

Using the XML Mapping Editor

This chapter covers the following topics:

- Starting the XML Mapping Editor
- The XML Mapping Editor Pages
- XML Schema Export
- XML Mapping Editor Settings
- Removing Unused Namespaces

See also *Using the XML Tester*.

Starting the XML Mapping Editor

➤ **To start the XML Mapping Editor**

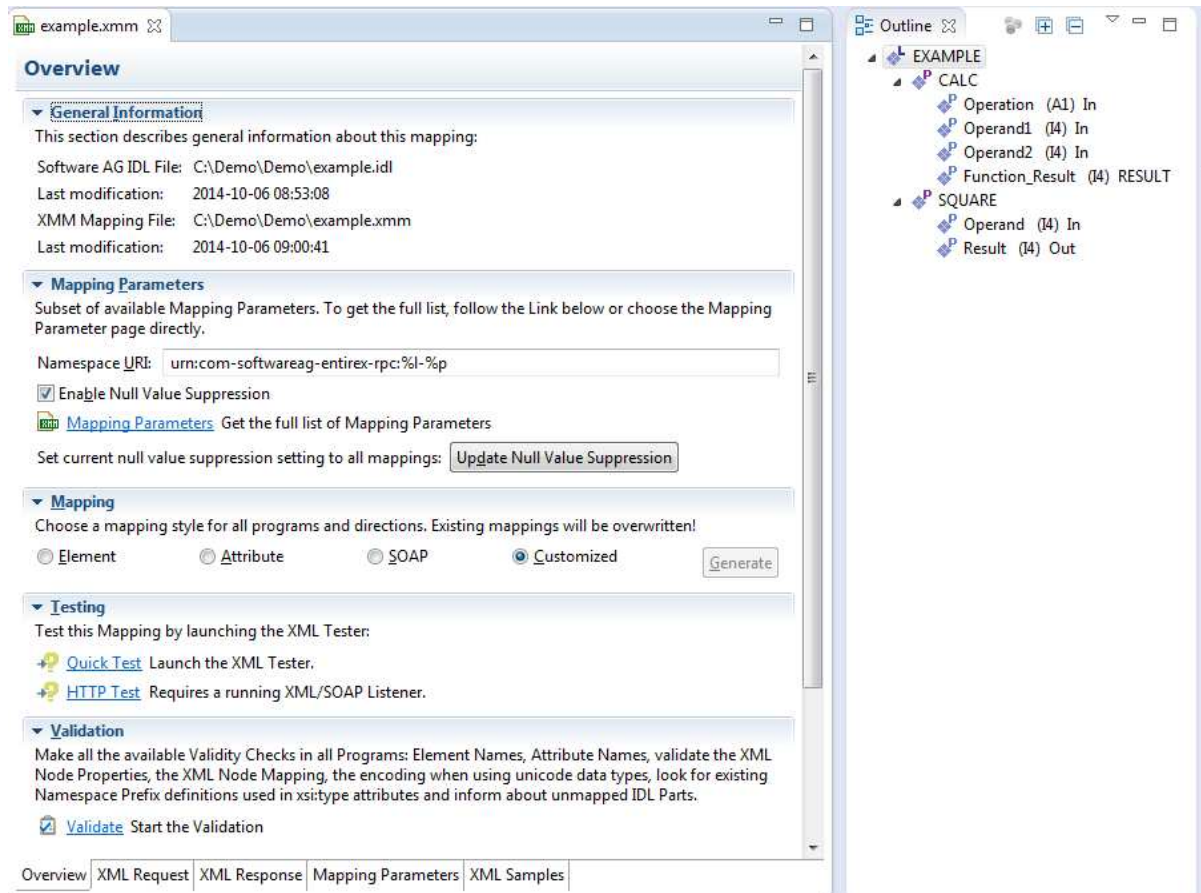
In the *EntireX Workbench*

- Double click on an XMM file.

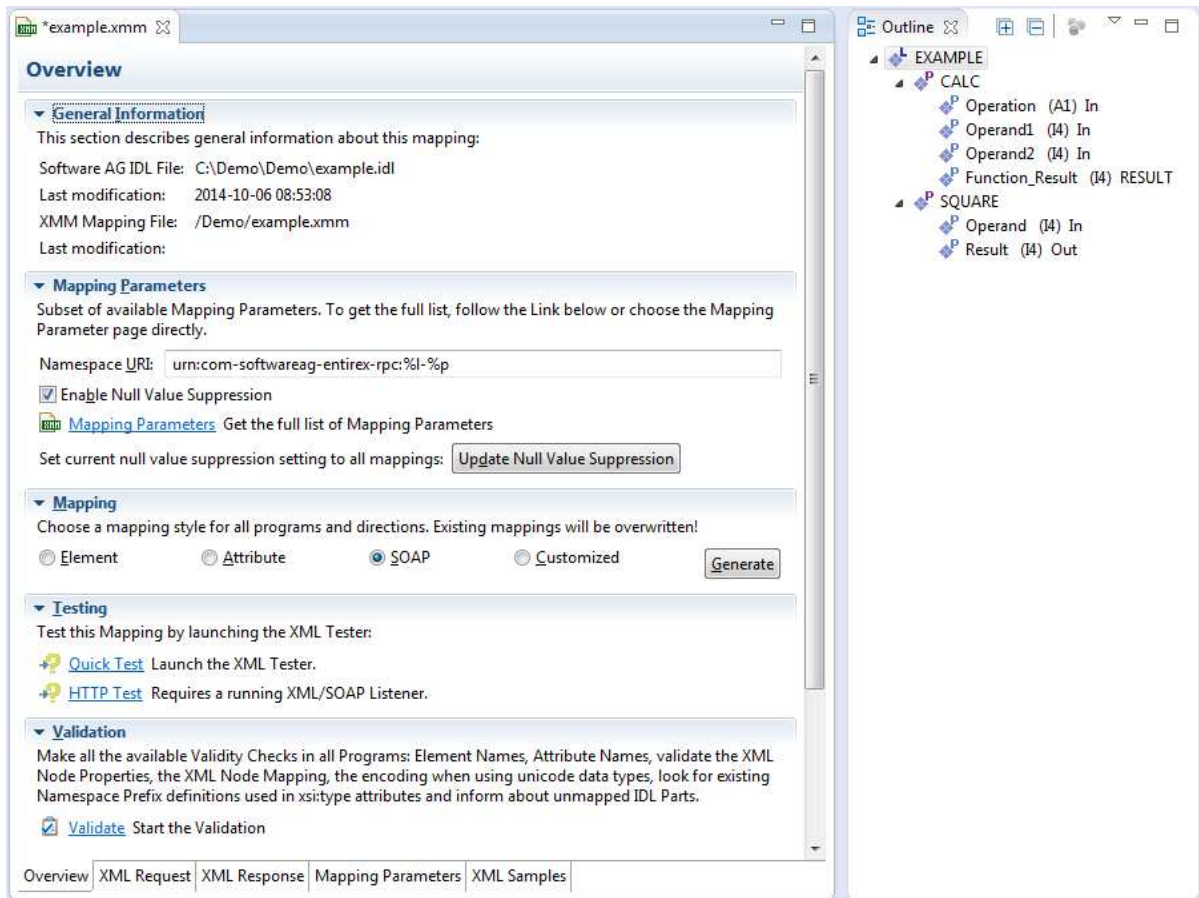
Or:

Select an IDL or XMM file and choose **Open With > EntireX XML Mapping Editor**.

If the XMM file for the selected IDL file exists, the XML Mapping Editor loads this. The **Generate** button is disabled because a mapping exists.



If no XMM file exists for the IDL file, the mapping style is set to "SOAP" as default value. Choose the required mapping style and press **Generate**.



➤ To close the XML Mapping Editor

- From the **File** menu, choose **Close**.

Or:

Click the **Close** icon in the editor title bar.

