Extended Traceability Report for Enterprise Architect

Extended Traceability Report for Enterprise Architect 1
Disclaimer 2
Dependencies 2
Overview 2
Limitations of the free version of the extension 3
Installation 4
Verifying the installation 5
Launching the Traceability Report 8
Observing the Traceability Report progress 12
The Traceability Export interface 13
Leaf node naming conventions 15
The Traceability Filter interface 17
Sample Excel output 19
Sample RTF output with automatic template selection 20
Sample PDF output using the Model Report template 22
Troubleshooting 23
Support and contact information 23

Disclaimer

Version 1.3.x of the *Extended Traceability Report* extension has been successfully tested for deployment against Enterprise Architect versions 11.1 up to 16.x (32- and 64-bit versions)

This extension, as well as the guidelines, may or may not be applicable to any later version of the tool as released by the vendor, Sparx Systems. If required, updates to this software will be made available to support future versions of Enterprise Architect.

Great care has been taken during development to use SQL statements that are supported across the common backend database platforms. Nonetheless, should a statement fail to execute correctly, please refer to the Troubleshooting part of this User Guide for assistance.

If any problems are encountered with the installation or operation of this software, please contact us through any of the channels listed at the bottom of this document.

Dependencies

The add-in depends on the following components being installed on the system:

- Interop.EA.dll (part of the standard Sparx installation files).
- Microsoft .Net Framework 4.x (if missing, the installer will detect this and offer to automatically download and install the Framework).

Overview

The 1.3.x version of the *Extended Traceability Report* software extends the default functionality provided in Enterprise Architect by the Traceability Window with the following features:

- Tracing additional non-connector-based relationships such as:
  - Element level tagged value references.
  - Attribute data type relationships.
  - Attribute level tagged value references.
  - Operation return type and argument type relationships.
  - Operation level tagged value references.
- Allowing control over the depth/level that the traceability report should include.
- Inclusion of additional connector types, such as Sequence diagram Message flows.
- Quick expansion or collapse of the tree view nodes containing the traceability results.
- Modeless User Interface, permitting the full usage of all the Enterprise Architect windows without changing the context (and content) of the Traceability Report window.
- Allowing multiple Traceability Report windows to be open at the same time.
- For every (enabled) relationship type found, providing a filter capability:
  - By target type and stereotype.
  - Additionally, for connector-based relationships, by connector type and stereotype.

  - Filter settings are remembered across invocations, as long as the Enterprise Architect instance remains active.

- Locating the related element or feature (Attribute/Operation) in the Project Browser via a single mouse-click.
- Exporting the Traceability Report to Excel, with the following options:
  - Adding the path (in Enterprise Architect) of every element or feature as a cell level comment.
  - Adding hyperlinks between all related elements included in the spreadsheet.
  - Automatically opening the Excel file after creation.
- Exporting to a document the element being traced, as well as any related elements that are also expanded in the Report (i.e., leaf level traceability elements that are not themselves expanded are not explicitly included in the generated document, but references to them are likely to be present in the content depending on the documentation generation template selected).

  Also note that the documentation template may have element and/or connector filters defined WHICH WILL LIMIT ITS CONTENTS regardless of the traceability results found.

  The following export options are available:
  - Export in RTF, PDF, DOCX or HTML output format.
  - Insertion of Linked Documents if present.
  - Automatically opening the document after creation.
  - Selection of the template to be used for documentation generation from:
    - The templates predefined in Enterprise Architect.
    - User-defined templates.
    - Automatic selection: the extension will produce a single document with Requirements generated using the standard Requirements template, Use Cases using the standard Use Case template, and all other elements using the standard Model Report template.

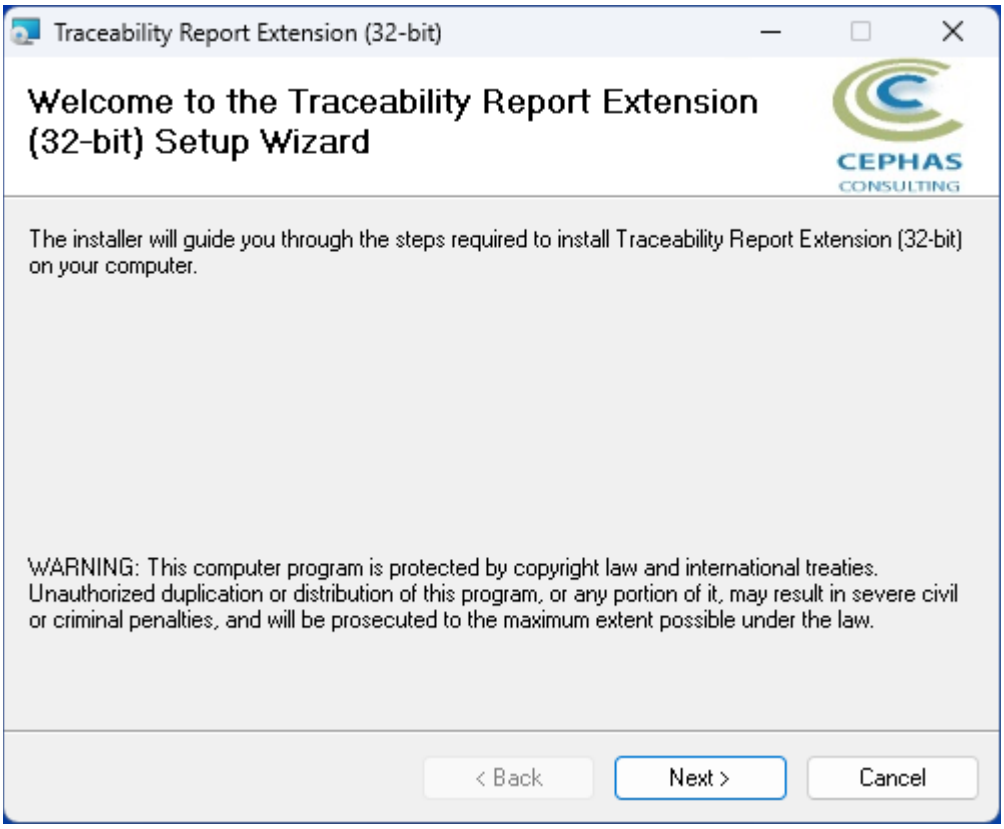
Limitations of the free version of the extension

Version 1.3.x of the add-in has the following limitations:

- The maximum depth of a Traceability Report is 5 levels.
- Custom References, which are rarely used, are not reported.
- The RTF/PDF/DOCX/HTML documentation generation is using the Enterprise Architect Document Generator Interface API and is thus subject to the limitations of that interface (e.g., its limited formatting options).
- Documentation generation templates loaded into Enterprise Architect from MDG Technologies are not included in the selection list.

