pointer to a class that implements the ISBPIIntegrationPlugin interface. The recommended implementation is:
	SBPI_INTEGRATION_PLUGIN CreatePlugin()
	{
	return new ExampleIntegrationPlugin;
	}
	The newly created ISBPIIntegrationPlugin can be deleted when it receives the ISBPIIntegrationPlugin::Release method.

ISBPIIntegrationPlugin Interface	The dll Plug-in must implement all methods in the ISBPIIntegrationPlugin interface.

ISBPIIntegrationPlugin interface

The ISBPIIntegrationPlugin interface provides a range of methods.

ISBPIIntegrationPlugin Interface Methods

Most methods have a single string parameter (char* in C++, string in C#) that is encoded as JSON to provide a variety of parameters.

The ISBPIIntegrationPlugin interface provides these methods:

Method Name	Notes
CreateWebEAHyperlink	Adds a link to WebEA within the External Item.
GenericRequest	Reserved for future use.
GetAuthorisationDetails	Returns the Authorization methods that your provider requires. Supported types are: none, basic and OAuth.
GetDefaultFieldMapping	Returns a list of how the External Item fields and properties map to the Enterprise Architect element fields,

	properties and Tagged Values.
GetDefaultTypeMapping	Returns a list of how the External Item types map to the Enterprise Architect element types.
GetFields	Returns a list of known fields/properties in the External Provider.
GetIcon	Returns an icon for the External Provider.
GetItem	Returns the full properties for the selected item.
GetItemDiscussion	Populates the 'External' tab in the main Discussion window with comments and discussions on the selected external item.
GetItemList	On receipt of a request to populate the list of items in the External Data window for the selected menu level, this method fills in the items.
GetItemNotes	Populates the 'External' tab in the main Notes window with details of the selected external item.
GetItems	Returns the full properties for the selected items.

GetMenuList	On receipt of a request to populate the menu in the External Data window, this method fills in the menu items.
GetTypes	Returns a list of known types in the External Provider.
ItemLinked	(Requires Enterprise Architect Release 15.2 build 1559 and above.) Notification that an external item has been linked to an element in Enterprise Architect.
ItemUnlinked	(Requires Enterprise Architect Release 15.2 build 1559 and above.) Notification that an external item has been unlinked from an element in Enterprise Architect.
ItemUpdated	(Requires Enterprise Architect Release 15.2 build 1559 and above.) Notification that a linked element in Enterprise Architect has been updated or modified.
PostNewDiscussion	Adds a new comment or discussion to the External Item.

PostNewItem	Creates a new item in the External Provider.
PostOAuthCode	Makes an Access Token Request to exchange the code for an OAuth Access Token (using the OAuth token endpoint). Returns the accessToken and refreshToken with AddProperty.
PostUpdateItem	Updates the selected item in the External Provider.
PostUpdateItemNotes	Updates the notes of the selected item in the External Provider.
RefreshOAuthToken	Performs a refresh request against the OAuth refresh endpoint and returns the new accessToken and refreshToken.
Release	C++ only. This method is called by the controlling application when the interface Class is no longer required. The method deletes the Class created during the CreatePlugin() function.
SetAuthorisation	Receives authorization information that the user enters within Enterprise Architect.

SetCallbacks(const void ** callbackFunctions)	C++ only. Passes in an array of callback function pointers that are used by the Plug-in later on to pass data back to Enterprise Architect.
SetConfiguration	Receives the settings defined when the user sets up the Custom Integration Plug-in in Pro Cloud Server.

CreateWebEAHyperlink

Add a link to WebEA in the External Item.

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string A JSON string of parameters. Includes details of the WebEA link to be added.

Outputs via Callbacks

[Optional] LogMessage or SetError - to provide user feedback.

Details

This method receives a request to a WebEA link to the External Item.

Some providers allow for external links to be added to an item, while others might only allow it to be added as a comment or to the notes section.

The input parameters string contains information about the

link to create, including:

- "itemID" - the ID of the item to be updated
- "webEALink" - the full URL of the WebEA link

Note that for the WebEA link to be valid, the 'Settings > Model > Options > Cloud' ribbon tab must have a valid WebEA base link specified.

GetAuthorisationDetails

Return the Authorization methods that your provider requires.

Supported types are:

- none
- basic
- OAuth

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string A JSON string of parameters. This is an empty string reserved for future use..

Outputs via Callbacks

- [Optional] AddProperty - to specify which authorisation methods are available and properties of these methods
- [Optional] LogMessage or SetError - to provide user feedback.

Details

For no authorisation, do nothing in this method.

Enterprise Architect only supports the OAuth 2.0 Authorization Code Grant type.

OAuth Authorisation will set Enterprise Architect to prompt to open a browser page for the user to log in to the OAuth provider (the 'authorization endpoint URI').

The OAuth provider will send back a 'code' to Enterprise Architect. Enterprise Architect will then call PostOAuthCode with the new code.

Note: If OAuth is specified then it will take priority and not allow basic authorisation.

OAuth requires the authorizationEndpointURIand redirectURI of the OAuth provider.

For Enterprise Architect, the redirectURI must be "http://localhost:8888/oauth/callback".

This usually needs to be added to your OAuth provider as an allowed redirect URI.

Basic Authorisation will set Enterprise Architect to prompt for a usename and password which will be passed back with each subsequent request.

Example Implementation

Basic Authorisation:

AddProperty(index, "basic", "true");

OAuth:

AddProperty(index, "OAuthConfiguration", "true");

AddProperty(index, "authorizationEndpointURI", "https://example.com/oauth/authorize");

AddProperty(index, "redirectURI", "http://localhost:8888/oauth/callback");

GetDefaultFieldMapping

Return a list of how the External Item fields and properties map to the Enterprise Architect element fields, properties and Tagged Values.

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string A JSON string of parameters. Reserved for future use.

Outputs via Callbacks

AddProperty - 2-3 values per field to map. Use a unique index value for each mapping:

- AddProperty(index, "externalField", "name");
- AddProperty(index, "internalField", "name");
- [Optional] AddProperty(index, "taggedValue", "My External Name");

[Optional] LogMessage or SetError - to provide user feedback.

Details

The values returned by this method are used to populate the Field Mapping in External Data Configuration.

The method defines the default values for how an External Item's fields/properties map to an Enterprise Architect element fields/properties/Tagged Values when linking an External Item.

The 'externalField' value should match the field ID as specified in GetFields, as well as the field names returned by, for example, GetItems.

The 'internalField' value should match an Enterprise Architect field name.

The options for internalField name are:

- 'Name'
- 'Alias'
- 'Author'
- 'Notes'
- 'Tagged Value' - This is a special case and requires AddProperty(index, "taggedValue", "tagged value name") to specify the name of the Tagged Value to use
- 'Status'
- 'Version'
- 'Phase'
- 'Keywords'
- 'Complexity'

- 'Scope'
- 'Multiplicity'
- 'Classifier'
- 'Abstract'
- 'Is Leaf'
- 'Language'
- 'Filename'
- 'Is Root'
- 'Is Specification'
- 'Persistence

Example:

- AddProperty(index, "externalField", "modifiedDate");
- AddProperty(index, "internalField", "Tagged Value");
- AddProperty(index, "taggedValue", "Modified Date");

GetDefaultTypeMapping

Return a default list of how the External Item types map to the Enterprise Architect element types.

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string A JSON string of parameters. Reserved for future use.

Outputs via Callbacks

AddProperty - 2-4 values per type to map. Use a unique index value for each mapping:

- AddProperty(index, "externalType", "Requirement");
- AddProperty(index, "internalType", "Requirement");
- [Optional] AddProperty(index, "stereotype", "document");
- [Optional] AddProperty(index, "toolbox", "Extended::Requirements");

[Optional] LogMessage or SetError - to provide user feedback.

Details

The values returned by this method are used to populate the Type Mapping in the External Data Configuration. It defines the default values for how an External Item's type maps to an Enterprise Architect element type when linking an item. This mapping is configurable by an end user in Enterprise Architect by editing the Type Mapping in the External Data Configuration.

The 'externalType' value should match the user-friendly Type name as specified in GetTypes, as well as the Type returned by, for example, GetItems.

The 'internalType' value should match an Enterprise Architect Type name.

Optionally, specify which Toolbox page this type came from. This will show when the user manually edits the mapping, allowing them to select another type from the same Toolbox page easily.

Always specify a default mapping type. This will be used if no matching mapping is found.

- AddProperty(index, "externalType", "Default");
- AddProperty(index, "internalType", "Requirement");

Optional stereotype:

- AddProperty(index, "stereotype", "document");

Specify the Toolbox page to allow users to change the mapping configuration to a type from the same Toolbox page.

- AddProperty(index, "toolbox", "Extended::Requirements");

Other mappings can be specified if required. For example:

- AddProperty(index, "externalType", "Feature");
- AddProperty(index, "internalType", "Feature");
- AddProperty(index, "toolbox", "Extended::Requirements");

GetFields

Return a list of known fields and properties in the External Provider.

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string A JSON string of parameters. Reserved for future use.

Outputs via Callbacks

AddProperty - 2 values per type to map. Use a unique index value for each mapping:

- AddProperty(index, "id", "type");
- AddProperty(index, "name", "Type");

[Optional] LogMessage or SetError - to provide user feedback.

Details

Most systems have an ID or non user-friendly name, as well

as a user-friendly display name. Only Fields that are returned here will be recognized by Enterprise Architect.

The values returned here will be matched against the Type returned in DefaultFieldMapping and GetItem.

If possible, this list of fields should be dynamically created by querying the External Provider for a list of its known fields. The list of fields can be hard-coded for providers that do not provide a list of fields.

GetIcon

Return an icon for the External Provider.

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string A JSON string of parameters. Reserved for future use.

Outputs via Callbacks

- AddBinaryProperty - the binary data of the icon.
- [Optional] LogMessage or SetError - to provide user feedback.

Details

Specify an icon to be shown in Enterprise Architect External Data. The returned icon should be in png format, 20x16 pixels, with the left 4 columns of pixels blank and transparent.

Bit depth of 32 to include an alpha channel.

- AddBinaryProperty(index, <pointer to binary data array>, iconBytesSize);

GetItem

Return the full properties for the selected item.

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string A JSON string of parameters. Includes: • itemID - the unique id of the item (as passed back in GetItemList) • level - the level hierarchy of the requested menu (starting with 1 for first level) • currentID, currentName - the id and name of the current menu level • levelXID - the ID of the menu level X for all previous levels in the hierarchy; that is, level1ID, level2ID • levelXName - the name of the menu level X for all previous levels in the hierarchy; that is, level1Name, level2Name

Outputs via Callbacks

AddProperty - multiple calls per menu item:

- AddProperty(index, "id", "item1");
- AddProperty(index, "name", "Item 1");

Optional extra properties. These can be custom propertyID:propertyValue pairs, eg:

- AddProperty(index, "notes", "Example notes for item 1");
- AddProperty(index, "modifiedDate", "2022-04-06T11:33:44");
- AddProperty(index, "type", "Requirement");
- AddProperty(index, "url", "http://example.com/item/1");
- AddProperty(index, "propertyX", "Item 1 Property X");
- AddProperty(index, "propertyY", "Item 1 Property Y");
- AddProperty(index, "propertyZ", "Item 1 Property Z");

[Optional] LogMessage or SetError - to provide user feedback.

Details

This method receives a request to populate the 'External' tab in the main properties list with details about this external item.

This method should fill in the item's properties by calling the AddProperty callback method.

Each item requires two calls to AddProperty, and can accept

extra optional calls to specify custom properties. The 'index' value is not required for this call. Set it to 0.

Mandatory properties:

- id - the id passed in specifies a unique id representing this menu item; the Plug-in can receive this id back in subsequent calls (such as when requesting the next sub-menu level)
- name - the user-displayable name of the menu item

Optional Properties:

The optional properties can be any propertyID:propertyValue pair. For the property to be listed in the Properties window, the property name must match a field value as returned by the GetFields method.

GetItemDiscussion

Return the discussions and comments for the selected item.

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string A JSON string of parameters. Includes: • itemID - the unique id of the item (as passed back in GetItemList)

Outputs via Callbacks

AddProperty - 3 calls are required per comment. Use a unique index value per comment:

- AddProperty(index, "authorName", "Example Person 1");
- AddProperty(index, "createdDate", "2022-04-06T12:34:56");
- AddProperty(index, "comment", "Example comment on Item 1 by Person 1");
- [Optional] AddProperty(index, "id", "0002"); // See note below

- [Optional] AddProperty(index, "parentID", "0001"); // See note below

[Optional] LogMessage or SetError - to provide user feedback.

Details

This method receives a request to populate the 'External' tab in the main Discussion Window with comments/discussion about this external item.

If an "id" property is added, this will be passed back if a user replies to a specific comment, allowing threaded discussions.

"parentID" will link this comment as a reply to the parent comment with the parentID

GetItemList

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string A JSON string of parameters. Includes: • level - the level hierarchy of the requested menu (starting with 1 for first level) • currentID, currentName - the id and name of the current menu level • levelXID - the ID of the menu level X for all previous levels in the hierarchy; that is, level1ID, level2ID • levelXName - the name of the menu level X for all previous levels in the hierarchy; that is, level1Name, level2Name

Outputs via Callbacks

[Optional] AddProperty - multiple calls per item:

- AddProperty(index, "id", "myitemid");
- AddProperty(index, "name", "Example Item 3");
- AddProperty(index, "modifiedDate", "2022-04-06T11:33:44");
- AddProperty(index, "type", "Requirement");
- AddProperty(index, "url", "http://example.com/item/3");

[Optional] LogMessage or SetError - to provide user feedback.

Details

This method receives a request to populate the list of items in the External Data window for the selected menu level. This method should fill in the items by calling the AddProperty callback method.

Each item requires five calls to AddProperty to fill in the details of the item in the list. And each item must use a unique 'index' value as the first parameter.

- id - the id passed in specifies a unique id representing this item; the Plug-in can receive this id back in subsequent calls (such as when requesting item details or notes)
- name - the user-displayable name of the item
- modifiedDate - the date/time that the item was last modified, in the format: YYYY-MM-DDTHH:MM:SS
- type - the type of the item as understood by the external provider; for example Defect, Task, Requirement, Document

- url - a url that will take a user directly to this item; this is used by Enterprise architect when selecting 'Open External Item in Browser Window'

Don't return an error for a blank response. Simply do nothing.

GetItemNotes

Return the notes or descriptions for the selected item.

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string A JSON string of parameters. Includes: • itemID - the unique id of the item (as passed back in GetItemList)

Outputs via Callbacks

- AddProperty - notes - the text value of the notes for the item; some basic HTML markup is allowed
- [Optional] LogMessage or SetError - to provide user feedback.

Details

This method receives a request to populate the 'External' tab in the main Notes Window with details about this external item.

GetItems

Return the full properties for the requested items.

