

Release Details for Enterprise Architect 8

Performance

- Created new index packs for repositories to improve overall performance
- EAP models now support 'Load on Demand' (Lazy Load).
- Improved performance in the following situations (Results may vary on different machines):
 - Editing of rich text notes as experienced by some users.
 - Enterprise Architect start up.
 - Load time and general handling of very large models (3 million+ rows).
 - XMI generation in very large models.
 - Load of models when multiple profiles are loaded in memory.
 - Element deletion from very large models.
 - Moving elements between different parents in the Project Browser.
 - Synchronizing ports and parts of instances with the classifier.
 - Changing element selection when Element Browser is showing.
 - Diagram rendering when using "Enhanced 1" rendering mode, especially for diagrams with a large number of connectors.
 - RTF report generation.
 - Reverse engineering code.
 - Loading of the Project Browser.
 - Saving large files in the Source Code editor.
 - Traceability window modified to use lazy loading, significantly improving responsiveness in large models.
 - Loading of Diagrams containing Elements from Packages not yet opened in Project Browser.

User Interface

- Start Page updated:
 - New layout to improve usability and visual appeal.
 - Icons added to each command.
 - Icons added to distinguish between .EAP files and Server based repositories in the "Recent Projects" list.
 - Link to Sparx Systems Enterprise Architect Community Site added.
- Workspace Layout manager added to better support loading, saving and switching between window layouts.
 - Allows users to load and save screen layouts from file.
 - A collection of Workspace Layouts for common domains is

- included in the default installation.
- Users may define their own named Workspace Layouts.
- Quickly switch between layouts.
- Custom views may be saved in Workspace Layouts.
- There is now no limit to the number of named Workspace Layouts that can be saved.
- The 'Workspace Views' and 'Other Views' toolbars have been merged into the new 'Workspace Layouts' toolbar.
 - Supports quickly switching between Workspace Layouts.
 - Allows saving the current Workspace Layout to file.
 - Quick access all Views, Dockable Windows and Toolbars.
 - Provides an "Enter full screen mode" shortcut.
 - Option to check spelling as you type added to Linked Documents and Notes windows.
- New Diagram Filters capability may restrict or highlight elements shown on a diagram.
 - Supports hiding elements completely.
 - Also supports two disabled (suppressed) visual styles.
 - Allows for convenient selection of matched objects.
 - Extensive filter definition capability.
 - Easy to select and swap between various predefined filters.
- Many docked windows modified to give a cleaner and more open appearance.
- Improved Vista/Windows 7 compatibility by storing various per-user files (such as spelling dictionaries and the EAExample.eap) to the user's Application Data directory.
- Relationship matrix updated to correct error message shown when creating an aggregation relationship with the SPEM technology active.
- Relationship Matrix improved to support filtering based on Trace dependencies.
- Traceability Window extended to show elements on currently selected diagram.
- Traceability Window updated to prevent unwanted scrolling when expanding nodes.
- 'Open' toolbar button now contains a drop down menu, listing the most recently opened projects.
- Many dialogs modified to support resizing.
- Select <Item> dialog multi select improved to retain selection, when expanding node.
- Project Integrity Check dialog now supports saving results to file.
- "Use Circle Notation" and "Use Rectangle Notation" will now enable or disable according to the lock state of the current diagram.
- Diagram context menu now has command to insert an existing model element.

- Two new Visual Styles added to provide additional clean and modern user interface elements.
- Class Properties -> Details TAB -> Add Template. Increased the manual entry size of Class Templated parameter types to 50 characters.
- Insert Related dialog updated to better handle specifying the type of relationship or element when profiles specify metatypes with the same name.
- Creation of profiled Ports and Parts from the Project Browser is now possible.
- Tagged values page added to the table attributes and table operations dialogs for viewing and editing tagged values.
- Connectors can be automatically created between new instances of objects based off of the connectors between the original objects using the Paste Elements dialog.
- Selecting "Locate Related Object" from the context menu in the hierarchy window for a package will select the package in the project browser.
- XSD & WSDL dialogs have been enhanced with rich edit windows.
- Template Class Parameter type input field increased to 50 characters.
- Test Case element added to Use Case toolbox.
- Glossary definitions type now accepts user defined types.
- Note fields across EA allow the current selection to be added to the glossary.
- Swimlanes matrix heading font sizes have been tweaked to match the same size font in elements.
- DDL Name Templates button (found at Tools -> Options | Code Editors) will now give an informative error message if no model is loaded.
- Type combo boxes in dialogs (eg. Attributes, Operations, Parameters) modified for easier use and faster loading.
- Attribute properties dialog Constraints tab usability improved.
- Methods property dialog controls sizing improved.
- Type selection for Tagged value of type Classifier improved.
- Tagged values of type 'RTFTemplate' modified to also list templates provided by MDG Technologies.
- Cut and Paste support added to diagrams.
- Element type change using 'Change Type' improved to refresh type in Project Browser.
- Diagram editing will be disabled for any Diagram after Enterprise Architect detects it has been saved by another user.
- Floating toolbar for element appearance updated to show current value of border width.
- Element List can now be opened to show contents of root package.
- Element List in non-recursive list of packages will now allow double click to navigate into child packages.
- GoF Patterns technology added to allow dropping of GoF Patterns from the Toolbox.

- Custom toolboxes can define items for patterns stored in a model's resources.
- Relationship Matrix now supports drag and drop of project browser packages to set source and target.
- Scroll-wheel behavior of diagrams improved to follow Windows settings.
- Docked testing window modified to improve look of labels.
- Paste Composite Element dialog's 'Instance Of' combo box now dynamically resizes its drop down list based on the longest diagram name it contains.
- Validation of System Issues for resolution when closing corrected.
- Composite Elements created from profiles will now always be set as composite.
- Actions created using Quick Linker will now prompt for Action Kind when Control key is held down or user has not suppressed the prompt.
- The Help menu now contains an entry to open the Example Model.
- Web Browser view modified to accept addresses longer than address field.
- Nested Non-Package Elements in the Project Browser modified for consistent sorting in lazy loaded models.
- Find in Diagrams command updated to improve behavior after using keyboard to navigate model.
- Tabs in main view updated to include icons to reflect view or diagram type.
- Menu button added to Caption Bar for listing and selecting from the list of currently open views.
- UML Help pages linked to by element types updated.
- Timeline Dialog improved to allow easier reordering of states.
- Transitions page of Timeline Dialog improved to update state list after state changes.
- Test Cases imported from Constraints, Requirements, Scenarios and other elements are now set to 'Not Run' by default.
- Technology settings dialog updated to list technologies in alphabetical order.
- Technology settings dialog updated to show icon and logo for the Basic UML 2 Technology.
- Double click on element features updated to handle trailing whitespace when finding feature in docked windows.
- Toolbox page tooltips improved to remove information included from previously displayed tooltips.
- Docked note windows now receive an update after correcting a mistake using the model spell checker.
- Border width selection from floating context menu improved with consistent restriction on valid widths as format toolbar.
- 'Edit Connector on New' option updated to apply to all connector types instead of only sequence connectors.
- Security password dialogs updated to prevent entering passwords longer

than will be accepted.

