Using an external Data Source

This function enables you to parse an XML document into a Natural variable defined in a local, global or parameter data area.

The following topics are covered:

- Generate from Document Type Definition or XML Schema
- Select Root Element or Document Type
- Select Recursion Level
- Generate Natural Data Area
- Generate Copycode for Serialization
- Generate Subprogram for Serialization
- Generate Copycode for XML Parser Callback
- Generate Subprogram for XML Parser Callback
- Show Generation Results

See also:

- Using a Natural Data Area as Data Source
- Setting up Specific Generation Options

Note:
When using an XML Schema (XSD) as input document type, the first XSD element will be used as the root element.

Generate from Document Type Definition or XML Schema

This dialog is used to select a Document Type Definition (DTD), XML Schema (XSD) or Tamino Schema 2 (TSD) as input Document Type.

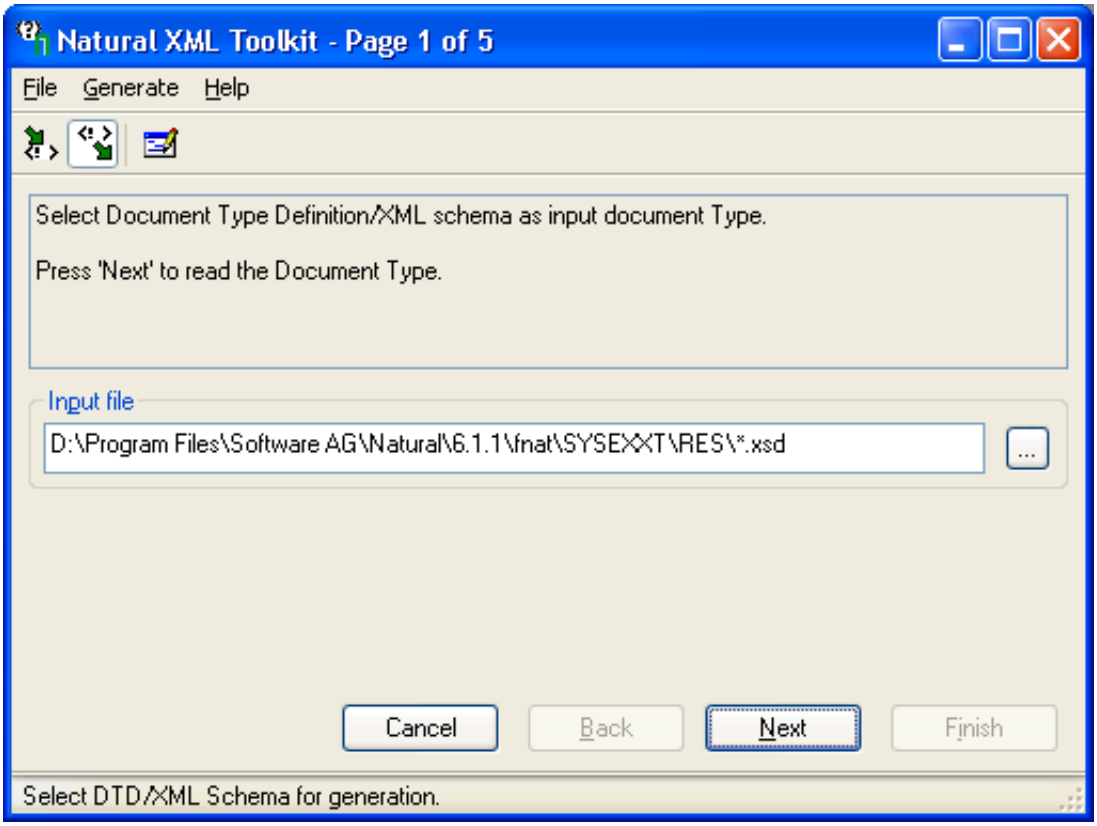
To invoke the dialog shown below

1. Choose From DTD/XSD from the Generate menu.

   Or:

   Choose the  button.

Note:
The field entries shown in the dialogs below are default or example values.