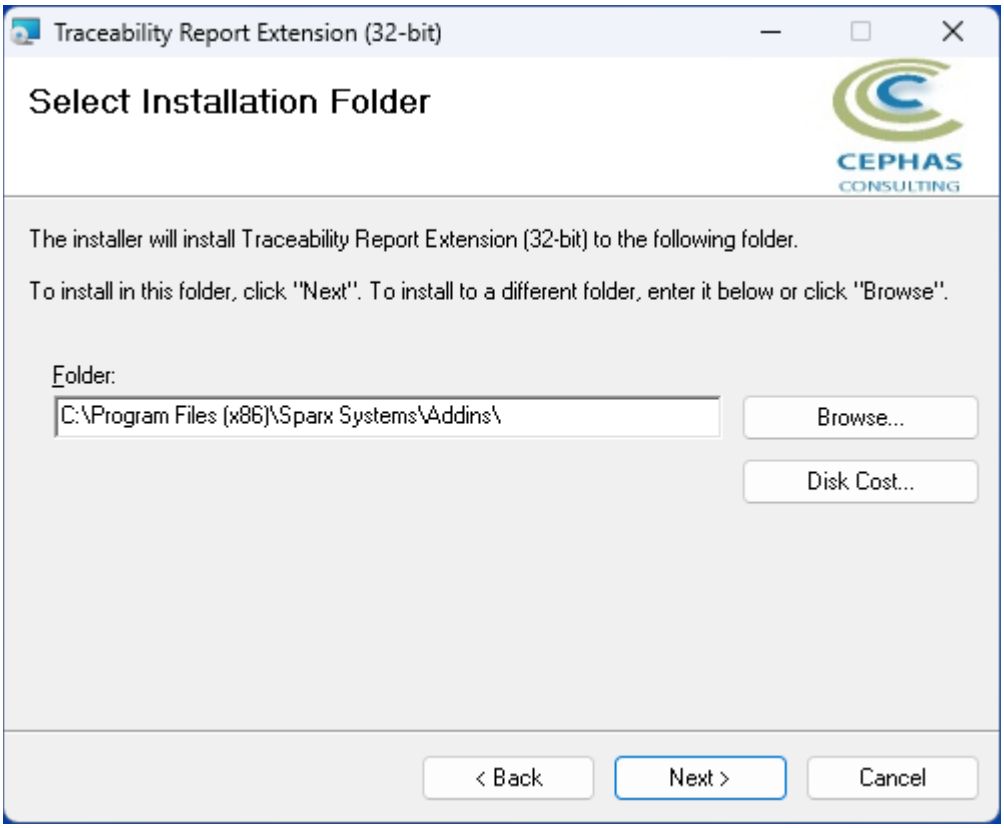
Installation

Exit any running instances of Enterprise Architect, then launch the “setup.exe” program provided as part of the installation software and follow the on-screen instructions:



The installation will attempt to update the Windows registry, so the User needs to ensure that s/he has sufficient privileges to run the setup program.

The recommended install path is to place the DLL and any supporting files in an *Addins* folder in the Sparx Systems installation directory, e.g., for the 32-bit version:



While older versions of the software can be automatically upgraded, it is recommended to remove and re-install the extension.

Should the installation fail for any reason other than insufficient User privileges, please take appropriate screenshots and email the data to the support address listed at the bottom of this document.

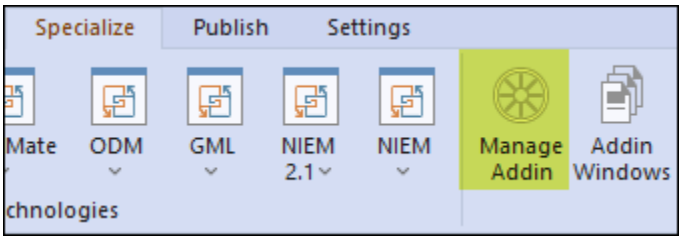
Verifying the installation

Bring up Enterprise Architect and verify that the extension is loaded.

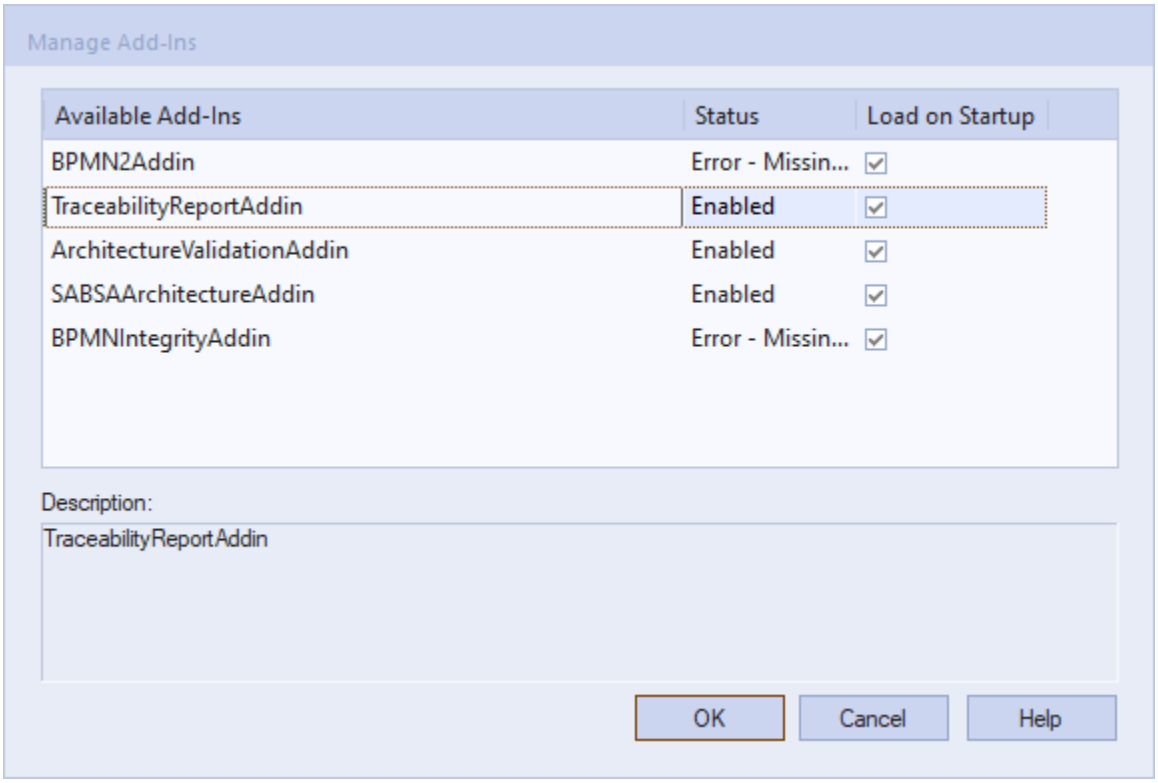
- Use the *Specialize* ribbon:

Note that the other extensions shown in the above screenshots may or may not be present, depending on your Enterprise Architect version and configuration.

Should the menu entry not be present, select the following option from the same ribbon:



And confirm that the “TraceabilityReportAddin” extension is loaded and enabled:

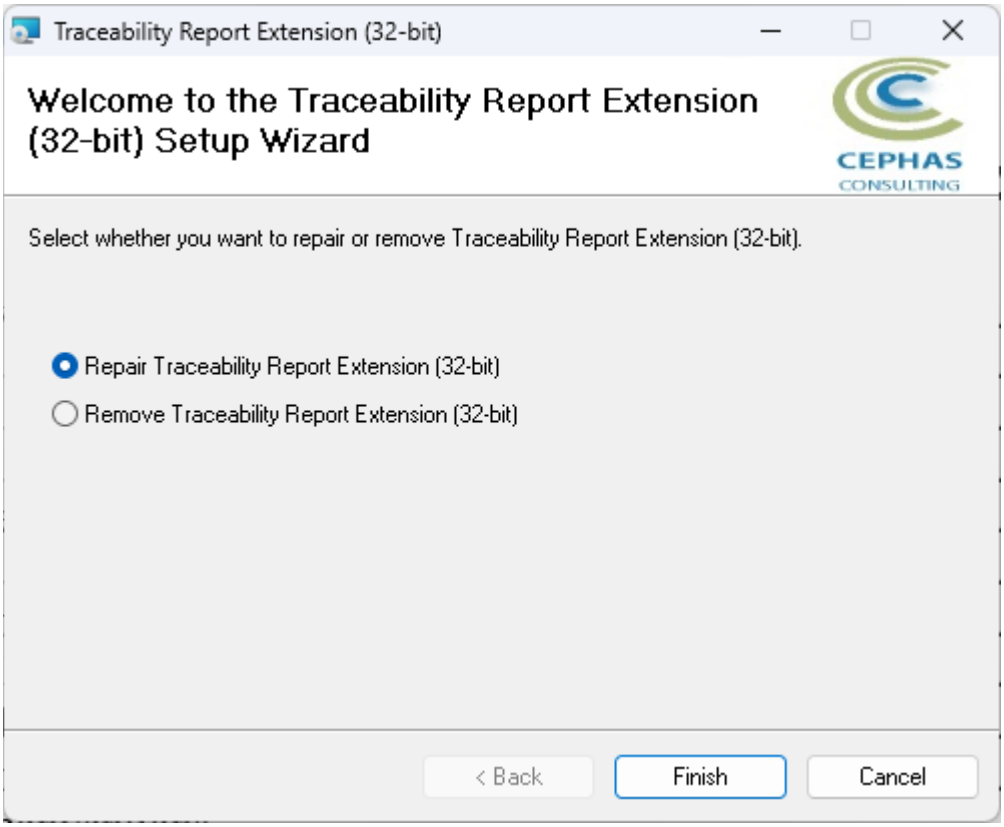


If an error status is shown, this typically means that either:

- The DLL installation process failed and EA cannot locate its reference, either in the Windows registry or in the file system.
- The installation did succeed but the DLL file was later moved or deleted.

If the “TraceabilityReportAddin” entry itself is not found, then the add-in installation did not complete successfully.

To fix an incorrect installation:

- Exit out of all instances of Enterprise Architect.
- Launch the setup process again. The installer will automatically provide a repair option. For example:
- 

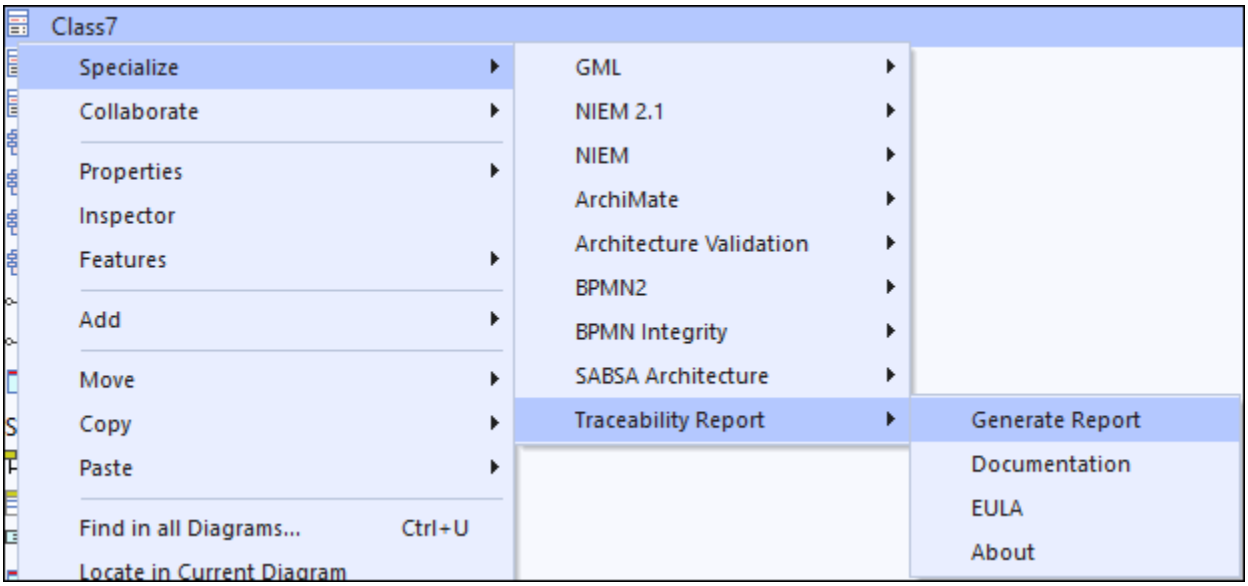
If, after the repair procedure, the extension is still not loaded correctly in Enterprise Architect, remove the program through the Windows control panel and start the installation process over.