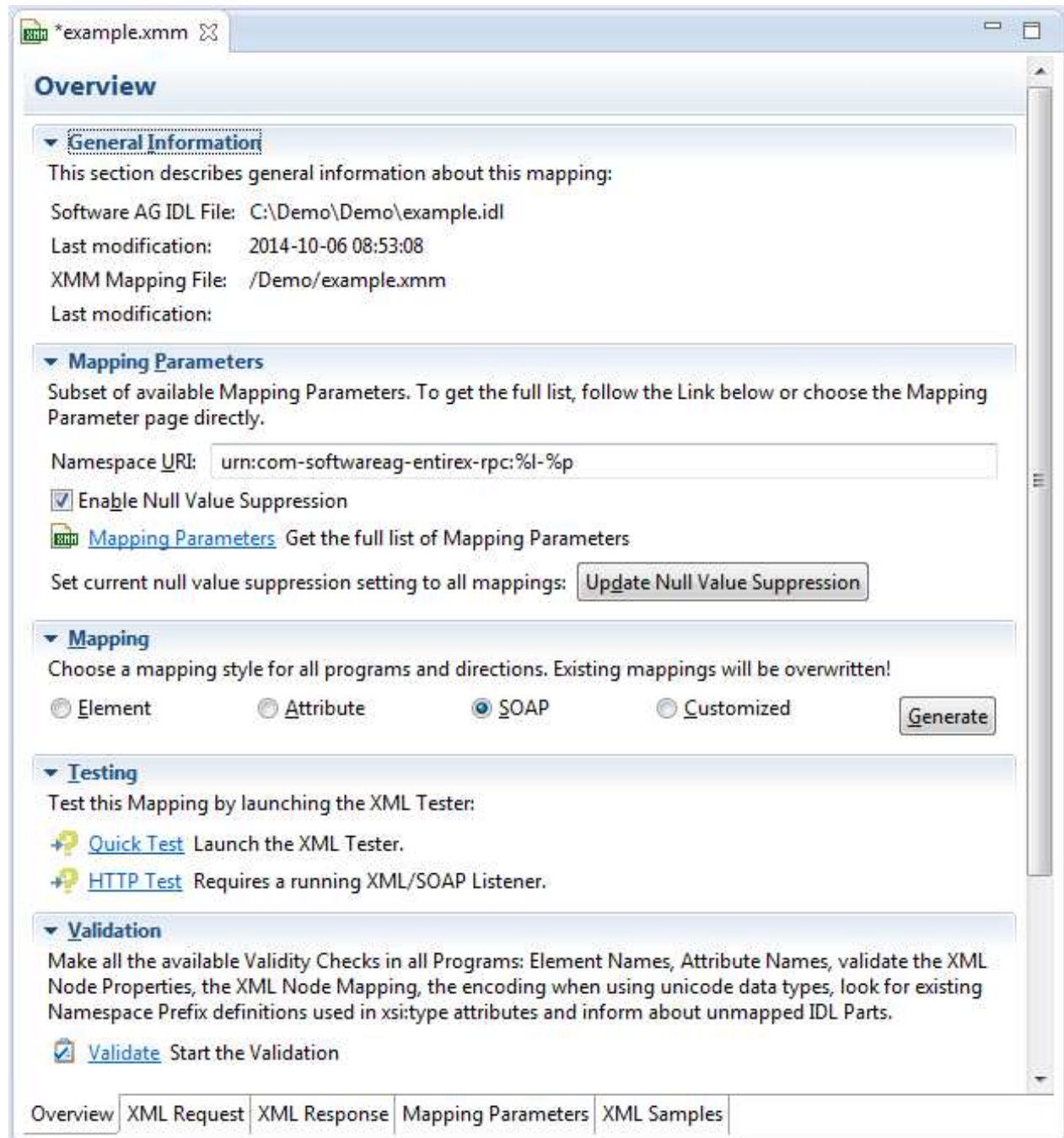
If the current XML mapping has not yet been saved, a dialog box appears from which the close operation can be stopped or the IDL-XML mapping can be saved. An XML mapping that is not clean is indicated with a leading asterisk in the file name, displayed in the editor title.

The XML Mapping Editor Pages

- Overview Page
- XML Request Page
- XML Response Page
- Mapping Parameters Page
- XML Sample Documents Page

Overview Page

The **Overview Page** is the central page and provides sections for file information, the most important mapping parameters, mapping generation, testing and validation.



These sections are:

- **General Information**

This section describes the names and locations of the selected XML mapping (XMM) file and the related IDL file.

- **Mapping Parameters**

Provides the **Namespace** and **Null Value Suppression** controls and contains a link to the **Mapping Parameters Page**.

Note:

Press **Update Null Value Suppression** to apply the mapping parameters (see *Mapping Parameters Page*) to a generated mapping.

- **Mapping**

Starts the mapping generation for all programs.

Note:

Existing mappings will be overwritten!

- **Testing**

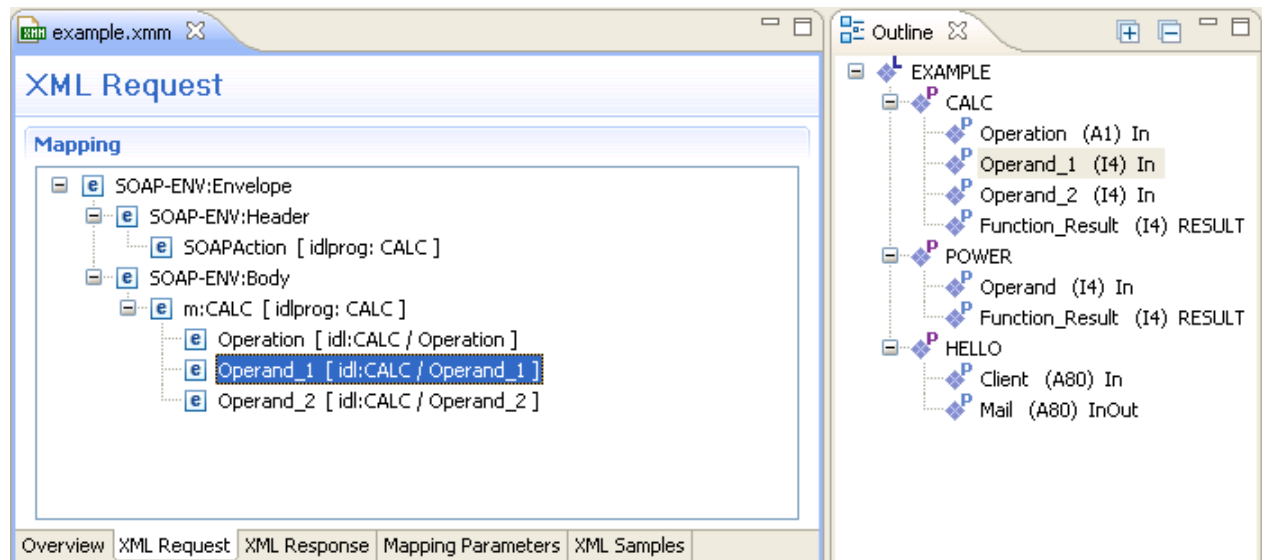
Allows you to launch the XML Tester in Quick mode or HTTP mode.

- **Validation**

Performs validity checks in all programs and reports the result in a new dialog. Detected problems will be displaced in the **Problems** view.

XML Request Page

This page contains the **Mapping** tree of the XML Request, which is linked together by mapping paths with the IDL tree in the **Outline** view.



The XML Request shows the XML tree loaded or created so far for the selected IDL program. The XML tree can be modified in various ways. See *IDL to XML Mapping with the XML Mapping Editor*.

The **Outline** view shows the IDL tree of the currently selected IDL file. The IDL tree display has the same functionality as the IDL Editor's tree display; however, due to the fact that every IDL program must be separately mapped, only one of the IDL program tree nodes can be expanded.

