

**Sparx Systems**

# **Enterprise Architect 9.3**

**im Überblick**

*© Copyright SparxSystems GmbH 2012*



[www.sparxsystems.eu](http://www.sparxsystems.eu)

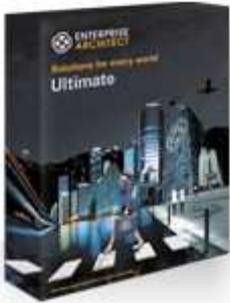
# Inhalt

<i>Einleitung</i> .....	4
<i>Was ist Enterprise Architect?</i> .....	4
<i>Was unterscheidet Enterprise Architect von anderen UML Werkzeugen?</i> .....	4
<i>Wie verbreitet ist Enterprise Architect heute?</i> .....	5
<i>Welche Vorteile bietet Enterprise Architect?</i> .....	5
<i>Enterprise Architect – Funktionen im Überblick</i> .....	8
<i>Auf offenen Standards basierende Modellierung</i> .....	9
<i>UML, BPMN und SysML</i> .....	9
<i>Einige Enterprise Architect Features im Detail</i> .....	10
UML 2.4.1 Unterstützung .....	10
Geschäftsprozessmodellierung und Simulation .....	11
<i>Das BPMN Profil</i> .....	11
<i>Die BPEL-Generierung</i> .....	11
Datenmodellierung .....	12
Das eingebaute Datenmodellierungsprofil .....	12
Unterstützte DBMS Produkte.....	12
Anforderungsmanagement .....	13
Strategisches Modellieren .....	14
Systementwurf (Systems Engineering) .....	15
SysML und Modellsimulation.....	15
Erweitertes Verhaltensmodellieren und erweiterte Codeerzeugung .....	15
Hardwarebeschreibende Sprachen .....	15
Test Management .....	16
In den Entwicklungszyklus integriertes Unit-Testing.....	16
Benutzeroberfläche, Werkzeuge und Arbeitsbeschleunigung.....	17
Freie Diagrammpositionierung versus Verankerung .....	17
Direkte Dateneingabe zu Diagrammbestandteilen .....	17
Der Schnellzeichenpfeil .....	18
Weitere Diagrammfunktionen.....	18
Modell- und Anwendungsmuster (Pattern) .....	18
Modellprüfung (Validierung).....	18
Dokumentenerstellung und Reports .....	19
<i>Rich-Text Format (RTF) Ausgabe</i> .....	19
<i>Der HTML Generator</i> .....	19
<i>Der eingebaute Dokumenteneditor und das Verlinken von Dokumenten</i> .....	19
Strukturierte Szenarios/Storyboards.....	19
Verkettung und Nachweisfähigkeit .....	20
<i>Protokollierung (Auditing)</i> .....	20
<i>Abweichungs- und Abhängigkeitsanalyse</i> .....	20
<i>Abweichungs-Analyse-Matrix</i> .....	20
<i>Abhängigkeits-Analyse-Matrix</i> .....	20
<i>Das Verkettungsfenster – Traceability View</i> .....	20
<i>Suche nach Elementen und Meta-Daten</i> .....	21
<i>Die Modellsuche</i> .....	21
<i>Die Modellsichten (Model Views)</i> .....	21
<i>Weitere Reports</i> .....	21
Entwicklung und Zusammenarbeit im Team .....	22
<i>Unterstützung großer Modelle oder vieler gleichzeitiger Anwender</i> .....	22
<i>Security - Sperrfunktion</i> .....	22

<i>XML Import und Export</i> .....	22
<i>Die Team Review Funktionalität (internes Diskussionsforum)</i> .....	23
Projektmanagement .....	24
<i>Aufwandsbestimmung mit der Use Case-Point-Methode</i> .....	24
<i>Ressourcen</i> .....	24
<i>Aufgaben, Gantt-Darstellung und Projektkalender</i> .....	24
<i>Projektglossar</i> .....	24
Änderungsmanagement und Versionskontrolle .....	25
<i>Baseline, Vergleich und Mischen (Merge)</i> .....	25
Enterprise Architecture Frameworks.....	27
Service Oriented Architecture (SOA) .....	27
XSD und WSDL Unterstützung .....	27
Modellgetriebene Architektur (MDA) .....	28
Codebearbeitung.....	29
<i>Import von .jar-Dateien und .NET Assemblies</i> .....	29
<i>Gestaltbare Codeerzeugung</i> .....	29
<i>Bedarfsgesteuerte oder fortwährende Codeerzeugung</i> .....	29
<i>Syntaxhervorhebung und dynamische Einrückung</i> .....	30
Darstellung, Debugging und Profilierung von ausführbarem Code.....	31
Modellsimulation .....	32
Enterprise Architect erweitern .....	33
Domänenspezifisches Modellieren .....	33
<i>UML Profile</i> .....	33
<i>MDG Technologien</i> .....	33
<i>Automation Interface</i> .....	33
<i>Add-Ins</i> .....	34
Anbindung an und Integration mit anderen Werkzeugen .....	35
Enterprise Architect Editionen .....	36
Prozessunterstützung .....	36
Über Sparx Systems .....	37
Firmen-Vision .....	37
Langfristige Ausrichtung auf unternehmensweite UML Werkzeuge .....	37
Kontakt-daten .....	38

# Einleitung

Dieses Dokument gibt einen umfassenden Überblick über die Funktionalität und die Leistungsfähigkeit von Sparx Systems' **Enterprise Architect 9.3 (EA)**. Jedem speziellen Merkmal wird ein Abschnitt gewidmet - jeweils mit Erläuterung des Zwecks und des Nutzens der Funktion.



## Was ist Enterprise Architect?

Enterprise Architect ist ein grafisches Computer Aided Software Engineering (CASE) Tool, also ein Softwareentwicklungswerkzeug zum Entwurf und zur Herstellung von Softwaresystemen, zur Geschäftsprozessmodellierung und zur Modellierung beliebiger Prozesse oder Systeme.

Enterprise Architect stützt sich auf die letztgültige UML® 2.4<sup>1</sup> Spezifikation (Unified Modeling Language, siehe [www.omg.org](http://www.omg.org)). UML definiert eine grafische Symbolsprache, um ein System zu beschreiben – während der

Planungsphase oder zur Dokumentation eines bestehenden Systems.

Enterprise Architect ist ein fortschrittliches Werkzeug, das **alle Teile des Entwicklungszyklus** abdeckt und eine umfassende Nachvollziehbarkeit des Projekts vom ersten Entwurf bis zur Auslieferung und bis in die Wartungsphase hinein zulässt. Requirements Management, die Testphase und auch das Änderungsmanagement werden unterstützt.

## Was unterscheidet Enterprise Architect von anderen UML Werkzeugen?

- Die umfassende Unterstützung von UML® 2.4
- Die Möglichkeit, Anforderungen vollständig zu managen (*Requirements Management*).
- Eine umfassende Projektmanagementunterstützung, einschließlich Ressourcenplanung, Aufgaben, Projektkalender, Metriken, Testvorgaben und Testverfolgung.
- Eine Testunterstützung: Testfälle, JUnit und *NUnit* werden unterstützt.
- Flexible Dokumentationsfunktionen: HTML und Rich-Text (RTF) Reportgeneratoren.
- Codeerstellungsunterstützung für zahlreiche Programmiersprachen 'out of the box'.
- Eine integrierte Debuggingfunktion für ausführbare Java und .Net Projekte/Programme, mit automatischer Erstellung eines Laufzeitmodells und eines Sequenzdiagramms aus dem jeweiligen Debuggingprotokoll.
- Eine erweiterbare Modellierungsumgebung mit der Möglichkeit, benutzerdefinierte Profile und Technologien einzubinden.
- Die leichte Bedienbarkeit.
- Geschwindigkeit: EA arbeitet spektakulär schnell.
- Skalierbarkeit: EA kann problemlos extrem große Modelle verarbeiten und unterstützt das parallele Arbeiten mehrerer Benutzer an einem Projekt.
- Kosten: Die Lizenzpreise von EA sind ausgelegt, um das ganze Team ausstatten zu können. Interaktive Zusammenarbeit und eine Entwicklung im Team werden dadurch Realität.

---

<sup>1</sup> UML®, BPMN™ und SoaML™ sind geschützte Handelsmarken der Object Management Group Inc.

## Wie verbreitet ist Enterprise Architect heute?

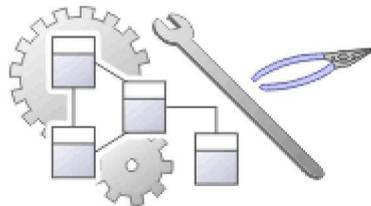
Mit mehr als 300.000 verkauften Lizenzen hat Enterprise Architect seine Popularität in vielen Industriesparten bewiesen und wird von tausenden Firmen weltweit genutzt. Von großen, bekannten, multinationalen Organisationen bis zu unabhängigen Kleinfirmen und Beratern ist Enterprise Architect das UML-Modellierungswerkzeug der ersten Wahl für Entwickler, Konsulenten und Analysten in über 1300 Ländern geworden.

Enterprise Architect von SparxSystems wird zur Entwicklung verschiedenster Softwaresysteme in einem breiten industriellen Spektrum verwendet: Luft-/Raumfahrt, Bankwesen, Internetentwicklung, Fertigung, Finanzwesen, Medizin, Militär, Forschung, Lehre, Transport, Einzelhandel, Energieversorgung und Elektrotechnik. EA wird auch weltweit in vielen prominenten Schulen, Universitäten und Schulungsunternehmen effektiv zur UML-Schulung und zur Businessprozessmodellierung eingesetzt. Die Einsatzbreite reicht von einzelnen Anwendern bis zu Großfirmen mit über 1000 Nutzern, die an großen, verteilten Projekten arbeiten.

## Welche Vorteile bietet Enterprise Architect?

*Modellieren und Managen von komplexer Information.*

Enterprise Architect unterstützt einzelne Personen und Gruppen bis hin zu großen Organisationen beim Modellieren und Managen von komplexen Informationssystemen. Meistens handelt es sich dabei um Softwareentwicklungen, IT-Systeme und deren Design und Entwicklung. Darüber hinaus kann Enterprise Architect zum Modellieren von Unternehmensstrukturen und Geschäftsprozessen verwendet werden. Enterprise Architect integriert und verbindet umfangreiche Struktur- und Verhaltensbeschreibungen um ein kohärentes und überprüfbares Modell eines Soll- oder Istzustandes zu erstellen. Weitere Features sind Funktionen zum Managen von Modellversionen, zur Verfolgung von Modelldifferenzen, Funktionen zum Konfigurieren von Zugriffsrechten und vieles mehr. Mithilfe dieser Konfigurationsmöglichkeiten unterstützt Enterprise Architect das strukturierte Erstellen und Abwickeln von Projekten – abgestimmt auf Qualitätssicherungsnormen.



*Modellieren, Managen und Verfolgen von Anforderungen.*

Enterprise Architect unterstützt die Erfassung von Basisanforderungen und deren nachvollziehbare Zuordnung zu Designanforderungen sowie Verteilungsanforderungen. Vorgeschlagene Anforderungen können analysiert, überprüft und akzeptiert werden. Dies ermöglicht die Nachvollziehbarkeit aller Anforderungen von deren Erstellung über Änderungen bis hin zur korrekten Realisierung des gewünschten Systems.



### *Zusammenarbeit im Team für eine gemeinsame Projektsicht.*

Eine skalierbare, einfach zu erstellende Mehrbenutzerumgebung erlaubt in Enterprise Architect Teammitgliedern aus verschiedenen Bereichen und Phasen eines (System-) Produktentwicklungs- und Wartungslebenszyklus den Zugriff auf für sie bestimmte Informationen. Die Architektur von EA-Projekten erlaubt eine gemeinsame Verwendung eines Repositories (Datenhaltung) für Businessanalytiker, Softwarearchitekten, Entwickler, Projektmanager, Tester, Roll-Out- und Supportmitarbeiter. Dadurch werden Sichten aus verschiedenen Blickwinkeln auf ein komplexes System und dessen Subsysteme möglich, ohne Inhalte mehrfach anlegen und führen zu müssen.



### *UML® basiertes Entwerfen und Entwickeln von Systemen.*

UML 2.4 ist ein offener Standard, der es erlaubt, Softwaresysteme, Business- und IT-Systeme zu entwerfen und zu dokumentieren. Durch eine interaktive Führung ermöglicht Enterprise Architect die Mächtigkeit von UML® umfassend zu nutzen.. Mithilfe der erstellten Modelle können Code, Datenbanken, Dokumentationen und Metriken erstellt werden, ebenso Transformation von Modellen und Struktur- und Verhaltensspezifikationen als Grundlage für vertragliche Vereinbarungen.



