

OpcAddAnalogItemNode<T> Members

Namespace: Opc.UaFx

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

The [OpcAddAnalogItemNode<T>](#) type exposes the following members.

Constructors

OpcAddAnalogItemNode(OpcName)

Initializes a new instance of the [OpcAddAnalogItemNode`1](#) class using the [name](#) of the analog item node to add. The according [OpcNodeId](#) to identify and access the new node is determined by the service. The new node will be a child of the [ObjectsFolder](#) node using [HasComponent](#) as the type of reference.

C#

```
public OpcAddAnalogItemNode(OpcName name)
```

Parameters

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

The [OpcName](#) through that the new analog item node can be accessed.

Exceptions

[ArgumentException](#)

The [name](#) is equals [Null](#).

[ArgumentNullException](#)

The [name](#) is a null reference (Nothing in Visual Basic).

OpcAddAnalogItemNode(OpcName, OpcNodeId)

Initializes a new instance of the [OpcAddAnalogItemNode`1](#) class using the [name](#) of the analog item node to add, which shall be additionally accessible by the [nodeId](#) defined. The new node will be a child of the [ObjectsFolder](#) node using [HasComponent](#) as the type of reference.

C#

```
public OpcAddAnalogItemNode(OpcName name, OpcNodeId nodeId)
```

Parameters

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

The **OpcName** through that the new analog item node can be accessed.

nodeId **OpcNodeId**

The **OpcNodeId** through that the new node can be identified and accessed. In case there **Null** is specified the server will determine the according **OpcNodeId** by its own.

Exceptions

ArgumentException

The **name** is equals **Null**.

ArgumentNullException

The **name** or **nodeId** is a null reference (Nothing in Visual Basic).

OpcAddAnalogItemNode(OpcName, OpcNodeId, OpcNodeId)

Initializes a new instance of the **OpcAddAnalogItemNode`1** class using the **name** of the analog item node to add, which shall be additionally accessible by the **nodeId** defined. The new node will be a child of the node identified by **parentNodeId** using **HasComponent** as the type of reference.

C#

```
public OpcAddAnalogItemNode(OpcName name, OpcNodeId nodeId, OpcNodeId parentNodeId)
```

Parameters

name **OpcName**

The **OpcName** through that the new analog item node can be accessed.

nodeId **OpcNodeId**

The **OpcNodeId** through that the new node can be identified and accessed. In case there **Null** is specified the server will determine the according **OpcNodeId** by its own.

parentNodeId **OpcNodeId**

The **OpcNodeId** of the parent node to reference using **HasComponent** as the type of reference.

Exceptions

ArgumentException

The **parentNodeId** is equals **Null** or **name** is equals **Null**.

ArgumentNullException

The **name**, **nodeId** or **parentNodeId** is a null reference (Nothing in Visual Basic).

OpcAddAnalogItemNode(OpcName, OpcNodeId, OpcNodeId, OpcNodeId)

Initializes a new instance of the [OpcAddAnalogItemNode`1](#) class using the [name](#) of the analog item node to add, which shall be additionally accessible by the [nodeId](#) defined. The new node will be a child of the node identified by [parentNodeId](#) using the type of reference identified by the [referenceTypeId](#) specified.

C#

```
public OpcAddAnalogItemNode(OpcName name, OpcNodeId nodeId, OpcNodeId parentNodeId, OpcNodeId referenceTypeId)
```

Parameters

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

The [OpcName](#) through that the new analog item node can be accessed.

[nodeId](#) [OpcNodeId](#)

The [OpcNodeId](#) through that the new node can be identified and accessed. In case there [Null](#) is specified the server will determine the according [OpcNodeId](#) by its own.

[parentNodeId](#) [OpcNodeId](#)

The [OpcNodeId](#) of the parent node to reference using the type of reference identified by [referenceTypeId](#).

[referenceTypeId](#) [OpcNodeId](#)

The [OpcNodeId](#) which identifies the type of reference to use for the new node and the existing parent node identified by [parentNodeId](#).

Exceptions

[ArgumentException](#)

The [parentNodeId](#) or [referenceTypeId](#) is equals [Null](#) or [name](#) is equals [Null](#).

[ArgumentNullException](#)

The [name](#), [nodeId](#), [parentNodeId](#) or [referenceTypeId](#) is a null reference (Nothing in Visual Basic).

