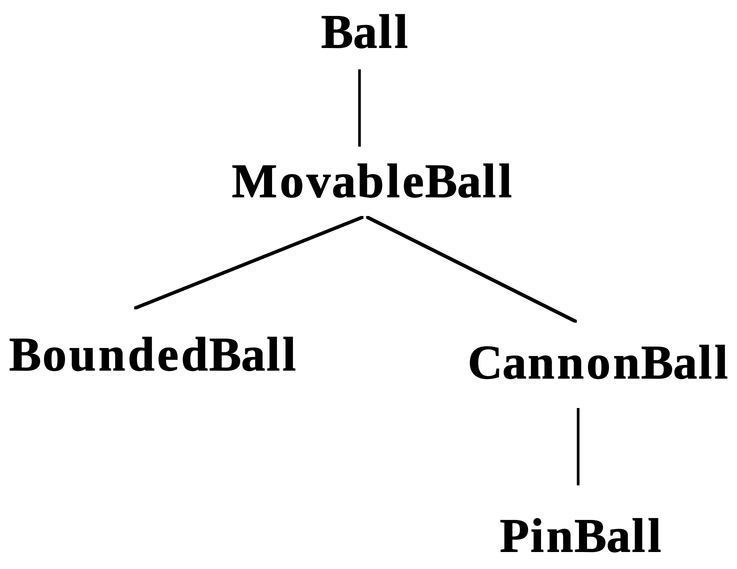
Using Patterns to Help Students See the Power of Polymorphism

Supplement: Using the Decorator Pattern

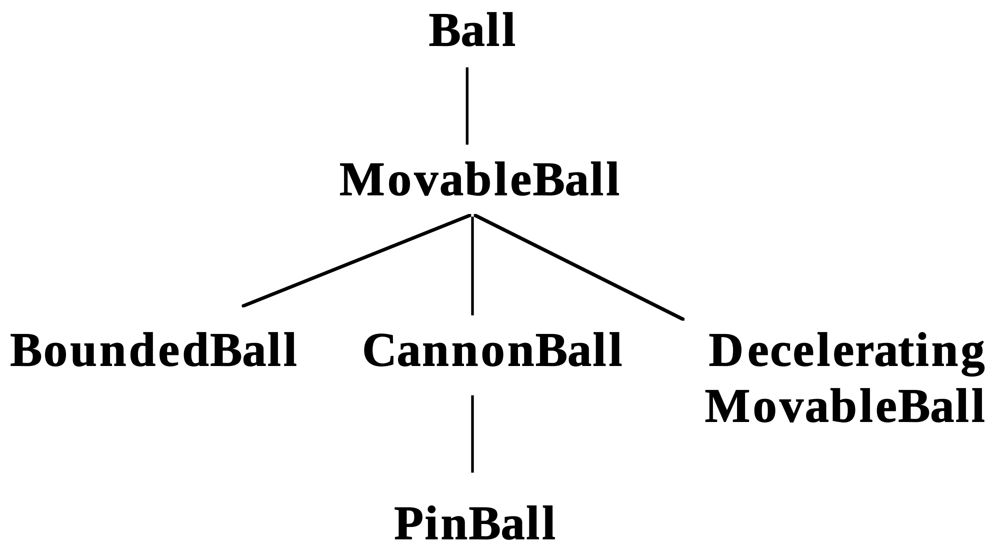
Eugene Wallingford
University of Northern Iowa
wallingf@cs.uni.edu