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string A JSON string of parameters. Includes: • itemIDs - a comma separated list of item IDs • level - the level hierarchy of the requested menu (starting with 1 for first level) • currentID, currentName - the ID and name of the current menu level • levelXID - the ID of the menu level X for all previous levels in the hierarchy; that is, level1ID, level2ID • levelXName - the name of the menu level X for all previous levels in the hierarchy; that is level1Name, level2Name

Outputs via Callbacks

AddProperty - multiple calls per menu item:

- AddProperty(index, "id", "item1");
- AddProperty(index, "name", "Item 1");

Optional extra properties. These can be custom propertyID:propertyValue pairs, eg:

- AddProperty(index, "notes", "Example notes for item 1");
- AddProperty(index, "modifiedDate", "2022-04-06T11:33:44");
- AddProperty(index, "type", "Requirement");
- AddProperty(index, "url", "http://example.com/item/1");
- AddProperty(index, "propertyX", "Item 1 Property X");
- AddProperty(index, "propertyY", "Item 1 Property Y");
- AddProperty(index, "propertyZ", "Item 1 Property Z");

[Optional] LogMessage or SetError - to provide user feedback.

Details

Return the properties for a list of items. Recommended: If the provider has a way of querying a list of items, then it is recommended to run a single query and return the results. If the provider can't query multiple items, then run multiple individual 'GetItem' calls and concatenate the results.

The returned values should be the same as for GetItem, but specify a unique 'index' value for each different item.

GetMenuList

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string A JSON string of parameters. Includes: • level - the level hierarchy of the requested menu (starting with 1 for first level) • currentID, currentName - the ID and name of the current menu level • levelXID - the id of the menu level X for all previous levels in the hierarchy; that is, level1ID, level2ID • levelXName - the name of the menu level X for all previous levels in the hierarchy; that is level1Name, level2Name

Outputs via Callbacks

- AddProperty - 2 calls per menu item: "id", "name".

- [Optional] LogMessage or SetError - to provide user feedback.

Details

This method receives a request to populate the menu in the External Data window. Each level in the menu hierarchy will trigger a new request, and the parameters will include the level being requested.

This method should fill in the menu items by calling the AddProperty callback method.

Each item in the menu requires two calls to AddProperty. Each item must use a unique 'index' value as the first parameter.

- id - the id passed in specifies a unique id representing this menu item; the Plug-in could receive this id back in subsequent calls (such as when requesting the next sub-menu level)
- name - the user-displayable name of the menu item

Don't return an error for a blank response. Simply do nothing.

GetTypes

Return a list of known types in the External Provider.

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string A JSON string of parameters. Reserved for future use.

Outputs via Callbacks

AddProperty - 2 values per type to map. Use a unique index value for each mapping:

- AddProperty(index, "id", "artifact");
- AddProperty(index, "name", "Artifact");

[Optional] LogMessage or SetError - to provide user feedback.

Details

Most systems have an ID or non user-friendly name, as well as a user-friendly display name. Only Types that are

returned here will be recognized by Enterprise Architect.

The values returned here will be matched against the Type returned in DefaultTypeMapping and GetItem.

If possible, this list of types should be dynamically created by querying the External Provider for a list of its known types. The list of types can be hard-coded for providers that do not provide a list of types.

ItemLinked

(Requires Enterprise Architect Release 15.2 build 1559 and above.)

Notification that an external item has been linked to an element in Enterprise Architect.

Inputs

parameters	• C++: const char* • C#: string A JSON string of parameters. • itemID - the unique ID of the item (as passed back in GetItemList)

Outputs via Callbacks

[Optional] LogMessage or SetError - to provide user feedback.

Details

This method is called whenever an external item is linked to an element in Enterprise Architect - either creating a new element or linking to an existing element.

This method is part of version 2 of the ISBPIIntegrationPlugin interface. For C++ Plug-ins ensure that CheckVersion handles version 2 correctly.

ItemUnlinked

(Requires Enterprise Architect Release 15.2 build 1559 and above.)

Notification that an external item has been unlinked from an element in Enterprise Architect.

Inputs

parameters	• C++: const char* • C#: string A JSON string of parameters. • itemID - the unique ID of the item (as passed back in GetItemList) • eaElementGUID - the GUID of the Enterprise Architect element

Outputs via Callbacks

[Optional] LogMessage or SetError - to provide user feedback.

Details

This method is called whenever an element in Enterprise

Architect is unlinked from the external item. This can be due to the element being deleted, or through the menu option 'Disconnect from External Object'.

This method is part of version 2 of the ISBPIIntegrationPlugin interface. For C++ Plug-ins ensure that CheckVersion handles version 2 correctly.

ItemUpdated

(Requires Enterprise Architect Release 15.2 build 1559 and above.)

Notification that a linked element in Enterprise Architect has been updated or modified.

Inputs

parameters	• C++: const char* • C#: string A JSON string of parameters. • itemID - the unique ID of the item (as passed back in GetItemList)

Outputs via Callbacks

[Optional] LogMessage or SetError - to provide user feedback.

Details

This method is called whenever a linked element in Enterprise Architect is modified; for example, name change, property modified, notes updated.

This method is part of version 2 of the ISBPIIntegrationPlugin interface. For C++ Plug-ins ensure that CheckVersion handles version 2 correctly.

PostNewDiscussion

Add a new comment/discussion to the External Item

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string A JSON string of parameters, including details of the new comment to be added.

Outputs via Callbacks

[Optional] LogMessage or SetError - to provide user feedback.

Details

This method receives a request to create a new discussion/comment in the External Item. The input parameters string contains information about the comment to create, including:

- "itemID" - the ID of the item to be updated
- "author"

- "comment"
- [optional] parentID - the ID of the parent comment when using threaded comments; this is the ID that was passed back in GetItemDiscussion.

PostNewItem

Creates a new item in the External Provider.

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string A JSON string of parameters. Includes details of the new item to be created.

Outputs via Callbacks

AddProperty - returns the ID of the newly created item. Enterprise Architect will not consider the creation successful unless a valid ID is returned.

- AddProperty(0, "id", "itemX");

[Optional] LogMessage or SetError - to provide user feedback.

Details

This method receives a request to create a new item in the External Provider. The input parameters string contains

information about the item to create, including:

- "title"
- "type"
- "stereotype"
- "description"

PostOAuthCode

Inputs

Parameter	Details
code	• C++: const char* • C#: string The OAuth code value returned to the user after successful authorization.

Outputs via Callbacks

- AddProperty - "accessToken" to be used for OAuth
- [Optional] AddProperty - "refreshToken" for OAuth
- [Optional] LogMessage or SetError - to provide user feedback.

Details

After a successful login to the OAuth provider by the user, Enterprise Architect will call this method with the OAuth 'code'. Make an Access Token Request to exchange the code for an OAuth Access Token (using the OAuth token endpoint).

Return the accessToken and refreshToken with AddProperty.

PostUpdateItem

Updates the selected item in the External Provider. The Notes of the selected item can be updated by the PostUpdateItemNotes method.

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string A JSON string of parameters. Includes details of the item to be updated.

Outputs via Callbacks

[Optional] LogMessage or SetError - to provide user feedback.

Details

This method receives a request to update an item in the External Provider. The input parameters string contains information about the item to create, including:

- "itemID" - the ID of the item to be updated

- "title"
- "type"
- "stereotype"
- "description"

Note: the new values might not actually be different to the current values.

PostUpdateItemNotes

Updates the notes of the selected item in the External Provider.

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string A JSON string of parameters that include details of the item to be updated.

Outputs via Callbacks

[Optional] LogMessage or SetError - to provide user feedback.

Details

This method receives a request to update the notes of an item in the External Provider. The input parameters string contains information about the item to create, including:

- "itemID" - the ID of the item to be updated
- "notes"

RefreshOAuthToken

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string The OAuth refresh token to be used to get a fresh access token.

Outputs via Callbacks

- AddProperty - "accessToken" to be used for OAuth
- [Optional] AddProperty - "refreshToken" for OAuth
- [Optional] LogMessage or SetError - to provide user feedback.

Details

If any method here returns a 401 (Unauthorized) via SetErrorCode then Enterprise Architect will attempt to use the refresh token (if it was supplied) and call this method.

Perform a refresh request against the OAuth refresh endpoint and return the new accessToken and refreshToken.

SetAuthorisation

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string A JSON string of parameters. Available fields are: • username • password • accessToken - OAuth access token

Outputs via Callbacks

[Optional] LogMessage or SetError - to provide user feedback.

Details

This method receives authorisation information that the user enters within Enterprise Architect.

It consists of either username:password credentials or an OAuth access token.

Enterprise Architect will call SetAuthorisation at the start of each session (e.g. open System Integration, open a model, or change user). Use the values provided for any calls to the external provider.

Do not store these as 'static' in any way. Storing as simple members is OK as a new instance of this Class will be created for each new session.

Example Implementation

```
void ExampleIntegrationPlugin::SetAuthorisation(const char* parameters)
{
	LogMessage(LOG_TRACE, __FUNCTION__);

	Json::Value jsonParameters;
	if (strlen(parameters))
	{
		std::stringstream(parameters) >> jsonParameters;
	}

	m_username = jsonParameters["username"].asString();
			// If basic authorisation is used.
	m_password = jsonParameters["password"].asString();
	m_accessToken = jsonParameters["accessToken"].asString();	// If OAuth is used.
```

}

SetCallbacks

C++ only. This only needs to be implemented in C++. The code in the example is sufficient and doesn't need to be modified. Extra error or bounds checking can be added.

Inputs

Parameter	Details
const void ** callbackFunctions	An array of callback function pointers to be used to pass data back to Enterprise Architect.

Outputs via Callbacks

None

Details

C++ Plug-ins will receive this method soon after creation. It passes in an array of callback function pointers which are used by the Plug-in later on to pass data back to Enterprise Architect

Example Implementation

```
void ExampleIntegrationPlugin::SetCallbacks(const void ** callbackFunctions)
{
	if (callbackFunctions)
	{
		AddProperty = (AddPropertyPtr)callbackFunctions[0];
		AddBinaryProperty = (AddBinaryPropertyPtr)callbackFunctions[1];
		SetErrorCode = (SetErrorCodePtr)callbackFunctions[2];
		SetError = (SetErrorPtr)callbackFunctions[3];
		LogMessage = (LogMessagePtr)callbackFunctions[4];
	}
}
```

SetConfiguration

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string A JSON string of parameters. See Details for more information about available parameters.

Outputs via Callbacks

[Optional] LogMessage - set log messages about the configuration settings received. Be careful not to log sensitive information.

Note: SetError callbacks will be ignored for this method.

Details

This method receives the details that the user inputs into the Pro Cloud Server configuration when enabling this Custom Integration Plug-in.

It includes these details:

External Server to connect to:

- serverName
- serverPort
- serverProtocol
- baseURL - the url folder to be appended to the url

The server settings combine to form a URL as such: <protocol>://<serverName>:<serverPort>/<baseURL>.

Hardcoded Credentials - These are optional and can be used to connect to a provider with a generic account:

- username
- password

Permissions - sets whether users can perform the specified actions on the external provider:

- allowCreateItems
- allowModifyItems
- allowPostDiscussions

Proxy settings:

- proxyServer
- proxyBypass
- proxyUsername
- proxyPassword

Example Implementation

void ExampleIntegrationPlugin::SetConfiguration(const char* parameters)

```
{
	LogMessage(LOG_TRACE, __FUNCTION__);

	Json::Value jsonParameters;
	if (strlen(parameters))
	{
		std::stringstream(parameters) >> jsonParameters;
	}

	// Store the settings as member variables for later use.
	m_serverName =
jsonParameters["serverName"].asString();
	m_serverPort =
jsonParameters["serverPort"].asString();
	m_serverProtocol =
jsonParameters["serverProtocol"].asString();
	m_baseURL = jsonParameters["baseURL"].asString();

	m_settingsUsername =
jsonParameters["username"].asString();
	m_settingsPassword =
jsonParameters["password"].asString();

	m_allowCreateItems =
jsonParameters["allowCreateItems"].asString();
	m_allowModifyItems =
```

```
jsonParameters["allowModifyItems"].asString();
        m_allowPostDiscussions =
jsonParameters["allowPostDiscussions"].asString();


        m_proxyServer =
jsonParameters["proxyServer"].asString();
        m_proxyBypass =
jsonParameters["proxyBypass"].asString();
        m_proxyUsername =
jsonParameters["proxyUsername"].asString();
        m_proxyPassword =
jsonParameters["proxyPassword"].asString();
}
```

Custom SBPI Services

Custom SBPI Services are user-defined integration Plug-ins that can be invoked by Enterprise Architect's scripting or Add-ins, which can then use the responses to manipulate the repository data. The Custom Plug-In can be called from Enterprise Architect using the Repository.CallSBPI automation interface method.

The scope of what can be requested of the service and what it can return to Enterprise Architect is not limited. For example, the Custom Service could respond to a request to gather information from a third-party service, and return it to the Enterprise Architect script. Other possible uses include performing actions on the model data itself via OSLC calls, or running custom-built processes on model data.

Benefits

- Allow arbitrary requests and responses
- Once configured for a model, can be called from scripting or Add-Ins
- Lifetime and request forwarding automatically handled by Pro Cloud Server
- Can be written in multiple programming languages, including C++ or C#

Overview

To write your own Custom Service Plug-in you can either start from scratch or make a copy of one of the examples and modify it.

When installing the Pro Cloud Server, enable the 'SBPI Examples' component to include the custom service examples. When enabled, the default location for the example files is within the 'SBPI Examples\ExampleServicePlugins' folder. For example:

C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\SBPI Examples\ExampleServicePlugins

See the Pro Cloud Server Installation Help topic for more information.

Note, the 'SBPI Examples' installation option is not enabled by default. If you have already installed the Pro Cloud Server without the 'SBPI Examples', you can either perform a full re-install (enabling the 'SBPI Examples'), or use the installer's 'Change' option to add just the 'SBPI Examples' component.

The Plug-ins can be written in either C++ or C#.

The examples are written using Visual Studio 2017, but this is not a pre-requisite.

The Custom Service Plug-in must implement the interface defined in ISBPIServicePlugin, which is included in ISBPIServicePlugin.h (for C++) or ISBPIServicePlugin.cs (for C#).

The general flow of the program is:

- The user performs an action within Enterprise Architect

that needs information from the Service Plug-in via the CallSBPI automation interface
- The Plug-in receives the request
- The Plug-in parses the request and performs any action required (for example, call third-party service, run program, make OSLC calls to Pro Cloud Server)
- The Plug-in sends a response to Enterprise Architect via the provided callback functions; this can either be the actual data requested or an error value
- Enterprise Architect receives the callback data and uses it in the script or Add-in

Interface

Function/Class	Details
Create Plug-in (not required in C#)	The Plug-in must implement this export function: extern "C" SBPI_SERVICE_API SBPI_SERVICE_PLUGIN CreatePlugin(); It must return a pointer to a Class that implements the ISBPIServicePlugin interface. The recommended implementation is: SBPI_SERVICE_PLUGIN

	CreatePlugin()
	{
	return new ExampleServicePlugin;
	}
	The newly created ISBPIServicePlugin can be deleted when it receives the ISBPIServicePlugin::Release method.