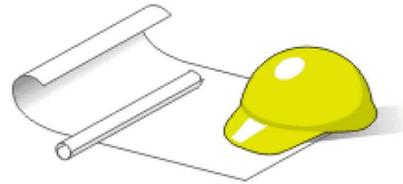
### *Visualisieren, Überprüfen und Verstehen komplexer Software.*

Softwaresysteme sind komplex und meist schwer zu verstehen. Verwenden Sie Enterprise Architect, um bestehenden Source Code reverse zu engineeren - um die statische Struktur der Implementierung zu verstehen. Um das bestehende Bild eines vorhandenen Systems zu vervollständigen, können die in Enterprise Architect eingebauten Debugging- und Aufzeichnungs-features verwendet werden. Damit können Laufzeitmodelle visualisiert werden. Die Datenmodelle bestehender Datenbanken verschiedener DBMS können ebenfalls reverse engineered werden, um sie als visuelles Modell zu repräsentieren. Darüber hinaus können Laufzeitinstanzen eines Modells erstellt werden, um sie mittels der Object Workbench in Enterprise Architect aufrufen zu können.



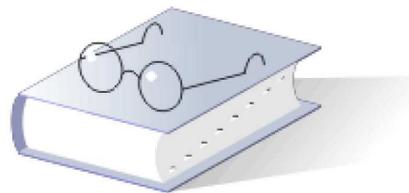
*EA unterstützt die Modellierung eines Systemlebenszyklus und das Projektmanagement.*

Das Erfassen und Verfolgen von erfolgskritischen Informationen über Modellelemente ist vorgesehen, zum Beispiel: Teststatus, Projektmanagement- und Wartungsdetails. Diese Informationen werden verwendet, um den aktuellen Entwicklungsprozess und den Projektzustand zu verfolgen.



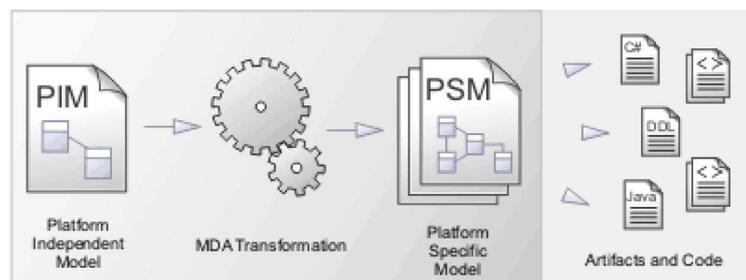
*Verteilen und Benützen von Informationen über Toolgrenzen hinaus.*

Enterprise Architect unterstützt mehrere Mechanismen zum Importieren und Exportieren von Modellen. Durch diese Mechanismen können Informationen aus anderen Tools benützt werden, sowie Informationen zwischen Enterprise Architect Modellen und Modellen anderer Modellierungswerkzeuge ausgetauscht werden. EA unterstützt mehrere XMI-Standards ([www.omg.org](http://www.omg.org)) und csv. Der Modellinhalt im Repository kann auch über ein im EA eingebautes Scripting-Tool (VB-, J- und Java-Script) gelesen und verändert werden. Sie können aber auch von außen aus jeder active-x/ole-fähigen Programmiersprache lesend und schreibend zugreifen – eine ausführlich Dokumentation unterstützt Sie dabei.



*Erstellung von plattformunabhängigen Modellen durch modellgetriebene Softwareentwicklung.*

Modellgetriebene Softwareentwicklung (*Model Driven Architecture*) (MDA) ist ein offener Standard um konzeptuelle und plattformunabhängige Modelle (*Plattform Independent Models*) (PIMs) mit Hilfe von Transformationen in verschiedene plattformspezifische Modelle (*Plattform Specific Models*) (PSMs) zu übersetzen. Aus den erstellten plattformspezifischen Modellen kann Code generiert werden. Enterprise Architect beinhaltet mehrere vordefinierte Transformationen, um konzeptuelle Modelle in plattformspezifische Modelle zu übersetzen. Diese Transformationen werden über Templates gesteuert, die der Anwender auch verändern kann. Ebenso können Sie weitere Transformationen mittels selbst erstellter, neuer Templates hinzufügen.



# Enterprise Architect – Funktionen im Überblick

EA ermöglicht:

- die Modellierung komplexer Software- und Hardwaresysteme in UML-gerechter Darstellung.
- die Modellierung und das Verwalten von Requirements und die Nachvollziehbarkeit deren Umsetzung im modellierten System – bis in den Code hinein.
- die Erstellung detaillierter und hochqualitativer Dokumentation in den Formaten RTF, PDF und HTML.
- die Nutzung von Industriestandards in Form von architektonischen Frameworks
- die Erzeugung und die Rückeinbindung (*Reverse Engineering*) von mehr als 10 Programmiersprachen; die bekanntesten sind: Action Script, Ada, C, C++, C#, Delphi, Java, PHP, Python, System C, Verilog, Visual Basic, VHDL und VB.NET<sup>2</sup>
- die Modellierung von Datenbanken und das automatische Erstellen von DDL Scripts, sowie die Rückeinbindung (*Reverse Engineering*) von Datenbankschemata direkt über ODBC Verbindungen
- die Versionierung eines Projektes (intern mittels *Baselines* oder durch Anbindung von VCS<sup>4</sup>) und das Aufzeichnen eines Änderungsprotokolls (*Audit*). Die Versionierung erlaubt Differenzbildung und das Wiederherstellen älterer Versionen, auch wahlfrei auszugsweise.
- den zentralen und unternehmensweiten Zugriff auf modellierte Inhalte.
- die Modellierung von Abhängigkeiten zwischen Modellelementen, wie z. B. das Verbinden von dynamischen Modellen mit statischen Modellen .
- das Modellieren von Implementierungsdetails, wie Klassenhierarchien, deren Zusammenfassung zu Komponenten und die physische Verteilung dieser Komponenten.
- die Verwaltung von Projektvorfällen (*issues*), Aufgaben (*tasks*) sowie die Verwendung eines projektspezifischen Wörterbuchs (*glossary*).
- eine Zuordnung von Ressourcen zu Modellelementen und die Erfassung und Gegenüberstellung von tatsächlichem und geplantem Aufwand.
- eine Ausgabe der Modelle in Formaten kompatibel zu XMI 1.0, XMI 1.1, XMI 1.2, XMI 2.0 und XMI 2.1 zum Import in andere XMI-kompatible Werkzeuge.
- eine Übernahme von Modellen in den Formaten XMI 1.0, XMI 1.1, XMI 1.2, XMI 2.0 und XMI 2.1 von anderen Werkzeugen. Weiters werden EMX und UML2 als Importformat unterstützt.
- eine Versionierung über XMI unter Benutzung von SCC, CVS oder Subversion
- die Benutzung von *UML Profilen*, um kundenspezifische Erweiterungen von UML bereitzustellen. Hierdurch können auch weitere Normen, z. B. die eEPK emuliert werden.
- komplette Diagramme als *UML Pattern* zu speichern und diese Pattern im selben oder auch in anderen Projekten wiederzuverwenden.
- die Analyse von Beziehungen zwischen Elementen in Tabellenform mittels der sogenannten Beziehungs-Matrix (*relationship-matrix*).
- Scripts zu schreiben um wiederkehrende Aufgaben durch ein *Automation Interface* (API Schnittstelle zu Enterprise Architect) zu automatisieren.

---

<sup>2</sup> In bestimmten Editionen, siehe <http://www.sparxsystems.de/uml/ea-price/>

- das Verbinden zu einem datenbankbasiertem EA-Repository. EA unterstützt eine Reihe von Datenbanksystemen als Projektrepository, wie z. B.: SQL Server, MySQL, Oracle9i, PostgreSQL, Adaptive Server Anywhere, und Progress OpenEdge
- die Verteilung von Modelländerungen basierend auf versionskontrollierten Paketen.
- die Erstellung von Modell-zu-Modell Transformationen mittels *Model Driven Architect* (MDA<sup>®</sup>) Technologien.
- das Erstellen und Verteilen von dynamischen Sichten auf EA Projekte (Elemente und Diagramme) mittels *Model Views*.
- das Erstellen von *Mind Maps*, *Business Process Models* und *Data Flow* Diagrammen basierend auf UML.
- die automatische Generierung von BPEL Skripten als Geschäftsprozessen in BPMN Notation.
- Generierung ausführbarer Geschäftsprozesslogik basierend auf Rule Tasks und die Nachvollziehbarkeit der Umsetzung zu natürlich-sprachlichen Geschäftsprozessregeln.
- die Visualisierung der ausgeführten Anwendung mittels Visual Execution Analyzers.
- die Transformation von Verhaltensmodellen in ausführbaren Code, sowohl für Software als auch für Geräte beschreibende Sprachen (HDLs) wie Verilog, VHDL und SystemC<sup>3</sup>.
- die Simulation von SysML parametrisierten Modellen<sup>2</sup>.

## Auf offenen Standards basierende Modellierung

Als beitragendes Mitglied der Object Management Group ist sich Sparx Systems der Bedeutung offener Standards als Kommunikationsmittel zwischen Anwendern bewusst. In diesem Sinne unterstützt Sie Enterprise Architect:

- Visualisierung von Systemen in der letztgültigen UML 2.4 Notation
- Dokumentation und Beschreibung von Geschäftsprozessen mit BPMN 2.0
- Modellierung und Simulation von System Engineering Projekten mit SysML 1.2
- Nutzung zahlreicher anderer offener Modellierungsstandards – alle in derselben Modellumgebung!



## UML, BPMN und SysML

Enterprise Architect unterstützt alle UML 2.4.1 Modelle und Diagramme. Sie können Geschäftsprozesse, Webseiten, Benutzeroberflächen, Netzwerke, Hardwarekonfigurationen, Meldungen und viele andere Aspekte Ihrer Entwicklung modellieren.

In Kurzform:

<sup>3</sup> In bestimmten Editionen, siehe <http://www.sparxsystems.de/uml/ea-price/>

- Enterprise Architect war das erste Werkzeug, das UML 2 umfassend im April 2004 zur Verfügung stellte.
- Enterprise Architect wurde an die Weiterentwicklung der UML 2 fortwährend angepasst.
- Enterprise Architect unterstützt alle 14 Diagrammarten der UML 2.

Zusätzlich zur UML unterstützt Enterprise Architect die Letztfassung der *Business Process Modeling (BPMN)* Notation, der *System Modeling (SysML)* Spezifikation. Weitere bereits enthaltene Modellnotationen sind unter anderem auch *ArchiMate®*, *SoaML* und *SOMF™*.

Enterprise Architect unterstützt zahlreiche weitere Diagrammarten, die UML überschreiten, zum strategischen Modellieren, Mindmappen, zur formalen Anforderungsspezifikation, für Datenflussdiagramme, zur Modellierung von Benutzeroberflächen und zum domainspezifischen Modellieren. Das Werkzeug stellt auch alternative Ansichten der UML-Basisdiagramme zur Verfügung - intuitiver und effektiver. Ein Beispiel dafür ist der Zustandstabelleneditor, der ein UML-Zustandsdiagramm als editierbare Logiktablelle.

## Einige Enterprise Architect Features im Detail

Jeder der nachfolgenden Abschnitte behandelt einen Teilaspekt der Fähigkeiten von Enterprise Architect und zeigt den Zweck und Nutzen der Funktion auf.

### **UML 2.4.1 Unterstützung**

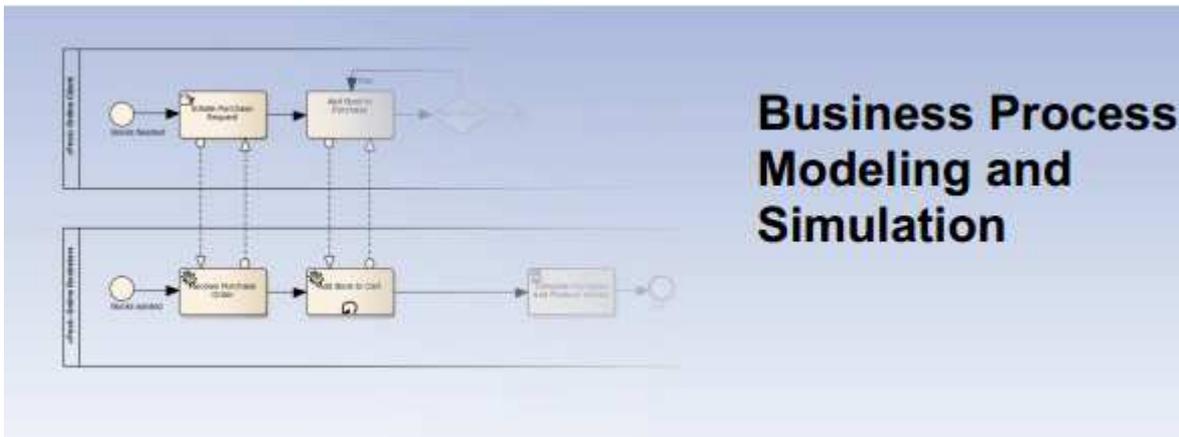
Enterprise Architect unterstützt alle in UML 2.4.1 definierten Modelle und Diagrammarten. Damit können Sie Geschäftsprozesse, Webseiten, Benutzerschnittstellen, Netzwerke, Hardware-konfigurationen, Nachrichten und viele andere Aspekte Ihrer Entwicklung modellieren. Sie können den Aufwand Ihrer Projektarbeit in Stunden planen. Sie können Anforderungen, Ressourcen, Testpläne, Störungen und Änderungsanträgen erfassen und verfolgen, vom ersten Konzept bis zur Produktivsetzung mit Wartung und Produktionsunterstützung. Enterprise Architect hat alle Funktionalitäten, die Sie für Ihren Entwurf und das Management Ihrer Umsetzung und Implementierung brauchen.