OpcAddAnalogItemNode(OpcName, OpcNodeId, OpcNodeId, T)

Initializes a new instance of the [OpcAddAnalogItemNode`1](#) class using the [name](#) of the analog item node to add, which shall be additionally accessible by the [nodeId](#) defined. The new node will be a child of the node identified by [parentNodeId](#) using the type of reference identified by the [referenceTypeId](#) specified.

C#

```
public OpcAddAnalogItemNode(OpcName name, OpcNodeId nodeId, OpcNodeId parentNodeId, OpcNodeId referenceTypeId, T value)
```

Parameters

name `OpcName`

The `OpcName` through that the new analog item node can be accessed.

nodeId `OpcNodeId`

The `OpcNodeId` through that the new node can be identified and accessed. In case there `Null` is specified the server will determine the according `OpcNodeId` by its own.

parentNodeId `OpcNodeId`

The `OpcNodeId` of the parent node to reference using the type of reference identified by `referenceTypeId`.

referenceTypeId `OpcNodeId`

The `OpcNodeId` which identifies the type of reference to use for the new node and the existing parent node identified by `parentNodeId`.

value `T`

The initial value of the new analog item node.

Exceptions

`ArgumentException`

The `parentNodeId` or `referenceTypeId` is equals `Null` or `name` is equals `Null`.

`ArgumentNullException`

The `name`, `nodeId`, `parentNodeId` or `referenceTypeId` is a null reference (Nothing in Visual Basic).

OpcAddAnalogItemNode(OpcName, OpcNodeId, OpcNodeId, OpcReferenceType)

Initializes a new instance of the `OpcAddAnalogItemNode`1` class using the `name` of the analog item node to add, which shall be additionally accessible by the `nodeId` defined. The new node will be a child of the node identified by `parentNodeId` using the `referenceType` specified as the type of reference.

C#

```
public OpcAddAnalogItemNode(OpcName name, OpcNodeId nodeId, OpcNodeId parentNodeId,
OpcReferenceType referenceType)
```

Parameters

name `OpcName`

The `OpcName` through that the new analog item node can be accessed.

nodeId `OpcNodeId`

The `OpcNodeId` through that the new node can be identified and accessed. In case there `Null` is specified the server will determine the according `OpcNodeId` by its own.

parentNodeId [OpcNodeId](#)

The [OpcNodeId](#) of the parent node to reference using [HasComponent](#) as the type of reference.

referenceType [OpcReferenceType](#)

One of the members defined by the [OpcReferenceType](#) enumeration to use to setup the reference between the new node and the existing parent node identified by **parentNodeId**.

Exceptions

[ArgumentException](#)

The **parentNodeId** is equals [Null](#) or **name** is equals [Null](#).

[ArgumentNullException](#)

The **name**, **nodeId** or **parentNodeId** is a null reference (Nothing in Visual Basic).

OpcAddAnalogItemNode(OpcName, OpcNodeId, OpcNodeId, OpcReferenceType, T)

Initializes a new instance of the [OpcAddAnalogItemNode`1](#) class using the **name** of the analog item node to add, which shall be additionally accessible by the **nodeId** defined. The new node will be a child of the node identified by **parentNodeId** using the **referenceType** specified as the type of reference.

C#

```
public OpcAddAnalogItemNode(OpcName name, OpcNodeId nodeId, OpcNodeId parentNodeId,
    OpcReferenceType referenceType, T value)
```

Parameters

name [OpcName](#)

The [OpcName](#) through that the new analog item node can be accessed.

nodeId [OpcNodeId](#)

The [OpcNodeId](#) through that the new node can be identified and accessed. In case there [Null](#) is specified the server will determine the according [OpcNodeId](#) by its own.

parentNodeId [OpcNodeId](#)

The [OpcNodeId](#) of the parent node to reference using [HasComponent](#) as the type of reference.

referenceType [OpcReferenceType](#)

One of the members defined by the [OpcReferenceType](#) enumeration to use to setup the reference between the new node and the existing parent node identified by **parentNodeId**.

value [T](#)

The initial value of the new analog item node.

Exceptions

ArgumentException

The `parentNodeId` is equals `Null` or `name` is equals `Null`.

ArgumentNullException

The `name`, `nodeId` or `parentNodeId` is a null reference (Nothing in Visual Basic).

