
Observer Pattern

CS580 Advanced Software Engineering

<http://cs356.yusun.io>

October 31, 2014

Yu Sun, Ph.D.

<http://yusun.io>

yusun@csupomona.edu



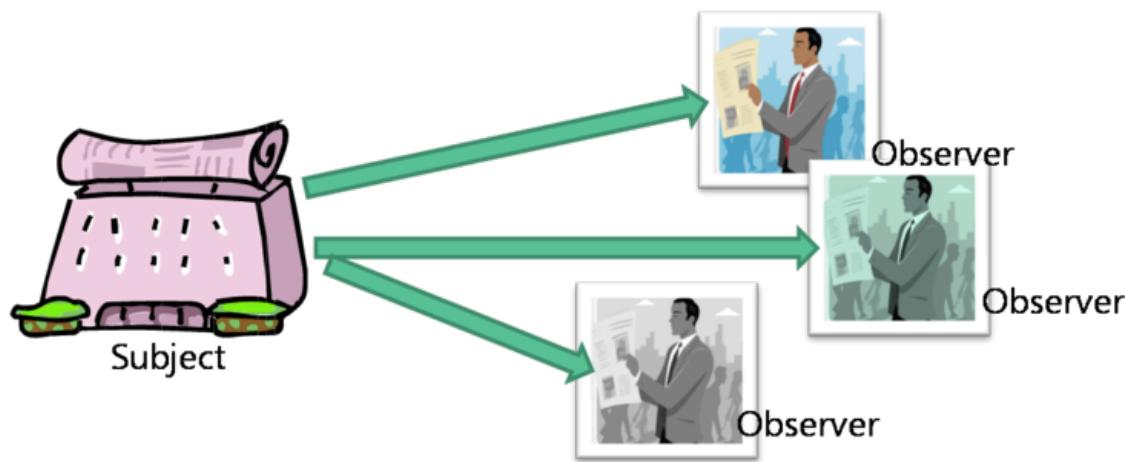
CAL POLY POMONA

Announcements

- ◆ Quiz 5
 - ◆ Singleton Pattern
 - ◆ Abstract Factory Pattern
 - ◆ Observer Pattern
- ◆ Next Wed: Assignment 2

Observer

- ◆ *Intent*
 - ◆ Define a **one-to-many dependency** between objects so that when one object changes state, all its dependents are notified and updated automatically
- ◆ *Also Known As – Dependents, Publish-Subscribe*



Non-Software Problem

- ◆ A professor wants to notify the secretary and the students every time there is a new exam
 - ◆ The secretary would then book the rooms
 - ◆ The students would (hopefully) prepare for the exam
- ◆ 1st pass: The prof knows who is interested, and call them directly

```
void setExam(...) {  
    // let the secretary know  
    mySecretary.bookRoom(...);  
  
    // let the students know  
    while (students.hasNext()) {  
        Student s = (Student)students.next();  
        s.study(...);  
    }  
}
```

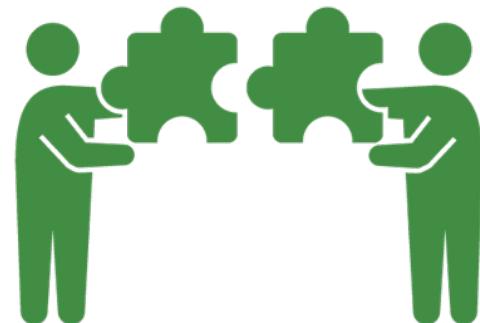
Non-Software Problem

- ◆ The prof needs to know about all the categories of people interested (the observers)
- ◆ What if a new person is interested?
 - ◆ What if a janitor needs to know room schedule?
- ◆ What if a given student is *not* interested?
- ◆ The prof needs to know what method to invoke on each interested party (e.g., study, clean, bookRoom)

