

种类： Generalization(泛化)、Dependency(依赖关系)、Association(关联关系)、Aggregation(聚合关系)、Composition(合成关系)。
其中 Aggregation(聚合关系)、Composition(合成关系)属于 Association(关联关系)，是特殊的 Association 关联关系。

Generalization(泛化)表现为继承或实现关系(is a)。具体形式为类与类之间的继承关系，接口与接口之间的继承关系，类对接口的实现关系。

Association 关联关系表现为变量(has a)。类与类之间的联接，它使一个类知道另一个类的属性和方法。例如如果 A 依赖于 B，则 B 体现为 A 的全局变量。关联关系有双向关联和单向关联。双向关联：两个类都知道另一个类的公共属性和操作。单向关联：只有一个类知道另外一个类的公共属性和操作。大多数关联应该是单向的，单向关系更容易建立和维护，有助于寻找可服用的类。

Aggregation(聚合关系) 是关联关系的一种，是强的关联关系。聚合关系是整体和个体的关系。普通关联关系的两个类处于同一层次上，而聚合关系的两个类处于不同的层次，一个是整体，一个是部分。同时，是一种弱的“拥有”关系。体现的是 A 对象可以包含 B 对象，但 B 对象不是 A 对象的组成部分。具体表现为，如果 A 由 B 聚合成，表现为 A 包含有 B 的全局对象，但是 B 对象可以不在 A 创建的时刻创建。

Composition(组合关系)是关联关系的一种，是比聚合关系强的关系。它要求普通的聚合关系中代表整体的对象负责代表部分对象的生命周期。Composition(组合关系)是一种强的“拥有”关系，体现了严格的部分和整体的关系，部分和整体的生命周期一致。如果 A 由 B 组成，表现为 A 包含有 B 的全局对象，并且 B 对象在 A 创建的时刻创建。

Dependency(依赖关系)表现为函数中的参数(use a)。是类与类之间的连接，表示一个类依赖于另一个类的定义，其中一个类的变化将影响另外一个类。例如如果 A 依赖于 B，则 B 体现为局部变量，方法的参数、或静态方法的调用。