Sparx Systems

Enterprise Architect 12

Reviewer's Guide

Copyright Sparx Systems 2015

Table of Contents

Introduction

This document provides a comprehensive overview of the capabilities of Sparx Systems **Enterprise Architect 12**. Each section focuses on a particular aspect of Enterprise Architect, providing an introduction to the purpose and benefits of each capability.

What is Enterprise Architect?

Enterprise Architect is a visual platform for designing and constructing software systems, for business process modeling, and for more generalized modeling purposes.

Enterprise Architect is based on the latest UML® 2.5 specification. UML defines a visual language that is used to model a particular domain or system (either proposed or existing).

Enterprise Architect is a progressive tool that **covers all aspects of the development cycle**, providing full traceability from the initial design phase through to deployment, maintenance, testing and change control.

What differentiates Enterprise Architect from other UML tools?

• Comprehensive UML 2.5-based modeling

• Built-in Requirements Management

• Extensive Project Management support, including resources, tasks, project calendar and metrics

• Test Management built-in: Testpoint management, model-based test execution, test case specification and support for JUnit and NUnit

• Flexible documentation options: HTML, PDF and DOCX report writers

• Code engineering support for many languages *out of the box*

• An integrated Visual Execution Analyzer to profile, debug and document executing applications; instantiate run-time model objects; and record sequence diagrams from a stack trace

• Extensible modeling environment that can host user-defined profiles and technologies

• Usability: Enterprise Architect makes it easy to get up and running quickly with UML

• Speed: Enterprise Architect is a spectacularly fast performer

• Scalability: Enterprise Architect can handle extremely large models and many concurrent users with ease

• Price: Enterprise Architect is priced to outfit the entire team, making collaboration and team development cost effective

How popular is Enterprise Architect now?

With over 350,000 licenses sold, Enterprise Architect has proven remarkably popular across a wide range of industries and is used by thousands of companies world-wide. From large, well-known, multi-national organizations to smaller independent companies and consultants, Enterprise Architect has become the UML modeling tool of choice for developers, consultants and analysts in over 130 countries.

Sparx software is used in the development of many kinds of software systems in a wide range of industries, including: aerospace, automotive, banking and finance, defense, electrical engineering, medicine, research and academia, retail, transport and utilities. It is also used effectively for UML and business architecture training in many prominent colleges, training companies and universities around the world. Actual implementations range from single users to companies with over 1000 seats working on large, distributed projects.

What benefits does Enterprise Architect provide?

Model and Manage Complex Information.

Enterprise Architect helps individuals, groups and large organizations model and manage complex information. Often this relates to software development and IT systems design and deployment, but it can also relate to business analysis and business process modeling. Enterprise Architect integrates and connects a wide range of structural and behavioral information, helping to build a coherent and verifiable architectural model, either what-is or what-will-be. Tools to manage versions, track differences, audit changes and enforce security help control project development and enforce compliance with standards.

Model, Manage and   *Trace Requirements.*

Capture requirements and use full traceability from base requirements to design, build, deployment and beyond. Use impact analysis to trace from proposed changes to original requirements. Build the 'right' system.

Integrate Teams and Share a Vision.

A scalable, easily deployed, multi-user environment, Enterprise Architect integrates team members from all sections and all phases of a product's (or system's) development and maintenance life-cycle, providing significant benefits from built-in collaboration and inherent information sharing. A single repository for business analysts, software architects, developers, project managers, testers, roll-out and support staff. A 'unified' view of a complex system having many view points and many possible sub-systems. Shared models can be accessed easily and securely by remote team members with Enterprise Architect's Cloud Server.
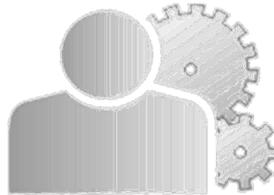
Design and Build Diverse Systems using UML.

UML 2.5, an open standard, provides a rich language for describing, documenting and designing software, business and IT systems in general. Enterprise Architect allows you to leverage the full expressive power of UML 2.5 to model, design and build diverse systems in an open and well understood manner. Generate code, database structures, documentation and metrics. Transform models. Specify behavior and structure as the basis for contractual agreements.

Visualize, Inspect and Understand Complex Software.

Software is complex and often hard to understand. Use Enterprise Architect to reverse engineer a wide variety of source code to understand static structure. To complete the picture, use the unique built-in profiling and debugging tools at run-time to capture and visualize executing software. Create run-time instances of model elements and invoke methods using the built in Object Workbench. Integrate existing data models by reverse engineering database schema for a wide range of systems.

Use Full Life-Cycle Modeling and Project Management.

Capture and track information about model elements that are important to success: for example, Testing, Project Management and Maintenance details. Use this information to drive and track product development and delivery.

Share and Re-Use Information Across Tools.

Enterprise Architect supports a number of mechanisms for exporting and importing models using industry standard XMI. This allows modelers to use information created in other tools, to copy information between Enterprise Architect models and even to write and use custom tools that take XMI directly as input. Enterprise Architect can also be used as an Open Services for Lifecycle Collaboration (OSLC) Provider, allowing other tools to identify and access modeled Requirements via a unique URL.
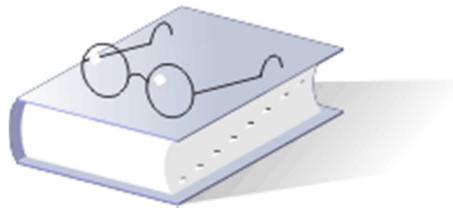
Create Platform Independent Models using Model Driven Architecture.

Model Driven Architecture (MDA) is an open standard designed to facilitate rapid application development in a platform independent manner. Models can be built at a high level of abstraction and using MDA based tools, transformed into models and code targeting a specific platform or domain. Enterprise Architect has a rich set of tools built-in to support MDA.
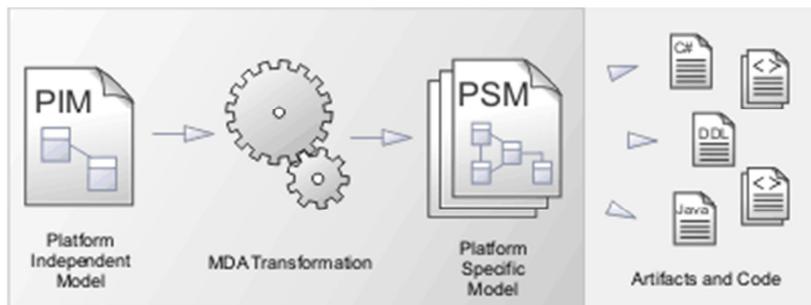
Summary of Enterprise Architect Features

Enterprise Architect enables you to:

- Model complex information, software and hardware systems using UML-compliant notation

- Specify, model, manage and trace requirements to deployed solutions

- Produce detailed and quality documentation in PDF, HTML, DOCX and RTF formats

- Leverage industry-standard enterprise architecture frameworks

- Generate and reverse engineer code in 10+ programming languages[*]

- Model databases, generate DDL scripts, and reverse engineer database schema via ODBC[*]

- Manage, track and control change using baseline model merge and auditing capabilities

- Centralize enterprise-wide documentation of processes and information systems

- Model dependencies between elements, system dynamics and state

- Model class hierarchies, deployment, components and implementation details

- Record project issues, tasks and system glossary

- Assign resources to model elements and track effort expended against required effort

- Share models via a cloud server or reusable asset service (RAS)

- Import models in XMI format from other tools

- Manage version control through XMI using SCC, CVS and Subversion configurations

- Use UML Profiles to create custom extensions for domain-specific modeling

- Save and load complete diagrams as UML patterns

- Analyze and trace relationships between elements using the tabular Relationship Matrix

- Script and automate common tasks using the Automation Interface and Model Scripts

- Connect to shared database repositories using MS SQL Server, MySQL, Oracle and more[*]

- Migrate changes across a distributed environment using Controlled XMI Packages

- Perform model-to-model transformations using Model Driven Architecture (MDA)[*]

- Create and share dynamic views of model elements and diagram sets using Model Views

- Create Mind Maps, Business Process Models, Data Flow Diagrams and Wireframes for mobile apps

- Generate BPEL scripts automatically from business process models in BPMN™ notation[*]

- Generate executable business logic from rule tasks traceable to natural language business rules[*]

- Visualize executing applications using the Visual Execution Analyzer

- Transform behavioral models into executable source code both for software and for hardware description languages (HDLs) such as Verilog, VHDL, and SystemC[*]

- Simulate models, including activity models, business process models and SysML parametric models[*]

- Model and generate XML schema (XSD) and WSDL; Debug XSLT; Edit and Validate XML, XSD

[*] Available features vary across editions. See http://www.sparxsystems.com/products/ea/editions.html

Modeling Based on Open Standards

As a contributing member of the Object Management Group, Sparx Systems understands the importance of open standards to communicate effectively to a wide range of stakeholders. To this end, Enterprise Architect helps you to:

- Visualize systems using the latest UML 2.5 notation

- Document and describe business processes with BPMN 2.0

- Model and simulate systems engineering projects with SysML 1.4

- Leverage numerous other open modeling standards – all in the one modeling environment!

UML, BPMN and SysML

Enterprise Architect supports all UML 2.5 models and diagrams. You can model business processes, web sites, user interfaces, networks, hardware configurations, messages and many other aspects of your development.

In brief, Enterprise Architect:

- Was the first UML tool to introduce comprehensive UML 2 support in April 2004

- Continues to refine and update UML 2 support

- Supports all 14 diagram types from UML 2.

In addition to UML, Enterprise Architect supports the latest Business Process Modeling Notation (BPMN) and Systems Modeling Language (SysML) specifications. Enterprise modeling notations are also supported out-of-the-box, including ArchiMate® 2.0, SoaML and SOMF™.

Enterprise Architect supports numerous other diagram types that extend core UML diagrams for strategic modeling, mind mapping, formal requirements specifications, data-flow diagrams, user interface prototyping and domain-specific modeling. The tool also provides alternative views that make editing the core UML diagrams more intuitive and effective. One example is the State Table editor, which renders a standard UML State Machine diagram as an editable logic table.

