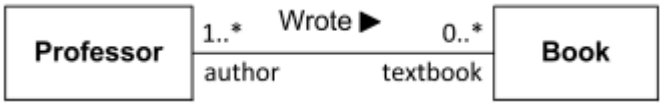
Advance class Modeling

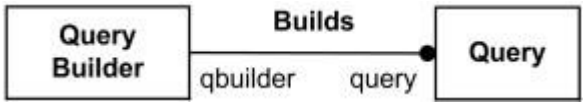
Advance object and class concepts and Association Ends.

- Association end is a connection between the line depicting an association and the icon depicting the connected classifier.
- Name of the association end may be placed near the end of the line. The association end name is commonly referred to as role name (but it is not defined as such in the UML 2.4 standard).
- The role name is optional and suppressible.



*Professor **"playing the role" of** author **is associated with** textbook **end typed as** Book.*

- - The idea of the role is that the same classifier can play the same or different roles in other associations.
- - For example, Professor could be an author of some Books or an editor.
- Association end could be owned either by end classifier, or association itself
- Association ends of associations with more than two ends must be owned by the association. Ownership of association ends by an associated classifier may be indicated graphically by a small filled circle (aka dot).
- The dot is drawn at the point where line meets the classifier.
- It could be interpreted as showing that the model includes a property of the type represented by the classifier touched by the dot.
- This property is owned by the classifier at the other end.

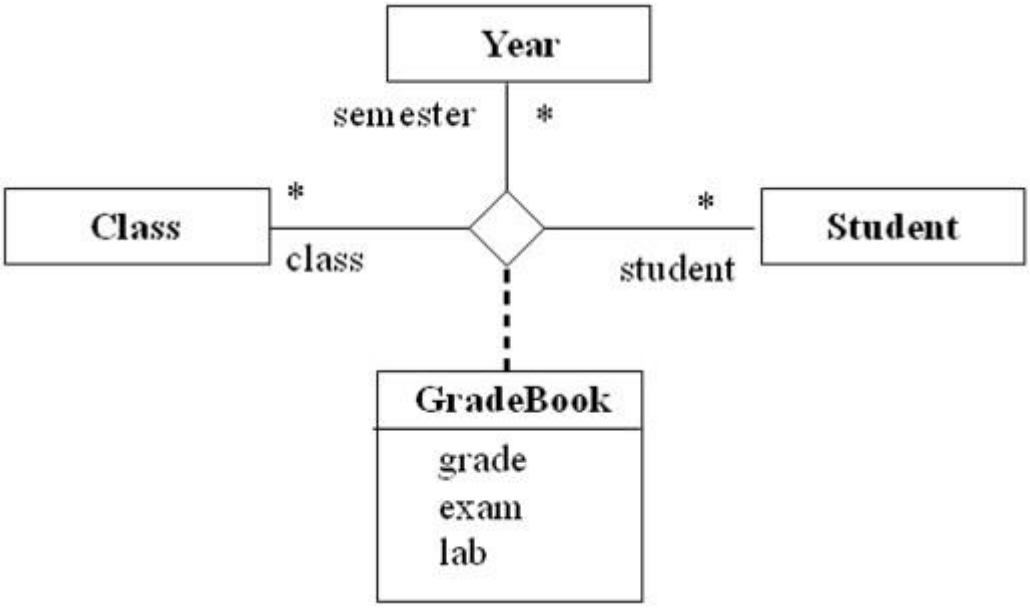


Association end query is owned by classifier QueryBuilder and association end qbuilder is owned by association Builds itself

- The "ownership" dot may be used in combination with the other graphic line-path notations for properties of associations and association ends. These include aggregation type and navigability.

N-ary associations/ Ternary Association

- N-ary association means associations among three or more classes.
- A ternary association is an association with three roles that cannot be restated as binary associations.
- The notation for a ternary association is a large diamond; each associated class connects to a vertex of the diamond with a line.