- Project Glossary dialog updated to allow delete of multiple selected terms.
- Getting Started window improved to ensure scrollbar is shown when EA is loaded if necessary.
- Hyperlinks to files can now be set to open the file for viewing or editing using the windows default program for each action.
- Hyperlink option available when dragging file onto diagram or from hyperlink properties dialog.
- Tagged Values to specify specific connector line styles updated to ensure that line style can not be modified on a diagram.
- Diagram Caption Bar display improved to ensure separation of diagram icon and text.
- Select Trial Version dialog updated:
 - Workspace Layout selection added to assist users to customize display.
 - Hyperlinks to useful videos and web pages added to help users get going quickly.
- Deletion of elements from Model Search results improved.
- Rename of Operation Pre/Post Conditions improved.
- 'Convert to Instance' command renamed to 'Convert to Property' where required.
- Keyboard navigation of 'Link note to element feature' dialog improved.
- Element properties dialog updated to ensure changes to Status or Concurrency are recognised and will trigger a prompt to save if necessary.
- Added support for pasting diagrams under an element instead of only under packages.
- UML Stereotypes dialog updated to remove 'composition' from the base class list. Composition is not a base class itself and you can use association or aggregation instead.
- Type combo for Attribute and Operation types improved clear existing classifier when typing new type.
- Text Fields and Combo Boxes defining classifiers now support [Ctrl]+[Space] to invoke the Select Classifier dialog.
- Baseline compare difference dialog modified to provide better opening position.
- Option to automatically create instances when dropping a classifier on a diagram improved to remove situations where it could become on by default.
- 'Getting Started' task pane has been modified for easier navigability and to include features new to version 8.
- Element List modified to restore automatic expansion of all items in hierarchy view.
- Workspace Layout dialog modified to list Custom layouts before System layouts.

- Hyperlink creation with Unicode text in Linked Document Editor and note fields in WINE improved.
- Team Review Window updated to set focus to editor after creating an item.
- Many features adding new elements to a model updated to better handle models that are 'Load on Demand'.
- Workspace Layout files (.eaworkspace) created by previous versions of Enterprise Architect are now upgraded to be compatible with the current version when applied.
- New Glossary Term dialog will now fill with existing data if attempting to create a duplicate entry.
- Select Item Dialog improved to show current selection consistently in lazy loaded models.
- Auditing View updated to better support Lazy Loaded models.
- Select <Item> Dialog updated to restore behavior when clicking 'OK' button.
- 'Apply' button removed from Visual Style dialog when running under WINE to avoid hidden window issues.

Code and Note Editors

- Note editor now supports highlighting of glossary terms (blue underline) and spelling mistakes (wavy red underline).
- Glossary Manager now includes hyperlinked terms within rich text notes and the new Structured Scenarios.
- Glossary term auto-completion has been added to all text fields that support glossary hyperlinking. [Ctrl]+[Space] will bring up the Auto-complete suggestion box.
- A 'Search For...' context menu item added to all text edit controls, enabling you to search the current model, files and the internet for selected keywords.
- Ability to select scripts using Autocomplete in the Scripting console.
- Code editor controls can be configured to highlight the line that the caret is on through the CARET_LINE_VISIBLE property.
- Source Code Editor now prompts the user if the file that is being edited has been externally modified on the file system since it was opened.
- Source Code Editor now highlights only quoted strings in unrecognized file types rather than assuming the code is C++ by default. This enhances the way Enterprise Architect manages unknown code segments and generic text files.
- Source Code Editor enhanced with the inclusion of syntax highlighting support for VBScript and JScript.
- Source Code Editor autocomplete performance has been improved on larger models.
- Source Code Editor now handles Ctrl/Alt/Shift key modifiers correctly.
- Code Editor context menu now contains an entry to toggle the visibility of

the line numbering margin.

- Source Code Editor context menu now contains an entry to toggle the visibility of the structure tree.
- Code Editor options changed to default to always parse files when unexpected contents are found in size limit field.
- Code Editors no longer display calltips while their context menus are active.
- Source Code Editor context menu now appears in the correct location when invoked from the keyboard (eg using the [App] key or [Shift]+[F10]).
- Source Code Editor now advises if files have been modified by another process since they were last opened/saved in the editor.
- Source Code Editor now supports highlighting of parameters in method calltips.
- Source Code Editor now indents new lines after defining a function parameter list and scope start on the same line. (eg `void SomeFunction(){`).
- New toolbar added to code editor, allowing quick access to commonly used functionality.
- Modification of read only files is now prevented.
- New search function 'Search In Open Files'. Available from editor context menu and File Search window.
- Numerous improvements to the note editor under WINE.
- Source Code Editors now process up/down arrow key presses when Mouseover tooltips are displayed.
- Find/Replace dialog in Source Code Editor is now moved if it would obscure found search terms.
- Note Editors updated to improve handling of hyperlinks after the current cursor position.
- Source Code Editor intellisense now provides mouseover information for symbols.
- Source Code Editor's Autocomplete list now restricts selections to Classes and Interfaces of the current editor language.
- Signature tooltips no longer shown when attempting to cast a variable.
- Intellisense now provides information from the entire class hierarchy.
- Find/replace dialog now appears in the same place it was last closed.
- Local scripts no longer become editable after being executed.
- File Search dialog results improved to always highlight term being searched for after double clicking on results.
- Spelling mistakes now formatted with red wavy underline instead of red text with underline.
- Glossary terms now formatted with blue underline instead of blue text with underline.
- Glossary terms can be added to notes windows using the autocomplete menu by pressing [Ctrl]+[Space].