Use the right mouse button to display the context menu. Double-click on a tree node to display detailed information on that node. The following keyboard options are also available in the XML request mapping area:

Key	Description
A or SHIFT+INS	Create new parameter node after currently selected one.
D or DELETE	Delete the selected IDL tree node.
F	Find IDL node with given name pattern.
I or ALT+INS	Create a new group parameter node.
L	Create new IDL library node.
M	Find mapped XML node for selected IDL tree node.
N or INS	Create new parameter node.
O or CTRL+INS	Create new RPC program node.
P or ALT+ENTER	Open property dialog on selected IDL tree node.
R	Rename the selected IDL tree node.
F3	Follow-up of M, find next mapped XML node for selected IDL tree node.

XML Node Properties Dialog

The XML Node Properties dialog contains the XML node details, the bottom panel contains namespace settings.

XML Mapping Editor

XML Node Properties
Properties of Operation [idl:CALC / Operation]

Element Name:

Mapped to:

Program Mapping:

Format:

Value Length:

Default Value:

Namespace Prefix:

Min. Occurrence:

Max. Occurrence:

Null Value Suppression:

Nilable:

Default URI:

Table of defined Namespace definitions:

Prefix	Namespace	Default

Buttons: Insert..., Edit..., Remove, Default, OK, Cancel

➤ **To open the XML Node Properties Dialog**

- Double-click on an XML node in the tree display.

Or:

Choose the Properties menu item in the context menus.

Or:

Select an XML tree node and press Enter.

The XML Node Properties can be modified. The node details panel consists of two subpanels.

The upper panel contains the following:

Item	Description																																
Element or attribute name	Name of the element or attribute.																																
Mapped to	The IDL mapping link in the full path name format. For error (fault) trees, context information.																																
Program Mapping	<p>Combo Box</p> <p>No No Program Mapping</p> <p>Element value The value of this node identifies the program</p> <p>Element name The name of this node identifies the program</p>																																
Format	<table border="0"> <thead> <tr> <th data-bbox="373 887 858 918">Format</th> <th data-bbox="880 887 1254 918">IDL</th> </tr> </thead> <tbody> <tr> <td data-bbox="373 936 858 967">string</td> <td data-bbox="880 936 1254 967">A, AV, K, KV</td> </tr> <tr> <td data-bbox="373 990 858 1021">-</td> <td data-bbox="880 990 1254 1021">G</td> </tr> <tr> <td data-bbox="373 1043 858 1075">I1 (deprecated)</td> <td data-bbox="880 1043 1254 1075">I1</td> </tr> <tr> <td data-bbox="373 1097 858 1128">I2 (deprecated)</td> <td data-bbox="880 1097 1254 1128">I2</td> </tr> <tr> <td data-bbox="373 1151 858 1182">I4 (deprecated)</td> <td data-bbox="880 1151 1254 1182">I4</td> </tr> <tr> <td data-bbox="373 1205 858 1236">I8 (deprecated)</td> <td data-bbox="880 1205 1254 1236">I4</td> </tr> <tr> <td data-bbox="373 1258 858 1290">integer</td> <td data-bbox="880 1258 1254 1290">I1, I2, I4</td> </tr> <tr> <td data-bbox="373 1312 858 1344">date</td> <td data-bbox="880 1312 1254 1344">D</td> </tr> <tr> <td data-bbox="373 1366 858 1397">time (deprecated)</td> <td data-bbox="880 1366 1254 1397">T</td> </tr> <tr> <td data-bbox="373 1420 858 1451">dateTime</td> <td data-bbox="880 1420 1254 1451">T</td> </tr> <tr> <td data-bbox="373 1473 858 1505">float</td> <td data-bbox="880 1473 1254 1505">F4, F8</td> </tr> <tr> <td data-bbox="373 1527 858 1559">Boolean</td> <td data-bbox="880 1527 1254 1559">L</td> </tr> <tr> <td data-bbox="373 1581 858 1612">binary</td> <td data-bbox="880 1581 1254 1612">B, BV</td> </tr> <tr> <td data-bbox="373 1635 858 1666">number</td> <td data-bbox="880 1635 1254 1666">N, P, NU, PU</td> </tr> <tr> <td colspan="2" data-bbox="373 1688 1254 1729">For Tamino/special nodes: Describes the data type of the Tamino node (for example, integer, string).</td> </tr> </tbody> </table>	Format	IDL	string	A, AV, K, KV	-	G	I1 (deprecated)	I1	I2 (deprecated)	I2	I4 (deprecated)	I4	I8 (deprecated)	I4	integer	I1, I2, I4	date	D	time (deprecated)	T	dateTime	T	float	F4, F8	Boolean	L	binary	B, BV	number	N, P, NU, PU	For Tamino/special nodes: Describes the data type of the Tamino node (for example, integer, string).	
Format	IDL																																
string	A, AV, K, KV																																
-	G																																
I1 (deprecated)	I1																																
I2 (deprecated)	I2																																
I4 (deprecated)	I4																																
I8 (deprecated)	I4																																
integer	I1, I2, I4																																
date	D																																
time (deprecated)	T																																
dateTime	T																																
float	F4, F8																																
Boolean	L																																
binary	B, BV																																
number	N, P, NU, PU																																
For Tamino/special nodes: Describes the data type of the Tamino node (for example, integer, string).																																	

Item	Description
Value Length	<p>The length parameter contains the length or precision of the data type, i.e.</p> <ul style="list-style-type: none"> ● length of string for alphanumeric ● length of field in bytes for binary ● length of string representation for floating point ● length of string representation for integers ● length of string for Kanji ● length of bit vector for logical (Boolean) <p>Length V is legal for string types. It denotes variable-length strings, e.g. alphanumeric or Kanji.</p> <p>For Tamino/special nodes: Length of the storage reserved for the data must match format.</p>
Default Value	<p>Must match the data type of the node, e.g. do not use alphanumeric defaults for integer nodes. Defaults that cannot be interpreted by the runtime component are ignored. See also <i>Assigning Default Values</i>.</p>
Namespace Prefix	<p>Contains the namespace prefix to the tag name.</p>
Min. Occurrence	<p>The minimum and maximum occurrence properties are numeric values greater than or equal to zero. They describe the number of allowed occurrences of the node in the incoming XML, and the number of occurrences to be generated in the outgoing XML. The minimum occurrences value must be less than or equal to the maximum occurrences value. Incorrect settings will be found in the validity check Validate XML node properties. If Min. Occurrence is greater than 0 for incoming documents, the corresponding attribute or element is required, i.e. must be set.</p>
Max. Occurrence	<p>See Min. Occurrence.</p>

Item	Description																												
Null Value Suppression (NVS)	<p>Specify suppression of null values (NVS), that is, whether empty elements or attributes may be omitted in the outgoing (generated) XML documents. Mainly determined by Min.Occurrences and Max. Occurrences as follows:</p> <ul style="list-style-type: none"> ● For min = 0, max = 0, there is no NVS. ● For min = 0, max = 1, the element / attribute may be omitted if empty. ● For min = 0, max > 1, you can omit all empty elements or just omit empty elements at the end of the sequence (trimming). ● For min = 1, max = 1, there is no NVS. ● For min = 1, max > 1, you can omit all empty elements or just omit empty elements at the end of the sequence (trimming). ● For $0 < \text{min} < \text{max}$, you can omit all empty elements or just omit empty elements at the end of the sequence (trimming). <p>It is also possible to suppress an attribute (independent of its value) if the associated element has a null value and null value suppression is enabled; to do this choose Depends on element in property Null Value Suppression.</p> <p>For the data types, the following null values are defined:</p> <table border="0" style="width: 100%;"> <thead> <tr> <th style="text-align: left;">Format</th> <th style="text-align: left;">NVS Default</th> </tr> </thead> <tbody> <tr> <td>string</td> <td>" " (empty string)</td> </tr> <tr> <td>-</td> <td>" " (empty string)</td> </tr> <tr> <td>I1 (deprecated)</td> <td>0</td> </tr> <tr> <td>I2 (deprecated)</td> <td>0</td> </tr> <tr> <td>I4 (deprecated)</td> <td>0</td> </tr> <tr> <td>integer</td> <td>0</td> </tr> <tr> <td>date</td> <td></td> </tr> <tr> <td>time (deprecated)</td> <td></td> </tr> <tr> <td>dateTime</td> <td></td> </tr> <tr> <td>float</td> <td>0.0</td> </tr> <tr> <td>Boolean</td> <td>false</td> </tr> <tr> <td>binary</td> <td>Only BV with length =0; Only binaries corresponding to IDL parameter with type BV have an NVS default [element/attribute with data length=0].</td> </tr> <tr> <td>number</td> <td>0.0</td> </tr> </tbody> </table>	Format	NVS Default	string	" " (empty string)	-	" " (empty string)	I1 (deprecated)	0	I2 (deprecated)	0	I4 (deprecated)	0	integer	0	date		time (deprecated)		dateTime		float	0.0	Boolean	false	binary	Only BV with length =0; Only binaries corresponding to IDL parameter with type BV have an NVS default [element/attribute with data length=0].	number	0.0
Format	NVS Default																												
string	" " (empty string)																												
-	" " (empty string)																												
I1 (deprecated)	0																												
I2 (deprecated)	0																												
I4 (deprecated)	0																												
integer	0																												
date																													
time (deprecated)																													
dateTime																													
float	0.0																												
Boolean	false																												
binary	Only BV with length =0; Only binaries corresponding to IDL parameter with type BV have an NVS default [element/attribute with data length=0].																												
number	0.0																												
Null Value	If NVS is switched on, this value is compared with the values of the XML document and it is decided whether the values need to be transmitted. The defaults are in the table for NullValue Suppression (NVS).																												

