

CHAPTER 8

Class Relationships

**Data Abstraction and Problem Solving with JAVA:
Walls and Mirrors
Carrano / Prichard**

Figure 8.1

Inheritance: Relationships among timepieces

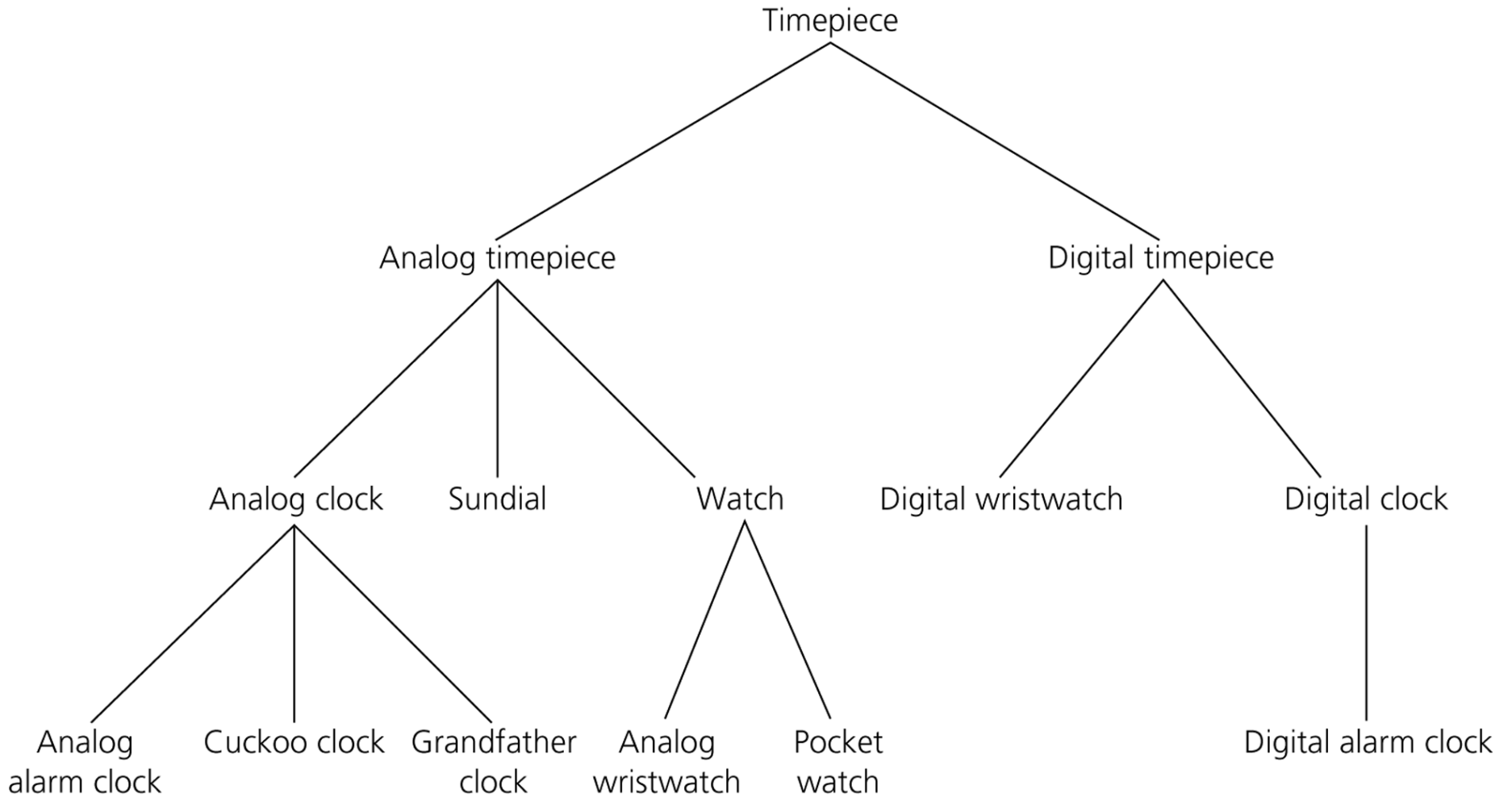


Figure 8.2

The subclass *Ball* inherits members of the superclass *Sphere* and overrides and adds methods