Business Process Modeling and Simulation
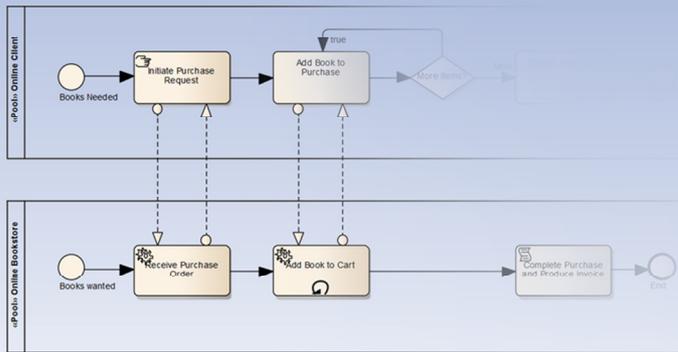
There are numerous approaches to business process modeling (BPM) using UML as the base modeling language. In particular, Activity diagrams, Object diagrams and custom Profiles provide a wealth of modeling possibilities for BPM analysts to use. Enterprise Architect complements the basic UML 2.5 language palette with custom analysis, requirements management and process management elements (such as change, feature and issue elements).

The BPMN profile

One popular notation among business analysts is the Business Process Modeling Notation (BPMN). This notation specifically targets the business modeling community and is mapped to UML through a BPMN Profile. Enterprise Architect provides built-in support of the latest BPMN 2.0 profile.

BPMN 2.0 models are simulated using Enterprise Architect's built-in simulation engine. This allows you to dynamically examine process flows, validate the model and collect timing and resource information on proposed or legacy processes with a view to business process improvement.

BPEL generation

Enterprise Architect generates Business Process Execution Language (BPEL) scripts from BPMN models. Generated BPEL scripts can then be executed using a third-party execution engine. Specific model validation rules help to ensure that models generate valid BPEL code.

Data Modeling, and Database Engineering
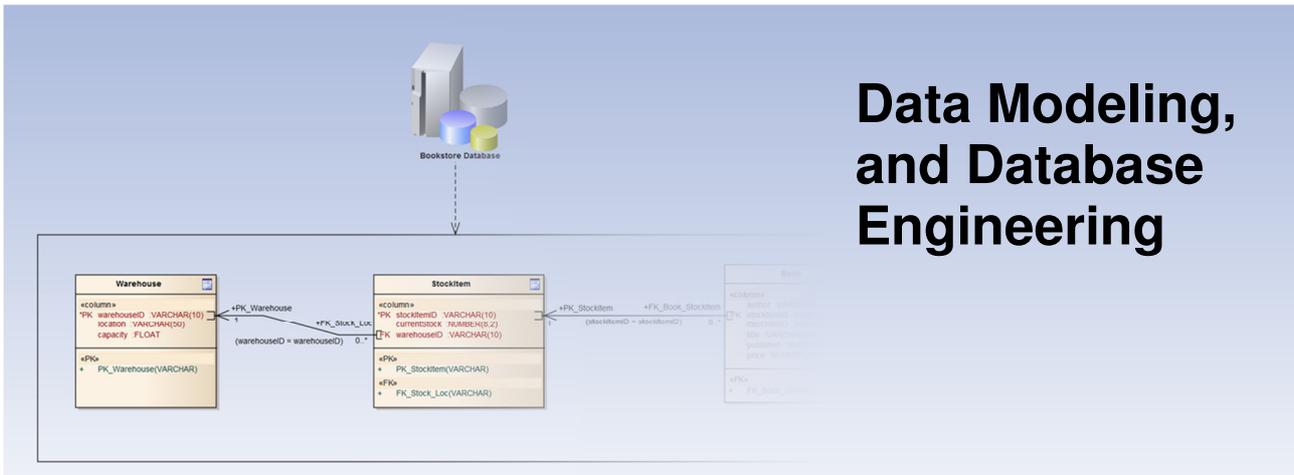
Enterprise Architect captures conceptual, logical and physical data models with one tool in the same model repository. You can import schemas from existing databases or automatically generate database scripts from your modeled schema, providing end-to-end traceability – from concept to implementation.

Data modeling

When designing databases Enterprise Architect's built-in **Data Modeling profiles** extend the UML to provide intuitive representations of database concepts. Supported notations include:

- IDEF1X
- Information Engineering
- UML DDL
- Entity-Relationship (ER) notation.

These extensions model database tables, columns keys, triggers, constraints and other aspects of relational database schemas.

Database Implementation

When implementing or maintaining databases, leverage Enterprise Architect's **Database Builder** to:

- Quickly visualize your database schema with DBMS-specific model patterns and diagrams
- Create tables, columns, primary keys, foreign keys, views and stored procedures
- Define indexes, sequences, functions and triggers
- Generate DDL and Alter DDL statements for all or part of your schema
- Selectively update your live database from Generated DDL statements
- Import an entire database schema via ODBC, or selectively update your physical model
- Connect to and query your databases from within the model environment.
- Maintain updates across Dev, Testing and Production environments

Database engineering support for major DBMS products

Enterprise Architect supports database engineering for numerous DBMS products including:

- DB2
- InterBase
- Informix
- Ingres
- Microsoft Access 2007
- MySQL

- Oracle 9i/10g/11g
- PostgreSQL
- Microsoft SQL Server 7/2000/2005/2008
- Sybase Adaptive Server Anywhere, Enterprise
- Firebird.

Customizable DDL generation templates are provided for each DBMS

Requirements Management

Gathering requirements is typically the first step in developing a solution, be it a software application or a business process. Requirements are essentially 'what the system needs to do'. Enterprise Architect's built-in requirements management features help you to:

• Define an organized and hierarchical formal requirements model

• Clearly differentiate requirement types, such as functional and non-functional requirements

• Link and trace system requirements to analysis and design elements and implementation artifacts

• Search on requirements and perform impact analysis with respect to requirement changes

• Generate custom reports or a complete requirements specification directly from the model

• Create custom attributes or properties appropriate to your organization or project

• Relate formal requirement elements directly to use case scenario steps, connecting early analysis artifacts to subsequent system development.

Enterprise Architect is distinguished among UML tools by its built-in requirements management capabilities. Being able to create requirements directly in the model resolves a number of traditional development issues such as traceability, interdisciplinary team divisions, integration with change and configuration management systems.

You can readily connect your requirements model with downstream processes, external artifacts, such as files or other resources, and technical documentation including architecture and implementation models.

Business analysts will be especially productive with Enterprise Architect's Specification Manager – a document-based interface for rapid entry of requirements into the model repository. A key feature of the Specification Manager is the speed and ease with which you can create, filter and review a large number of elements from one point, without necessarily developing or examining complex detail on each element.

Spatial Information Modeling with GML and ArcGIS™

Enterprise Architect supports model-based engineering of spatial data and geodatabase designs to aid development of geographic information systems (GIS). Interoperable spatial data is modeled in Enterprise Architect using Open Geospatial Consortium's Geography Markup Language (GML).

You can also design commercial geospatial databases specifically for the ArcGIS platform developed by Esri Inc. Enterprise Architect provides a built-in UML profile for ArcGIS, which means your spatial database design is directly traceable to your corporate information model and leverages industry standard modeling notation. You can export geodatabase models to ArcCatalog as an XML Workspace document that contains feature class definitions, feature datasets, spatial references, domains and more.

Existing spatial databases can be documented and visualized with ease – simply pass the XML Workspace document to Enterprise Architect and reverse engineer the ArcGIS geodatabase schema!

Enterprise Architect supports the latest ArcGIS 10.2 platform and the GML 3.2.1 specification[i].

[i]The UML Profile for GML was released as a beta implementation with Enterprise Architect 10.0

ArcGIS and ArcCatalog are trademarks, registered trademarks, or service marks of Esri in the United States, the European Community, or certain other jurisdictions.
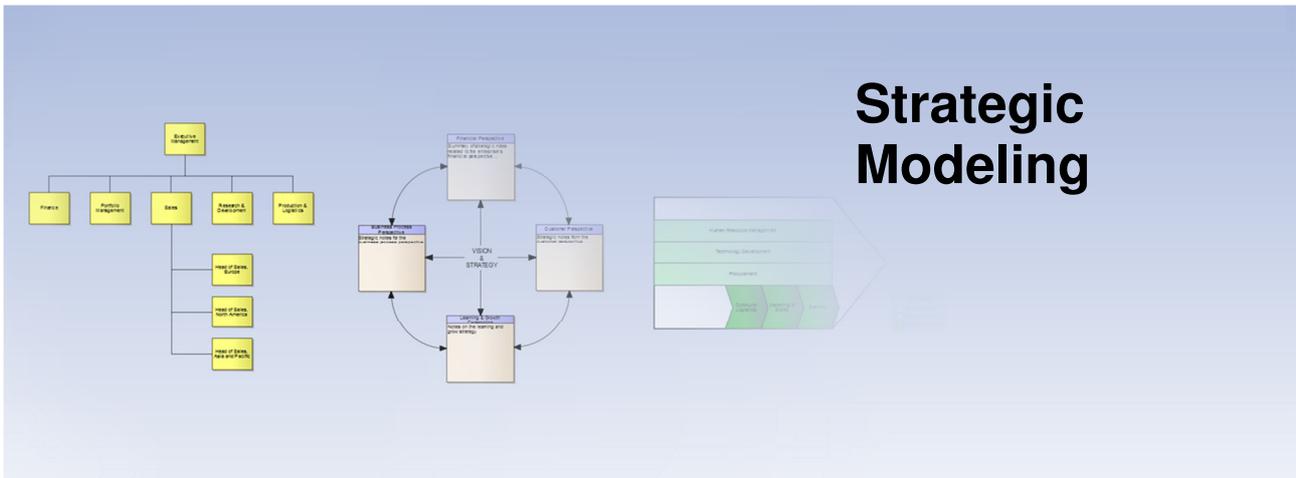
Strategic Modeling

Strategic Modeling enables an organization to plan for the future and make decisions in accordance with its mission and values. Enterprise Architect can model every stage of the planning and development process, taking an idea 'from the clouds to reality.'

