

# OpcNodeValue<T> Members

**Namespace:** Opc.UaFx

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

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

## Constructors

### OpcNodeValue(OpcNodeId)

Initializes a new instance of the [OpcNodeValue`1](#) class with the [nodeId](#) of the node the [Value](#) attribute represented belongs to.

**C#**

```
public OpcNodeValue(OpcNodeId nodeId)
```

#### Parameters

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

The [OpcNodeId](#) of the node the value of its [Value](#) attribute is to be represented.

#### Exceptions

[ArgumentNullException](#)

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

### OpcNodeValue(OpcNodeId, T)

Initializes a new instance of the [OpcNodeValue`1](#) class with the [nodeId](#) of the node the [Value](#) attribute and its [value](#) specified belongs to.

**C#**

```
public OpcNodeValue(OpcNodeId nodeId, T value)
```

#### Parameters

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

The [OpcNodeId](#) of the node its [Value](#) attribute value is to be represented.

[value](#) [T](#)

The value to represent.

#### Exceptions

## ArgumentNullException

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

# OpcNodeValue(OpcNodeId, T, DateTime)

Initializes a new instance of the `OpcNodeValue`1` class with the `nodeId` of the node the `Value` attribute and its `value` and the `sourceTimestamp` of the `value` specified belongs to.

## C#

```
public OpcNodeValue(OpcNodeId nodeId, T value, DateTime sourceTimestamp)
```

## Parameters

`nodeId` `OpcNodeId`

The `OpcNodeId` of the node its `Value` attribute value is to be represented.

`value` `T`

The value to represent.

`sourceTimestamp` `DateTime`

The timestamp at which the source produced the `value`.

## Exceptions

[ArgumentNullException](#)

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

# OpcNodeValue(OpcNodeId, T, DateTime, DateTime)

Initializes a new instance of the `OpcNodeValue`1` class with the `nodeId` of the node the `Value` attribute and its `value`, `sourceTimestamp` and `serverTimestamp` of the `value` specified belongs to.

## C#

```
public OpcNodeValue(OpcNodeId nodeId, T value, DateTime sourceTimestamp, DateTime serverTimestamp)
```

## Parameters

`nodeId` `OpcNodeId`

The `OpcNodeId` of the node its `Value` attribute value is to be represented.

`value` `T`

The value to represent.

`sourceTimestamp` `DateTime`

The timestamp at which the source produced the `value`.

`serverTimestamp` [DateTime](#)

The timestamp at which the server recorded the `value`.

## Exceptions

[ArgumentNullException](#)

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

# OpcNodeValue(OpcNodeId, T, DateTime, DateTime, OpcStatus)

Initializes a new instance of the `OpcNodeValue`1` class with the `nodeId` of the node the `Value` attribute and its `value`, `sourceTimestamp`, `serverTimestamp` and `status` of the `value` specified belongs to.

## C#

```
public OpcNodeValue(OpcNodeId nodeId, T value, DateTime sourceTimestamp, DateTime serverTimestamp, OpcStatus status)
```

## Parameters

`nodeId` [OpcNodeId](#)

The [OpcNodeId](#) of the node its `Value` attribute value is to be represented.

`value` [T](#)

The value to represent.

`sourceTimestamp` [DateTime](#)

The timestamp at which the source produced the `value`.

`serverTimestamp` [DateTime](#)

The timestamp at which the server recorded the `value`.

`status` [OpcStatus](#)

The status information associated with the `value`.

## Exceptions

[ArgumentNullException](#)

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

# OpcNodeValue(OpcNodeId, T, DateTime, DateTime, OpcStatus)

## OpcStatusCode)

Initializes a new instance of the `OpcNodeValue<T>` class with the `nodeId` of the node the `Value` attribute and its `value`, `sourceTimestamp`, `serverTimestamp` and `statusCode` of the `value` specified belongs to.

### C#

```
[CLSCompliant(false)]  
public OpcNodeValue(OpcNodeId nodeId, T value, DateTime sourceTimestamp, DateTime  
serverTimestamp, OpcStatusCode statusCode)
```

### Parameters

`nodeId` [OpcNodeId](#)

The [OpcNodeId](#) of the node its `Value` attribute value is to be represented.

`value` [T](#)

The value to represent.

`sourceTimestamp` [DateTime](#)

The timestamp at which the source produced the `value`.

`serverTimestamp` [DateTime](#)

The timestamp at which the server recorded the `value`.

`statusCode` [OpcStatusCode](#)

One of the [OpcStatusCode](#) members which defines the status information associated with the `value`.

### Exceptions

[ArgumentNullException](#)

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

## OpcNodeValue(OpcNodeId, T, DateTime, OpcStatus)

Initializes a new instance of the `OpcNodeValue<T>` class with the `nodeId` of the node the `Value` attribute and its `value`, `sourceTimestamp` and `status` of the `value` specified belongs to.

### C#

```
public OpcNodeValue(OpcNodeId nodeId, T value, DateTime sourceTimestamp, OpcStatus status)
```

### Parameters

`nodeId` [OpcNodeId](#)

The [OpcNodeId](#) of the node its `Value` attribute value is to be represented.

