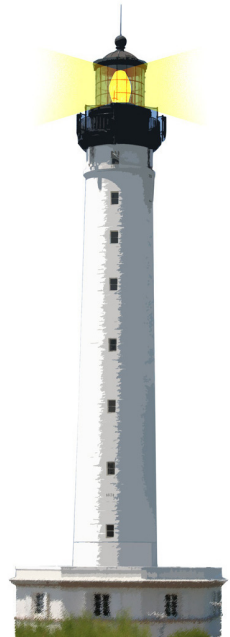
Inheritance Basics

Damien Cassou, Stéphane Ducasse and Luc Fabresse

W4S01

Pharo

http://www.pharo.org



Goal

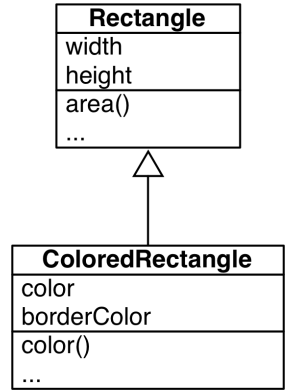
- What is inheritance?
- When to use it?

Pharo treats inheritance the same way as Java



The Basics

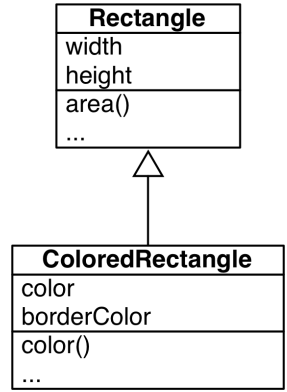
- Often we want small adaptations
- we want to extend existing behavior and state
- solution: **class inheritance**
- a class extends the definition of its superclass



The Basics

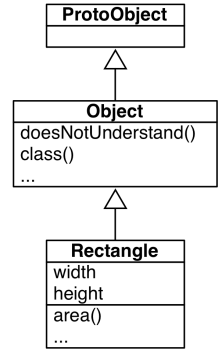
a subclass

- can add state and behavior:
  - color, borderColor, ...
- can use superclass behavior and state
- can specialize and redefine superclass behavior



Root of Inheritance Hierarchy

- Object is the root of most classes
- ProtoObject (Object's superclass) is for special purposes...
  - ...but we will ignore it as it is not important



The Basics

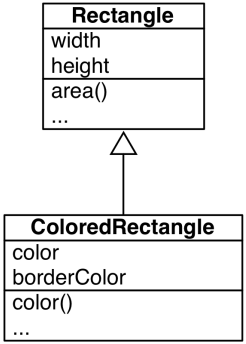
inheritance is

- static for state (i.e., during class creation)
- dynamic for behavior (i.e., during execution)



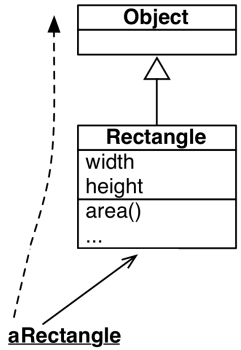
Inheritance of Instance Variables

- happens during class definition
- computed from
  - the class own instance variables
  - the ones of its superclasses



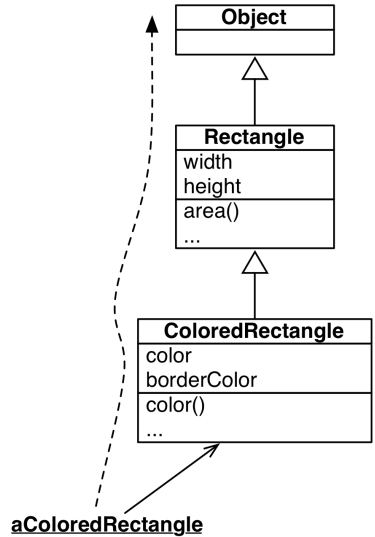
Inheritance of Behavior

- happens at runtime
- the method is searched
  - starting from the receiver's class
  - then going to the superclass



Inheritance of Behavior

- happens at runtime
- the method is searched
  - starting from the receiver's class
  - then going to the superclass



What You Should Know

- Inheritance allows a class to refine state and behavior
- A class has 1 and only 1 superclass
- A class eventually inherits from Object
- Inheritance of state is static
- Inheritance of behavior is dynamic



A course by



and



in collaboration with











Inria 2020

Except where otherwise noted, this work is licensed under CC BY-NC-ND 3.0 France
https://creativecommons.org/licenses/by-nc-nd/3.0/fr/