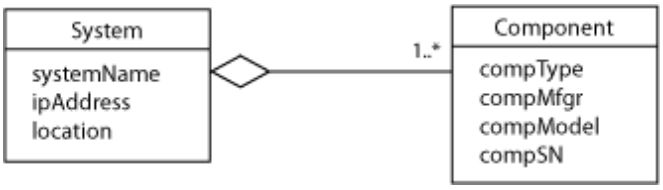


- The above figure represents another example of Ternary Association.
- You should try to avoid n-ary associations.
- Normally it is decomposed into binary associations with possible qualifiers and attributes.
- Many relationships involve just two things and can be modeled with the simple binary association.
- It is not however uncommon for three or more things to be involved in a relationship.
- An n-ary association can be used in these circumstances and allows any or "n" number of things to be related in a single cohesive group.
- An n-ary association is used when the three or more things are all related to each other in a structural or behavioral way.
- It does not replace the use of two binary associations where a classifier is related to two other classifiers, but the latter two classifiers aren't related to each other.
- In below Figure a professor teaches a listed course for a semester.
- The delivered course may use many textbooks; the same textbook may be used for multiple delivered courses.

Aggregation and Composition

Aggregation

- Aggregation is a strong form of association in which an aggregate object is made of constituent parts.
- An aggregation as relating an assembly class to one constituent part class. An assembly with many kinds of constituent parts corresponds to many aggregations.
- Aggregation is a special form of association.



- If two objects are tightly bound by a part-whole relationship, it is an aggregation.
- Aggregation is drawn like association; accept a small diamond indicates the assembly end.
- The UML has two forms of part-whole relationships.
- A general form called aggregation
- A more restrictive form called composition.

Property of an Aggregation:-

i. Transitivity: If A is part of B and B is part of C, then A is part of C. Aggregation
ii. Anti-symmetric: If A is part of B, then B is not part of A.

Composition

- Composition is a restricted form of aggregation with two additional constraints.

i. A constituent part can belong to at most one assembly.
ii. Once a constituent part has been assigned an assembly, it has a coincident lifetime with the assembly.

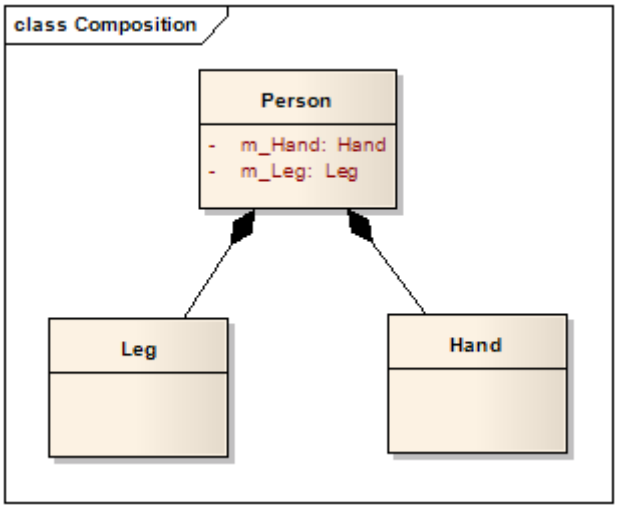
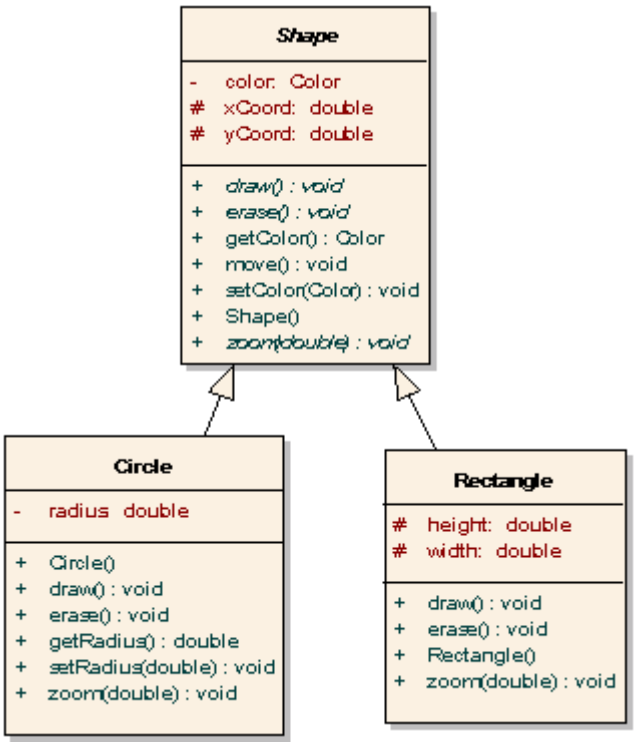