ISBPIService Plugin interface	The dll Plug-in must implement all methods in the ISBPIServicePlugin interface.

ISBPIServicePlugin interface

ISBPIServicePlugin interface methods

Method Name	Notes
Release	C++ only. This is called by the controlling application when the interface class is no longer required. The method should delete the class created during the CreatePlugin() function.
SetCallbacks	C++ only. Passes in an array of callback function pointers that are used by the Plug-in later on to pass data back to Enterprise Architect.
SetConfiguration	Receives the settings defined when the user sets up the Custom Service Plugin in Pro Cloud Server.
HandleRequest	Generic request from Enterprise Architect. The plug-in can perform any action required and return data or error-codes back to Enterprise Architect via the callback methods.

HandleRequest

Inputs

method	• C++: const char* • C#: string The name of the method to perform. This allows the plugin to make a choice on what action to perform without having to parse the parameters string fully first.
parameters	• C++: const char* • C#: string A JSON string of parameters. See Details for more information about available parameters.

Outputs via Callbacks

[Optional] Result, LogMessage or SetError - to provide user feedback.

Details

This is the main function to the plug-in. It should handle all

incoming 'method' requests and perform the action required, and return any data via the 'Result' callback.

Example Implementation

```
void ExampleServicePlugin::HandleRequest(const char* method, const char* parameters)
{
   LogMessage(LOG_TRACE, std::string(__FUNCTION__ + " - Method = "s + method).c_str());

   // This example method demonstrates how to extract various parameter types from parameters.
   // This is done here with jsoncpp library but can be done with any compliant JSON library.
   Json::Value jsonParameters;
   if (strlen(parameters))
   {
      std::stringstream(parameters) >> jsonParameters;
   }

   if (std::string(method) == "DoSomething")
   {
      int myNumber = jsonParameters["myNumber"].asInt();
      double myFloat = jsonParameters["myFloat"].asDouble();
```

```
        std::string myString =
jsonParameters["myString"].asString();
        std::list<int> myArrayOfNumbers;
        for (auto& myValue :
jsonParameters["myArrayOfNumbers"])
        {
            myArrayOfNumbers.push_back(myValue.asInt());
        }
        std::list<std::string> myArrayOfStrings;
        for (auto& myValue :
jsonParameters["myArrayOfStrings"])
        {
            myArrayOfStrings.push_back(myValue.asString());
        }

        std::string result = "Example User SBPI Service Plugin
in C++. DoSomething received parameters: myNumber = "
+ std::to_string(myNumber)
            + ", myfloat = " + std::to_string(myFloat)
            + ", myString = " + myString;

        // Set the result string.
        Result(result.c_str());
    }
    else if (std::string(method) ==
"DoSomethingToElement")
```

```
    {
        // This example method demonstrates how to respond
to a user performing a task on a single element.
        std::string elementID =
jsonParameters["elementGUID"].asString();


        Result(std::string("Example User SBPI Plugin in C++.
DoSomething to element with GUID: " +
elementID).c_str());
    }
    else
    {
        // Set an error string
        SetError(std::string("Unknown method: " +
std::string(method)).c_str());
    }
}
```

SetCallbacks

C++ only. This only needs to be implemented in C++. The code in the example is sufficient and doesn't need to be modified. Extra error or bounds checking can be added.

Inputs

Parameter	Details
const void ** callbackFunctions	An array of callback function pointers to be used to pass data back to Enterprise Architect.

Outputs via Callbacks

None

Details

C++ Plug-ins will receive this method soon after creation. It passes in an array of callback function pointers which are used by the Plug-in later on to pass data back to Enterprise Architect

Example Implementation

```
void ExampleServicePlugin::SetCallbacks(const void **
callbackFunctions)
{
   if (callbackFunctions)
   {
      Result = (ResultPtr)callbackFunctions[0];
      SetErrorCode =
(SetErrorCodePtr)callbackFunctions[1];
      SetError = (SetErrorPtr)callbackFunctions[2];
      LogMessage = (LogMessagePtr)callbackFunctions[3];
   }
}
```

SetConfiguration

Inputs

Parameter	Details
parameters	• C++: const char* • C#: string A JSON string of parameters. See Details for more information about available parameters.

Outputs via Callbacks

[Optional] LogMessage - set log messages about the configuration settings received. Be careful not to log sensitive information.

Note: SetError callbacks will be ignored for this method.

Details

This method receives the details that the user inputs into the Pro Cloud Server configuration when enabling this Custom Integration Plug-in.

It includes these details:

- Custom item/value pairs of settings entered in 'Custom Properties' section.

Proxy settings:

- proxyServer
- proxyBypass
- proxyUsername
- proxyPassword

Example Implementation

```
void ExampleServicePlugin::SetConfiguration(const char* parameters)
{
   LogMessage(LOG_TRACE, std::string(__FUNCTION__).c_str());

   Json::Value jsonParameters;
   if (strlen(parameters))
   {
      std::stringstream(parameters) >> jsonParameters;
   }

   for (auto& myProperty : jsonParameters.getMemberNames())
   {
      m_properties[myProperty] =
```

```
jsonParameters[myProperty].asString();
    }
}
```

Example Script

This JavaScript script details how to send a simple request to a Custom Service plugin:

```
!INC Local Scripts.EAConstants-JavaScript

/*
* Script Name: Custom Service Example
* Author: Sparx Systems
* Purpose: Demonstrate the use of the SBPI automation interface for Custom Service plugins
* Date: 2022-02-28
*/

// Sends a simple request to the plugin with some parameters.
function SimpleRequest()
{
	// Show the script output window
	Repository.EnsureOutputVisible( "Script" );
	Session.Output("JavaScript Custom Plugin EXAMPLE");

Session.Output("================================================");
```

```
	// Send data with the request by adding parameters using InsertSBPIParameter.
	var packedParameters = '';

	// Optional data to send with extra parameters
	packedParameters = Repository.InsertSBPIParameter(packedParameters, 'myNumber', 25);
	packedParameters = Repository.InsertSBPIParameter(packedParameters, 'myFloat', 123.456);
	packedParameters = Repository.InsertSBPIParameter(packedParameters, 'myString', 'Hello World');

	Session.Output("Sending simple request to plugin to 'DoSomething' method");
	var response = SBPIRequest('csvc', 'DoSomething', packedParameters);
}

// Helper function to send a request to the Custom plugin and check for errors.
function SBPIRequest(prefix, method, packedParameters)
{
```

```
	// Specify the prefix of the plugin. This is configured in the Pro Cloud Config Client.
	var response = Repository.CallSBPI(prefix, method, packedParameters);
	if (response == '')
	{
		Session.Output('Error from plugin: ' + Repository.GetLastError());
	}
	else
	{
		Session.Output('Success: ' + response);
	}

	return response;
}

function main()
{
	// Sends a simple request to the plugin with some parameters.
	SimpleRequest();
}

main();
```

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 what rows of data an individual database user can access. Traditional database security only provides the ability to control the permissions of a user at the table level; that is, can they read, write or delete data in a particular table? 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 database queries. The database ensures that users can only view and update data they are authorized to access.

Given that Row Level Security is 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 perspective.

Database / Application Users

It is important to understand the distinction between

database users and application users. The introduction states that visibility levels are based on Database User permissions. In the Sparx Systems Pro Cloud Server environment, each Database Manager is defined with all of its connection details (including the user ID and password) to a given database. It is the database user, the user identified in the connection details, that visibility levels are based on.

This concept is not to be confused with Enterprise Architect model users; that is, 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 DBMSs require a database user to be defined in order to access data within the database.

How it works

In addition 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 column stores the visibility level that is required to view or update the corresponding 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 Browser window 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 (that is, 1 to 20); however, this can easily be adjusted by manually 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, representing the default value; it implies that all database users (even those without an assigned visibility level) can view or update records of this level.

The Drawback

As with most things, enabling Visibility Levels entails a cost when compared to the same model without Visibility Levels being enabled. Row Level Security in both SQL Server and Oracle imposes extra processing on the database server each time data is selected. The impact of this extra processing on database performance is difficult to estimate, as it depends on the type and amount of data being selected. Enterprise Architect's support of Visibility Levels has been designed with efficiency in mind; however, the performance will inevitably be reduced when Visibility Levels are enabled.

Installation and Configuration

The Sparx Systems Pro Cloud Server product is installed using a standard Microsoft Windows Installer package that includes a number of optional components. One of these components is Visibility Levels, which is installed by default into the C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\VisibilityLevels\ folder (assuming a 64 bit machine is being used and the 32 bit version of Pro Cloud Server). If necessary, you can change the install location of the Visibility Levels files.

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 repository that uses the EA1220 schema definition (or later)
- An SQL-based management console tool that supports the execution of SQL scripts
- Enterprise Architect build 14.0 or later
- Enterprise Architect Ultimate Edition for assignment of Visibility Levels on a Package

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 repository so that it can support the DBMS's native Row Level Security feature.

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 sufficiently different for each to need its own creation script.

Task	Script
Apply Visibility Level changes to Oracle repository	EASchema_VL_Updates_Oracle.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 repository	EASchema_VL_Updates_SQLServer.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	EASchema_VL_Removal_Oracle.sql - Oracle SQL script that will remove the

Level changes from Oracle repository	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, however all visibility level permissions will be permanently lost.
Remove Visibility Level changes from SQL Server repository	EASchema_VL_Removal_SQLServer.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, however all visibility level permissions will be permanently lost.

How to execute

In order for an Enterprise Architect repository to support Visibility Levels, the appropriate update script must be executed on that repository's database. The script is standard SQL, so any tool can be used; however, you must connect

using an ID with database owner privileges because updates will be made to the physical schema.

Script Execution for Oracle

1. Identify the Enterprise Architect repository database to which to apply Visibility Levels, ensure database schema is at least EASchema1220 and if not update it according to the instructions on DBMS Based Repositories web page.
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.sql') into the console.
7. Review the comments at the top of the script, in particular the section that outlines the tasks that must be performed before you run the script.
8. Make the required 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 Ultimate Edition, connect to the Pro Cloud model with the highest Visibility Level (as defined in step 12) and set the Visibility Levels on all Packages to their required values.

Script Execution for MS SQL Server

1. Identify the Enterprise Architect repository database to which to apply Visibility Levels; ensure the database schema is at least EASchema1220 and if not update it according to the instructions on DBMS Based Repositories web page.
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.sql').
7. Review the comments at the top of the script, in particular the section that outlines the tasks that must be performed before you run the script.
8. Make the required 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 Ultimate Edition, connect to the Pro Cloud model with highest Visibility Level (as defined in step 12) and set the Visibility Levels on all Packages to their required values.

User Guide

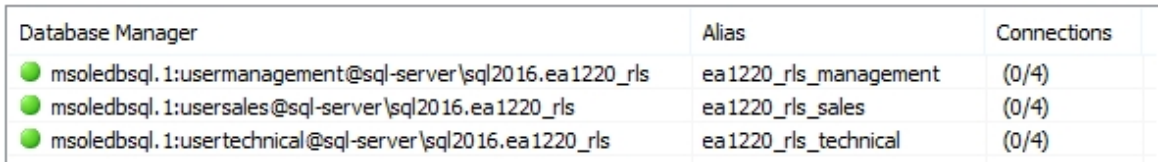
The functionality of the Visibility Levels feature is primarily restricted to System Administrator users in Enterprise Architect, in order to set up and configure models to limit access to certain Packages in a model. Generally, users will not be aware of Visibility Levels; they simply will not see certain Packages in the model.

Defining Visibility Level Aware Database Managers

For the most part, the process of defining Database Managers in the Sparx Systems Pro Cloud Server Configuration Client is the same as the process for non Visibility Level Database Managers. However, to facilitate different user groups being able to access different sections of the repository, a separate Database Manager will be needed for each user group, with the database user defined within the Database Manager's connection string controlling what areas of the repository the Enterprise Architect user has access to.

Consider the scenario of an environment that consists of three user groups: Management, Sales and Technical. Each user group will have one or more Packages within the repository for their exclusive use; members from the other two groups will not be able to view the contents of these Packages. Each user group will require their own Database Manager defined on the Pro Cloud Server and therefore

must be provided with the appropriate connection string to their Database Manager. This means that the Pro Cloud Server connection strings should be kept private and only shared with the appropriate group of users. Note: the use of encrypted connection strings should be considered.



Database Manager	Alias	Connections
msoledbsql.1:usermanagement@sql-server\sql2016.ea1220_rls	ea1220_rls_management	(0/4)
msoledbsql.1:usersales@sql-server\sql2016.ea1220_rls	ea1220_rls_sales	(0/4)
msoledbsql.1:usertechnical@sql-server\sql2016.ea1220_rls	ea1220_rls_technical	(0/4)

Differences between a Traditional and a Visibility Level Aware Database Manager are outlined here:

DBMS
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 repository 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 repository using the Visibility Levels feature these options are different, so the 'DefaultSchema' option has been added to define the table owner.

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 repository.

In previous versions (pre version 2) of the Pro Cloud Server, 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 Pro Cloud Server. 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 Pro Features (OSLC, WebEA and Integration)' option must be selected. This is achieved by opening the Sparx Systems Pro Cloud Server 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 Pro Features (OSLC, WebEA and Integration)' checkbox.

Visibility Level Administration

When the Ultimate Edition of Enterprise Architect is used to connect to a model that has Visibility Levels configured, there is an additional menu item, 'Set Visibility Level...' available under the 'Package Control' context menu of each Package object in the Browser window.

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' or 'Administer Database' security permission, using standard Enterprise Architect permission assignment.

Set a Package's Visibility Level

From within the Ultimate Edition of Enterprise Architect, the Visibility Level of a Package can be set using the Browser window 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,
- are being accessed via a Pro Cloud server connection and
- where the user has an Enterprise Architect Ultimate License.

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.

To remove a visibility level from a Package (and all its

children), simply set the Visibility Level back to unrestricted.

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, an 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 unrestricted, 3 or 4.

Troubleshooting

'Set Visibility Level' menu is 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
- You are not using Enterprise Architect release 14 (or above)
- You are not using the Ultimate Edition of Enterprise Architect
- The Cloud Server you are connected to 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 Pro Features (OSLC, WebEA and Integration)' option
- The model you are connecting to has not been configured/adjusted for Visibility Levels (that is, 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

Pro Cloud Server Standard Features

The Pro Cloud Server provides a number of features regardless of any licensing, including the facility to connect to models over secure http connections (HTTPS). These core features include:

- HTTP(S) Connectivity
- Reusable Asset Service (RAS)
- Floating Licensing
- Scheduled Tasks

Feature	Description
HTTP(S) Connectivity	A fundamental feature of the unlicensed Pro Cloud Server is the ability to connect Enterprise Architect on your workstations to a repository via HTTP or HTTPS. This can easily be extended to include off-site connections to branch offices, off-site workers or outsource vendors. The external connections can use HTTPS connections, via a firewall or using more dedicated VPN-style connections. A core feature is the network data-compression, providing improved performance for remote access across a high-latency WAN.

