

OpcAddAnalogItemNode<T> Class

Namespace: Opc.UaFx

Assemblies: Opc.UaFx.Advanced.dll

Defines a single command of the [IOpcAddNodesService](#) used to add one analog item node.

C#

```
public class OpcAddAnalogItemNode<T> : OpcAddAnalogItemNode
```

Inheritance [Object](#) > [OpcServiceCommand](#) > [OpcNodeServiceCommand](#) > [OpcAddNode](#) > [OpcAddInstanceNode](#) > [OpcAddVariableNode](#) > [OpcAddDataItemNode](#) > [OpcAddAnalogItemNode](#) > [OpcAddAnalogItemNode<T>](#)

Constructors

Name	Description
OpcAddAnalogItemNode`1 (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.
OpcAddAnalogItemNode`1 (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.
OpcAddAnalogItemNode`1 (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.
OpcAddAnalogItemNode`1 (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.
OpcAddAnalogItemNode`1 (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.
OpcAddAnalogItemNode`1 (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.

Name	Description
<code>OpcAddAnalogItemNode`1(OpcName, OpcNodeId, OpcNodeId, OpcReferenceType,)</code>	Initializes a new instance of the <code>OpcAddAnalogItemNode`1</code> class using the <code>name</code> of the analog item node to add, which shall be additionally accessible by the <code>nodeId</code> defined. The new node will be a child of the node identified by <code>parentNodeId</code> using the <code>referenceType</code> specified as the type of reference.
<code>OpcAddAnalogItemNode`1(OpcName, OpcNodeId, OpcNodeId,)</code>	Initializes a new instance of the <code>OpcAddAnalogItemNode`1</code> class using the <code>name</code> of the analog item node to add, which shall be additionally accessible by the <code>nodeId</code> defined. The new node will be a child of the node identified by <code>parentNodeId</code> using <code>HasComponent</code> as the type of reference.
<code>OpcAddAnalogItemNode`1(OpcName, OpcNodeId,)</code>	Initializes a new instance of the <code>OpcAddAnalogItemNode`1</code> class using the <code>name</code> of the analog item node to add, which shall be additionally accessible by the <code>nodeId</code> defined. The new node will be a child of the <code>ObjectsFolder</code> node using <code>HasComponent</code> as the type of reference.
<code>OpcAddAnalogItemNode`1(OpcName,)</code>	Initializes a new instance of the <code>OpcAddAnalogItemNode`1</code> class using the <code>name</code> of the analog item node to add. The according <code>OpcNodeId</code> to identify and access the new node is determined by the service. The new node will be a child of the <code>ObjectsFolder</code> node using <code>HasComponent</code> as the type of reference.

Properties

Name	Description
<code>AccessLevel</code>	Gets or sets a value which indicates in which ways the <code>Value</code> attribute of the variable node can be accessed (read/write) and if it provides current and/or historic data. (Inherited from <code>OpcAddVariableNode</code>)
<code>ArrayDimensions</code>	Gets the number/lengths of dimensions for an array <code>Value</code> with one or more fixed dimensions. (Inherited from <code>OpcAddVariableNode</code>)
<code>Category</code>	Gets a value indicating the classification of the node in the address space. (Inherited from <code>OpcAddNode</code>)
<code>Children</code>	Gets a collection of <code>OpcAddNode</code> instances which define the sub-ordinated nodes to add as children to the node to add. (Inherited from <code>OpcAddInstanceNode</code>)
<code>DataType</code>	Gets or sets a value which defines a pre-defined used <code>DataTypeId</code> as one of the members defined by the <code>OpcDataType</code> 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.
<code>DataType</code>	Gets or sets a value which defines a pre-defined used <code>DataTypeId</code> as one of the members defined by the <code>OpcDataType</code> 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. (Inherited from <code>OpcAddVariableNode</code>)
<code>DataTypeId</code>	Gets or sets the identifier which identifies the node that defines the type of data represented by the variable node. A null reference (Nothing in Visual Basic) indicates that the attribute is undefined and its default value is used. (Inherited from <code>OpcAddVariableNode</code>)