At a minimum the following files should have been installed in the selected installation directory:

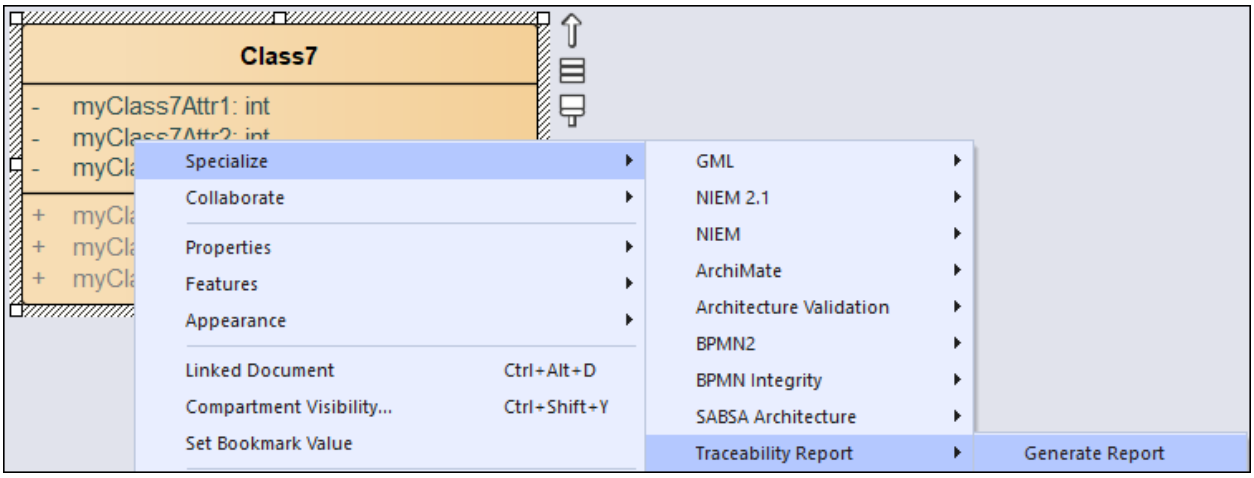
- Cephas_Software_EULA.pdf
- TraceabilityReport.pdf (this file)
- TraceabilityReportAddin.dll

Launching the Traceability Report

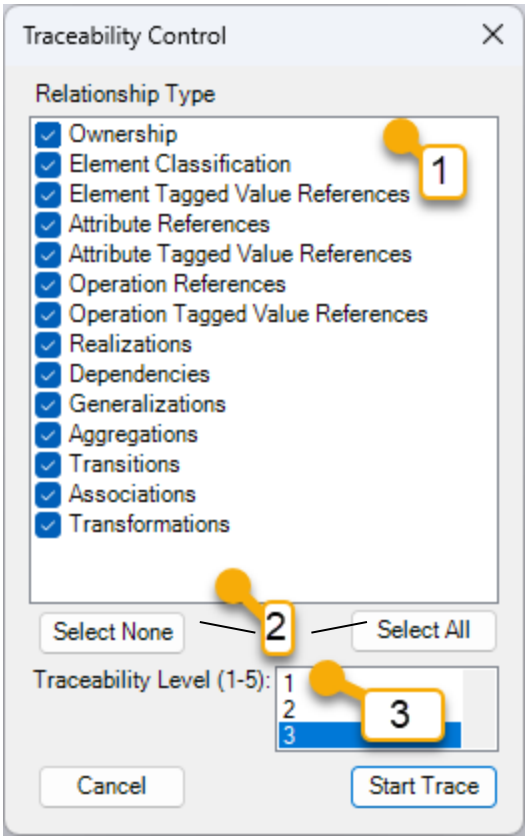
1. In the Project Browser, right click any element to be traced and select “Generate Report” from the context menu:



The same action can be performed in a diagram context:



The following dialog is then presented:



Field descriptions and User actions:

Field	Description
1.	Enable or disable the desired relationship types. See next table for a definition of each type.
2.	Optionally use these buttons to quickly select or deselect all types.
3.	Set the maximum depth of the report (between 1 and 5). Be aware that in large models selecting a higher depth level will significantly impact the time required to generate the report!

Click “Start Trace” to begin the process. To exit the interface without further action, simply click the “Cancel” button.

The following table describes each of the relationship types from the perspective of the element selected for traceability (referred to as “this element”). Note that each type is represented by two directions:

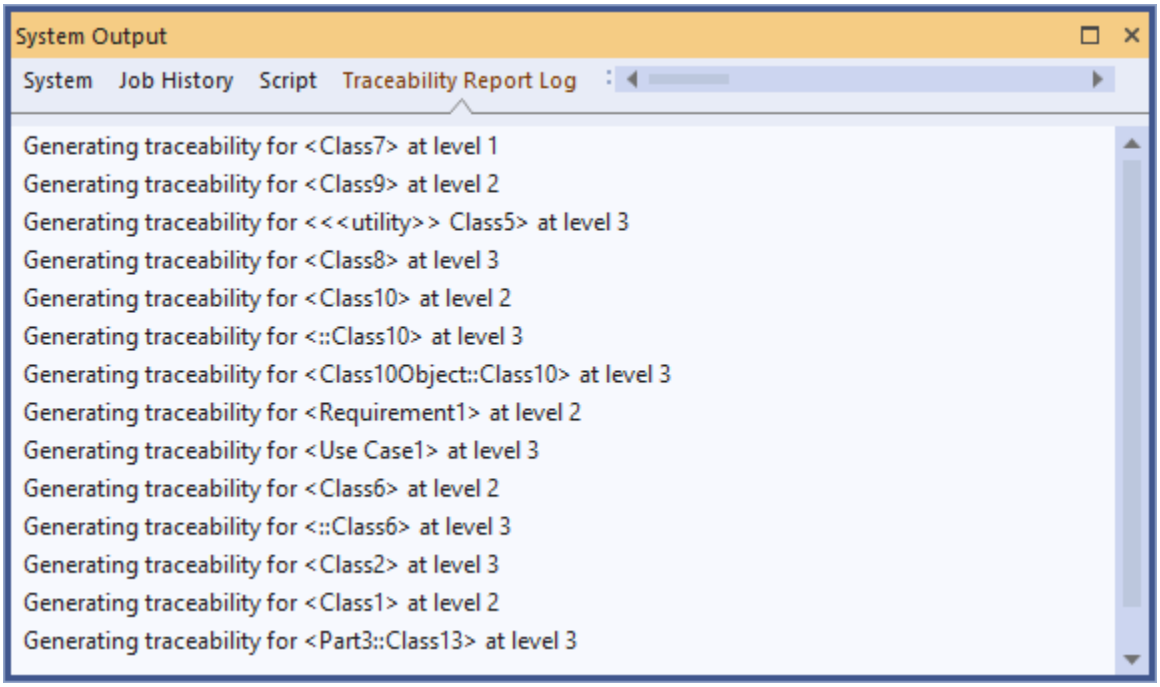
Relationship Type	Directions	Reports
Ownership	Owns	Any child elements owned by this element (i.e. for which it is the parent).
	Owned by	The parent element (if any) of this element.
Element Classification	Classifies	All the objects (instances) that are classified by this element.
	Classified by	The element/type (if any) that classifies this object/instance.
Element Tagged Value references	Owns tag referencing	The elements or features (Attributes/Operations) that are referenced by Tagged Values of this element.
	Is referenced by tag	Any elements or features in the repository which have Tagged Values that reference this element.
Attribute references	Owns Attribute with data type	The Attributes of this element for which the data type references another element.
	Is the data type of Attribute	The Attributes of other elements in the repository for which the data type is this element.
Attribute Tagged Value references	Owns Attribute tag referencing	The elements or features that are referenced by Attribute level Tagged Values of this element.
	Has Attribute referenced by tag	Any elements or features which have Tagged Values that reference an Attribute of this element.
Operation references	Owns Operation referencing	The Operations of this element for which a return type or an argument type references another element in the repository.
	Is referenced by Operation	The Operations of other elements in the repository for which a return type or argument type references this element.
Operation Tagged Value references	Owns Operation tag referencing	The elements or features that are referenced by Operation level Tagged Values of this element.
	Has Operation referenced by tag	Any elements or features which have Tagged Values that reference an Operation of this element.
Realizations (Realization	Realizes	The elements realized by this element (i.e. the Realization association sources for which

connectors)		this element is the association source).
	Realized by	The elements that realize this element (i.e. the Realization association ends for which this element is the association end).
Dependencies (Dependency and Usage connectors)	Depends on	The elements that this element depends on.
	Needed by	The elements that depend on this element.
Generalizations	Subtype of	The elements that are generalizations of this element.
	Supertype of	The elements specialized by this element.
Aggregations (includes Compositions)	Part of	The elements that aggregate (contain) this element.
	Composed of	The elements that are aggregated into (are part of) this element.
Transitions (Control Flow, Object Flow and State Flow)	Transitions to	The elements that this element transitions (i.e. passes control) to.
	Transitions from	The elements that transition (i.e. pass control to) this element.
Associations (all other connector types)	Links to	The relationship target of every connector where this element is the source.
	Links from	The relationship source of every connector where this element is the target.
Transformations	Transforms to	The elements for which this element is a transformation source.
	Transforms from	The elements for which this element is a transformation target.

Observing the Traceability Report progress

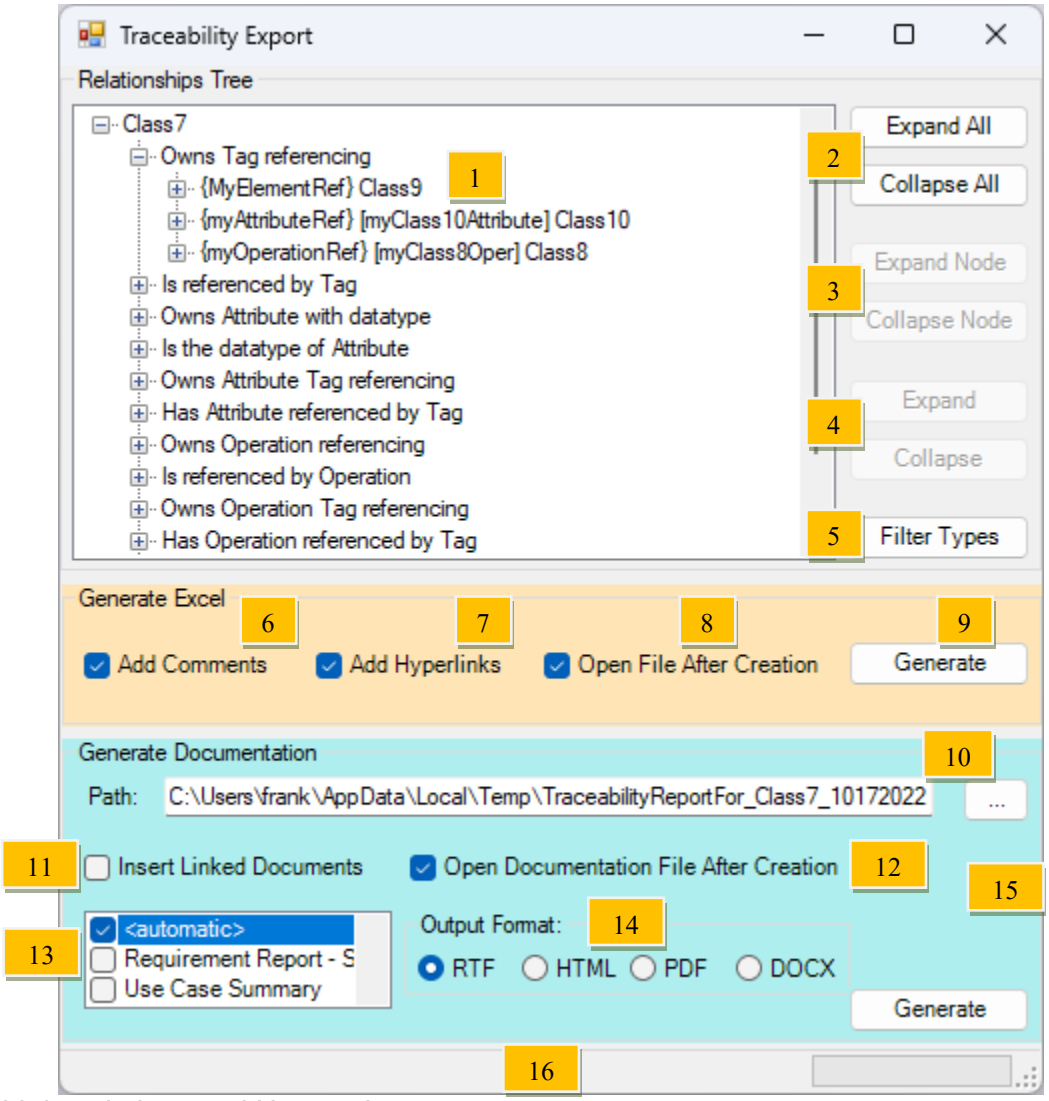
The report's progress can be viewed in the "Traceability Report Log" tab of the System Output window in Enterprise Architect. This window is automatically made visible by the extension.

