The University of Iowa 22c22: Object-Oriented Software Development Fall 2011

Classes and Objects

Objects

An object is an entity that has

- Identity
- State
- Behavior

Object Identity

Essential feature that makes an object distinct from another

Note:

- two distinct objects may as well be identical in all other aspects
- two objects are distinct iff they have different identities

Object State

A set of attributes (properties) together with their values

- Attributes are usually static
- Attribute values are usually dynamic
- An object's state can be seen as a mapping from attributes to their values
- An object's behavior depends on its state
- The internal representation of the state is usually hidden

Object Behavior

How an object acts on other objects, reacts to other objects, and changes its state

- behavior is defined by a set of operations, or messages, the objects responds to
- an operation is a service provided by the object, possibly using services from other objects

Common Operations

Modifier

changes the object's state
Selector

accesses the state without changing it lterator

accesses parts of the object in some well defined order

Common Operations

Constructor

- creates an object and initializes its state
- Destructor
 - destroys the object and releases its resources to the system

Objects vs Classes

- attributes and behavior are defined collectively in a class, for all objects that are instances of that class
- An object's attributes and behavior are then obtained from the class it instantiates
- Only identity and attribute values are specific to each object

Objects as Machines

- Objects can be understood as little machines
- Technically, they are transition systems:
 - They have an initial state and
 - they move from one state to another in response to external messages or internal events

Objects as Machines

- Objects can be active or passive
- An active object runs independently
 - it can change its state autonomously from other objects
 - it is sometimes called an *actor*
- A passive object changes its state only when acted upon by another object

The Role of Classes/Objects in Analysis and Design

Primary tasks in analysis and early design

- I. Identify relevant classes in the problem domain
- 2. Figure out how instances of those classes can cooperate to achieve the desired functionality

This is an incremental, iterative process

Design Quality Metrics

- Coupling
- Cohesion
- Sufficiency
- Completeness
- Primiteveness

References

 G. Booch *et al.* Object-Oriented Analysis and Design with Applications, 3rd Edition. Addison-Wesley, 2007.