In Kurzfassung, EA

- war das erste UML-Werkzeug, das eine Abdeckung des UML 2 Standards implementierte - im April 2004
- wird fortlaufend an die UML 2 Erweiterungen angepasst
- unterstützt alle 14 Diagrammarten von UML 2

Leser, die mit UML nicht vertraut sind, finden eine Kurzeinführung unter folgender Internetadresse: [http://www.sparxsystems.com/UML\\_Tutorial.htm](http://www.sparxsystems.com/UML_Tutorial.htm)

Enterprise Architect unterstützt folgende Modelle, welche den Kern von UML erweitern: Geschäftsprozessmodelle, Mind Mapping, Requirementsspezifikation, Datenflussmodelle und weitere domänenspezifische Modelle. Die Modellierungsumgebung von Enterprise Architect unterstützt weitere Sichten, welche das Editieren von UML Modellen erleichtern und damit effizienter macht. Ein konkretes Beispiel ist die Zustandstabelle (*state table*), welche UML Zustandsdiagramme als editierbare Tabelle mit Zuständen und Zustandsübergängen darstellt.



## **Geschäftsprozessmodellierung und Simulation**

Es gibt zahlreiche Zugänge zu Geschäftsprozessmodellierung mit UML. Insbesondere sind Aktivitätsdiagramme, Objektdiagramme und spezielle Profile starke Hilfsmittel für den BPM-Analysten. Enterprise Architect ergänzt die UML 2.4 Sprachpalette mit Elementen zur Anforderungsanalyse, zum Anforderungs- und Prozessmanagement, wie z. B. Elemente für Änderungsantrag, Feature und Vorfällen.

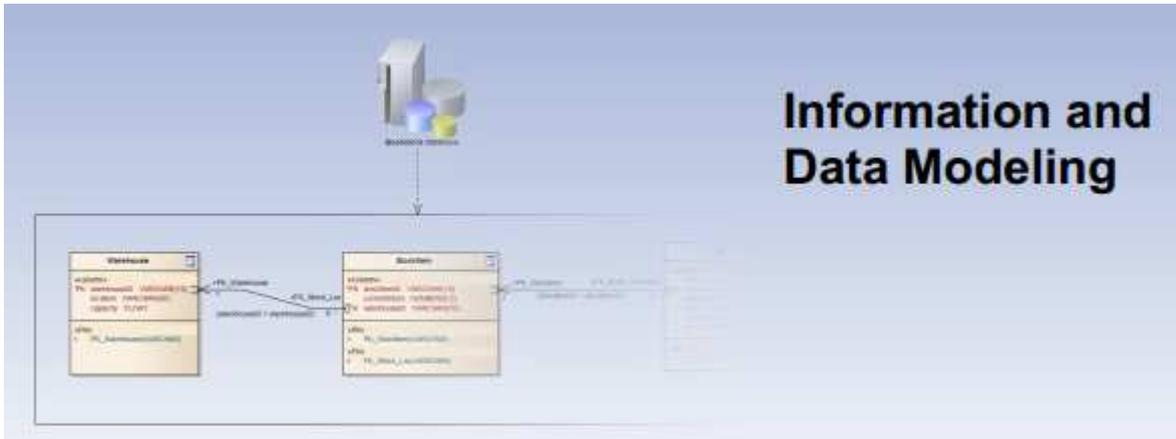
### **Das BPMN Profil**

Eine populäre Notation der Geschäftsprozessmodellierung ist die *Business Process Modeling Notation* (BPMN) (siehe auch [www.bpmi.org](http://www.bpmi.org)). BPMN ist eine eigene Sprache neben UML und ist der UML Notation und Arbeitsweise sehr ähnlich. Sparx Systems implementiert ein BPMN-Profil und stellt die letzte BPMN 2.0-Technologie als Add-In für Enterprise Architect kostenlos zur Verfügung.

BPMN 2.0 Modelle können mit der eingebauten Simulationsfunktion durchgespielt werden. Sie erlaubt das dynamische Erkunden von Prozessabläufen, Modellüberprüfung das Ermitteln von Zeitwerten und Ressourcennutzung mit der Möglichkeit zur Prozessoptimierung.

### **Die BPEL-Generierung**

Enterprise Architect unterstützt die Generierung von BPEL Skripten (Business Process Execution Language) aus BPMN Modellen heraus. So erstellte BPEL Skripte können mit Hilfen externen Werkzeugen ausgeführt werden. Spezifische Modellvalidierungsregeln helfen, korrekt ausführbaren BPEL Code generieren zu können.



## Datenmodellierung

Enterprise Architect erlaubt die Erstellung von konzeptionellen, logischen und physischen Datenmodellen in derselben Datenhaltung. Schemas von bestehenden Datenbanken können importiert werden und Datenbankskripte können aus Ihren Modellen erstellt werden. Dadurch wird eine End-zu-End-Verkettung sichergestellt, vom Konzept bis zur Umsetzung.

### Das eingebaute Datenmodellierungsprofil

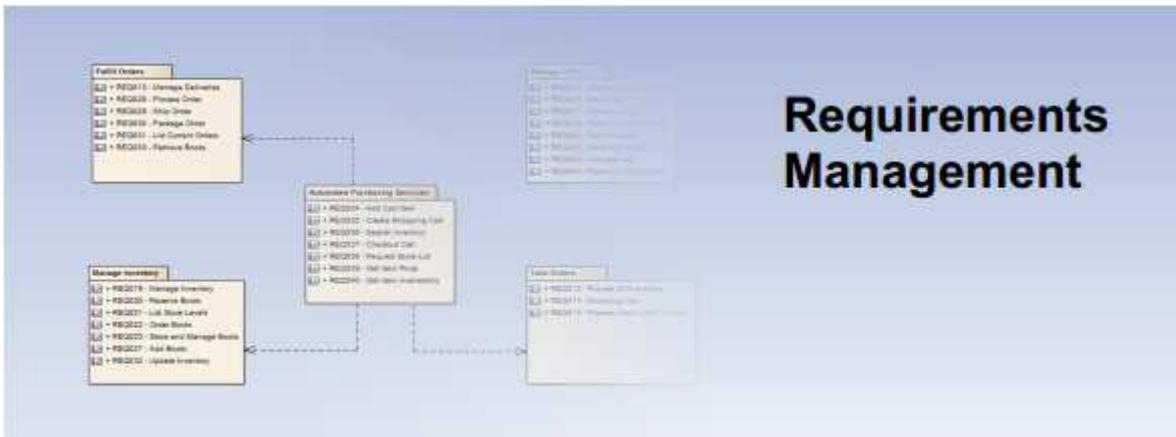
Das in Enterprise Architect eingebaute Datenmodellierungsprofil erweitert die UML um eine intuitive Zuordnung zwischen Tabellen, deren Inhalten und UML-Klassen mit ihren Relationen zu ermöglichen. Diese Erweiterung modelliert Schlüssel, Trigger, Bedingungen und andere relationale Datenbankfunktionen.

Beim Entwurf oder der Dokumentation wird Ihnen Enterprise Architect helfen, um:

- ein Datenmodell anzulegen, das Ihre Datenbankstruktur visualisiert.
- Tabellen und Views anzulegen.
- Spalten, Primär- und ausgehende Schlüssel anzulegen.
- Stored Procedures zu erzeugen.
- Indizierung, Sequenzierung, Funktionen und Trigger anzulegen.
- die DDL für eine Tabelle oder ein ganzes Package anzulegen.
- Datentypumwandlungen für eine Tabelle, ein Package oder eine ganze Datenbank auszuführen.
- ein Datenbankschema über eine ODBC Datenquelle zu erstellen.

### Unterstützte DBMS Produkte

- |                         |   |
|-------------------------|---|
| • DB2                   | • Oracle 9i, 10g und 11g                      |
| • Interbase             | • PostgreSQL                                  |
| • Informix              | • Microsoft SQL Server 7, 2000, 2005 und 2008 |
| • Ingres                | • Sybase Adaptive Server Anywhere, Enterprise |
| • Microsoft Access 2007 | • Firebird                                    |
| • MySQL                 |   |



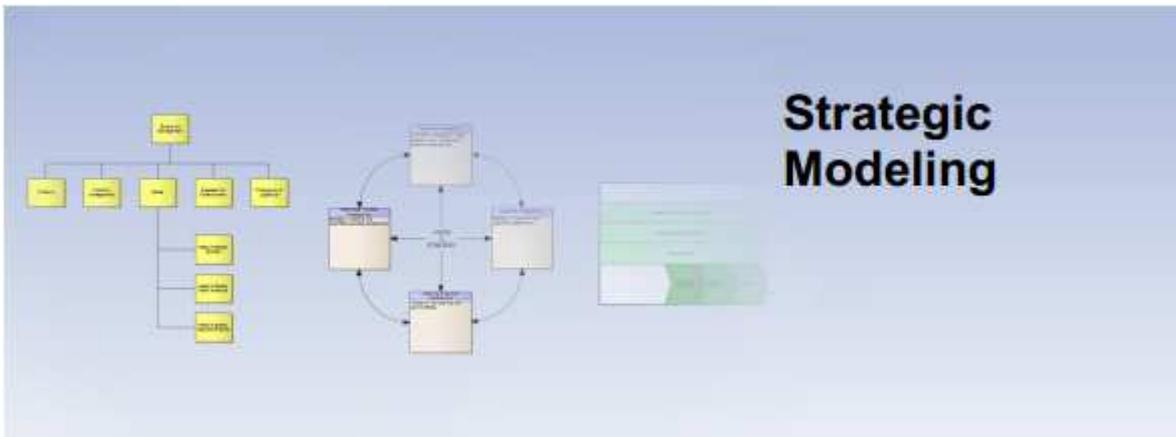
## Anforderungsmanagement

Typischerweise ist der erste Schritt der Lösungsfindung die Sammlung von Anforderungen, sei es für eine Softwareentwicklung oder eine Geschäftsprozessoptimierung. Anforderungen stellen dar, was ein System leisten soll. Die im EA eingebaute Anforderungsverwaltung kann verwendet werden zur:

- Anlage eines gegliederten und hierarchisch organisierten Anforderungsmodells.
- Saubere Unterscheidung zwischen verschiedenen Anforderungsarten, z. B. zwischen funktionalen und nicht-funktionalen Anforderungen.
- Verknüpfung der Systemanforderungen mit Modellelementen in der Analyse und in der Umsetzung und Nachverfolgung der Umsetzung über die Verknüpfungen.
- Suche nach Anforderungen und der Reporterstellung für Anforderungen als auch für eine Auswirkungsanalyse bei Anforderungsänderungen.
- Erstellung spezieller Berichte oder auch kompletter Anforderungsspezifikationen direkt aus dem Modell.
- Verknüpfung formale Anforderungen direkt zu Schritten in Szenarios (Storys) – Verknüpfung von Anforderungspunkten mit der fortschreitenden Systementwicklung.

Enterprise Architect unterscheidet sich insbesondere durch diese ausgereifte Anforderungsmanagement-Funktion von vielen anderen UML-Werkzeugen. Die Möglichkeit, Anforderungen direkt im Modell anlegen zu können, befriedigt viele Entwicklungsanforderungen – auch die Einhaltung verschiedener Qualitätssicherungsnormen – wie Nachweisketten, Arbeitsaufteilung in Teams, Änderungs- und Konfigurationsmanagement.

Enterprise Architect unterstützt die intensive Verkettung des Anforderungsmodells mit dem weiteren Umsetzungsprozess, die Einbindung externer Dokumentation als auch die Dokumentationserstellung des Architektur- und Umsetzungsmodells.



