Programming Paradigms

Data Abstraction and Object-Orientation (Part 2)

Prof. Dr. Michael Pradel

Software Lab, University of Stuttgart

Summer 2022

Overview

- Inheritance
- Initialization and Finalization
- Dynamic Method Binding
- Mix-in and Multiple Inheritance ←

Motivation

- **Designing an inheritance tree with exactly one parent class: Difficult in practice**
- **Examples**
  - `Cat` may be both `Animal` and `Pet`
  - `Widget` in database maybe `Sortable`, `Graphable`, and `Storable`

Mix-in Inheritance

- **Mix-in: Class-like abstraction that provides methods to be used in other classes without inheriting from the mix-in**
- **A class can mix in multiple mix-ins**
- **Example:**
  - Combine mix-ins `Animal` and `Pet` into `Cat`

Support in Popular PLs

Many variants of the mix-in idea

- Java: Interfaces are a lightweight version of mix-ins
  - Since Java 8: Default implementations of interface methods
- Scala: “Traits”
- Ruby: `include` modules into a class

True Multiple Inheritance

- Sometimes, want to expand multiple classes
- Allowed in some PLs, e.g., C++, Python, OCaml
- Example (C++):

```
class student : public person, public system_user {
  ...
}
```

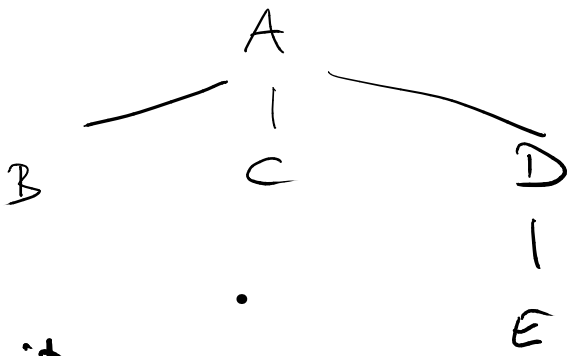
True Multiple Inheritance

- Sometimes, want to expand multiple classes
- Allowed in some PLs, e.g., C++, Python, OCaml
- Example (C++):

```
class student : public person, public system_user {
  ...
}
```

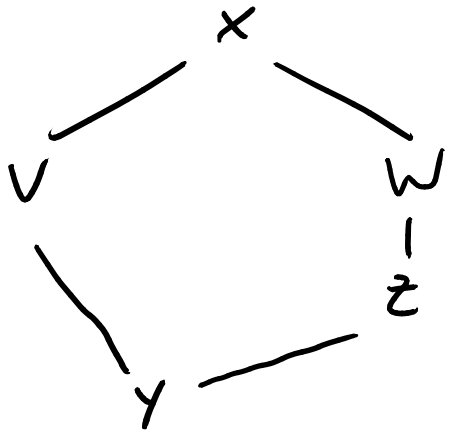
Gets all fields and methods of both superclasses

- Single inheritance: Class hierarchy is a tree



- Multiple inheritance:

  Directed, acyclic graph (DAG)



→ "diamond problem":

multiple paths to same superclass

(here: $Y \to \dots \to X$)

Semantic Issues

- What if two parent classes provide a method with the same name?
- What if two parent classes are both derived from a common "grandparent"? Does the "grandchild" have one or two copies of the grandparent's fields?
- How to represent objects in memory? Can each parent be a prefix of the child?

Semantic Issues

- What if two parent classes provide a method with the same name?
- What if two parent classes are both derived from a common "grandparent"? Does the "grandchild" have one or two copies of the grandparent's fields?
- How to represent objects in memory? Can each parent be a prefix of the child?

Answers depend on the PL and go beyond this lecture

Quiz: Data Abstraction

Which of the following is true?

- Java makes it impossible to violate Liskov's substitutability principle.
- Static and dynamic method binding are the same in statically typed PLs.
- vtables store the fields of an object.
- In C++, a class can directly extend multiple classes.

Quiz: Data Abstraction

Which of the following is true?

- ~~Java makes it impossible to violate Liskov's substitutability principle.~~
- ~~Static and dynamic method binding are the same in statically typed PLs.~~
- ~~vtables store the fields of an object.~~
- In C++, a class can directly extend multiple classes.