

## Langfristige Beziehungen zwischen Objekten

### Inhalt

Themen.....	1
Typen von langfristigen Objektbeziehungen .....	2
Symbole und Begriffe.....	2
Abgrenzung von Objektbeziehung zu Klassenbeziehung.....	2
Dauer der Beziehung zwischen Objekten und Darstellung im Programm.....	2
Unterscheidung der langfristigen Objektbeziehungen .....	2
Gerichtete Beziehungen, Navigation zwischen Objekten.....	3
Typen gerichteter Beziehungen .....	3
Navigation zwischen Objekten .....	3
Darstellung im Programm .....	3

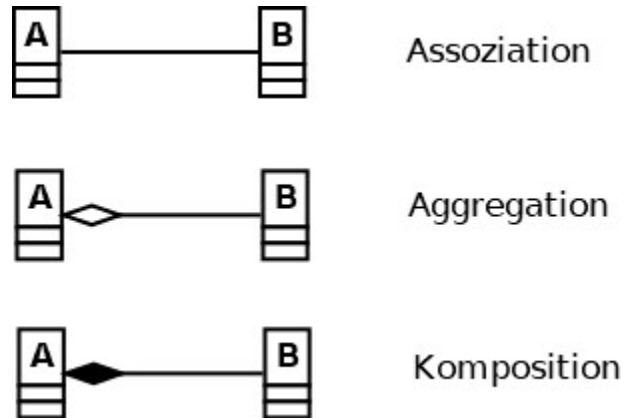
### Themen

- Typen von langfristigen Objektbeziehungen.
- Notation in einem UML-Klassendiagramm.
- Fachliche Bedeutung der Objektbeziehungen.
- Umsetzung von Objektbeziehungen im Programm.
- Verfeinerung mit gerichteten Beziehungen.

## Typen von langfristigen Objektbeziehungen

### Symbole und Begriffe

In der folgenden Abbildung sind die Symbole und Bezeichnungen für langfristige Objektbeziehungen dargestellt.



### Abgrenzung von Objektbeziehung zu Klassenbeziehung

Der Begriff ‚Objektbeziehung‘ bezeichnet die Beziehung zwischen Datenpaketen, also Objekten. Er steht damit im Gegensatz zu ‚Klassenbeziehung‘, also der Beziehung zwischen Bauplänen von Objekten, wie die Vererbung.

### Dauer der Beziehung zwischen Objekten und Darstellung im Programm

Der Begriff ‚langfristig‘ beschreibt die Dauer der Beziehung zwischen den Objekten. ‚Langfristig‘ bedeutet länger als der Aufruf einer Methode der beteiligten Klassen und steht im Gegensatz zu ‚kurzfristig‘, bei dem sich die Beziehung auf die Dauer eines Methodenaufrufs beschränkt. Langfristige Beziehungen werden in einem Programm durch Attribute dargestellt, z.B. durch ein Attribut vom Typ B in der Klasse A.

### Unterscheidung der langfristigen Objektbeziehungen

Die drei Beziehungstypen sind in der obigen Darstellung fortschreitend vom Allgemeinen zum Speziellen dargestellt, d.h. eine Aggregation ist immer auch eine Assoziation und eine Komposition ist auch eine Assoziation und eine Aggregation.

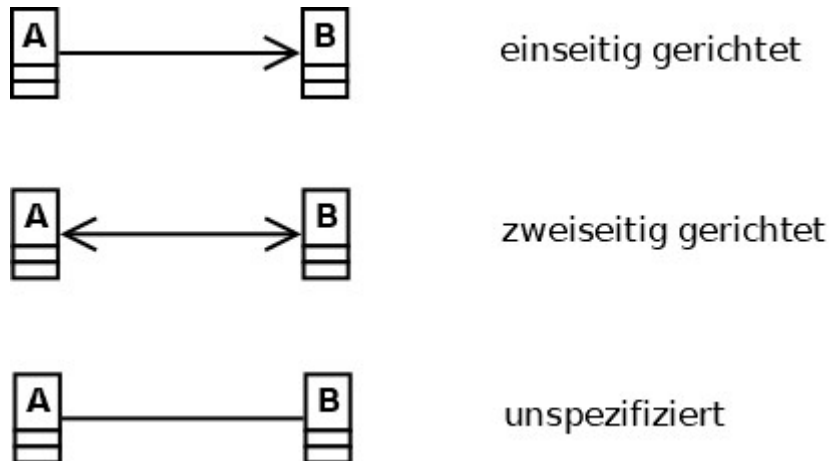
- Eine ‚Assoziation‘ ist eine allgemeine Beziehung zwischen Objekten.
- Eine ‚Aggregation‘ ist eine Assoziation, bei der ein Objekt der Klasse A Objekte der Klasse B enthält.
- Eine ‚Komposition‘ ist eine Aggregation, bei der die Objekte existenziell miteinander verbunden sind.

Die verschiedenen Typen von Beziehungen können in Programmiersprachen wie C# oder Java nicht zwingend umgesetzt werden (anders in C++). Im UML-Diagramm dient die Unterscheidung der Beziehungstypen also weniger der Beschreibung eines Programms als vielmehr des Sinns der Klassen im Programm.

## Gerichtete Beziehungen, Navigation zwischen Objekten

### Typen gerichteter Beziehungen

Mit Hilfe von Pfeilspitzen kann man bei Beziehungen die Navigationsrichtung beschreiben.



### Navigation zwischen Objekten

Unter ‚Navigation‘ versteht man die Möglichkeit, aus einem Objekt auf ein anderes Objekt zugreifen zu können. Bei dem ersten Beispiel (einseitig gerichtet) kann man aus einem Objekt der Klasse A auf das zugeordnete Objekt der Klasse B zugreifen und dessen Methoden aufrufen, aber nicht umgekehrt.

### Darstellung im Programm

Gerichtete Beziehungen legen also fest, in welcher Klasse ein Attribut der anderen Klasse vorliegen muss. Bei dem ersten Beispiel gibt es in der Klasse A ein Attribut vom Typ B. Beim zweiten Beispiel (zweiseitig gerichtet) gibt es zusätzlich in der Klasse B ein Attribut vom Typ A. In einem früheren Stadium der Modellierung macht man im Allgemeinen keine Aussage über die Navigationsrichtung und man lässt die Pfeilspitzen weg (unspezifiziert).