- List styles in the notes window will indent consistently when applying the style to multiple lines that contain both lists and plain text.

Structured Scenario Editor

- New capability to edit and manage structured scenarios for Use Cases (as well as other types as required).
- Additional window to manage structured scenarios (additional to existing Notes based description).
- Structured Scenarios include:
 - Numbered steps of type "Actor" or "System".
 - Step description (Action) field.
 - Uses field for Inputs, Requirements, References, and Rules.
 - Result field to describe testable results.
 - State field to record the current state of the progressing Scenario.
 - Links to Alternate paths.
 - Links to Glossary items.
 - Links to other elements.
- Optimized for rapid entry of scenarios.
- Context References window added to list all referenced and linked elements.
- Referenced and/or linked elements will appear hyperlinked in structured scenario text.
- Ability to reverse engineer existing notes or clipboard content into structured steps.
- Ability to reverse engineer simple activity graphs into structured steps.
- Ability to generate a variety of activity diagrams from the structured scenarios and alternate paths.
- Ability to generate state diagrams based on steps and the "State" field of each step.
- Ability to generate Rule Composer task graphs based on Scenarios (requires a preexisting "fact model").
- Ability to generate Internal tests for the entire scenario suite (all logical paths) based on path and Result field.
- Ability to generate External "Test Case" tests for the entire scenario suite (all logical paths) based on path and Result field.
- Ability to generate Sequence diagrams based on each step containing one or more referenced or linked Elements.
 - Each step containing two elements referenced will become a call message between the two referenced elements.
 - Each step containing one element will become a self-message on that element.
- Ability to generate Robustness diagrams based on each step containing one or more referenced or linked Elements.

- All diagrams and test cases can be generated from the automation interface. (See Automation Interface and Add-In Support for more details)
- Structured scenarios exported in XML format within standard XMI import/export.
- Full support via Automation Interface for adding, reading and editing structured scenarios.
- The Rules and Scenarios window is now dockable in the main view.
- The Rules and Scenarios window has been restructured to behave consistently with other windows. This includes:
 - Replaced the tree that was part of the window with navigation from the Element Browser.
 - The three different views previously used are now explicitly available by selecting the relevant tab.
- Docked Window modified to no longer take focus from other windows.
- Context menus modified open under selected items when using keyboard stroke.
- Constraints list updated with informative tooltips on its toolbar.
- Docked Window updated to Enabled visual styles for the Scenario & Requirements docked window.
- Reloading the Basic Path improved to load faster.
- Generation of Activity and State diagrams from Structured Specification improved to support synchronization of Diagram Layout with existing diagram.
- Creation of extends connectors modified to reverse the direction of the connector created.
- Alternate Path creation commands improved to prevent situations that appeared to allow Alternate Paths to be added to Alternate Paths.
- Context Switching while unsaved changes exist in Scenario editor improved.
- 'Join' column of 'Entry Points' list updated after reordering of steps.
- Scenarios & Requirements View updated to select current item being edited in tree.
- Docked Scenarios & Requirements updated to ensure Basic Paths are always available.
- Import Test Cases from Scenarios updated to import text from structured scenarios.
- Scenario steps can be created from notes or clipboard text delimited by new lines or sentences.
- Link Note to Element Feature now can be set to link to Structured Scenario Text.

Element Browser Window

- Viewing of Test, Project and Maintenance items has been added.
- Viewing of Attributes and Operations that the currently selected element is

linked to has been added.

- Performance improved when selecting objects.
- Clear visual cue of grayed out icons provided when no element is selected.
- Scenarios shown will also include Structured Scenarios and their steps.
- Expanded possible actions that can be performed from the Element Browser including:
 - Add breakpoints and view code for Attributes and Operations.
 - Control selection in the Scenario and Requirements window by selecting Constraints, Requirements or Scenarios.
 - Open to the properties of any item by double clicking an item or selecting Properties from the context menu.
 - Double click on a link will open properties for the link, properties from context menu opens properties for target element.

Team Review Window

- Name changed from Discussion Forum to describe how it can be effectively used.
- Discussion tree moved from view into separate docked view.
- General usability improved.
- Changed default visual style of posts to "page view" and allowed for saving user preference.
- Saving a post now better handles when another user may have modified a post while it was being edited.
- Added Review Status on Team Review items.
- Opening a Post will now always retrieve latest data instead of using cached data.

Model Search Window

- Search configuration and options window merged into main view. (No longer a popup dialog.)
- Improved and additional capabilities for the advanced SQL search editor provided.
- Ability to view source code on returned items using [Ctrl]+[E].
- New search: 'Element Name'. Provides a unique match on element name or alias including a possible attribute or method name.
- Added the Alias field to the Simple search.

Diagramming

- Entering in-place edit will now scroll active Diagram to ensure editor is visible.
- Action effect text can now be truncated and wrapped using the Feature

Visibility dialog for the action element.

- Layout Diagram no longer moves Note or Text elements when there are no connectors on the diagram, consistent with the hierarchical diagram layout.
- Interaction Occurrence elements enable the diagram save button when changes are pending.
- Background color used for feature stereotype groups improved when using strong colors.
- Text elements aligned to Center or Right will now draw correctly under Crossover.
- ArchiMate shape scripts improved for Grouping, Meaning, Object and Product elements.
- Interaction Occurrence elements modified to allow for filtering using diagram filters.
- Sequence diagrams updated to allow source activation from asynchronous messages to be extended.
- Quicklinker updated to support multiple stereotypes and will now show quicklinks for all stereotypes that apply.
- Diagrams created from a Template Package now use current author and creation date instead of original values.
- Diagram Filters updated to support filtering of elements with Alternate Images.
- Package Elements showing contents improved when multiple packages with the same name are being shown on the diagram.
- Copy and Paste behavior of elements restored to diagram using undo/redo improved.
- Creation of some elements from builtin profiles improved.
- Creation of profiled packages from the Toolbox improved to add them to the same package as the diagram.
- Requirement added to "Common" quicklinker sub-menu.