You can document initial concepts using **mind mapping** diagrams and capture important strategic goals, business objectives and structure using:

- Strategy Maps
- Value Chains
- Decision Trees
- Balanced Scorecards
- Flow Charts
- Org. Charts.

Each of these high-level models can be traced directly to the analysis and design of systems and business processes that implement corporate strategy – helping to verify that resources are committed to initiatives that reflect business priorities.

Systems Engineering
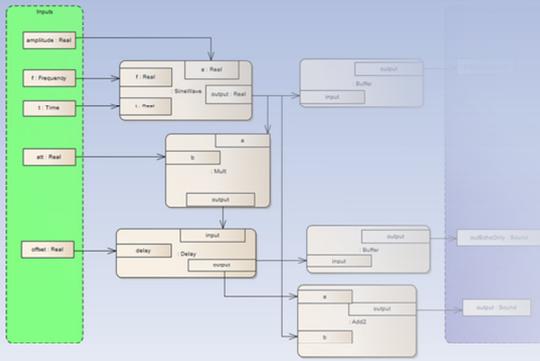
Integrating many high-end features for systems engineers, the Ultimate and Systems Engineering editions of Enterprise Architect provide SysML modeling, parametric model simulation, executable code generation, and model-to-code transformations for Hardware Description Languages (HDLs) and ADA 2005.

SysML and model simulation

Enterprise Architect enables you to develop SysML models quickly and simply, through a tightly integrated profile. The SysML profile for Enterprise Architect supports each of the nine SysML 1.3 diagram types and model guidance using the Quick Linker. The Systems Engineering and Ultimate editions of Enterprise Architect also provide simulation of SysML Parametric diagrams. This supports engineering analysis of critical system parameters, including the evaluation of key metrics such as performance, reliability and other physical characteristics.

Advanced behavioral modeling and executable code generation

Enterprise Architect can associate detailed behaviors with UML Activity and Interaction elements, allowing for executable code to be generated from Interaction (Sequence) and Activity models. This allows for more than just code stubs to be derived from models, by generating any nested behavioral models that define executable statements such as conditional logic, looping constructs and assignments.

Hardware description languages

The Systems Engineering and Ultimate editions of Enterprise Architect support code generation from State Machine models into Hardware Description Languages (HDLs) such as VHDL, Verilog and SystemC.

Test Management
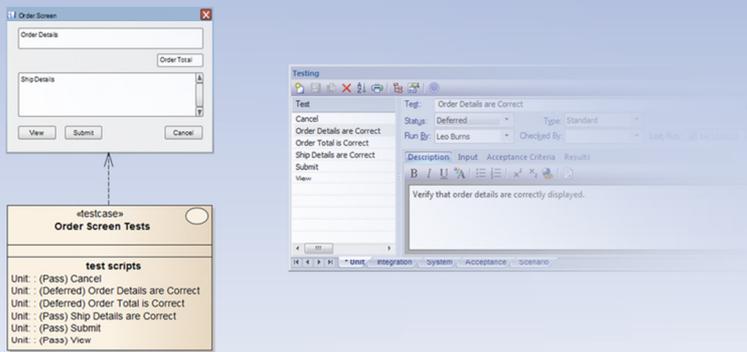
Enterprise Architect helps you to define test specifications, execute tests and record results directly from model elements. Keeping model elements and testing documentation in one integrated model can significantly improve the communication between quality assurance teams, software developers, analysts and architects.

Enterprise Architect provides the tools you need for numerous test-related activities including:

⑩ **Testpoint management**, which provides powerful model-based test execution. You can define Testpoints on your system design elements, execute these in the modeling environment and record the results in real-time as your application runs - without the need to manually construct test harness code. Testpoints can also be aggregated into reusable Test Suites, saving you valuable time.

⑩ **Test specification**: You can attach detailed test specifications to any element in Enterprise Architect, along with a record of test results, when tests were last run, and by whom. Unit, System, Integration, Acceptance and Scenario tests can all be captured and reported on, providing tight traceability between architecture and test information. Enterprise Architect can even automatically generate test cases from your structured Use Case scenarios!

⑩ **JUnit** and **NUnit** support: Enterprise Architect provides model transformations that automatically create testing elements from your system design elements. Code stubs can then be automatically generated by Enterprise Architect, allowing you to focus on defining the testing logic. You can compile, execute and record test results using your unit testing program, all without leaving Enterprise Architect.

Enterprise Architect is not only a modeling environment, it is also a complete Test Management environment.

Unit testing integrated into the modeling-development cycle

One of the key principles of unit testing is that you should always write your tests first. Enterprise Architect helps you to do this. When you add a new method to a class, run the test case transformation on the class. Enterprise Architect creates a corresponding test method that you can generate and fill in the details. This can easily be done before the code you are testing has been created.

UI Design and Wireframes for Mobile Apps and Web Pages

A key process in modeling a new system is designing the screens that users interact with. Enterprise Architect supports user interface design or wireframes for a wide range of devices and platforms including Android, Apple and Windows operating systems, and web pages.

Wireframes can be created with varying levels of detail – from a simple, abstract dialog, through to precise designs that visualize all the controls to be used on implemented device interfaces or web pages. You can leverage this capability to model apps for tablets, phones and other devices with rich graphical user interfaces.

Wireframes are fully traceable to other modeling elements, such as requirements and use cases, for end-to-end model design. This provides customers, managers and developers with a complete picture of how the user will interact with the system.

Wireframing Features:

- Comprehensive wireframing library for modeling common devices and user interfaces
- Diagram types and toolboxes for Android, Apple and Windows and web dialogs
- Pre-defined wireframe patterns and models supplied
- Supports customization of control data displays
- Full traceability of UI controls to other model elements
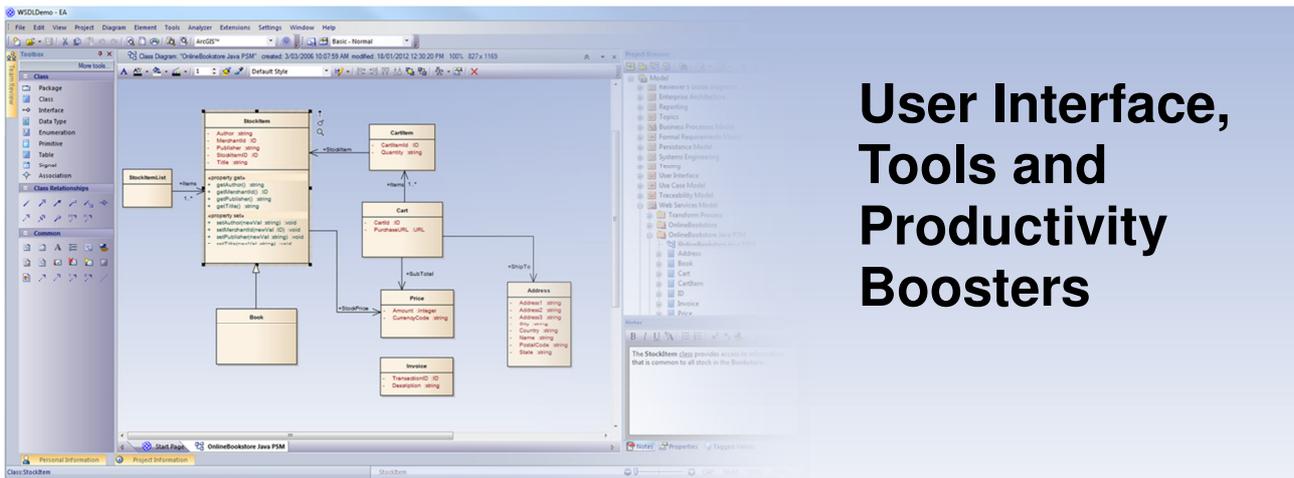
User Interface, Tools and Productivity Boosters

Enterprise Architect's user interface consists of a range of high-impact windows, menus and toolbars that you can arrange to suit your work methods. Key amongst these windows are:

- **The Project Browser**, which displays the model hierarchy of your project and facilitates adding, selecting, reorganizing or deleting packages, diagrams and elements

- **The Diagram Toolbox**, which is context-sensitive to the diagram being edited, and provides an efficient means of creating appropriate model elements or connectors

- **The Diagram View**, which brings the model to life visually:

    - Stylize and color connectors and elements for enhanced presentation and identification

    - Display diagrams in 'Hand-drawn' and 'Whiteboard' styles to denote draft status and encourage feedback

    - Edit **multiple views concurrently**, easily transferring elements between several open diagrams

    - View elements in tabular or list format, browse search results, trace seamlessly between visual representations and underlying source code, and more...

- **Context menus**, which provide options specific to the type of object and its environment.

Floating diagram and docking views

It is often useful to see multiple views simultaneously for comparisons, editing of related diagrams or referring to downstream representations of a given model. Enterprise Architect allows you to open several diagrams at once and dock these anywhere on screen. You can do the same for other views, such as the Model Search, Linked Document Editor, Source Code Editor, State Table or Element List views. This means, for example, you can edit a diagram with all its related views immediately visible. Easily move or copy elements between open diagrams and can take advantage of large or multiple monitors. Maintain context and continuity between editing sessions simply by saving your window layouts and open views to a Working Set in Enterprise Architect.

In-line editing of diagram elements

Enterprise Architect enables fast editing of element properties directly from the diagram. Using in-place edit commands, you can rapidly add and update element features such as attributes, operations and parameters, all without leaving the Diagram View. Rapid editing of diagram elements is further enhanced by numerous keyboard shortcuts for creating and connecting elements.

Quick linking technology

The Quick Linker provides a fast and 'in-place' mechanism for creating new elements and connectors on a diagram. Its context-sensitive selection menus guide the creation of 'correct' models, saving users time and improving overall productivity.

Other diagramming features

- Export diagrams to a range of image formats (.bmp, .jpg, .png, .gif, .emf and .wmf)

- Swimlanes enable logical partitioning of diagrams

- Pan and Zoom window provides a preview and easy navigation of complex diagrams

- Diagram Filters dynamically compare 'as-is' and 'to-be' architectures or highlight changes

- Compare and merge diagram baselines visually and roll-back changes between revisions

- Locks on diagrams prevent accidental modification

- Shape Scripts customize rendering of elements and support alternative modeling notations

- Alternative Images can be superimposed on elements, to replace the standard image

- Kanban diagrams help you to automatically arrange elements into lanes according to a particular status value or other filter.

