

Attributes and Associations of Class Property

By Anneke Kleppe, Klasse Objecten (a.kleppe@klasse.nl)

Introduction

The UML Infrastructure (3rd revised submission, dated 3 March 2003, document number: ad/2003-03-01) has recently been approved by the OMG ADFT. As one who has an interest in the UML and one who has to work with the new version of UML, I have studied the material. My conclusion is that it is very difficult to find out which attributes and associations are defined for a certain (meta)class.

I have worked this aspect out for the class Property in the package Constructs. Because of all the effort it took, I like to share this information with others who are interested. In this way not everyone has to undertake the same time-consuming task.

This document contains an inheritance tree for the class Property and a listing of all attributes and associations.

From Core::Abstractions::Elements::Element, page 34

none

From Core::Abstractions::Ownerships::Element, page 60

Associations

- / ownedElement: Element[*] The Elements owned by this element. This is a derived union.
- / owner: Element [0..1] The Element that owns this element. This is a derived union.

From Core::Basic::NamedElement, page 76

Attributes

- name: String [0..1]. The name of the element.

From Core::Abstractions::Multiplicities:: MultiplicityElement, page 52

Attributes

- isOrdered: Boolean For a multivalued multiplicity, this attribute specifies whether the values in an instantiation of this element are sequentially ordered. Default is *false*.
- isUnique : Boolean For a multivalued multiplicity, this attributes specifies whether the values in an instantiation of this element are unique. Default is *true*.
- lower : Integer [0..1] Specifies the lower bound of the multiplicity interval. Default is one.
- upper : UnlimitedNatural [0..1] Specifies the upper bound of the multiplicity interval. Default is one.

From Core::Abstractions::NameSpaces::NamedElement, page 57

Attributes

- name: String [0..1] The name of the NamedElement.
- / qualifiedName: String [0..1] A name which allows the NamedElement to be identified within a hierarchy of nested Namespaces. It is constructed from the names of the containing namespaces starting at the root of the hierarchy and ending with the name of the

NamedElement itself. This is a derived attribute.

Associations

- / namespace: Namespace [0..1] Specifies the namespace that owns the NamedElement. Subsets *Element::owner*. This is a derived union.

From Core::Abstractions::Comments::Element, page 29

Associations

- ownedComment: Comment[*] The Comments owned by this element. Subsets *Element::ownedElement*.

From Core::Basic::TypedElement, page 77

Attributes

- type: Type [0..1]. The type of the element.

From Core::Abstractions::Visibilities::NamedElement, page 72

Attributes

- visibility: VisibilityKind [0..1] Determines the visibility of the NamedElement within different Namespaces within the overall model.

From Core::Abstractions::TypedElements::TypedElement, page 71

Associations

- type: Type [0..1] The type of the TypedElement.

From Core::Constructs::Element, page 88

Associations

- ownedComment: Comment[*] Redefines the corresponding property in Abstractions. Subsets *Element::ownedElement*.
- /ownedElement: Element[*] Redefines the corresponding property in Abstractions. This is a derived union.
- /owner: Element[0..1] Redefines the corresponding property in Abstractions. This is a derived union.

From Core::Basic::Property, page 80

Attributes

- class : Class [0..1] The class that owns the property, and of which the property is an attribute.
- default : String [0..1] A string that is evaluated to give a default value for the attribute when an object of the owning class is instantiated.
- isComposite : Boolean If isComposite is true, the object containing the attribute is a container for the object or value contained in the attribute. The default value is false.
- isDerived : Boolean If isDerived is true, the value of the attribute is derived from information elsewhere. The default value is false.
- isReadOnly : Boolean If isReadOnly is true, the attribute may not be written to after initialization. The default value is false.
- opposite : Property [0..1] Two attributes attr1 and attr2 of two objects o1 and o2 (which may be the same object) may be paired with each other so that o1.attr1 refers to o2 if and only if o2.attr2 refers to o1. In such a case attr1 is the opposite of attr2 and attr2 is the opposite of attr1.

From Core::Abstractions::Redefinitions::RedefinableElement, page 62

Associations

- /redefinedElement: RedefinableElement[*]The redefinable element that is being redefined by this element. This is a derived union.
- /redefinitionContext: Classifier[*]References the contexts that this element may be redefined from. This is a derived union.

From Core::Constructs::NamedElement, page 121

Attributes

- name: String [0..1] Redefines the corresponding attributes from Basic::NamedElement and Abstractions::Visibilities::NamedElement.

Associations

- namespace: NamedElement [0..1]The Namespace that owns this NamedElement. Redefines the corresponding property from Abstractions::Namespaces::NamedElement.

From Core::Constructs::MultiplicityElement, page 108

none

From Core::Constructs::RedefinableElement, page 109

Associations

- /redefinedElement: RedefinableElement[*]This derived union is redefined from Abstractions.
- /redefinitionContext: Classifier[*]This derived union is redefined from Abstractions.

From Core::Constructs::TypedElement, page 110

none

From Core::Constructs::Feature, page 108

Associations

- featuringClassifier : Classifier [1..*]
Redefines the corresponding association in Abstractions. This is a derived union.

From Core::Constructs::StructuralFeature, page 109

none

From Core::Constructs::Property, page 103

Attributes

- isDerivedUnion : Boolean Specifies whether the property is derived as the union of all of the properties that are constrained to subset it. The default value is *false*.
- isReadOnly : Boolean This redefines the corresponding attribute in Basic::Property and Abstractions::StructuralFeature. The default value is *false*.

Associations

- association: Association [0..1] References the association of which this property is a member, if any.
- owningAssociation: Association [0..1] References the owning association of this property, if any. Subsets *Property::association*, *NamedElement::namespace*, and *Feature::featuringClassifier*.
- redefinedProperty : Property [*] References the properties that are redefined by this property. Subsets *RedefinableElement::redefinedElement*.

- `subsettingProperty` : `Property` [*]

References the properties of which this property is constrained to be a subset.

- `/ opposite` : `Property` [0..1] In the case where the property is one navigable end of a binary association with both ends navigable, this gives the other end.

Additional to Core::Constructs::Property, page 118

Associations

- `datatype` : `DataType` [0..1] The `DataType` that owns this `Property`. Subsets `NamedElement::namespace`, `Feature::featuringClassifier`, and `Property::classifier`.