A typical report generation will yield a result similar to the following:



Once the software has completed building the traceability map, the User interface shown in the next section is presented.

The Traceability Export interface



Field descriptions and User actions:

Field	Description
1.	Expand the tree view nodes as required (similar to the Enterprise Architect Traceability window interface). Each relationship type is represented by a header node. Single click on any leaf node to locate the corresponding element or feature in the Project Browser. See the Naming Conventions section below for a description of how elements and features are represented in the tree view (and in Excel).
2.	Optionally use these buttons to expand or collapse the entire tree view. Note that for a complex tree with many nodes this can take a few seconds…

3.	Optionally use these buttons to expand or collapse a single node, including all of its child nodes.
4.	Optionally use these buttons to expand or collapse a single node without including its child nodes.
5.	Refer to the Traceability Filter interface.
Excel generation panel.	
6.	When enabled: for each cell in the spreadsheet representing an element or feature in Enterprise Architect, add its (Project Browser) path as a comment.
7.	When enabled: for each cell referencing an element that is present in the spreadsheet, add a hyperlink to its location in the sheet.
8.	When enabled: automatically open the spreadsheet after creation or update.
9.	Click to generate the spreadsheet. Provide the path to the file name when prompted. If the Excel file already exists, please ensure that it is not currently open since this will prevent the completion of the update.
Documentation generation panel. Please note that the generated document reports the element being traced as well as all the related elements found as part of the traceability drilldown. Its actual content is totally dependent on the selected template which may, or may not, include all of the relationships found!	
10.	Enter the path to the document file to be created (or updated) either manually, by paste, or by clicking the ellipsis (…). The default path is set to the “temp” directory of your system.
11.	When enabled: includes the Linked Document content (if any) of the elements being reported. Most RTF templates have this option set by default, so for those templates it should not be enabled here; otherwise the Linked Document content will appear twice.
12.	Open the generated document (RTF, PDF, DOCX or HTML) after creation or update.
13.	Select one of the RTF templates to generate the report with. The default <automatic> option will produce a single document with Requirements (if any) generated using the *Requirements Report – Summary* template, Use Cases (if any) using the *Use Case Summary* template, and all other elements using the standard *Model Report* template.
14.	Select the output format of the document (RTF, PDF or HTML). Changing the option automatically updates the extension of the specified file path (if any).
15.	Click to generate the document.
For both generation options:	

16.	Watch the message and progress bar for status updates!

To exit this User interface, simply click the X (close) button in the top right hand corner of the window.

Leaf node naming conventions

For elements:

<<Optional element stereotype>> Element-name

For Objects (instances), Ports and Parts:

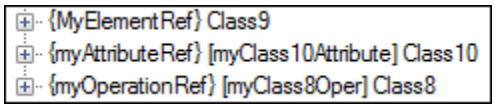
Name-of-object :: Name-of-classifier

Example: Class10Object::Class10

For element Tagged Value references:

{Name-of-tag} [Optional-name-of-feature] <<Optional element stereotype>> Element-name

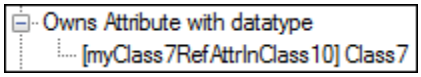
Example:



For Attribute (data type) references:

[Name-of-attribute] <<Optional element stereotype>> Element-name

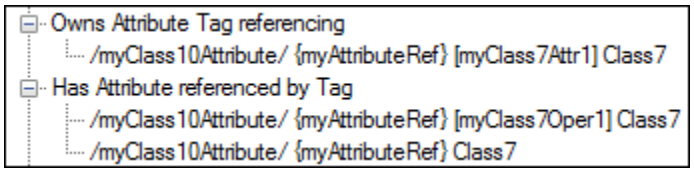
Example:



For Attribute Tagged Value references:

/Name-of-owned-attribute/ {Name-of-tag} [Optional-name-of-referenced-feature] <<Optional element stereotype>> Element-name

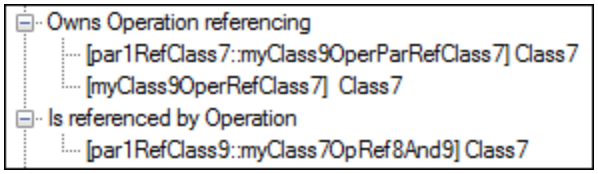
Examples:



For Operation (return type or parameter type) references:

[Optional-name-of-parameter :: Name-of-operation] <<Optional element stereotype>> Element-name

Examples:

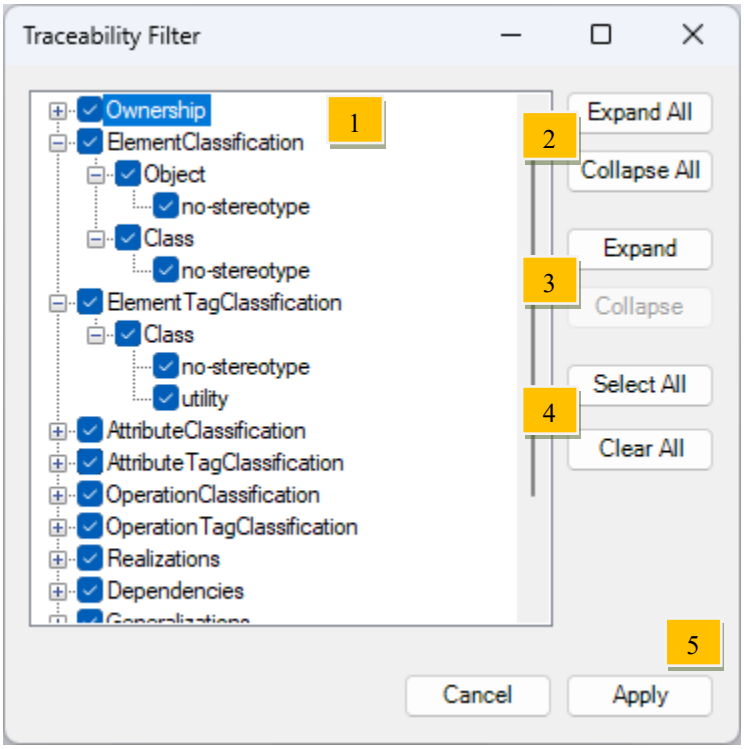


For Operation Tagged Value references:

/Name-of-owned-operation/ {Name-of-tag} [Optional-name-of-referenced-feature] <<Optional element stereotype>> Element-name

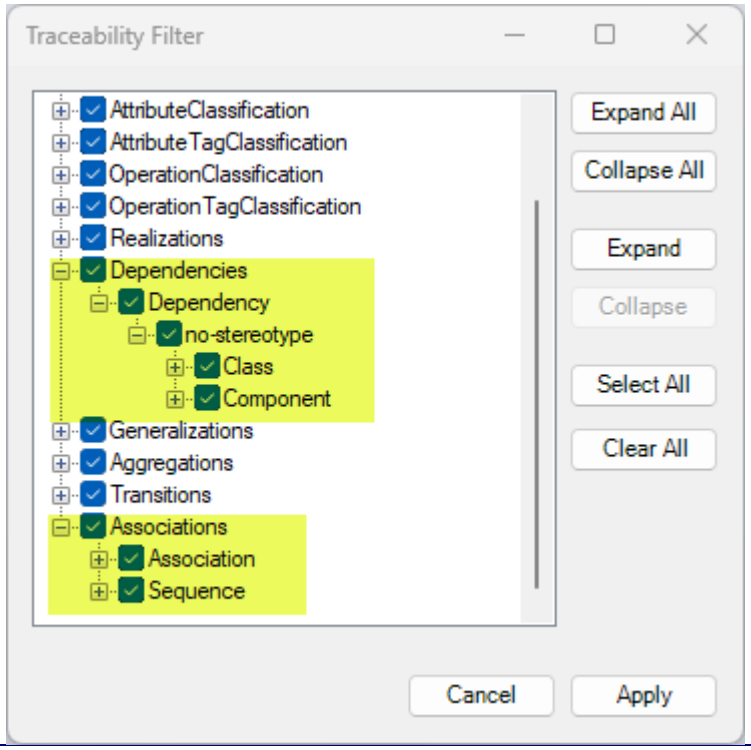
Example:

The Traceability Filter interface



Field descriptions and User actions:

Field	Description
1.	Expand the tree view nodes as required. Each enabled relationship type for which traceable elements were found is represented by a separate node. The node then lists every related element type/stereotype combination found across the relationship tree. Note that a related element can be on the source or the destination side of a connector. Disable (i.e., filter out) the entire relationship, or one or more of its related types and/or related stereotypes. For relationship types that are connector based (e.g., Dependencies), the filter can also be applied at the specific connector type and stereotype level. For example:

	
2.	Optionally use these buttons to expand or collapse the entire tree view.
3.	Optionally use these buttons to expand or collapse a single node.
4.	Optionally use these buttons to quickly select or deselect all target types.
5.	Click “Apply” to filter the current result set based on your selections (the content of the Traceability Export window is automatically refreshed), or “Cancel” to leave the current filter settings unchanged.

Sample Excel output

	A	B	C	D	E	F	G	H	I	J	K	L
1	**Class7**											
2	Owns	Owned by	Classifies	Classified by	Realizes	Realized by	Depends on	Needed by	Part of	Composed of	Association to	
3					Requirement1		Class6				Class1	
4											Class2	
5	**Requirement1**											
6	Owns	Owned by	Classifies	Classified by	Realizes	Realized by	Depends on	Needed by	Part of	Composed of		
7						Class7						
8						Use Case1						
9	**Use Case1**											
10	Owns	Owned by	Classifies	Classified by	Realizes	Realized by	Depends on	Needed by	Part of	Composed of		
11					Requirement1	<<executable>> Component3						
12	**Class6**											
13	Owns	Owned by	Classifies	Classified by	Realizes	Realized by	Depends on	Needed by	Part of	Composed of		
14			::Class6					Class7		Class2		
15	**::Class6**											
16	Owns	Owned by	Classifies	Classified by	Realizes	Realized by	Depends on	Needed by	Part of	Composed of	Sequence from	
17				Class6							::Class4	
18	**Class2**											
19	Owns	Owned by	Classifies	Classified by	Realizes	Realized by	Depends on	Needed by	Part of	Composed of	Association to	Association from
20			Object1::Class2		Requirement2		<<utility>> Class4	Component1	Class6	Class1	Class2	Class2
21			::Class2		Interface1						<<internal worker>> Class3	Class7
22	**Class1**											
23	Owns	Owned by	Classifies	Classified by	Realizes	Realized by	Depends on	Needed by	Part of	Composed of	Association from	
24	Part3::Class13								Class2		Class7	
25	**Part3::Class13**											
26	Owns	Owned by	Classifies	Classified by	Realizes	Realized by	Depends on	Needed by	Part of	Composed of		
27		Class1		Class13								

