OpcVariableNode Members

Namespace: Opc.UaFx
Assemblies: Opc.UaFx.Advanced.dll, Opc.UaFx.Advanced.dll
The OpcVariableNode type exposes the following members.

Constructors

OpcVariableNode(IOpcNode, OpcName)

Initializes a new instance of the OpcVariableNode class accessible by the `name` specified as a child node of the `parent` node given.

C#

```
protected OpcVariableNode(IOpcNode parent, OpcName name)
```

Parameters

`parent` IOpcNode

The IOpcNode used as the parent node or a null reference (Nothing in Visual Basic) in the case there is no parent node available.

`name` OpcName

The OpcName through that the new variable node can be accessed.

OpcVariableNode(IOpcNode, OpcName, Object)

Initializes a new instance of the OpcVariableNode class accessible by the `name` specified with the initial value defined by `value` as a child node of the `parent` node given.

C#

```
protected OpcVariableNode(IOpcNode parent, OpcName name, object value)
```

Parameters

`parent` IOpcNode

The IOpcNode used as the parent node or a null reference (Nothing in Visual Basic) in the case there is no parent node available.

`name` OpcName

The OpcName through that the new variable node can be accessed.

`value` Object

The initial value of the new variable node.

OpcVariableNode(IOpcNode, OpcName, OpcNodeId)

Initializes a new instance of the OpcVariableNode class accessible by the `name` and `id` specified as a child node of the `parent` node given.

C#

```
protected OpcVariableNode(IOpcNode parent, OpcName name, OpcNodeId id)
```

Parameters

`parent` IOpcNode

The IOpcNode used as the parent node or a null reference (Nothing in Visual Basic) in the case there is no parent node available.

`name` OpcName

The OpcName through that the new variable node can be accessed.

`id` OpcNodeId

The OpcNodeId through that the new variable node can be identified and accessed.

OpcVariableNode(IOpcNode, OpcName, OpcNodeId, Object)

Initializes a new instance of the OpcVariableNode class accessible by the `name` and `id` specified with the initial value defined by `value` as a child node of the `parent` node given.

C#

```
protected OpcVariableNode(IOpcNode parent, OpcName name, OpcNodeId id, object value)
```

Parameters

`parent` IOpcNode

The IOpcNode used as the parent node or a null reference (Nothing in Visual Basic) in the case there is no parent node available.

`name` OpcName

The OpcName through that the new variable node can be accessed.

`id` OpcNodeId

The OpcNodeId through that the new variable node can be identified and accessed.

`value` Object

The initial value of the new variable node.

OpcVariableNode(OpcName)

Initializes a new instance of the OpcVariableNode class accessible by the `name` specified.

C#

```
protected OpcVariableNode(OpcName name)
```

Parameters

`name` OpcName

The OpcName through that the new variable node can be accessed.

OpcVariableNode(OpcName, Object)

Initializes a new instance of the OpcVariableNode class accessible by the `name` specified with the initial value given by `value`.

C#

```
protected OpcVariableNode(OpcName name, object value)
```

Parameters

`name` OpcName

The OpcName through that the new variable node can be accessed.

`value` Object

The initial value of the new variable node.

OpcVariableNode(OpcName, OpcNodeId)

Initializes a new instance of the OpcVariableNode class accessible by the `name` and `id` specified.

C#

```
protected OpcVariableNode(OpcName name, OpcNodeId id)
```

Parameters

`name` OpcName

The OpcName through that the new variable node can be accessed.

`id` OpcNodeId

The OpcNodeId through that the new variable node can be identified and accessed.

OpcVariableNode(OpcName, OpcNodeId, Object)

Initializes a new instance of the OpcVariableNode class accessible by the `name` and `id` specified with the initial value given by `value`.

C#

```
protected OpcVariableNode(OpcName name, OpcNodeId id, object value)
```

Parameters

`name` OpcName

The OpcName through that the new variable node can be accessed.

`id` OpcNodeId

The OpcNodeId through that the new variable node can be identified and accessed.