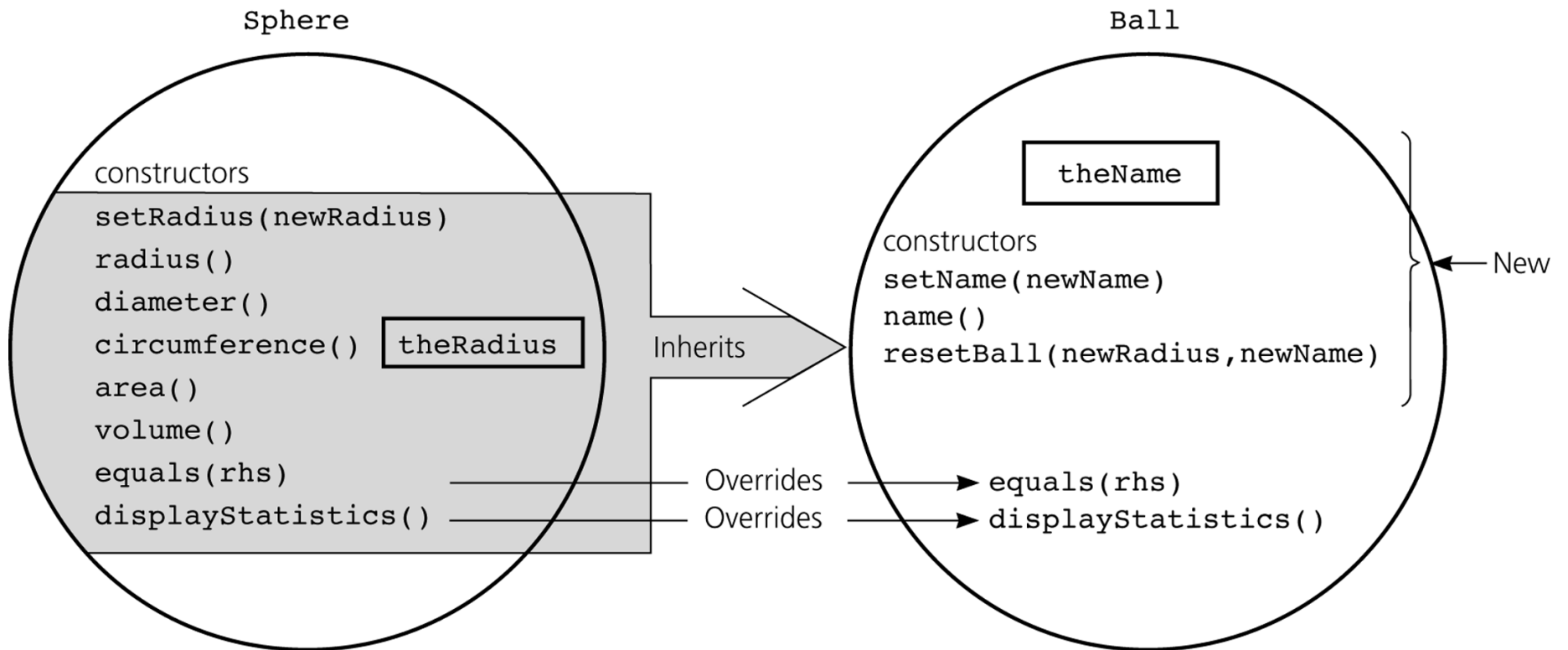


Figure 8.3

An object invokes the correct version of a method

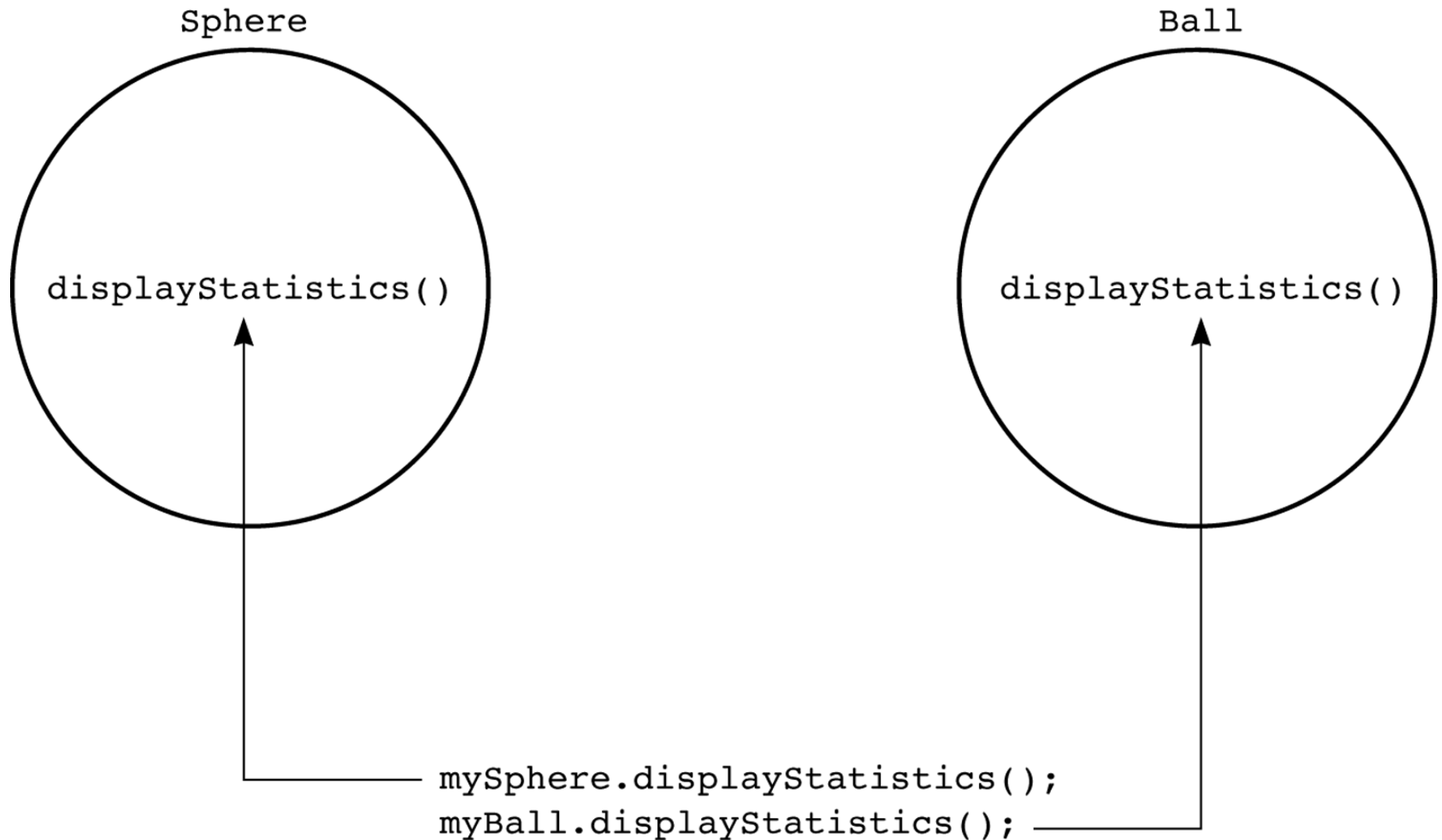


Figure 8.4

Access to public, protected, package access, and private members of a class by a client and a subclass