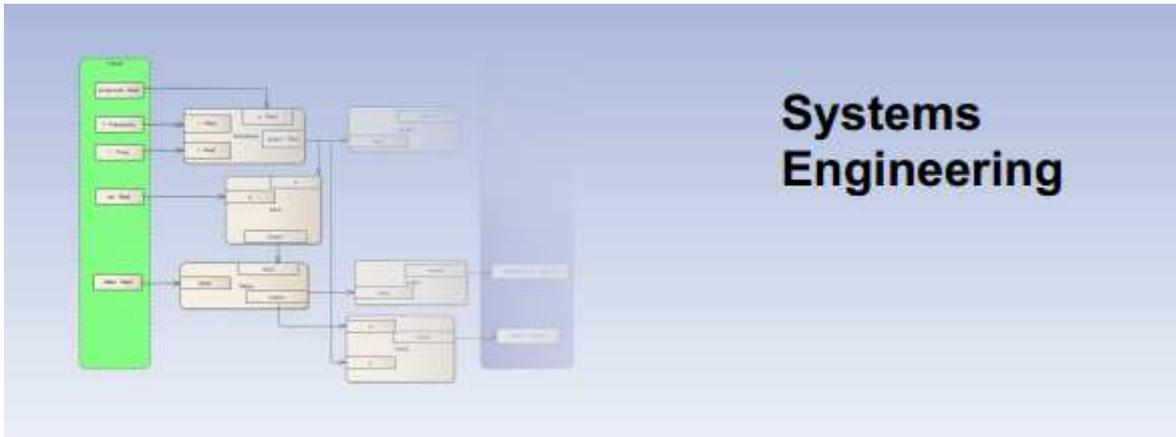
## ***Strategisches Modellieren***

Strategisches Modellieren erlaubt Organisationen die Planung Ihrer Zukunft und die Entscheidungsfindung abgestimmt auf ihre Mission und Werte. Enterprise Architect kann jeden Schritt des Planungs- und Entwurfsprozesses modellieren – von der Idee bis zur Umsetzung.

Ausgangskonzepte können mit Mindmaps dokumentiert werden, diese können auch strategische Ziele, Vorgaben und Zielstrukturen enthalten, indem folgende Diagramme eingeschlossen werden:

- Strategie-Zeichnungen
- Werteketten
- Entscheidungsbäume
- Balanced Scorecards
- Ablaufpläne
- Organigramme

Alle diese Übersichtsmodelle können direkt mit Analyseelementen und Geschäftsprozessen verkettet werden, die die Geschäftsstrategie umsetzen, um sicherzustellen, dass die Ressourcen eingebunden sind und die Prioritäten ausreichend berücksichtigt sind.



## **Systementwurf (Systems Engineering)**

Die *Ultimate* und die *System Engineering* Edition des Enterprise Architects bieten für Systemingenieure wesentliche Funktionen: SysML Modellierung, parametrische Modellsimulation, Erstellung ausführbaren Codes und Modell-zu-Code-Transformationen für hardwarebeschreibende Sprachen (HDLs) und ADA 2005.

### **SysML und Modellsimulation**

Mit Enterprise Architect können SysML-Modelle rasch und einfach mit einem integrierten Profil erstellt werden. Das SysML-Profil für Enterprise Architect unterstützt alle neun SysML 1.2 Diagrammarten, Modellvalidierung und eine Grammatikführung mit dem „Schnellzeichenpfeil“. Die Editionen *Ultimate* und *System Engineering* unterstützen auch die Simulation von *SysML Parametric Diagrams*. Dies betrifft die Analyse kritischer Systemparameter, einschließlich der Bestimmung von Kennwerten wie Performance, Verlässlichkeit und anderer physischer Kennwerte.

### **Erweitertes Verhaltensmodellieren und erweiterte Codeerzeugung**

Enterprise Architect kann die detaillierten Verhaltensbeschreibungen UML-Aktivitäten und Interaktionselementen zuordnen, was die Erzeugung ausführbaren Codes aus Sequenz- und Aktivitätsmodellen ermöglicht. Dies ermöglicht weit mehr als die Erzeugung von reinen Codehüllen, indem beliebig verschachtelte Verhaltensmodelle einschließlich Bedingungen, Schleifen und Zuordnungen umgesetzt werden.

### **Hardwarebeschreibende Sprachen**

Die Editionen *Ultimate* und *System Engineering* unterstützen die Codeerzeugung aus Zustandsdiagrammen für die hardwarebeschreibenden Sprachen (HDLs) wie VHDL, Verilog und SystemC.



## Test Management

Enterprise Architect erlaubt die Erstellung von Testspezifikationen, die Testdurchführung und das Aufzeichnen der Testergebnisse direkt auf den betroffenen Modellelementen. Die gemeinsame Abspeicherung der Modellelemente und der Testdokumentation in einem integrierten Modell kann die Kommunikation zwischen dem Qualitätssicherungsteam, den Entwicklern, den Analysten und den Architekten wesentlich verbessern.

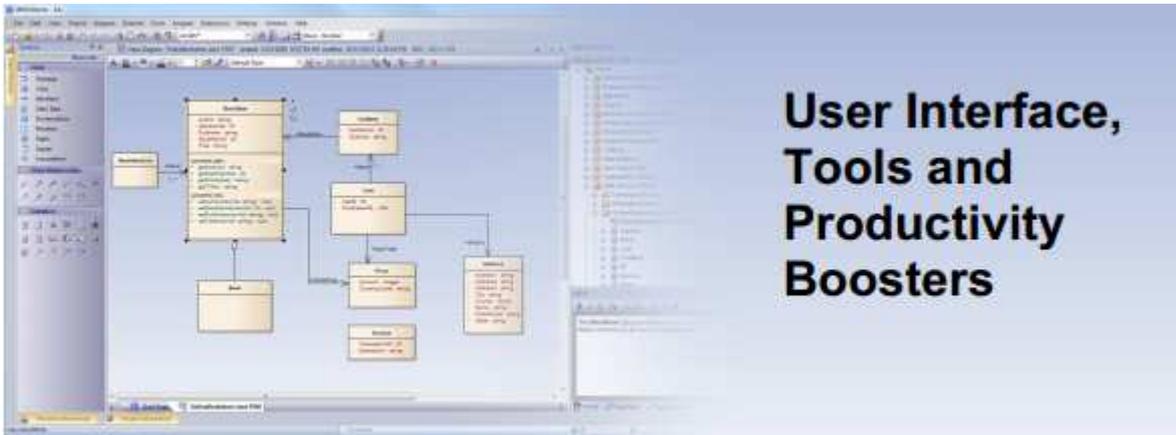
Enterprise Architect bietet Werkzeuge für verschiedene testrelevante Aufgaben wie:

- **Testpoint Management** bietet eine mächtige modellbasierte Testausführung. Sie können Testpunkte in Ihren Elementen des Systementwurfs definieren, diese in Ihrer Modellumgebung ausführen und die Ergebnisse in Echtzeit aufzeichnen, während die Anwendung ausgeführt wird – ohne den Testcode manuell zu erstellen. Testpunkte können auch zeitschonende in wiederverwendbare Test-Suiten zusammengefasst werden.
- **Testspezifikation:** Sie können zu jedem beliebigem Modellelement in Enterprise Architect Testspezifikationen hinzufügen, einschließlich der Erfassung des Testergebnisses, wann Tests von wem durchgeführt wurden. Unit-, System-, Integrations-, Abnahme- und Szenariotests können erfasst und als Bericht ausgegeben werden mit einer Verkettung zwischen Architektur und Tests, auch über mehrere Testzyklen hinweg. Testfälle können auch direkt aus Szenarios erstellt werden.
- **JUnit und NUnit:** Enterprise Architect unterstützt eine Transformation, die aus Ihren Designelementen automatisch Testelemente erzeugt. Codehülsen (Stubs) können automatisch erzeugt werden, Sie können sich auf die Definition der Testlogik konzentrieren. Sie können Ihr Unit-Testprogramm kompilieren, ausführen und die Testergebnisse erfassen, ohne den Enterprise Architect zu verlassen.

Enterprise Architect ist nicht nur eine Modellierungsumgebung, Enterprise Architect ist auch eine vollständige Testumgebung.

## In den Entwicklungszyklus integriertes Unit-Testing

Grundsätzlich gilt die Regel, Unit-Tests vorab zu schreiben, Enterprise Architect unterstützt dies. Wenn Sie eine neue Methode schreiben, führen Sie die Testfalltransformation für diese Klasse aus. Enterprise Architect erzeugt eine korrespondierende Testmethode, die Sie sofort erzeugen können und in die Sie sofort die Details eintragen können – noch bevor der zu testende Code geschrieben ist.



## **Benutzeroberfläche, Werkzeuge und Arbeitsbeschleunigung**

Die Bedienoberfläche des Enterprise Architects besteht aus zahlreichen Fenstern, Menüs und Werkzeugbalken, die Sie an Ihre Arbeitssituation anpassen können. Die wesentlichen Fenster sind:

- **Der Project-Browser** zeigt die Modellhierarchie Ihres Projektes an und erlaubt das Hinzufügen, die Auswahl und die Reorganisation von Packages, Elementen und Diagrammen.
- **Die Diagrammtoolbox** ist kontextabhängig zum gerade editierten Diagramm und stellt die für diese Diagrammart passenden Elemente und Konnektoren bereit.
- **Die Diagrammsicht**, die Ihr Modell sichtbar macht:
  - Konnektoren und Elemente können zwecks Hervorhebung in Präsentationen gestaltet und auch farblich hervorgehoben werden.
  - Ein eigener „handgezeichnet“-Modus und ein „Whiteboard“-Modus helfen den Entwurfszustand hervorzuheben und Kommentare einzufordern.
  - Mehrere Diagramme können gleichzeitig angezeigt werden, auch die einfache Verschiebung oder Umkopierung von Elementen zwischen diesen geöffneten Diagrammen ist möglich.
  - Elemente können auch in Listenform dargestellt werden, Suchergebnisse ebenso; Sie können zwanglos zwischen Visualisierungen, Listen und Codeeinsicht navigieren.
- **Kontextmenüs** bieten auf das/die betroffene/betroffenen Elemente abgestimmte Befehle.

## **Freie Diagrammpositionierung versus Verankerung**

Oft ist es zweckmäßig mehrere Ansichten gleichzeitig einzusehen - zum Vergleich, zum Eingeben in Diagramme oder zur Bearbeitung von Verkettungen im Modell. Enterprise Architect erlaubt das gleichzeitige Öffnen mehrerer Diagramme mit der Freiheit, diese beliebig am Bildschirm anzuordnen. Dies gilt auch für Suchergebnisfenster, den Editor für angehängte RTF-Dokumente, den Sourcecode-Editor, die Zustandstabelle oder die Listendarstellung von Elementen. Dies bedeutet insbesondere, dass Sie Diagramme editieren können, während gleichzeitig alle relevanten Ansichten angezeigt werden. Durch einfaches Verschieben oder Kopieren zwischen Fenstern können Sie Arbeitsplätze mit Mehrfachmonitoren voll ausnutzen. Die von Ihnen gewählten Fensteranordnungen können Sie unter einem Profilnamen verspeichern und jederzeit wieder aufrufen.

## **Direkte Dateneingabe zu Diagrammbestandteilen**

Enterprise Architect erlaubt die rasche Dateneingabe zu Diagrammelementen direkt im Diagramm. So können Sie Elementehalte wie z. B. Attribute, Methoden, Parameter rasch erfassen oder abändern, ohne die Diagrammsicht verlassen zu müssen.

## Der Schnellzeichenpfeil

Der Schnellzeichenpfeil bietet eine effiziente Nutzerführung zum Anlegen neuer Elemente und Konnektoren in einem Diagramm. Seine kontextabhängige Menüführung besichert die Erstellung syntaktisch korrekter Modelle, spart Zeit ein und erhöht die Produktivität.

## Weitere Diagrammfunktionen

- Diagramme können in zahlreichen Formaten exportiert werden (.bmp, .jpg, .png, .gif, .emf und .wmf).
- Bahnen (Swimlanes) erlauben die logische Unterteilung von Diagrammen.
- Das *Pan and Zoom*-Fenster erleichtert das Navigieren in komplexen Diagrammen.
- Die *Diagram Filters* können verwendet werden, um Ist- und Sollzustand oder Änderungen hervorzuheben.
- Sie können Diagramme mit Baseline-Versionen des Diagramms visuell vergleichen und auch umgehend einzelne Differenzen aufheben.
- Diagramme können gegen unbeabsichtigte Änderungen gesperrt werden.
- *Shape Scripts* erlauben die freie Gestaltung von stereotypbehafteten Elementen – alternative Modellnotationen sind möglich.
- *Alternative Images* (beliebige Bilder) können Elemente überladen – die Bilder werden anstelle des Standardsymbols angezeigt.

## Modell- und Anwendungsmuster (Pattern)

Bereits im Lieferumfang sind zahlreiche vordefinierte Modell- und Applikationspatterns enthalten, um die Erstellung neuer Modelle und Projekte zu vereinfachen. Jedes Pattern enthält Verwendungshinweise, Referenzen und Ausgangselemente – zusammengenommen ein Framework, auf dem Sie Ihr Modell aufbauen können. Zum Beispiel enthält Enterprise Architect Java und .Net Applikationspattern, die Ihnen ein Ausgangs-Implementationsmodell liefern.

## Modellprüfung (Validierung)

Die Modellprüfungsfunktion prüft UML und SysML Modelle hinsichtlich Normspezifikation als auch Bedingungen auf Einhaltung der *Object Constraint Language* (OCL). Die Prüfung kann auf ein einzelnes Modellelement, ein Diagramm oder ein ganzes Package angewandt werden. Dies stellt ein mächtiges Instrument speziell während des Modell-Reviews dar.



