

OpcEngineeringUnitInfoPropertyNode Members

Namespace: Opc.UaFx

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

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

Constructors

OpcEngineeringUnitInfoPropertyNode(IOpcNode, OpcName)

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

C#

```
public OpcEngineeringUnitInfoPropertyNode(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 engineering unit info node can be accessed.

OpcEngineeringUnitInfoPropertyNode(IOpcNode, OpcName, OpcEngineeringUnitInfo)

Initializes a new instance of the [OpcEngineeringUnitInfoPropertyNode](#) class accessible by the **name** specified with the initial value defined by **value** as a child node of the **parent** node given.

C#

```
public OpcEngineeringUnitInfoPropertyNode(IOpcNode parent, OpcName name, OpcEngineeringUnitInfo 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 engineering unit info node can be accessed.

value OpcEngineeringUnitInfo

The initial value of the new engineering unit info node.

OpcEngineeringUnitInfoPropertyNode(IOpcNode, OpcName, OpcNodeId)

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

C#

```
public OpcEngineeringUnitInfoPropertyNode(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 engineering unit info node can be accessed.

id OpcNodeId

The **OpcNodeId** through that the new engineering unit info node can be identified and accessed.

OpcEngineeringUnitInfoPropertyNode(IOpcNode, OpcName, OpcNodeId, OpcEngineeringUnitInfo)

Initializes a new instance of the **OpcEngineeringUnitInfoPropertyNode** 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 OpcEngineeringUnitInfoPropertyNode(IOpcNode parent, OpcName name, OpcNodeId id, OpcEngineeringUnitInfo 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 engineering unit info node can be accessed.

id `OpcNodeId`

The `OpcNodeId` through that the new engineering unit info node can be identified and accessed.

value `OpcEngineeringUnitInfo`

The initial value of the new engineering unit info node.

OpcEngineeringUnitInfoPropertyNode(OpcName)

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

C#

```
public OpcEngineeringUnitInfoPropertyNode(OpcName name)
```

Parameters

name `OpcName`

The `OpcName` through that the new engineering unit info node can be accessed.

OpcEngineeringUnitInfoPropertyNode(OpcName, OpcEngineeringUnitInfo)

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

C#

```
public OpcEngineeringUnitInfoPropertyNode(OpcName name, OpcEngineeringUnitInfo value)
```

Parameters

name `OpcName`

The `OpcName` through that the new engineering unit info node can be accessed.

value `OpcEngineeringUnitInfo`

The initial value of the new engineering unit info node.

OpcEngineeringUnitInfoPropertyNode(OpcName, OpcNodeId)

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

C#

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

Parameters

name [OpcName](#)

The [OpcName](#) through that the new engineering unit info node can be accessed.

id [OpcNodeId](#)

The [OpcNodeId](#) through that the new engineering unit info node can be identified and accessed.

OpcEngineeringUnitInfoPropertyNode(OpcName, OpcNodeId, OpcEngineeringUnitInfo)

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

C#

```
public OpcEngineeringUnitInfoPropertyNode(OpcName name, OpcNodeId id, OpcEngineeringUnitInfo value)
```

Parameters

name [OpcName](#)

The [OpcName](#) through that the new engineering unit info node can be accessed.

id [OpcNodeId](#)

The [OpcNodeId](#) through that the new engineering unit info node can be identified and accessed.

value [OpcEngineeringUnitInfo](#)

The initial value of the new engineering unit info node.

Properties

ReadEngineeringUnitInfoPropertyValueCallback

Gets or sets a callback used to read the engineering unit info value.

C#