Sample RTF output with automatic template selection

Table of Contents

Requirement1 3
Use Case1 4
Class5 5
Class1 5
Class10 5
Class2 6
Class6 7
Class7 8
Class7 diagram 9
Class8 9
Class9 10

Requirement1

Requirement Report - Summary

Requirement in package 'Requirements'

Requirement1
Version 1.0 Phase 1.0 Proposed
Frank Truyen created on 11/20/2012. Last modified 11/20/2012

Use Case1

Use Case Summary

UseCase in package 'Use Cases'

Documentation for Use Case1

Use Case1
Version 1.0 Phase 1.0 Proposed
Frank Truyen created on 11/20/2012. Last modified 11/20/2012
Alias UC100

Basic Path
Basic Path:

Constraints

Pre-condition1
Type: Pre-condition
Status: Approved Weight: 0

Class5

Model Report

Class «utility» in package 'Traceability'

Class5
Version 1.0 Phase 1.0 Proposed
Frank Truyen created on 11/8/2012. Last modified 11/8/2012

INCOMING STRUCTURAL RELATIONSHIPS

Generalization from Class2 to «utility» Class5

[Direction is 'Source -> Destination'.]

Class1

Class in package 'Traceability'

Class1
Version 1.0 Phase 1.0 Proposed
Frank Truyen created on 11/7/2012. Last modified 11/8/2012

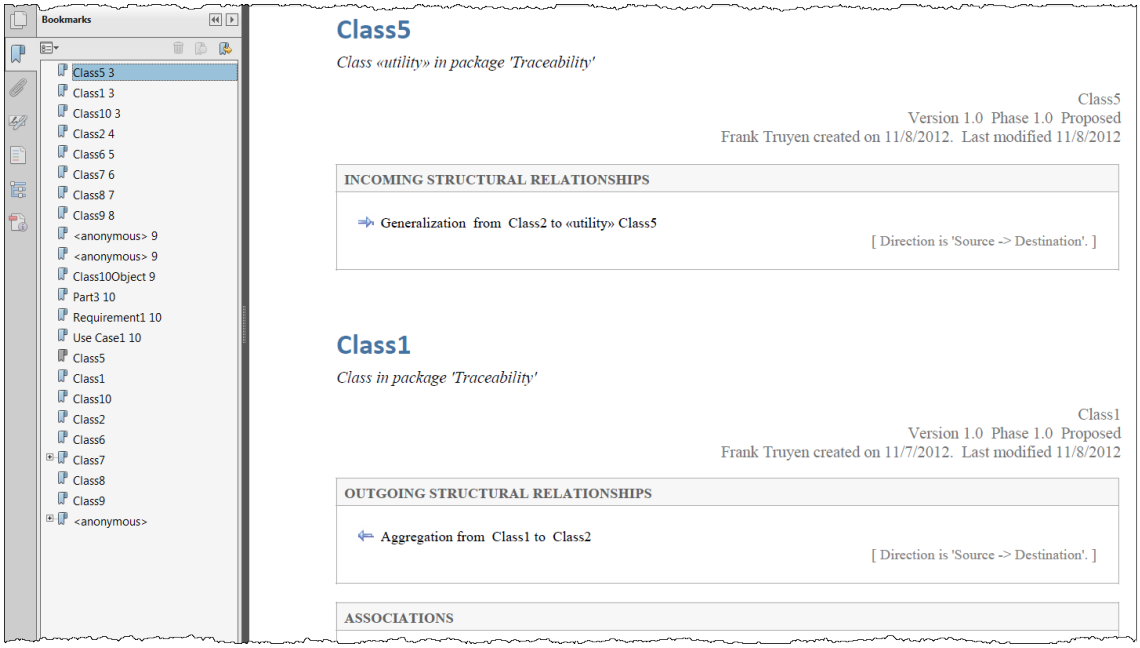
OUTGOING STRUCTURAL RELATIONSHIPS

Aggregation from Class1 to Class2

[Direction is 'Source -> Destination'.]

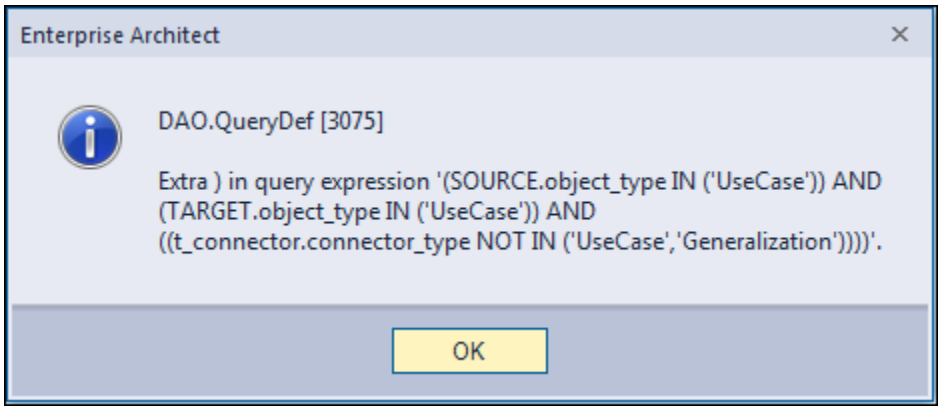
ASSOCIATIONS

Sample PDF output using the Model Report template



Troubleshooting

Should a SQL statement fail to execute properly, Enterprise Architect will display an error message dialog similar to this:



This type of error is not relayed back to the API function call, which is unaware that a problem occurred.

Please follow this procedure:

- Take a screenshot of the error message.
- Locate the DBError.txt file in %APPDATA%\Sparx Systems\EA and include it in your message.
- Before dismissing the dialog box, look at the System Output window to help determine the execution context at the time of the failure.
- Also provide your database type (Microsoft Access, SQL Server, Oracle, etc.) and version number.

Support and contact information

Use the contact information below for any installation or runtime issues with the add-in. Feature requests or suggestions for improvement are also welcome!

Contact: Frank Truyen

Email: contact@enterprisemodelingsolutions.com

Phone: 208-462-4863.

Web: enterprisemodelingsolutions.com