Shape Scripts

- Custom properties of the element can now be tested against and displayed.
- Multi-segment connectors will now only draw Shape Script for custom line on middle segment.
- Added classifier.type element property.
- Modified display of valid properties with invalid values.
- Shapescrpts and metafiles can be applied to element stereotypes to change their appearance on sequence diagrams using the UML Types dialog.
- Connectors on sequence diagrams can now have their appearance changed using shape scripts.
- HasProperty function updated to support retrieving values from add-in functions.

Code Engineering

- Generation of get properties for Delphi improved.
- Generate Package Source Code dialog improved to list SystemC files.
- File Editor improved to handle modified default file extensions.
- Generation of PHP 5 abstract methods improved.
- Generation of templated C++ classes modified to generate body for default constructors and destructor if they are being generated..
- Transformation modified to allow specification of additional values of attribute containment during a transform.
- Operations on C# classes now have the modified "unsafe" in the Advanced Options list.
- Python reverse engineering now supports version 3.3.
- Delphi reverse engineering supports additional Delphi 7 constructs.
- C# reverse engineering now supports the fixed modifier on fields.
- Python reverse engineering modified to accept empty class inheritance lists.
- C++ parsing updated to support forward declaration of template instantiations.
- C++ reverse engineering updated to import the scope of nested types.
- Actionscript parsing updated to include support for regular expressions.
- Code Template Editor updated as follows:
 - Sorting of the template list will now preserve the current selection.
 - Prompts to save unsaved changes added in a number of circumstances including when clicking a stereotype override and creating a new template.
- Preprocessor Macro changes will now be reflected in parser from next parse instead of needing to restart EA.
- Code Template auto-complete list updated for macro attAllowDuplicates.
- Round trip engineering support for Delphi 'object' and 'metaclass' types.
- VB.Net State Machine code generation templates improved.
- Delphi get property code generation templates improved.
- Code Generation of current class updated to ensure current context is use for generation.

Database Engineering

- Prevent duplication of Oracle packages when synchronizing.
- Display details of DBMS, server and database on import DDL dialog.
- Backward compatibility for ODBC DDL schema import from PostgreSQL version 7.
- Import multiple PostgreSQL table triggers.
- Added option to override default foreign key name template.
- Support for PostgreSQL citext and date/time/interval - with/without time

zone data types.

- PostgreSQL generated DDL includes identifier delimiter characters when that option set.
- Modified drop statements and SQL terminator for Sybase ASE generated SQL.
- Set unique column flag on importing table as new object, as well as when synchronizing.
- Increase attribute initial field size to maximum of 65535 characters.
- Added auto-increment Column Properties for Sybase ASE and Sybase ASA.
- Ensure the Class properties dialog opens instead of the database function dialog where the class is stereotyped <<block,function>>.
- Ensure DDL schema generation creates unique Oracle sequence and trigger names.
- ODBC Import from SQL Server - import correct extended property for table and column comments.
- Remove random "Unspecified Error" during ODBC Import of schemas into an Oracle repository.
- Ensure SQL Server view definitions are fully imported without truncation.
- Column Comments no longer generated when comment level is 'None'.
- PostgreSQL generation now sorts the generated database objects taking into account the dependencies between them.
- Oracle import updated to handle NUMBER data type when using Oracle ODBC Driver 11.01.00.07.
- Access 2007 data modeling support added.
- ODBC import modified to allow apostrophes in ODBC data source names.
- ODBC import now allows unicode characters in ODBC connection string.
- Oracle Database Engineering improved including:
 - Support for importing and generated Duration/ON COMMIT clauses.
 - Generation of COMPRESS clause improved.
 - Added option for handling of character length semantics.
 - User name now automatically added as schema during import.
- Sybase ASE 15 ODBC Import updated to include foreign keys.
- Informix Database Engineering improved including:
 - Binary characters no longer saved to Stored Procedure text.
 - User defined functions can now be imported.
 - Foreign Keys import improved.
 - Maximum precision for decimal/numeric data types increased to 32.
 - Imported Diagrams updated to ensure display of objects, attributes and operations.
 - Generated DDL improved to match data model content.

- Informix datatypes updated. Available either by download from the Resources page of the Sparx website, in EABase.eap or EAExample.eap.
- SQL Server Database Engineering improved including:
 - Added option for handling of default constraint clauses.
 - SQL Server 2005 & 2008 datatypes added to EABase.eap and EAExample.eap.
- DDL Transformation updated to set foreign key column to NULL if source multiplicity is optional (*) or (0..*).
- Data Model - Synchronize the datatype of unique constraint parameters to with datatype of the column that is part of the unique constraint.

Business Rules Engineering

- Fact Model renamed to Business Domain model to more accurately reflect its purpose.
- Added support to model business rules logic for operations using Rule Composer.
- Rule Composer window modified to check for valid data before opening.
- Rule Composer updated to handle cell contents that are too large to display.
- Rule Composer will no longer show context menu when right clicking on an empty cell.
- Rule Composer will now allow Activity Parameters to be used as Condition and Action Variables.
- Rule Composer will now allow Operation return values to be used as Condition Variables.
- Rule Composer updated to support passing parameters to Operation Calls in Decision Table.
- Rule Composer modified to save multiple links to an Attribute in Computational Rule table when the expression values are empty.
- Rule Composer enhanced to allow multiple links to an operation to exist in decision table.
- Rule Composer modified so that tab name is preserved when editing.
- Rule Composer modified to select the appropriate tree item when dropping any attribute/operation/activity parameter on the Decision Table.
- Rule Composer now allows resizing of Condition and Action grids.
- Rule Composer usability enhanced by enabling option to add row from header row of the rule composer tables.
- Rule Composer interface improved by ensuring vertical alignment of columns in the decision table.
- Code generation modified to generate conditional logic as using nested If-else blocks instead of simple If statements.
- Code Generation modified to resolve role names for Parameters passed to

Operations that are linked to Rule Composer.

- Drag and Drop of multiple tree nodes onto Rule Composer displays appropriate message.
- Object deletion will now update the Rule Composer if the object was being used.
- Object and package deletion updated to remove associated Rule Composer data.
- XMI round trip will no longer add data for Rule Tasks.
- Now possible to save from the Context Menu on the Rule Composer tab.
- Rule Tasks created from quick linker will now have a default value for Effect.
- Edit Allowable Values dialog will now preserve focus after clicking Save.
- Rule Composer now viewable in Lite Edition.
- Validation command added to Rule Composer.
- Drop-down options in Rule Composer modified to open with single click.
- Column Headings updated to be more informative.
- Traceability between Decision Table and Rule Statements improved by highlighting relevant information from each table when selecting anything in the other table.
- Keyboard navigation of 'Edit Allowable Values' dialog improved with accelerators, creating a new entry after previous is accepted and support for delete key to remove entries.
- Task Pane for Business Rules added to provide easy access point to help.
- In place editing of Condition, Action and Computation Action columns added including auto-complete functionality.

