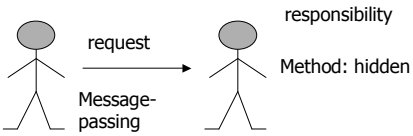
Chapter 1: OO Thinking

- Not just learn Java, but also *understand why* it is the way it is
- Learn the OO worldview

A way of viewing the world

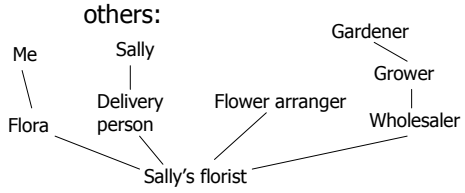
- Want to send flowers to another town
- How: ask your local florist



"don't even want to know the details (delegation)"

Agents and communities

- OO program = community of interacting agents called objects
- Each object provides a service, used by others:



Messages and methods

- Action: initiated by a message to an object responsible for the action
- Arguments: additional necessary information for the request
- Receiver: accepts message, performs some method to satisfy the request

More on methods

- Information hiding, be lazy, let someone else perform the task, reuse, use others' services.
- Method <> Function
  - Designated receiver
  - Different interpretation depending on the specific receiver
  - Receiver only known at runtime (late binding)

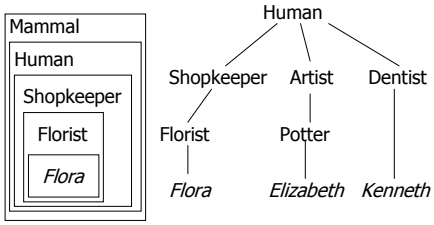
Responsibilities

- "ask not what you can do to your data structures, but what your data structures can do for you."
- Act on a data structure vs.
- The data structures act on themselves
- Independence: more abstract
- Protocol: collection of responsibilities

Classes and instances

- All objects are instances of classes.
- Receiver's class determines method for a given message.
- All objects/instances of a given class use the same method servicing similar messages.

Class hierarchies - Inheritance



Classes

- Are organized as a hierarchical inheritance structure
- Child class inherits from parent classes higher in the tree
- Abstract parent classes (e.g. Mammal) don't have direct instances

Search for method

- Starts at receiver, then goes up the tree
- Child may define method with same name and arguments: overriding
- Different methods for same message: polymorphism
- Java: can determine at compile-time that there will be some appropriate method, but may not know which

OO concept summary

1. Everything is an object.
2. Computation: sending & receiving messages (requests + arguments)
3. Each Object has its own memory (other objects)
4. Every object is instance of some class.
5. Class is a repository for behaviors.
6. Classes form a singly-rooted tree.

Computation is simulation

- "Instead of a bit-grinding processor … plundering data structures, we have a universe of well-behaved objects that courteously ask each other to carry out their various desires. [Ingalls 1981]
- Powerful metaphor