## **Dokumentenerstellung und Reports**

Die Dokumentenerstellung trägt wesentlich zur Wertschöpfung der Verwendung von Enterprise Architect bei. Der EA erzeugt hochqualitative Dokumentation, entweder im RTF-, PDF- oder im HTML-Format. Sie können auf einfache Weise einen Report für ein ganzes Projekt, ausgewählte Teile davon oder auch für eine Gruppe von Packages auch abweichend von der Gliederungsform im Project Browser erstellen, auch mittels der Modellsuche.

### **Rich-Text Format (RTF) Ausgabe**

Kompatibel mit Microsoft® Word®<sup>4</sup> oder OpenOffice erzeugt der Enterprise Architect RTF Dokumentengenerator Spezifikationsdokumente direkt aus dem Modell. Dokumente werden mittels anpassbarer Vorlagen erstellt, die sicherstellen, dass die Ausgaben Ihren Firmenstandards genügen. Sie können wahlfrei einzelne Inhalte ein- oder ausschließen, um Reports für spezielle Zielgruppen zu erstellen. Auch eine Ausgabe im .pdf-Format ist möglich.

### **Der HTML Generator**

Enterprise Architect kann ein ganzes Modell oder auch einen einzelnen Zweig des Modells als HTML-Seiten ausgeben, um z. B. eine Sichtung im Firmenintranet zu ermöglichen. Der Modellinhalt wird detailliert und einfach navigierbar dargestellt. Hyperlinks in der HTML-Präsentation erlauben das Navigieren zwischen zusammenhängenden Modellinhalten. Auch die HTML-Generierung erfolgt templategesteuert, sodass Sie auch diese Darstellung an Ihre Firmenstandards anpassen können.

### **Der eingebaute Dokumenteneditor und das Verlinken von Dokumenten**

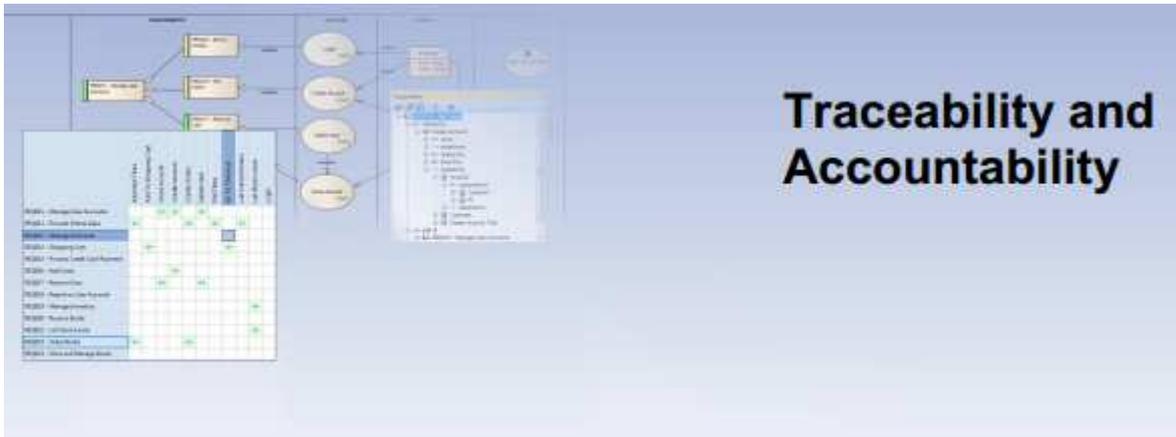
Enterprise Architect ermöglicht die Verlinkung und Abspeicherung von Rich-Text-Dokumenten zu jedem einzelnen Modellelement. Verlinkte Dokumente (*linked documents*) können Vorlagen verwenden und können wahlweise bei der Generierung von HTML und RTF Reports automatisch mit ausgegeben werden.

### **Strukturierte Szenarios/Storyboards**

Szenarios sammeln Anforderungsinformation in natürlicher Sprache. Enterprise Architect erlaubt Ihnen, diese Inhalte elegant zu verwenden - um die Verknüpfung in den Entwicklungszyklus hinein optimieren zu können. Sie können Szenarioschritte mit Domänelementen, Geschäftsregeln und Fachbegriffen verlinken. Aus strukturierten Szenarios können Sie automatisch Testfälle, Aktivitätsdiagramme und andere Verhaltensdiagramme erstellen lassen. Sie können auch bestehende Prozessdiagramme in Gegenrichtung in strukturierte Szenarios umwandeln.

---

<sup>4</sup> Microsoft Word™ ist eine Handelsmarke der Microsoft Corporation



## **Verkettung und Nachweisfähigkeit**

### **Protokollierung (Auditing)**

Die Protokollfunktion von Enterprise Architect zeichnet die Modellveränderungen chronologisch auf. Administratoren können über das Protokoll ermitteln:

- Wer hat die Änderung gemacht?
- Wie viele Elemente wurden verändert?
- Wann war die Änderung?
- Was war der Inhalt vor und nach der Änderung?
- Welche Art von Element wurde geändert?

Die Protokollfunktion kann parametrisiert werden, auf einzelne Elementarten beschränkt werden. Die Abfrage kann gefiltert werden nach Benutzer und auch nach Zeitraum.

### **Abweichungs- und Abhängigkeitsanalyse**

#### **Abweichungs-Analyse-Matrix**

Die in Enterprise Architect enthaltene *Gap Analysis Matrix* hilft beim Modellieren der Unterschiede zwischen einer Ist- und einer Sollarchitektur. Diese Matrix bietet eine effiziente Darstellung zum Modellieren und Überwachen von erkannten Abweichungselementen, die auch mit anderen Elementen des Modells verknüpft werden können.

#### **Abhängigkeits-Analyse-Matrix**

Die *Relationshipmatrix* erlaubt eine Ansicht der Beziehungen zwischen Modellelementen in Tabellenform. Sie kann benutzt werden um Verkettungsbrüche zu identifizieren, Verkettungen anzulegen, Verkettungen zu modifizieren oder auch zu löschen.

#### **Das Verkettungsfenster – Traceability View**

Die Traceability View zeigt eine dynamische, navigierbare Ansicht der Beziehungen des gerade selektierten Elements zu anderen Modellelementen. Die angezeigten Relationen umfassen auch die Arten Aggregation, Vererbung und Dependency und eingebettete Elemente. Durch Selektion der Verbindungen zwischen unterschiedlichen Abstraktionssektionen des Modells ist dies ein mächtiges Instrument um die Auswirkung von Anforderungsänderungen auf andere Modellelemente nachzuvollziehen.

## **Suche nach Elementen und Meta-Daten**

### **Die Modellsuche**

Die flexible Suchfunktion des Enterprise Architects sucht Ihnen Elemente im Modell anhand eines frei definierbaren Suchkriteriums. Die im Suchergebnis aufgelisteten Elemente können zum Drucken, zum Reporterstellen, zur Hinzufügung in eine Dokumentation oder auch zur Hinzufügung in die Gruppendiskussion ausgewählt werden.

### **Die Modellsichten (Model Views)**

Das *Model Views* Fenster bietet eine dynamische, gefilterte Ansicht von Elementen aus der Modellstruktur. Sie können Elemente mittels eines Suchkriteriums auswählen, Elemente und Diagramme zu Favoriten erklären aber auch Elemente nach einem speziellen Blickwinkel sortieren, z. B: nach Zugehörigkeit zu einem Framework. Diese Sichten können mit ihrem Suchkriterium sowohl lokal für den Benutzer aber auch in der gemeinsamen Datenhaltung abgelegt werden, sodass sie allen Nutzern zur Verfügung stehen. Eine automatische Verständigung kann eingerichtet werden, um Sie zu informieren, wenn ein neues Element im Suchergebnis erscheint.

### **Die Elemente-Ansicht (Element Browser)**

Der *Element Browser* bietet eine kontextabhängige Ansicht der Metadaten zum selektierten Element. Diese Ansicht ermöglicht den einfachen Einblick in Tests, Projektmanagement, strukturierte Szenarios, in Wartungseinträge und in die Modellsemantik. Der *Element Browser* wird so zum Ausgangspunkt für Reviews und damit verknüpfter, weiterer Aktionen.

### **Der Package Browser**

Der *Package Browser* ist eine editierbare Tabellenansicht von Elementen, was den Prozess des Anlegens und Bearbeitens von Elementen in einem Package vereinfacht. Dies kann insbesondere hilfreich sein, um formale Anforderungen in einem Modell zu editieren. Die Liste kann auch gedruckt werden oder einer RTF-Reporterstellung zugeführt werden.

### **Elementeverwendung nachvollziehen**

Enterprise Architect macht es Ihnen einfach, die Verwendung eines Elements in verschiedenen Diagrammen nachzuvollziehen. Die *Find*-Funktion des Enterprise Architects in Diagrammen, im *Project Browser* und im *Package Browser* erlaubt das einfache Auffinden der Verwendungsstellen und eine direkte Navigation zu den entsprechenden Diagrammen.

### **Automatische Diagrammerstellung für verkettete Elemente**

Enterprise Architect kann automatisch die mit dem aktuellen, selektierten Element verketteten Elemente in ein Diagramm einzeichnen. Dabei kann nach Elementart, Beziehungsart, Beziehungsrichtung und Schrittweite gefiltert werden. Dies kann effizient zur Darstellung spezieller Beziehungsübersichten hinsichtlich Ihres Frameworks oder zum reverse engineerten Code dienen.

## **Weitere Reports**

Enterprise Architect stellt weitere, unmittelbar anwendbare Reports zur Verfügung: *Resource and Task Details* (Arbeitszuteilung), *Project Issues* (Projektvorfälle), *Project Glossary* (Projektwörterbuch), *Project (size) Statistics*, *Dependency* (Abhängigkeitsliste) und *Implementation Details* (Überprüfung der Mindestanforderungen an die Umsetzungskette) und *Testing Details*.



## **Entwicklung und Zusammenarbeit im Team**

Enterprise Architect bietet Funktionalitäten, um Projekte gemeinsam und zeitparallel im Team und in verteilten Entwicklungsumgebungen abzuwickeln. Projekte können gemeinsam über Netzwerk, mit Modell-Repositories, Replikation, XMI Import/Export, Versionierungssystemen, Packageversionierung und nutzer-/gruppenbezogenen Sicherheits-/Sperrmechanismen betrieben werden.

## **Unterstützung großer Modelle oder vieler gleichzeitiger Anwender**

Die *Corporate* Edition und höhere Editionen von Enterprise Architect erlaubt auch das Speichern und Bearbeiten von gemeinsam genutzten Modellen mittels eines DBMSs (Datenbanksystems), anstelle des standardmäßigen .EAP-Files. Enterprise Architect unterstützt folgende Datenbanksysteme (DBMS):

- Access 2007
- MS SQL Server
- MySQL
- Oracle
- PostgreSQL
- Progress OpenEdge
- MSDE Server
- Adaptive Server Anywhere

## **Security - Sperrfunktion**

Mittels einer Rollendefinition können im Enterprise Architect einzelnen Nutzern verschiedene Bedienungsberechtigungen zugewiesen werden, die über eine Anmeldung am Modell besichert werden. Dies erlaubt auch eine gruppenweise oder nutzerspezifische Sperrung von Elementen, Diagrammen oder Packages. Dies verbessert die Zusammenarbeit indem verhindert wird, dass verschiedene Nutzer unabsichtlich gleichzeitig die gleichen Inhalte editieren. Ebenso können Nutzer von der Bearbeitung von Modellteilen ausgeschlossen werden, für die sie als Autor nicht zuständig sind. Die Sperrfunktion kann auch zum Einfrieren von Modellteilen verwendet werden, die bereits freigegeben sind und nicht mehr ohne Zustimmung des Projektleiters verändert werden dürfen. Die Sperrfunktion funktioniert auf .eap- und Datenbankrepositories gleichartig.

**Dies ist im Vergleich zu den anderen verfügbaren Mechanismen (Paketkontrolle) die weitaus agilere Verwendungsweise.**

## **XML Import und Export**

Enterprise Architect unterstützt das XML-basierte Model Interchange Format, bekannt unter dem Titele XML Metadata Interchange (XMI). Diese XMI-Funktion des Enterprise Architects kann benutzt werden, um Modellinhalte zwischen Entwicklern auszutauschen. XMI gestattet den Export einzelner Packages oder ganzer Modellzweige in XML Dateien, die in andere Modelle importiert werden können oder in einem Versionierungssystem verwaltet werden können. **Werden Packages direkt unter Kontrolle des Versionierungssystems gestellt, bedeutet dies allerdings, dass**

**einzelne Packages gleichzeitig nur von einem Nutzer bearbeitet werden können!**

**Die Team Review Funktionalität (internes Diskussionsforum)**

Diese Funktion des Enterprise Architects hilft den Anwendern, die Entwicklung und den Projektfortschritt zu diskutieren. Mitarbeiter können Nachrichten in der Modellumgebung absetzen und einsehen und die Nachrichten direkt mit Modellelementen verlinken. Bei verteilten Arbeitsumgebungen können die Nutzer ihr Modell mit einem Diskussionsforum, das in einer anderen, abgesetzten Datenhaltung liegt, verbinden.