Execution Analyzer/Visualizer (Profiler)

- New Profiler feature providing sampling of native Windows applications.
- Enables rapid analysis and real time visualization of process with call graphs of individual threads.
- No instrumentation required.
- Sampling frequencies typically set between 1ms and 15ms
- Reports quickly identify code in application with highest call frequency.
- Profiler Reports can be loaded/saved in either XML or binary format.
- Profiler Reports can be stored as resource items in the Team Review center.
- Suitable for profiling any Windows application for which source code and PDB files are available.
- Also works under WINE running under Linux or CrossOver Office running on Mac OSX 10 (Intel).

Execution Analyzer/Visualizer (Debugger)

- New main window Toolbar added with Debugging and Recording

operations.

- List controls in debug windows modified to accept Ctrl+A to toggle selection of all items.
- List controls in debug windows updated to support copying contents to the clipboard.
- Watch Items window now has a toolbar to assist in managing watches.
- UAC issues in Windows 7 have been addressed.
- Recording Marker Sets can now be created and stored for selected Elements within the model.
 - Rapidly bind all methods (or a subset) of a single or multiple elements with recording markers
 - Record all hits on the bound markers between user interface actions or other "found messages".
 - Visualize the recorded scenario as a detailed Sequence diagram showing all incoming messages and system responses.
 - Record complex and ad hoc interactions over time and capture actual system responses in graph form.
 - Store, manage and re-use marker sets and rapidly change between sets.
- Ability to quickly markup entire class for recording whilst attached to a process. Drag and drop to breakpoint/marker set window will cause debugger to bind and record any methods it discovers for class.
- Stack Record Threshold can now be set for all platforms.
- Attach to Process no longer requires a debug command to be set up.
- Breakpoints can now be set when package doesn't have debugging configured.
- Build command will now save all modified source files.
- Code Editors no longer marked read-only during debugging.
- Native code debugging improved to prevent breakpoints being left in after selecting 'Run' or 'Step Out'.
- Build Scripts can now be set as the default for a model instead of just a package.
- Debug Output window will now stop scrolling when new items are added after the user has selected one.
- Modifications to handling of Package Build Scripts for Non-Namespace Root Packages.
- Breakpoints for model now stored in local users common data folder .
- Call Stack for native Windows applications expanded to show greater detail.
- Watched items modified to store for current user instead of directly against the package. Allowing each user to have a different set of watched items.
- Native debugger improved display of array elements declared as pointers.
- Improved debugging of C operations.

- 'Attach To Process' dialog updated to remember last filter for user or system processes.
- Hyperlink elements can now specify a specific script to run.
- Added ability to drop a Package Build Script on to a Diagram to create a hyperlink.
- Stack window updated with new button to toggle stack display to show all frames or those with source code associated.
- Locals window updated to maintain scroll position while stepping through a function.
- Locals window updated to highlight changed variables while stepping through a function.
- Memory Viewer available for inspecting memory addresses when debugging native applications.
- Variables with a primitive type can now be edited in-place during debug session.
- Warnings of source being more recent than breakpoint module now shown only when this is true.
- Package scripts in the debug window can be pinned. Ensuring that it is always the active build script regardless of how selection changes.
- Profiler modified to list partial stack traces in a collected <unknown> folder.
- Saving Debug Output to File, updated to save to the path specified instead of the default folder.
- Tree controls contained in the Debug Tasks, Debug Stack and File Search dockable windows will now expand/contract if double clicked.
- Fixed issue with connecting Execution Analyzer to Tomcat and other Java services
- Enabled limiting of Sequence diagram trace depth within .NET and Java (previously limited to Native only)
- Enabled full stack trace when "pause" button hit
- Enabled full stack trace on exception caught
- WINE (Linux and MAC support)
 - Full debugging and profiling support of debuggable Windows native executables on Linux and Apple MAC systems.
 - Debugging of Native Windows applications (debug executable with source and PDB files).
 - Works under WINE or Cross Over Office on Linux based systems.
 - Works under Cross Over Office running on Mac OSX 10 (Intel).

Scripting

- Updated Mozilla Script Engine to 1.8.
- Messages output by Script Session object now use proper Locale for translation.
- Script toolbar Refresh button disabled while refresh in progress.

- Drag and Drop/Copy introduced to manage user scripts.
- Search in Scripts added to Search Toolbar and Editor context menu.
- Saving scripts from the script editor modified to use the target directory specified by the user.
- User defined templates can now be applied to new Scripts created through the Scripter dockable window. These script templates in the Config\Script Templates directory of the Enterprise Architect installation root.
- The Scripting Window now allows users to classify Scripts under different classes of Script Groups. These groups include:
 - Normal Scripts: Ordinary scripts, executable from the Scripting window.
 - Project Browser Scripts: Scripts that can be executed from the Project Browser's context menu.
 - Workflow Scripts: Scripts that will be called by Enterprise Architect's workflow engine.
- Updated appearance of the Scripting dockable window's tree control and toolbar.
- Tree control in the Scripting dockable window will now expand/contract if double clicked.
- Script console updated to handle auto complete invocation when no script engine is created.
- Scripting tree improved for deletion of script groups.

Workflow Support

- Workflow Script Processing introduced to provide enhanced model management and collaborative development.
- New script engine available in Corporate and above editions.
- Consumes "Workflow Scripts" written by end users.
- "Workflow Scripts" define the rules and behavior concerning modifying and managing elements within a model.
- Rules manage changes to Status, Version, Name, Alias, Author, Phase and other critical fields.
- Rules may deny changes or limit access to fields.
- Script engine provides access to the current security profile of the current user, proposed change and existing state objects.
- Workflow searches may also be defined to provide running lists of current workflow assigned to groups and/or users.
- Workflow scripts are able to report back to the user the reason why a validation check did not pass.
- Workflow scripts have the ability to veto changes to an Element in the same way an addin is able to.