Item	Description																																																												
Nillable	If NVS is set to "No Suppression", the Nillable option is available in the dialog. If the option is selected, an empty element is represented as an empty-element tag with attribute <code>xsi:nil="true"</code> . This means that if a nillable group element only has elements without value and without attributes, only the enclosing group tag with attribute <code>xsi:nil="true"</code> is displayed.																																																												
Time Pattern	<p>Optional for date, time (deprecated) and dateTime.</p> <p>The default patterns conform to the XSD Schema 2001 specification:</p> <p>date: yyyy-MM-dd, for example 2003-04-15 dateTime: yyyy-MM-dd 'T' HH:mm:ss, for example 2003-04-15T18:48:23</p> <p>Value for user-defined time pattern:</p> <table border="1"> <thead> <tr> <th>Letter</th> <th>Date or Time Component</th> <th>Examples</th> </tr> </thead> <tbody> <tr> <td>G</td> <td>Era Designator</td> <td>AD</td> </tr> <tr> <td>y</td> <td>Year</td> <td>1996;96</td> </tr> <tr> <td>M</td> <td>Month in year</td> <td>July; Jul; 07</td> </tr> <tr> <td>w</td> <td>Week in year</td> <td>27</td> </tr> <tr> <td>W</td> <td>Week in month</td> <td>2</td> </tr> <tr> <td>D</td> <td>Day in year</td> <td>189</td> </tr> <tr> <td>d</td> <td>Day in month</td> <td>10</td> </tr> <tr> <td>F</td> <td>Day of week in month</td> <td>2</td> </tr> <tr> <td>E</td> <td>Day in week</td> <td>Tuesday; Tue</td> </tr> <tr> <td>a</td> <td>AM/PM marker</td> <td>PM</td> </tr> <tr> <td>H</td> <td>Hour in day (0-23)</td> <td>0</td> </tr> <tr> <td>k</td> <td>Hour in day (1-24)</td> <td>24</td> </tr> <tr> <td>K</td> <td>Hour in AM/PM (0-11)</td> <td>0</td> </tr> <tr> <td>h</td> <td>Hour in AM/PM (1-12)</td> <td>12</td> </tr> <tr> <td>m</td> <td>Minute in hour</td> <td>30</td> </tr> <tr> <td>s</td> <td>Second in minute</td> <td>55</td> </tr> <tr> <td>S</td> <td>Millisecond</td> <td>978</td> </tr> <tr> <td>z</td> <td>General Time Zone</td> <td>Pacific Standard Time: PST; GMT-08:00</td> </tr> <tr> <td>Z</td> <td>RFC 822 Time Zone</td> <td>-0800</td> </tr> </tbody> </table>	Letter	Date or Time Component	Examples	G	Era Designator	AD	y	Year	1996;96	M	Month in year	July; Jul; 07	w	Week in year	27	W	Week in month	2	D	Day in year	189	d	Day in month	10	F	Day of week in month	2	E	Day in week	Tuesday; Tue	a	AM/PM marker	PM	H	Hour in day (0-23)	0	k	Hour in day (1-24)	24	K	Hour in AM/PM (0-11)	0	h	Hour in AM/PM (1-12)	12	m	Minute in hour	30	s	Second in minute	55	S	Millisecond	978	z	General Time Zone	Pacific Standard Time: PST; GMT-08:00	Z	RFC 822 Time Zone	-0800
Letter	Date or Time Component	Examples																																																											
G	Era Designator	AD																																																											
y	Year	1996;96																																																											
M	Month in year	July; Jul; 07																																																											
w	Week in year	27																																																											
W	Week in month	2																																																											
D	Day in year	189																																																											
d	Day in month	10																																																											
F	Day of week in month	2																																																											
E	Day in week	Tuesday; Tue																																																											
a	AM/PM marker	PM																																																											
H	Hour in day (0-23)	0																																																											
k	Hour in day (1-24)	24																																																											
K	Hour in AM/PM (0-11)	0																																																											
h	Hour in AM/PM (1-12)	12																																																											
m	Minute in hour	30																																																											
s	Second in minute	55																																																											
S	Millisecond	978																																																											
z	General Time Zone	Pacific Standard Time: PST; GMT-08:00																																																											
Z	RFC 822 Time Zone	-0800																																																											

Correlation of Occurrence Setting and Null Value Suppression

The value for null value suppression has a higher priority than the occurrence setting of affected elements or attributes. The following rules apply:

1. **NVS="Suppress Element"/"Suppress Attribute"**
Setting NVS to "Suppress Element" or "Suppress Attribute" results in a value of minimum occurrence of zero, which makes it optional.

2. NVS="No Suppression"

Setting NVS to "No Suppression" means that the element/attribute is always displayed so the value of #minOccur must be 1 - otherwise the element/attribute would be optional and could be suppressed.

3. NVS for arrays

For arrays, the dependency of minimum occurrence and Null Value Suppression is more complex:

1. NVS set to "No Suppression"

For an array with fixed size the setting is #minOccur = #maxOccur = #array size.

An array with variable size will contain all non-suppressable elements, but at least #minOccur elements. Empty elements will be generated to guarantee the number of #minOccur.

2. NVS set to value other than "No Suppression"

The number of array elements is at least #minOccur (empty elements will be generate and appended if number of elements is lower than #minOccur).

The lower panel is only available for element nodes and contains namespace-related properties:

Item	Description
Namespace URI	A default Namespace (xmlns=) for this element node.
Namespace definitions (table)	A table of namespace prefix / URI assignments for this element node. The listed namespaces are defined for this element and all subelements, as described in the XML namespaces specification document. Two buttons, Add and Delete, allow you to add or delete namespace definitions, respectively.

Using the Context Menu

The context menu of the XML tree enables you to modify the XML structure. It contains menu items to insert new nodes, remove or rename nodes, remove an IDL mapping, change the XML part type to element or attribute, move the selected node to the top or bottom of the current subtree, move the selected node up or down within the subtree, or show the XML Node Properties dialog.

> To open the context menu

- Select the XML tree node and click the right mouse button.