Model and application patterns

Enterprise Architect ships with a number of predefined model and application patterns to assist in the creation of new projects and models. Each pattern contains useful notes, references and starter elements that together provide a framework on which you can build your model. For example, Enterprise Architect includes Java and .Net application patterns that get you started with a basic implementation model, generated code and appropriate scripts to build, run and debug your application.

Model validation

Model validation checks UML and SysML models against known rules from the specification, as well as any constraints defined within the model using the Object Constraint Language (OCL). You can perform model validation against a single UML element, a diagram or an entire package. The Model validation capability is therefore a powerful tool during your model review review process.

Document Generation and Reporting

The production of documentation is essential to realizing the full benefit of Enterprise Architect. Enterprise Architect generates high quality documentation in DOCX, PDF, HTML and RTF formats. You can easily report on an entire project, selected parts of the model, or even group packages in a manner different from the project view using Virtual Documents or Model Searches.

Document-based reports (DOCX, PDF and RTF)

Compatible with Microsoft® Word® or OpenOffice, Enterprise Architect's document generator can create entire specification directly from the model. Documents are generated from customizable templates, helping you to create reports and project deliverables that suit your corporate standards. You can selectively include or exclude specific information items and apply custom stylesheets to tailor reports for diverse stakeholders. Generate reports in Docx, RTF or PDF formats.

The HTML report writer

Enterprise Architect can export an entire model or a single branch of the model to HTML pages for convenient browsing via the web or corporate intranet. The HTML report provides an easy-to-use and highly detailed model tree. In addition, hyperlinked elements make browsing to related information very easy. The HTML documentation is based on user-customizable HTML templates, so you can tailor the generated web-pages to suit your company's standards.

Built-in document editor, linked documents and document artifacts

Enterprise Architect features a built-in document editor, with which you can attach rich-text documents to any element in the model. These **Linked Documents** are created from customizable templates and are included in generated web and document-based reports. Linked Documents thus provide a way to associate rich, free-form information with model elements. You can also store externally created office documents and other files directly in the model as **Document Artifacts**.

Structured use case scenarios

Use case scenarios capture vital analysis information using natural language. Enterprise Architect's Structured Scenario editor lets you use this information to drive downstream development and maximize traceability across the development life-cycle. You can link scenario steps to domain elements, business rules and glossary terms. From structured scenarios, you can generate test cases, Activity diagrams and other UML behavior diagrams. You can even reverse engineer existing process diagrams into structured, textual specifications to produce documentation deliverables.

Microsoft Word™ is a trademark of Microsoft Corporation.
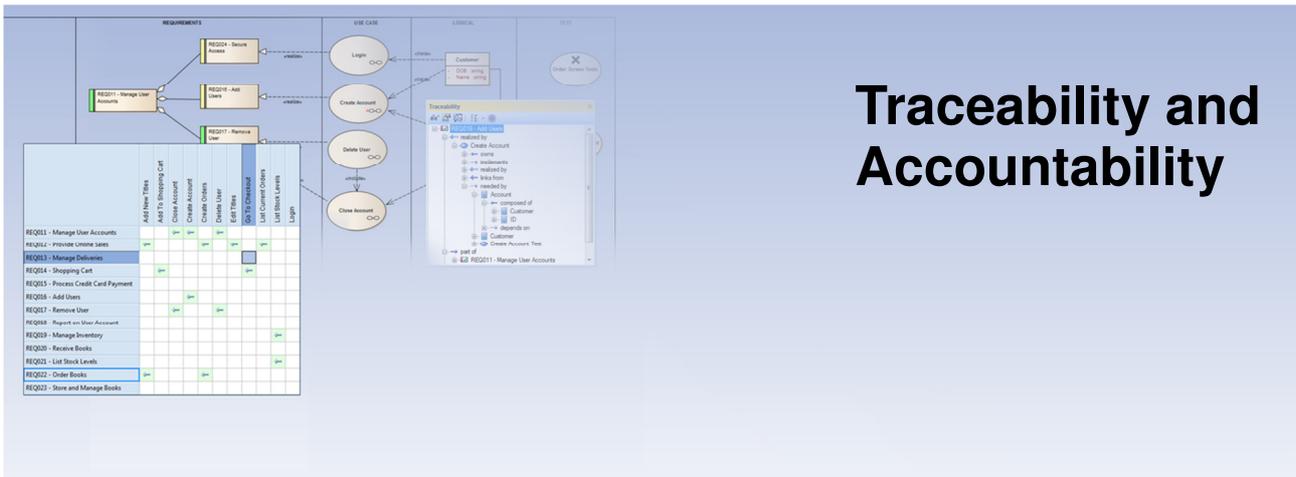
Traceability and Accountability

Model auditing

Enterprise Architect's Auditing feature tracks and record changes made to the model over time. Model administrators can use Auditing to monitor information about changes, such as:

- Who changed an element

- How many elements they changed

- When they changed the data

- What the previous values were, and

- What type of elements they changed.

The Audit View can be tailored to display specific types of change, and to filter changes by time period or user.

Gap and dependency analysis

Gap Analysis Matrix

Enterprise Architect's built-in Gap Analysis Matrix helps you to model gaps between your as-is and to-be architectures. The matrix provides a convenient interface for defining and monitoring identified gaps elements, which can be traced to other elements in the enterprise model.

Relationship Matrix

The Relationship Matrix helps you to study the relationships between model elements in a tabular view. Use it to easily identify traceability gaps and to conveniently create, modify or delete relationships. Document which relationships form CRUD operations using textual overlays, or customize the matrix overlays to suit your particular modeling domain.

Traceability Window

The Traceability Window provides a dynamic, navigable view of the current element's relationships to other elements in the model. Relationships shown include Aggregation, Inheritance and Dependency; embedded elements are also shown. By highlighting interconnections between levels of abstraction in the model, the Traceability Window provides a powerful impact analysis tool – helping you to see the effect of changing requirements on downstream elements.

Charts and Dashboards

Enterprise Architect can provide you with a wealth of data that is crucial to business planning, organizational strategy, decision making and project management. One way of summarizing this data in a format suitable for swift and easy appraisal is to present it in the form of charts and graphs, which are ideal for including in reports and distributing via the internet.

Within Enterprise Architect you can create Chart elements that define the type, source, content and appearance of a chart, either on their own Dashboard diagram or on other types of diagram as best suits your requirements. This provides a simple and fast mechanism for collating and presenting a lot of information automatically, such as summaries of Requirement Status or Test Case Status values across the current project.

Using Enterprise Architect, you can create a variety of charts including:

- Pie - 2D and 3D
- Doughnut - 2D and 3D
- Torus
- Line Graph (Time Series)
- Table (Model Views)
- Horizontal bar - 2D and 3D
- Vertical column - 2D and 3D

You can also generate some of these charts filtered according to another data quantity, presenting this as table columns, segments of a bar or separate bars in a cluster.

Finding Elements and Searching Meta-data

Model Search

The powerful Model Search facility retrieves each element in the model that meets the versatile criteria you define. The elements listed in the search results are selectable for printing, reporting, editing, adding to documentation and inserting into Team Review topics.

Model Views

Enterprise Architect's Model Views window provides a dynamic, filtered view of elements from the underlying model hierarchy. You can organize elements according to search criteria, favorite elements and diagrams, or technology-specific information, such as elements belonging to a particular framework view point. Views can be stored locally for use by individuals or included in a shared repository to achieve collaborative views. Automatic notifications can be set for a given Model View to alert you when an element created by another author is added to the view.

Element Browser

The Element Browser provides a context-sensitive view of all meta-data related to the selected element. This allows you to trace information on testing, project management, structured scenarios, maintenance and model semantics. The Element Browser thus provides a central hub for dynamic model reviews and a launchpad for related actions.

Package Browser

The Package Browser is a tabular, editable view of elements – use it to streamline the process of creating and updating elements in a given package. This can be particularly useful for analysts to edit formal requirement definitions within the model. You can also print the list or generate an RTF document directly from the entries in the Package Browser.

Track element usage

Enterprise Architect makes it easy to track and display the use of an element. The 'Find' and 'Usage' features for Enterprise Architect's diagrams, Project Browser and Package Browser show all occurrences of a given element throughout the model, and enables you to easily navigate to any occurrence.

Create diagrams with related elements

Enterprise Architect can automatically populate a diagram with all elements that relate to a given element. You can filter the inserted elements based on the type, direction and depth of the relationship. The 'Insert Related Element' feature provides a fast and powerful way to build up specific relationship overviews for your frameworks or reverse engineered source code.

Other reports

Enterprise Architect supports a variety of useful out-of-the-box reports including: Resource and Task Details, Project Issues, Project Glossary, Project (size) Statistics, Dependency and Implementation Details and Testing Details.
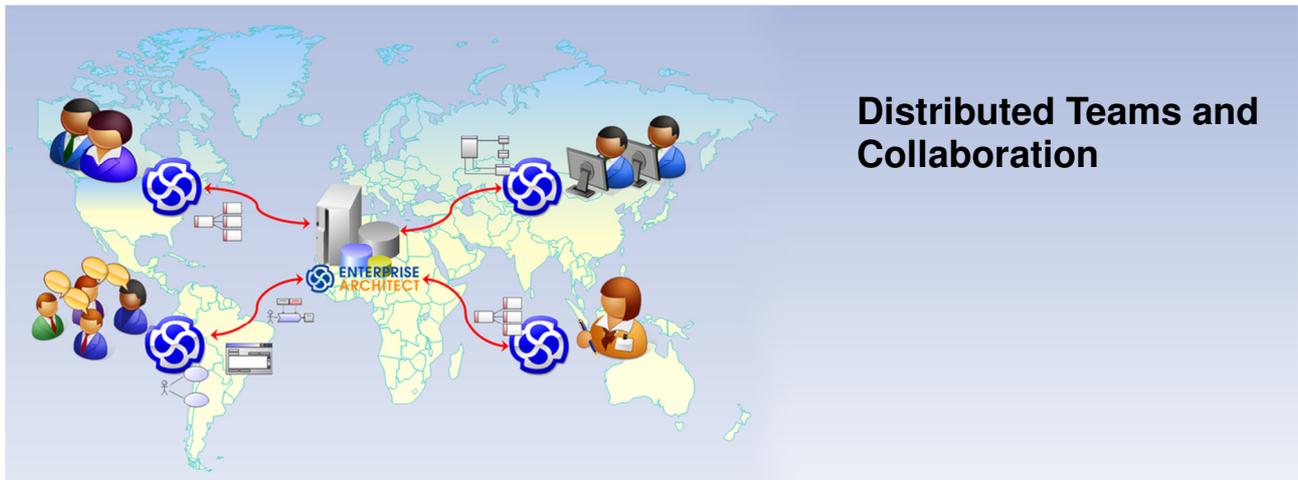
Distributed Teams and Collaboration

Enterprise Architect supports sharing of projects in team-based and distributed development environments. Projects can be shared via a cloud server, network deployment of model repositories, replication, XMI Import/Export, Version Control, Package Control and User Security.

