INaturalStudioPlugInTree

This chapter covers the following topics:

- Purpose

- Methods

Purpose

A plug-in provides this interface in order to provide information about tree view and list view nodes to Natural Studio. Natural Studio calls the methods of this interface if the plug-in has defined its own node types and has opened a tree view or list view document window with a node of one of these types as root node. Natural Studio calls the methods to gather information about these nodes whenever this is required to expand or refresh a tree view or list view.

Note:
A plug-in defines its own node types by using the interface `INatAutoNodeTypes`.

In order to provide the interface, plug-ins written in Natural include the interface module (copycode) `NSTPLG-T` from the example library `SYSEXPLG` in their class definition. Plug-ins written in other languages use the type library `naturalstudioplugin.tlb`. This type library is also contained in the example library `SYSEXPLG`.

Methods

The following methods are available:

- GetData

- GetChildren

- HasChildren

- GetAttributes

- GetAttributeValues

GetData

Natural Studio calls this method to retrieve additional information about a plug-in defined node identified by type and key.

Parameters

Name	Natural Data Format	Variant Type	Remark
ReturnValue	L	VT_BOOL	
Type	I4	VT_I4	
Key	A	VT_BSTR	
Info	A	VT_BSTR	
Template	I4	VT_I4	
Data	A	VT_BSTR	By reference

ReturnValue

The plug-in returns TRUE if it knows the node, FALSE otherwise.

Type

The type of the node.

Key

The key that identifies the node within its type.

Info

An additional information string that the plug-in has previously assigned to the node.

Template

The TemplateID that the plug-in has passed to the method `INatAutoObjectTrees::Open` or `INatAutoObjectLists::Open` when opening the tree view or list view document window.

Data

The plug-in returns a string containing an XML document that describes the node. The XML document is structured according to the following DTD.

```
<?xml version="1.0"?>
<!ELEMENT data (pinfo?, pname?, (ntype, nkey)?)>
<!ELEMENT pinfo (#PCDATA)>
<!ELEMENT pname (#PCDATA)>
<!ELEMENT ntype (#PCDATA)>
<!ELEMENT nkey  (#PCDATA)>
```

Element	Meaning
pinfo	Additional information about the node that the plug-in wants to receive back whenever Natural Studio later refers to the node. Natural Studio never considers the content of this element, but just passes it back and forth.
pname	The text to be displayed with the node in a tree view or list view.
ntype	If the node has been defined by a plug-in, but corresponds to an node of a predefined Natural Studio type, this element contains the type of the corresponding predefined node.
nkey	If the node has been defined by a plug-in, but corresponds to an node of a predefined Natural node type, this element contains the key of the corresponding predefined node.

GetChildren

Natural Studio calls this method to retrieve the child nodes of a node defined by a plug-in. This node is identified by type and key.

Parameters

Name	Natural Data Format	Variant Type	Remark
ReturnValue	I4	VT_I4	
Type	I4	VT_I4	
Key	A	VT_BSTR	
Info	A	VT_BSTR	
Template	I4	VT_I4	
Children	A	VT_BSTR	By reference

ReturnValue

Indicates the number of child nodes.

Type

The type of the node.

Key

The key that identifies the node within its type.

Info

An additional information string that the plug-in has previously assigned to the node.

Template

The TemplateID that the plug-in has passed to the method `INatAutoObjectTrees::Open` or `INatAutoObjectLists::Open` when opening the tree view or list view document window. The plug-in can interpret the parameter `Template` to return different tree structures for different values of `Template`.

Children

The plug-in returns a string containing an XML document that describes the child nodes. The XML document is structured according to the following DTD.

```
<?xml version="1.0"?>
<!ELEMENT children (child*)>
<!ELEMENT child (ptype, pkey, pinfo?, pname?,
phch?, children?, (ntype, nkey)?)>
<!ELEMENT ptype (#PCDATA)>
<!ELEMENT pkey  (#PCDATA)>
<!ELEMENT pinfo (#PCDATA)>
<!ELEMENT pname (#PCDATA)>
<!ELEMENT phch  (#PCDATA)>
<!ELEMENT ntype (#PCDATA)>
<!ELEMENT nkey  (#PCDATA)>
```

Element	Meaning
ptype	The type of the child node.
pkey	The key that identifies the child node within its type.
pinfo	Additional information about the child node that the plug-in wants to receive back whenever Natural Studio later refers to the node. Natural Studio never considers the content of this element, but just passes it back and forth.
pname	The text to be displayed with the child node in a tree view or list view.
children	Allows specifying a subtree of child nodes in arbitrary depth. If this element is specified, it is implicitly assumed that the child node itself has children and the method HasChildren is not called for this child node.
phch	Allows specifying in advance if the given child node itself has child nodes. ● A value of 1 means that the child node itself has child nodes. ● A value of -1 means that the child node itself has no child nodes. ● A value of 0 (default) means that the plug-in cannot determine now if the child node itself has child nodes and wants to be asked in a subsequent call to the method HasChildren.
ntype	If the node has been defined by a plug-in, but corresponds to a node of a predefined Natural Studio type, this element contains the type of the corresponding predefined node.
nkey	If the node has been defined by a plug-in, but corresponds to a node of a predefined Natural node type, this element contains the key of the corresponding predefined node.