Menu Item	Shortcut	Description
New child node	Ctrl-N	Insert a child node under the selected node.
Insert before	Ctrl-B	Insert a new node before the selected node.
Insert after	Insert	Insert a new node after the selected node.
Set to Attribute	Ctrl-S	Turn an attribute into an element. Also works for multiple selected nodes.
Set to Element	Ctrl-E	Turn an element into an attribute. Also works for multiple selected nodes.
Bring to top	Ctrl-T	Move the selected node to the top of the current subtree.
Bring to bottom	Ctrl-O	Move the selected node to the bottom of the current subtree.
Move up	Ctrl-U	Move the selected node up within the subtree.
Move down	Ctrl-D	Move the selected node down within the subtree.
Cut	Ctrl-X	Cut the selected node.
Copy	Ctrl-C	Copy the selected node.
Paste	Ctrl-V, Ctrl-P	Paste a node from the clipboard (after copy or cut).
Unmap	Ctrl-M	Unmap the selected IDL and XML nodes, i.e, remove link. Also works for multiple selected nodes.
Delete	Delete/Backspace	Remove the selected node. Also works for multiple selected nodes.
Rename	Ctrl-R	Rename the selected node.
Properties	Alt-Enter	Open the XML Details Panel.

Using Drag-and-drop

You can use drag-and-drop operations within the XML tree instead of move and cut-and-paste. Select a node (even with subnodes) and drag it to another place. The dragged subtree is then inserted after, before or under the corresponding drop node (to insert it under the drop node, use the Ctrl key as a toggle).

Some drag-and-drop operations are illegal, e.g. dragging a node into one of its descendant nodes (children). This would result in a cyclic reference and is thus forbidden.

Dropping a dragged node into an attribute node will convert the attribute node to an element node. This is because attributes may not have descendant nodes.



Warning:

If you drag a subtree into or out of an array, the IDL mapping links of all nodes of that subtree will be deleted. This is because the cardinality of the node occurrence has changed, and the resulting IDL mapping is very likely to be incorrect.

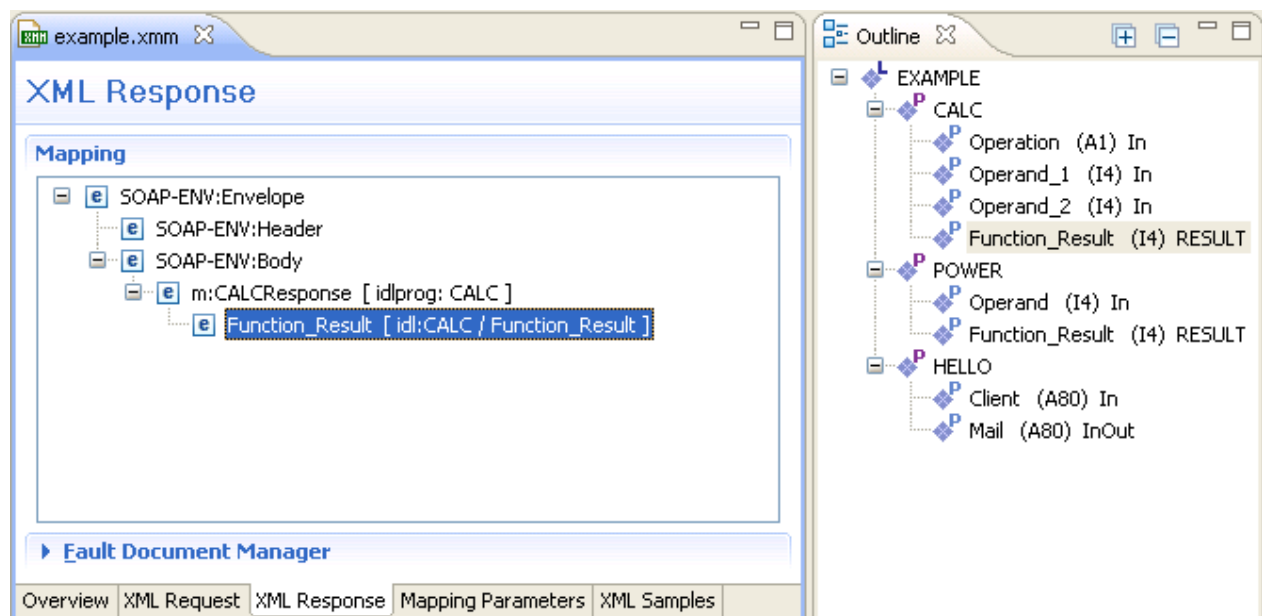
Drag-and-drop from an IDL Tree Node to the XML Tree

If you drag an IDL tree node onto an XML tree node, the IDL-XML mapping link for that XML node is changed. The new XML-IDL mapping of this XML tree node is to the dragged IDL node. The IDL tree is not changed.

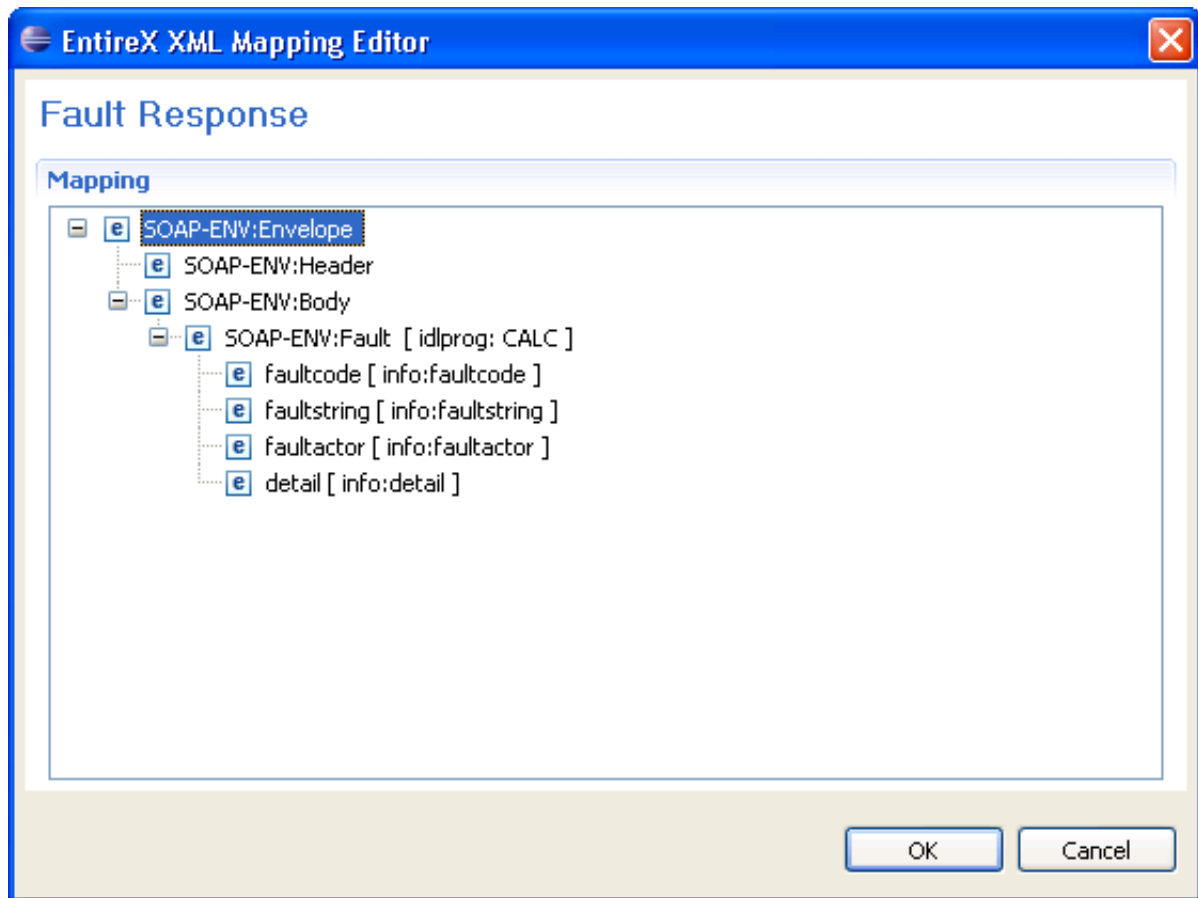
This is a quick way to change IDL mappings for XML parts. Note that there is no immediate check for duplicate assignment of an IDL node; however, the validity checks will detect it.

