

Systems Development: Object Oriented Analysis and Design

(H172 35)

Class Diagram Associations

**Taken from “Introductory Java”, David Parsons (ISBN 1-85805-366-8)**

“Objects that work alone do not produce very useful systems

An object oriented program consists of many objects **collaborating** to produce the required system behaviour

Programs can use any number of both pre-defined (built-in) and programmer-defined classes

When objects of different classes **communicate** with one another they are said to **associate**

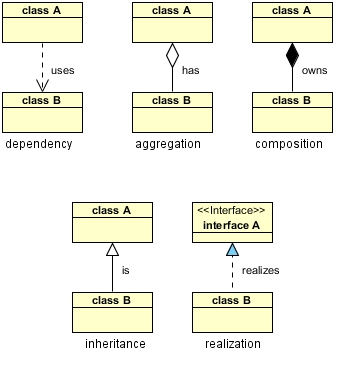
One particular kind of association is known as aggregation

Here groups of objects are used as components to make a larger object

Where aggregated objects are very tightly bound together, we refer to this as composition

Summary:

|  |  |
| --- | --- |
| **Association** | **Meaning** |
| Aggregation | “has” |
| Composition | “owns” / “consists of” |
|  |  |
| Inheritance | “is a” |
|  |  |
| Association / Dependency / Relationship | “uses” |
|  |  |
| Realisation / Implementation | “realises”  “implements” |



Dependency: class A uses class B

Aggregation: class A has a class B

Composition: class A owns a class B

Inheritance: class B is a Class A (or class A is extended by class B)

Realization: class B realizes interface A (or interface A is realized by class B)