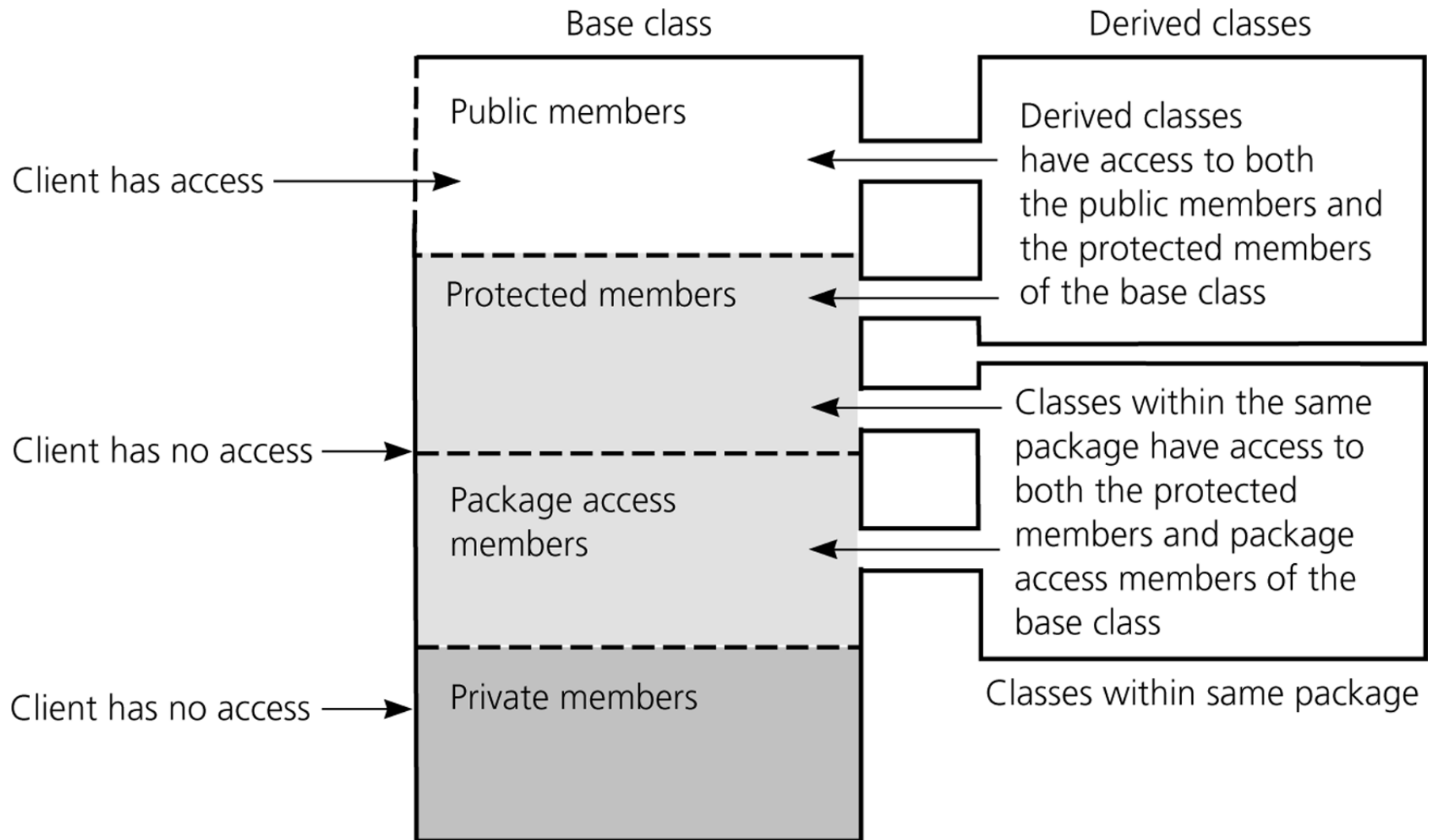


Figure 8.5

A ball “is a” sphere

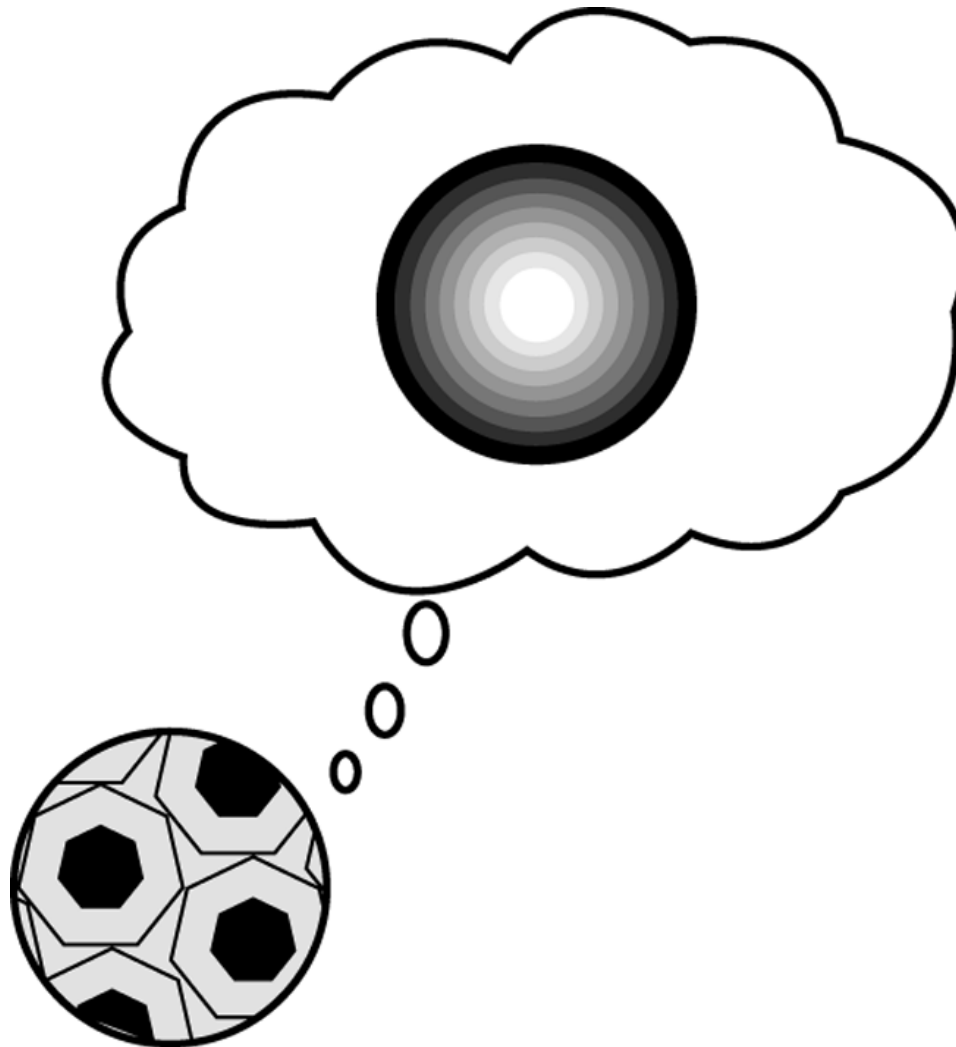


