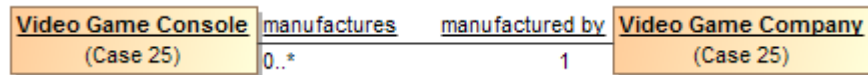# Inverse properties

A property is a unidirectional relation between two classes, or between a class and a datatype. In the case where there is a relation between two classes, it is often useful to define a property that goes in the opposite direction. For example, if a Video Game Company **manufactures** Video Game Consoles, the opposite would be that a Video Game Console is **manufactured by** a Video Game Company. Rather than draw two separate unidirectional associations, properties drawn on opposite ends of one association are *inverses* of one another. When an instance has a value for a property that has an inverse defined, a reasoner can infer that an opposite value also exists, and automatically create it.

The next diagram asserts that for every (Video Game Console, Video Game Company) related by the **manufactures** property, there is a corresponding (Video Game Company, Video Game Console) related by the **manufactured by** property.

Inverse properties shown on the opposite ends of an association.

Importing Inverse Properties

In most cases, when importing an OWL ontology, information in OWL is enough to assert that two properties are the inverse of each other.

- However, if the definition is insufficient to prove that two are inverses of each other, or which class owns the property and what the type is, the Concept Modeler will create **two unidirectional associations,** and use a stereotyped dependency «Inverse of» between the properties to show that they are inverses of one another.
- An OWL property with **multiple inverses** now results in an «Equivalent Property» and an «Inverse of» instead of resulting in a repeatedly subsetted UML property.

Related page

- Concept Modeling Semantics