XML Response Page

This page contains the Mapping Tree of the XML Response, which is linked together by mapping paths with the IDL tree in the Outline View. The page extends the XML Request page with an additional section, Fault Document Manager, at the bottom.



The Fault Document Manager contains the entry for the XML Fault Response Document. By selecting the entry, the **Edit...** button will be enabled. Press **Edit...** to display the following dialog:



The same operations as for the XML Request and XML Response trees are possible. Confirm the changes with **OK**, or click **Cancel** to exit without changes.

Mapping Parameters Page

This page covers the Mapping Parameters to affect the Mapping Structure. The default values for that page will be managed in the preferences and can be loaded by using the **Restore Defaults** button in the upper right corner.

example.xmm
Restore Defaults

Mapping Parameters

The mapping parameters can only be used for generating a new IDL-XML mapping. Just changing mapping parameters has no effect on an existing mapping; generate the mapping again for the changes to take effect.

Document Style

Generate Array Envelope Element

WSDL Style: document/literal

Encoding Settings

XML Default Encoding: UTF-8

Use incoming Encoding

Null Value Suppression

Control empty elements or attributes may be omitted in the XML documents.

Enable Null Value Suppression

Elements

Simple Element

No Suppression

Complex Types

Suppress Group Elements

Array Items

Cells at End (Trim)

Attributes

No Suppression

Preview

```
<e1 />
<group1>
  <eg1 />
  <eg2>aaa</eg2>
</group1>
<group2 />
<array>
  <item1 />
  <item2>two</item2>
  <item3 />
  <item4>four</item4>
</array>
<e1 att1="" att2="red" />
```

Namespace Definitions

Table of defined Namespace definitions:

Prefix	Namespace	Default	
m	urn:com-softwareag-entirex-rpc:%l-%p	(default)	Insert...
SOAP-ENC	http://schemas.xmlsoap.org/soap/encodin...		Edit...
SOAP-ENV	http://schemas.xmlsoap.org/soap/envelop...		Remove
xsd	http://www.w3.org/2001/XMLSchema		Default
xsi	http://www.w3.org/2001/XMLSchema-inst...		

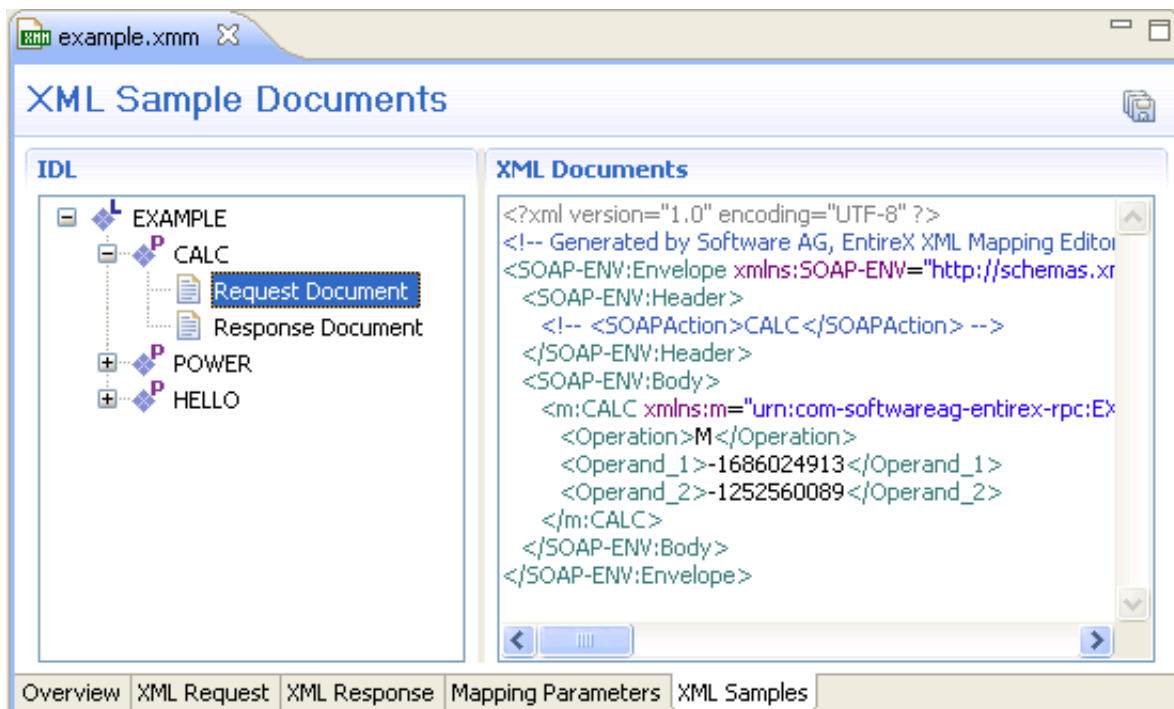
Overview | XML Request | XML Response | **Mapping Parameters** | XML Samples

Parameter	Description
Generate Array Envelope Element	Determines whether for each array a surrounding additional element (envelope) is generated or not.
WSDL Style	Prepare the SOAP Mapping for selected WSDL Style. Possible values: document/literal or RPC/encoded.
XML Default Encoding	This encoding is used for the XML/SOAP document sent if the box Use incoming XML encoding is not checked (for XML-based clients), or if the XML/SOAP RPC Server is used.
Use Incoming Encoding	Check this box to enable the XML/SOAP Wrapper to use same encoding for the incoming document as for the outgoing document.
Enable Null Value Suppression ⁽¹⁾	Switch on/off the null value suppression. ⁽¹⁾
Simple Element ⁽¹⁾	Suppress Elements. Possible Values: No Suppression or Suppress Element. ⁽¹⁾
Simple Attribute ⁽¹⁾	Suppress Attributes. Possible Values: No Suppression or Suppress Attribute. ⁽¹⁾
Array Types ⁽¹⁾	Suppress Array Types. Possible Values: No Suppression, All empty cells or Cells at end (Trim). ⁽¹⁾
Complex Types ⁽¹⁾	Suppress Complex Types. Possible Values: No Suppression Suppression (no special handling of complex types - null value suppression defined for 'Simple Element' is used), or Suppress Group Elements. ⁽¹⁾
Namespace Definitions Table	Manage all Namespaces with prefix and URI.

⁽¹⁾ For more details on null value suppression, see below or *Null Value Suppression* in *Writing Advanced Applications with the XML/SOAP Wrapper*.

XML Sample Documents Page

This page allows the generation and modification of XML Sample Documents. They can be used to test the Mapping by sending the generated Request Document to the XML Tester. Another useful point is to compare the XML Sample Document structure with the real XML Document returned by your application to detect differences, for example Namespace Definitions, typos or anything else.



All generated XML Sample Documents can be stored in parallel to the selected XMM file by using the Save All toolbar button in the upper right corner of this page or the Save command in the context menu. The generated file name is built as follows:

```
[xmm name] .[library name].[program name].[direction (request | response)].xml
```

For example: example.EXAMPLE.CALC.request.xml

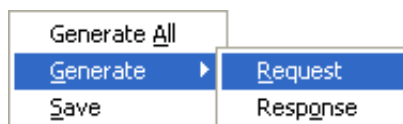
If a file already exists, a dialog will prompt you for confirmation to overwrite it.

Using the Context Menu of XML Samples

The XML Samples context menu allows the generation of XML Sample Documents.

There are two different menus:

- **Library** or **Program** is selected



Generate All means both directions (Request and Response). When a Library is selected, the commands here will be inherited by all Programs.

