

# OpcAddNode Members

**Namespace:** Opc.UaFx

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

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

## Constructors

### OpcAddNode(OpcNodeCategory, OpcName)

Initializes a new instance of the [OpcAddNode](#) class using the specified **category** of node to add, which shall be accessible by the **name** defined. The according [OpcNodeId](#) to identify and access the new node is determined by the service.

**C#**

```
protected OpcAddNode(OpcNodeCategory category, OpcName name)
```

#### Parameters

**category** [OpcNodeCategory](#)

The [OpcNodeCategory](#) which classifies the new node to add regarding one of the classifications defined by the [OpcNodeCategory](#) enumeration.

**name** [OpcName](#)

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

#### Exceptions

[ArgumentException](#)

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

[ArgumentNullException](#)

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

### OpcAddNode(OpcNodeCategory, OpcName, OpcNodeId)

Initializes a new instance of the [OpcAddNode](#) class using the specified **category** of node to add, which shall be accessible by the **name** and **nodeId** defined. The new node will be a child of the [ObjectsFolder](#) node.

**C#**

```
protected OpcAddNode(OpcNodeCategory category, OpcName name, OpcNodeId nodeId)
```

#### Parameters

### category `OpcNodeCategory`

The `OpcNodeCategory` which classifies the new node to add regarding one of the classifications defined by the `OpcNodeCategory` enumeration.

### name `OpcName`

The `OpcName` through that the new 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).

## `OpcAddNode(OpcNodeCategory, OpcName, OpcNodeId, OpcNodeId)`

Initializes a new instance of the `OpcAddNode` class using the specified `category` of node to add, which shall be accessible by the `name` and `nodeId` defined. The new node will be a child of the node identified by `parentNodeId`.

## C#

```
protected OpcAddNode(OpcNodeCategory category, OpcName name, OpcNodeId nodeId, OpcNodeId
parentNodeId)
```

## Parameters

### category `OpcNodeCategory`

The `OpcNodeCategory` which classifies the new node to add regarding one of the classifications defined by the `OpcNodeCategory` enumeration.

### name `OpcName`

The `OpcName` through that the new 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.

## 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).

# Properties

## Category

Gets a value indicating the classification of the node in the address space.

### C#

```
public OpcNodeCategory Category { get; }
```

### Property Value

#### OpcNodeCategory

One of the members defined by the `OpcNodeCategory` enumeration which classifies the node within the address space.

## Description

Gets or sets the localized description of the meaning of the node.

### C#

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

### Property Value

#### OpcText

An instance of the `OpcText` class which provides the localized description of the node or a null reference (Nothing in Visual Basic). In case of a null reference (Nothing in Visual Basic) the server defined default value of the attribute is used.

## DisplayName

Gets or sets the localized name of the node.

### C#

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

## Property Value

### OpcText

An instance of the [OpcText](#) class which provides the localized name of the node or a null reference (Nothing in Visual Basic). In case of a null reference (Nothing in Visual Basic) the server defined default value of the attribute is used.

## Name

Gets the non-localised human-readable name of the node in the address space.

### C#

```
public OpcName Name { get; }
```

## Property Value

### OpcName

An instance of the [OpcName](#) class used to identify a node when browsing the address space.

## ParentNodeId

Gets the identifier of the existing parent node of the new node.

### C#

```
public OpcNodeId ParentNodeId { get; }
```

## Property Value

### OpcNodeId

An instance of the [OpcNodeId](#) which identifies an existing node to use as the parent node of the new node.

## SupportsNullNodeId

Gets a value indicating whether the [OpcAddNode](#) supports instances of the [OpcNodeId](#) class its [IsNull](#) provides a value equals to the value true.

### C#

```
protected override bool SupportsNullNodeId { get; }
```

## Property Value

### Boolean

The value true in the case there the [OpcNodeId](#) instance of the command can be a null-node-id; otherwise the value false.

## Remarks

In case there this property provides the value false and an instance of the [OpcNodeId](#) is passed to the constructor of this class an [ArgumentException](#) will be thrown.

## UserWriteAccess

Gets or sets a value which exposes the possibilities of a client to write the attributes of the node taking user access rights into account.

### C#

```
public OpcAttributeWriteAccess? UserWriteAccess { get; set; }
```

### Property Value

[Nullable<OpcAttributeWriteAccess>](#)

One of the members defined by the [OpcAttributeWriteAccess](#) enumeration if specified; otherwise a null reference (Nothing in Visual Basic). In case of a null reference (Nothing in Visual Basic) the server defined default value of the attribute is used.

## WriteAccess

Gets or sets a value which exposes the possibilities of a client to write the attributes of the node without taking user access rights into account.

### C#

```
public OpcAttributeWriteAccess? WriteAccess { get; set; }
```

### Property Value

[Nullable<OpcAttributeWriteAccess>](#)

One of the members defined by the [OpcAttributeWriteAccess](#) enumeration if specified; otherwise a null reference (Nothing in Visual Basic). In case of a null reference (Nothing in Visual Basic) the server defined default value of the attribute is used.



# Table of Contents

<b>Constructors</b>	1
OpcAddNode(OpcNodeCategory, OpcName)	1
OpcAddNode(OpcNodeCategory, OpcName, OpcNodeId)	1
OpcAddNode(OpcNodeCategory, OpcName, OpcNodeId, OpcNodeId)	2
<b>Properties</b>	3
Category	3
Description	3
DisplayName	3
Name	4
ParentNodeId	4
SupportsNullNodeId	4
UserWriteAccess	5
WriteAccess	5