`value` [T](#)

The value to represent.

`sourceTimestamp` [DateTime](#)

The timestamp at which the source produced the value.

`status` [OpcStatus](#)

The status information associated with the `value`.

## Exceptions

[ArgumentNullException](#)

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

# OpcNodeValue(OpcNodeId, T, DateTime, OpcStatusCode)

Initializes a new instance of the [OpcNodeValue`1](#) class with the `nodeId` of the node the `Value` attribute and its `value`, `sourceTimestamp` and `statusCode` of the `value` specified belongs to.

## C#

```
[CLSCompliant(false)]  
public OpcNodeValue(OpcNodeId nodeId, T value, DateTime sourceTimestamp, OpcStatusCode  
statusCode)
```

## Parameters

`nodeId` [OpcNodeId](#)

The [OpcNodeId](#) of the node its `Value` attribute value is to be represented.

`value` [T](#)

The value to represent.

`sourceTimestamp` [DateTime](#)

The timestamp at which the source produced the value.

`statusCode` [OpcStatusCode](#)

One of the [OpcStatusCode](#) members which defines the status information associated with the `value`.

## Exceptions

[ArgumentNullException](#)

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

# Methods

# CompareTo(Object)

Compares the current [OpcNodeValue`1](#) with the [other](#).

**C#**

```
public override int CompareTo(object other)
```

## Parameters

[other](#) Object

The [OpcNodeValue`1](#) to compare with this [OpcNodeValue`1](#).

## Returns

[Int32](#)

A 32-bit signed integer that indicates the relative order of the objects being compared ([CompareTo\(Object\)](#)).

# CompareTo(OpcNodeAttribute<T>)

Compares the current [OpcNodeValue`1](#) with the [other](#).

**C#**

```
public override int CompareTo(OpcNodeAttribute<T> other)
```

## Parameters

[other](#) [OpcNodeAttribute<T>](#)

The [OpcNodeValue`1](#) to compare with this [OpcNodeValue`1](#).

## Returns

[Int32](#)

A 32-bit signed integer that indicates the relative order of the objects being compared ([CompareTo\(Object\)](#)).

# CompareTo(OpcNodeValue<T>)

Compares the current [OpcNodeValue`1](#) with another [OpcNodeValue`1](#).

**C#**

```
public virtual int CompareTo(OpcNodeValue<T> other)
```

## Parameters

other `OpcNodeValue`1`

The `OpcNodeValue`1` to compare with this `OpcNodeValue`1`.

### Returns

`Int32`

A 32-bit signed integer that indicates the relative order of the objects being compared (`CompareTo()`).

## CompareTo(OpcValue)

Compares the current `OpcNodeValue`1` with the `other`.

### C#

```
public override int CompareTo(OpcValue other)
```

### Parameters

`other` `OpcValue`

The `OpcNodeValue`1` to compare with this `OpcNodeValue`1`.

### Returns

`Int32`

A 32-bit signed integer that indicates the relative order of the objects being compared (`CompareTo(Object)`).

## CompareTo(OpcValue<T>)

Compares the current `OpcNodeValue`1` with the `other`.

### C#

```
public override int CompareTo(OpcValue<T> other)
```

### Parameters

`other` `OpcValue<T>`

The `OpcNodeValue`1` to compare with this `OpcNodeValue`1`.

### Returns

`Int32`

A 32-bit signed integer that indicates the relative order of the objects being compared (`CompareTo(Object)`).

## Equals(Object)

Determines whether the specified `other` is equal to this `OpcNodeValue`1`.

**C#**

```
public override bool Equals(object other)
```

### Parameters

`other` Object

The `OpcNodeValue`1` to compare to the current `OpcNodeValue`1`.

### Returns

Boolean

The value true if the specified `OpcNodeValue`1` is equal to the current `OpcNodeValue`1`; otherwise the value false.

## Equals(OpcNodeAttribute<T>)

Determines whether the specified `other` is equal to this `OpcNodeValue`1`.

**C#**

```
public override bool Equals(OpcNodeAttribute<T> other)
```

### Parameters

`other` `OpcNodeAttribute<T>`

The `OpcNodeValue`1` to compare to the current `OpcNodeValue`1`.

### Returns

Boolean

The value true if the specified `OpcNodeValue`1` is equal to the current `OpcNodeValue`1`; otherwise the value false.

## Equals(OpcNodeValue<T>)

Determines whether the specified `other` is equal to this `OpcNodeValue`1`.

**C#**

```
public virtual bool Equals(OpcNodeValue<T> other)
```

### Parameters



other `OpcNodeValue`1`

The `OpcNodeValue`1` to compare to the current `OpcNodeValue`1`.

### Returns

Boolean

The value true if the specified `OpcNodeValue`1` is equal to the current `OpcNodeValue`1`; otherwise the value false.

## Equals(OpcValue)

Determines whether the specified `other` is equal to this `OpcNodeValue`1`.

### C#

```
public override bool Equals(OpcValue other)
```

### Parameters

other `OpcValue`

The `OpcNodeValue`1` to compare to the current `OpcNodeValue`1`.