Reusable Asset Service (RAS)	The RAS portion of the Pro Cloud Server helps you to define Packages that can be used in any model. When a Package is registered, Enterprise Architect and the Pro Cloud Server will track cross-Package dependencies and save that Package in a RAS storage. This makes everything required by that Package available to any repository that has access to the Pro Cloud Server hosting that RAS. The RAS service can also host Baselines, as Baselines created for Package-trees can be stored within the current model or be published via the RAS.
Floating License Server	The Floating License Server provides a mechanism to allocate and share Enterprise Architect and other Sparx System Product licenses between multiple users. Licenses are only allocated to users when they open Enterprise Architect; once the application is closed the license is released back into the pool and is available to another user. Therefore, this mechanism of sharing licenses can allow an organization to purchase a smaller number of licenses and make more efficient use of the licenses purchased.

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

- Connecting to Pro Cloud Server based repositories requires Enterprise Architect Corporate, Unified or Ultimate Edition.

PCS Troubleshooting

Browser Test

For a simple check that the Pro Cloud Server 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, using the default settings on the machine with Pro Cloud Server installed:

http://localhost:1804/

Or

https://sparxcloud.com:1804/

If the Port is enabled this message is returned:

Sparx Systems Pro Cloud Server vX.X.XX

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 name and Port, check for other services listening on the specified Port, check the server log file for error messages, and check any firewalls between the client and the server to confirm there is connectivity and that the relevant Port(s) are open.

Connection Test

Where you have issues with the speed of response for an

existing connection to a Pro Cloud Server server, then a simple initial check is to use the Pro Cloud Server Connection Test. This test is accessible from:

- Ribbon: Start > Help > Help > Pro Cloud Server Connection.

It returns the details in the System Output window in this format:

- The Pro Cloud Server version number
- The Ping response time in milliseconds from the current machine to the Pro Cloud Server Machine
- The timing to complete a simple query.

For example:

- Pro Cloud Server: Version '5.0.100'
- Pro Cloud Server: Response time: 26 ms
- Pro Cloud Server: Simple database query execution time: 23 ms

Where the Ping test is high, then the general connectivity, including the WAN connection or proxy servers, must be reviewed.

Where the Ping test is in a reasonable range, but the query execution is high, then there is reason to check the PCS connection to the DBMS or the response of the DBMS itself.

Errors shown in Enterprise Architect System Output

Error Message	Meaning
Http Status Code: 503 Service Unavailable	Indicates that the Pro Cloud Service is not running on the Server. • Check you have the correct Server Name and Port • Check that the 'Sparx Systems Professional Cloud' Service is currently running on the server
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. If the model in question had user security enabled after the 'Require authenticated and secure connection' option was enabled in the Pro Cloud Server get the system administrator to turn off and then back on again the 'Require secure and authenticated connection' option, making sure to save the Database Manager each time. This will reset the internal flag that Pro Cloud Server retains so that it can validate user credentials.
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 Pro Cloud Server, 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 manager in question.
There is no database manager configured for the requested repository	Enterprise Architect established a connection to a Pro Cloud Server, but the model name requested does not match any defined database on the server. When accessing the model using the Enterprise Architect Cloud Connection screen, ensure that the Model Name entered matches the Alias that is shown in the Pro Cloud Server Configuration Client. For the Alias see the *Configure Database Manager* Help topic.
Unable to connect to Database	Enterprise Architect established a connection to a Pro Cloud Server 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 Pro 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 (and 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 Pro Cloud Server service is running and 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
12007 - The server name or address could not be resolved	Indicates the server name (or IP address) is incorrect. • Check that the server name is correct • Attempt the Browser Test described at the start of this topic
12029 - A connection with the server could not be established	This could indicate that the Pro Cloud Server is not installed, the Service is not running, or an incorrect Port was entered. • Confirm the Pro Cloud Server is installed • Check that the 'Sparx Systems Professional Cloud' Service is running • Check that you entered a valid Port (i.e. it should match one defined in the Configuration Client) • Attempt the Browser Test described at the start of this topic
12031 - The connection with the server was reset	This can occur if you are attempting to connect with the Protocol HTTP, but the Port you have entered is configured for HTTPS (e.g 805). • If you have selected the Protocol

	HTTP, confirm that the Port which you entered is configured for HTTP access • If you have selected the Protocol HTTPS, confirm that the Port which you entered is configured for HTTPS access See the *Port Configuration* Help topic for details.
After Reinstall Enterprise Architect no longer works	Your environment was working correctly, then you updated to the latest version of the Pro Cloud Server and now Enterprise Architect clients are not able to connect to your Cloud models. During the reinstall process the Pro Cloud Server installer actually uninstalls the original version and then installs the new one. As a result of this the Firewall exception that is automatically added by the installer is removed and re-created. If the default Firewall exception had been adjusted (after the original install) to suit your environment, these changes would have be lost by the re-install process. One method of stopping this from occurring in the future is to create a completely new firewall exception (with a different name) that is a copy of the one created by the installer, and make any manual changes

	to that exception.

Reading Server Logs

As mentioned previously, the Sparx Systems Pro Cloud Server is a Windows Services application, and as such it needs to conform to some basic rules. Troubleshooting Windows Services can be problematic, since they run independently of all user input/interaction and as such do not contain screens/GUI of any kind. For this reason the Sparx Systems Pro Cloud Server writes any potential message to Log files.

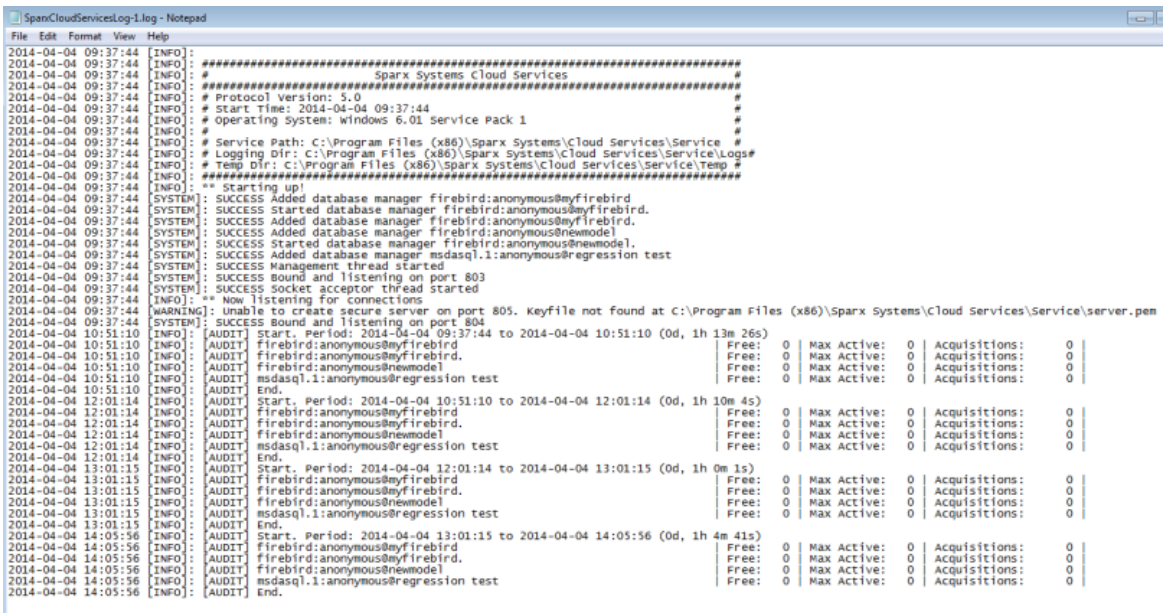
The Pro Cloud Server configuration file contains a number of settings that control where, how many and how large log files can be. By default the Pro Cloud Server will save all log files to a \Logs\ folder under:

\Services\ (C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service\Logs)

It will keep up to three files. A new file will be created whenever the Cloud Service is started or whenever the maximum file size if reached. Filename 'SSProCloudServerLog-1.log' will always be the most recent log file and each time a new file is created the older ones are renamed by increasing the number. For example, SSProCloudServerLog-2.log is renamed to SSProCloudServerLog-3.log.

The configuration file also contains a setting that controls

the level of logging that is performed; see the logging related options in the *Extended Server Properties* Help topic for more details. When troubleshooting, it is recommended that the Pro Cloud Server LOG_LEVEL property be set to the highest level, SYSTEM.



Each log entry will consist of the Time, Log Level and Message; if an error occurs the Message will normally contain a description of the problem and from this description the appropriate action can be undertaken.

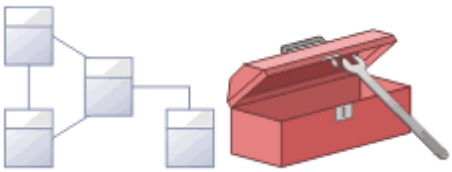
Log Message	Explanation
Log File Header	Each time new Log file is created the Pro Cloud Server will write out a summary of the main settings; this information is particularly useful if you are requesting support from Sparx Systems.
Started database	When a Pro Cloud Server is initially started, and the Log Level is set to INFO

managers	(or lower) it will log out a summary of each configured model that includes the connection string of the model. *[INFO]: Added database manager - ConnectionStr: 'eaexample --- Connect=Provider=MSDASQL.1;...* If the Log level is lower than INFO, additional log entries will be written for each model; for example: *[SYSTEM]: SUCCESS Started database manager firebird:anonymous@eaexample*
House keeping	The Pro Cloud Server performs regular 'house keeping' tasks, the frequency of which is controlled by the AUDIT_TIME_PERIOD setting in the Pro Cloud Server configuration file. When the log level is set to INFO, each time house keeping is performed the Pro Cloud Server will log out a summary of connections for each defined database manager and report how many are currently available, the maximum number used concurrently, and the total number of uses since the last audit. *[AUDIT] firebird:anonymous@eaexample | Free: 0 | Max Active: 3 | Acquisitions:*

	192 | Normally these entries can be ignored as they are purely reference information only.
Open Port list	When a Pro Cloud Server is initially started, and the Log Level is set to SYSTEM, it will log out a success message for each configured Port, which includes the management Port (803) and any http/https Ports. You should see a line such as: *[SYSTEM] SUCCESS Bound and listening on port 804.* 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: • Certificate/Private Key file not found at C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service\server.pem • The server.pem file does not contain a valid certificate and private key

	• Certificate Authority file not found or invalid at C:\Program Files (x86)\Sparx Systems\Pro Cloud Server\Service\cacert.pem
Unable to connect to database	During general operations, the Pro Cloud Server will write entries to its log file, which include any failed attempts to open a model. This will generally not occur until a user attempts to connect to the database. Errors that can occur here include: *REQUEST_CONNECT FAIL. Error (5): Unable to connect to database* This means that the server attempted to open a connection but it failed. If you see this message, review the list of potential reasons outlined in the *Errors shown in Enterprise Architect System Output* table.

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, and can be shared between repositories, ensuring consistency. The Browser window allows repositories to be refactored easily by dragging and dropping Packages, elements, features and diagrams to new locations. The Discussion and Library features allow the model to be critically analyzed and reviewed.

Facilities

Facility	Detail
Check the integrity of the data in a	If you have a failed XMI import, network crash or other unforeseen event that could disrupt the integrity of information in the

project	model, it is recommended that you run a Project Integrity Check.
Reset Auto Increment Columns in Tables	XML Import and Export can affect the table auto increment column values and push them towards their maximum allowed 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 file-based project, you can do so through Windows Explorer.
Compact a project	As with many relational database products .QEA/.QEAX and .EAP/.EAPX files do not physically release allocated space after data is deleted, therefore from time to time it can be beneficial to perform a 'compact', to reduce the physical size of the project file.

Repair a project if it did not close properly	If a .EAP/.EAPX project has not been closed properly, such as during system or network outages, it is possible that the project will report inconsistency errors the next time it is opened. To resolve such issues a 'Repair' can be performed.

Notes

- You can only rename, compact and repair models created as .EAP/.EAPX files with Enterprise Architect; while most DBMS products provide similar functionality, they typically require an elevated level of permissions and therefore are usually performed by your Database Administrator (DBA).

Maintenance Overview

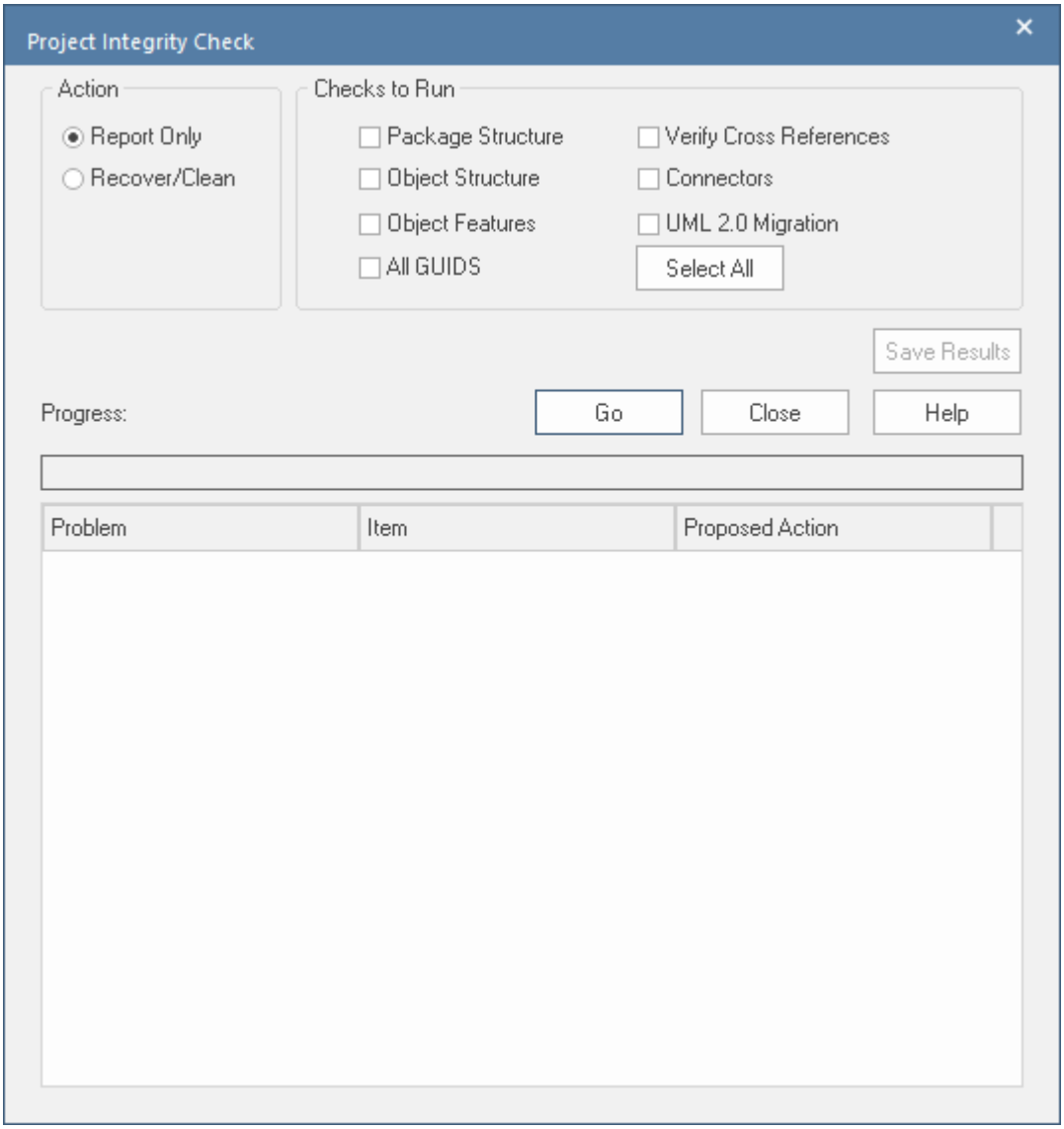
An Enterprise Architect repository is continuously evolving as you add new items or change or delete existing items. During this evolution events can occur that have some minor internal impact on the integrity of the records stored in the database. New product enhancements are also made as a result of the company's strong commitment to innovation and from feedback from customers. These enhancements need to be incorporated into existing product installations so that customers continue to derive the benefit and value from the tool. In this topic you will learn about some of the remedial functions that can be used to keep you repository in good health and operating at optimal speed, as listed here:

- Check Project Data Integrity - perform a health check
- Reset Auto Increments - reset the auto-increment column on system tables
- Upgrade a Project - augment the files to support new features or enhancements
- Run SQL Patches - run a SQL script to correct or enhance a database
- Rename a Project - change the name of a project file
- Compact a Project - decrease the size of a file based database
- Repair a Project - correct a problem due to file corruption from an unexpected network shutdown

Check Data Integrity

As the models in your repository evolve over time, small aberrations can occur in the database, typically as a result of badly formed XMI imported from third-party systems, network crashes or other unforeseen events that disrupt the integrity of information in the repository. To ensure your repository is in good-health 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 or other modeling language conformance, only the data relationships and repository structure. So it will check that all attributes have identifiers but it will NOT check that you have used the correct relationship between a Use Case and an Actor.