WINE/Crossover Office Support

- Speed of drawing UI and startup improved to comparable levels with native Windows performance.
- Numerous improvements to speed and quality of drawing user interface controls.
- Numerous other fixes specific to running under WINE.
- Display Images in internal RTF Viewer improved under WINE.
- Improved Notes entry windows under WINE to support spell checking, glossary and other UI enhancements.
- Execution Analyzer changes for WINE:
 - Debugging of native Windows applications running under Wine/Crossover 9 now supported.
 - Execution Analysis of native windows applications under WINE supported, including sequence diagram generation.
 - Profiling of native Windows applications running under WINE now supported.

UML Specific

- Actions' context menu improved to load 'Advanced options' based on the action's type.
- Class property 'isActive' handling improved.
- Set Object State command modified to allow selection of states owned by an object classifier and superclasses.
- Syntax Validation of Control Flows modified to prevent occasional error when between two Ref Fragments.
- Qualifier support improved on connector ends, including support for Property based qualifiers.
- Parameters can now have multiplicity information specified.
- Modeling of Properties improved as follows:
 - Qualifier support improved on connector ends, including support for Property based qualifiers.
 - Added Qualifiers for Attribute, Ports and Parts.
 - Added Subsetting, redefining capabilities for Attributes, Ports and Parts.
- Interaction Occurrence Arguments modified to load correctly from models saved prior to version 7.5.
- Loop Nodes and Conditional Nodes support enhanced.
- Triggers modeling improved, with ability to reference to other model elements based on their 'event' type.
 - SendSignalAction, BroadcastSignalAction support improved with provision to define arguments to be transmitted to target objects.
- Ability to associate an operations' behavior to any existing behavior element added.
- Value Specification Action has been added to the list of actions that can be

created.

- Embedded elements dialog improved to show owned and inherited elements with same names and unnamed inherited elements.
- Added ability to associate input/output pins as properties of Actions.
- Asynchronous signal messages improved to refer to signal elements, and specify arguments for signal attributes.
- Rendering of part multiplicity in sequence diagram improved.
- Reused Ports and Parts updated to allow them to be inserted and deleted.

SysML Specific

- Drawing of Properties modified to use dashed border when isReference custom property is set.
- Drawing of Properties modified to display name at top of shape when shape is made larger.
- Quicklinking of SysML Requirement elements improved to include definitions from EA base Requirement elements.
- Allocate dependency added to quicklinker between two Block elements.
- Allocate dependency added to the Block Diagram toolbox.
- Parametric Diagram toolbox improved ('Argument' removed, 'Property' added).
- SysML Requirement element has "text" tagged value added.
- Block and Internal Block toolboxes now include UML Signal element.
- Diagram Rendering of SysML elements when SysML is disabled has been improved.
- Diagram Frame text for SysML diagram improved.
- Model template for capturing system requirements, constraints and designs in SysML has been added.

BPMN/BPEL Specific

- Added support for migrating BPMN 1.0 constructs to BPMN 1.1.
- BPEL Generator modified to generate the Gates in the order specified in the SequenceFlow tag 'GatewayOrdering' for Exclusive Data and Inclusive gateways.
- Tagged values on diagrams now refreshed after being being changed using the BPEL Properties dialog.
- BPEL Generator improved when resolving a Send Task in the Exception Path to its upstream Receive Task in the Normal Path.
- BPEL Generator Improved when resolving a Message End Event to its upstream Message Intermediate Event.
- BPEL Model Validator updated to add the following checks.
 - Message selected in a Task/Event belongs to the correct Web Service.
 - Correct Participant has been selected for a WebService.

- Message belonging to a Synchronous Operation is used asynchronously.
 - Validate Assignments referenced from BPMN 1.1 SequenceFlow connector.
- BPEL Generator updated to support generating Assignments set on Gates.
- BPMN 1.1 SequenceFlow connector on a BPEL diagram modified to show custom BPEL Properties dialog.
- Autosize behavior of BPMN Activity, Start Event, Intermediate Event, End Event and Gateway elements improved.

HTML Reporting

- Report generation on Oracle repository having more than 1000 elements in a package.
- Modified generated files to use Windows style line endings consistently for all files.
- Improved memory usage when generating on large projects.
- Added the following new fields:
 - Operation Behavior: #BEHAVIOR# (Was previously part of #NOTES# field.)
 - Stylesheet name: #CSS#.
- Message and Message Item sections added to support Sequence Messages.
- Body-Object section: #KEYWORDS# and #LEAF# fields updated to return correct result.
- Icons used in generated tree control improved when generating under WINE.

RTF Reporting

- Section Tree improved to sort sections in the same order they are found in the document when loading templates.
- Added the following fields and sections:
 - Added "Version" field to the External Requirement section.
 - Added sections Pre/Post Constraints.
 - Added sections for Structured Scenarios.
 - Added Linked Feature section.
 - Added Connector Trigger section.
 - Added StereotypeEx to External Requirement section.
- Report generation modified as follows:
 - Indent style generated for bulleted lists from notes changed to indent word wrapped lines the same as in the notes editor.
 - Rendering of Diagram_element section improved when an Element filter is in place.

- Generation from a Model Document modified to ensure Attribute sections are generated correctly.
- Rendering of empty Diagram_element section improved when the Element section isn't enabled.
- Legend and Boundary elements are now documented in the Diagram Element section.
- Hyperlink and other Diagram elements no longer get generated in the Child Element section.
- "Signature" field for methods in the "Linked Feature" section modified to include parameter types.
- Fields being generated will now have they previous style applied to them.
- Stopped headings being removed in sections when immediately followed by a table.
- Heading levels now automatically indent on Package and Element sections to provide consistent styles across a document.
- Model Document generation for templates where the Package child section isn't enabled have been improved.
- Report generation time slightly improved.
- Prevented SQL error that could occur when generating an RTF document from an EAP file.
- Element.Realize updated to generate a a value regardless of if realized classes are included in the report.
- Template and document editor modified as follows:
 - Insertion of Package section updated to allow insertion outside of a table when the Model section is in a table.
 - Editor modified so that changing between Fitted and Page Mode layouts updates menu correctly.
 - Loading and Importing of RTF documents modified to always update the SSBookmark style.
 - Template list modified to prevent wait cursor when navigating using the keyboard.
 - Elements with Unicode characters in the name can now be dropped into Linked Documents to create a link.
- Import/Export of "RTF Project Constants" via reference data added.
- Project Constants are now initially sorted by name in 'Generate RTF Documentation' dialog.
- Generate Dialog modified to set focus when it loaded.
- Connector Trigger generation improved to remove SQL error.
- RTF templates that have an empty Element Diagram section will use the Package Diagram section of the template instead.