Name	Description
Definition	Gets or sets a vendor-specific, human readable string that specifies how the value of the data item is calculated. A null reference (Nothing in Visual Basic) indicates that the property is undefined and its default value is used. (Inherited from OpcAddDataItemNode)
Description	Gets or sets the localized description of the meaning of the node. (Inherited from OpcAddNode)
DisplayName	Gets or sets the localized name of the node. (Inherited from OpcAddNode)
EngineeringUnit	Gets or sets the engineering unit information which specifies the unit of the value attribute of the analog item node to add. A null reference (Nothing in Visual Basic) indicates that the property is undefined and its default value is used. (Inherited from OpcAddAnalogItemNode)
EngineeringUnitRange	Gets or sets the range information which applies to normal operation. A null reference (Nothing in Visual Basic) indicates that the property is undefined and its default value is used. (Inherited from OpcAddAnalogItemNode)
InstrumentRange	Gets or sets the range information which applies to the instrumental values returned by the instrument. A null reference (Nothing in Visual Basic) indicates that the property is undefined and its default value is used. (Inherited from OpcAddAnalogItemNode)
IsHistorizing	Gets or sets a value indicating whether the server is actively collecting data for the history of the variable. (Inherited from OpcAddVariableNode)
Name	Gets the non-localised human-readable name of the node in the address space. (Inherited from OpcAddNode)
NodeId	Gets the node identifier of the node on which a node orientated service have to operate on. (Inherited from OpcNodeServiceCommand)
ParentNodeId	Gets the identifier of the existing parent node of the new node. (Inherited from OpcAddNode)
ReferenceType	Gets a value which defines a pre-defined used ReferenceTypeId as one of the members defined by the OpcReferenceType enumeration to simplify querying standard reference types. (Inherited from OpcAddInstanceNode)
ReferenceTypeId	Gets the identifier which identifies the node that defines the semantic of the reference between a source and a target node and generally reflects an operation between the two, such as "A contains B". (Inherited from OpcAddInstanceNode)
SupportsNullNodeId	Gets a value indicating whether the OpcNodeServiceCommand supports instances of the OpcNodeId class its IsNull provides a value equals to the value true. (Inherited from OpcNodeServiceCommand)
SupportsNullNodeId	Gets a value indicating whether the OpcAddNode supports instances of the OpcNodeId class its IsNull provides a value equals to the value true. (Inherited from OpcAddNode)
Type	Gets value indicating the predefined underlying type definition the new node will represent an instance of. (Inherited from OpcAddVariableNode)
TypeDefinitionId	Gets the identifier which identifies the node that defines the underlying node type from that the instance node is to be created. (Inherited from OpcAddInstanceNode)
UserAccessLevel	Gets or sets a value which indicates in which ways the Value attribute of the variable node can be accessed (read/write) and if it provides current and/or historic data taking user access rights into account. (Inherited from OpcAddVariableNode)

Name	Description
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. (Inherited from OpcAddNode)
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.
Value	Gets or sets the value of the variable node which may be simple or complex. A null reference (Nothing in Visual Basic) indicates that the attribute is undefined and its default value is used. (Inherited from OpcAddVariableNode)
ValuePrecision	Gets or sets a value which specifies the maximum precision that the server can maintain for the item based on restrictions in the target environment. A null reference (Nothing in Visual Basic) indicates that the property is undefined and its default value is used. (Inherited from OpcAddDataItemNode)
ValueRank	Gets or sets a value which indicates whether the value attribute of the variable is an array and how many dimensions the array has. (Inherited from OpcAddVariableNode)
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. (Inherited from OpcAddNode)

Methods

Name	Description
DenyNullIdentifier(OpcNodeId, String)	Verifies whether the value is a null identifier by checking the IsNull property. (Inherited from OpcNodeServiceCommand)
OfType(OpcNodeId)	Retrieves an instance which represents the definition of a variable type that can be used to define OpcAddVariableNode command instances using the type of variable node represented by the TypeDefinition instance provided. (Inherited from OpcAddVariableNode)

Table of Contents

Constructors	1
Properties	2
Methods	4