Support for large models and many concurrent users

The Corporate Edition of Enterprise Architect enables the use of dedicated DBMS (server based) repositories rather than the standard .EAP files to store shared model data. Enterprise Architect supports the following DBMSs as model repositories:

- Access 2007
- MS SQL Server
- MySQL
- Oracle

- PostgreSQL
- Firebird
- Adaptive Server Anywhere.

Cloud sharing model repositories

Enterprise Architect cloud server lets you host and connect to your model repositories in a local or public cloud. There are several benefits to sharing and accessing models this way:

Firstly, it helps you access shared models no matter where you are. With a cloud-enabled model, team members only needs Enterprise Architect installed on their machine and an internet connection. They do not need to install client database drivers that match the particular RDBMS product that you chose as a model repository.

Secondly, Enterprise Architect's **Connect-To-Cloud** facility is geared for enhanced performance over remote networks. It has been specifically designed to enhance performance over a WAN by reducing network chatter between the model repository and the Enterprise Architect client.

Corporate modeling teams will find the cloud connection particularly helpful. For example, if you're only permitted to share information over http or https in a locked down environment, you can still access and edit shared models, because the Connect-To-Cloud option only uses http or https.

Creating shared models is easy with the Cloud Services Client. You might use a dedicated DBMS back end for the model repository or take advantage of the Cloud Server's built-in Firebird server. Either way, administrators can easily create shared models on the fly and monitor active repository connections.

Enterprise Architect's cloud server also supports two additional standards for model sharing – the Reusable Asset Service (RAS) and Open Service for Lifecycle Collaboration (OSLC).

Reuseable Asset Service

Within a large organization, groups of users – such as model developers – can be separated by geographical distance and/or being on different networks. This can make it difficult to share common data, standards and modeling structures easily, without the complexity of using external version control tools or manually distributing model files between projects. However, within Enterprise Architect the Reusable Asset Service (RAS) provides a simple and convenient mechanism for modelers to distribute or download reusable model structures, information, corporate directives and standards, through a shared repository, accessible via a Cloud Service connection. The person who sets up the reusable data can retain ownership and management of the resource – or asset – whilst their distant colleagues can quickly review the currency of the information and download the latest versions into their models or file folders.

The RAS gives distributed teams convenient access to a single 'source of truth' for shared data, including project milestones, architectural frameworks and industry standards.

Security

Role-based (user) security in Enterprise Architect helps you to control access to various editing functions by requiring that authors log in to the model with certain privileges. It also allows model authors to lock elements per-user or per-group. This can help to improve collaborative modeling by preventing different users unintentionally editing the same information at the same time. It also limits the possibility of inadvertent model changes by users not designated as model authors.

XML import and export

Enterprise Architect supports an XML-based model interchange format known as XML Metadata Interchange (XMI). You can use Enterprise Architect's XMI facility to share model information between developers. XMI enables you to export discrete packages or entire model branches into XML files, which can be imported into other models or maintained in a version control repository.

The Team Review facility

The Enterprise Architect Team Review facility helps users to discuss the development and progress of a project. Team members can view and post messages within the modeling environment and can link their posts directly to elements within the model. For distributed team environments, users can connect their Enterprise Architect model to a Team Review hosted in a remote model repository.

Element Discussions

Using the dockable Element Discussions window you can develop a discussion on, or review of, an individual element. All comments are directly associated with that element. This makes it easy to locate, view and contribute to discussions on a selected element.

You can review and participate in discussions simply by clicking on the relevant element in the model; the Element Discussions window immediately switches to the discussions of the new selected element.

As a discussion develops, the number of responses is displayed so you can see when a discussion has a new reply without having to expand and work through the thread. You can also set a status on a discussion, helping you to resolve and close off points that have been raised.
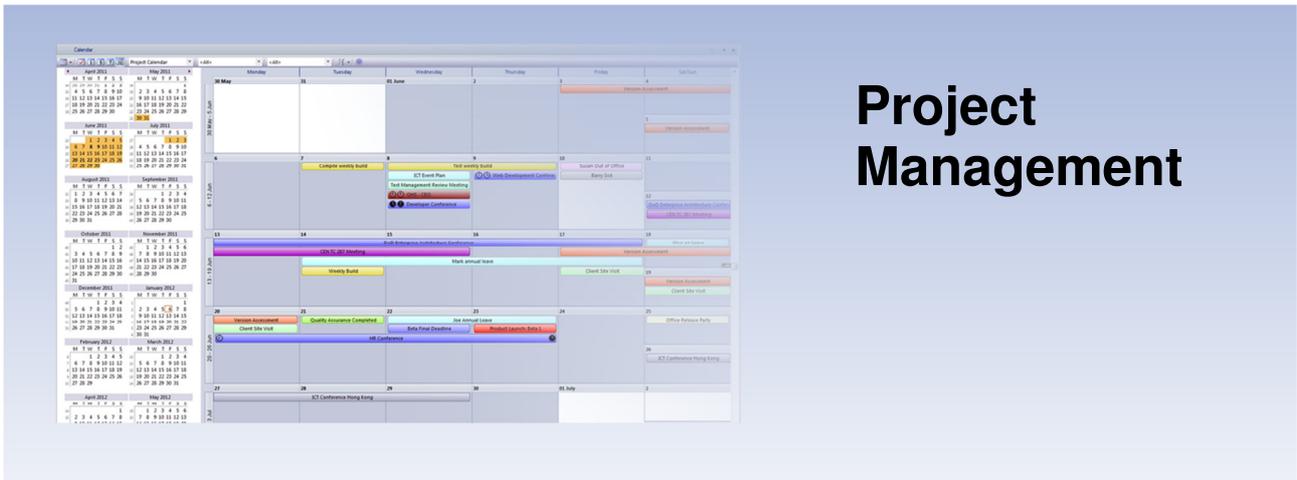
Project Management

Enterprise Architect provides support for managing your project. Project managers can use Enterprise Architect to assign resources to elements, to measure risk and effort and to estimate project size.

Project estimation with use case metrics

The Use Case Metrics capability within Enterprise Architect makes it easy to assess the relative complexity of a software project, based on the number and type of use cases within the model, and on the type of development project and the capabilities of the development environment. With experience, the use case metrics approach is a great way to quickly assess the scope of a project.

Resources

Enterprise Architect can store resource information along with the model. This helps to connect the Project Manager and development team, allowing for up-to-the-minute reporting and assessment of how a project is developing. What 'hot spots' have occurred, what elements are overdue and other critical resource issues can be tied directly to model elements and searched with ease using the comprehensive Model Search facility.

Model tasks, Gantt View and Project Calendar

Tracking, prioritizing and assigning tasks is a critical part of managing a model's development over time. Enterprise Architect allows you track model tasks against individual resources and against the project as a whole. Task allocations can be viewed as a Gantt chart, allowing you to monitor progress visually. The built-in Project Calendar allows you to define and track important events, milestones and meetings directly within the modeling environment.

Each model author has access to a personalized view of the project via the Personal Information window, which allows users to record progress of their own tasks, send and receive Model Mail and define Working Sets of diagrams and other views specific to their roles.

Project Glossary

Enterprise Architect adds a model glossary to each development project, making it simple to define and disseminate new terms and definitions, which may be unfamiliar to team members new to the project or the problem domain.

Change Management and Version Control
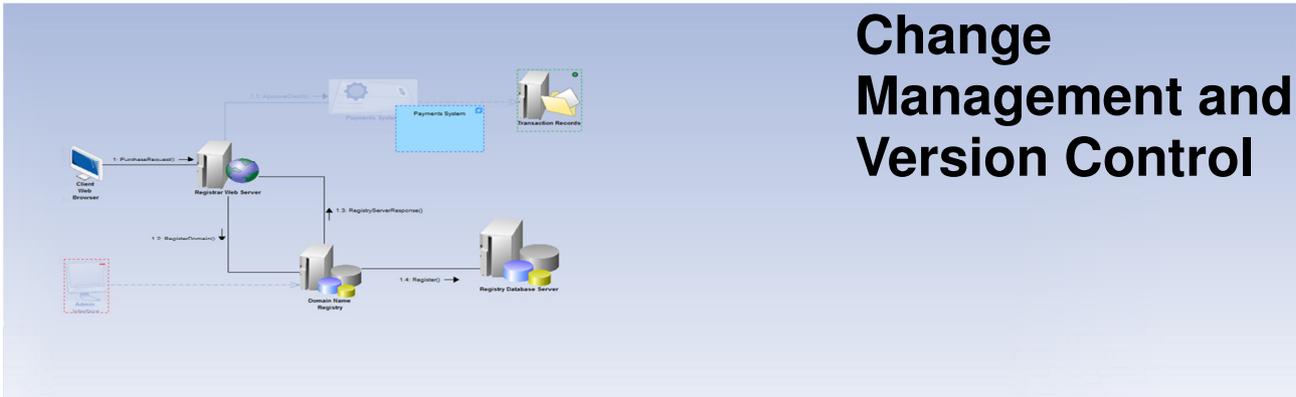
Enterprise Architect supports version control of model packages and their sub-packages. Package revisions can be stored and managed in an Enterprise Architect project directly, or they can be maintained using a dedicated third-party version control application.