OpcAddAnalogItemNode(OpcName, OpcNodeId, OpcNodeId, T)

Initializes a new instance of the `OpcAddAnalogItemNode`1` class using the `name` of the analog item node to add, which shall be additionally accessible by the `nodeId` defined. The new node will be a child of the node identified by `parentNodeId` using `HasComponent` as the type of reference.

C#

```
public OpcAddAnalogItemNode(OpcName name, OpcNodeId nodeId, OpcNodeId parentNodeId, T value)
```

Parameters

`name` `OpcName`

The `OpcName` through that the new analog item node can be accessed.

`nodeId` `OpcNodeId`

The `OpcNodeId` through that the new node can be identified and accessed. In case there `Null` is specified the server will determine the according `OpcNodeId` by its own.

`parentNodeId` `OpcNodeId`

The `OpcNodeId` of the parent node to reference using `HasComponent` as the type of reference.

`value` `T`

The initial value of the new analog item node.

Exceptions

ArgumentException

The `parentNodeId` is equals `Null` or `name` is equals `Null`.

ArgumentNullException

The `name`, `nodeId` or `parentNodeId` is a null reference (Nothing in Visual Basic).

OpcAddAnalogItemNode(OpcName, OpcNodeId, T)

Initializes a new instance of the `OpcAddAnalogItemNode`1` class using the `name` of the analog item node to add, which shall be additionally accessible by the `nodeId` defined. The new node will be a child of the `ObjectsFolder` node using `HasComponent` as the type of reference.

C#

```
public OpcAddAnalogItemNode(OpcName name, OpcNodeId nodeId, T value)
```

Parameters

name [OpcName](#)

The [OpcName](#) through that the new analog item node can be accessed.

nodeId [OpcNodeId](#)

The [OpcNodeId](#) through that the new node can be identified and accessed. In case there [Null](#) is specified the server will determine the according [OpcNodeId](#) by its own.

value [T](#)

The initial value of the new analog item node.

Exceptions

[ArgumentException](#)

The **name** is equals [Null](#).

[ArgumentNullException](#)

The **name** or **nodeId** is a null reference (Nothing in Visual Basic).

OpcAddAnalogItemNode(OpcName, T)

Initializes a new instance of the [OpcAddAnalogItemNode`1](#) class using the **name** of the analog item node to add. The according [OpcNodeId](#) to identify and access the new node is determined by the service. The new node will be a child of the [ObjectsFolder](#) node using [HasComponent](#) as the type of reference.

C#

```
public OpcAddAnalogItemNode(OpcName name, T value)
```

Parameters

name [OpcName](#)

The [OpcName](#) through that the new analog item node can be accessed.

value [T](#)

The initial value of the new analog item node.

Exceptions

[ArgumentException](#)

The **name** is equals [Null](#).

[ArgumentNullException](#)

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

Properties

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. A null reference (Nothing in Visual Basic) indicates that the attribute is undefined and its default value is used.

C#

```
public override OpcDataType? DataType { get; set; }
```

Property Value

`Nullable<OpcDataType>`

One of the members defined by the `OpcDataType` enumeration.

Value

Gets or sets the value of the data variable node. A null reference (Nothing in Visual Basic) indicates that the attribute is undefined and its default value is used.

C#

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

Property Value

`T`

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

Table of Contents

Constructors	1
OpcAddAnalogItemNode(OpcName)	1
OpcAddAnalogItemNode(OpcName, OpcNodeId)	1
OpcAddAnalogItemNode(OpcName, OpcNodeId, OpcNodeId)	2
OpcAddAnalogItemNode(OpcName, OpcNodeId, OpcNodeId, OpcNodeId)	3
OpcAddAnalogItemNode(OpcName, OpcNodeId, OpcNodeId, OpcNodeId, T)	3
OpcAddAnalogItemNode(OpcName, OpcNodeId, OpcNodeId, OpcReferenceType)	4
OpcAddAnalogItemNode(OpcName, OpcNodeId, OpcNodeId, OpcReferenceType, T)	5
OpcAddAnalogItemNode(OpcName, OpcNodeId, OpcNodeId, T)	6
OpcAddAnalogItemNode(OpcName, OpcNodeId, T)	6
OpcAddAnalogItemNode(OpcName, T)	7
Properties	8
DataType	8
Value	8

