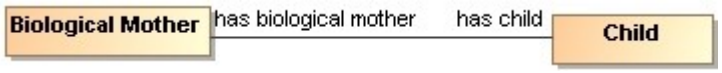
Object properties

Object properties have the following property characteristics: Inverse, Symmetric, Asymmetric, Reflexive, Irreflexive, Functional, Inverse Functional, and Transitive. An example accompanies each characteristic. The modeling tool supports exporting each characteristic; the respective OWL ontology will accompany each example.

Inverse Object Property

Sometimes you can invert directions of properties between classes. As shown in the example below, the property *hasBiologicalMother* can be seen as the inverse property of *hasChild.* Furthermore, you can see the Biological Mother and Child are connected by *hasChild,* and Child and Biological Mother are connected by *hasBiologicalMother.*

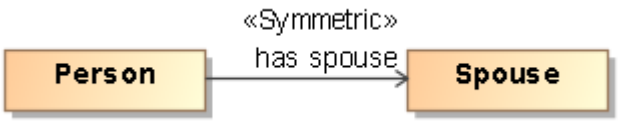
The OWL ontology is described here: Class with inverse property



Symmetric Object Property

At other times, you may want to connect classes which have the same properties. This can be done with a symmetric object property. The example below shows a Person having a Spouse. This is symmetric because that same Spouse would also have a spouse which is a Person.

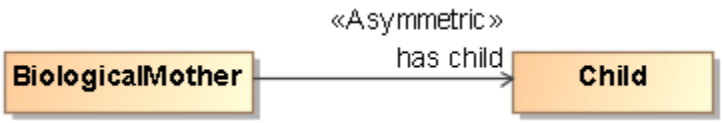
The OWL ontology is described here: Class with symmetric object property



Asymmetric Object Property

Conversely, you may want to characterize properties that are asymmetric. For instance, a biological mother has a child but a child would not have a child and characterize them as a biological mother. Hence, this is property of *hasChild* is asymmetric between a biological mother and her child.

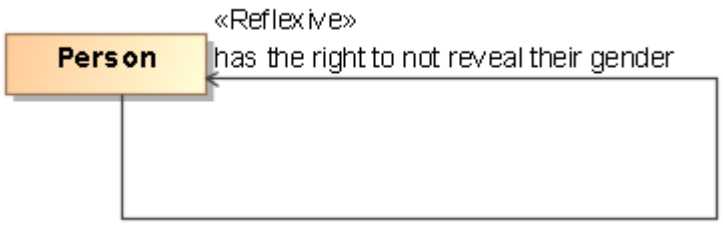
The OWL ontology is described here: Class with Asymmetric Object Property



Reflexive Object Property

This object property is used to depict properties that are reflexive to the class. For example, a Person has the right to not reveal their gender on an arbitrary application. This property relates only to that Person.

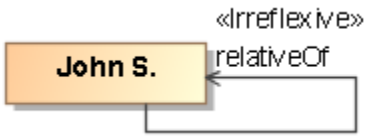
The OWL ontology is described here: Class with Reflexive Object Property



Irreflexive Object Property

On the other hand, some properties are irreflexive, meaning you cannot relate this property to the class itself. For example. John S. is not a relative of himself. Hence, the property *relativeOf* is irreflexive to John S.

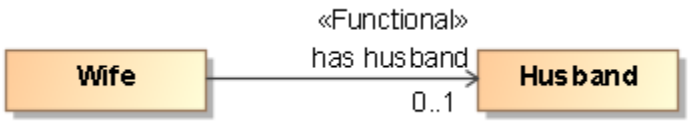
The OWL ontology is described here: Class with Irreflexive Object Property



Functional Object Property

A functional object property has a multiplicity 0..1 attached to it. The example below is about a husband and a wife, and the functional property *hasHusband* shows that the wife can have at most one husband.

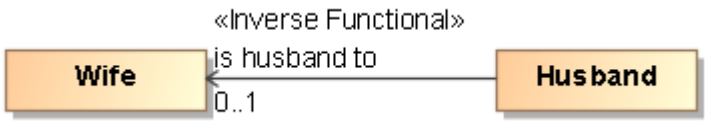
The OWL ontology is described here: Class with Functional Object Property



Inverse Functional Object Property

Like the functional object property, the inverse functional object property also has a multiplicity 0..1 attached to it. However, the inverse functional object property, according to the example below, shows that the husband can be a husband to at most one individual, a wife.

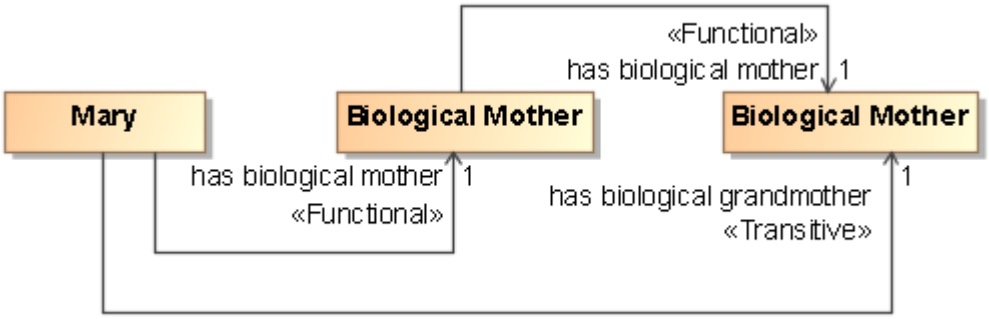
The OWL ontology is described here: Class with Inverse Functional Object Property



Transitive Object Property

The transitive object property connects two classes when one of the linked class has a connection with another class. The example below shows the transitive object property between Mary and her grandmother.

The OWL ontology is described here: Class with Transitive Object Property



Related Pages

- Concept Modeling Semantics