

OpcTwoStateVariableNode Members

Namespace: Opc.UaFx

Assemblies: Opc.UaFx.Advanced.dll, Opc.UaFx.Advanced.dll

The [OpcTwoStateVariableNode](#) type exposes the following members.

Constructors

OpcTwoStateVariableNode(IOpcNode, OpcName)

Initializes a new instance of the [OpcTwoStateVariableNode](#) class accessible by the **name** specified as a child node of the **parent** node given.

C#

```
public OpcTwoStateVariableNode(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 two state variable node can be accessed.

OpcTwoStateVariableNode(IOpcNode, OpcName, OpcNodeId)

Initializes a new instance of the [OpcTwoStateVariableNode](#) class accessible by the **name** and **id** specified as a child node of the **parent** node given.

C#

```
public OpcTwoStateVariableNode(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 two state variable node can be accessed.

id [OpcNodeId](#)

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

OpcTwoStateVariableNode(IOpcNode, OpcName, OpcNodeId, OpcText)

Initializes a new instance of the `OpcTwoStateVariableNode` 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#

```
public OpcTwoStateVariableNode(IOpcNode parent, OpcName name, OpcNodeId id, OpcText 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 two state variable node can be accessed.

`id` `OpcNodeId`

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

`value` `OpcText`

The initial value of the new two state variable node.

OpcTwoStateVariableNode(IOpcNode, OpcName, OpcText)

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

C#

```
public OpcTwoStateVariableNode(IOpcNode parent, OpcName name, OpcText 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 two state variable node can be accessed.

`value` `OpcText`

The initial value of the new two state variable node.

OpcTwoStateVariableNode(OpcName)

Initializes a new instance of the [OpcTwoStateVariableNode](#) class accessible by the **name** specified.

C#

```
public OpcTwoStateVariableNode(OpcName name)
```

Parameters

name [OpcName](#)

The [OpcName](#) through that the new two state variable node can be accessed.

OpcTwoStateVariableNode(OpcName, OpcNodeId)

Initializes a new instance of the [OpcTwoStateVariableNode](#) class accessible by the **name** and **id** specified.

C#

```
public OpcTwoStateVariableNode(OpcName name, OpcNodeId id)
```

Parameters

name [OpcName](#)

The [OpcName](#) through that the new two state variable node can be accessed.

id [OpcNodeId](#)

The [OpcNodeId](#) through that the new two state variable node can be identified and accessed.

OpcTwoStateVariableNode(OpcName, OpcNodeId, OpcText)

Initializes a new instance of the [OpcTwoStateVariableNode](#) class accessible by the **name** and **id** specified with the initial value given by **value**.

C#

```
public OpcTwoStateVariableNode(OpcName name, OpcNodeId id, OpcText value)
```

Parameters

name [OpcName](#)

The [OpcName](#) through that the new two state variable node can be accessed.

id [OpcNodeId](#)

The [OpcNodeId](#) through that the new two state variable node can be identified and accessed.

value [OpcText](#)

The initial value of the new two state variable node.

OpcTwoStateVariableNode(OpcName, OpcText)

Initializes a new instance of the [OpcTwoStateVariableNode](#) class accessible by the [name](#) specified with the initial value given by [value](#).

C#

```
public OpcTwoStateVariableNode(OpcName name, OpcText value)
```

Parameters

[name](#) [OpcName](#)

The [OpcName](#) through that the new two state variable node can be accessed.

value [OpcText](#)

The initial value of the new two state variable node.

Properties

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).

EffectiveTransitionTime

Gets or sets a value which specifies the time when the current state or one of its substates was entered.

C#

```
public DateTime EffectiveTransitionTime { get; set; }
```

Property Value

DateTime

The effective [DateTime](#) of the time the current state has been entered.

Remarks

If, for example, a level alarm is active and - while active - switches several times between high and highhigh, then the [TransitionTime](#) stays at the point in time where the alarm became active whereas the [EffectiveTransitionTime](#) changes with each shift of a sub state.

EffectiveTransitionTimeNode

Gets the [OpcPropertyNode`1](#) of the [EffectiveTransitionTime](#) property.

C#

```
public OpcPropertyNode<DateTime> EffectiveTransitionTimeNode { get; }
```

Property Value

OpcPropertyNode<DateTime>

An instance of the [OpcPropertyNode`1](#) class.

FalseState

Gets or sets a value which contains a human readable representation for the variable value when its [VariableId](#) property has the value true.

C#

```
public OpcText FalseState { get; set; }
```

Property Value

OpcText

The human readable representation for the value true.

FalseStateNode

Gets the [OpcTextPropertyNode](#) of the [FalseState](#) property.

C#

```
public OpcTextPropertyNode FalseStateNode { get; }
```

Property Value

OpcTextPropertyNode

An instance of the [OpcTextPropertyNode](#) class.

TransitionTime

Gets or sets a value which specifies when the current state was entered.

C#

```
public DateTime TransitionTime { get; set; }
```

Property Value

[DateTime](#)

The [DateTime](#) of the time the current state was entered.

TransitionTimeNode

Gets the [OpcPropertyNode`1](#) of the [TransitionTime](#) property.

C#

```
public OpcPropertyNode<DateTime> TransitionTimeNode { get; }
```

Property Value

[OpcPropertyNode<DateTime>](#)

An instance of the [OpcPropertyNode`1](#) class.

TrueState

Gets or sets a value which contains a human readable representation for the variable value when its [VariableId](#) property has the value true.

C#

```
public OpcText TrueState { get; set; }
```

Property Value

[OpcText](#)

The human readable representation for the value true.

TrueStateNode

Gets the [OpcTextPropertyNode](#) of the [TrueState](#) property.

C#

```
public OpcTextPropertyNode TrueStateNode { get; }
```

Property Value

[OpcTextPropertyNode](#)

An instance of the [OpcTextPropertyNode](#) class.

VariableId

Gets or sets a node identifier which uniquely identifies the current state within the state machine.

C#

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

Property Value

[Boolean](#)

A [Boolean](#) which uniquely identifies the current state within the state machine.

Methods

InitializeDefaults()

Initializes the default values used by the [OpcTwoStateVariableNode](#).

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](#).

Table of Contents

Constructors	1
OpcTwoStateVariableNode(IOpcNode, OpcName)	1
OpcTwoStateVariableNode(IOpcNode, OpcName, OpcNodeId)	1
OpcTwoStateVariableNode(IOpcNode, OpcName, OpcNodeId, OpcText)	2
OpcTwoStateVariableNode(IOpcNode, OpcName, OpcText)	2
OpcTwoStateVariableNode(OpcName)	3
OpcTwoStateVariableNode(OpcName, OpcNodeId)	3
OpcTwoStateVariableNode(OpcName, OpcNodeId, OpcText)	3
OpcTwoStateVariableNode(OpcName, OpcText)	4
Properties	4
DefaultTypeDefinitionId	4
EffectiveTransitionTime	4
EffectiveTransitionTimeNode	5
FalseState	5
FalseStateNode	5
TransitionTime	6
TransitionTimeNode	6
TrueState	6
TrueStateNode	6
VariableId	7
Methods	7
InitializeDefaults()	7