`value` Object

The initial value of the new variable node.

Properties

AccessLevel

Gets or sets a value which indicates in which ways the Value attribute of the variable node can be accessed (read/write) and if it provides current and/or historic data.

C#

```
public OpcAccessLevel AccessLevel { get; set; }
```

Property Value

OpcAccessLevel

One of the members defined by the OpcAccessLevel enumeration. Which is used to indicate the ways the Value attribute of the current OpcVariableNode can be accessed (read/write) and if it provides current and/or historic data. The AccessLevel does not take any user access rights into account, although the variable is writable this may be restricted to a certain user / user group.

ArrayDimensions

Gets the number/lengths of dimensions for an array Value with one or more fixed dimensions.

C#

```
public OpcArrayDimensions ArrayDimensions { get; set; }
```

Property Value

OpcArrayDimensions

An instance of the OpcArrayDimensions class which offers the number/lengths of dimensions for an array Value.

Remarks

If the ValueRank does not identify an array of a specific dimension (i.e. ValueRank ⇐ 0) ArrayDimensions can be a null reference (Nothing in Visual Basic).

DataType

Gets or sets a value which defines a pre-defined used DataTypeId as one of the members defined by the OpcDataType enumeration to simplify querying standard data types.

C#

```
public virtual OpcDataType DataType { get; set; }
```

Property Value

OpcDataType

One of the members defined by the OpcDataType enumeration.

DataTypeId

Gets or stets the identifier which identifies the node that defines the type of data represented by the variable node.

C#

```
public OpcNodeId DataTypeId { get; set; }
```

Property Value

OpcNodeId

The OpcNodeId of the data type node which defines the type of data represented by the variable node. These data type node defines either a simple or a complex type of data accessible by the Value property.

DefaultReferenceTypeId

Gets the default identifier which identifies the type that defines the underlying node reference within this OpcInstanceNode is referenced by its parent node.

C#

```
protected override OpcNodeId DefaultReferenceTypeId { get; }
```

Property Value

OpcNodeId

The OpcNodeId of the reference wihin this OpcInstanceNode is referenced. These reference defines the typical behaviour of an instance node and its role in the address space regarding its parent. If there exists no specific reference type a null reference (Nothing in Visual Basic).

DefaultTypeDefinitionId

Gets the default identifier which identifies the node that defines the underlying node type from that this OpcInstanceNode has been created.

C#

```
protected override OpcNodeId DefaultTypeDefinitionId { get; }
```

Property Value

OpcNodeId

The OpcNodeId of the type node from that this OpcInstanceNode has been created from. These type node defines the typical structure of an instance node of its type definition. If there exists no specific type definition node a null reference (Nothing in Visual Basic).

HistoryConfiguration

C#

```
public OpcHistoryConfigurationNode HistoryConfiguration { get; }
```

Property Value

OpcHistoryConfigurationNode

IsArray

Gets a value indicating whether the Value of the OpcVariableNode represented is an array.

C#

```
public bool IsArray { get; }
```

Property Value

Boolean

The value true if the Value of the OpcVariableNode represented is an array; otherwise the false.

IsHistorizing

C#

```
public bool IsHistorizing { get; set; }
```

Property Value

Boolean

ReadAccessLevelCallback

C#

```
public OpcReadAttributeValueCallback<OpcAccessLevel> ReadAccessLevelCallback { get; set; }
```

Property Value

OpcReadAttributeValueCallback<OpcAccessLevel>

ReadArrayDimensionsCallback

C#

```
public OpcReadAttributeValueCallback<OpcArrayDimensions> ReadArrayDimensionsCallback { get;
set; }
```

Property Value

OpcReadAttributeValueCallback<OpcArrayDimensions>

ReadDataTypeCallback

C#

```
public OpcReadAttributeValueCallback<OpcNodeId> ReadDataTypeCallback { get; set; }
```

Property Value

OpcReadAttributeValueCallback<OpcNodeId>

ReadIsHistorizingCallback

C#

```
public OpcReadAttributeValueCallback<bool> ReadIsHistorizingCallback { get; set; }
```

