CISC 275: Introduction to Software Engineering

Using Polymorphism

Charlie Greenbacker
University of Delaware
Fall 2010

Quick Review

- Lets type A be used like type B
  - A inherits from B (*A extends B*), or
  - A implements an interface representing B

Quick Review

- Objects of different classes can have methods with the same name
  - Each method has a class-specific behavior

Without Polymorphism

```
interface Animal {

}
```

Without Polymorphism

```
class Dog implements Animal {
    String talk() {
        return "woof";
    }
}
```

Without Polymorphism

```
class Cow implements Animal {
        String talk() {
            return "moo";
        }
}
```

Without Polymorphism

```
class TestAnimals {
    public static void main(String[] args) {
        Animal a = new Cow();
        if (a instanceof Dog)
            System.out.println(((Dog)a).talk());
        else if (a instanceof Cow)
            System.out.println(((Cow)a).talk());
        else System.out.println("type error");
    }
}
```

With Polymorphism

```
interface Animal {
    // classes implementing the
    // Animal interface must
    // provide this method
     String talk();
}
```

With Polymorphism

```
class Dog implements Animal {
    String talk() {
        return "woof";
    }
}
```

With Polymorphism

```
class Cow implements Animal {
        String talk() {
            return "moo";
        }
}
```

With Polymorphism

```
class TestAnimals {
    public static void main(String[] args) {
        Animal a = new Cow();
        Animal b = new Dog();
        // no need for casting, instanceof, or
        // if-statements... just call the method
        System.out.println(a.talk());
        System.out.println(b.talk());
    }
}
```

Video

- Google: "The Clean Code Talks -- Inheritance, Polymorphism, & Testing" by Misko Hevery - Nov. 20, 2008
- http://www.youtube.com/watch?v=4F72VULWFvc

Terminology

- Dispatch: executing the appropriate code when a named method is invoked
- State: runtime type (class) of an object
- Guice: Java framework from Google for creating objects
- Also, an italic method signature in a UML class diagram shows the method is abstract