HasChildren

Natural Studio calls this method to check if the plug-in defined node identified by type and key has child nodes of any type. This is used to decide if the node will be shown as expandable in a tree view document window.

Parameters

Name	Natural Data Format	Variant Type	Remark
ReturnValue	L	VT_BOOL	
Type	I4	VT_I4	
Key	A	VT_BSTR	
Info	A	VT_BSTR	
Template	I4	VT_I4	
Children	A	VT_BSTR	By reference

ReturnValue

> The plug-in returns TRUE if the node has child nodes, FALSE otherwise.

Type

> The type of the node.

Key

> The key that identifies the node within its type.

Info

> An additional information string that the plug-in has previously assigned to the node.

Template

> The TemplateID that the plug-in has passed to the method `INatAutoObjectTrees::Open` or `INatAutoObjectLists::Open` when opening the tree view or list view document window. The plug-in can interpret the parameter `Template` to return different tree structures for different values of `Template`.

Children

> If the plug-in not only knows that the given node has child nodes, but knows also the child nodes themselves, it can return them in this parameter. The plug-in then returns a string containing an XML document that describes the child nodes. The XML document is structured according to the DTD defined with the method `GetChildren`.

GetAttributes

Natural Studio calls this method to retrieve meta information about the attributes of the given node type. This method is called when the attributes of the node are to be displayed in a list view document window.

Parameters

Name	Natural Data Format	Variant Type	Remark
ReturnValue	I4	VT_I4	
Type	I4	VT_I4	
Template	I4	VT_I4	
Attributes	A	VT_BSTR	By reference

ReturnValue

The number of defined attributes.

Type

The type of the node.

Template

The TemplateID that the plug-in has passed to the method `INatAutoObjectTrees::Open` or `INatAutoObjectLists::Open` when opening the tree view or list view document window. The plug-in can interpret the parameter `Template` to return different attribute sets for different values of `Template`.

Attributes

The plug-in returns a string containing an XML document that describes the attributes. The XML document is structured according to the following DTD.

```
<?xml version="1.0"?>
<!ELEMENT attributes (attdef*)>
<!ELEMENT attdef        (akey, acaption?)>
<!ELEMENT akey    (#PCDATA)>
<!ELEMENT acaption (#PCDATA)>
```

Element	Meaning
akey	A key that identifies the attribute internally. This key is freely defined by the plug-in.
acaption	A caption that is used to display the attribute externally, for instance, in list view column headers.

For nodes that correspond to predefined Natural Studio node types the attributes of the corresponding Natural node type can also be specified. In order to refer to these attributes, the following predefined attribute keys are used. If one of these attribute keys is used, no attribute caption needs to be specified, because the captions are predefined by Natural Studio.

Key	Caption
NOLibrary	Library name
NOFnr	File number
NODBID	Database number
NOType	Natural object type
NOMode	Mode (Structured/Report)
NOUserID	User ID
NOSrcDate	Date of last save
NOSrcSize	Source size
NOCatDate	Date of last catalog
NOCatSize	GP size

GetAttributeValues

Natural Studio calls this method to retrieve the attribute values of the node defined by a plug-in. This node is identified by type and key.

Parameters

Name	Natural Data Format	Variant Type	Remark
ReturnValue	I4	VT_I4	
Type	I4	VT_I4	
Key	A	VT_BSTR	
Info	A	VT_BSTR	
Template	I4	VT_I4	
AttributeValues	A	VT_BSTR	By reference

ReturnValue

> The number of attribute values.

Type

> The type of the node.

Key

> The key that identifies the node within its type.

Info

An additional information string that the plug-in has previously assigned to the node.

Template

The TemplateID that the plug-in has passed to the method `INatAutoObjectTrees::Open` or `INatAutoObjectLists::Open` when opening the tree view or list view document window. The plug-in can interpret the parameter `Template` to return different attribute sets for different values of `Template`.

AttributeValues

The plug-in returns a string containing an XML document that describes the attribute values. The XML document is structured according to the following DTD.

```
<?xml version="1.0"?>
<!ELEMENT attributevalues (attval*)>
<!ELEMENT attval        (akey, avalue)>
<!ELEMENT akey (#PCDATA)>
<!ELEMENT avalue (#PCDATA)>
```

Element	Meaning
akey	A key that identifies the attribute internally. The key must match one of the keys previously returned in a call to `GetAttributes`, otherwise the element is ignored.
avalue	The attribute value.