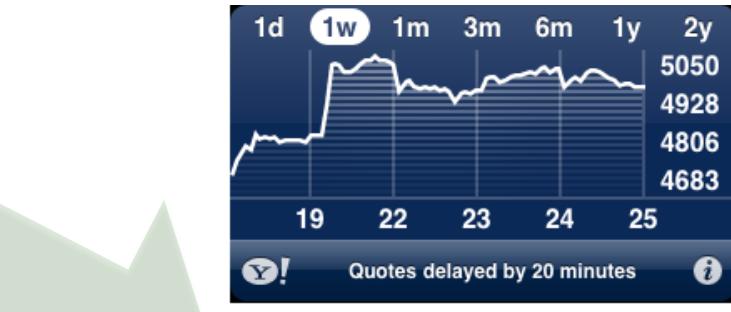


Motivation

- ◆ We want to notify objects
 - ◆ Without having to know **how many** there are
 - ◆ Without having to know **who** they are
- ◆ In the previous example, we want to “decouple” the professor from the people who should be notified



Stock Data Notification



A screenshot of a stock trading application. On the left is a candlestick chart for the Dow Jones Industrial Average (DJIA). A right-click context menu is open over the chart, with the "Send" option highlighted by a red box and a red arrow pointing to it. To the right of the chart is a detailed summary for "DOW JONES INDU...". It shows the current price of 11288.54, a 73.03 point increase or 0.65%. It also provides a 52-week summary with a high of 14198.1 and a low of 11157.21. Below the summary are buttons for "Industry Movers" and "Stock Movers". At the bottom, there is a brief description of the Dow Jones Industrial Average and a "Last updated: 22:40" timestamp. The overall interface is dark-themed.



Game Score Broadcasting



QUARTER	1st	2nd	3rd	4th	OT1	TOT
CHI	21	26	23	23	11	104
BOS	23	21	22	27	13	106

Team	PTS	REB	AST	BLK	STL	FOU
LAKERS	95	41	21	10	10	10
TRAIL BLAZERS	100	41	21	10	10	10

Team	PTS	REB	AST	BLK	STL	FOU
DAL	84	41	21	10	10	10
CHI	93	41	21	10	10	10
OKC	82	41	21	10	10	10
MIA	88	41	21	10	10	10
DEN	97	41	21	10	10	10

Social News Feed Update

facebook

Profile edit Friends Networks Inbox

Search Applications

Photos Like Posted Items Marketplace Movies Developer My Flair more

Get more sleep.
Work closer to home.

Find your dream job in your neighborhood.

SEARCH LOCAL JOBS

News Feed

Robert Scoble posted a video.
Ask The Facebooks About Gnomedex by Schlomo Rabinowitz 3:07 Recorded on Monday

Add a comment

Robert Scoble joined the group Social Computing for Business, Jeff Hammerbacher is attending The Singularity Summit 2007, Robert Scoble is off to visit Apple. Watch my KYTE.TV channel for more info (it's on my Facebook profile).

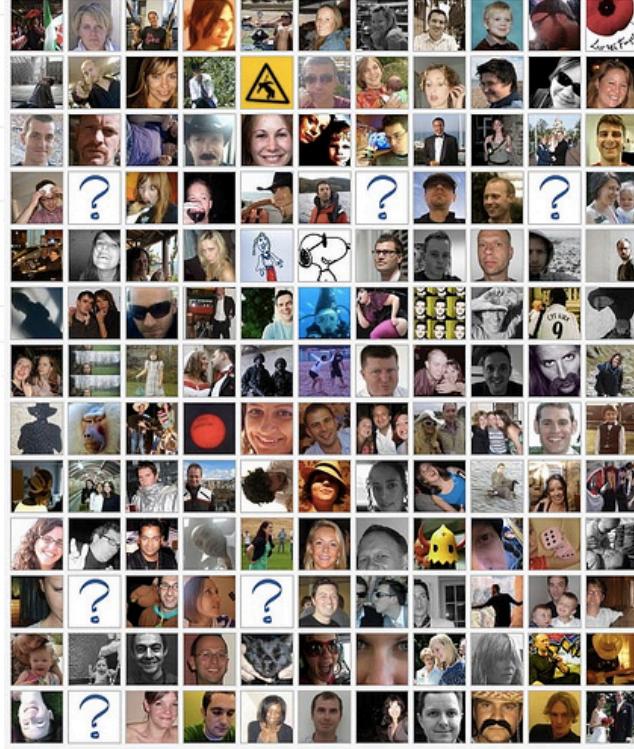
Miguel De Icaza received a gift.
Cuddle Monkey Designer: Susan Kare Give a Gift