Field Descriptions

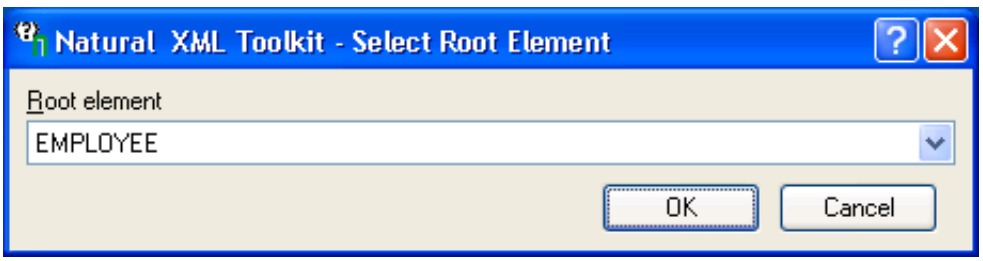
Input File

Select a DTD, XSD or TSD file. You can use the browse button ... to search for an existing DTD, XSD or TSD file.

Choose **Next** to continue.

Select Root Element or Document Type

This dialog is used to select an element or document type that should be the root of your XML document.



Field Descriptions

Root Element (for DTDs)

Default Value:	(All Elements)

Select the desired element, e.g. EMPLOYEE, and choose **OK**.

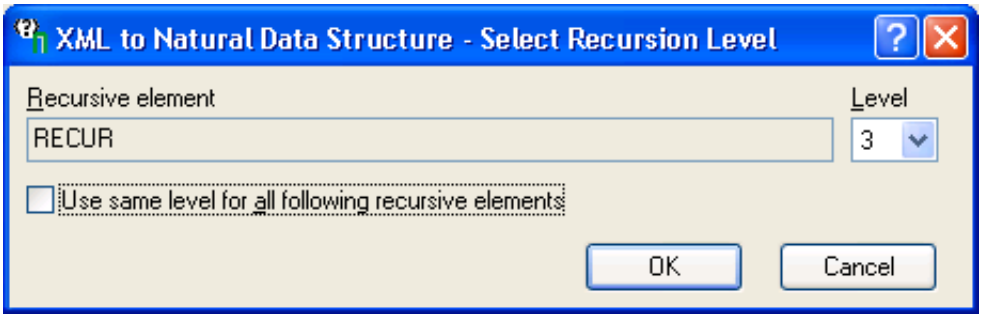
Document Type (for Tamino Schema)

Default Value:	(All Elements)

Select the desired element, e.g. EMPLOYEE, and choose **OK**.

Select Recursion Level

This dialog is only displayed, if the DTD, XSD or TSD selected in the first dialog includes recursive elements.



Field descriptions

Recursive Element

Name of the Element that is used recursively.

Default Value:	(All Libraries)

Level

Number of recursion levels that should be generated.