SIGCSE Technical Symposium
February 24, 2001

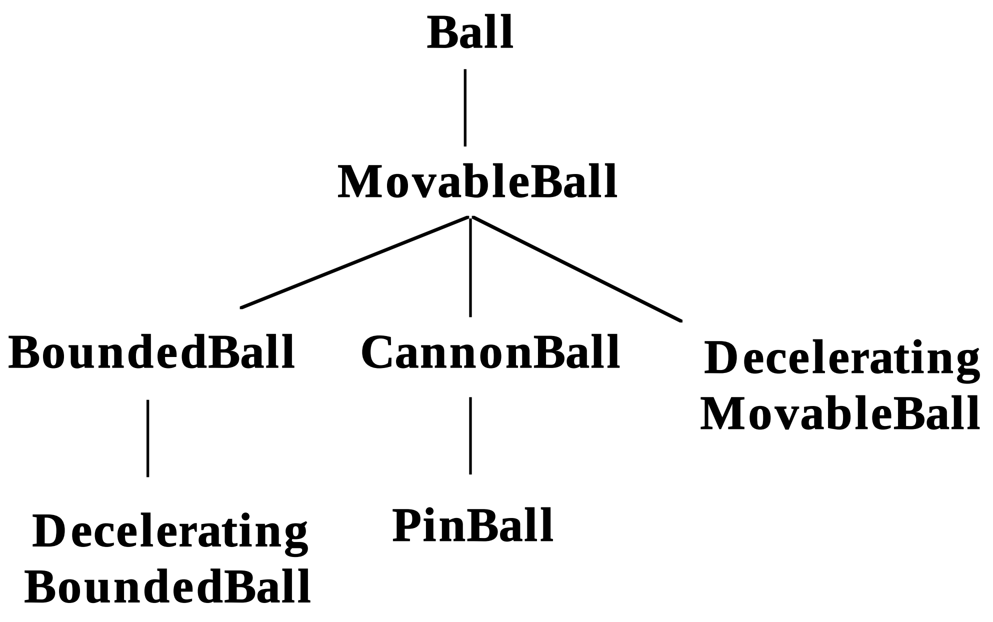
1 We begin with a simple `Ball` hierarchy.



2 Students implement a `Decelerating-MovableBall` class.



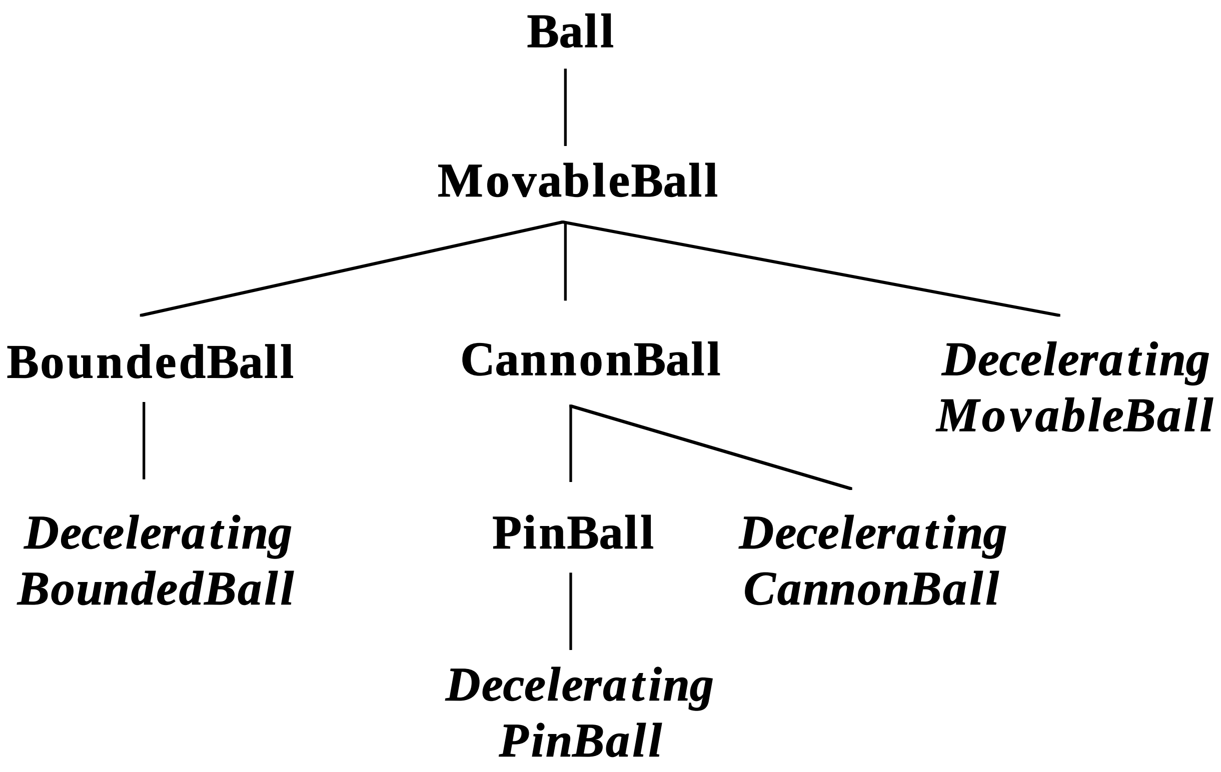
3 Students implement a `Decelerating-BoundedBall` class.



