# OpcNodeServiceCommand Members

**Namespace:** Opc.UaFx.Services
**Assemblies:** Opc.UaFx.Advanced.dll, Opc.UaFx.Advanced.dll
The OpcNodeServiceCommand type exposes the following members.

# Constructors

## OpcNodeServiceCommand(Byte[])

Initializes a new instance of the OpcNodeServiceCommand class using the `nodeId` specified.

**C#**

```
protected OpcNodeServiceCommand(byte[] nodeId)
```

**Parameters**

`nodeId` Byte[]

The opaque node identifier of the node on which the service, who will execute this command, will operate.

## OpcNodeServiceCommand(Byte[], Int32)

Initializes a new instance of the OpcNodeServiceCommand class using the `nodeId` and `namespaceIndex` specified.

**C#**

```
protected OpcNodeServiceCommand(byte[] nodeId, int namespaceIndex)
```

**Parameters**

`nodeId` Byte[]

The opaque node identifier of the node on which the service, who will execute this command, will operate.

`namespaceIndex` Int32

The index of the namespace within that the node with the `nodeId` specified can be located.

## OpcNodeServiceCommand(Byte[], OpcNamespace)

Initializes a new instance of the OpcNodeServiceCommand class using the `nodeId` and `nodeNamespace` specified.

**C#**

```
protected OpcNodeServiceCommand(byte[] nodeId, OpcNamespace nodeNamespace)
```

**Parameters**

`nodeId` Byte[]

The opaque node identifier of the node on which the service, who will execute this command, will operate.

`nodeNamespace` OpcNamespace

The namespace within that the node with the `nodeId` specified can be located.

# OpcNodeServiceCommand(Guid)

Initializes a new instance of the OpcNodeServiceCommand class using the `nodeId` specified.

**C#**

```
protected OpcNodeServiceCommand(Guid nodeId)
```

**Parameters**

`nodeId` Guid

The general unique identifier (= GUID) of the node on which the service, who will execute this command, will operate.

# OpcNodeServiceCommand(Guid, Int32)

Initializes a new instance of the OpcNodeServiceCommand class using the `nodeId` and `namespaceIndex` specified.

**C#**

```
protected OpcNodeServiceCommand(Guid nodeId, int namespaceIndex)
```

**Parameters**

`nodeId` Guid

The general unique identifier (= GUID) of the node on which the service, who will execute this command, will operate.

`namespaceIndex` Int32

The index of the namespace within that the node with the `nodeId` specified can be located.

# OpcNodeServiceCommand(Guid, OpcNamespace)

Initializes a new instance of the OpcNodeServiceCommand class using the `nodeId` and `nodeNamespace` specified.

**C#**

```
protected OpcNodeServiceCommand(Guid nodeId, OpcNamespace nodeNamespace)
```

**Parameters**

`nodeId` Guid

The general unique identifier (= GUID) of the node on which the service, who will execute this command, will operate.

`nodeNamespace` OpcNamespace

The namespace within that the node with the `nodeId` specified can be located.

# OpcNodeServiceCommand(Int32)

Initializes a new instance of the OpcNodeServiceCommand class using the `nodeId` specified.

**C#**

```
protected OpcNodeServiceCommand(int nodeId)
```

**Parameters**

`nodeId` Int32

The numeric node identifier of the node on which the service, who will execute this command, will operate.

# OpcNodeServiceCommand(Int32, Int32)

Initializes a new instance of the OpcNodeServiceCommand class using the `nodeId` and `namespaceIndex` specified.

**C#**

```
protected OpcNodeServiceCommand(int nodeId, int namespaceIndex)
```

**Parameters**

`nodeId` Int32

The numeric node identifier of the node on which the service, who will execute this command, will operate.

`namespaceIndex` Int32

The index of the namespace within that the node with the `nodeId` specified can be located.

# OpcNodeServiceCommand(Int32, OpcNamespace)

Initializes a new instance of the OpcNodeServiceCommand class using the `nodeId` and `nodeNamespace` specified.

**C#**

```
protected OpcNodeServiceCommand(int nodeId, OpcNamespace nodeNamespace)
```

**Parameters**

`nodeId` Int32

The numeric node identifier of the node on which the service, who will execute this command, will operate.

`nodeNamespace` OpcNamespace

The namespace within that the node with the `nodeId` specified can be located.

# OpcNodeServiceCommand(OpcNodeId)

Initializes a new instance of the OpcNodeServiceCommand class using the `nodeId` specified.

**C#**

```
protected OpcNodeServiceCommand(OpcNodeId nodeId)
```

**Parameters**

`nodeId` OpcNodeId

The OpcNodeId of the node on which the service, who will execute this command, will operate.

**Exceptions**

ArgumentException

The command does not support empty node identifiers.

ArgumentNullException

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

# OpcNodeServiceCommand(String)

Initializes a new instance of the OpcNodeServiceCommand class using the `nodeId` specified.