## Project Management

### **Projektmanagement**

Enterprise Architect bietet auch Unterstützung, um Ihr Projekt zu managen. Projektmanager können in Enterprise Architect Ressourcen einzelnen Elementen zuordnen, Risiken verwalten, Aufwände verwalten und automatisch berechnen.

### **Aufwandsbestimmung mit der Use Case-Point-Methode**

Die in Enterprise Architect eingebaute *Use Case Metrics* Funktion macht es sehr einfach, die relative Komplexität eines Projekts zu ermitteln. Basierend auf der Anzahl und Art der Use Cases eines Modells, der Art des Entwicklungsprojekts und basierend auf Faktoren, die sich auf die Entwicklungsumgebung beziehen, kann mit dieser Methode der erforderliche Projektaufwand rasch und einfach ermittelt werden.

### **Ressourcen**

Enterprise Architect kann Ressourcen gemeinsam mit den Modelldaten verwalten. Dies erspart dem Entwicklungsteam zusätzliches Reporting an den Projektmanager, der Projektmanager kann jederzeit nachvollziehen, wie weit das Projekt vorangekommen ist. Engpassermittlung, Zeitverzug und andere kritische Vorfälle können direkt an die Modellelemente angebunden werden und über die Modellsuche einfach ermittelt werden.

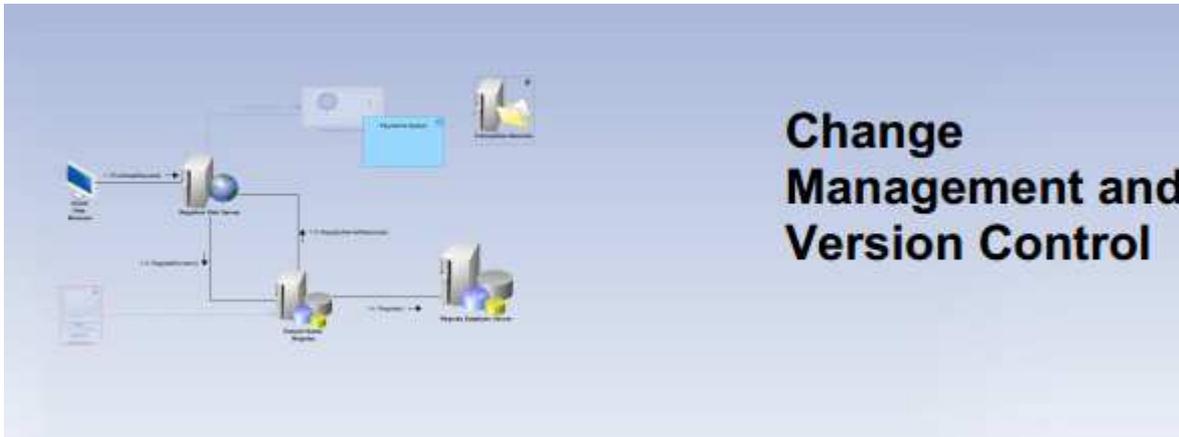
### **Aufgaben, Gantt-Darstellung und Projektkalender**

Überwachung, Priorisierung und Arbeitszuteilung sind die kritischen Punkte während des kontinuierlichen Projektmanagements. Enterprise Architect kann individuelle, ressourcenzugeordnete Aufgaben als auch Aufgaben, die dem gesamten Projekt zugeordnet werden, verwalten. Arbeitszuteilungen können als Gantt-Diagramm visualisierend dargestellt werden. Der eingebaute Projektkalender erlaubt das Anlegen und Nachverfolgen wichtiger Zeitpunkte, Meilensteine und Besprechungen direkt in der Modellumgebung.

Jeder Modellmitarbeiter hat Zugriff auf eine personalisierte Projektansicht mit dem *Personal Information* Fenster, das dem Nutzer die Eingabe von Fortschrittsberichten erlaubt, das Versenden und Empfangen von Modellnachrichten als auch das Einrichten von Diagrammsammlungen und anderer Ansichten speziell zur eigenen Rolle.

### **Projektglossar**

Enterprise Architect fügt jedem Modell ein Glossar hinzu, das die eindeutige Begriffsbelegung sicherstellt, speziell sinnvoll für neu hinzukommende Mitarbeiter oder Mitarbeiter, die mit der Begriffswelt der Aufgabenstellung nicht vertraut sind.



## Änderungsmanagement und Versionskontrolle

Enterprise Architect unterstützt die Versionskontrolle von Packages und ihren Sub-Packages. Packageversionen können in der Enterprise Architect Datenhaltung (Repository) selbst gespeichert werden oder sie können in einem angeschlossenen Versionierungssystem eines Fremdherstellers verwaltet werden.

Eine Versionskontrolle in auf ein Enterprise Architect Modell anzuwenden hat zwei Vorteile:

- Koordination der gemeinsamen Packagenutzung zwischen verschiedenen Nutzern
- Eine Historisierung der Veränderungen in Packages mit der Möglichkeit, Vorgängerversionen wiederherstellen zu können

Enterprise Architect unterstützt *Subversion*, *CVS* und den *Microsoft Team Foundation Server (TFS)* als anschließbare Versionsverwaltungssysteme als auch alle Versionsverwaltungssysteme, die mit dem *Microsoft Common Source Code Control (SCC)* Standard, Version 1.1 oder höher, kompatibel sind. Zum Beispiel *Visual Source Safe* oder *Clear Case*.

**Hinweis: Arbeiten mit einem angeschlossenen Versionsverwaltungssystem erlaubt – verglichen mit den internen Mechanismen (Security und Baselineing) - nur ein weitaus weniger agiles Arbeiten, weil immer nur ein Nutzer in einem Package zur gleichen Zeit editieren kann!**

### Baseline, Vergleich und Mischen (Merge)

Ab der Enterprise Architect Edition *Corporate* wird die Baseline-Funktion (Snapshot) für Packages unterstützt, die Sie zu einem Zeitpunkt Ihrer Wahl anwenden können. Erstellte Baselines können mit der Vergleichsfunktion des Enterprise Architects (Diff) um Differenzen zwischen dem aktuellen Modell und dem Baseline-Inhalt bei Elementen und Diagrammen zu ermitteln. Elementedifferenzen werden listenartig dargestellt, Diagrammanordnungsdifferenzen werden direkt im Diagramm visualisiert. Einzelne Differenzen können aus der Baseline wieder ins Modelle eingemischt werden – eine selektive oder wahlweise generische „Rückrollmöglichkeit“. Die Baseline-Erstellung kann auch offline in einer abgesetzten, kopierten Datenhaltung geschehen und später in die gemeinsame Datenhaltung übernommen werden.

Eine mächtige Diagrammvergleichsfunktion erlaubt den visuellen Diagrammvergleich zwischen verschiedenen Versionsständen. Eine farbcodierte Darstellung hebt unterschiedliche Elemente und Konnektoren hervor und erleichtert Ihnen zu erkennen, was einem Diagramm hinzugefügt, aus ihm herausgelöscht oder auch nur verschoben wurde. Sie können auch direkt in dieser Ansicht verfügen, welche Elemente in den ursprünglichen Zustand zurückzusetzen sind, wenn dies notwendig ist.

Die Vergleichsfunktion von Enterprise Architect kann sowohl die Differenzen zu einer Baseline aufzeigen und wiederherstellen, Sie können sie aber auch verwenden, um einen Packageinhalt zu vergleichen mit:

- Einer Datei auf einem Massenspeicher, die durch einen Baseline-Export oder durch eine

XMI-Erstellung (Achtung Sie auf das richtige Format!) für das betreffende Package entstanden ist.

- Einer XMI-Datei des betreffenden Packages, das unter der Kontrolle eines Versionsmanagementsystems steht.
- Jeder Baseline dieses Packages in einer Datenhaltung, zu der Sie Zugriff haben.



## Enterprise Architecture Frameworks

Sparx Systems unterstützt architektonische Frameworks zur Enterprise-Modellierung nach Industriestandard. Die Frameworkimplementierung in Enterprise Architect basiert auf UML und den zugehörigen Spezifikationen, wodurch architektonische Strenge gestärkt wird und den Austausch von Modellinhalten über genormte Formate wie XMI ermöglicht. Folgende architektonischen Frameworks sind als Plug-ins für den Enterprise Architect verfügbar:

- Das Zachman Framework (<http://www.sparxsystems.com/zachman>)
- DoDAF (<http://www.sparxsystems.com/dodaf-modaf>)
- MODAF (<http://www.sparxsystems.com/dodaf-modaf>)
- UPDM (<http://www.sparxsystems.com/updm>)
- TOGAF der Open Group (<http://www.sparxsystems.com/togaf>)  
... einschließlich des *Federal Architecture Framework (FEAF) Referenzmodells*

Als Ergänzung zu den architektonischen Frameworks enthält der Enterprise Architect die *Archimate 2.0 Enterprise Notation* der Open Group.

## Service Oriented Architecture (SOA)

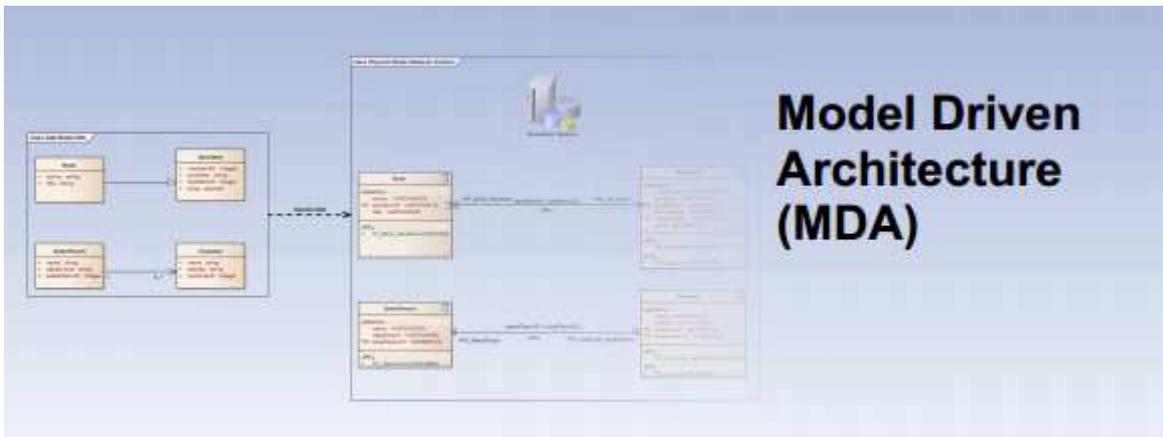
Enterprise Architect implementiert die *Service Oriented Modeling Language* der OMG (SoaML), die einen Standard zur Modellierung von serviceorientierten Lösungen in UML darstellt.

Als vorgesehener Bestandteil der SoaML unterstützt der Enterprise Architect auch das *Service Oriented Modeling Framework (SOMF)*, das eine gesamtheitliche Sicht auf Enterprise-Software-Systemteile in einer technologieunabhängigen Notation erlaubt. Die SOMF Modellnotation erlaubt einen intuitive Zugang zur Darstellung von Zielvorgaben, aktuellen Ständen und Übergängen auf dem Gebiet von Enterprise-Diensten.

## XSD und WSDL Unterstützung

Enterprise Architect erlaubt das Modellieren, die Vorwärts- und Rückwärtsübersetzung von W3C-Technologien: XML Schema (XSD) und der Web Service Definition Language (WSDL).

XSD- und WSDL-Unterstützung ist zur Entwicklung einer kompletten Service Orientierten Architektur. Die Verknüpfung von UML mit XML bietet einen intuitiven Mechanismus zur Festlegung, Konstruktion und Umsetzung XML-basierter SOA Bestandteile in einer Organisation. Die XSD- und WSDL-Funktionen des Enterprise Architects unterstützen auch die BPEL-Erzeugung aus UML Modellen.



