Frames and Inheritance
—
Winston, Chapter 9

Michael Eisenberg and Gerhard Fischer
TA: Ann Eisenberg

AI Course, Fall 1997

Programming Languages — A Knowledge Representation View

procedural:
- examples: Fortran, Pascal, Lisp, C
- main features: subroutines, recursion
- AI contribution: dynamic memory structures (beyond arrays)

rule-based:
- examples: OPS-5, expert system shells, embedded rule interpreters
- main features: simplicity, uniformity, modelling of expert knowledge
- AI contribution: production systems (protocol analysis), modelling of expert knowledge

object-oriented:
- examples: Simula, Smalltalk, C++, Clos
- main features: class/instance, single/multiple inheritance
- AI contribution: frames, modelling of common sense knowledge

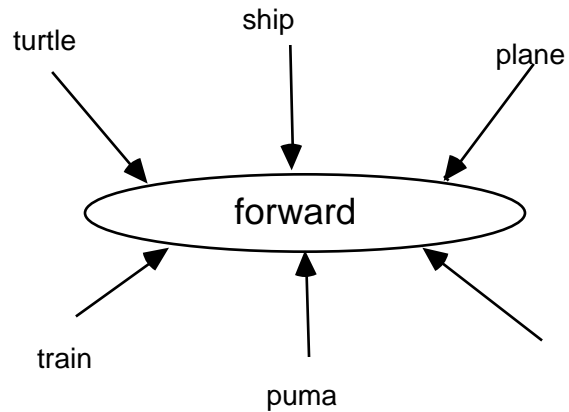# An Example: One Turtle ---> Multiple Turtles ---> Multiple Objects

- one (existing) turtle: {<object>
  <message>, <verb><noun>}
       forward 10

- multiple turtles:
       option 1:   forward t_1 10
       option 2:   talk to t_1
             forward 10
             right 90
          talk to t_2

- multiple kind of objects
       forward enterprise 10

- define forward (object, amount)
      select (object)
         object = turtle   ---> forward-turtle amount
         object = ship   ---> forward-ship   amount
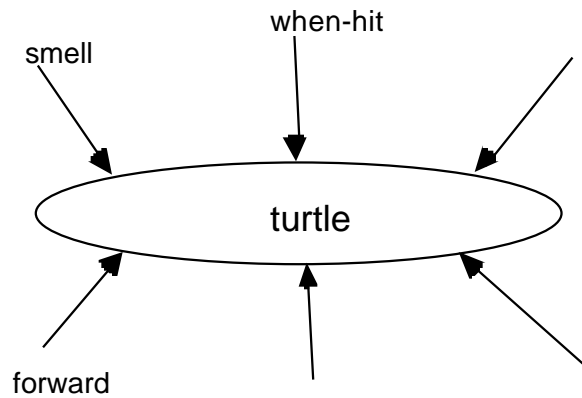
Turtles — Object-Oriented Design (OOD)

- t1 forward 10   {<object>
  <message>, <noun><verb>}

- "turtles" and "ship" can be subclasses from "moving
  objects"

- Object-Oriented design methodology:

    - what objects do exist (which have a state
      associated with it)

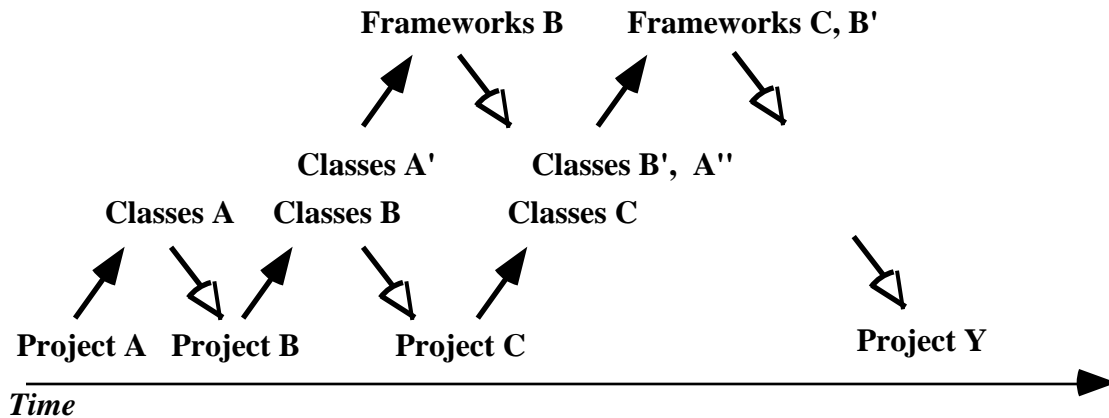   - how can these objects be grouped into classes

