

Classes and Objects

Object-Oriented Analysis and Design
CSCI 6448 - Fall 1998
Kenneth M. Anderson

Overall Goals of the Course

- Develop the student's skills in the area of requirements and design
 - Construct a solid conceptual understanding of object-oriented analysis and design principles
- Familiarize the student with UML
- Create opportunities for advancing a student's teamwork skills

CSCI 6448
Kenneth M. Anderson

Overall Goal of this Lecture

- Further explore the concepts of class and object
- Examine in detail the UML notation for classes
- Describe the possible evolution of a class throughout a software lifecycle
 - i.e. we will relate it to the phases of Objectory

CSCI 6448
Kenneth M. Anderson

What is an object?

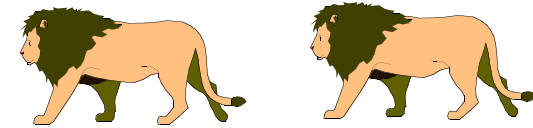
- An object has
 - Structure (attributes)
 - Behavior (operations)
- Example
 - Car
 - Color, number of wheels, horsepower
 - Accelerate, Stop, Turn, etc

CSCI 6448
Kenneth M. Anderson

Object Identity

- An object has unique identity
 - Each object exists independently
- Each object should have a name
 - Name not need be unique but it should be reasonable and concise
- Example
 - Ken's Car: (Make:Subaru, Year:1997)

Further examples of Object Identity



Name	Leo	Leo
Age	7	15



Dot



Dot



Dot



Dot

What is a Class?

- A group of objects with similar structure and behavior
- A class defines the information shared by all objects of its class
 - Each object may have different values for the set of attributes but they all have the same set of attributes

Relation to last week

- Class (or Concept)
 - Intension (A concept's definition + "test")
 - Extension (All objects which pass the test)
- Object
 - An entity in the world
 - its membership in different classes can evolve over its lifetime

Class Notation (UML)

Class Name

The UML symbol is a rectangle containing the name of a class

Example:

Compact Disc

Object Notation

object Name: Class Name

The UML Symbol for an object is a rectangle containing the object name and a class name separated by a colon. The underline signifies an instance.

Example:

trumpet: Brass Instrument

Naming Objects

Objects may have proper names or generic names

Ken: Human

aPerson: Human

Objects may be of unknown class or may be anonymous

Flute

: Wind Instrument

Instance Notation

Car ← Ken's Car : Car

Ken's Car is an instance of Car

Attribute Notation

Class Name
Attribute 1
Attribute 2

Car
Make
Model

<u>Object Name : Class Name</u>
Attribute 1 = value
Attribute 2: Type 2 = default value

<u>Ken's Car : Car</u>
Make = Subaru
Model: Car Line = Legacy

Derived Attributes

- A derived attribute is one whose value can be computed from the values of other attributes
- Example
 Birthday: 08/25/1968
 /Age: 30 years
- Derived attributes have a '/' in front of their name

Operations

- An operation is an action performed by or on an object
- A class defines all the operations common to its objects
- Operations have names, can take typed parameters, and can return a typed value

Operation Notation

Class Name
Attribute 1
Attribute 2
Operation 1
Operation 2
Operation 3

<u>Ken's Car: Car</u>
Make = Subaru
Model = Legacy
Accelerate()
Stop()
SetCruiseContol(Speed)

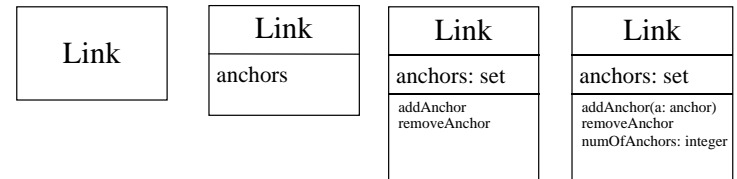
More information on Operations

- **Format**
operation(param:type = value,...): return type
- **Example**
accelerate(topSpeed: integer = 60): status
- A name, parameters and return type constitute an operation's signature
- If an operation needs to return more than one value, return a class

CSCI 6448
Kenneth M. Anderson

Evolution of a Class

Elaboration \longrightarrow Construction



CSCI 6448
Kenneth M. Anderson

Acknowledgments

- Some information for this lecture has been drawn from material produced by the Advanced Concepts Center of Lockheed Martin Corp.

CSCI 6448
Kenneth M. Anderson