4 Students recognize the duplication: the decelerating ball classes override their superclass in exactly the same way.

What happens if we need to have cannonballs and pinballs that decelerate, too?

5 The worst-case scenario (or is it?):



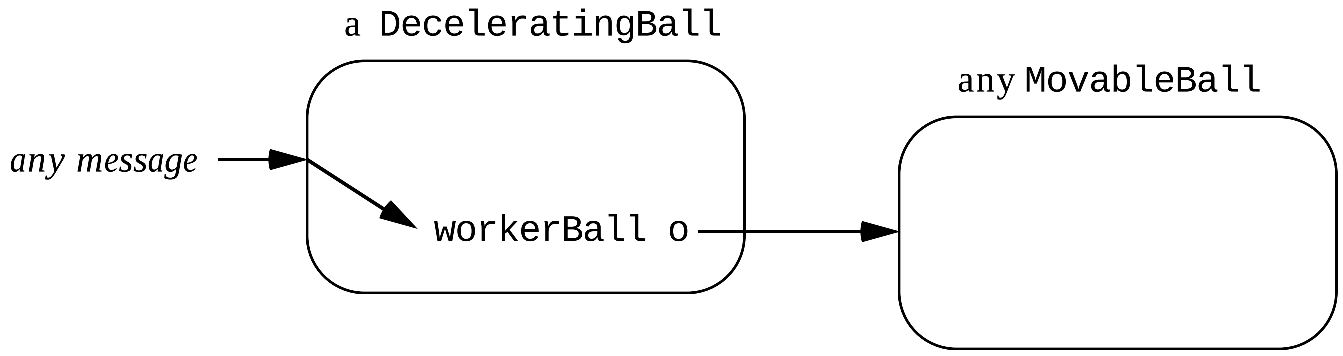
6 How can we avoid this duplication?

`BoundedBall`s respond to the same messages as `MovableBall`s. So, they are *substitutable* for one another.

How can we use this to our advantage?

7 Create a class that holds a `MovableBall` as an instance variable. Instances of the new class respond to all the same messages as `MovableBall`s.

An instance of the new class delegates all its messages to its instance variable. The only method that is different is `move()`, which also tells its instance variable to slow down a bit.