Robert Scoble joined the group People Generated Media, Leah Pearlman is attending Tuesday Ultimate, Omar Shahine added the (fluff)friends application.

Robert Scoble was tagged in an album.

Scott Hanselman is running out of usb ports.

Brian Bain received a gift from Damon Dodge.



facebook

Reach Further Edit Page

News Feed Insights Events Photos Notes Links

Search

What's on your mind?

58 THE JESUS AND MARY CHAIN - HERE COMES ALICE THE JESUS AND MARY CHAIN-HERE COMES ALICE mmtv.youtube.com

13 minutes ago · 3 Like · Comment · Share

The Wool Room While you're on the BBC website, why not check out the competition on Good chance to win a wool bedding set from The Wool Room (<http://bit.ly>)

Win wool bedding and a sheep yogurt goodie box - Competitions - BBC Good Food Two lucky winners will win a selection of goodies from Thethedroom.com and Woodlands Dairy.

ago · Like · Comment · Share

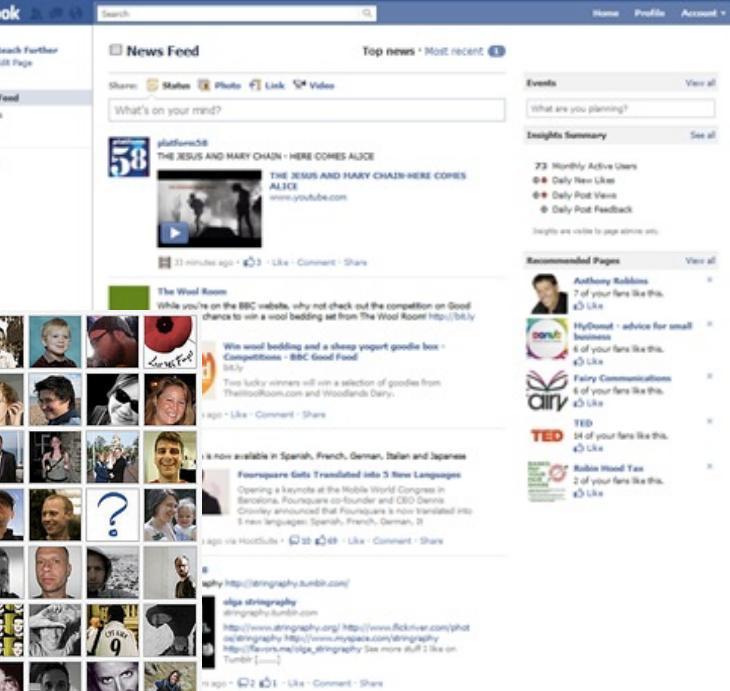
i is now available in Spanish, French, German, Italian and Japanese Foursquare Gets Translated into 5 New Languages Opening a keynote at the Mobile World Congress in Barcelona, Foursquare co-founder and CEO Dennis Crowley announced that Foursquare is now translated into 5 new languages: Spanish, French, German, Italian and Japanese. It's now available in Spanish, French, German, Italian and Japanese

info <http://foursquare.com> 22 49 Like · Comment · Share

info <http://strigraphy.tumblr.com/> strigraphy tumblr.com 1 https://www.strigraphy.org/ http://www.flickr.com/photos/strigraphy/ http://tinyurl.com/strigraphy See more stuff like on Tumblr [...]