Default Value:	3

Use same level for all following recursive elements

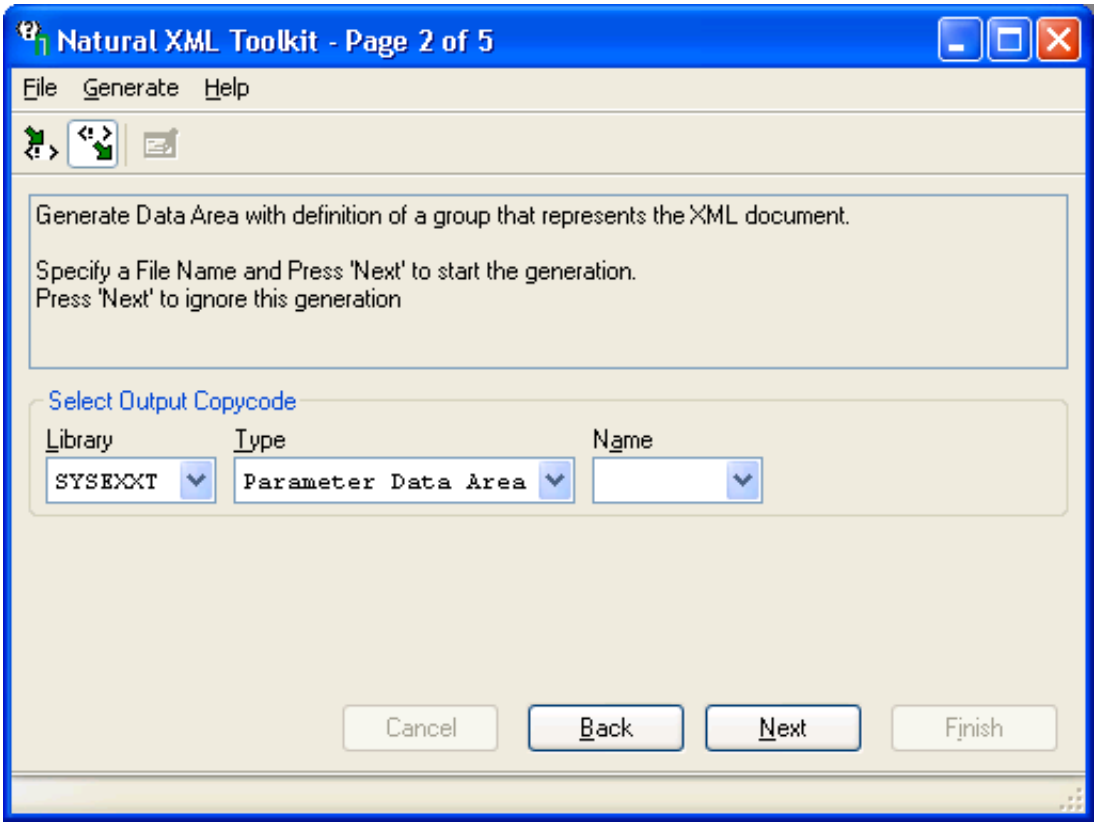
If another recursive element is found, the same recursion level will be used.

Default Value:	unchecked

Choose **OK** to continue.

Generate Natural Data Area

This dialogscreen is used to generate a Natural Data Area with definition of a group that represents the XML document.



Field Descriptions

Library

Belongs to Group:	Select Output Copycode
Default Value:	(All libraries)

