OpcReferenceType Enum

Namespace: Opc.UaFx
Assemblies: Opc.UaFx.Advanced.dll

Identifies how source nodes are related to target nodes and generally reflects an operation between the two, such as 'A contains B'.

C#

```
[Obfuscation]
public enum OpcReferenceType
```

Inheritance Object › ValueType › Enum › OpcReferenceType

Attributes ObfuscationAttribute

Fields

Name	Value	Description
Aggregates	44	Defines an abstract OpcReferenceType; only subtypes of it can be used. It is a subtype of the HasChild reference type. The semantic is to indicate a part (the target node) belongs to the source node. It does not specify the ownership of the target node. There are no constraints defined for this abstract OpcReferenceType.
AlwaysGeneratesEvent	3065	Defines a concrete OpcReferenceType and can be used directly. It is a subtype of GeneratesEvent. The semantic of this reference type is to identify the types of events methods have to generate on each method call. The source node of references of this type shall be a method. The target node of this reference type shall be an object type representing event types, that is, the base event type or one of its subtypes.
FromState	51	Defines a concrete OpcReferenceType especially for state machines and can be used directly. It is a subtype of NonHierarchicalReferences. The semantic of this reference type is to point form a transition to the starting state the transition connects. The source node of this reference type shall be an object of the object type transition type or one of its subtypes. The target node of this reference type shall be an object of the object type state type or one of its subtypes.

Name	Value	Description
GeneratesEvent	41	Defines a concrete OpcReferenceType and can be used directly. It is a subtype of NonHierarchicalReferences. The semantic of this reference type is to identify the types of events instances of object types or variable types may generate and methods may generate on each method call. The source node of references of this type shall be an object type, a variable type or a method. The target node of this reference type shall be an object type representing event types, that is, the base event type or one of its subtypes.
HasCause	53	Defines a concrete ReferenceType especially for state machines and can be used directly. It is a subtype of NonHierarchicalReferences. The semantic of this reference type is to point form a transition to something that causes the transition. In this annex we only define methods as causes. However, the reference type is not restricted to point to methods. The referenced methods can, but do not have to point to a method of the state machine type. For example, it is allowed to point to a server-wide restart method leading the state machine to go into its initial state. The source node of this reference type shall be an object of the object type transition type or one of its subtypes. The target node can be of any OpcNodeCategory.
HasChild	34	Defines an abstract OpcReferenceType; only subtypes of it can be used. It is a subtype of the HierarchicalReferences reference type. The semantic is to indicate that references of this type span a non-looping hierarchy. Starting from Node 'A' and only following references of the subtypes of the HasChild reference type shall never be able to return to 'A'. But it is allowed that following the references there may be more than one path leading to another Node 'B'.

Name	Value	Description
HasComponent	47	Defines a concrete OpcReferenceType that can be used directly. It is a subtype of the Aggregates reference type. The semantic is a part-of relationship. The target node of a reference of the HasComponent reference type is a part of the source node. This reference type is used to relate objects or object types with their containing objects, data variables and methods as well as complex variables or variable types with their data variables. Like all other reference types, this reference type does not specify anything about the ownership of the parts, although it represents a part-of relationship semantic. That is, it is not specified if the target node of a reference of the HasComponent reference type is deleted when the source node is deleted. The target node of this reference type shall be a variable, an object or a method. If the target node is a variable, the source node shall be an object, an objectType, a data variable or a variable type. By using the HasComponent reference, the variable is defined as data variable. If the target node is an object or a method, the source node shall be an object or object type.
HasCondition	9006	Defines a concrete OpcReferenceType especially for alarms and conditions and can be used directly. It is a subtype of NonHierarchicalReferences. The semantic of this reference type is to specify the relationship between a condition source and its conditions. Each condition source shall be the target of a HasEventSource reference or a sub type of HasEventSource. HasCondition references can be used in the type definition of an object or a variable. In this case, the source node of this reference type shall be an object type or variable type node or one of their instance declaration nodes. The target node shall be a condition instance declaration or a condition type. The following rules for instantiation apply: * All HasCondition references used in a type shall exist in instances of these types as well. * If the target node in the type definition is a condition type, the same target node will be referenced on the instance. HasCondition references may be used solely in the instance space when they are not available in type definitions. In this case the source node of this reference type shall be an object, variable or method node. The target node shall be a condition instance or a condition type.