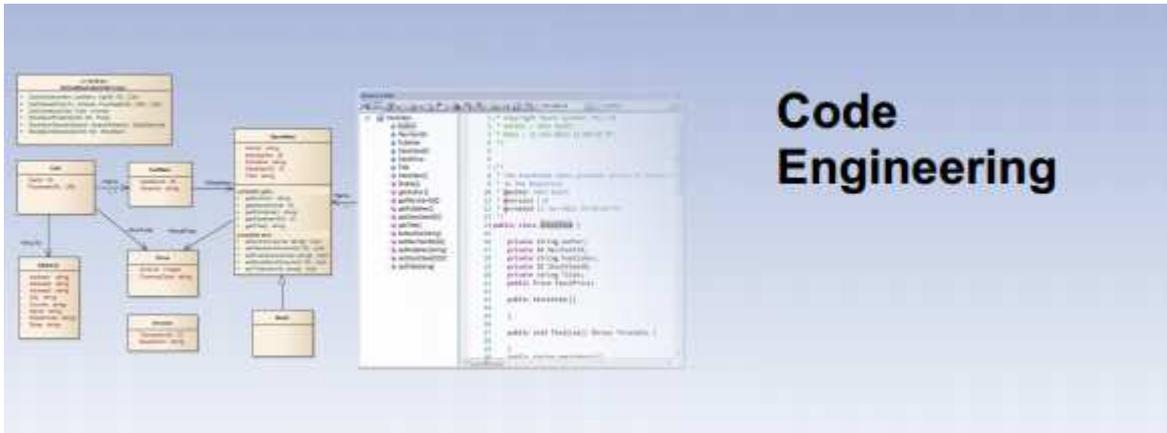
## **Modellgetriebene Architektur (MDA)**

Enterprise Architect bietet Funktionen zur Durchführung von MDA-Transformationen. Eine voll konfigurierbare Konvertierungsmethode zum Umwandeln von Modellteilen aus einer Domäne in eine andere wurde eingebaut. Dies umfasst auch die typische Umwandlung von plattformunabhängigen Modellen (PIMs) in plattformspezifische (PSMs). Ein einzelnes Modellelement des PIMs kann dabei für die Erstellung mehrerer PSM-Elemente in verschiedenen Domänen verantwortlich sein.

Transformationen sind wesentlich produktivitätsverbessernd und reduzieren den manuelle Erstellungsaufwand Klassen und Elementen in einzelnen Implementierungsdomänen. Zum Beispiel können Datenbanktabellen automatisch von PIM-Persistenz-Klassen abgeleitet werden.

Die MDA-Funktionen des Enterprise Architects erlauben:

- Vorbereitete Transformationen liefern automatisch:
  - Datenmodelle (DDL)
  - Code Modelle, einschließlich C# und Java
  - XML-Modelle, wie XSD und WSDL
  - Test-Modelle für JUnit und NUnit
  - Aktivitätsdiagramme und Testskripte aus strukturierten UseCase Scenarios (=Stories)
- das Anlegen neuer, weiterer Transformationen mit einem leistungsfähigen Templateeditor
- wiederholte Transformationsausführung, um die Konsistenz zwischen Ausgangs- und Zielmodellen sicherzustellen, auch wenn sie sich während der Entwicklungszeit verändern.



## Codebearbeitung

Die Codebearbeitungsfunktionen umfassen automatische Codeerstellung, Rückumwandlung bestehender Codes und die fortlaufende Synchronisation zwischen Code und Modell. Enterprise Architect unterstützt unmittelbar die Codeverarbeitung von mehr als 10 Programmiersprachen!

- ActionScript (Macromedia Flash Entwicklungssprache)
- C
- C# (sowohl .NET 1.1 als auch .NET 2.0)
- C++ (Standard und .NET bedingte C++ Erweiterungen)
- Delphi
- Java (einschließlich Aspects und Generics)
- PHP
- Python
- Visual Basic
- Visual Basic .NET

Ada, System-C, Verilog und VHDL stehen in den höherwertigen Editionen zur Verfügung. Auch eine Erweiterung für Perl ist von einem Drittanbieter verfügbar.

## Import von .jar-Dateien und .NET Assemblies

Zusätzlich zum Import von Quellendateien kann der Enterprise Architect folgende Binärmodule rückwärtsübersetzen:

- Java Archive Dateien (.jar)
- .Net PE Dateien (.exe, .dll)
- Zwischen-Übersetzungsdateien (.il)

## Gestaltbare Codeerzeugung

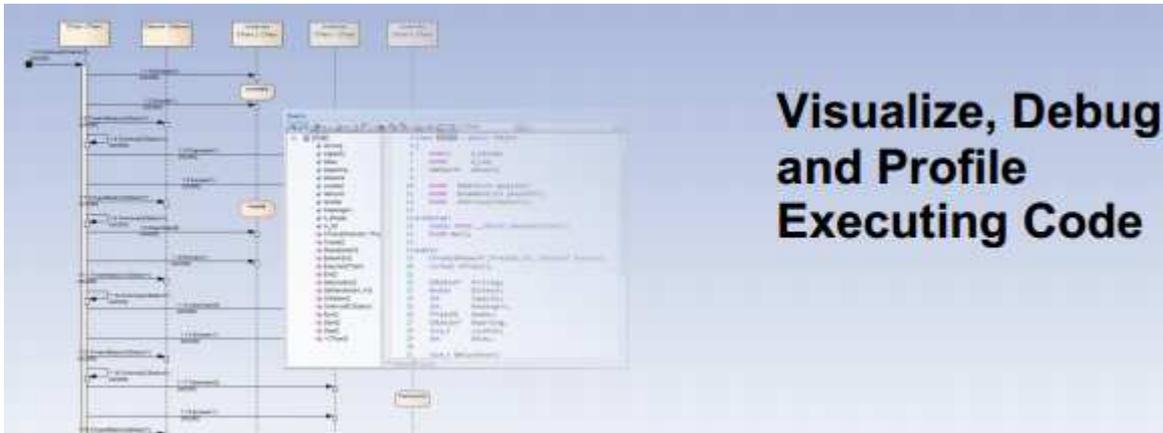
Das Code Template (Vorlagen-) Framework des Enterprise Architects bietet einen leistungsstarken, flexiblen Vorwärtsübersetzungsmechanismus um UML-Modellinhalte in Sourcecode zu übersetzen. Die Codetemplates spezifizieren anpassbare Transformationsregeln für Transformation von Modellelementen und ihrer Inhalte in Sourcecode. Dies bedeutet auch, dass Sie die Umsetzung an Ihre Standards anpassen können. Sie können die Codevorlagen auch verwenden, um in weitere Sprachen neben den bereits implementierten zu übersetzen.

## Bedarfsgesteuerte oder fortwährende Codeerzeugung

Enterprise Architect bietet auch eine automatische Codeerstellungsfunktion, die bei jeder Modelländerung den Sourcecode automatisch auf den aktuellen Stand bringt. Wenn Sie zum Beispiel neue Methoden oder Attribute im Modell anlegen, werden diese automatisch in den Sourcecode eingetragen.

## **Syntaxhervorhebung und dynamische Einrückung**

Sie können den eingebauten Codeeditor verwenden, um Sourcecode-dateien einzusehen oder zu bearbeiten. Wenn Sie ein Element im Modell selektieren, zeigt der Editor die korrespondierende Stelle mit Syntaxhervorhebung und navigierbarer Einrückung. Die Steuerleiste im Editor erlaubt den Generierungs- oder Synchronisationsaufruf.



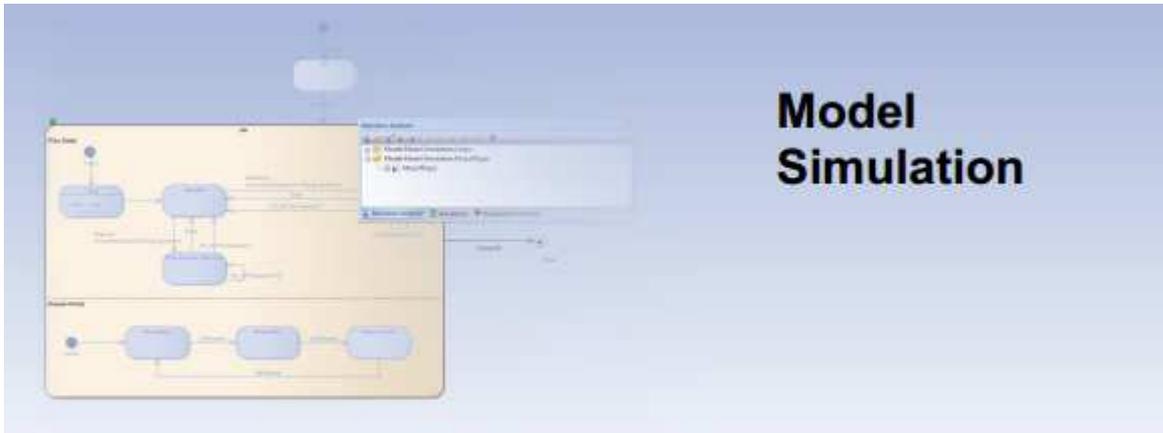
## ***Darstellung, Debugging und Profilierung von ausführbarem Code***

Der *Visual Execution Analyzer* des Enterprise Architects stellt Funktionen bereit um Anwendungen zu modellieren, zu debuggen, zu profilieren und direkt aus der Modellumgebung zu managen. Die Ausgaben des *Visual Execution Analyzers* erleichtern den Entwicklungsprozess:

- Besseres Verständnis, wie Ihr System arbeitet
- Automatische Dokumentation der Systemfunktionen
- Ausforschung von Auslösern, die zu einem unerwarteten Systemverhalten führen

Die Analysewerkzeuge des *Visual Execution Analyzers* können verwendet werden, um:

- Sequenzdiagramme zu erstellen, die die Ausführung einer Applikation oder das Verhalten einzelner Call Stacks beschreiben.
- Zustandswechsel zu protokollieren
- dynamisch Objektdiagramme während eines Debuggings zu erstellen. Wahlweise können Objekte mit ihren Laufzeitwerten und ihren Zusammenhängen zu anderen Objekten eingefügt werden.
- Profilreports zu erstellen, die Anwendungssequenzen und Aufrufhäufigkeit zeigen.
- Systemressourcen zu optimieren und die Ressourcenzuweisung zu erkunden.
- sicherzustellen, dass sich das System entwurfsentsprechend verhält.
- hochqualitative Dokumentation zu erstellen, die das Systemverhalten korrekt wiedergibt.
- zu verstehen, wie und warum das System und der Code funktionieren.
- neue Mitarbeiter in die Struktur und Funktion eines Systems einzuführen.
- unnötige Funktionsaufrufe zu erkennen.
- Interaktionen, Datenstrukturen und wesentliche Zusammenhänge innerhalb eines Systems zu illustrieren.
- Probleme auf eine Codestelle, Systeminteraktion oder einen Auslöser zurückzuführen.
- Hervorzuheben, warum eine Abfolge von Auslösern wichtig ist.
- die Auslöserfolge zu ermitteln, die unmittelbar vor einer Systemstörung auftreten.



## ***Modellsimulation***

Die Simulationsfunktionen des Enterprise Architects bringen Ihre Verhaltensmodelle in Echtzeit in Leben. Die Simulation von Modellen bringt zahlreiche Vorteile:

- Besseres Verständnis, wie ein Modell zur Laufzeit funktioniert.
- Überprüfung, ob Ihre Verhaltensmodelle den Prozessverlauf richtig repräsentieren.
- Identifizieren von Engpässen, ineffizienten Prozessteilen oder anderer Probleme in Geschäftsprozessen.
- Frühzeitige Entdeckung von Fehlern im Entwicklungsprozess, noch vor der Umsetzung.

Die Modellsimulation kann auf vier Arten von Verhaltensmodellen im Enterprise Architect angewandt werden:

- UML Aktivitäten,
- UML Interaktionen,
- UML Zustandsmaschinen, auch auf solche, die als Zustandstabelle formuliert wurden und
- Geschäftsprozessmodelle, die die BPMN-Schreibweise benutzen.

Sie können die Geschwindigkeit und den Verzweigungsverlauf der Simulation im Modell steuern: Entweder geben Sie die Verzweigungsentscheidungen manuell ein, oder Sie verwenden ein von Ihnen erstelltes Skript, in dem Sie festlegen, wann welcher Trigger feuert. Im letzteren Fall können Sie verschiedene Simulation automatisch anwenden, um zu erkunden, wie sich das System unter verschiedenen Bedingungen verhält. Durch die Möglichkeit, beliebige Unterbrechungspunkte zu setzen, wird die Simulationsfunktion des Enterprise Architects zu einem mächtigen Werkzeug zur Analyse der Entscheidungsregeln und erlaubt die Verbesserung von Geschäftsprozessen oder Systemverhalten in einer risikofreien Umgebung.



## **Enterprise Architect erweitern**

Enterprise Architect stellt zahlreiche Funktionen bereit, die es erlauben den Funktionsumfang de über den Auslieferungsumfang hinaus selbst zu erweitern. Anwender können die Modellnotationen für spezifische Domänen erweitern, können kundenspezifische Modellressourcen wie zum Beispiel Pattern hinzufügen oder auch gänzliche neue Funktionen, die auf das Modell über das *Automation Interface* oder als Add-In einwirken können.

## **Domänenspezifisches Modellieren**

### **UML Profile**

Die Patternunterstützung von Enterprise Architect ist ein exzellentes Mittel zur Wiederverwendung und zur Erzielung von Einheitlichkeit. Pattern repräsentieren eine zusammengehörige Sammlung von Objekten und Klassen, die aus einer Menge von Modellszenarios abstrahiert werden kann. Wird der bedarf nach einem Pattern in einem neuen Projekt erkannt, können die zuvor abgelegten Pattern wiederverwendet werden, natürlich auch mit zum neuen Projekt passenden Umbenennungen.