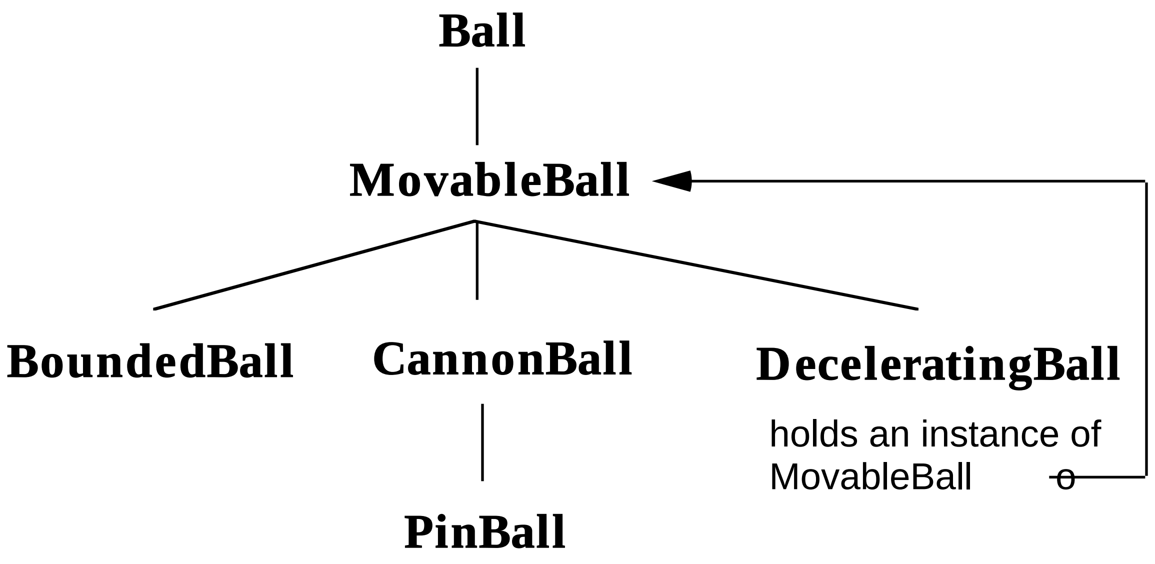


We can create `DeceleratingBall`s that wrap `MovableBall`s and `BoundedBall`s:

```
new DeceleratingBall(
    new MovableBall( ... ) );

new DeceleratingBall(
    new BoundedBall( ... ) );
```

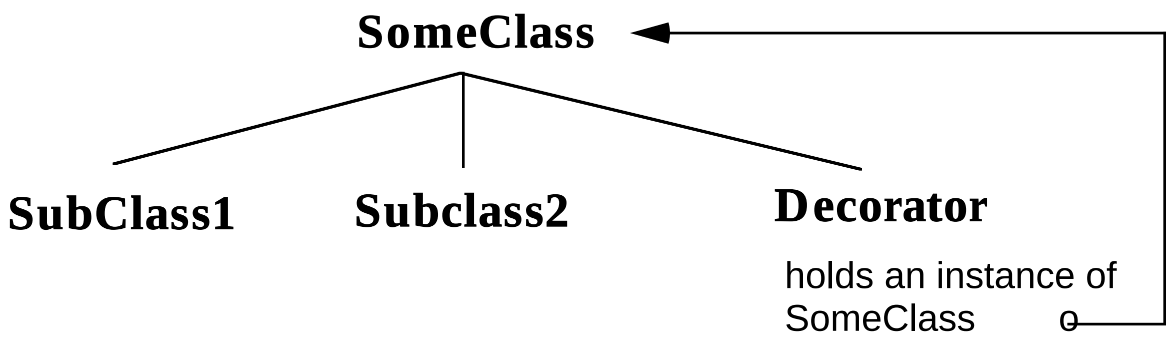
8 How can clients use `Decelerating-Balls` in places where they expect to use `MovableBalls`?



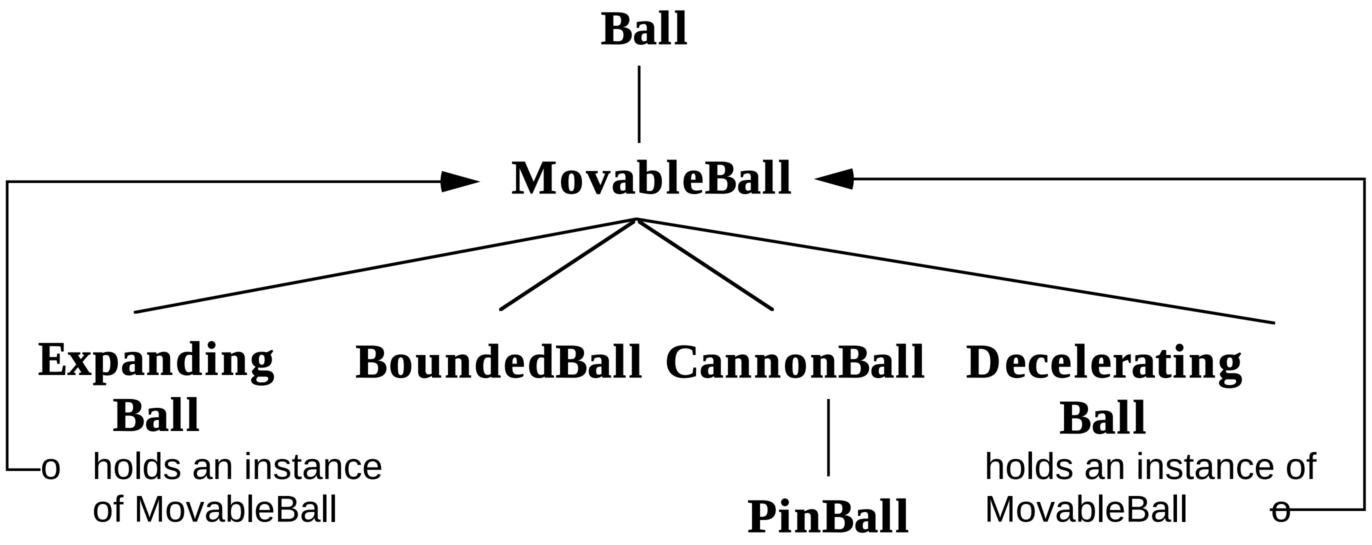
Now, a ball that decelerates can be used polymorphically in place of a `Movable-Ball`.

