

# OpcDialogConditionNode Members

**Namespace:** Opc.UaFx

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

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

## Constructors

### OpcDialogConditionNode(IOpcNode, OpcName)

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

**C#**

```
public OpcDialogConditionNode(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 dialog condition node can be accessed.

### OpcDialogConditionNode(IOpcNode, OpcName, OpcNodeId)

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

**C#**

```
public OpcDialogConditionNode(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 dialog condition node can be accessed.

**id** [OpcNodeId](#)

The [OpcNodeId](#) through that the new dialog condition node can be identified and accessed.

## OpcDialogConditionNode(OpcName)

Initializes a new instance of the [OpcDialogConditionNode](#) class accessible by the [name](#) specified.

**C#**

```
public OpcDialogConditionNode(OpcName name)
```

### Parameters

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

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

## OpcDialogConditionNode(OpcName, OpcNodeId)

Initializes a new instance of the [OpcDialogConditionNode](#) class accessible by the [name](#) and [id](#) specified.

**C#**

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

### Parameters

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

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

[id](#) [OpcNodeId](#)

The [OpcNodeId](#) through that the new dialog condition node can be identified and accessed.

# Properties

## CancelResponse

Gets or sets a value which defines the index of the response in the [ResponseOptions](#) array that will cause the dialog to go into the inactive state without proceeding with the operation described by the prompt.

**C#**

```
public int CancelResponse { get; set; }
```

### Property Value

[Int32](#)

The value allows a client to identify the Cancel option if a special handling for this option is available. If no

Cancel option is available the value is -1.

## CancelResponseNode

Gets the [OpcPropertyNode`1](#) of the [CancelResponse](#) property.

**C#**

```
public OpcPropertyNode<int> CancelResponseNode { get; }
```

### Property Value

[OpcPropertyNode<Int32>](#)

An instance of the [OpcPropertyNode`1](#) class.

## DefaultResponse

Gets or sets a value which identifies the response option that should be shown as default to the user.

**C#**

```
public int DefaultResponse { get; set; }
```

### Property Value

[Int32](#)

The index to an item in the [ResponseOptions](#) array. If no response option is the default, the value is -1.

## DefaultResponseNode

Gets the [OpcPropertyNode`1](#) of the [DefaultResponse](#) property.

**C#**

```
public OpcPropertyNode<int> DefaultResponseNode { get; }
```

### Property Value

[OpcPropertyNode<Int32>](#)

An instance of the [OpcPropertyNode`1](#) class.

## DefaultTypeDefinitionId

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 DefaultTypeDefinitionId { 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).

## IsActive

Gets a value indicating whether the dialog is active and is waiting for a response.

### C#

```
public bool IsActive { get; }
```

## Property Value

### Boolean

A value indicating whether the dialog is active and is waiting for a response.

## IsActiveNode

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

### C#

```
public OpcTwoStateVariableNode IsActiveNode { get; }
```

## Property Value

### OpcTwoStateVariableNode

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

## LastResponse

Gets or sets a value which defines the last response provided by a client in the respond method.

### C#

```
public int LastResponse { get; set; }
```

## Property Value

### Int32

The index to an item in the [ResponseOptions](#) array. If no previous response exists then the value is -1.

# LastResponseNode

Gets the [OpcPropertyNode`1](#) of the [LastResponse](#) property.

**C#**

```
public OpcPropertyNode<int> LastResponseNode { get; }
```

## Property Value

[OpcPropertyNode<Int32>](#)

An instance of the [OpcPropertyNode`1](#) class.

# OkResponse

Gets or sets a value which defines the index of the OK option in the [ResponseOptions](#) array.

**C#**

```
public int OkResponse { get; set; }
```

## Property Value

[Int32](#)

This choice is the response that will allow the system to proceed with the operation described by the prompt. This allows a client to identify the OK option if a special handling for this option is available. If no OK option is available the value is -1.

# OkResponseNode

Gets the [OpcPropertyNode`1](#) of the [OkResponse](#) property.

**C#**

```
public OpcPropertyNode<int> OkResponseNode { get; }
```

## Property Value

[OpcPropertyNode<Int32>](#)

An instance of the [OpcPropertyNode`1](#) class.