Property Value

OpcReadAttributeValueCallback<Boolean>

ReadUserAccessLevelCallback

C#

```
public OpcReadAttributeValueCallback<OpcAccessLevel> ReadUserAccessLevelCallback { get; set;
}
```

Property Value

OpcReadAttributeValueCallback<OpcAccessLevel>

ReadValueRankCallback

C#

```
public OpcReadAttributeValueCallback<int> ReadValueRankCallback { get; set; }
```

Property Value

OpcReadAttributeValueCallback<Int32>

ReadVariableValueCallback

Gets or sets a callback used to read the variable value.

C#

```
public OpcReadVariableValueCallback ReadVariableValueCallback { get; set; }
```

Property Value

OpcReadVariableValueCallback

A OpcReadVariableValueCallback used to read the variable value as an Object. The value can also be a null reference (Nothing in Visual Basic).

Status

C#

```
public OpcStatus Status { get; }
```

Property Value

OpcStatus

Timestamp

C#

```
public DateTime? Timestamp { get; set; }
```

Property Value

Nullable<DateTime>

UserAccessLevel

Gets or sets a value which indicates in which ways the Value attribute of the variable node can be accessed (read/write) and if it provides current and/or historic data taking user access rights into account.

C#

```
public OpcAccessLevel UserAccessLevel { get; set; }
```

Property Value

OpcAccessLevel

One of the members defined by the OpcAccessLevel enumeration. Which is used to indicate the ways the Value attribute of the current OpcVariableNode can be accessed (read/write) and if it provides current and/or historic data. The UserAccessLevel takes user access rights into account therefore its access may be restricted to a certain user / user group.

Value

Gets or sets the value of the variable node which may be simple or complex.

C#

```
public object Value { get; set; }
```

Property Value

Object

A Object representing the value of the variable node. This can be also a null reference (Nothing in Visual Basic).

ValueRank

C#

```
public int ValueRank { get; set; }
```

Property Value

Int32

WriteAccessLevelCallback

C#

```
public OpcWriteAttributeValueCallback<OpcAccessLevel> WriteAccessLevelCallback { get; set; }
```

Property Value

OpcWriteAttributeValueCallback<OpcAccessLevel>

WriteArrayDimensionsCallback

C#

```
public OpcWriteAttributeValueCallback<OpcArrayDimensions> WriteArrayDimensionsCallback {
get; set; }
```

Property Value

OpcWriteAttributeValueCallback<OpcArrayDimensions>

WriteDataTypeCallback

C#

```
public OpcWriteAttributeValueCallback<OpcNodeId> WriteDataTypeCallback { get; set; }
```

Property Value

OpcWriteAttributeValueCallback<OpcNodeId>

WriteIsHistorizingCallback

C#

```
public OpcWriteAttributeValueCallback<bool> WriteIsHistorizingCallback { get; set; }
```

Property Value

OpcWriteAttributeValueCallback<Boolean>

WriteUserAccessLevelCallback

C#

```
public OpcWriteAttributeValueCallback<OpcAccessLevel> WriteUserAccessLevelCallback { get;
set; }
```

Property Value

OpcWriteAttributeValueCallback<OpcAccessLevel>

WriteValueRankCallback

C#

```
public OpcWriteAttributeValueCallback<int> WriteValueRankCallback { get; set; }
```

Property Value

OpcWriteAttributeValueCallback<Int32>

WriteVariableValueCallback

Gets or sets a callback used to write the variable value.

C#

```
public OpcWriteVariableValueCallback WriteVariableValueCallback { get; set; }
```

Property Value

OpcWriteVariableValueCallback

A OpcWriteVariableValueCallback used to write the variable value as an Object. The value can also be a null reference (Nothing in Visual Basic).

Methods

InitializeDefaults()

Initializes the default values used by the OpcVariableNode.

C#

```
protected override void InitializeDefaults()
```

Remarks

This method is used to ensure the availability of appropriate node specific default values. For more information like when this method is to be overwritten see InitializeDefaults.