9 Later, we can consider...

- the general idea



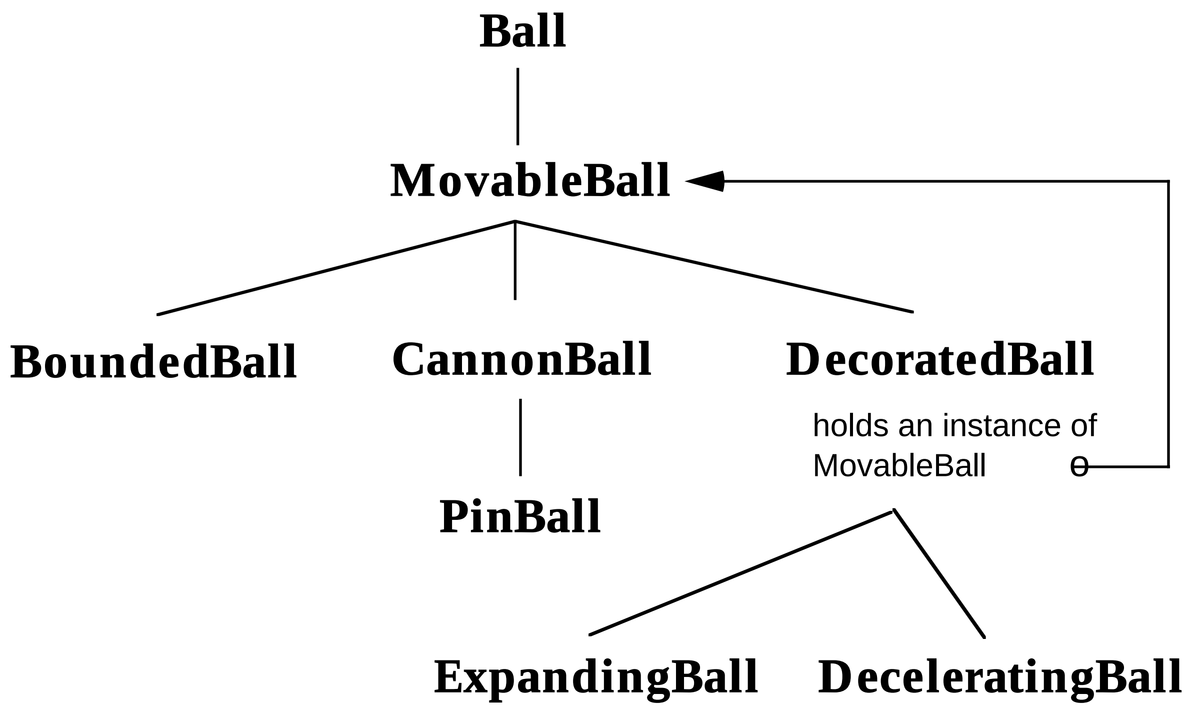
- implementing other decorators



- how one decorator can wrap another

```
new DeceleratingBall(
    new ExpandingBall(
        new MovableBall( ... ) );
```

- how to implement the delegation methods only once


Ball
MovableBall
BoundedBall
CannonBall
DecoratedBall
holds an instance of MovableBall
PinBall
ExpandingBall
DeceleratingBall