

OpcAcknowledgeableConditionNode Members

Namespace: Opc.UaFx

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

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

Constructors

OpcAcknowledgeableConditionNode(IOpcNode, OpcName)

Initializes a new instance of the [OpcAcknowledgeableConditionNode](#) class accessible by the **name** specified as a child node of the **parent** node given.

C#

```
public OpcAcknowledgeableConditionNode(IOpcNode parent, OpcName name)
```

Parameters

parent [IOpcNode](#)

The [IOpcNode](#) used as the parent node or a null reference (Nothing in Visual Basic) in the case there is no parent node available.

name [OpcName](#)

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

OpcAcknowledgeableConditionNode(IOpcNode, OpcName, OpcNodeId)

Initializes a new instance of the [OpcAcknowledgeableConditionNode](#) class accessible by the **name** and **id** specified as a child node of the **parent** node given.

C#

```
public OpcAcknowledgeableConditionNode(IOpcNode parent, OpcName name, OpcNodeId id)
```

Parameters

parent [IOpcNode](#)

The [IOpcNode](#) used as the parent node or a null reference (Nothing in Visual Basic) in the case there is no parent node available.

name [OpcName](#)

The `OpcName` through that the new acknowledgeable condition node can be accessed.

`id OpcNodeId`

The `OpcNodeId` through that the new acknowledgeable condition node can be identified and accessed.

OpcAcknowledgeableConditionNode(OpcName)

Initializes a new instance of the `OpcAcknowledgeableConditionNode` class accessible by the `name` specified.

C#

```
public OpcAcknowledgeableConditionNode(OpcName name)
```

Parameters

`name OpcName`

The `OpcName` through that the new acknowledgeable condition node can be accessed.

OpcAcknowledgeableConditionNode(OpcName, OpcNodeId)

Initializes a new instance of the `OpcAcknowledgeableConditionNode` class accessible by the `name` and `id` specified.

C#

```
public OpcAcknowledgeableConditionNode(OpcName name, OpcNodeId id)
```

Parameters

`name OpcName`

The `OpcName` through that the new acknowledgeable condition node can be accessed.

`id OpcNodeId`

The `OpcNodeId` through that the new acknowledgeable condition node can be identified and accessed.

Properties

AcknowledgeCallback

Gets or sets a callback used to acknowledge the state of the condition node.

C#

```
public OpcAddCommentCallback AcknowledgeCallback { get; set; }
```

Property Value

OpcAddCommentCallback

A [OpcAddCommentCallback](#) used to acknowledge the state of the condition node. The value can also be a null reference (Nothing in Visual Basic).

AcknowledgeNode

Gets the [OpcAddCommentMethodNode](#) used to handle 'Acknowledge' method calls to acknowledge the state of the condition node.

C#

```
public OpcAddCommentMethodNode AcknowledgeNode { get; }
```

Property Value

OpcAddCommentMethodNode

An instance of the [OpcAddCommentMethodNode](#) class. Which uses an [OpcAcknowledgeableConditionNode](#) defined callback to acknowledge the state of the condition node.

ConfirmCallback

Gets or sets a callback used to confirm the state of the condition node.

C#

```
public OpcAddCommentCallback ConfirmCallback { get; set; }
```

Property Value

OpcAddCommentCallback

A [OpcAddCommentCallback](#) used to confirm the state of the condition node. The value can also be a null reference (Nothing in Visual Basic).

ConfirmNode

Gets the [OpcAddCommentMethodNode](#) used to handle 'Confirm' method calls to confirm the state of the condition node.

C#

```
public OpcAddCommentMethodNode ConfirmNode { get; }
```

Property Value

OpcAddCommentMethodNode

An instance of the [OpcAddCommentMethodNode](#) class. Which uses an [OpcAcknowledgeableConditionNode](#)

defined callback to confirm the state of the condition node.