Applying version control to Enterprise Architect models provides two key benefits:

- The ability to coordinate sharing of packages between users

- A history of changes to Enterprise Architect packages, facilitating retrieval of prior versions.

Enterprise Architect supports **Subversion**, **CVS** and Microsoft Team Foundation Server (**TFS**) version control applications, as well as any version control product that complies with the Microsoft Common Source Code Control (**SCC**) standard, version 1.1 or higher. For example Visual Source Safe or Clear Case.

Model baseline, compare and merge

Enterprise Architect Corporate edition provides a facility to 'Baseline' (snapshot) a model package at a particular point in time. The baseline can then be used with Enterprise Architect's Compare (diff) utility to visually explore changes to the package, its elements or its diagrams at a later point in development. Differences can be merged from the baseline into the current model, allowing you to 'roll-back' changes to a previous revision of that package. Multiple users can thus contribute revisions to a package offline and later incorporate them back into the common model.

A powerful **diagram compare** utility helps you to visually analyze changes to diagrams between revisions. Color-coded change items and connectors help you to see what has been added, deleted or moved in the diagram – with the ability to instantly restore any element to a previous state if required.

As well as comparing and merging changes from a baseline stored within the current model, Enterprise Architect's Compare utility allows you to compare a package against:

- A file on disk, created using the Enterprise Architect XMI export facility on the package

- A version-controlled XMI file for the selected package

- Any baseline of the package residing in an external model to which you have access.

Enterprise Architecture and SOA
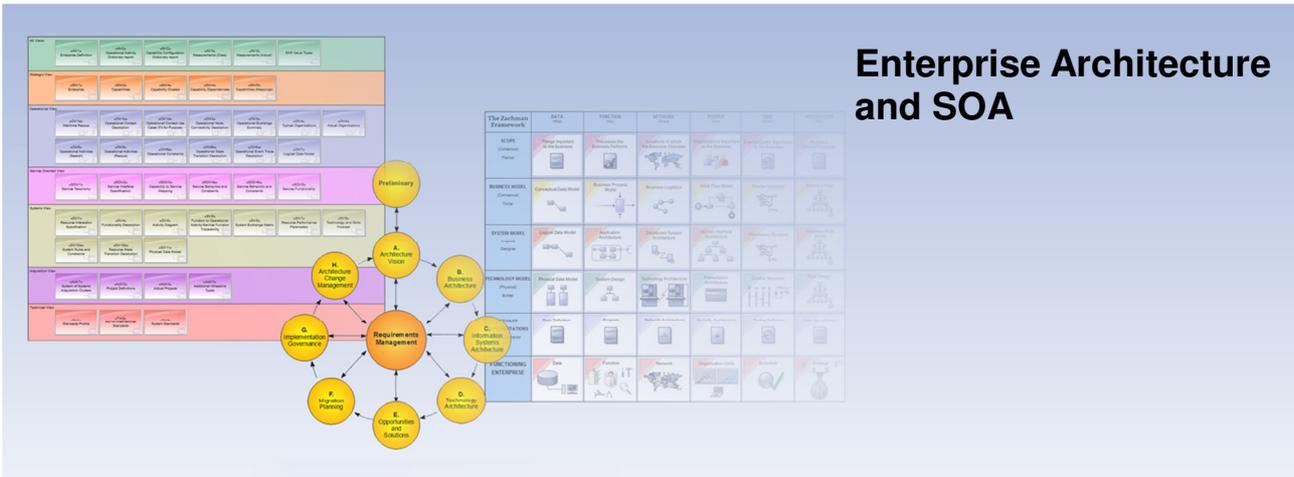
Enterprise architecture frameworks

Sparx Systems supports industry-standard architectural frameworks to facilitate enterprise modeling. Framework implementations in Enterprise Architect are based on the UML and its related specifications, which maximizes architectural rigor and allows users to exchange enterprise model information using standards such as XMI. The following architectural frameworks are available as plug-in extensions to Enterprise Architect:

- The Zachman Framework (see http://www.sparxsystems.com/zachman)

- DoDAF (see http://www.sparxsystems.com/dodaf-modaf)

- MODAF (see http://www.sparxsystems.com/dodaf-modaf)

- UPDM (see http://www.sparxsystems.com/updm)

- The Open Group's TOGAF (see http://www.sparxsystems.com/togaf).

  Including the Federal Enterprise Architecture Framework (FEAF) reference model

In addition to architectural frameworks, Enterprise Architect has built-in support for The Open Group's latest ArchiMate 2.0 enterprise modeling notation.

Service oriented architecture (SOA)

Enterprise Architect implements the OMG's Service oriented architecture Modeling Language (SoaML), which provides a standards-based approach to modeling SOA solutions using the UML.

As a natural complement to SoaML, Enterprise Architect also supports the Service Oriented Modeling Framework (SOMF), which encourages a holistic view of enterprise software entities via a technology-independent notation. The SOMF modeling notation provides an intuitive approach to visualize "used-to-be", "as-is" and "to-be" states of the enterprise service portfolio.

XSD and WSDL capability

Enterprise Architect can model, forward engineer and reverse engineer two key W3C technologies: XML Schema (XSD) and Web Service Definition Language (WSDL).

XSD and WSDL support is critical for the development of a complete Service Oriented Architecture. The coupling of UML with XML provides a natural mechanism for specifying, constructing and deploying XML-based SOA artifacts within an organization. Enterprise Architect's XSD and WSDL capabilities also support BPEL generation from UML models.

XSLT Debugger

Using built-in visual analysis tools, Enterprise Architect provides comprehensive support for working with XSLT documents, allowing you to perform transformations on XML data with the ability to debug the transformation process.

When working with XSLT documents you can:

- Create XLST, XSD and XML documents as model elements (UML Artifacts)
- View and Edit XML content in a dedicated Document Editor with a structural overview
- Run XML transformations
- Step through the execution transformations using built-in debug tools
- Inspect the instance values of transformation variables.

Schema Composer

The Schema Composer helps you rapidly build XSD schemas (and other XSD-based schemas) from selected elements in your model. The elements may be selected from your own models or sourced from generic reference models like CIM, NIEM and UN/CEFACT NDR. The Schema Composer can compose messages based on a variety of schema sets including:

- Common Information Model (CIM)
- National Information Exchange Model (NIEM)
- Core Components (UN/CEFACT) - NDR 3.0
- Core Components (UN/CEFACT) - NDR 3.1
- Universal Business Language (UBL) - NDR 3.0

Depending on the selected schema set, the Schema Composer can then export messages in the following formats:

- Generic XML Schema (XSD)
- JavaScript Object Notation (JSON)
- Resource Description Framework Schema (RDFS)
- CIM Augmented RDFS
- Business Data Type (BDT)
- Business Information Entity (BIE)
- Unqualified Data Type (UDT)
- Qualified Data Type
- National Information Exchange Model (NIEM)
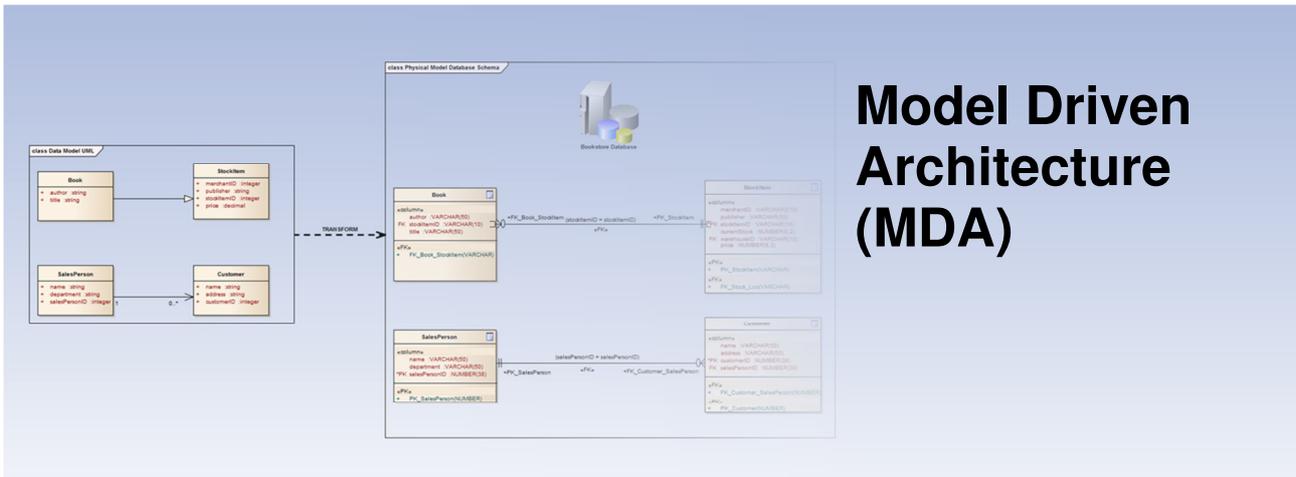
Model Driven Architecture (MDA)

Enterprise Architect provides the capability of executing MDA transforms. It provides a fully configurable method of converting model elements and model fragments from one domain to another. This typically involves converting Platform-Independent Model (PIM) elements to Platform-Specific Model (PSM) elements. A single element from the PIM could be responsible for creating multiple PSM elements across multiple domains.

Transformations are a huge productivity boost, and reduce the need to manually implement stock classes and elements for a particular implementation domain. For example, database tables can be automatically derived from persistent PIM classes.