Factorization of Knowledge

• in procedural environments

turtle   ship   plane

forward

train   puma

• in object-oriented environments

smell   when-hit

turtle

forward

An Evolutionary of Model Object-Oriented Development

Frameworks B   **Frameworks C, B'**

Classes A'   **Classes B', A''**

Classes A   **Classes B**   **Classes C**

Project A   Project B   Project C   **Project Y**

Time

- *solid-tip arrows:* evolution, driven by software developers creating new software objects to accommodate new projects

- *hollow-tip arrows:* indicate software developers reusing components, although reuse at times leads to redesign

- stable structures of class libraries and frameworks emerge over multiple projects

- domain-orientation permeates the model as all of the creation, reuse, and redesign of components is driven by problems arising in the development of specific projects

Evolving Components in Object-Oriented Development
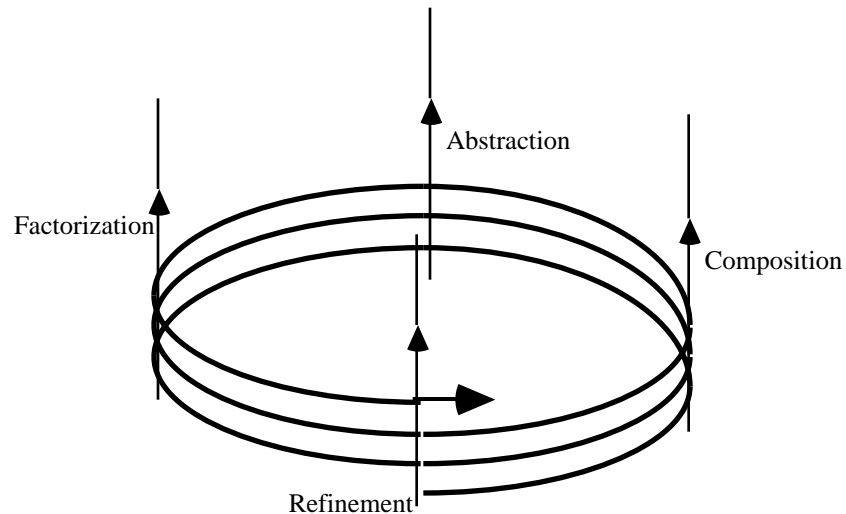
- **refinement**: subclassing

- **composition**: combine classes

- **abstraction**: from commonalties in behaviors and properties

- **factorization**: partition properties and methods of an object into simpler, more cohesive and presumably more reusable components

Claims of Object-Oriented Development Technologies

• OO systems support software evolution, extension, and/or modification

• OO systems are reusable and easily constructed from existing components

• OO systems are understandable to domain experts (users) and developers

AI Course, Fall97

Beyond Current Object-Oriented Design

- focus on the needs of the *human* software developers (Fischer 1987) more than on the formal properties of classes, instances, and inheritance (Stefik & Bobrow 1986).

- OO technology presents not simply a new way to program, but a new way for people to think about problems and problems domains.