DefaultTypeId

Gets the default identifier which identifies the node that defines the underlying node type from that this [OpcInstanceNode](#) has been created.

C#

```
protected override OpcNodeId DefaultTypeId { get; }
```

Property Value

[OpcNodeId](#)

The [OpcNodeId](#) of the type node from that this [OpcInstanceNode](#) has been created from. These type node defines the typical structure of an instance node of its type definition. If there exists no specific type definition node a null reference (Nothing in Visual Basic).

IsAked

Gets a value indicating whether the condition requires acknowledgement for the reported condition state.

C#

```
public bool IsAked { get; }
```

Property Value

[Boolean](#)

The value false if an acknowledgement is required; otherwise the value true.

IsAkedNode

Gets the [OpcTwoStateVariableNode](#) of the [IsAked](#) property.

C#

```
public OpcTwoStateVariableNode IsAkedNode { get; }
```

Property Value

[OpcTwoStateVariableNode](#)

An instance of the [OpcTwoStateVariableNode](#) class.

IsConfirmed

Gets a value indicating whether the condition requires confirmation for the reported condition state.

C#

```
public bool IsConfirmed { get; }
```

Property Value

Boolean

The value false if an confirmation is required; otherwise the value true.

IsConfirmedNode

Gets the [OpcTwoStateVariableNode](#) of the [IsConfirmed](#) property.

C#

```
public OpcTwoStateVariableNode IsConfirmedNode { get; }
```

Property Value

[OpcTwoStateVariableNode](#)

An instance of the [OpcTwoStateVariableNode](#) class.

Methods

Acknowledge(OpcContext)

Acknowledges the state reported by an event notification which can be identified by the [EventId](#) using the specified [context](#).

C#

```
public void Acknowledge(OpcContext context)
```

Parameters

[context](#) [OpcContext](#)

The [OpcContext](#) to use.

Exceptions

[ArgumentNullException](#)

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

[OpcException](#)

The call failed (see exception details for more information).

Acknowledge(OpcContext, Byte[], OpcText)

Acknowledges the state reported by an event notification which can be identified by the `eventId` using the specified `context`.

C#

```
public void Acknowledge(OpcContext context, byte[] eventId, OpcText comment)
```

Parameters

`context` [OpcContext](#)

The [OpcContext](#) to use.

`eventId` [Byte\[\]](#)

The identifier which identifies the particular event notification its reported state for the condition node is to be commented.

`comment` [OpcText](#)

The text to apply on the condition state.

Exceptions

[ArgumentNullException](#)

The `context` or `eventId` is a null reference (Nothing in Visual Basic).

[OpcException](#)

The call failed (see exception details for more information).

Acknowledge(OpcContext, OpcText)

Acknowledges the state reported by an event notification which can be identified by the `EventId` using the specified `context`.

C#

```
public void Acknowledge(OpcContext context, OpcText comment)
```

Parameters

`context` [OpcContext](#)

The [OpcContext](#) to use.

`comment` [OpcText](#)

The text to apply on the condition state.

Exceptions

ArgumentNullException

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

OpcException

The call failed (see exception details for more information).

AcknowledgeCore(OpcNodeContext<OpcConditionNode>, Byte[], OpcText)

Acknowledges the state of the condition node.

C#

```
protected virtual OpcStatusCode AcknowledgeCore(OpcNodeContext<OpcConditionNode> context,
byte[] eventId, OpcText comment)
```

Parameters

`context` [OpcNodeContext<OpcConditionNode>](#)

The [OpcNodeContext](#) to use to acknowledge the state.

`eventId` [Byte\[\]](#)

The identifier identifying a particular event notification where a state was reported for a condition.

`comment` [OpcText](#)

A localized text to be applied to the condition.

Returns

[OpcStatusCode](#)

The [OpcStatusCode](#) specifying the outcome of the operation using the [AcknowledgeCallback](#) or [Good](#) if there is no custom callback routine defined.