Enterprise Architect's MDA capabilities allow you to:

• Leverage built-in transformations to automatically derive:

o Data Models (DDL)

o Code Models, including C# and Java

o XML models, such as XSD and WSDL

o Test Models for JUnit and Nunit

o Activity diagrams and test scripts from Structured Use Case Scenarios

• Define new transformations using a powerful, template driven approach

• Repeat transformations to ensure consistency between source and target models as they change over time.
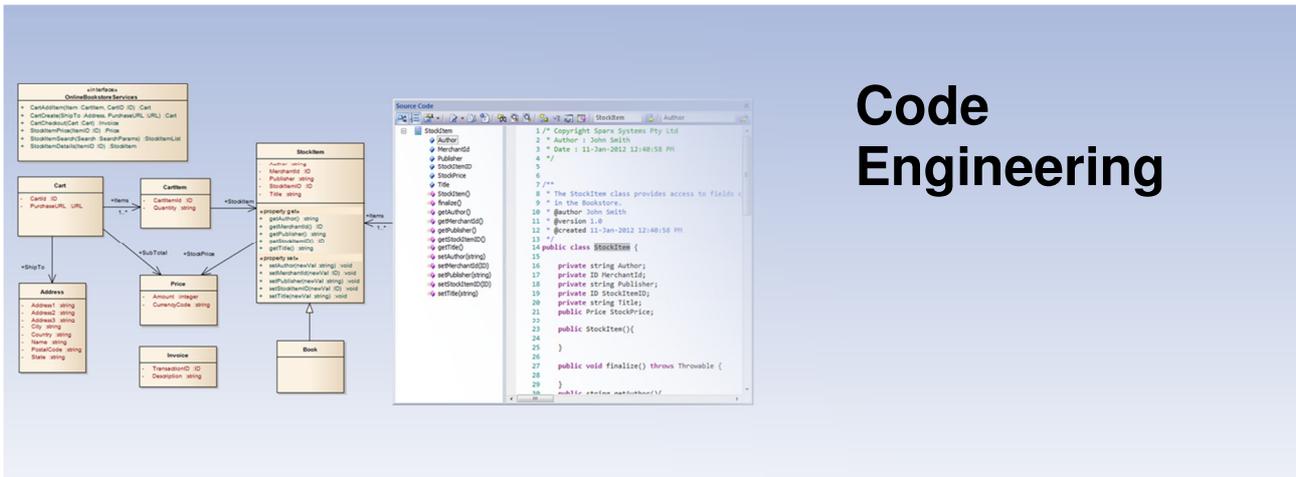
Code Engineering

Code engineering encompasses automated code generation, reverse engineering of source code and synchronization between the source code and model. Enterprise Architect supports code engineering for **more than ten programming languages out of the box!**

- ActionScript
- C
- C# (for.NET 1.1 and .NET 2.0)
- C++ (and .NET managed extensions)
- Delphi

- Java (including Aspects and Generics)
- PHP
- Python
- Visual Basic
- Visual Basic .NET.

Import .jar files and .NET assemblies

In addition to source code files, Enterprise Architect reverse-engineers binary modules from:

- Java Archive files (.jar)
- .Net PE files (.exe, .dll)*
- Intermediate Language files (.il).

Customizable source code generation

Enterprise Architect's **Code Template Framework** provides powerful, flexible forward engineering of UML models into source code. Code templates specify customizable transformations from UML elements into a target programming language. This means you can tailor generated source code to suit your standards. Furthermore, you can use the Code Template Framework to forward engineer additional languages, that are not already built into Enterprise Architect.

On-demand and live-generation of code

Enterprise Architect provides a Live Code Generation feature that automatically updates your source code as you make changes to your model. For example, when you create new operations and attributes for a class in the model, these are instantly written out to the source file.

## Built-in syntax highlighting source code with dynamic source 'outliner'

You can use the built-in source code editor to open, view and modify source code files. Simply select an element in the model and the editor displays its source code with syntax highlighted and a navigable code outline. Use the editor's toolbar to quickly generate code or synchronize the model.

Visualize, Debug and Profile Executing Code

Enterprise Architect's Visual Execution Analyzer provides facilities to model, develop, debug, profile and manage an application from within the modeling environment. Outputs generated by the Visual Execution Analyzer benefit the development process by:

- Giving you a better understanding of how your system works

- Enabling you to document system features automatically

- Providing information on the sequence of events that lead to erroneous events or unexpected system behavior.

Analysis tools provided by the Visual Execution Analyzer can be used to:

- Generate Sequence diagrams, recording live execution of an application or specific call stacks. Visualize how multiple instances of each class interact at run-time

- Derive State Transition diagrams, illustrating changes in data structures

- Dynamically build Object diagrams as you control a debug session. Selectively add objects with their run-state information and their relationships to other objects

- Create Profiler reports, showing application sequences and operation call frequency

- Optimize existing system resources and understand resource allocation

- Ensure that the system is following the rules as designed

- Produce high quality documentation that accurately reflects system behavior

- Understand how and why systems and existing code work

- Train new employees in the structure and function of a system

- Identify costly or unnecessary function calls

- Illustrate interactions, data structures and important relationships within a system

- Trace problems to a specific line of code, system interaction or event

- Visualize why a sequence of events is important

- Establish the sequence of events that occur immediately prior to system failure

- Debug .Net, C++ and Java applications, including remote debugging and support of Android physical devices and emulator using the JDWP protocol.

State Machine Code Generation and Execution

Enterprise Architect offers industry leading code generation from State Machinemodels. Generating code that **is clean and rea**dy to compile, Enterprise Architect removes the need to hand-write the source, helping you reduce development time and avoid 'human erro**r'. Detailed mod**el validation performed prior to code generation further helps you to define the right model. Advanced UML constructs are supporte**d, including** deep **history and par**allel regions, with an array of targe**t languages (C, C++, Jav**a, Javascript and more). Combined with customizable code generation templates, Enterprise Architect provides superior support for engineers that leverage State Machines.

Not only does Enterprise Architect generate quality code from State Machines, it can trace the application code back to the visual model as the program executes! The built-in Debugger leverages Enterprise Architect **model simulation** capability to highlight state transitions diagrammatically during execution. So you can effectively watch the State Machine execute within the modeling environment. As your compiled code executes, the diagrams update live to show what's happening. On top of that, Enterprise Architect's simulation capabilities help you **interact with and stimulate** the running application. For example, you can **fire triggers** to test how the system responds. So before you invest resources into deploying the State Machine application to its final environment – you can verify that both the design and the code behave as expected first. If you need to make adjustments, you can do so quickly and easily.

To simplify your modeling and assist with the generate-build-run process, Enterprise Architect provides an Executable State Machine element that encapsulate your state models. You can even model several of these artifacts to **generate and visualize interacting State Machines!**

Model Simulation
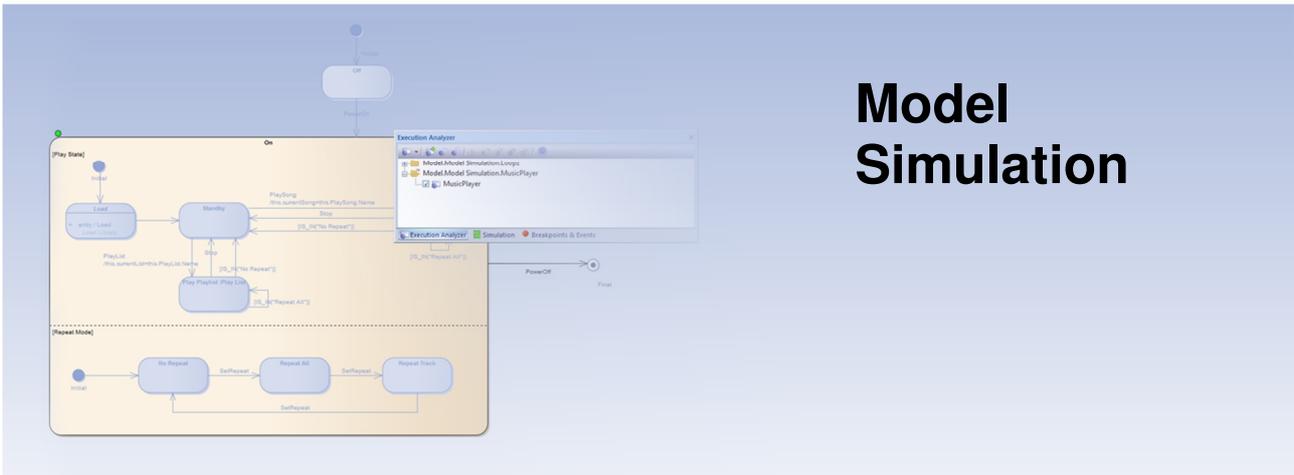
Enterprise Architect's model simulation brings your behavioral models and user interface designs to life with real-time execution. Simulating models offers several benefits by helping you to:

• Gain a better understanding of how a model actually works at run-time

• Validate that your behavioral models describe the correct process or event flow

• Verify the behavior of user interface wire-frames before committing to implementation

• Identify potential bottlenecks, inefficiencies and other problems in your system model or business process

• Detect errors early in the development cycle – prior to committing resources for implementation.

Model simulation can be applied to four types of behavioral models in Enterprise Architect, including:

• UML Activities

• UML Interactions

• UML State Machines, including those rendered as a State Table

• Business Process Models using BPMN notation.

You can also simulate the behavior of dialogs and controls that have been modeled using Enterprise Architect's Win32 profile for user interface designs.

