

OpcNodeSet Members

Namespace: Opc.UaFx

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

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

Properties

LastModified

Gets the last time the node set document was modified.

C#

```
public DateTime? LastModified { get; }
```

Property Value

[Nullable<DateTime>](#)

The last time the node set document was modified or a null reference (Nothing in Visual Basic) if there is no time of the last modification available.

Remarks

The value of this property is primarily determined by the contents of the node set document the current [OpcNodeSet](#) originates from. In case there the node set document does not provide the last modified information this property represents the [FileSystemInfo.LastWriteTime](#) in case there the node set has been created from a file; otherwise a null reference (Nothing in Visual Basic).

Models

C#

```
public OpcReadOnlyModelCollection Models { get; }
```

Property Value

[OpcReadOnlyModelCollection](#)

Name

Gets the name of the resource from which node set information originates from.

C#

```
public string Name { get; }
```

Property Value

String

The name of the node set resource its information is represented by the current [OpcNodeSet](#). In case there the [OpcNodeSet](#) has been created from a file the name of the file without extension; otherwise a null reference (Nothing in Visual Basic).

Namespaces

Gets a read-only collection of [OpcNamespaces](#) referred to.

C#

```
public OpcReadOnlyNamespaceCollection Namespaces { get; }
```

Property Value

OpcReadOnlyNamespaceCollection

An instance of the [OpcReadOnlyNamespaceCollection](#) class with the [OpcNamespace](#) information used by the current [OpcNodeSet](#).

Methods

GetDataType(OpcEncoding)

Retrieves the [OpcDataTypeInfo](#) object which declares the type which is identified by the [encoding](#) specified.

C#

```
public OpcDataTypeInfo GetDataType(OpcEncoding encoding)
```

Parameters

[encoding](#) [OpcEncoding](#)

The [OpcEncoding](#) which identifies the [OpcDataTypeInfo](#) to retrieve.

Returns

OpcDataTypeInfo

The [OpcDataTypeInfo](#) object which is known under the [encoding](#) specified or a null reference (Nothing in Visual Basic) if there doesn't exist a known [OpcDataTypeInfo](#) which is associated with the [encoding](#) specified.

Exceptions

ArgumentNullException

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

GetDataTypeInfo(OpcName)

Retrieves the `OpcDataTypeInfo` object which is known under the `name` specified.

C#

```
public OpcDataTypeInfo GetDataType(OpcName name)
```

Parameters

`name` `OpcName`

The `OpcName` of the `OpcDataTypeInfo` object to retrieve.

Returns

`OpcDataTypeInfo`

The `OpcDataTypeInfo` object its `Name` is equals to the `name` specified; otherwise a null reference (Nothing in Visual Basic).

Exceptions

ArgumentNullException

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

GetDataTypeInfo(OpcNodeId)

Retrieves the `OpcDataTypeInfo` object which declares the type which is identified by the `typeId` specified.

C#

```
public OpcDataTypeInfo GetDataType(OpcNodeId typeId)
```

Parameters

`typeId` `OpcNodeId`

The `OpcNodeId` which identifies the `OpcDataTypeInfo` to retrieve.

Returns

`OpcDataTypeInfo`

The [OpcDataTypeInfo](#) object which is known under the [typeId](#) specified or a null reference (Nothing in Visual Basic) if there doesn't exist a known [OpcDataTypeInfo](#) object which is associated with the [typeId](#) specified.

Exceptions

[ArgumentNullException](#)

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

GetDataType(String)

Retrieves the [OpcDataTypeInfo](#) object which is known under the [name](#) specified.

C#

```
public OpcDataTypeInfo GetDataType(string name)
```

Parameters

[name](#) [String](#)

The [String](#) to use to identify the [OpcDataTypeInfo](#) object to retrieve.

Returns

[OpcDataTypeInfo](#)

The [OpcDataTypeInfo](#) object its [Name](#) is equals (regarding its [Value](#)) to the [name](#) specified; otherwise a null reference (Nothing in Visual Basic).

Exceptions

[ArgumentException](#)

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

[ArgumentNullException](#)

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

GetDataType(Type)

Retrieves the [OpcDataTypeInfo](#) object which declares the type implemented by the [underlyingType](#) specified.

C#

```
public OpcDataTypeInfo GetDataType(Type underlyingType)
```

Parameters

underlyingType Type

The **Type** which implements the **OpcDataTypeInfo** to retrieve.