You can conveniently run the Integrity Checker first in 'Report Only' mode to discover and analyze problems it has located. Once you have looked through the list of problems that the check has found and viewed the proposed actions you can then run it again in 'Recover/Clean' mode.

Other than for quite small repositories it is recommended that you don't run all the available checks together as this can take some time to complete. Either run them individually or in small sets.

When Enterprise Architect 'Cleans' the model, it attempts to recover any lost Packages and elements, and generates a

new Package called _Recovered_. Check through any elements that are found and, if required, drag them back to where they belong in the Browser window. If they are not required, you can choose to delete them or export them as an XMI export to serve as a backup.

Access

Ribbon	Settings > Model > Integrity > Project Integrity

Check the data integrity of your project

Step	Action
1	Open the project.
2	Select the 'Project Integrity' 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 Alternatively, to select all checks together, click on the Select All button. This does not select the 'UML 2.0 Migration' option. Do not run the migration with any other check; this is a facility that should be run by itself. (See the *UML 2.0 Migration* section after this table.)
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
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, Unified and Ultimate Editions of Enterprise Architect, if security is enabled you must have 'Check Data Integrity' permission to perform a data integrity check

Reset Auto Increments

Enterprise Architects models are natively stored in tables in a relational database. It uses the auto-increment feature of these database system when inserting new rows in its system tables. There are a number of operations such as bulk imports and exports from XML files that can cause a table's auto-increment column to approach its largest allowed value. This topic explains how auto-increment issues can arise and what is the best practice for re-sequencing the identifier values in any affected tables.

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 largest previous 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 QEA or EAP replica or design master. The SQLite and Jet engines assign 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 QEA or EAP project is transferred to another repository.

Access

Ribbon	Settings > Model > Transfer > Full Project Transfer via Native XML > Export Complete Project Settings > Model > Transfer > Full Project Transfer via Native XML > Import Complete Project

Re-sequence auto increment columns

Step	Description
1	Open the project.
2	Create a base model of the repository-type that you want to export to.
3	• Select: Settings > Model > Transfer > Full Project Transfer via Native XML > Export Complete Project • Set the *Directory* for where the XEA export is to be stored.
4	Open the newly created repository.
5	• Select: Settings > Model > Transfer > Full Project Transfer via Native XML > Import Complete Project • Ensure the *Directory* is set to the same file-directory where the XEA was exported above.

Notes

- It is recommend that this task is performed when there are no active users accessing the Project.

Upgrade a Project

The structure of Enterprise Architect project files is occasionally changed to support new 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

Enterprise Architect is a robust platform and there are many internal checks that are applied to ensure that the integrity of the information in the repository is maintained. Occasionally an issue might arise and Sparx Systems might choose to release a patch to rectify the problem.

The patch generally checks how many records are to be updated, and reports on what is to be done.

Access

Ribbon	Settings > Model > Integrity > Run Patch

Run SQL Patch

1	Open Enterprise Architect and load the Project that requires the patch applied.
2	Select the 'Run Patch' option. The 'Run Patch' dialog displays.

3	Click the Browse button and select the supplied patch file (*.XML) in the file browser. Upon loading the select file, the Description will then be loaded with the explanation of what action the patch will perform.
4	Click the Run button to apply the patch to the current Project.

Rename a Project

Occasionally the purpose of your repository will change, or you might for any number of reasons want to change its name. While you can change the name of the root Package in the repository to change the name of the repository itself, you will need to rename the file 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 .QEA/.QEAX Project

After a period of time, through general use, a .QEA or .QEAX project 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.

Enterprise Architect provides two separate but similar options to compact a .QEA or .QEAX repository:

1. Run the compact process against the selected project
2. Compact the selected project to a new file, this option keeps the original repository unchanged

Access

Ribbon	Settings > Model > Integrity > Manage .QEA/.QEAX File > Compact .QEA/.QEAX File Settings > Model > Integrity > Manage .QEA/.QEAX File > Compact .QEA/.QEAX to new file

Compact a .QEA/.QEAX file

Ste	Action

p	
1	Identify a .QEA/.QEAX project file to compact
2	Ensure no other users (or other instances of Enterprise Architect) have the identified .QEA/.QEAX project .file open.
3	Select the 'Compact .QEA/.QEAX File...' option.
4	When prompted, select (or enter) the filename of the .QEA/.QEAX project file to compact.
5	Follow the on-screen instructions to complete the process. Once the process completes Enterprise Architect will display a message indicating both original and compacted sizes.

Compact a .QEA/.QEAX file to new file

Step	Action
1	Identify a .QEA/.QEAX project file to compact
	Ensure no other users (or other instances of

2	Enterprise Architect) have the identified .QEA/.QEAX project .file open.
3	Select the 'Compact .QEA/.QEAX to new file...' option.
4	When prompted, select (or enter) the filename of the .QEA/.QEAX project file to compact.
5	When prompted, enter the file name of the new file; this will be the file that receives the compacted version of the selected repository.
6	Follow the on-screen instructions to complete the process. Once the process completes Enterprise Architect will display a message indicating both original and compacted sizes.

Notes

- Always compact projects on a local drive, never on a network drive
- In the Corporate, Unified and Ultimate Editions of Enterprise Architect, if security is enabled you must have 'Administer Database' permission to compact a project

Compact a .EAP/.EAPX Project

After a period of time, through general use, a project .eapx or .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	Settings > Model > Integrity > Manage .EAPX/.EAP File > Compact .EAPX/.EAP File

Compact a .EAPX/.EAP File

Step	Action
1	Either: • Open the .EAP or EAPX project to compact or • Ensure that there is no project open (you will be prompted after step 3)
2	Ensure no other users have the file open.

3	Select the 'Compact .EAPX/.EAP File...' 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, Unified and Ultimate Editions of Enterprise Architect, if security is enabled you must have 'Administer Database' permission to compact a project

Re-Index a Firebird Project

After a period of time, through general use, a Firebird-based project file can develop non-optimized indexes, causing data retrievals to be slow and affecting the general performance of Enterprise Architect while using that repository.

To resolve such issues you can move the project to a local drive and request the database engine to rebuild all indexes the next time they are used; this is referred to as Updating the Statistics of the Indexes.

Access

Ribbon	Settings > Model > Integrity > Manage Firebird > Update Index Statistics

Index a Firebird Project

Step	Action
1	Open the Firebird project to re-index.
2	Ensure no other users have the file open.

3	Select the 'Update Index Statistics...' option.
4	Follow the on-screen instructions to complete the process.

Notes

- The process to re-index a project should always be performed on a local drive, never on a network drive
- Over the years Sparx Systems have improved the default indexes that are defined within an Enterprise Architect repository, so whenever a new .EAP/EAPX is created it will be created with the latest indexes; however, if a .EAP/.EAPX repository has been in use for many years then the defined indexes are most likely not ideal, but you can use another ribbon option - 'Update .EAP and .EAPX file indexes' - to remove all old indexes and generate the latest index definitions
- In the Corporate, Unified and Ultimate Editions of Enterprise Architect, if security is enabled you must have 'Administer Database' permission to re-index 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	Settings > Model > Integrity > Manage .EAPX/.EAP File > Repair .EAPX/.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	Close and restart Enterprise Architect - you do not need to open any model, including the one you are repairing.
4	Select the 'Repair .EAPX/.EAP File...' menu option, and follow the on-screen instructions.
5	Once you have repaired the project, it is recommended that you perform a data integrity check.

Notes

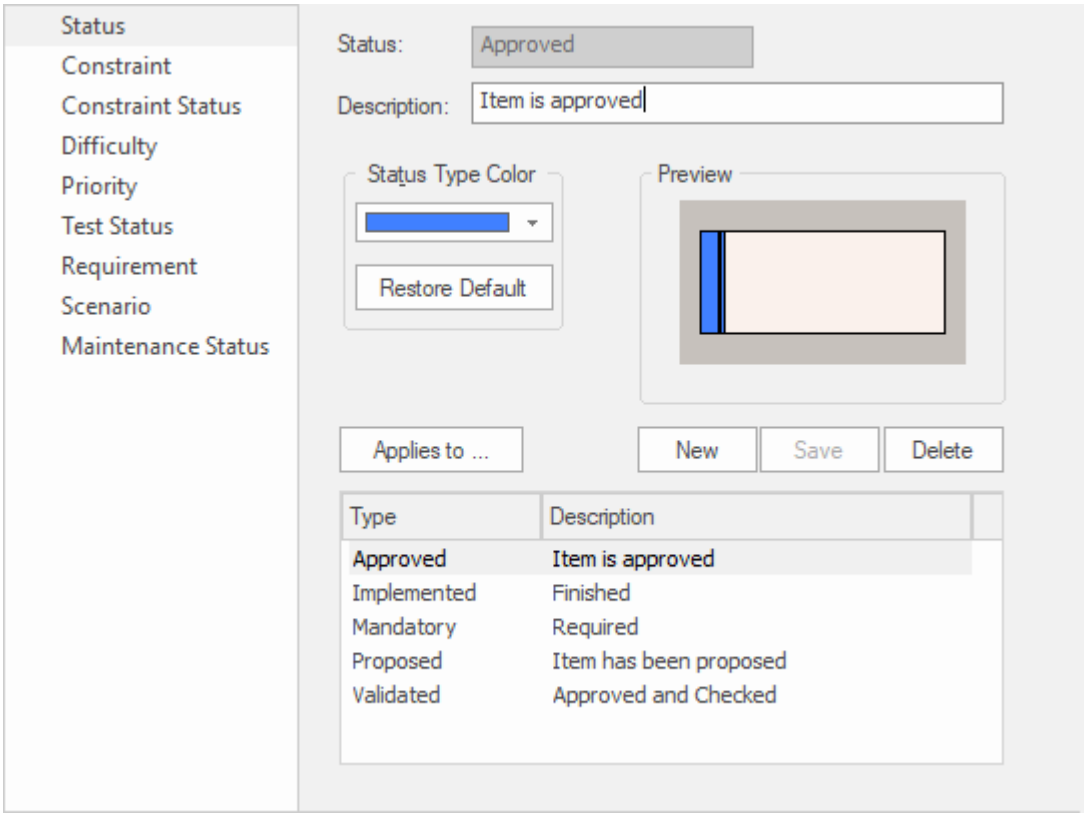
- Always repair projects on a local drive, never on a network drive
- The 'Compact a Project' option can also repair corrupt .eap files
- In the Corporate, Unified and Ultimate Editions of Enterprise Architect, if security is enabled you must have 'Administer Database' permission to repair a project

Reference Data



Reference Data Defines Meta Data for items in the Repository

Reference data is used universally in Enterprise Architect as meta-information for properties and base values for repository items including elements, features and connectors and tool features. It is often seen in drop-down lists and selection boxes, and includes such items as Status Codes, Requirement Types, Model Authors and more.



General Types window showing Status Types including

color settings for each type.

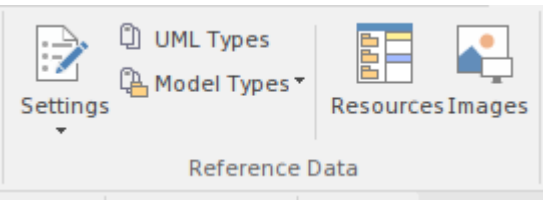
For convenience, Enterprise Architect comes pre-configured with useful reference data, but it is best practice to review this existing data when setting up a project and either replace or edit it to ensure it is appropriate for your organization's projects.

Reference Overview

Reference data is an important part of a repository and, apart from ensuring integrity and consistency, can drive the visualizations of reports and dashboard widgets. You can, for example, visualize the Requirements grouped by Status and Difficulty using Charts and graphs. You can define reference data by using the dialogs for each reference data type or by importing it from another repository. It is best practice to set up the metadata before populating the repository with your organization's modeling information. This ensures that when you add elements you will have the right information available, and removes the need to change the elements again.

There are different types of reference data, some of which apply at a repository level - such as Status Codes and Authors - and some that are more technical in nature - such as Code Engineering and Database Datatypes - that apply to particular element or tool types.

You can transfer reference data between repositories by exporting it from one repository and importing it into another; this mechanism can save time and ensure consistency between repositories.



Settings ribbon Reference Data panel.

- UML Types - Stereotypes, Tagged Value Types and Cardinality Values
- People - Authors
- General Types - Status Types, Constraints, Requirement Types
- Project Indicators - Risk, Metric, Effort
- Estimation Factors - Technical and Environmental
- Datatypes and Namespaces
- People
- General Types

UML Types

These are part of the UML Grammar and allow you to extend the core language to make it more suitable for your organization's projects. You use them to create new types (Stereotypes) that are relevant to the industry you work in or to add specific properties (Tagged Values) that help to specify the important aspects of elements of our domain. You can also specify a set of number pairs that specify how many (Cardinalities) of one item relates to another item.