MDG Technology Support

- MDG Technology Wizard options for visible pages improved when using

back button.

- MDG Technology Wizard default selections improved for when building a technology in parts.
- MDG Technology Wizard modified to handle document templates containing ', ', '(' or ') '.
- Introduced MDG Technology for Eriksson-Penker Extensions.
- UML Profile saving and loading updated to support a stereotype named 'Documentation'.

Automation Interface and Add-In Support

- New docked window support:
 - New Add-Ins docked window that any Add-In can add a window to.
 - Uses the new `Repository.AddWindow` method.
- New broadcast events. (See User Guide for more information):
 - Open and close of diagrams: `EA_OnPostDiagramOpen` and `EA_OnPostDiagramClose`.
 - Creation of diagrams: `EA_OnPreNewDiagram` and `EA_OnPostNewDiagram`.
 - Deletion of methods: `EA_OnPreDeleteMethod`.
 - Deletion of attributes: `EA_OnPreDeleteAttribute`.
 - Creating an object instance makes existing broadcast `EA_OnPostNewElement`.
- New functions. (See User Guide for more information):
 - `Repository.GetTreeSelectedElements()` .
 - `Repository.VersionControlResynchPkgStatuses(bool ClearSettings)` .
 - `Package.VersionControlGetLatest(bool ForceRefresh)` .
 - `Package.VersionControlPutLatest(string Comment)` .
 - `Package.VersionControlResynchPkgStatus(bool ClearSettings)` .
 - `Project.CreateBaselineEx()` .
 - `Project.ExportPackageXMIEx()` .
 - `Project.GenerateDiagramFromScenario()` .
 - `Project.GenerateTestFromScenario()` .
- Improvements made to the following:
 - `Package.VersionControlAdd()` . Improved handling of the error condition where the object package is already under version control.
 - `Package.VersionControlRemove()` . No longer raises an exception when no error occurred. No longer attempts to delete the XMI package file. Now refreshes tree icon.
 - `Repository.GetTreeXML()` . Modified to ensure tree is fully loaded.

- Repository.GetTreeXMLByGUID() . Modified to ensure tree is fully loaded.
- Project.RunHTMLReport() . Updated to support GUIDs passed in as XML GUIDs or regular EA GUIDs.
- Repository.GetElementSet() . Now returns an empty collection when provided with an object query that produces no results.
- Package.Elements . Now excludes elements contained within sub-elements.
- Package.Diagrams . Now excludes diagrams contained within sub-elements.
- Diagram.DiagramLinks - Now includes connectors that haven't yet been saved on the diagram.
- EA_OnPostNewElement now supports Add-In modifying Tagged Values.
- Package objects modified to handle situation where the object connector collection is accessed immediately after Package.Update() is called on a new package.
- Package objects modified to update the package name after updating the Package.Element.Name property.
- Add-Ins menu on Diagram no longer duplicates entries.
- Java wrapper for automation interface improved mapping of EA methods returning integers or booleans.
- Locking check for Connector updates improved in lazy loaded models.

XML Technologies

- XMI 1.1 import of Interaction Parameters and Occurrence Arguments improved.
- XMI 1.1 and XMI 2.1 will now round trip Sequence Message co-ordinates.
- XMI 1.1 and XMI 2.1 will now round trip Custom References between Elements.
- XMI 1.1 and XMI 2.1 export of Attributes, Ports, Parts improved.
- XMI 2.1 serialization of Activity Parameters improved.
- XMI will now round trip Relationship Matrix Profiles used in a package.
- XMI 2.1 import will now import RefGUID typed tagged values from other tools.
- EMX importer updated to include support for the following:
 - Input/Output/Value Pins.
 - Interaction Operands under a Fragment.
 - Linking a Message to an Operation signature.
 - Resolving Interaction Use to its Linked Diagram within a model.
- Eiffel Studio XMI 1.0 import will now import Generalization connectors.
- Rose XMI 1.1 exporter updated to include the metamodel information in the XMI.header element of the XMI file.
- Schema import now supports schemas containing multiple prefixes for the

Target-Namespace.

- WSDL generation can now validate the correct operations between WSDL Binding and PortType.
- WSDL Service dialog modified to prevent clearing of Location after double clicking on a Service in a diagram.
- XMI Import of State Flow (Transitions) modified to prevent Guard and Trigger information being included in the name.
- XMI Import from other tools updated to support the following:
 - Constraints from Oracle JDeveloper.
 - Cross references in an alternate format from RSA.
- Virtual Document Package links maintained when using 'Strip GUIDs' and the target packages are outside of the XMI being imported.
- WSDL Importer updated to validate compatibility between the Bindings and PortTypes with multiple operations.
- UML Pattern merge updated to check for existing connectors before creating new ones.
- UML Pattern merge updated to support setting attribute classifiers to classes outside of the imported pattern.
- XMI 1.1 import updated to handle attributes without a name.
- XMI 1.1 import updated to improve import of operation parameter of "return" type.
- XMI 2.1 import will now import UML Attribute properties from the UML section of an EA exported XMI 2.1 file.
- XMI 2.1 imported will now import language specific data types from files that have been exported without EA specific data.
- XMI import updated to handle empty element representing a model-document, element-script or rule-composer.
- XMI import updated to ensure child packages with version control applied are maintained.
- XMI import updated to preserve Version Control state.
- XMI export performance improved in situations where a large number of attribute tagged values exist in the model.
- XSD Element Properties dialog modified to set default type of incorrectly set the dataType of parent XSDtopLevelElement to the created XSDsimpleType.
- XMI import updated to better support auditing.

Model Management

- Added support for copying and pasting a Package or Element(s) within a model.
- Reference Data import of CSV styles into Oracle improved.
- Reference Data transfer of code templates improved to prevent duplicate templates being created.
- Connection to empty repository will now prompt to transfer some data.