Returns

OpcDataTypeInfo

The **OpcDataTypeInfo** object which declares the **underlyingType** specified or a null reference (Nothing in Visual Basic) if there isn't a **OpcDataTypeInfo** object associated with the **underlyingType** specified.

Exceptions

ArgumentNullException

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

GetDataType(XmlQualifiedName)

Retrieves the **OpcDataTypeInfo** object which is known under the **xmlName** specified.

C#

```
public OpcDataTypeInfo GetDataType(XmlQualifiedName xmlName)
```

Parameters

xmlName XmlQualifiedName

The **XmlQualifiedName** of the **OpcDataTypeInfo** object to retrieve.

Returns

OpcDataTypeInfo

The **OpcDataTypeInfo** object its **XmlName** is equals to the **xmlName** specified; otherwise a null reference (Nothing in Visual Basic).

Exceptions

ArgumentNullException

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

GetDataTypes()

Retrieves all **OpcDataTypeInfo** objects offered by the **IOpcTypeProvider**.

C#

```
public OpcDataTypeInfo[] GetDataTypes()
```

Returns

[OpcDataTypeInfo\[\]](#)

An array that contains all [OpcDataTypeInfo](#) objects that are offered by the [IOpcTypeProvider](#).

GetDataTypeSystem()

Retrieves the system used to describe the data types defined in the [OpcNodeSet](#) using [Binary](#) as the type of encoding.

C#

```
public OpcDataTypeSystem GetDataTypeSystem()
```

Returns

[OpcDataTypeSystem](#)

An instance of the [OpcDataTypeSystem](#) class which can be used to understand the structure of the data described for the [Binary](#) encoding.

GetDataTypeSystem(OpcEncodingType)

Retrieves the system used to describe the data types defined in the [OpcNodeSet](#) using [OpcEncodingType](#) specified by [encodingType](#) as the type of encoding.

C#

```
public OpcDataTypeSystem GetDataTypeSystem(OpcEncodingType encodingType)
```

Parameters

[encodingType](#) [OpcEncodingType](#)

Returns

[OpcDataTypeSystem](#)

An instance of the [OpcDataTypeSystem](#) class which can be used to understand the structure of the data described for the encoding specified.

GetNodeType(OpcName)

Retrieves the [OpcNodeTypeInfo](#) object which is known under the [name](#) specified.

C#

```
public OpcNodeTypeInfo GetNodeType(OpcName name)
```

Parameters

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

The [OpcName](#) of the [OpcNodeTypeInfo](#) object to retrieve.

Returns

[OpcNodeTypeInfo](#)

The [OpcNodeTypeInfo](#) object its [Name](#) is equals to the [name](#) specified; otherwise a null reference (Nothing in Visual Basic).

Exceptions

[ArgumentNullException](#)

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

GetNodeType(OpcNodeId)

Retrieves the [OpcNodeTypeInfo](#) object which declares the type which is identified by the [typeId](#) specified.

C#

```
public OpcNodeTypeInfo GetNodeType(OpcNodeId typeId)
```

Parameters

[typeId](#) [OpcNodeId](#)

The [OpcNodeId](#) which identifies the [OpcNodeTypeInfo](#) to retrieve.

Returns

[OpcNodeTypeInfo](#)

The [OpcNodeTypeInfo](#) object which is known under the [typeId](#) specified or a null reference (Nothing in Visual Basic) if there doesn't exist a known [OpcNodeTypeInfo](#) object which is associated with the [typeId](#) specified.

Exceptions

ArgumentNullException

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

GetNodeType(String)

Retrieves the `OpcNodeTypeInfo` object which is known under the `name` specified.

C#

```
public OpcNodeTypeInfo GetNodeType(string name)
```

Parameters

`name` String

The String to use to identify the `OpcNodeTypeInfo` object to retrieve.

Returns

`OpcNodeTypeInfo`

The `OpcNodeTypeInfo` object its `Name` is equals (regarding its `Value`) to the `name` specified; otherwise a null reference (Nothing in Visual Basic).

Exceptions

`ArgumentException`

The `name` is equals `Empty`.

`ArgumentNullException`

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

GetNodeType(Type)

Retrieves the `OpcNodeTypeInfo` object which declares the type implemented by the `underlyingType` specified.

C#

```
public OpcNodeTypeInfo GetNodeType(Type underlyingType)
```

Parameters

`underlyingType` Type

The Type which implements the `OpcNodeTypeInfo` to retrieve.

