Object-Oriented Design

IT4GIS
Keith T. Weber, GISP
GIS Director
ISU-GIS Training and Research Center

Topics
• During the balance of this semester, we will pursue and follow two learning threads
  – Object-relational databases
  – The Geo-Web
• These two threads are interwoven

To understand Object-Relational Databases...

• We need to understand both relational concepts and
• Object-oriented concepts (this week)
The Early Days...

- Computer programming from the caveman era

Why...Object-Oriented

- A brief history of computer programming...
- Early waterfall programming

The concept of encapsulation!

Today's Goals

- We will dissect "Object-Oriented" design and learn what it really is and how it relates to object-relational databases.
- At its core, we need to know
  - What is a CLASS
  - What is an OBJECT
What is a CLASS?
• A class is a computer construct representing a concept bound in a cohesive package
  – Some are concrete (i.e., real world)
    • Bank account
    • Rental item
    • Database item
    • Pile
  – Others are abstract
    • Scanner
    • Stream
    • Math

Discovering CLASSES
• Simple Rule:
  – Look for nouns in descriptions
  – Obviously not all nouns are classes
  – But at least this approach can allow one to create a list of candidate classes

What is an OBJECT
• An instance of a CLASS
• That contains meaningful data
• OBJECTS occupy memory space at runtime
  – If not, they are CLASSES
  – For example: data type vs. double
A Little Quiz...

• #1 Class or Object?
  
  Dog
  
  Scooby-Doo

  Dog is a generalization of Scooby-Doo

A Little Quiz (cont’d)...

• #2 Class or Object?
  
  The concept of subclass!
  
  Dog
  
  Animal
  
  Scooby-Doo

  Dog is a subclass of the Animal class
  
  Animal is a generalization of Dog

A Little Quiz (cont’d)...

• #3 Class or Object?
  
  The concept of polymorphism!
  
  Animal
  
  Bird
  
  Dog
Questions so far...

Key Points

• Many classes already exist and are at our disposal when we design a database
• We use inheritance to add capabilities
  – A subclass inherits from its superclass
  – i.e., a child inherits from its parent

Additional Key Concepts

• Identify behaviors and relationships between classes
  – Inheritance
  – Aggregation
  – Dependency
• Understand class attributes and object properties

Note: a Class attribute is not to be confused with the attributes/fields of a table.
Inheritance in the Geodatabase

Defining a CLASS

• After a CLASS has been identified we can Define:
  – Its behavior (i.e., Methods (verbs))
  – And its attributes
Relationships Between CLASSES

• A key behavior is the relationship or interaction between classes
• We have learned about inheritance as one (1) type of relationship that may exist between classes
• There are three (3) important relationship types
  Inheritance
  Aggregation
  Dependency

1- Inheritance

• Is-a relationship
• Relationship between a more specialized class (subclass) and a generalized class (superclass)
• Every...
  – Savings account is a bank account
  – DVD rental is a rental
  – Dog is a mammal
  – Parcel is a polygon

2- Aggregation

• Has-a relationship
  – Each Dog has a Paw (dog is not a generalization of paw!)
• One class (Appendages) contains references to another class (Dog)
• Each line has a vertex (actually, at least two vertices)
Inheritance vs. Aggregation

- Sometimes confused
- Questions?

Example

- Car is a Vehicle – Inheritance
- Car has a set of Tires – Aggregation

Figure 8. UML Notations for Inheritance and Aggregation

3- Dependency

- Dependency occurs when a class uses or relies on another class
- This is a Uses relationship
  - Example: an application may depend on the Scanner class to read input
  - ArcGIS depends upon the Mouse class to enable user interaction
What type of Method behaviors are these?

This is an Example of Dependency

- The Deposit CLASS interacts with the Bank Account class to modify the Balance attribute.
Attributes help define a given class and instantiate it into an object.

Brainstorm....the Dog Class

- Name of the class = Dog
- Methods:
  - Example of inheritance
  - Example of aggregation
  - Example of dependence
- Attributes?

Instantiate into an Object

- Three features characterize objects:
  - Identity: specific attribute (property) settings have been made for the class. This distinguishes it from all other objects.
  - State: Describes the data stored in the object. WHERE DID THIS COME FROM?
  - Behavior: describes the method in the object’s interface through which the object can be used (how do we make the dog bark?)
Instantiating the Dog CLASS

• CLASS (DOG)
• Attributes (Properties)
  – NAME = Scooby-Doo
  – HEIGHT = 36
  – WEIGHT = 145
• Methods
  – [Uses] bark- “Rooby roo”
  – etc.

Key Concepts

• Understand the difference between a CLASS and an OBJECT
• Understand new terms:
  – Encapsulation, polymorphism, superclass, subclass, behavior, attributes, instantiation
• Understand --and be able to differentiate-- the three types of behaviors

Questions...