Name	Value	Description
HasDescription	39	Defines a concrete OpcReferenceType and can be used directly. It is a subtype of NonHierarchicalReferences. The semantic of this reference type is to reference the data type description of a data type encoding. The source node of references of this type shall be an object of the object type data type encoding type or one of its subtypes. The target node of this reference type shall be a variable of the variable type data type description type or one of its subtypes.
HasEffect	54	Defines a concrete OpcReferenceType especially for state machines and can be used directly. It is a subtype of NonHierarchicalReferences. The semantic of this reference type is to point form a transition to something that will be effected when the transition is triggered. In this annex we only define event types as effects. However, the reference type is not restricted to point to event types. The source node of this reference type shall be an object of the object type transition type or one of its subtypes. The target node can be of any OpcNodeCategory.
HasEncoding	38	Defines a concrete OpcReferenceType and can be used directly. It is a subtype of NonHierarchicalReferences. The semantic of this reference type is to reference data type encodings of a data type. The source node of references of this type shall be a data type. The target node of this reference type shall be an object of the object type data type encoding type or one of its subtypes.
HasEventSource	36	Defines a concrete OpcReferenceType and can be used directly. It is a subtype of HierarchicalReferences. The semantic of this reference type is to relate event sources in a hierarchical, non-looping organization. This reference type and any subtypes are intended to be used for discovery of event generation in a server. They are not required to be present for a server to generate an event from its source (causing the event) to its notifying nodes. In particular, the root notifier of a server, the server object, is always capable of supplying all events from a server and as such has implied HasEventSource references to every event source in a server. The source node of this reference type shall be an object that is a source of event subscriptions. A source of event subscriptions is an object that has its 'SubscribeToEvents' bit set within the EventNotifier attribute. The target node of this reference type can be a node of any OpcNodeCategory that can generate event notifications via a subscription to the reference source. Starting from node 'A' and only following references of the HasEventSource reference type or of its subtypes shall never be able to return to 'A'. But it is permitted that, following the references, there may be more than one path leading to another Node 'B'.

Name	Value	Description
HasFalseSubState	9005	Defines a concrete OpcReferenceType especially for alarms and conditions that can be used directly. It is a subtype of the NonHierarchicalReferences reference type. The semantics indicate that the sub state (the target node) is a subordinate state of the FALSE super state. If more than one state within a condition is a sub state of the same super state (i.e. several HasFalseSubState references exist for the same super state) they are all treated as independent substates. The source node of the reference shall be an instance of a two state variable type and the target node shall either be an instance of a two state variable type or an instance of a subtype of a state machine type. It is not required to provide the has false sub state reference from super state to sub state, but it is required that the sub state provides the inverse reference to its super state.
HasHistoricalConfiguration	56	Defines a concrete OpcReferenceType especially for historical access that can be used directly. It is a subtype of the Aggregates reference type and will be used to refer from a historical node to one or more historical data configuration type objects.
HasModellingRule	37	Defines a concrete OpcReferenceType and can be used directly. It is a subtype of NonHierarchicalReferences. The semantic of this reference type is to bind the modelling rule to an object, variable or method. The source node of this reference type shall be an object, variable or method. The target node shall be an object of the object type 'modelling rule' or one of its subtypes. Each node shall be the source node of at most one HasModellingRule reference.
HasNotifier	48	Defines a concrete OpcReferenceType and can be used directly. It is a subtype of HasEventSource. The semantic of this reference type is to relate object nodes that are notifiers with other notifier object nodes. The reference type is used to establish a hierarchical organization of event notifying objects. It is a subtype of the HasEventSource reference type. The source node of this reference type shall be objects or views that are a source of event subscriptions. The target node of this reference type shall be objects that are a source of event subscriptions. A source of event subscriptions is an object that has its 'SubscribeToEvents' bit set within the EventNotifier attribute. If the target node of a reference of this type generates an event, then this event shall also be provided in the source node of the reference.