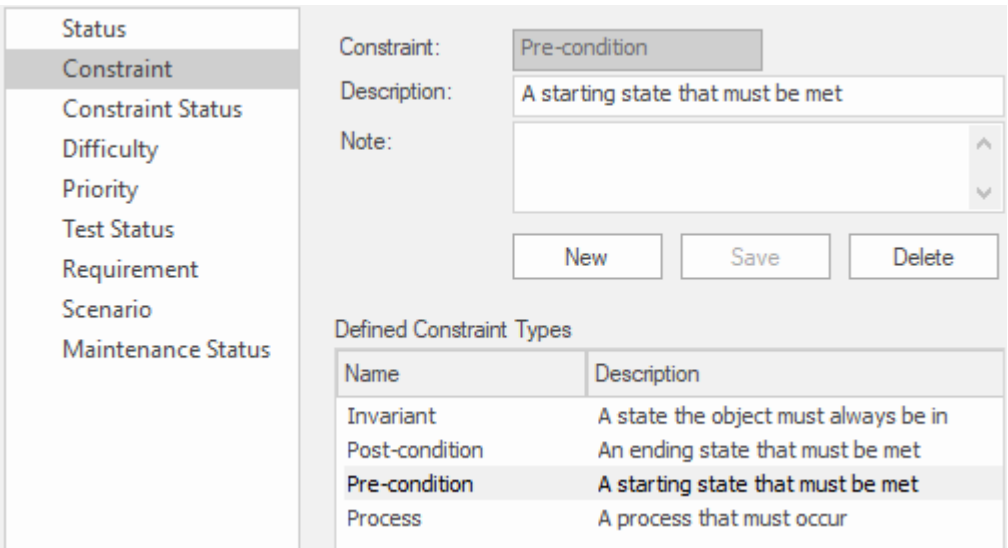
People

People are the stakeholders who participate in some way in the modeling of the enterprise and its projects. They include the people who create and maintain the models (Authors), the parts they play in the modeling effort (Roles), the people

who carry out project related tasks (Resources) and the people who ultimately requisitioned the models and system (Clients).

General Types

These are collections of meta data values that you use in drop-down lists when defining element properties. They add rigor to your models and are also used extensively in creating visualizations.



Constraints Type node of the General Types window showing the Pre Condition type.

Project Indicators

You can add a new type to the global set of risk types.

Maintenance

You can define Risk type and Test types which can be used when creating project based Risks and Test Respectively. There are a set of predefined types but you can add new types or modify or delete existing ones.

Metrics and Estimation

You can estimate the size of a software centric system by using Use Case points which utilize Environment and Technical Complexity factors and an hourly rate to calculate the effort required.

Auto Names and Counters

You can set a predefined name prefix including a counter for all element types which is used whenever a new element is created incrementing the counter each time.

Data Types and Namespaces

These define the set of data types used by both programming languages and database systems. These are preset for each of the languages and systems but you can define new types as required. You can also add completely new products that are not part of the predefined set and add types to these. Namespaces can also be defined.

Sharing Reference Data

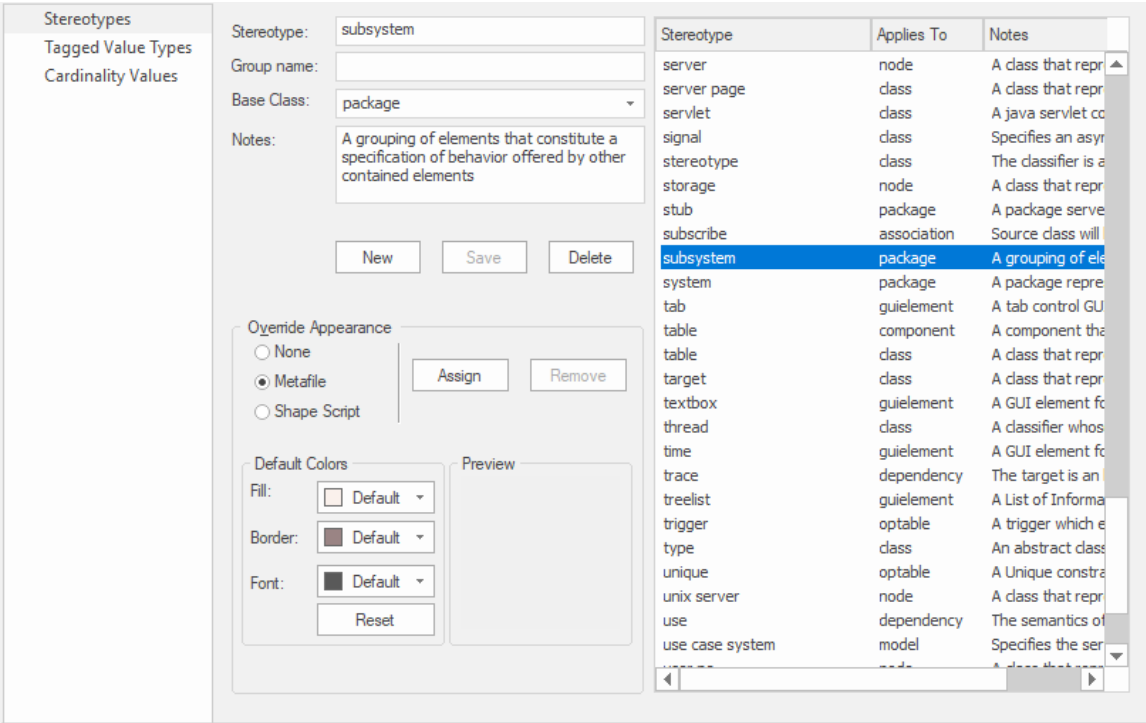
Enterprise Architect pre-populates reference data as default data, which you can then edit as required. Another productive way of getting data into a repository is to import it from an existing Enterprise Architect repository. This is a simple process of exporting the data from one repository and importing it into another.

UML Types

When the UML was devised its 'grammar' was designed to allow for extensions much in the same way that natural language grammars such as English and French are extended over time. Two of the most important of these extension mechanisms are Stereotypes and Tagged Values.

Using the 'UML Types' dialog you can configure stereotypes, Tagged Value Types and Cardinality (Multiplicity) Values for your project. In more detail, you can:

- Add, modify and delete stereotypes
- Customize stereotypes' appearance
- Update Shape Scripts and metafiles associated with stereotypes
- Create and modify Tagged Values
- Define cardinality rules



Stereotypes window showing the details for the Subsystem stereotype.

Access

Ribbon	Settings > 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	Settings > 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	Settings > 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 'Tags' tab of the Properties 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	Settings > 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 'Settings > 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	Settings > 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

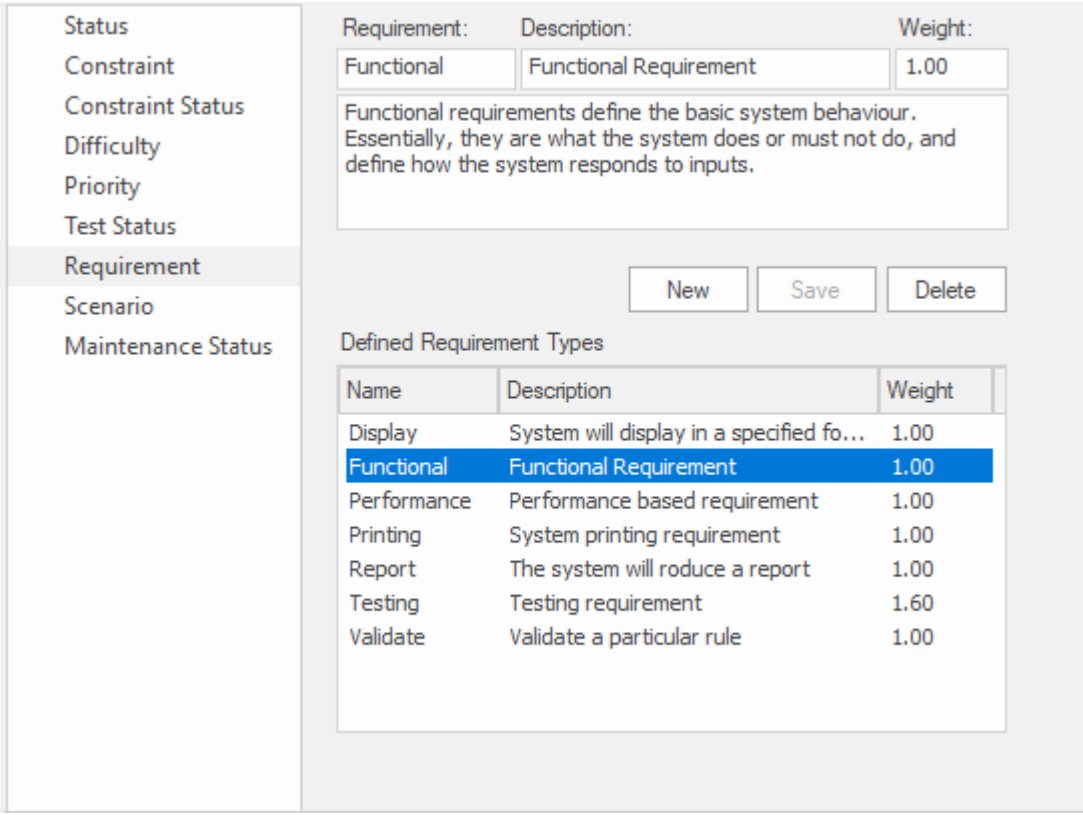
- You can transport these cardinality values between

models, using the 'Settings > 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
- Maintenance Status



Requirement page of the General Types window showing the details for the Functional Requirement Type.

Each of these can be separately managed through the 'General Types' dialog.

Access

Ribbon	Settings > Reference Data > Model Types > General Types

Status Types

The status of an element can define a point in a workflow or a milestone point in project management processes, such as business analysis or product development. 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 Settings > Reference Data > Model Types > General Types > Status

Customize Element Status Types

Option	Description
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 'Settings > 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	Settings > Reference Data > Model 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 'Settings > 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	Settings > Reference Data > Model 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.
	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 'Settings > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options

Difficulty Types

The elements that you add to a repository can be simple or complex, and as you analyze elements to understand the amount of work required to review, develop or implement them it is expedient to set the Difficulty property that allows you to record this dimension.

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	Settings > Reference Data > Model 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.
	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 'Settings > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options

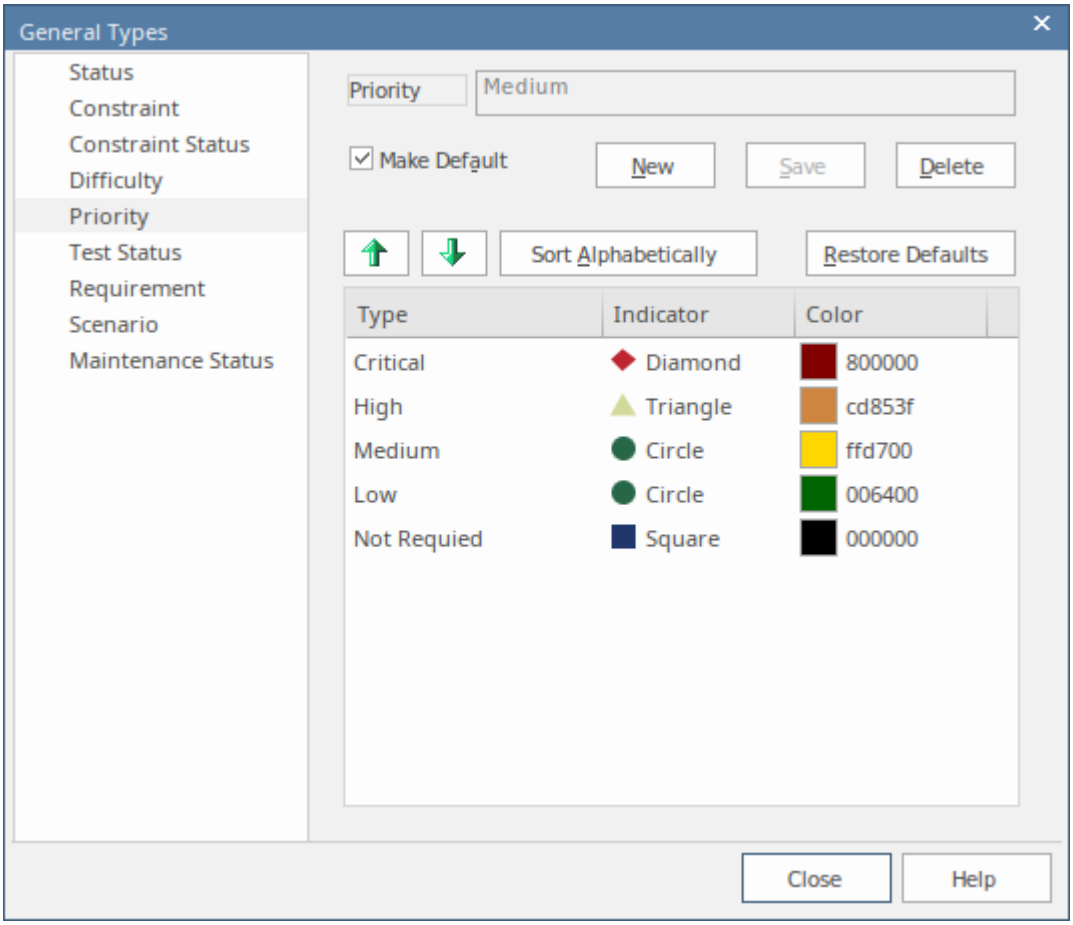
Priority Types

Not all elements are equal, and as you define the elements in a repository you can assign a Priority to indicate the relative importance of the element. You define the element's priority using the 'Priority' field of the elements's 'Properties' dialog. 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.

You can also set a shape and color for the display of each Priority Type that is displayed in the Inspector Status Tab. For more details see the *Status Tab of the Inspector Window* Help topic.



Access

Ribbon	Settings > Reference Data > Model 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.
	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 'Settings > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options

Test Status Types

All tests defined in the Test Cases 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	Settings > Reference Data > Model 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.
	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 'Settings > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options

Requirement Types

Requirements are critical to the success of a project, and methodologies typically specify a number of requirement types used to make sure the business, technical, architecture and transition requirements are well-defined. You define the type of requirement using 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 from your methodology or from an industry standard method..

Access

Ribbon	Settings > Reference Data > Model 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 'Settings > 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	Settings > Reference Data > Model 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 'Settings > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon

options

Maintenance Status Types

Model Maintenance items such as Changes, Issues, Defects and Tasks pass through a series of stages whereby each item is raised, assessed, approved, worked on, reviewed and completed. As an item passes through each stage, it can be assigned a Status from this system-provided list:

- Active
- Complete
- Deferred
- Merged
- New (the initial default status)
- No Action
- Paused
- Rejected
- Rework
- Verified

The Maintenance Status values are defined on the 'Maintenance Status' page of the 'General Types' dialog. You can add to or replace any of these values with your own custom values.