- Export of Packages to XMI (1.1 and 2.1) enhanced to support export of child packages as stubs only:
 - Stubbed package export enabled for Controlled packages and Diff/Merge.
 - Diff/Merge enhanced to work with stubbed packages.
 - Support for baselining root and high order packages within deeply nested and complex model branches.
 - Support for baselining stubbed packages via automation allows for scripts to process entire branches.
 - Fast Diff/Merge capability in large models based on use of stubbed packages instead of full branch baselining.
- CSV Import/Export imports data when the CSV_KEY field is not populated but the 'Preserve Hierarchy' option is on.
 - Previously only the last entry in the CSV input file would be updated or imported if no hierarchy data was specified. Now all entries are imported.
- Data Transfer to Oracle repositories improved for large models.
- Project Transfer dialog Source and Target fields made larger to allow full connection string to be displayed.
- Connection Name dialog modified to no longer change the case of the connection string.
- Project Transfer to Open Edge repository updated to preserve connector notes.
- Change Project Transfer so t_secpolicies is transferred last, to avoid requests for security details in the event of a transfer failing after the enable security flag and before the security user data.
- Allow connection to Oracle repository using Oracle ODBC driver.

Security and Version Control

- 'Apply/Release User Lock' command in Lock to Edit mode now enabled for version controlled, checked-in packages.
- Checkout operation improved when used in 'Lock to Edit' security mode. Now forces refresh of lock status as part of operation, to ensure that newly applied locks are recognized.
- Checkout using 'Lock to Edit' security mode: after application of the user lock (part of check-out), now updates lock status immediately. Previously, the import was failing due to use of stale values for lock status.
- Import Model Branch file selection dialog now responds to double-click on file name.
- New function for resynchronizing VC status of a selected package. (Existing function resynchronizes ALL packages.)
- 'Check-in Branch' functionality modified to ensure that all checked-out packages are reliably discovered when using Lazy-Loaded models.
- Project Browser modified to draw Alternate Icons and Icon Overlays for partially visible items.

- Added and improved several automation functions to manipulate version controlled packages. See 'Automation Interface and Add-In Support' and the User Guide for more information.
- Added support for advanced security support via user configurable scripts to give fine grained edit permission to users based on security groups. (See Workflow Support)
- Checking of locks for very large packages under Oracle improved.
- MDG Integration: Context menu "Version Control" performance improvement determining package status for enabling/disabling (and hence displaying) menu items.
- User and Group locking mode will now automatically lock a diagram when a user attempts to modify it and unlock when it is saved.
- Locked Elements will no longer allow Behavioral Diagrams to be created under them from the context menu.
- Port instances will no longer report locking error when not locked.
- Release of security locks for a dirty diagram now give prompt to save changes.
- Security passwords can be changed on Access 2007 repositories.
- Resynch VC statuses command updated to handles Packages that are checked-out to another user.
- Check Out of Package associated with an uncontrolled XMI file, updated to report 'XMI Not Controlled', instead of 'XMI Checked-Out to Other'.
- Version Control Settings dialog updated to support sorting of the defined configurations.
- Manage Locks dialog updated to improve sorting on timestamp column.
- 'Add Branch to Version Control' will now prompt user for check-in comment.

Baseline and Model Comparison

- Number of characters displayed for a given property has been limited. (Full text can be displayed by double clicking on it.)
- Comparison of the "Default" value of a given Element improved.
- Merging of Packages and Elements that have been moved out of a baselined package improved.
- Merging of State and Activity composite diagrams from the baseline, synchronizes display properties.
- 'Find in Project Browser' option improved when the selected element is now outside of the package being compared.
- Comparison to exported XMI 2.1 will now give a clear error message stating that only XMI 1.1 can be compared.
- Element Constraints and Requirements containing line breaks will now compare.
- New Baselines will now always be created with the options 'Use DTD' and 'Generate Diagram Images' disabled.
- Compare to Model to File updated to prevent messages saying that the file

doesn't exist.

License Management

- New Windows Service based Key Store provided:
 - Shared keys for floating licenses can now be obtained from a Sparx Systems Keystore Service (SSKS).
 - The Keystore Service is a network based license management service.
 - Instances of Enterprise Architect are now able to contact an SSKS server running on a central remote machine and request keys from it.
 - Enables serial and controlled access to the underlying keystore file.
 - Provides extensive support for logging/reporting shared key usage events and statistics.
 - New capability added to Keystore manager interface to manage existing file or new Service based stores.
 - The SSKS installer is available for separate download from the Sparx Systems website.
 - The installer includes both service and file based version of the Key Store program.
 - Shared Keystore Selection dialog advises the user if it could not connect to the SSKS server with the provided details.

Miscellaneous

- Matrix Profile names can now contain a quote character.
- Saving Operation behavior improved to avert sporadic failure.
- Unnamed Ports or Parts of a Class will no longer be duplicated when creating an instance.
- Renaming of diagrams and objects from Project Browser updated to handle ' in the new name.
- WAN Optimizer now supports queries over 64k characters in length.
- Saving a Transition will no longer set the name to the Guard and Trigger information.
- Toolbox modified to ignore Ctrl+Page Up and Ctrl+Page Down while using a tool.
- Reloading of Package contents, and deleting a package updated to allow packages exceeding 1000 elements under Oracle.
- Profiled Association Classes created from the toolbox will now have the stereotype and shape script applied to both the class and association.
- Model Validation of Statechart Choice elements updated to allow one more outgoing transitions.
- SysML Flow Ports on Internal Block Diagrams updated to show direction notation.

- 'Paste as New' command will now set the created date instead of preserving the original creation date.
 - Model Validation handling of OCL rules on elements using '<' or '>' improved.
 - Model Validation improved to handle packages that haven't been previously loaded into the Project Browser.
 - Removed Zicom mentor toolbar and menu commands
 - Trigger Kind handling improved in translated versions.
 - Workspace Layouts improved to prevent problem experienced by some users that caused EA to minimize when switching layouts.
 - New connector properties that can be defined for stereotypes in profiles: `_LineStyle`, `_SourceMultiplicity`, `_TargetMultiplicity`, `_SourceAggregation`, `_TargetAggregation`, `_SourceNavigability` and `_TargetNavigability`.
 - Tasks Pane updated to always use 'Getting Started' as initial page instead of page from a loaded technology.
 - Copying of Attributes and Operations with RefGUID tagged values improved to preserve references.
-