```
public OpcReadEngineeringUnitInfoPropertyValueCallback  
ReadEngineeringUnitInfoPropertyValueCallback { get; set; }
```

Property Value

[OpcReadEngineeringUnitInfoPropertyValueCallback](#)

A [OpcReadEngineeringUnitInfoPropertyValueCallback](#) used to read the engineering unit info property value as the type [OpcEngineeringUnitInfo](#). The value can also be a null reference (Nothing in Visual Basic).

Remarks

This callback is used in call chain with the [ReadPropertyValueCallback](#) and will be called with the outcome of that callback routine (in case there is one defined).

Value

Gets or sets the value of the property node.

C#

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

Property Value

[OpcEngineeringUnitInfo](#)

A [OpcEngineeringUnitInfo](#) representing the value of the property node. This can be also a null reference (Nothing in Visual Basic).

WriteEngineeringUnitInfoPropertyValueCallback

Gets or sets a callback used to write the engineering unit info value.

C#

```
public OpcWriteEngineeringUnitInfoPropertyValueCallback  
WriteEngineeringUnitInfoPropertyValueCallback { get; set; }
```

Property Value

[OpcWriteEngineeringUnitInfoPropertyValueCallback](#)

A [OpcWriteEngineeringUnitInfoPropertyValueCallback](#) used to write the engineering unit info property value as the type [OpcEngineeringUnitInfo](#). The value can also be a null reference (Nothing in Visual Basic).

Remarks

This callback is used in call chain with the [WritePropertyValueCallback](#) and will be called with the outcome of that callback routine (in case there is one defined).

Methods

[ReadEngineeringUnitInfoPropertyValueCore\(OpcReadPropertyContext<OpcEngineeringUnitInfoPropertyNode](#)

>, OpcPropertyValue<OpcEngineeringUnitInfo>)

Reads the engineering unit info value of the property node using the **context** and **value** information specified.

C#

```
protected virtual OpcPropertyValue<OpcEngineeringUnitInfo>
ReadEngineeringUnitInfoPropertyValueCore(OpcReadPropertyValueContext<OpcEngineeringUnitInfoP
ropertyNode> context, OpcPropertyValue<OpcEngineeringUnitInfo> value)
```

Parameters

context OpcReadPropertyValueContext<OpcEngineeringUnitInfoPropertyNode>

The OpcReadPropertyValueContext`1 to use to read the property node value.

value OpcPropertyValue<OpcEngineeringUnitInfo>

The OpcPropertyValue`1 containing the currently used value constructed by the value information contained in the property node cache.

Returns

OpcPropertyValue<OpcEngineeringUnitInfo>

The OpcPropertyValue`1 read using the ReadEngineeringUnitInfoPropertyValueCallback or the **value** if there is no custom callback routine defined.

ReadEngineeringUnitInfoValue(OpcReadVariableValueContext)

Reads the engineering unit info value of the property node.

C#

```
public OpcEngineeringUnitInfo ReadEngineeringUnitInfoValue(OpcReadVariableValueContext
context)
```

Parameters

context OpcReadVariableValueContext

The OpcReadVariableValueContext to use when reading the value.

Returns

OpcEngineeringUnitInfo

The OpcEngineeringUnitInfo value of the engineering unit info node read.

Exceptions

ArgumentNullException

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

ReadPropertyValueCore(OpcReadPropertyValueContext, OpcPropertyValue<Object>)

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

C#

```
protected override sealed OpcPropertyValue<object>
ReadPropertyValueCore(OpcReadPropertyValueContext context, OpcPropertyValue<object> value)
```

Parameters

`context` OpcReadPropertyValueContext

The OpcReadPropertyValueContext to use to read the property node value.

`value` OpcPropertyValue<Object>

The OpcPropertyValue'1 containing the currently used value constructed by the value information contained in the property node cache.

Returns

OpcPropertyValue<Object>

The OpcPropertyValue'1 read using the ReadPropertyValueCallback or the `value` if there is no custom callback routine defined.

WriteEngineeringUnitInfoPropertyValueCore(OpcWritePropertyValueContext<OpcEngineeringUnitInfoPropertyNode>, OpcPropertyValue<OpcEngineeringUnitInfo>)

Writes the engineering unit info value of the property node using the `context` and `value` information specified.

C#

