

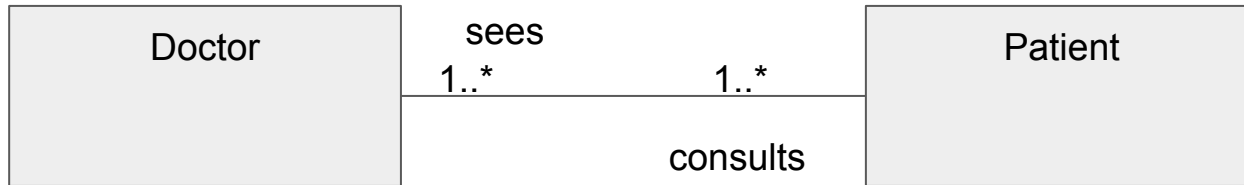
ESS 201: Programming in Java

Composition
Term 1: 2019-20
T K Srikanth

Relationships between objects

Association: a weak relationship between objects. E.g. Doctor and Patient

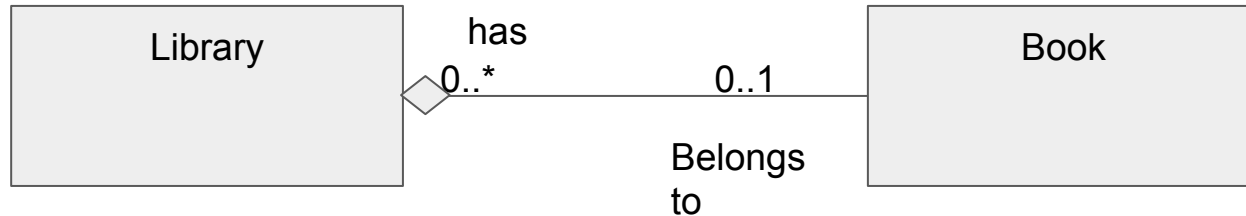
- A path of communication or link between objects, where one object “uses” possibly multiple instances of the other. No ownership or other semantics are implied
- The lifecycles of the objects are independently defined



Relationships: Aggregation

An object consists of (or is made up of) instances of objects of another class, but the lifecycle of the two objects are independent

E.g. Library and Books

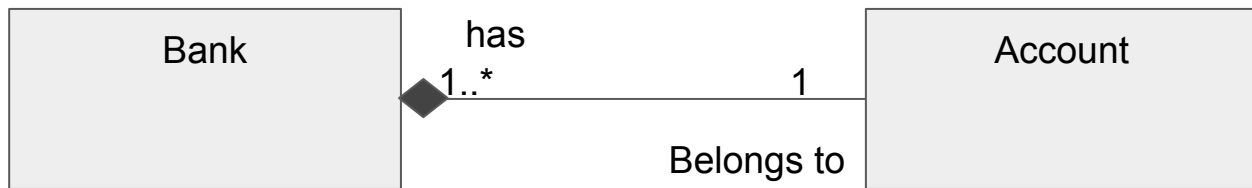


Relationships: Composition

An object contains (and *owns*) instances of another object, and the second object does not exist if the owning object is deleted.

E.g. Bank and Account

Existence dependency: Component exists only when its container is around



Composition - Re-use of Implementation

Composition enables the design of complex classes by re-using existing classes: we re-use implementation of these classes

The new class is able to leverage all the functionality of one or more existing classes.

However, it typically hides the existence of these classes - hopefully keeping the references private, and exposes new interfaces.