Confirm(OpcContext)

Confirms the state reported by an event notification which can be identified by the [EventId](#) using the specified `context`.

C#

```
public void Confirm(OpcContext context)
```

Parameters

`context` [OpcContext](#)

The [OpcContext](#) to use.

Exceptions

[ArgumentNullException](#)

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

[OpcException](#)

The call failed (see exception details for more information).

Confirm(OpcContext, Byte[], OpcText)

Confirms the state reported by an event notification which can be identified by the `eventId` using the specified `context`.

C#

```
public void Confirm(OpcContext context, byte[] eventId, OpcText comment)
```

Parameters

`context` [OpcContext](#)

The [OpcContext](#) to use.

`eventId` [Byte\[\]](#)

The identifier which identifies the particular event notification its reported state for the condition node is to be commented.

`comment` [OpcText](#)

The text to apply on the condition state.

Exceptions

[ArgumentNullException](#)

The `context` or `eventId` is a null reference (Nothing in Visual Basic).

[OpcException](#)

The call failed (see exception details for more information).

Confirm(OpcContext, OpcText)

Confirms the state reported by an event notification which can be identified by the `EventId` using the specified `context`.

C#


```
public void Confirm(OpcContext context, OpcText comment)
```

Parameters

context [OpcContext](#)

The [OpcContext](#) to use.

comment [OpcText](#)

The text to apply on the condition state.

Exceptions

[ArgumentNullException](#)

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

[OpcException](#)

The call failed (see exception details for more information).

ConfirmCore(OpcNodeContext<OpcConditionNode>, Byte[], OpcText)

Confirms the state of the condition node.

C#

```
protected virtual OpcStatusCode ConfirmCore(OpcNodeContext<OpcConditionNode> context, byte[]  
eventId, OpcText comment)
```

Parameters

context [OpcNodeContext<OpcConditionNode>](#)

The [OpcNodeContext](#) to use to confirm the state.

eventId [Byte\[\]](#)

The identifier identifying a particular event notification where a state was reported for a condition.

comment [OpcText](#)

A localized text to be applied to the condition.

Returns

[OpcStatusCode](#)

The [OpcStatusCode](#) specifying the outcome of the operation using the [ConfirmCallback](#) or [Good](#) if there is no custom callback routine defined.

CreateBranchCore()

Creates a new instance of the [OpcAcknowledgeableConditionNode](#) using the same [Id](#) and [Name](#) as this node.

C#

```
protected override OpcConditionNode CreateBranchCore()
```

Returns

[OpcConditionNode](#)

A new instance of the [OpcAcknowledgeableConditionNode](#) identifiable and accessible through the same [Id](#) and [Name](#) as this node.

Table of Contents

Constructors	1
OpcAcknowledgeableConditionNode(IOpcNode, OpcName)	1
OpcAcknowledgeableConditionNode(IOpcNode, OpcName, OpcNodeId)	1
OpcAcknowledgeableConditionNode(OpcName)	2
OpcAcknowledgeableConditionNode(OpcName, OpcNodeId)	2
Properties	2
AcknowledgeCallback	2
AcknowledgeNode	3
ConfirmCallback	3
ConfirmNode	3
DefaultTypeDefinitionId	4
IsAcked	4
IsAckedNode	4
IsConfirmed	4
IsConfirmedNode	5
Methods	5
Acknowledge(OpcContext)	5
Acknowledge(OpcContext, Byte[], OpcText)	6
Acknowledge(OpcContext, OpcText)	6
AcknowledgeCore(OpcNodeContext<OpcConditionNode>, Byte[], OpcText)	7
Confirm(OpcContext)	7
Confirm(OpcContext, Byte[], OpcText)	8
Confirm(OpcContext, OpcText)	8
ConfirmCore(OpcNodeContext<OpcConditionNode>, Byte[], OpcText)	9
CreateBranchCore()	10