ago · 2 10 Like · Comment · Share

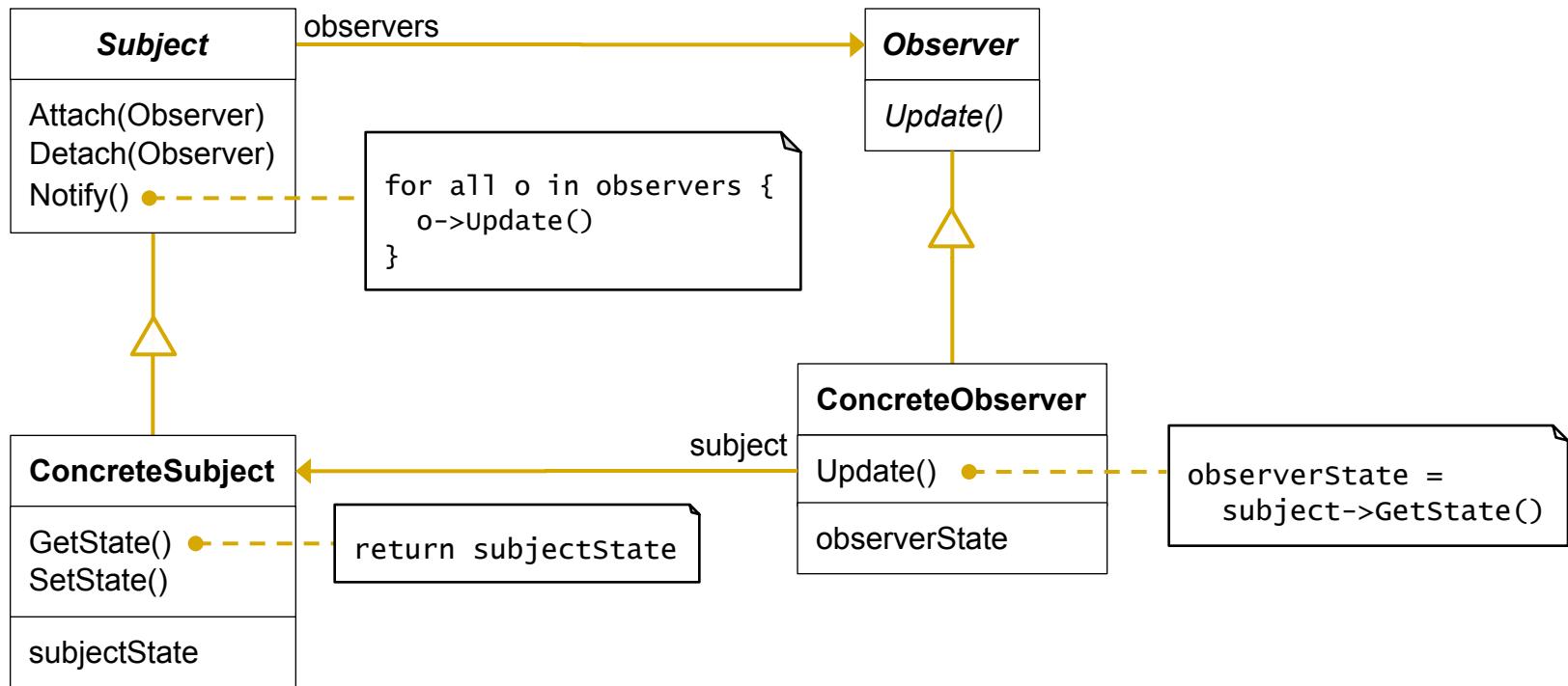
Events View all What are you planning? Insights Summary See all 23 Monthly Active Users Daily New Likes Daily Post Views Daily Post Feedback Insights are visible to page source sites Recommended Pages View all Anthony Robbins 7 of your fans like this. i Like Hybromat - advice for small business 6 of your fans like this. i Like Fairly Communications 6 of your fans like this. i Like TED 14 of your fans like this. i Like Robin Hood Tax 2 of your fans like this. i Like