Figure of Composition

Abstract Class

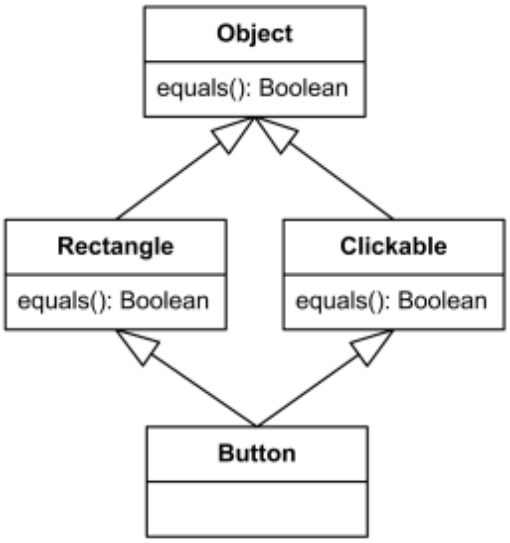
- An abstract class is a class that has no direct instances but whose descendant classes have direct instances.
- A concrete class is a class in which it can have direct instances.
- Abstraction is a process to allow focusing on most important aspects while ignoring less important details.
- In the UML notation an abstract class name is listed in an italic font. Or you may place the keyword {abstract} below or after the name.
- Use abstract class to define the signature for an operation without supplying a corresponding method.
- An abstract operation defines the signature of an operation for which each concrete subclass must provide its own implementation.
- As shown in example draw () is an abstract operation.
- Within abstract class (Graphic Object), draw () is just a definition and not implementation.

- Each subclass (Circle and Rectangle) must apply method draw () in its implementation. In other words all the super class are abstract class all the subclass are concrete class.
- It is advisable to avoid concrete super class.
- We can eliminate concrete super class by introducing other class. Differentiate Abstract class and Concrete class



Multiple Inheritance

- Multiple Inheritance permits the class to have more than one super class
- Here subclass inherit feature form its all super class.



- In the multiple inheritance diamond problem example above Button class inherits two different implementations of equals() while it has no own implementation of the operation.
- When button.equals() is called, it is unknown which implementation from Rectangle or from Clickable will be used.
- It may arise conflicts among parallel definition creates ambiguities that implementation must resolve.

Meta Data

- Metadata is data that describes other data.
- Data about data.
- All the UML software models are inherently metadata, since describe the thing being modeled.

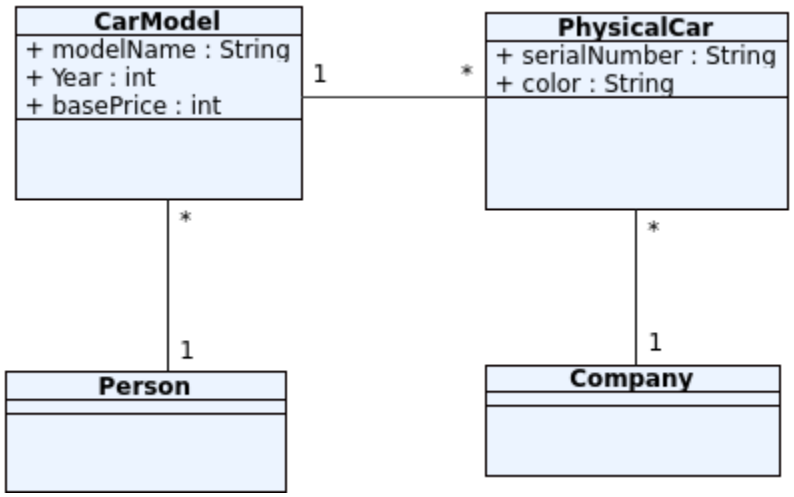


Figure: Metadata Example

- Many real world applications have metadata.
- Computer-language implementations also use metadata heavily.
- We can consider classes as an object, but classes are meta-objects and not real-world object.
- Class description objects have features, and they in turn have their own classes, which are called Meta classes.
- Treating everything as an object provides a more uniform implementation. Metadata provides greater functionality for solving complex problem.
- Accessibility of metadata varies from language to language.
- Some language access metadata at compile time and some at run time.