Figure 8.6

A pen “has a” or “contains a” ball

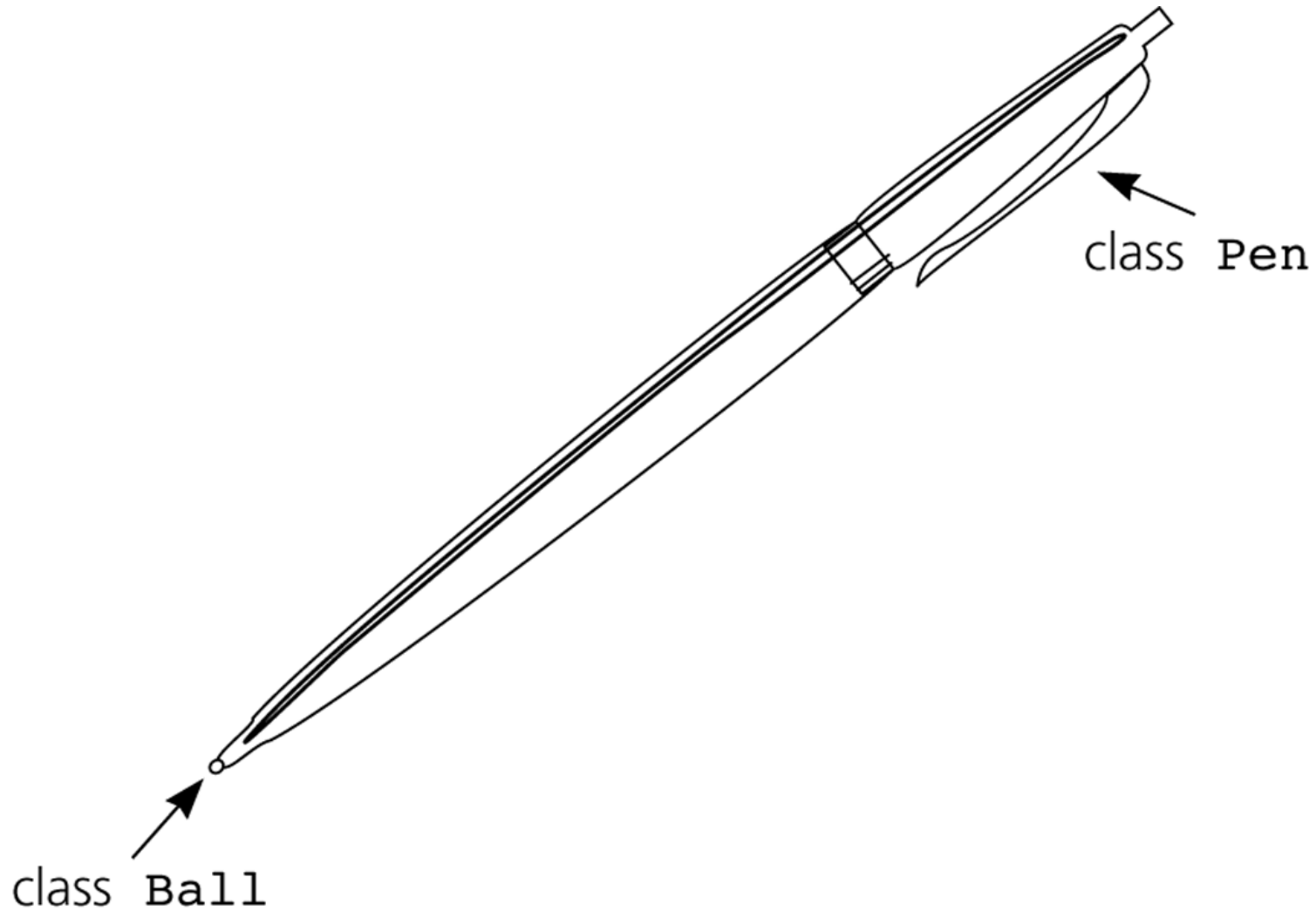


Figure 8.7a

area is overridden: a) *mySphere.DisplayStatistics()* calls *area* in *Sphere*

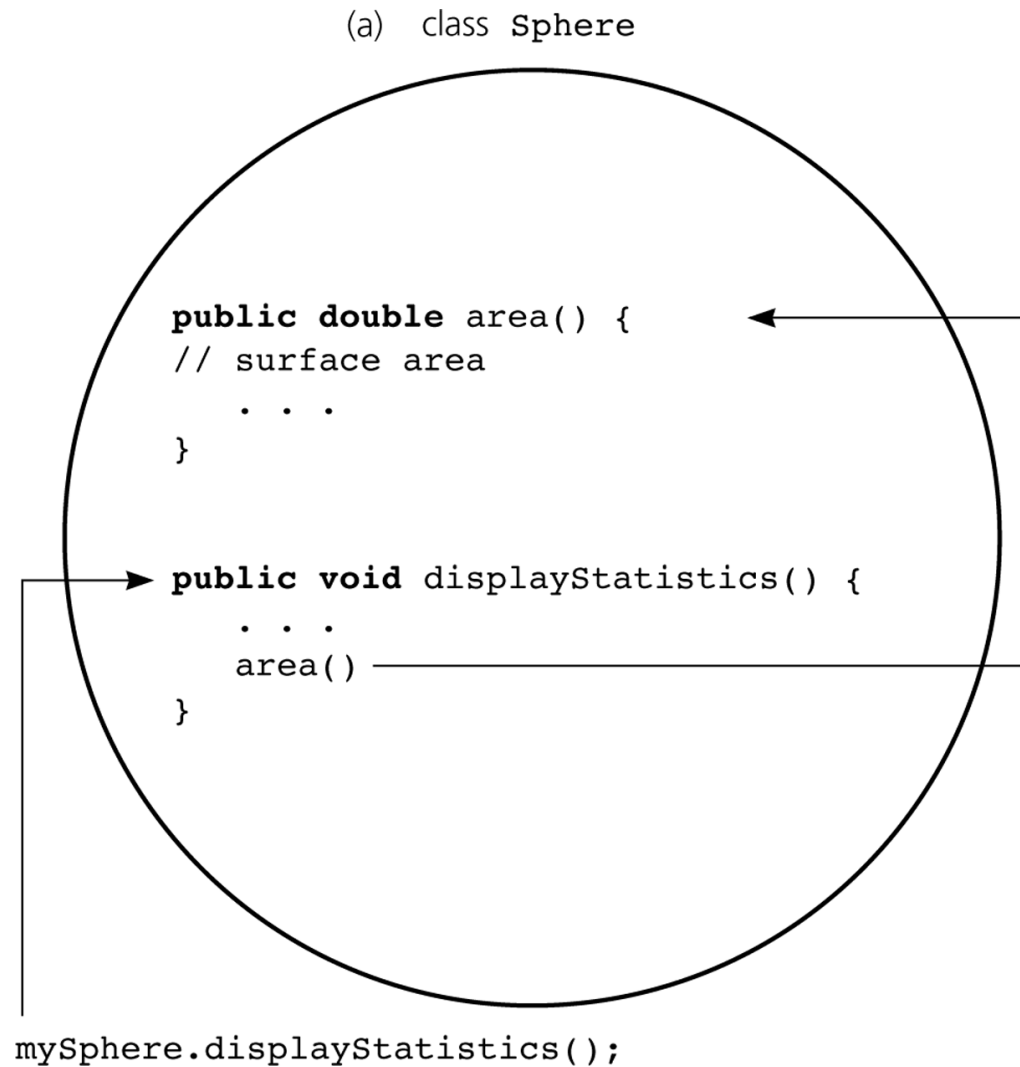


Figure 8.7b

