*Composition* (also called containment) indicates that one class contains objects of another

class. In this situation, the phrase "has-a" can be used to describe the relationship between the two objects. For example, a Car "has-a" a Engine and a Circle "has-a" Center. An Engine object would simply be a data field defined in the Car class and a Center (typically a point <x,y>) would simply be a data field defined in the Circle class.

*Inheritance* "is-a" relationship denoting specialization. Inheritance is used to extend the

capabilities of another class. The extended class logically conforms to another class, but has special properties. In the situation of inheritance, the phrase "is-a" can be used to describe the relationship between the two objects. For example, a Car "is-a" Vehicle and a Officer "is-a" Employee. If the phrase "is-a" cannot logically be used to describe the relationship between the classes, then inheritance does not apply.

**Design Hints for Inheritance**

Place **common** operations and fields in the superclass.

The "is-a" relationship **must** strictly describe the relationship between the classes.

Don't use inheritance unless **all** the inherited methods in the subclass make sense.