- OOD is necessary/advantageous for many problems, but not sufficient

Why is OOD Not Sufficient

- good abstractions are not given, but need to be created

- insufficient support for reuse and redesign (location, comprehension and modification)

- lack of domain-orientation

- claim: no single methodology and/or tool will be suited to all problems

Example: General Knowledge about Fairy-Tails Dwarfs

- fairy-tail competitors and gourmands are fairy-tale dwarfs

- most fairy-tale dwarfs are fat

- most fairy-tale dwarfs' appetites are small

- most fairy-tale gourmands' are huge

- most fairy-tale competitors are thin

Terminology: From Semantic Nets to Frames

- frame   =   nodes and links grouped together

- slot   =   link

- slot values   =   destination of links

- instances frames or instances: describe individual things (e.g., Grumpy, an individual dwarf)

- class frames of classes: describe groups of objects (e.g., all dwarfs)

- "is-a-member-of-the-class" slot: ties instances to the classes they are members of

- "a-kind-of" slot: ties classes together via sub- and superclass relationships

- direct subclass: the ako slot is filled with then name of the superclass

Shared Knowledge— Located Centrally

- Easier to construct when you write it down

- Easier to correct when you make a mistake

- Easier to keep up to date as times change

- Easier to distribute because it can be distributed automatically

Inheritance

- To fill the Physique slot when a new Dwarf is constructed ---> write "Fat" in the slot.

- To fill the Physique slot when a new Competitor is constructed ---> write "Thin" in the slot.

- the **class precedence list** (see Figure 9.2):
    - Blimpy

    - Managers class

    - Competitors class   ←   procedure stored here

    - Dwarfs class   ←   procedure stored here

    - Everything class

- question: is Blimpy fat or thin?

Inheritance with Multiple Superclasses

- To fill the Appetite slot when a new Dwarf is constructed - --> write Small in the slot.

- To fill the Appetite slot when a new Gourmand is constructed - --> write Huge in the slot.

- Search:
  - depth-first search
  - exhaustive depth-first search
  - up-to-join proviso

Example

- Blimpy
- Managers class
- Competitors class
- Dwarfs class   ⟵ procedure stored here
- Everything class
- Gourmands class   ⟵ procedure stored here
- Diarists class

- **rule:** each class should appear on class-precedence list before any of its superclasses (up-to-join proviso)

- Blimpy
- Managers class
- Competitors class
- Gourmands class   ⟵ procedure stored here
- Diarists class
- Dwarfs class   ⟵ procedure stored here
- Everything class

Demon Procedures

- reading or writing can activate:
    - when-requested procedures
    - when-read procedures
    - when-written procedures

- these procedures are called **demons**, because they lurk about doing something unless they see the operations they were designed to look for (compare to agents and critics)

- **when-requested procedures ---> overwrite slot values; examples:**
    - when a value for the Hobby slot of an Athlete is requested ----> return exercise
    - when a value for the Hobby slot of an Dwarf is requested ---->
        - * if the dwarf's Personality slot is filled with Shy, return reading
        - * otherwise, return dancing

More Demons

when-read procedures and when-written procedures ---->
maintain consistency
- when a value is written in the Physique slot of an
Athlete ----> if the new value is Muscular, write Large in
the Athlete's Appetite slot

with-respect-to procedures deal with perspectives
- when a value for the size slot of Blimpy, from the
perspective of a typical <u>dwarf</u>, is requested ---->
Return <u>Big</u>
- when a value for the size slot of Blimpy, from the
perspective of a typical <u>person,</u> is requested ---->
Return <u>Small</u>

with-respect-to procedures deal with contexts
- when a value for the Mood slot of Patrick, in the
context of <u>Mountain Hiking</u>, is requested ----> Return
<u>Happy</u>
- when a value for the Mood slot of Patrick, in the
context of <u>Airplane Travel</u>, is requested ----> Return
<u>Grumpy</u>

Frame Systems

A frame system is a representation that is a semantic net in which

- The language of nodes and links is replaced by the language of frames and slots.
- Ako slots define a hierarchy of class frames.
- Is-a slots determine to which classes an instance frame belongs.
- Various when-constructed, when-requested, when-read, when-written, and with-respect-to procedures supply default values, override slot values, and maintain constraints.
- A precedence procedure selects appropriate when-constructed, when-requested, when-read, when-written, and with-respect-to procedures by reference to the class hierarchy.