### Returns

Boolean

The value true if the specified `OpcNodeValue`1` is equal to the current `OpcNodeValue`1`; otherwise the value false.

## Equals(OpcValue<T>)

Determines whether the specified `other` is equal to this `OpcNodeValue`1`.

### C#

```
public override bool Equals(OpcValue<T> other)
```

### Parameters

other `OpcValue<T>`

The `OpcNodeValue`1` to compare to the current `OpcNodeValue`1`.

### Returns

Boolean

The value true if the specified `OpcNodeValue`1` is equal to the current `OpcNodeValue`1`; otherwise the value false.

## GetHashCode()

Retrieves a hash code for this [OpcNodeValue`1](#).

**C#**

```
public override int GetHashCode()
```

### Returns

[Int32](#)

An [Int32](#) that contains the hash code for the [OpcNodeValue`1](#).

## Operators

### Equality(OpcNodeValue<T>, OpcNodeValue<T>)

Returns a value indicating whether two instance of [OpcNodeValue`1](#) are equal.

**C#**

```
public static bool operator ==(OpcNodeValue<T> left, OpcNodeValue<T> right)
```

### Explicit(OpcNodeValue<T> to DataValue)

Converts a [OpcNodeValue`1](#) to an [DataValue](#) object.

**C#**

```
[CLSCompliant(false)]  
public static explicit operator DataValue(OpcNodeValue<T> value)
```

### GreaterThan(OpcNodeValue<T>, OpcNodeValue<T>)

Determines whether the first specified [OpcNodeValue`1](#) object is greater than the second specified [OpcNodeValue`1](#) object.

**C#**

```
public static bool operator >(OpcNodeValue<T> left, OpcNodeValue<T> right)
```

### GreaterThanOrEqual(OpcNodeValue<T>, OpcNodeValue<T>)

Determines whether the first specified [OpcNodeValue`1](#) object is greater than or equal to the second specified [OpcNodeValue`1](#) object.

**C#**

```
public static bool operator >=(OpcNodeValue<T> left, OpcNodeValue<T> right)
```

## Inequality(OpcNodeValue<T>, OpcNodeValue<T>)

Returns a value indicating whether two instances of [OpcNodeValue`1](#) are not equal.

**C#**

```
public static bool operator !=(OpcNodeValue<T> left, OpcNodeValue<T> right)
```

## LessThan(OpcNodeValue<T>, OpcNodeValue<T>)

Determines whether the first specified [OpcNodeValue`1](#) object is less than the second specified [OpcNodeValue`1](#) object.

**C#**

```
public static bool operator <(OpcNodeValue<T> left, OpcNodeValue<T> right)
```

### Exceptions

[ArgumentNullException](#)

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

## LessThanOrEqual(OpcNodeValue<T>, OpcNodeValue<T>)

Determines whether the first specified [OpcNodeValue`1](#) object is less than or equal to the second [OpcNodeValue`1](#) object.

**C#**

```
public static bool operator <=(OpcNodeValue<T> left, OpcNodeValue<T> right)
```

### Exceptions

[ArgumentNullException](#)

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



# Table of Contents

- Constructors** ..... 1
  - OpcNodeValue(OpcNodeId) ..... 1
  - OpcNodeValue(OpcNodeId, T) ..... 1
  - OpcNodeValue(OpcNodeId, T, DateTime) ..... 2
  - OpcNodeValue(OpcNodeId, T, DateTime, DateTime) ..... 2
  - OpcNodeValue(OpcNodeId, T, DateTime, DateTime, OpcStatus) ..... 3
  - OpcNodeValue(OpcNodeId, T, DateTime, DateTime, OpcStatusCode) ..... 3
  - OpcNodeValue(OpcNodeId, T, DateTime, OpcStatus) ..... 4
  - OpcNodeValue(OpcNodeId, T, DateTime, OpcStatusCode) ..... 5
- Methods** ..... 5
  - CompareTo(Object) ..... 6
  - CompareTo(OpcNodeAttribute<T>) ..... 6
  - CompareTo(OpcNodeValue<T>) ..... 6
  - CompareTo(OpcValue) ..... 7
  - CompareTo(OpcValue<T>) ..... 7
  - Equals(Object) ..... 8
  - Equals(OpcNodeAttribute<T>) ..... 8
  - Equals(OpcNodeValue<T>) ..... 8
  - Equals(OpcValue) ..... 9
  - Equals(OpcValue<T>) ..... 9
  - GetHashCode() ..... 10
- Operators** ..... 10
  - Equality(OpcNodeValue<T>, OpcNodeValue<T>) ..... 10
  - Explicit(OpcNodeValue<T> to DataValue) ..... 10
  - GreaterThan(OpcNodeValue<T>, OpcNodeValue<T>) ..... 10
  - GreaterThanOrEqual(OpcNodeValue<T>, OpcNodeValue<T>) ..... 10
  - Inequality(OpcNodeValue<T>, OpcNodeValue<T>) ..... 11
  - LessThan(OpcNodeValue<T>, OpcNodeValue<T>) ..... 11
  - LessThanOrEqual(OpcNodeValue<T>, OpcNodeValue<T>) ..... 11