Type

Belongs to Group:	Select Output Copycode
Default Value:	**L** - Local Data Area

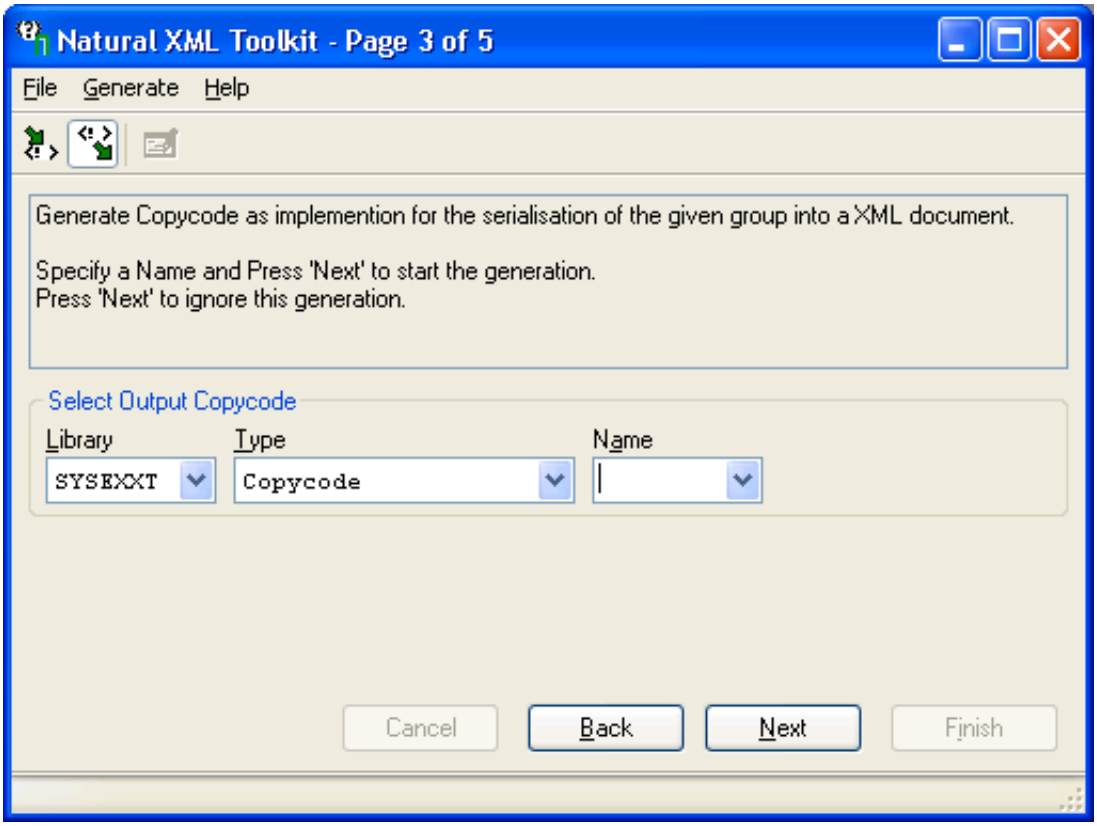
Name

Belongs to Group:	Select Output Copycode
Default Value:	(All objects of the selected library and type)

Choose **Next** to continue.

Generate Copycode for Serialization

This dialog is used to generate copycode as implementation for the serialization of the given group into an XML document.



See also Serialize Copycode (in the Examples document).

Field Descriptions

Library

Belongs to Group:	Select Output Copycode
Default Value:	(All libraries)

Type

Belongs to Group:	Select Output Copycode
Default Value:	Copycode

Name

Belongs to Group:	Select Output Copycode
Default Value:	(All objects of the selected library and type)

Choose **Next** to continue.

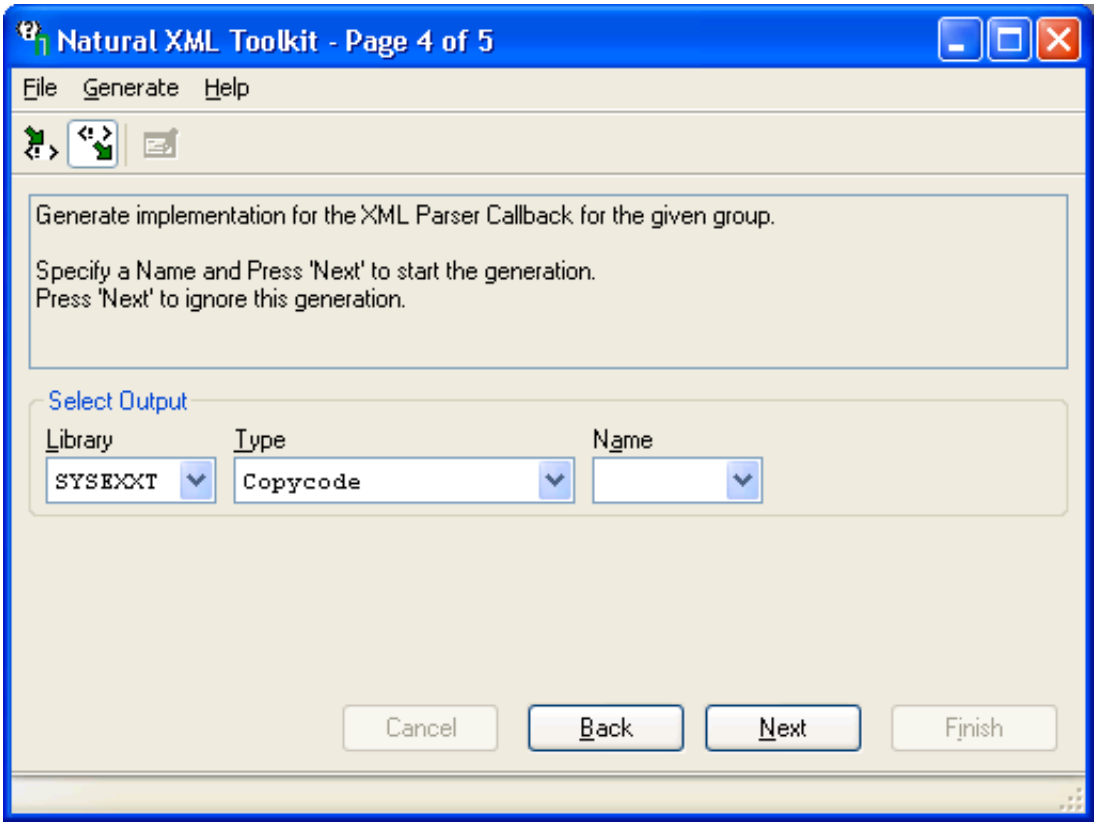
Generate Subprogram for Serialization

This dialog is used to generate a subprogram as implementation for the serialization of the given group into an XML document.