Returns

OpcNodeTypeInfo

The [OpcNodeTypeInfo](#) object which declares the [underlyingType](#) specified or a null reference (Nothing in Visual Basic) if there isn't a [OpcNodeTypeInfo](#) object associated with the [underlyingType](#) specified.

Exceptions

[ArgumentNullException](#)

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

GetNodeTypes()

Retrieves all [OpcNodeTypeInfo](#) objects offered by the [IOpcTypeProvider](#).

C#

```
public OpcNodeTypeInfo[] GetNodeTypes()
```

Returns

[OpcNodeTypeInfo\[\]](#)

An array that contains all [OpcNodeTypeInfo](#) objects that are offered by the [IOpcTypeProvider](#).

GetNodeTypeSystem()

Retrieves the system used to describe the node types defined in the [OpcNodeSet](#).

C#

```
public OpcNodeTypeSystem GetNodeTypeSystem()
```

Returns

[OpcNodeTypeSystem](#)

An instance of the [OpcNodeTypeSystem](#) class which can be used to understand the structure of the nodes defined.

Load(Stream)

Creates a new [OpcNodeSet](#) instance from the contents in the [stream](#) specified.

C#

```
public static OpcNodeSet Load(Stream stream)
```

Parameters

stream Stream

The [Stream](#) which contains the data from which the [OpcNodeSet](#) is to be created.

Returns

[OpcNodeSet](#)

A new instance of the [OpcNodeSet](#) class that represents the contents of the [stream](#).

Exceptions

[ArgumentNullException](#)

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

[InvalidDataException](#)

The node set specified does not provide any namespace information.

Load(String)

Creates a new [OpcNodeSet](#) instance from the file specified by [filePath](#).

C#

```
public static OpcNodeSet Load(string filePath)
```

Parameters

[filePath](#) String

The path to the file from which the [OpcNodeSet](#) is to be created.

Returns

[OpcNodeSet](#)

A new instance of the [OpcNodeSet](#) class that represents the contents of the file.

Exceptions

[ArgumentException](#)

The [filePath](#) is an empty string.

[ArgumentNullException](#)

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

[InvalidDataException](#)

The node set specified does not provide any namespace information.

Parse(String)

Creates a new [OpcNodeSet](#) instance from the [String](#) specified by [data](#).

C#

```
public static OpcNodeSet Parse(string data)
```

Parameters

[data](#) [String](#)

The [String](#) that contains the node set data.

Returns

[OpcNodeSet](#)

Exceptions

[ArgumentException](#)

The [data](#) is an empty string.

[ArgumentNullException](#)

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

[InvalidDataException](#)

The node set specified does not provide any namespace information.

Save(Stream)

C#

```
public void Save(Stream stream)
```

Parameters

[stream](#) [Stream](#)

The [Stream](#) this [OpcNodeSet](#) is to be written to.

Exceptions

[ArgumentNullException](#)

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

Save(String)

C#

```
public void Save(string filePath)
```

Parameters

filePath String

The path to the file this [OpcNodeSet](#) is to be written to.

Exceptions

[ArgumentException](#)

The **filePath** is an empty string.

[ArgumentNullException](#)

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

TryParse(String, out OpcNodeSet)

Determines whether a [String](#) is a valid [OpcNodeSet](#).

C#

```
public static bool TryParse(string data, out OpcNodeSet nodeSet)
```

Parameters

data String

The [String](#) to validate.

nodeSet [OpcNodeSet](#)

The [OpcNodeSet](#) version of the [String](#).

Returns

[Boolean](#)

The value true if the specified **data** is a valid [OpcNodeSet](#); otherwise the value false.

Table of Contents

Properties	1
LastModified	1
Models	1
Name	1
Namespaces	2
Methods	2
GetDataType(OpcEncoding)	2
GetDataType(OpcName)	3
GetDataType(OpcNodeId)	3
GetDataType(String)	4
GetDataType(Type)	4
GetDataType(XmlQualifiedName)	5
GetDataTypes()	5
GetDataTypeSystem()	6
GetDataTypeSystem(OpcEncodingType)	6
GetNodeType(OpcName)	7
GetNodeType(OpcNodeId)	7
GetNodeType(String)	8
GetNodeType(Type)	8
GetNodeTypes()	9
GetNodeTypeSystem()	9
Load(Stream)	9
Load(String)	10
Parse(String)	11
Save(Stream)	11
Save(String)	12
TryParse(String, out OpcNodeSet)	12