# Prompt

Gets or sets the dialog prompt to be shown to the user.

**C#**

```
public OpcText Prompt { get; set; }
```

## Property Value

### OpcText

The message to be shown to the user.

## PromptNode

Gets the [OpcTextPropertyNode](#) of the [Prompt](#) property.

### C#

```
public OpcTextPropertyNode PromptNode { get; }
```

## Property Value

### OpcTextPropertyNode

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

## RespondCallback

Gets or sets a callback used to respond on a dialog of the [OpcDialogConditionNode](#).

### C#

```
public OpcDialogRespondCallback RespondCallback { get; set; }
```

## Property Value

### OpcDialogRespondCallback

A [OpcDialogRespondCallback](#) used to respond on a dialog of the dialog condition node. The value can also be a null reference (Nothing in Visual Basic).

## RespondNode

Gets the [OpcDialogResponseMethodNode](#) used to handle 'Respond' method calls to respond on a dialog of the condition node.

### C#

```
public OpcDialogResponseMethodNode RespondNode { get; }
```

## Property Value

### OpcDialogResponseMethodNode

An instance of the [OpcDialogResponseMethodNode](#) class. Which uses an [OpcDialogConditionNode](#) defined callback to respond on a dialog of the condition node.

# ResponseOptions

Gets or sets an array of supported response options.

**C#**

```
public OpcText[] ResponseOptions { get; set; }
```

## Property Value

[OpcText\[\]](#)

An array of supported response options. The index in this array is used for the corresponding properties like [DefaultResponse](#), [LastResponse](#) and the selected response in the respond method.

# ResponseOptionsNode

Gets the [OpcTextArrayPropertyNode](#) of the [ResponseOptions](#) property.

**C#**

```
public OpcTextArrayPropertyNode ResponseOptionsNode { get; }
```

## Property Value

[OpcTextArrayPropertyNode](#)

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

# Methods

## CreateBranchCore()

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

**C#**

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

## Returns

[OpcConditionNode](#)

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

## Respond(OpcContext, Int32)

Responses on the dialog of the dialog condition node and changes the [IsActive](#) to its [FalseState](#) using the

specified `context`.

## C#

```
public void Respond(OpcContext context, int selectedResponse)
```

### Parameters

`context` [OpcContext](#)

The [OpcContext](#) to use.

`selectedResponse` [Int32](#)

The selected index of the item in the [ResponseOptions](#).

### Exceptions

[ArgumentNullException](#)

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

[OpcException](#)

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

## RespondCore(OpcNodeContext<OpcDialogConditionNode>, Int32)

Responds on the dialog of the dialog condition node using the `context` and `selectedResponse` information specified.

## C#

```
protected virtual OpcStatusCode RespondCore(OpcNodeContext<OpcDialogConditionNode> context,
int selectedResponse)
```

### Parameters

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

The [OpcNodeContext](#)'s to use to respond on the dialog of the dialog condition node.

`selectedResponse` [Int32](#)

The selected index of the item in the [ResponseOptions](#).

### Returns

[OpcStatusCode](#)

The [OpcStatusCode](#) specifying the outcome of the respond using the [RespondCallback](#) or [BadNotSupported](#) if there is no custom callback routine defined.



# Table of Contents

<b>Constructors</b>	1
OpcDialogConditionNode(IOpcNode, OpcName)	1
OpcDialogConditionNode(IOpcNode, OpcName, OpcNodeId)	1
OpcDialogConditionNode(OpcName)	2
OpcDialogConditionNode(OpcName, OpcNodeId)	2
<b>Properties</b>	2
CancelResponse	2
CancelResponseNode	3
DefaultResponse	3
DefaultResponseNode	3
DefaultTypeDefinitionId	3
IsActive	4
IsActiveNode	4
LastResponse	4
LastResponseNode	5
OkResponse	5
OkResponseNode	5
Prompt	5
PromptNode	6
RespondCallback	6
RespondNode	6
ResponseOptions	7
ResponseOptionsNode	7
<b>Methods</b>	7
CreateBranchCore()	7
Respond(OpcContext, Int32)	7
RespondCore(OpcNodeContext<OpcDialogConditionNode>, Int32)	8