You control the speed of the simulation and the pathways through the simulated model: Either manually control the choices taken at each decision point or script in advance how each trigger fires. Using the latter approach you can automate several simulations of the same model, revealing how the system behaves under different run-time scenarios. With the ability to set arbitrary breakpoints, Enterprise Architect's model simulation capability is a powerful tool for analyzing decision making, and improving business processes or executable system models in a risk-free environment.
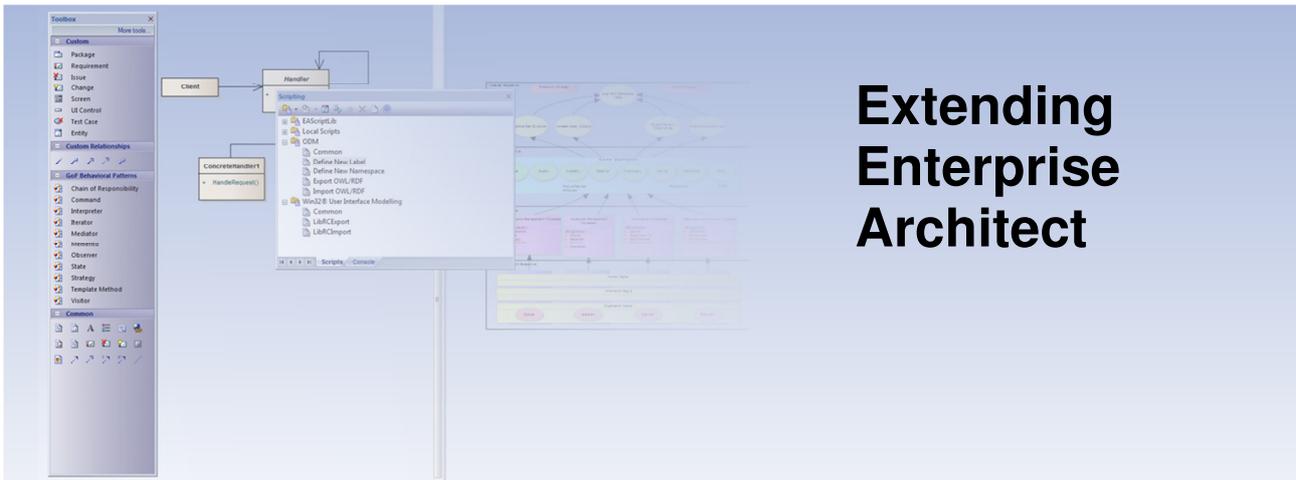
Extending Enterprise Architect

Enterprise Architect provides numerous facilities for extending functionality that is provided with the tool out-of-the-box. Users are able to expand the range of modeling notations for specific domains, add custom modeling resources such as model patterns, or even implement entirely new tools that operate on the model using the automation interface and add-ins.

Domain-specific modeling

UML profiles

UML Profiles extend the UML language for constructing models in particular domains. Enterprise Architect has a generic UML Profile mechanism that allows users to create, share and use profiles as seamless extensions to the core modeling environment. For example, the UML Profile for XML Schema defined by David Carlson in *Modeling XML Applications with XML* is available for Enterprise Architect. This profile describes a set of extensions to UML class models for accurate modeling of XSD schemas.

UML patterns

Enterprise Architect's support for UML Patterns provides an excellent means of achieving re-use and robustness. Patterns represent a group of collaborating objects and classes that can be abstracted from a general set of modeling scenarios. As patterns are discovered in any new project, the basic pattern template from previous engagements can be re-used with the appropriate variables renamed to suit the current project.

Patterns generally describe how to solve an abstract problem, and it is the task of the pattern user to modify elements in the pattern to meet the demands of the current engagement.

MDG Technologies

MDG Technologies encapsulate a logical collection of resources (such as UML patterns and profiles) that pertain to a specific technology or modeling domain. These are 'pluggable' resources for Enterprise Architect that reside either in a physical directory or URL.

Users may create their own MDG Technologies with the aid of **Profile Helpers**, which guide you through the process of creating a UML profile, associated toolboxes and new diagram types. Furthermore, users can take advantage of MDG Technologies that are packaged with the Enterprise Architect installer, such as ArchiMate and Mind Mapping. Free plug-in technologies that support such activities as CORBA code engineering and BPMN model validation are also available for download from: http://www.sparxsystems.com/resources/mdg_tech. In addition, licensed solutions for systems modeling (SysML), Distributed Data Services (DDS) and other modeling domains are available – see http://www.sparxsystems.com/products/#technology.

Automation interface

The Automation Interface enables you to access the internals of Enterprise Architect models. For example, using the Automation Interface you could:

- Perform repetitive tasks, such as update the version number for all elements in a model

- Generate code from a state machine diagram

- Produce custom reports

- Execute ad hoc queries against the model.

All development environments capable of generating ActiveX Com clients should be able to connect to the Enterprise Architect Automation Interface.

Add-ins

Add-ins enable you to add functionality to Enterprise Architect and extend its user interface. Enterprise Architect's Add-in framework builds on the Automation Interface, providing several key advantages over stand-alone automation clients:

- Add-ins can define Enterprise Architect menus and sub-menus

- Add-ins receive notifications about various Enterprise Architect user-interface events including menu clicks and file changes

- Add-ins can (and should) be written as in-process (DLL) components; this provides lower call overhead and better integration into the Enterprise Architect environment

- Because a current version of Enterprise Architect is already running there is no need to start a second copy of Enterprise Architect via the automation interface

- Because the add-in receives object handles associated with the currently running copy of Enterprise Architect, more information is available about the current user's activity, such as which diagram objects are selected

- You are not required to do anything other than install the add-in to make it usable; that is, you do not have to configure add-ins to run on your systems.

Connectivity and integration with other tools

Enterprise Architect provides a number of mechanisms for integrating your model with third-party tools, including a programmable API, an add-in framework and off-the-shelf Model Driven Generation (MDG) solutions provided by Sparx.

MDG Link

Sparx has developed a number of MDG products to provide interoperability with other tools. MDG Link products exemplify use of the Add-in framework to extend the functionality of Enterprise Architect. MDG Link for Visual Studio and MDG Link for Eclipse enable Enterprise Architect to interoperate with the Microsoft® Visual Studio® and Eclipse IDEs respectively.

MDG Integration

MDG Integration tightly integrates Enterprise Architect into the Eclipse and Microsoft® Visual Studio® 2005/2008/2010/2012/2013 development environments. This product enables users to explore and edit the UML model inside Visual Studio or Eclipse and also provides many of the key features of Enterprise Architect directly within these IDEs, including rich text and web-based document generation, MDA transformations, baseline management and engineering of key XML-based technologies.

Other Eclipse-based environments that MDG Integration supports include Adobe® Flex® Builder™ and Progress OpenEdge® Architect.

MDG Integration for Microsoft Office

Provides integration of Enterprise Architect with the Microsoft® Office suite, including the following components:

- Word
- Excel
- PowerPoint
- Visio

The integration allows you to leverage information captured in Word documents, Excel spreadsheets and Visio diagrams in your enterprise model, as well as publish models via PowerPoint.

Import - export capability

Exchanging data between different models, and between different tools, is an integral part of any development project. Enterprise Architect supports both of the two most common data exchange formats (XMI and CSV), making it easy for information and models to be ported into different tools and repositories as needed. Coupled with the Enterprise Architect API, you can automate model import/export to simplify dissemination of models and automate software construction.

Variety of XMI formats

Enterprise Architect supports import and export in a wide range of XML Model Interchange (XMI) based formats. XMI is a specification for how to render complex model information to human-readable XML, for the general purpose of exchanging information with other tools. XMI is an open standard managed by the OMG. Enterprise Architect supports XMI 1.0, XMI 1.1 and XMI 2.1. Support for a range of formats is essential, as many tools expect a specific XMI version.

Open Services for Lifecycle Collaboration (OSLC)

Enterprise Architect acts as an OSLC Provider and supports the Requirements Management 2.0 specification of OSLC, which allows for creating, retrieving and querying the Requirements in a model accessed via a Cloud connection. Requirements in an Enterprise Architect model are accessible using a unique URL that can be linked to resources in other lifecycle products and tools.

CSV

In addition to XMI import/export, Enterprise Architect offers a simple CSV data import/export facility. This is useful for outputting information to tools such as Microsoft Excel, and for importing similar lists of elements stored in spreadsheets.

Enterprise Architect editions

Enterprise Architect is available in six editions: Ultimate, Systems Engineering, Business and Software Engineering, Corporate, Professional and Desktop. Each edition offers a range of features to support the requirements of different groups of users, from single-person projects to large enterprise teams.

A floating license arrangement is also available for Ultimate, Systems Engineering, Business and Software Engineering and Corporate editions. The floating license is particularly useful for companies that need to manage a central store of license keys. Floating license keys may be used by different employees over time, on a temporary or permanent basis.

More information about Enterprise Architect editions is available from:
http://www.sparxsystems.eu/enterprisearchitect/enterprise-architect-editions/ and
http://www.sparxsystems.eu/enterprisearchitect/feature-matrix/

Process support

UML is a language, not a process. It prescribes the elements of a modeling language and how those elements can be joined together to represent things in the real world. It does not prescribe how you use those elements over time to build new software systems.

Like UML, Enterprise Architect is process neutral, meaning that it includes all the features and functionality necessary to implement some chosen development process, but does not dictate what that process should be or how it should be implemented.

Many Enterprise Architect users adopt highly structured processes, like RUP, while others use more flexible and lighter weight Agile processes. Regardless of the degree of process management you require, Enterprise Architect has the tools and features needed to support the ongoing process of software development.

About Sparx Systems

Sparx Systems is an Australia-based company with a solid history of innovation and development within the modeling tools market.

Sparx Systems is a Contributing Member of the Object Management Group (OMG), the standards body responsible for defining and maintaining the UML and related specifications.

Company vision

Sparx Systems believes that a complete modeling and design tool should be used throughout the full software life-cycle. Our subscription plan reflects this, as does our belief that 'life-cycle' software should be as dynamic and modern as the systems you design and maintain.

Modeling software developed by Sparx Systems is intended for use by analysts, designers, architects, developers, testers, project managers and maintenance staff; that is, almost everyone involved in a software development project and in business analysis. It is Sparx Systems' belief that highly priced CASE tools severely limit their usefulness to a team, and ultimately to an organization, by narrowing the effective user base and restricting easy access to the model and the development tool. To this end, Sparx Systems is committed to maintaining an accessible pricing model and to distributing a free 'Read Only' (Enterprise Architect Lite) version of Enterprise Architect for use by those who only need to view model information.

Ongoing commitment to enterprise modeling tools

Sparx Systems has been developing enterprise modeling tools for over ten years, and has been active in developing the capabilities of Enterprise Architect to reflect the needs of enterprise modeling. In addition, Sparx has been diligent in listening to requests and suggestions from its extensive user base, implementing many features critical to developing useful and well-rounded models. Sparx Systems is committed to the ongoing development both of its UML modeling tool, Enterprise Architect, and of the numerous plug-in technologies that address domain specific requirements.

Contact details

SparxSystems Software GmbH an independent subsidiary of Sparx Systems PTY Australia

SparxSystems Software GmbH Central Europe
Office: Handelskai 340/5, 1020 Vienna, Austria
Tel.: +43 (0)662 90 600 2041
Fax: +43 (0)662 90 333 3041
vendor@sparxsystems.eu
www.sparxsystems.eu
blog.sparxsystems.eu