Access

Ribbon	Settings > Reference Data > Model Types

	> General Types > Maintenance Status

Customize Maintenance Status Types

Option	Description
New	Click on this button to clear the fields to define a new Maintenance status.
Maintenance	Type the name of the Maintenance status to define.
Make Default	To make the selected status the default for all Maintenance item 'Status' fields, select the checkbox.
Save	Save the Maintenance status item and add it to the 'Type' list.
Delete	Remove the currently selected Maintenance status from the 'Type' list.
Type	Lists the currently available Test status types.
	Move the selected type higher or lower in

	the 'Type' list. This defines the ordering used in the Maintenance 'Status' drop down combo boxes, and in sorting any report or list when the Maintenance Status Type is used as the sort-by property. For example: you have set up a number of Change items on a Package, and you run a report on the Change items; then you sort on the 'Status' column. The report entries are grouped by status, in the order you defined in the 'Type' list.
Sort Alphabetically	Reorganize the entries in the 'Type' list into alphabetical order.
Restore Defaults	Remove all customized Maintenance status values and restore the system defaults.

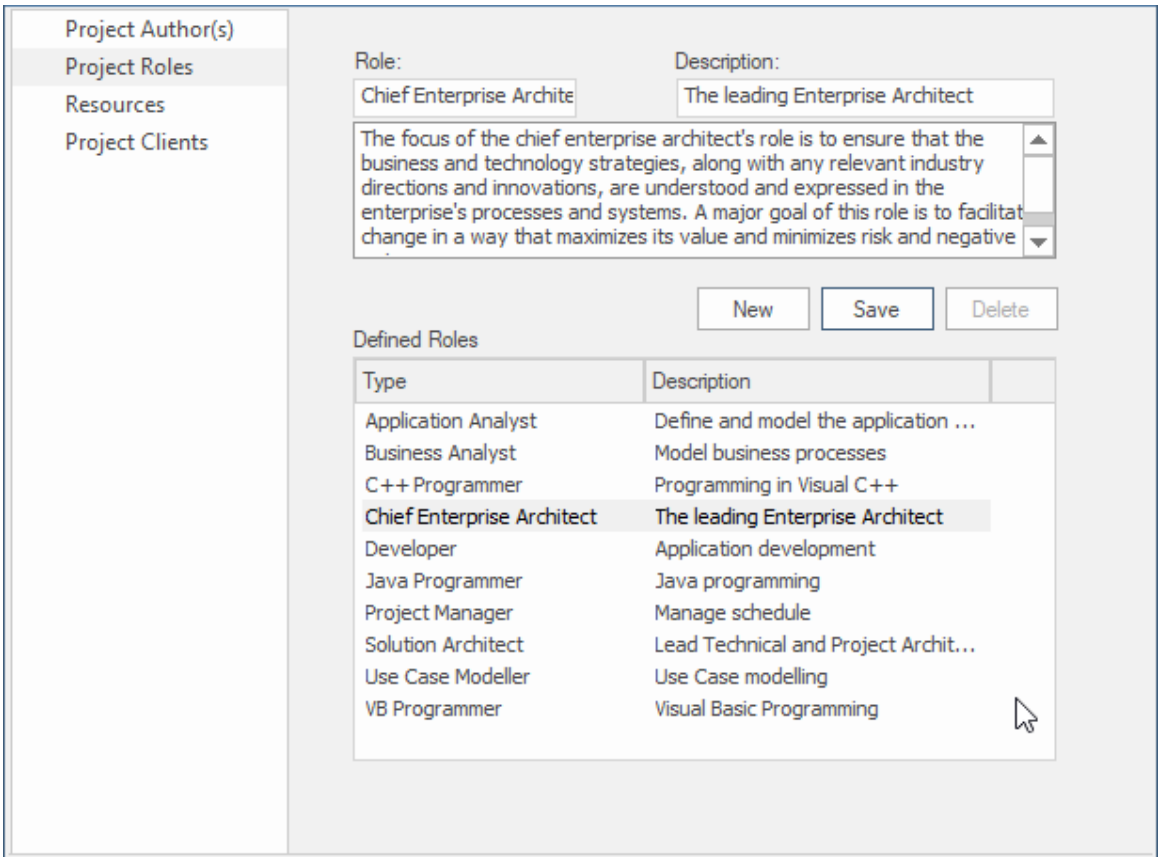
Notes

- You can transport customized Maintenance status types between models, using the 'Settings > 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



Project Roles page of the People window showing the details of the Chief Enterprise Architect role.

Access

Ribbon	Settings > Reference Data > Model Types > People

Project Authors

Project Authors are the individual people (e.g. John Brown) who work to create and maintain the models contained in the repository. When you create a model element your name will be assigned to the element and this name will be usedThey can be assigned as resources that work on model elements and

Access

Ribbon	Settings > Reference Data > Model 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 'Settings > 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.
	Type any additional notes concerning the

Notes	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	Settings > Reference Data > Model 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 one or more roles with respect to a model. The roles represent the disciplines associated with the model and system. They are typically defined as part of a system acquisition or development process. The roles could be names like: Chief Enterprise Architect, Business Analyst, Ux Designer, Product Owner. People can be assigned to a role.

Access

Ribbon	Settings > Reference Data > Model 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 'Settings > 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	Settings > Reference Data > Model 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 'Settings > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options

Project Clients

Project clients are the parties (individuals or organizations) that either requisitioned the system or part of the system being modeled or will be the users of the system. Defining these parties helps to keep the focus on the important reasons why the system is built modeled.

Access

Ribbon	Settings > Reference Data > Model 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 'Settings > Model > Transfer > Export

Reference Data' and 'Import Reference Data' ribbon options

Risk Types

Using Enterprise Architect you can add a risk type to the global list of risk types that can be assigned to any element in the model. The global list of risk types displays in the 'Type' field drop-down list on the 'Risks' page of the Project Indicators window.

Access

Open the Project Indicators window using one of the methods outlined here, then select the 'Risk' page.

Ribbon	Settings > Reference Data > Model Types > Project Indicators : Risk

Add a new risk type to the global list

Click on the New button to define a new Risk

Option	Action
Risk Type	Type the name of the risk type. (Or, to edit an existing risk type, click on the risk name in the 'Defined Risks' panel.)
Description	Type a short description of the risk type.

Weight	Type the default weighting to apply to the risk type.
<note>	Type any additional information on the risk type.
Save	Click on this button to save the changes that you have made to the dialog.
New	Click on this button to clear the data fields ready to define a new risk type.
Delete	Click on an entry in the 'Defined Risks' panel, and click on this button to immediately delete the risk type.
Close	Click on this button to close the 'Project Indicators' dialog. If you have not saved your changes, the system prompts you to save or abort those changes.

Notes

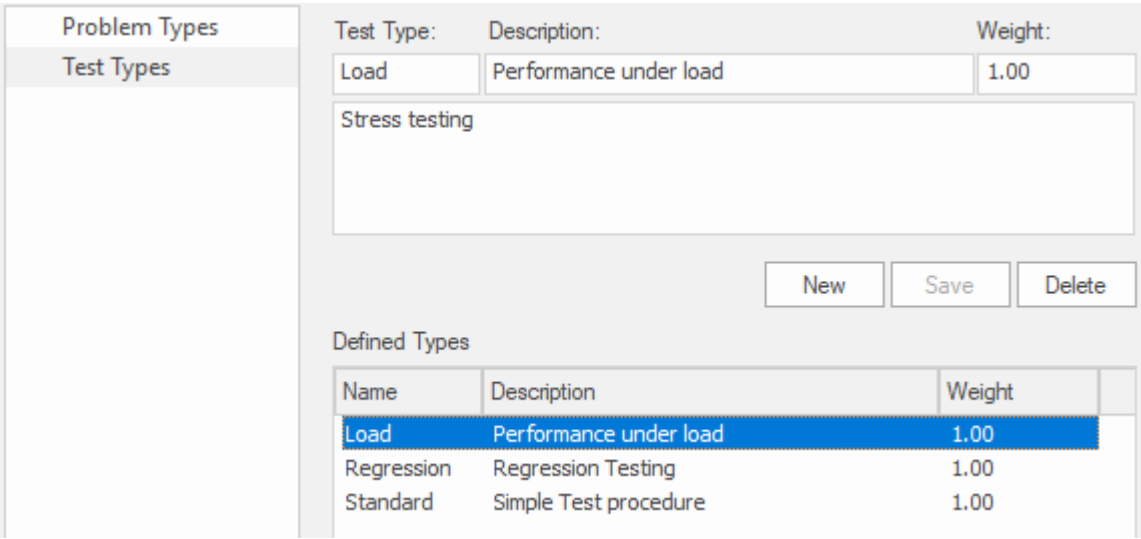
- Although Enterprise Architect does not currently provide detailed reports on risks within a model, you can use the Automation Interface or similar tools to create your own custom reports based on risk information you enter

- You can transport risk types between models, using the 'Settings > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options

Maintenance

Within your project, you can set up Maintenance types that determine the base Problem Types to be used within the model. These include:

- Risk Types
- Test Types



Access

Ribbon	Settings > Reference Data > Model 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	Settings > Reference Data > Model 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 'Settings > 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, regression tests and standard tests.

Access

Ribbon	Settings > Reference Data > Model Types > Maintenance > Test Types

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.

Notes

- You can transport these test types between models, using the 'Settings > 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

Metrics and Estimation

You can calculate the size of a planned software centric project by using a system of measurement called Use Case Points based on the fact that a set of Use Cases will specify the required functionality for a system. The method uses two important factors in the calculation:

- Environmental Complexity Factors (ECF)
- Technical Complexity Factors (TCF)

You specify a weighting for each factor to indicate how important the factor is in the overall measurement. You can also add any number of new factors or change or delete existing ones as required. These factors are specified and maintained in '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 Build & Deploy and Project Resources topics, or specifically:

- Technical Complexity Factors
- Environment Complexity Factors
- Default Hours
- Effort Types
- Metric Types
- Risk Types

Auto Names and Counters

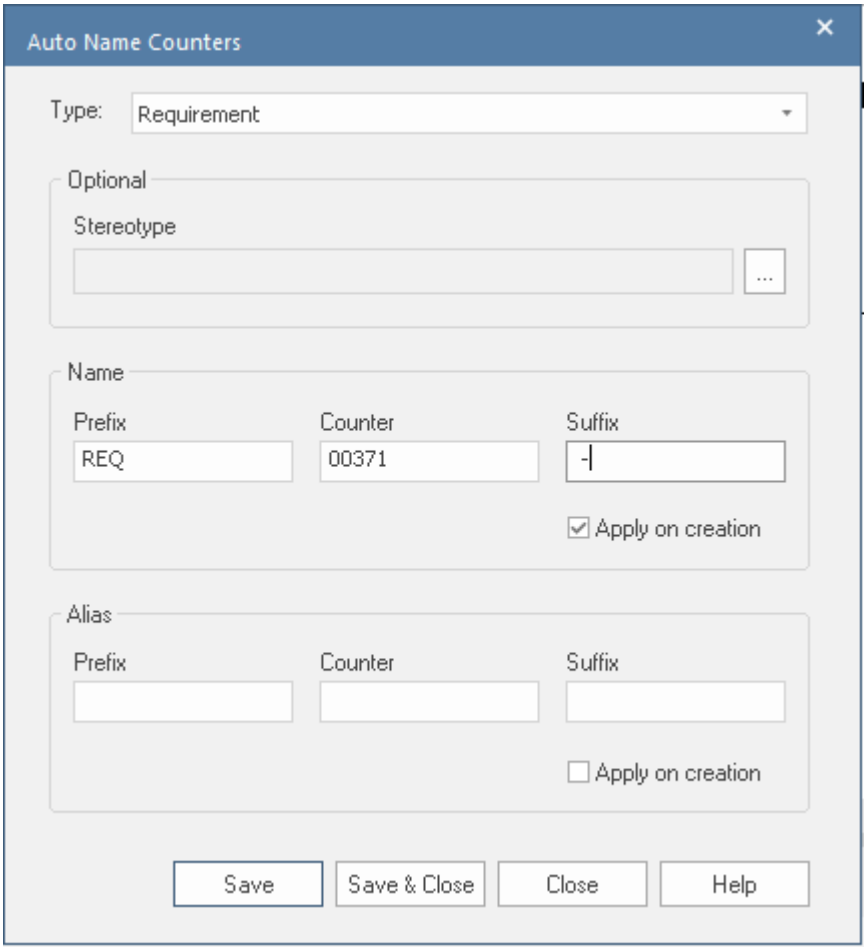
When you create a new element in your model, you have different options for naming the element depending on how you are creating it. If you create the element:

- On a diagram, the system applies a default naming convention of the element type plus the increment of the number of elements of that type in the same Package - for example, Actor4; you can change this to any other name by editing the 'Name' field on the element's 'Properties' dialog
- Within the Browser window, the system prompts you to type the element name into the 'New Element' dialog

You can also set up your own, automatic, naming and numbering convention for elements of the same type, so that:

- On a diagram, it is your naming/numbering convention that is automatically applied to the new element (which you can still change if you need to, as before)
- Within the Browser window, the 'New Element' dialog Auto button applies the naming convention to your new element

This is an example of the Auto Name Counter window showing an example name and counter for new Requirements.



If you have existing elements in a Package, and you want to apply the new naming convention to them, you can make this change very easily to all elements of the same type in the Package at once.

You define a naming convention for the element type as a model setting. At the same time, you can also optionally define a naming/numbering convention for the 'Alias' field and/or for a stereotype for elements of that type.

Access

Ribbon	Settings > Reference Data > Settings >

	Auto Names and Counters

Set up Auto Naming for an Element Type

Option	Action
Type	Click on the drop-down arrow and select the element type to which to apply auto naming/auto numbering.
Stereotype	If you want to apply the naming convention to a particular stereotype of the element type, click on the ... button to the right of the field, and locate and select the stereotype name using the Stereotype Selector dialog. Note that you can apply one naming convention to the base element type (such as a Class) and a different naming convention to a stereotype of that element (such as a SysML Block).
Name Prefix	Type a prefix for the new names (optional).
Name Counter	Type the counter value; use as many 0's as required to pad the number.

Name Suffix	Type a suffix for the new names (optional).
Name Apply on creation	Select the checkbox to apply auto naming to each new element of the specified type, on creation. If you do not select the checkbox (but click on Save), the autonaming details for the specified element type remain defined in the dialog, but are not applied to new elements of that type. The default element naming convention takes effect (<element type><n>).
Alias Prefix	Type a prefix for the new Aliases (optional).
Alias Counter	Type the counter value; use as many 0's as required to pad the number.
Alias Suffix	Type a suffix for the new Aliases (optional).
Alias Apply on creation	Select the checkbox to apply Alias auto naming to each new element of the specified type, on creation. If you do not select the checkbox (but click on the Save button), the Alias

	autonaming details for the specified element type remain defined in the dialog, but are not applied to new elements of that type. The default element naming convention takes effect (<element type><n>).
Save	(The default highlighted button.) Click on this button to preserve the values you have entered, and leave the dialog open for further chabnges. New elements of this type now have an automatically-generated name and/or Alias with an incrementing counter value. • If the user creates an element on a diagram, such as from the Diagram Toolbox or Quick Linker, the element is automatically assigned the defined name and/or Alias • If the user creates an element outside a diagram (such as in the Browser window) using the 'New Element' dialog, they can click on the Auto button next to the 'Name' and/or 'Alias' field on the appropriate dialog to apply the auto-generated text • If the user already has some text in the 'Name' or 'Alias' field, it is over-written by the auto-counter text

	If an Alias is active, to view the Alias in a diagram requires that the option 'Use Alias if Available' is selected in the 'Properties' dialog for the diagram.
Save & Close	Click on this button to preserve the values you have entered, and close the dialog. The button has the same effects on element names as the Save button.
Close	Click on this button to close the dialog and cancel the values you have entered, leaving element auto-naming unchanged.