With constructors that

- Construct a class frame, given a list of superclasses, and a list of slots
- Construct an instance frame, given a list of direct super-classes
- Construct a when-requested, when-read, when-written, or with-respect-to procedure

With writers that

- Establish a slot's value, given an instance, a slot, and a value

With readers that

- Produce a slot's value, given an instance and a slot

When-Applied Procedure

procedures helping to perform an action in a manner suited to the object acted on

example:
- To eat when Soup is to be eaten ----> use a big spoon.

- To eat when Salad is to be eaten ----> use a small fork.

- To eat when the Entree is to be eaten ----> use a big fork and a big knife.

overwrite inheritance:
- To eat when the Entree is Lobster ----> use a tiny fork and a nutcracker

Example: Digesting News — Frame Retrieval and Slot Filling

- To fill the Time slot when a new Event is constructed, find a number with a colon in it and write it in the slot.

- To fill the Fatalities slot when a new Disaster is constructed, find an integer near a word with a root such as kill or die, and write it in the slot.

- To fill the Damage slot when a new Disaster is constructed, find a number next to a dollar sign, and write it in the slot.

- To fill the Magnitude slot when a new Earthquake is constructed, find a decimal number between 1.0 and 10.0, and write it in the slot.

- Other simple procedures can fill in nonnumeric slots:
    - To fill the Day slot when a new Event is constructed, find a word such as today, yesterday, tomorrow, or the name of one of the days of the week, and write it in the slot.
    - To fill the Place slot when a new Event is constructed, find a name that appears in a dictionary of geographical places and write that name in the slot.
    - To fill the Fault slot when a new Earthquake is constructed, find a proper name near the word fault and write it in the slot.

Analyzing Stories with Title Evoking Earthquake Frame

Earthquake Hits Lower Slabovia

Today, an extremely serious earthquake of magnitude 8.5 hit Lower Slabovia, killing 25 people and causing $500 million in damage. The President of Lower Slabovia said that the hard-hit area near the Sadie Hawkins fault has been a danger zone for years.

Earthquake Summary Pattern

An earthquake occurred in <value in Location slot> <value in Day slot>. There were <value in Fatalities slot> fatalities and $<value in Damage slot> in property damage. The magnitude was <value in Magnitude slot> on the Richter scale; the fault involved was the <value in Fault slot>.

Instantiated Earthquake Summary Pattern

An earthquake occurred in Lower Slabovia today. There were 25 fatalities and $500 million in property damage. The magnitude was 8.5 on the Richter scale; the fault involved was the Sadie Hawkins.

Earthquake Study Stopped

Today, the President of Lower Slabovia killed 25 proposals totaling $500 million for research in earthquake prediction. Our Lower Slabovian correspondent calculates that 8.5 research proposals are rejected for every one approved. There are rumors that the President's science advisor, Sadie Hawkins, is at fault.

SUMMARY

- A frame system can be viewed as a generalized semantic net. When you speak about frames, however, your language stresses instances or classes, rather than nodes, and stresses slots and slot values, rather than links and link destinations.

- Inheritance moves default slot values from classes to instances through the activation of the appropriate when-constructed procedure.

- To determine which when-constructed procedure dominates all other applicable when-constructed procedures, you have to convert a class hierarchy into a class-precedence list. Generally, the conversion should be such that each class appears before all that class's superclasses and each class's direct superclasses appear in order.

- When-requested procedures override slot values. When-read and when-written procedures maintain constraints. With-respect-to procedures deal with perspectives and contexts.

- Digesting news seems to involve inheritance. Your understanding of an earthquake news story, for example, benefits from your knowledge of the connection between earthquakes and disasters and your knowledge of the connection between disasters and events in general.

- Shared knowledge, located centrally, is easier to construct when you write it down, easier to correct when you make a mistake, easier to keep up to date as times change, and easier to distribute because it can be distributed automatically.