Name	Value	Description
HasOrderedComponent	49	Defines a concrete OpcReferenceType that can be used directly. It is a subtype of the HasComponent reference type. The semantic of the HasOrderedComponent reference type – besides the semantic of the HasComponent reference type – is that when browsing from a node and following references of this type or its subtype all references are returned in the browse service in a well defined order. The order is server-specific, but the client can assume that the server always returns them in the same order. There are no additional constraints defined for this OpcReferenceType.
HasProperty	46	Defines a concrete OpcReferenceType that can be used directly. It is a subtype of the Aggregates reference type. The semantic is to identify the properties of a node.
HasSubStateMachine	117	Defines a concrete OpcReferenceType especiall for state machines and can be used directly. It is a subtype of NonHierarchicalReferences. The semantic of this reference type is to point from a state to an instance of a state machine type which represents the sub-states for the state. The source node of this reference type shall be an object of the object type state type. The target node shall be an object of the object type state machine type or one of its subtypes. Each object can be the target node of at most one HasSubStateMachine Reference. The source node (the state) and the target node (the sub state machine) shall belong to the same state machine, that is, both shall be referenced from the same object of type state machine type using a HasComponent reference or a subtype of HasComponent.
HasSubtype	45	Defines a concrete OpcReferenceType that can be used directly. It is a subtype of the HasChild reference type. The semantic of this reference type is to express a subtype relationship of types. It is used to span the reference type hierarchy; a data type hierarchy, as well as other subtype hierarchies. The source node of references of this type shall be an object type, a variable type, a data type or a reference type and the target node shall be of the same OpcNodeCategory as the source node. Each reference type shall be the target node of at most one reference of type HasSubtype.

Name	Value	Description
HasTrueSubState	9004	Defines a concrete OpcReferenceType especially for alarms and conditions that can be used directly. It is a subtype of the NonHierarchicalReferences reference type. The semantics indicate that the sub state (the target node) is a subordinate state of the TRUE super state. If more than one state within a condition is a sub state of the same super state (i.e. several HasTrueSubState references exist for the same super state) they are all treated as independent substates. The source node of the reference shall be an instance of a two state variable type and the target node shall either be an instance of a two state variable type or an instance of a subtype of a state machine type. It is not required to provide the HasTrueSubState reference from super state to sub state, but it is required that the sub state provides the inverse reference to its super state.
HasTypeDefinition	40	Defines a concrete OpcReferenceType and can be used directly. It is a subtype of NonHierarchicalReferences. The semantic of this reference type is to bind an object or variable to its object type or variable type, respectively. The source node of this reference type shall be an object or variable. If the source node is an object, then the target node shall be an object type; if the source node is a variable, then the target node shall be a variable type. Each variable and each object shall be the source node of exactly one HasTypeDefinition reference.
HierarchicalReferences	33	Defines an abstract OpcReferenceType; only subtypes of it can be used. The semantic of HierarchicalReferences is to denote that References of HierarchicalReferences span a hierarchy. It means that it may be useful to present nodes related with References of this type in a hierarchical-like way. HierarchicalReferences does not forbid loops. For example, starting from Node 'A' and following HierarchicalReferences may lead to browse to node 'A', again. It is not permitted to have a property as source node of a reference of any subtype of this abstract OpcReferenceType. It is not allowed that the source node and the target node of a reference of the OpcReferenceTypeHierarchicalReferences are the same, that is, it is not allowed to have self references using HierarchicalReferences.

Name	Value	Description
NonHierarchicalReferences	32	Defines an abstract OpcReferenceType; only subtypes of it can be used. The semantic of NonHierarchicalReferences is to denote that its subtypes do not span a hierarchy and should not be followed when trying to present a hierarchy. To distinguish hierarchical and non-hierarchical references, all concrete reference types shall inherit from either hierarchical references or non-hierarchical references, either direct or indirect. There are no constraints defined for this abstract OpcReferenceType.
Organizes	35	Defines a concrete OpcReferenceType and can be used directly. It is a subtype of HierarchicalReferences. The semantic of this reference type is to organise nodes in the address space. It can be used to span multiple hierarchies independent of any hierarchy created with the non-looping Aggregates references. The source node of references of this type shall be an object or a view. If it is an object then it should be an object of the object type folder or one of its subtypes. The target node of this OpcReferenceType can be of any OpcNodeCategory.
References	31	Defines an abstract OpcReferenceType; only subtypes of it can be used. There is no semantic associated with this OpcReferenceType. This is the base type of all reference types. All OpcReferenceTypes shall be a subtype of this base reference type – either direct or indirect. The main purpose of this OpcReferenceType is allowing simple filter and queries in the corresponding services. There are no constraints defined for this abstract OpcReferenceType.
ToState	52	Defines a concrete OpcReferenceType especially for state machines and can be used directly. It is a subtype of NonHierarchicalReferences. The semantic of this reference type is to point form a transition to the ending state the transition connects. The source node of this reference type shall be an object of the object type transition type or one of its subtypes. The target node of this reference type shall be an object of the object type state type or one of its subtypes. References of this reference type may be only exposed uni-directional. Sometimes this is required, for example, if a transition points to a state of a sub-machine.

Table of Contents