It uses the same entries as the above dialog for copycode, except that the Type field contains the entry Subprogram.

Generate Copycode for XML Parser Callback

This dialog is used to generate copycode as implementation for the XML Parser Callback for the given group.



Generates the parser CALLBACK copycode. See also Parser CALLBACK Copycode (in the Examples document).

Field Descriptions

Library

Belongs to Group:	Select Output
Default Value:	(All libraries)

Type

Belongs to Group:	Select Output
Default Value:	Copycode

Name

Belongs to Group:	Select Output
Default Value:	(All objects of the selected library and type)

Choose **Next** to continue.

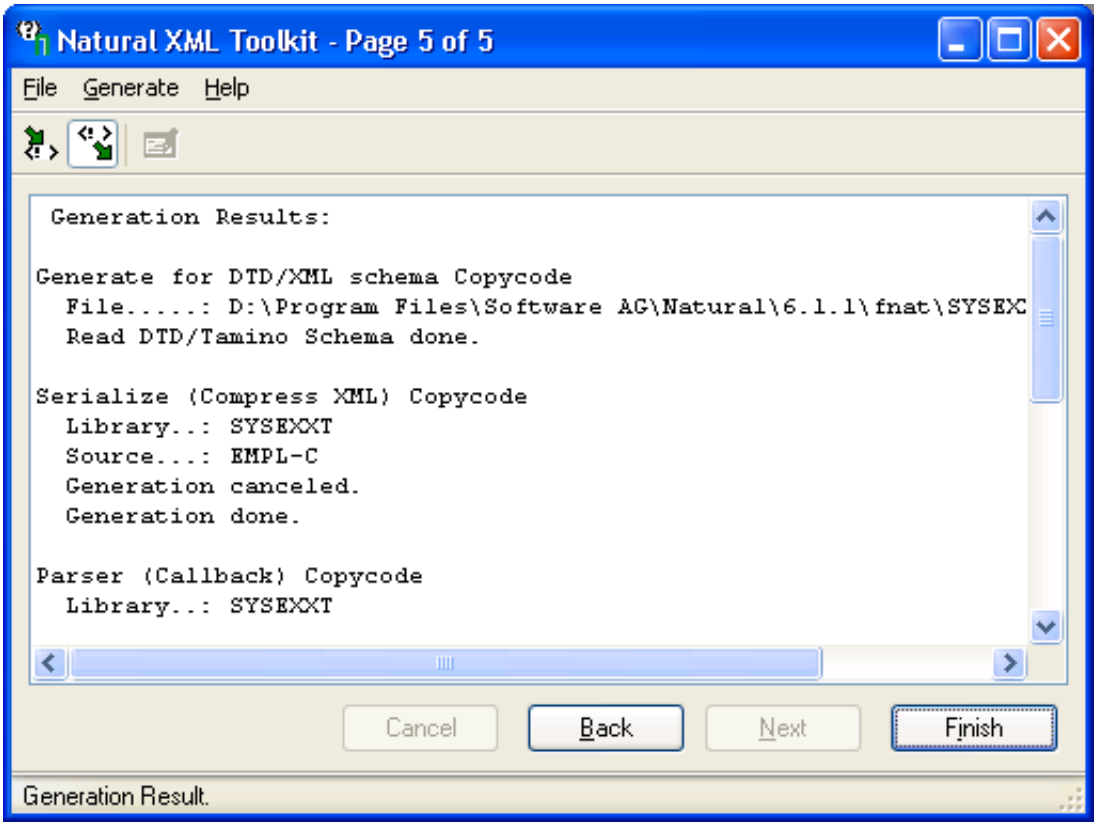
Generate Subprogram for XML Parser Callback

This dialog is used to generate a subprogram as implementation for the XML Parser Callback for the given group.

It uses the same entries as the above dialog for copycode, except that the Type field contains the entry Subprogram.

Show Generation Results

After the generation is complete, the generation results summary is displayed.



Choose **Finish** to end the generation process.