**Relationship**

In UML, a relationship is a connection between model elements. A UML relationship is a type of model element that adds semantics to a model by defining the structure and behavior between model elements.

Relationships in class diagrams show the interaction between classes and classifiers. Such relationships indicate the classifiers that are associated with each other, those that are generalizations and realizations, and those that have dependencies on other classes and classifiers.

The following topics describe the relationships that you can use in class diagrams:

* [**Abstraction relationships**](https://www.ibm.com/docs/en/SS8PJ7_9.7.0/com.ibm.xtools.modeler.doc/topics/cabstract.html)
An abstraction relationship is a dependency between model elements that represents the same concept at different levels of abstraction or from different viewpoints. You can add abstraction relationships to a model in several diagrams, including use-case, class, and component diagrams.
* [**Aggregation relationships**](https://www.ibm.com/docs/en/SS8PJ7_9.7.0/com.ibm.xtools.modeler.doc/topics/caggreg.html)
In UML models, an aggregation relationship shows a classifier as a part of or subordinate to another classifier.
* [**Association relationships**](https://www.ibm.com/docs/en/SS8PJ7_9.7.0/com.ibm.xtools.modeler.doc/topics/cassn.html)
In UML models, an association is a relationship between two classifiers, such as classes or use cases, that describes the reasons for the relationship and the rules that govern the relationship.
* [**Association classes**](https://www.ibm.com/docs/en/SS8PJ7_9.7.0/com.ibm.xtools.modeler.doc/topics/cassnclss.html)
In UML diagrams, an association class is a class that is part of an association relationship between two other classes.
* [**Binding relationships**](https://www.ibm.com/docs/en/SS8PJ7_9.7.0/com.ibm.xtools.modeler.doc/topics/cbind.html)
In UML models, a binding relationship is a relationship that assigns values to template parameters and generates a new model element from the template.
* [**Composition association relationships**](https://www.ibm.com/docs/en/SS8PJ7_9.7.0/com.ibm.xtools.modeler.doc/topics/ccompasn.html)
A composition association relationship represents a whole–part relationship and is a form of aggregation. A composition association relationship specifies that the lifetime of the part classifier is dependent on the lifetime of the whole classifier.
* [**Dependency relationships**](https://www.ibm.com/docs/en/SS8PJ7_9.7.0/com.ibm.xtools.modeler.doc/topics/cdepend.html)
In UML, a dependency relationship is a relationship in which one element, the client, uses or depends on another element, the supplier. You can use dependency relationships in class diagrams, component diagrams, deployment diagrams, and use-case diagrams to indicate that a change to the supplier might require a change to the client.
* [**Directed association relationships**](https://www.ibm.com/docs/en/SS8PJ7_9.7.0/com.ibm.xtools.modeler.doc/topics/cdirasn.html)
In UML models, directed association relationships are associations that are navigable in only one direction.
* [**Element import relationships**](https://www.ibm.com/docs/en/SS8PJ7_9.7.0/com.ibm.xtools.modeler.doc/topics/celeimport.html)
In UML diagrams, an element import relationship identifies a model element in another package, and allows the element in the other package to be referenced by using its name without a qualifier.
* [**Generalization relationships**](https://www.ibm.com/docs/en/SS8PJ7_9.7.0/com.ibm.xtools.modeler.doc/topics/cgeneral.html)
In UML modeling, a generalization relationship is a relationship in which one model element (the child) is based on another model element (the parent). Generalization relationships are used in class, component, deployment, and use-case diagrams to indicate that the child receives all of the attributes, operations, and relationships that are defined in the parent.
* [**Interface realization relationships**](https://www.ibm.com/docs/en/SS8PJ7_9.7.0/com.ibm.xtools.modeler.doc/topics/cimplement.html)
In UML diagrams, an interface realization relationship is a specialized type of implementation relationship between a classifier and a provided interface. The interface realization relationship specifies that the realizing classifier must conform to the contract that the provided interface specifies.
* [**Instantiation relationships**](https://www.ibm.com/docs/en/SS8PJ7_9.7.0/com.ibm.xtools.modeler.doc/topics/cinstantiate.html)
In UML diagrams, an instantiation relationship is a type of usage dependency between classifiers that indicates that the operations in one classifier create instances of the other classifier.
* [**Package import relationship**](https://www.ibm.com/docs/en/SS8PJ7_9.7.0/com.ibm.xtools.modeler.doc/topics/cpkgimport.html)
In UML diagrams, a package import relationship allows other namespaces to use unqualified names to refer to package members.
* [**Realization relationships**](https://www.ibm.com/docs/en/SS8PJ7_9.7.0/com.ibm.xtools.modeler.doc/topics/creal.html)
In UML modeling, a realization relationship is a relationship between two model elements, in which one model element (the client) realizes the behavior that the other model element (the supplier) specifies. Several clients can realize the behavior of a single supplier. You can use realization relationships in class diagrams and component diagrams.
* [**Usage relationships**](https://www.ibm.com/docs/en/SS8PJ7_9.7.0/com.ibm.xtools.modeler.doc/topics/cusage.html)
In UML modeling, a usage relationship is a type of dependency relationship in which one model element (the client) requires another model element (the supplier) for full implementation or operation.