b) `myBall.displayStatistics()` calls `area` in `Ball`

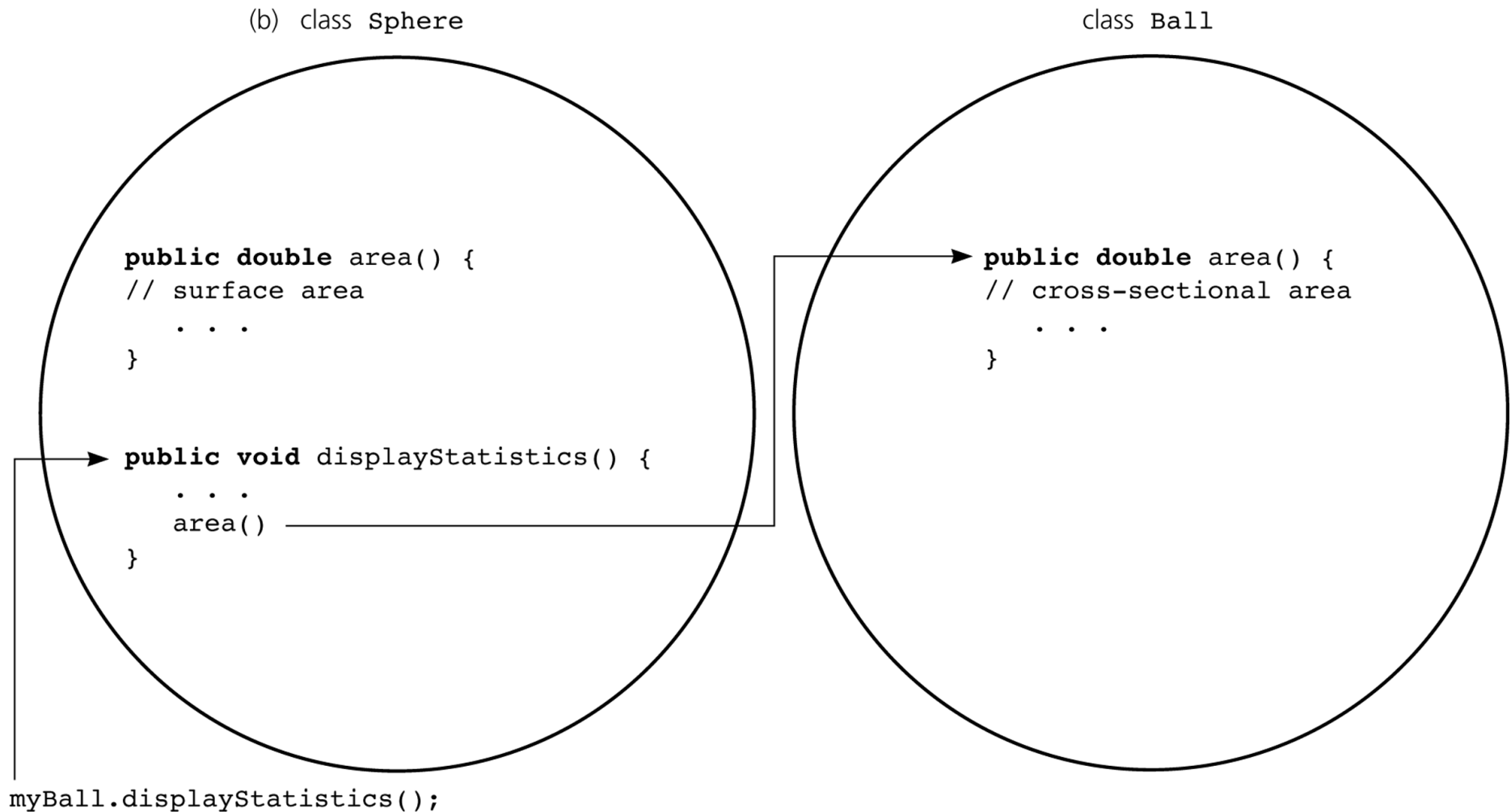


Figure 8.8

VCR is a subclass of ACR