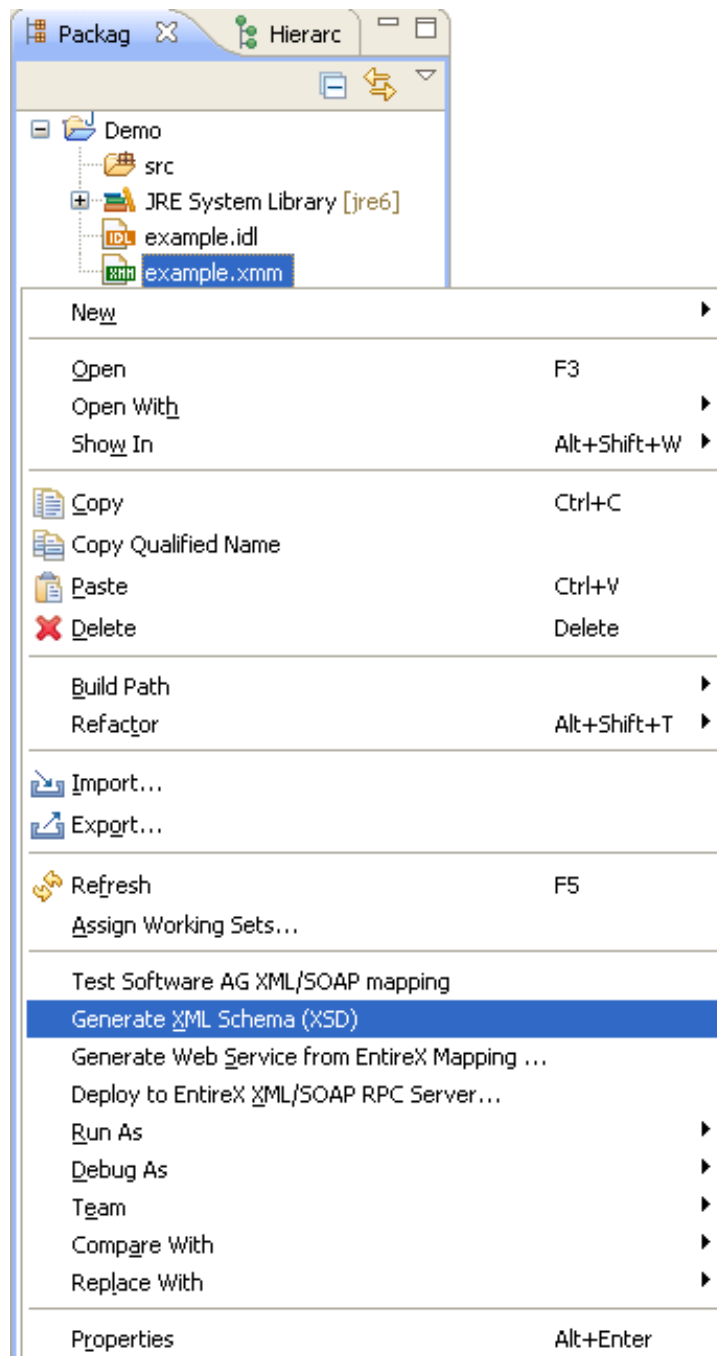
- **Request** or **Response** Document is selected



The **XML Tester...** command sends the selected XML Sample Document to the XML Tester as Quick Test and enters the name of the XMM file.

XML Schema Export

The current XML Mapping can be exported as XML Schema, using the context menu of the XMM file. The XML Schema style "Russian Doll" or "Venetian Blind" can be selected in the preferences.



The generated XML Schema files will be stored in parallel to the selected XMM file, where one file represents the direction (request or response) and the defined prefixed element.

For example: a SOAP mapping creates four files from the example CALC program: two prefixes ("SOAP-ENV" and "m"), and two directions (request and response). The generated file name is built as follows.

```
[xmm name].[library name].[program name].[direction (request | response)][(optional).additional prefix].xsd
```

Example

- example.EXAMPLE.CALC.request.xsd
- example.EXAMPLE.CALC.request.m.xsd
- example.EXAMPLE.CALC.response.xsd
- example.EXAMPLE.CALC.response.m.xsd

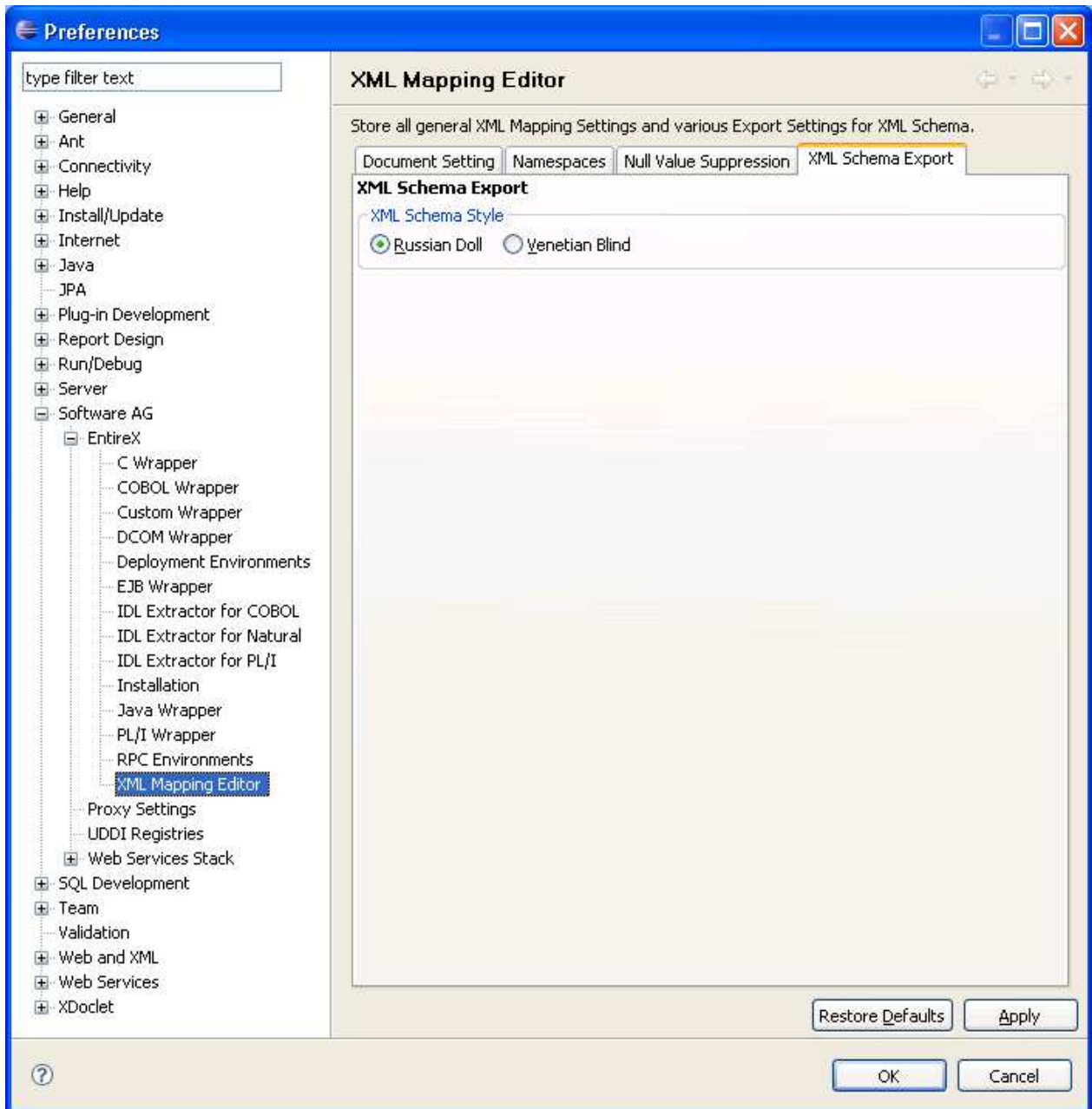
If a file already exists, a dialog will prompt you for confirmation to overwrite it.

XML Mapping Editor Settings

The preference page XML Mapping Editor manages the default values.

For XML Schema the style "Russian Doll" or "Venetian Blind" can be selected.