Pattern beschreiben, wie ein abstraktes Problem gelöst werden kann, es bleibt dem Pattern-Anwender überlassen, die Patternelemente so zu modifizieren, dass sie den Bedürfnissen im aktuellen Projekt entsprechen.

### **MDG Technologien**

MDG Technologien kapseln eine logische Sammlung an Ressourcen (wie z. B. UML Pattern und Profile), die zu einer speziellen Technologie oder Modelldomäne gehören. Dies sind zum Enterprise Architect hinzufügbare Ressourcen, die in einem physischen Verzeichnis oder in einer URL angelegt werden.

Anwender können ihre eigenen MDG Technologien erstellen oder Technologien verwenden, die in den Enterprise Architect Installationsmechanismus eingefügt wurden, wie z. B. Archimate and Mindmapping. Frei hinzufügbare Technologien wie zum Beispiel die CRBA Codeerstellung und die BPMN-Modell-Prüffunktion sind unter [http://www.sparxsystems.com/resources/mdg\\_tech](http://www.sparxsystems.com/resources/mdg_tech) erhältlich. Zusätzlich sind lizenzpflichtige Lösungen für Systems Modeling (SysML), Verteilte Datendienste (DDS) und andere auch erhältlich:

<http://www.sparxsystems.com/products/#technology>

### **Automation Interface**

Das Automation Interface ermöglicht Ihnen den programmatischen zugriff auf Enterprise Architect Modelle. Mit dem Automation Interface können Sie zum Beispiel:

- Wiederholte Aufgaben ausführen, z. B. die Versionsnummer aller Elemente eines Modellteils erhöhen.
- Code aus Zustandsdiagrammen nach eigener Logik erstellen.
- Spezielle Reports erstellen.
- Aktuell benötigte Abfragen gegen das Modell stellen.

Alle Entwicklungsumgebungen, die imstande sind ActiveX Com Clients erzeugen können, sollten imstande sein, sich mit dem Automation Interface zu verbinden.

## **Add-Ins**

Add-Ins ermöglichen Ihnen, weitere Funktionalität dem Enterprise Architect hinzuzufügen und seine Benutzeroberfläche zu erweitern. Das Add-In Framework des Enterprise Architects baut auf dem Automation Interface auf, hat jedoch einige wesentliche Vorteile gegenüber eigenständigen, außen liegenden, zugreifenden Anwendungen:

- Add-Ins können Menüs und Submenüs im Enterprise Architect definieren.
- Add-Ins erhalten Verständigungen über die verschiedenen Bedienungsschritte and der Enterprise Architect einschließ Menüauswahl und Dateiänderungen.
- Add-Ins können und sollten als DLL geschrieben werden; das bedeutet weniger Aufrufaufwand und bessere Integration in die Modellumgebung.
- Es braucht keine weitere Instanz des Enterprise Architect – wie im Falle der Nutzung des Automation Interfaces durch ein außen liegendes Programm - gestartet zu werden.
- Weil das Add-In Objektbezüge assoziiert mit der gegenwärtig laufenden Instanz erhält, steht über die aktuelle Anwenderaktivität mehr Information zur Verfügung – z. B., welche Objecte im Diagramm gerade selektiert sind.
- Sie müssen nichts tun, als das Add-In zu installieren, d. h., es ist keinerlei Konfiguration erforderlich, um das Add-In benutzen zu können.

## **Anbindung an und Integration mit anderen Werkzeugen**

Enterprise Architect stellt mehrere Mechanismen zur Verfügung, um Ihr Modell mit Werkzeugen anderer Hersteller zu verbinden, einschließlich einer programmierbaren API, eines Add-in-Frameworks und vorgefertigter *Model Driven Generation (MDA)*-Lösungen von Sparx.

### **MDG Link**

Sparx hat mehrere MDG-Produkte zum Anschluss anderer Werkzeuge entwickelt. MDG-Link-Produkte sind exemplarische Beispiele für den Gebrauch des Add-in-Frameworks zur Erweiterung des Funktionsumfangs des Enterprise Architects. Der MDG-Link für Visual Studio® und der MDG-Link für Eclipse erlauben dem Enterprise Architect mit Microsoft® Visual Studio®, respektive mit den Eclipse IDEs zusammenzuarbeiten.

### **MDG Integration**

Die MDG-Integration integriert Enterprise Architect direkt in die Entwicklungsumgebung von Eclipse, bzw. von Microsoft® Visual Studio® 2005/2008/2010. Dieses Produkt erlaubt Anwendern die Einsicht in und das Bearbeiten von Modellen in der Oberfläche von Visual Studio® bzw. Eclipse und stellt viele der Schlüsselfunktionen des Enterprise Architects direkt in den Entwicklungsumgebungen zur Verfügung - auch die RTF-Dokumentenerstellung und die Generierung von Web-Präsentationen, MDA Transformationen, Baseline Management (Versionierung) und die Verarbeitung von XML-Technologie.

Weitere Eclipse-basierte Umgebungen, die die MDG-Integration unterstützt, umfassen Adobe® Flex® Builder™ und den Progress OpenEdge® Architect.

### **Import- und Exportfunktionen**

Datenaustausch zwischen verschiedenen Modellen und verschiedenen Werkzeugen ist bei Entwicklungsprojekten ein gängiger Vorgang. Enterprise Architect stellt die beiden meist verwendeten Formate XMI und CSV zur Verfügung, sodass es einfach ist, Modellinformationen zwischen verschiedenen Werkzeugen und Repositories auszutauschen. Da diese Funktionen auch im Enterprise API zur Verfügung stehen, können Modellimport und -export auch einfach automatisiert werden, was auch eine einfache Verteilung der Modelle und automatische Softwareerstellung erlaubt.

### **Unterschiedliche XMI Formate**

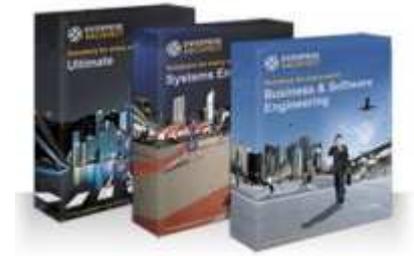
Enterprise Architect unterstützt den Import und Export für eine große Anzahl an XML Model Interchange (XMI) basierten Formaten. XMI ist eine Spezifikation zur Darstellung komplexer Modellinformation in lesbarer XML und als generische Lösung zum Austausch mit anderen Werkzeugen. XMI ist eine öffentliche Norm, die von der OMG gehalten wird. Enterprise Architect unterstützt XMI 1.0, XMI 1.1 und XMI 2.1. Diese Unterstützung mehrerer Formate ist essenziell, da viele Werkzeuge eine spezielle XMI Version voraussetzen.

### **CSV**

Zusätzlich zum XMI Import/Export bietet der Enterprise Architect auch eine einfache CSV Import-/Exportfunktion. Dies ist zur Ausgabe an Werkzeuge wie z. B. Microsoft® Excel® und zum Import ähnlicher Listen aus allen beliebigen Tabellenkalkulations-Werkzeugen.

## **Enterprise Architect Editionen**

Enterprise Architect ist in verschiedenen Editionen verfügbar; Ultimate, Systems Engineering, Business & Software Engineering, Corporate, Professional und Desktop. Jede Edition bietet unterschiedlichen Funktionsumfang, um die Bedürfnisse verschiedener Anwendergruppen von Einzelpersonenprojekten bis zu großen Projektteams zu erfüllen.



Ein *Floating-Lizenz*-Konzept ist für die Editionen Ultimate, Systems Engineering, Business and Software Engineering und Corporate verfügbar. Floating Lizenzen sind speziell für Firmen mit hoher Anwenderanzahl sinnvoll, wenn sie den Anwendern einen Lizenzpool zur Verfügung stellen wollen. Floating Lizenzen können von den einzelnen Anwendern temporär oder permanent – auch zum off-line Arbeiten – benutzt werden.

Eine ausführliche Aufstellung über die in den einzelnen Editionen enthaltenen Funktionen finden Sie unter <http://www.sparxsystems.de/uml/ea-price/> .

## **Prozessunterstützung**

UML ist eine Sprache, kein Prozess. UML beschreibt die Elemente einer Modellierungssprache und wie diese Elemente zusammengesetzt werden können, um Umstände in einer realen Welt zu repräsentieren. UML beschreibt nicht, wie diese Elemente der Reihe nach einzusetzen sind, um ein neues Softwaresystem aufzubauen.

Wie UML ist der EA prozess-neutral - d. h., er enthält alle Bestandteile und Funktionen um einen gewünschten Entwicklungsprozess umzusetzen, aber er schreibt nicht vor, wie dieser Prozess aussehen soll oder umzusetzen ist.

Viele EA-Anwender wenden hochstrukturierte Prozesse an, z. B. den Rational Unified Process (RUP), während andere Anwender flexiblere, agilere und weniger restriktive Prozesse bevorzugen. Unabhängig davon, welchen Grad der Prozessorientierung Sie wünschen, der EA enthält die benötigten Werkzeuge und Bestandteile, um den fortschreitenden Prozess einer Softwareentwicklung zu handhaben.

## Über Sparx Systems



Sparx Systems ist eine in Australien beheimatete Firma mit langjähriger Entwicklungserfahrung auf dem Gebiet der Modellierungswerkzeuge.

Sparx Systems ist beitragendes Mitglied der *Object Management Group* (OMG), dem verantwortlichen Standardisierungsgremium der UML-Spezifikation und vieler anderer, zugehöriger Spezifikationen.

SparxSystems Central Europe Software GmbH ist seit Mai 2004 in Europa aktiv, wobei zunächst mit der Übersetzung für den deutschsprachigen Markt begonnen wurde. Nunmehr haben wir uns als lokaler Ansprechpartner für den gesamten deutschsprachigen Raum etabliert, der Schulungen, Coaching und Projektbegleitung – auch auf Englisch - anbietet.

### **Firmen-Vision**

Sparx Systems ist der Überzeugung, dass ein umfassendes Modellierungs- und Entwurfswerkzeug für den *gesamten* Softwarelebenszyklus eingesetzt werden soll. Unsere langfristige Planung spiegelt dies wieder, ebenso wie unsere Überzeugung, dass eine Software für den Lebenszyklus genau so dynamisch und modern sein soll, wie die von Ihnen entworfenen und betriebenen Systeme.

Die Software von Sparx ist zum Einsatz durch Analysten, Designer, Architekten, Entwickler, Tester, Projektmanager und durch die Wartungsmannschaft gedacht; das ist also praktisch jeder, der in einer Softwareentwicklung oder einer Analyse beteiligt ist. Es ist die Überzeugung von Sparx, dass hochpreisige CASE-Tools in ihrer Sinnhaftigkeit für ein Team und damit für eine Organisation stark beschränkt sind, weil sie durch ihre Kosten den einfachen Zugang zum Modell und damit zum Entwicklungswerkzeug behindern. Daher fühlt sich Sparx Systems verpflichtet, einerseits eine akzeptable Preispolitik zu betreiben und andererseits einen EA-Reader kostenlos für jene zur Verfügung zu stellen, die nur die Einsicht in ein Modell benötigen.

### **Langfristige Ausrichtung auf unternehmensweite UML Werkzeuge**

Sparx Systems hat über zehn Jahre Modellierungswerkzeuge entwickelt und hat dabei den Enterprise Architect auf unternehmensweite Verwendung ausgelegt. Zusätzlich bemüht sich Sparx Systems, Anforderungen und Vorschläge aus der breiten Anwenderschar zu berücksichtigen. Dadurch konnten schon viele Funktionen eingebunden werden, die für Entwickler wichtig sind und viele Aspekte der Praxis abdecken. Sparx Systems fühlt sich der Weiterentwicklung sowohl des UML-Modellierungswerkzeugs Enterprise Architect als auch der zahlreichen Plug-Ins verpflichtet, die anwenderspezifische Anforderungen erfüllen.

## **Kontakt Daten**

SparxSystems Software GmbH

Office: Handelskai 340/5, 1020 Vienna, Austria

Tel.: +43 (0)662 90 600 2041

Fax: +43 (0)662 90 333 3041

e-Mail: [sales@sparxsystems.eu](mailto:sales@sparxsystems.eu)

Verkauf und Bestellabwicklung: [sales@sparxsystems.eu](mailto:sales@sparxsystems.eu)

Produkt-Support: [support@sparxsystems.eu](mailto:support@sparxsystems.eu)

Internet Deutsch: [www.sparxsystems.de](http://www.sparxsystems.de)

Internet Englisch: [www.sparxsystems.eu](http://www.sparxsystems.eu)

Deutscher Blog: [blog.sparxsystems.de](http://blog.sparxsystems.de)

Englischer Blog: [blog.sparxsystems.eu](http://blog.sparxsystems.eu)