Constraints in class modeling.

- A constraint is a Boolean condition involving model elements, such as objects, classes, attributes, links, associations and generalization sets.
- A constraint restricts the values that elements can assume.
- We can express constraint with

i. Natural Language
ii. Formal Language such as Object Constraint Language (OCL)

- Constraints on object is helpful to add explicit constraint on the object of a class.

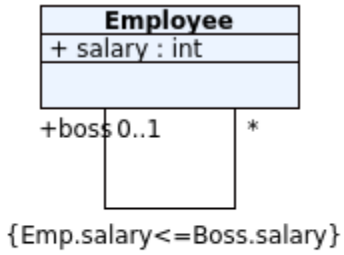


Figure: Constraint on object

- As shown in above example, represents that no employee's salary can exceed salary of employee Boss (A constraint between two things at same time).
- Another example, for maximum student in a Batch.

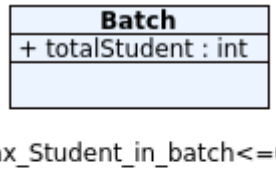


Figure: Constraint on object

Constraints on Generalization

- The semantics of generalization imply certain structural constraints.
- With the single inheritance the subclass is mutually exclusive.
- Furthermore, each instance of an abstract super class corresponds to exactly one subclass instance and each instance of a concrete super class corresponds to at most one subclass instance.
  - UML defines certain keywords to demonstrate constraint.

i. Disjoint:

- The subclasses are mutually exclusive.
- Each object of subclass belongs to exactly one of the subclasses.

ii. Overlapping:
- In an overlapping specialization, an individual of the parent class may be a member of more than one of the specialized subclasses.
- The subclasses can share some objects.
- An object may belong to more than one subclass.

iii. Complete:
- Generalization that lists all possible subclasses.

iv. Incomplete:
- Generalization in which some of the subclasses is missing.

v. Static
- Generalization in which subclass are static in nature.

vi. Dynamic
- Generalization in which subclass are dynamic with respect to time.

Constraints on Links

- Multiplicity is constraint on cardinality of set.
- Multiplicity restricts number of object related to given object.
- Qualification also adds constraint on an association.
- An association class has a constraint that an ordinary class does not have; i.e. it derives identity from instance of related classes.

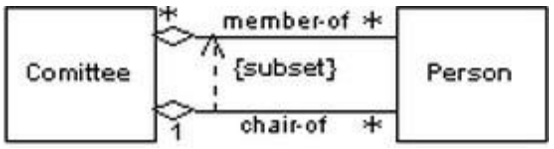


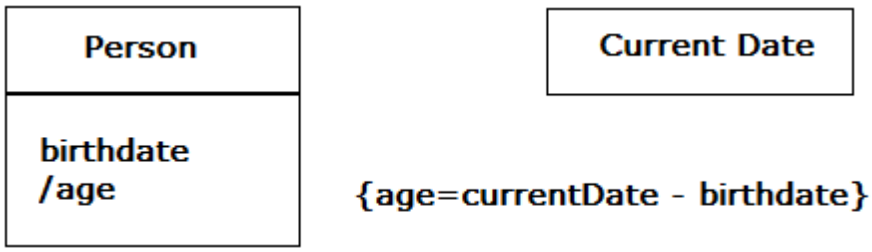
Figure: Subset constraint between association

- We favor expressing constraint in declarative manner in UML class diagram.
- If we don't use constraint in UML class modeling then we need to convert it into procedural form before the implementation; rather this is the straight forward way.
- Constraint provides criteria for measuring quality of class model.
- "A good software model" captures many constraints thought its structure.
- It often requires several iterations to get structure of a model right from the perspective of constraint.
- In practice we can't enforce every constraint, but should try to enforce the important ones.

Derived Data.

- A derived element is a function of one or more elements, which in turn may be derived.
- A derived element is redundant, because other element completely determines it.
- Classes, association and attribute may be derived.
- The notation for derived elements is a slash (/) in front of element name.

13 – Advanced Class Modeling



Packages.

- A package is a group of elements with a common theme.
- A package partitions a model, making it easier to understand and manage.
- Large applications may require several tiers of packages.
- The notation for a package is a box with a tab. The purpose of the tab is to suggest the enclosed contents.