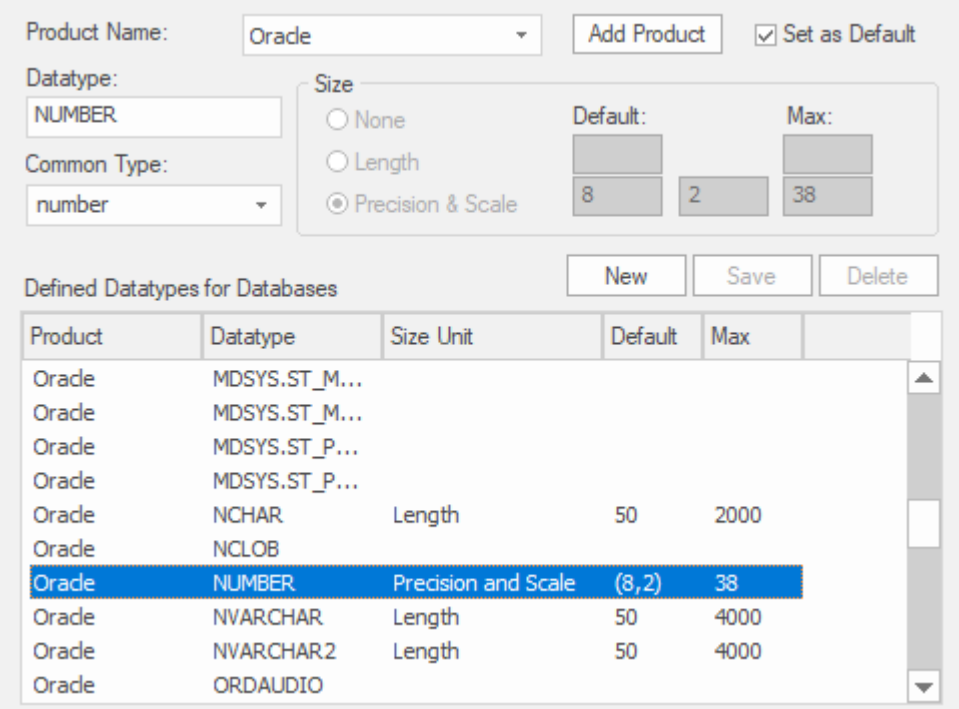
Notes

- Automatic numbering - whether system default or user-defined - does not increment across Package levels; if you have an element called Actor1 in Package A, and you create an Actor element in a child Package of Package A, that Actor will also be Actor1

Data Types and Namespaces

There is rich support in the tool for modeling database schemas and programming constructs. The tools supports a wide variety of database systems and programming languages each of which have a set of pre-defined data types that specify and constrain the information that the system will contain.

Enterprise Architect supports the definition and management of these data types and contains lists of each system's data types. When adding database objects such as Tables or Views a modeler can specify the type of RDBMS that is being modeled and the then when adding Columns the list of data types will be available for selection. A team will typically specify a default RDBMS system and this be the value used when a new database object is created.



The same applies to programming languages, when adding a Class a modeler specifies the programming language and then when working with attributes or operations the applicable language specific data types will be available.

Data Types

Every Table column that you define in your data model has a data type assigned that specifies the type of information that can be stored by the column. The available datatypes for a column are dependent on the selected DBMS for the Table, because each DBMS supports its own list of datatypes. Whilst each DBMS supports the same basic types, such as string, whole or decimal numbers, each DBMS calls them by different names and have different properties.

Each Enterprise Architect repository contains the definitions of the core datatypes for a number of standard DBMS products. However, since data types vary from one DBMS product to another, and from one version of a product to another, Enterprise Architect provides you with tools to:

- Define new data types for a new version of your DBMS product
- Define data types for a new, non-standard database product
- Automatically convert data types from one defined DBMS product to another
- Import and export datatypes between repositories

Coding 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	Settings > Reference Data > Settings > Coding Datatypes or Develop > Source Code > Options > Configure 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 'Settings > Model > Transfer > Export Reference Data' and 'Import Reference Data' ribbon options
- In the Corporate, Unified and Ultimate Editions of Enterprise Architect, if security is enabled you must have 'Configure Datatypes' permission to update and delete data types

Namespaces

Languages such as Java support Package structures or namespaces. In Enterprise Architect you can specify a Package as a namespace root, which denotes where the namespace structure for your Class model starts; all subordinate Packages below a namespace root will form the namespace hierarchy for contained Classes and Interfaces.

To define a Package as a namespace root, click on the Package in the Browser window and select the 'Develop > Source Code > Options > Set as Namespace Root' ribbon option. The Package icon in the Browser window changes to show a colored corner indicating this Package is a namespace root.



Generated Java source code, for example, will automatically add a Package declaration at the beginning of the generated file, indicating the location of the Class in the Package hierarchy below the namespace root.

To clear an existing namespace root, click on the namespace root Package in the Browser window and deselect the 'Develop > Source Code > Options > Set as Namespace Root' ribbon option

To view a list of namespaces, select the 'Settings > Reference Data > Settings > Namespace Roots' ribbon option; the 'Namespaces' dialog displays. If you double-click on a namespace in the list, the Package is highlighted in the Browser window; alternatively, right-click on the namespace and select the 'Locate Package

in Browser' option.

You can also clear the selected namespace root by selecting the 'Clear Namespace Attribute' option.

To omit a subordinate Package from a namespace definition, select the 'Develop > Source Code > Options > Suppress Namespace' ribbon option; to include the Package in the namespace again, deselect the ribbon option.

Notes

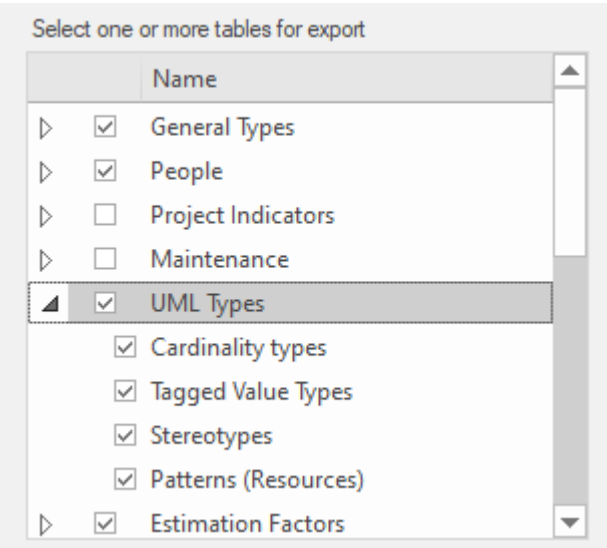
- When performing code generation, any Package name that contains whitespace characters is automatically treated as a namespace root

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.



Alternatively if your repository is MySQL, MS SQL Server or Oracle it is possible to configure your repository to use the Reference Data defined in another repository (on the

same server), see Linking Reference Data for details.

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	Settings > Model > Transfer > Import Reference Data > Import File Settings > 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 as you 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, Library 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 in Microsoft SQL Server Repositories:

- Importing images as reference data into a SQLServer repository is not supported when connected via Pro Cloud Server
- 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

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	Settings > 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

Linking Reference Data

If you have a number of projects that use the same reference data, you can use a central repository containing all common data and configure the other repositories to link directly to this data. This data includes system resources such as Security Users and Groups, Permissions, Scripts, Glossary terms or Data Types. Sharing common data between repositories makes it possible for teams working on multiple projects to leverage a common and consistent set of system resources, thereby promoting or enforcing common usage and standards.

Sharing common data can be applied to both existing and new projects, however each individual repository can be configured to use as much of the common data as is deemed necessary.

The ability to link to common data in a central repository uses the native features provided by the various Database Management System (DBMS), MySQL, SQL Server and Oracle. Meaning that it is configured at the database level. The benefit of this is that Enterprise Architect will function normally without needing any modifications in order to access the data, in fact it is not even aware that the common data is physically located in another repository.

Access

Ribbon	Settings > Model > Transfer > Shared Repository

Configure a Repository to use shared data

The 'Shared Repository' screen provides an Administrator the ability to generate a SQL script that can be run on the current repository's database and configure it to share common data of a centralized repository. The generated script is tailored based on user selection and the DBMS of the current repository.

The process to configure the current repository to share a centralized repository's data, is:

Step	Action
1	In 'Script File' field, enter the complete filename and path of the script, or click on the ... button and search for it.
2	In the 'External Repository' field enter the database name of the centralized repository. Enterprise Architect will only continue if the entered database name exists on the Database Server hosting the Current Repository.
3	Select which Table Group(s) will be shared, as

	explained in this table.
4	Click the Generate button to create the script and save it to the specified location.
5	Open the SQL console of your choice and execute the generated SQL script. Since the schema of the database is being updated an elevated level of permissions are required to execute these statements.

Select Table Groups

There are approximately 30 tables that can be potentially shared between repositories, the 'Shared Repository' screen organizes these tables into logical Groups. While some Table Groups only contain 1 table, others contain multiple.

You can select all Table Groups or as many as is appropriate for the current repository.

Option	Action
Security Users and Groups	This group of tables contains a number of tables that define the Model Users, the Security Groups and the Security Groups each Model User belongs to.
Security Permissions	This group of tables contains the security permissions that are assigned to each

	Model User and Security Group. While at first glance it might seem strange that the Security User and Group does not include Security Permissions, it was a deliberate decision to keep these two separate because it provides a greater level of flexibility.
People	This group of tables contains the definition of all people/resources in the repository.
Glossary	This group contains the Glossary table that defines the glossary terms for the repository.
Types	This group of tables contains the definition of reference types that are typically used to populate drop down lists within Enterprise Architect's screens. Types like Complexity, Constraints, Effort, Problem, Property, Risk, etc.
Complexity Factors	This group of tables contains the definition of complexity factors, like ECF and TCF.
Scripts	This group contains the Script table that defines all custom JSScript, JavaScript

	and VBScript.
Data Types	This group contains the Data Types table that defines both the data types for all code languages and DBMSs.

How it works

The scripts generated by this screen work by renaming each Enterprise Architect Table that has been selected and then creating a Synonym (or View) that references the centralized database's table of the same name.

Consider the example that the current repository/database called EA1 should be configured to share EAMAIN's T_GLOSSARY table.

The script will:

- renamed table EA1.T_GLOSSARY to EA1.T_GLOSSARY_BAK

- create a new synonym (or view) call EA1.T_GLOSSARY that references EAMAIN.T_GLOSSARY

Both Microsoft SQL Server and Oracle support database Synonyms, which are a mechanism that provides an alternative name for another database object, in the case of shared data the other database object is a table in another database. MySQL does not support Synonyms, however a

View can be used to achieve the same behavior.

Removing Shared data

As described in the 'How it works' section, since the scripts generated by this screen simply rename the original tables by appending "_BAK" to their names, it is a relative easy process to rename (or delete) the synonym (or view) and then rename the table back its original name.

Notes

- Enterprise Architect can only generate Shared Repository scripts for MySQL, SQL Server and Oracle based DBMS repositories
- In order to share common data all repositories must use the same DBMS and must be on the same network, preferable on the same database server (for performance reasons); that is, if the central repository is MySQL then all repositories that will use the common data must be MySQL and have direct network access
- The Shared Repository screen cannot be opened while the current connection is using a Cloud connection, because Cloud-based connections do not provide the necessary database information needed to generate a working SQL script; however, once the script has been generated (while connected with a direct ODBC connection) and then executed, Cloud-based connections will read shared data in the same way as their direct connection counter-parts

Shared Security Reference Data

When deciding which Table Groups of the centralized repository each individual repository should share, the Security related tables should be carefully considered, and the decisions based on a number of factors:

- How are users going to be validated - via an external Single Sign-On (SSO) mechanism (like OpenID or Windows NTLM) or via a simple User ID and password stored within the Enterprise Architect repository?
- Should all repositories share the same list of users?
- Should user passwords be synchronized across all repositories?
- Do all users have the same permissions across all repositories?

There are two Shared Table Groups related to Security: 1) 'Security Users and Groups' and 2) 'Security Permissions'. Whether a repository should be configured to share either or both of these Table Groups depends on the environment. Consider these scenarios:

Scenario 1. All repositories have the same Users, and all Users have the same permissions within the Repositories; the individual repository should share both Table Groups: 'Security Users and Groups' and 'Security Permissions' from the centralized repository. The user list and permissions will need to be managed within the centralized repository.

Scenario 2. All repositories have the same users, but the Users have different permissions across the Repositories;

the individual repository should share the Table Groups: 'Security Users and Groups' from the centralized repository. The user list will need to be managed within the centralized repository while the user permissions will need to be managed within the individual repositories.

With each of the above scenarios it is still a requirement that the passwords for Users are maintain within the individual repositories; the only way to avoid this is to implement SSO authentication or to use Pro Cloud Server's Global Authentication.

Authentication via SSO

When using an SSO authentication mechanism, Enterprise Architect is no longer responsible for determining if the entered user credentials are valid, instead it is OpenID or Windows NTLM that validates the user credentials and advises Enterprise Architect that the given user is valid. Enterprise Architect then uses the supplied UserID to determine what features are available within the current repository. This requires that the complete list of Users is defined within the repository, however if each individual repository is configured to share the Security related reference data the list of Users (and their permissions) can be defined and managed in the centralized repository.

Authentication via Global Authentication

It's possible to configure the Ports of the Pro Cloud Server

to use a 'Global' repository for authentication purposes. When this option is set the initial log in process uses the 'Global' repository to validate the supplied user credentials, once those credentials have been confirmed any further security related queries use the information saved in the current repository. The benefit of this configuration means that each repository has the ability for the same user to have different permissions in each repository. However the downside is that the security user definitions must be manually kept up-to-date in each repository.

Using Shared Repository data (as described in the topic Link Reference Data to a Shared Repository) it is possible to remove the duplicated data by choosing certain Table Groups to shared and the remaining to be stored locally, the exact configure will depend on the individual environment.

More Information

Project Build & Deploy

Project Resources

Edition Information

- In the Corporate, Unified and Ultimate Editions of Enterprise Architect, if security is enabled you must have 'Manage Reference Data - Update' permission to update and delete reference items