<https://www.linkedin.com/groups/Is-JIT-compiler-interpreter-name-70526.S.114937176>

<https://www.google.com/search?btnG=1&pws=0&q=JIT+compiler&safe=active&gws_rd=ssl>

<http://stackoverflow.com/questions/5074145/is-jit-compiler-a-compiler-or-interpreter>

<http://www.wsu.edu/UNIX_Systems/java/tools/java.html>

For two objects, Foo and Bar the relationships can be defined

Association - I have a relationship with an object. Foo uses Bar

public class Foo {

void Baz(Bar bar) {

}

};

Composition - I own an object and I am responsible for its lifetime, when Foo dies, so does Bar

public class Foo {

private Bar bar = new Bar();

}

Aggregation - I have an object which I've borrowed from someone else. When Foo dies, Bar may live on.

public class Foo {

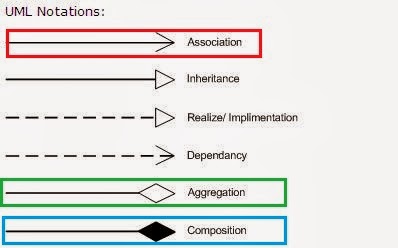
private Bar bar;

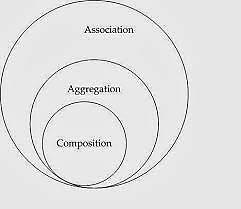
Foo(Bar bar) {

this.bar = bar;

}

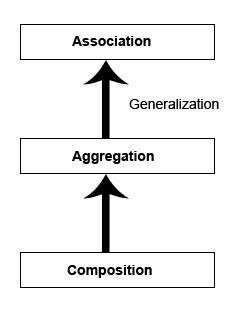
}



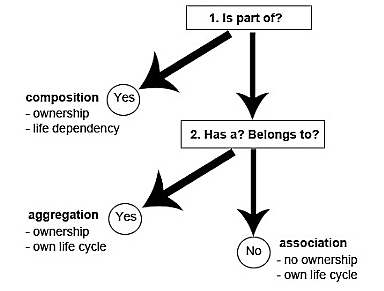
Relationships :

We will be discussing relationships between classes/objects.

**Association**:

1. One relationship between objects
2. http://javapapers.com/wp-content/uploads/2010/06/association.jpgObjects have their own life cycle
3. No ownership
4. It defines the multiplicity between objects. You may be aware of one-to-one, one-to-many, many-to-one, many-to-many all these words define an association between objects.

http://javapapers.com/wp-content/uploads/2010/06/aggregation.jpg**Aggregation** (has-a)

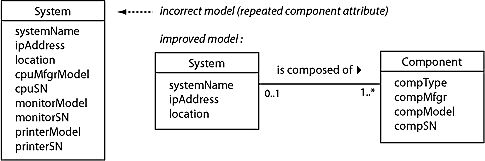
1. Objects have their own life cycle
2. http://javapapers.com/wp-content/uploads/2010/06/composition.jpgHas ownership (child object cannot belong to another parent object)

**Composition** (part-of)

1. Objects has life time dependency, no own life cycle
2. Has ownership (child object cannot belong to another parent object).

***Example:***A Library contains students and books. Relationship between library and student is aggregation. Relationship between library and book is composition.  A student can exist without a library and therefore it is aggregation. A book cannot exist without a library and therefore its a composition.

1. Body and arm = ?
2. Department and teacher=?
3. Teacher and student?
4. Project and project leader = ?
5. Order and orderline = ?
6. School and student = ?
7. Customer and order = ?
8. Project and student = ?
9. Human and soul=?
10. MAC address and Computer



**References**

<http://yanny0808.blogspot.com/2010/08/uml-association-aggregation-composition.html#comment-form>

1. Composition

2. Aggregation

3. Association

4. Composition

5. Composition

6. Aggregation

7. Aggregation

8. Aggregation