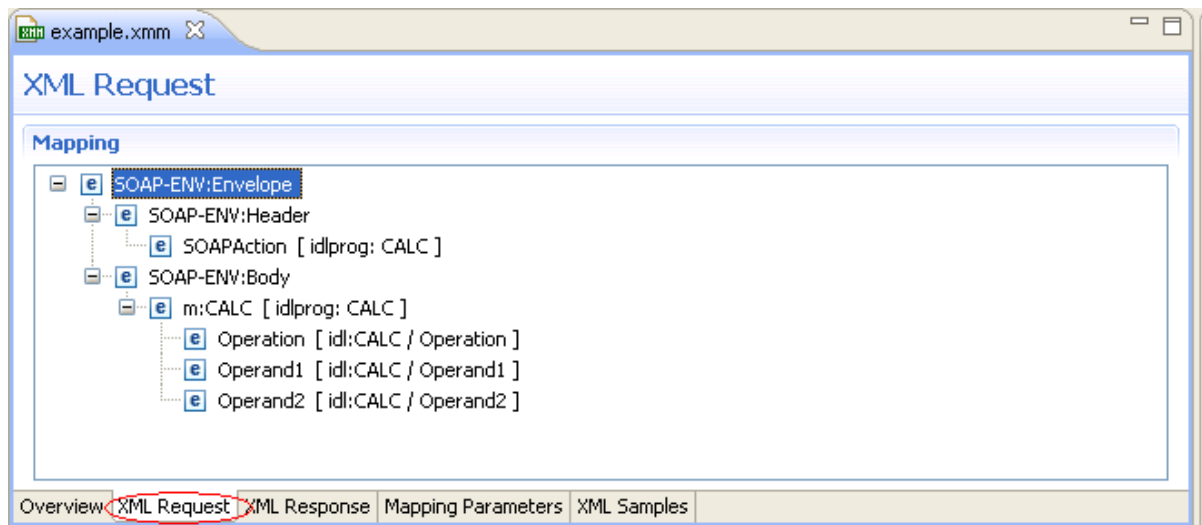
All other parameters are described under *Mapping Parameters*.



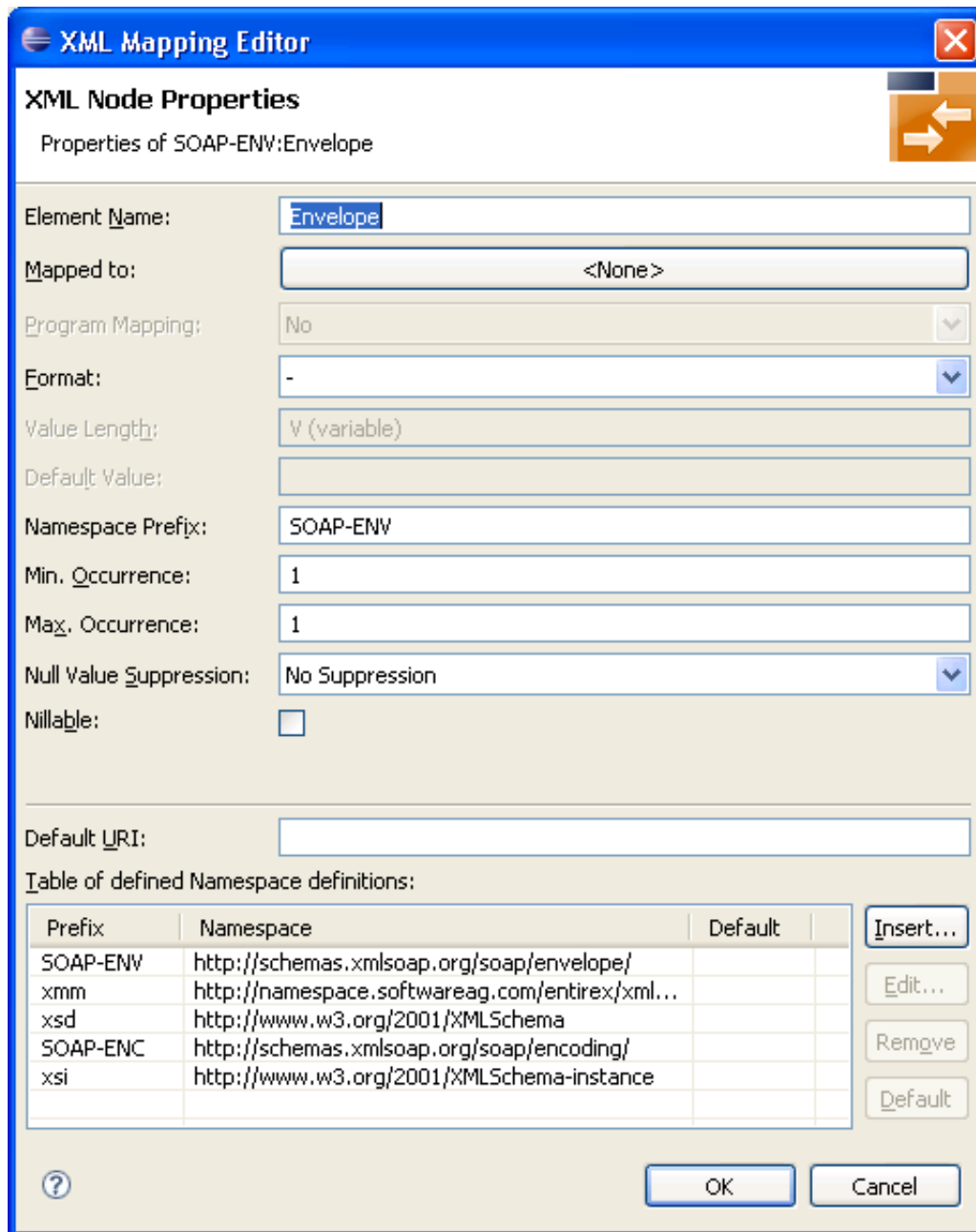
Removing Unused Namespaces

➤ To remove one or more unused namespaces

1. Open the XML Mapping Editor for the XMM file.
2. Select tab **XML Request**.



3. Select the element that defines namespace(s) to be removed.
4. From the context menu, choose **Properties**.



5. Select (unused) namespace entry/entries in list and choose **Remove**.
6. Press **OK**.
7. Repeat these steps for the XML response (select the **XML Response** tab and repeat the steps above).
8. Save the XMM file.
9. Switch to **XML Samples** tab, which shows the following:

XML Mapping Editor
XML Node Properties
Properties of SOAP-ENV:Envelope

Element Name:

Mapped to:

Program Mapping:

Format:

Value Length:

Default Value:

Namespace Prefix:

Min. Occurrence:

Max. Occurrence:

Null Value Suppression:

Nullable:

Default URI:

Table of defined Namespace definitions:

Prefix	Namespace	Default
SOAP-ENV	http://schemas.xmlsoap.org/soap/envelope/	
xmm	http://namespace.softwareag.com/entirex/xml...	
xsd	http://www.w3.org/2001/XMLSchema	
SOAP-ENC	http://schemas.xmlsoap.org/soap/encoding/	
xsi	http://www.w3.org/2001/XMLSchema-instance	

Buttons: Insert..., Edit..., Remove, Default, OK, Cancel

10. The XMM file defines the following:

```

...
<FromXml>
<Method relatedIdlLibrary="EXAMPLE" relatedIdlProgram="CALC" encoding="UTF-8"
useIncomingEncoding="true">
<XmlNode name="Envelope" length="0" min="1" max="1" namespacePrefix="SOAP-ENV"
nullSuppression="NVS_NONE" nullValue="" >
<XmlNamespaceDef prefix="SOAP-ENV" uri="http://schemas.xmlsoap.org/soap/envelope/" />
<XmlNamespaceDef prefix="SOAP-ENC" uri="http://schemas.xmlsoap.org/soap/encoding/" />
<XmlNode name="Header" length="0" min="1" max="1" namespacePrefix="SOAP-ENV"
nullSuppression="NVS_NONE" nullValue="" >
<XmlNode name="SOAPAction" format="string" type="xsd:string" length="0" min="1" max="1"
default="CALC" nullSuppression="NVS_NONE" nullValue="" programNode="ev" >
</XmlNode>
...

```