ReadAttributeValueCore<T>(OpcReadAttributeValueContext, OpcAttributeValue<T>)

C#

```
protected override OpcAttributeValue<T>
ReadAttributeValueCore<T>(OpcReadAttributeValueContext context, OpcAttributeValue<T> value)
```

Parameters

`context` OpcReadAttributeValueContext

`value` OpcAttributeValue<T>

Returns

OpcAttributeValue<T>

ReadValue(OpcReadVariableValueContext)

Reads the variable node value using the `context` specified.

C#

```
public object ReadValue(OpcReadVariableValueContext context)
```

Parameters

`context` OpcReadVariableValueContext

The OpcReadVariableValueContext to use to read the variable node value.

Returns

Object

The Object variable node value associated with this node and read using the `context` specified. This can also be a null reference (Nothing in Visual Basic).

Exceptions

ArgumentNullException

The `context` is a null reference (Nothing in Visual Basic).

ReadVariableValue(OpcReadVariableValueContext)

C#

```
public OpcVariableValue<object> ReadVariableValue(OpcReadVariableValueContext context)
```

Parameters

`context` OpcReadVariableValueContext

Returns

OpcVariableValue<Object>

Exceptions

ArgumentNullException

ReadVariableValueCore(OpcReadVariableValueContext, OpcVariableValue<Object>)

Reads the value of the variable node using the `context` and `value` information specified.

C#

```
protected virtual OpcVariableValue<object> ReadVariableValueCore(OpcReadVariableValueContext
context, OpcVariableValue<object> value)
```

Parameters

`context` OpcReadVariableValueContext

The OpcReadVariableValueContext to use to read the variable node value.

`value` OpcVariableValue<Object>

The OpcVariableValue´1 containing the currently used value constructed by the value information contained in the variable node cache.

Returns

OpcVariableValue<Object>

The OpcVariableValue´1 read using the ReadVariableValueCallback or the `value` if there is no custom

callback routine defined.

WriteAttributeValueCore<T>(OpcWriteAttributeValueContext, OpcAttributeValue<T>)

C#

```
protected override OpcAttributeValue<T>
WriteAttributeValueCore<T>(OpcWriteAttributeValueContext context, OpcAttributeValue<T>
value)
```

Parameters

`context` OpcWriteAttributeValueContext

`value` OpcAttributeValue<T>

Returns

OpcAttributeValue<T>

WriteValue(OpcWriteVariableValueContext, Object)

Writes the `value` to the variable node value using the `context` specified.

C#

```
public void WriteValue(OpcWriteVariableValueContext context, object value)
```

Parameters

`context` OpcWriteVariableValueContext

The OpcWriteVariableValueContext to use to write the variable node `value` specified.

`value` Object

The Object to write to the variable node value.

Exceptions

ArgumentNullException

The `context` is a null reference (Nothing in Visual Basic).

WriteVariableValue(OpcWriteVariableValueContext, OpcVariableValue<Object>)

C#

```
public void WriteVariableValue(OpcWriteVariableValueContext context,
OpcVariableValue<object> value)
```

Parameters

`context` OpcWriteVariableValueContext

`value` OpcVariableValue<Object>

Exceptions

ArgumentNullException

WriteVariableValueCore(OpcWriteVariableValueContext, OpcVariableValue<Object>)

Writes the value of the variable node using the `context` and `value` information specified.

C#

```
protected virtual OpcVariableValue<object>
WriteVariableValueCore(OpcWriteVariableValueContext context, OpcVariableValue<object> value)
```

Parameters

`context` OpcWriteVariableValueContext

The OpcWriteVariableValueContext to use to write the variable node value.

`value` OpcVariableValue<Object>

The OpcVariableValue´1 containing the currently used value constructed by the value information contained in the variable node cache.

Returns

OpcVariableValue<Object>

The OpcVariableValue´1 written using the WriteVariableValueCallback or the `value` if there is no custom callback routine defined.

Table of Contents