```
protected virtual OpcPropertyValue<OpcEngineeringUnitInfo>
WriteEngineeringUnitInfoPropertyValueCore(OpcWritePropertyValueContext<OpcEngineeringUnitInfoPropertyNode> context, OpcPropertyValue<OpcEngineeringUnitInfo> value)
```

Parameters

`context` OpcWritePropertyValueContext<OpcEngineeringUnitInfoPropertyNode>

The OpcWritePropertyValueContext'1 to use to write the property node value.

`value` `OpcPropertyValue<OpcEngineeringUnitInfo>`

The `OpcPropertyValue`1` containing the currently used value constructed by the value information contained in the property node cache.

Returns

`OpcPropertyValue<OpcEngineeringUnitInfo>`

The `OpcPropertyValue`1` written using the `WriteEngineeringUnitInfoPropertyValueCallback` or the `value` if there is no custom callback routine defined.

WriteEngineeringUnitInfoValue(OpcWriteVariableValueContext, OpcEngineeringUnitInfo)

Writes the value of the property node specified by `value`.

C#

```
public void WriteEngineeringUnitInfoValue(OpcWriteVariableValueContext context,
OpcEngineeringUnitInfo value)
```

Parameters

`context` `OpcWriteVariableValueContext`

The `OpcWriteVariableValueContext` to use when writing the value.

`value` `OpcEngineeringUnitInfo`

The `OpcEngineeringUnitInfo` value of the engineering unit info property node to write.

Exceptions

`ArgumentNullException`

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

WritePropertyValueCore(OpcWritePropertyValueContext, OpcPropertyValue<Object>)

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

C#

```
protected override sealed OpcPropertyValue<object>
WritePropertyValueCore(OpcWritePropertyValueContext context, OpcPropertyValue<object> value)
```

Parameters

context `OpcWritePropertyValueContext`

The `OpcWritePropertyValueContext` to use to write the property node value.

value `OpcPropertyValue<Object>`

The `OpcPropertyValue`1` containing the currently used value constructed by the value information contained in the property node cache.

Returns

`OpcPropertyValue<Object>`

The `OpcPropertyValue`1` written using the `WritePropertyValueCallback` or the `value` if there is no custom callback routine defined.

Table of Contents

Constructors	1
OpcEngineeringUnitInfoPropertyNode(IOpcNode, OpcName)	1
OpcEngineeringUnitInfoPropertyNode(IOpcNode, OpcName, OpcEngineeringUnitInfo)	1
OpcEngineeringUnitInfoPropertyNode(IOpcNode, OpcName, OpcNodeId)	2
OpcEngineeringUnitInfoPropertyNode(IOpcNode, OpcName, OpcNodeId, OpcEngineeringUnitInfo)	2
OpcEngineeringUnitInfoPropertyNode(OpcName)	3
OpcEngineeringUnitInfoPropertyNode(OpcName, OpcEngineeringUnitInfo)	3
OpcEngineeringUnitInfoPropertyNode(OpcName, OpcNodeId)	3
OpcEngineeringUnitInfoPropertyNode(OpcName, OpcNodeId, OpcEngineeringUnitInfo)	4
Properties	4
ReadEngineeringUnitInfoPropertyValueCallback	4
Value	5
WriteEngineeringUnitInfoPropertyValueCallback	5
Methods	5
ReadEngineeringUnitInfoPropertyValueCore(OpcReadPropertyValueContext<OpcEngineeringUnitInfoPropertyNode>, OpcPropertyValue<OpcEngineeringUnitInfo>)	5
ReadEngineeringUnitInfoValue(OpcReadVariableValueContext)	6
ReadPropertyValueCore(OpcReadPropertyValueContext, OpcPropertyValue<Object>)	7
WriteEngineeringUnitInfoPropertyValueCore(OpcWritePropertyValueContext<OpcEngineeringUnitInfoPropertyNode>, OpcPropertyValue<OpcEngineeringUnitInfo>)	7
WriteEngineeringUnitInfoValue(OpcWriteVariableValueContext, OpcEngineeringUnitInfo)	8
WritePropertyValueCore(OpcWritePropertyValueContext, OpcPropertyValue<Object>)	8