**C#**

```
protected OpcNodeServiceCommand(string nodeId)
```

**Parameters**

`nodeId` String

The textual node identifier of the node on which the service, who will execute this command, will operate.

# OpcNodeServiceCommand(String, Int32)

Initializes a new instance of the OpcNodeServiceCommand class using the `nodeId` and `namespaceIndex` specified.

**C#**

```
protected OpcNodeServiceCommand(string nodeId, int namespaceIndex)
```

**Parameters**

`nodeId` String

The textual node identifier of the node on which the service, who will execute this command, will operate.

`namespaceIndex` Int32

The index of the namespace within that the node with the `nodeId` specified can be located.

# OpcNodeServiceCommand(String, OpcNamespace)

Initializes a new instance of the OpcNodeServiceCommand class using the `nodeId` and `nodeNamespace` specified.

**C#**

```
protected OpcNodeServiceCommand(string nodeId, OpcNamespace nodeNamespace)
```

**Parameters**

`nodeId` String

The textual node identifier of the node on which the service, who will execute this command, will operate.

`nodeNamespace` OpcNamespace

The namespace within that the node with the `nodeId` specified can be located.

# OpcNodeServiceCommand(UInt32)

Initializes a new instance of the OpcNodeServiceCommand class using the `nodeId` specified.

**C#**

```
protected OpcNodeServiceCommand(uint nodeId)
```

**Parameters**

`nodeId` UInt32

The numeric node identifier of the node on which the service, who will execute this command, will operate.

# OpcNodeServiceCommand(UInt32, Int32)

Initializes a new instance of the OpcNodeServiceCommand class using the `nodeId` and `namespaceIndex` specified.

**C#**

```
protected OpcNodeServiceCommand(uint nodeId, int namespaceIndex)
```

#### Parameters

`nodeId` UInt32

The numeric node identifier of the node on which the service, who will execute this command, will operate.

`namespaceIndex` Int32

The index of the namespace within that the node with the `nodeId` specified can be located.

## OpcNodeServiceCommand(UInt32, OpcNamespace)

Initializes a new instance of the OpcNodeServiceCommand class using the `nodeId` and `nodeNamespace` specified.

#### C#

```
protected OpcNodeServiceCommand(uint nodeId, OpcNamespace nodeNamespace)
```

#### Parameters

`nodeId` UInt32

The numeric node identifier of the node on which the service, who will execute this command, will operate.

`nodeNamespace` OpcNamespace

The namespace within that the node with the `nodeId` specified can be located.

# Properties

## NodeId

Gets the node identifier of the node on which a node orientated service have to operate on.

#### C#

```
public OpcNodeId NodeId { get; }
```

#### Property Value

OpcNodeId

An instance of the OpcNodeId class identifying the node on which a node orientated service have to operate on.

# SupportsNullNodeId

Gets a value indicating whether the OpcNodeServiceCommand supports instances of the OpcNodeId class its IsNull provides a value equals to the value true.

**C#**

```
protected virtual 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.

# Methods

## DenyNullIdentifier(OpcNodeId, String)

Verifies whether the `value` is a null identifier by checking the IsNull property.

**C#**

```
protected void DenyNullIdentifier(OpcNodeId value, string valueName)
```

**Parameters**

`value` OpcNodeId

The OpcNodeId to verify.

`valueName` String

The name of the `value` passed. This is used in case there `value` is a null identifier.

**Exceptions**

ArgumentException

The IsNull of `value` is equals true.

# Table of Contents

***Constructors*** ..... 1
OpcNodeServiceCommand(Byte[]) ..... 1
OpcNodeServiceCommand(Byte[], Int32) ..... 1
OpcNodeServiceCommand(Byte[], OpcNamespace) ..... 1
OpcNodeServiceCommand(Guid) ..... 2
OpcNodeServiceCommand(Guid, Int32) ..... 2
OpcNodeServiceCommand(Guid, OpcNamespace) ..... 2
OpcNodeServiceCommand(Int32) ..... 3
OpcNodeServiceCommand(Int32, Int32) ..... 3
OpcNodeServiceCommand(Int32, OpcNamespace) ..... 3
OpcNodeServiceCommand(OpcNodeId) ..... 4
OpcNodeServiceCommand(String) ..... 4
OpcNodeServiceCommand(String, Int32) ..... 4
OpcNodeServiceCommand(String, OpcNamespace) ..... 5
OpcNodeServiceCommand(UInt32) ..... 5
OpcNodeServiceCommand(UInt32, Int32) ..... 5
OpcNodeServiceCommand(UInt32, OpcNamespace) ..... 6
***Properties*** ..... 6
NodeId ..... 6
SupportsNullNodeId ..... 7
***Methods*** ..... 7
DenyNullIdentifier(OpcNodeId, String) ..... 7