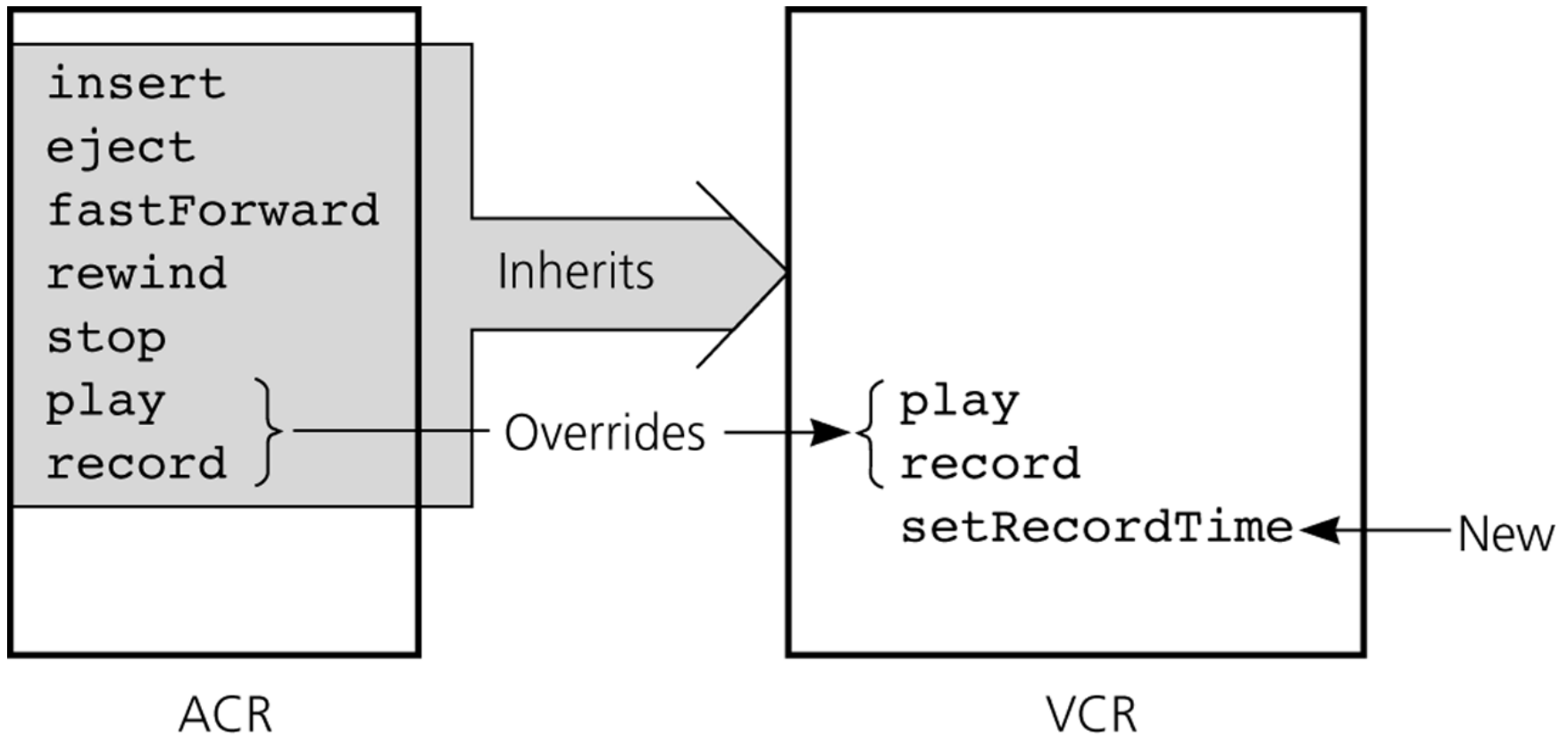


Figure 8.9

ACR has an abstract class (tape transport) and a derived class (VCR)