Observer

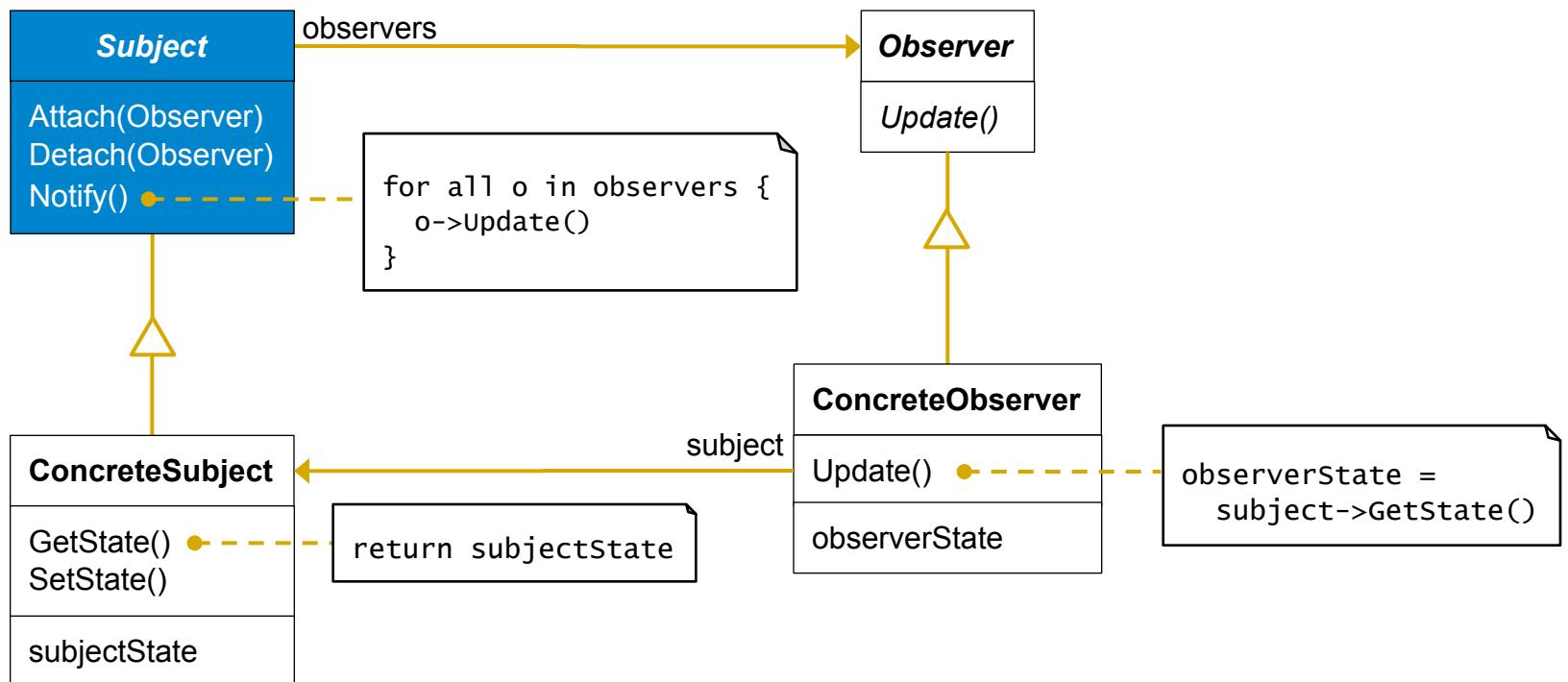
- ◆ A *publish-subscribe* mechanism
 - ◆ A **Subject** (object being observed) maintains a list of observers
 - ◆ **Observers** get added to that list by subscribing to the subject
 - ◆ **Observers** implement a common interface (an `update()` method)
 - ◆ When a change occurs, the **Subject** iterates through the list of **Observers** and calls the `update()` method on them

Structure



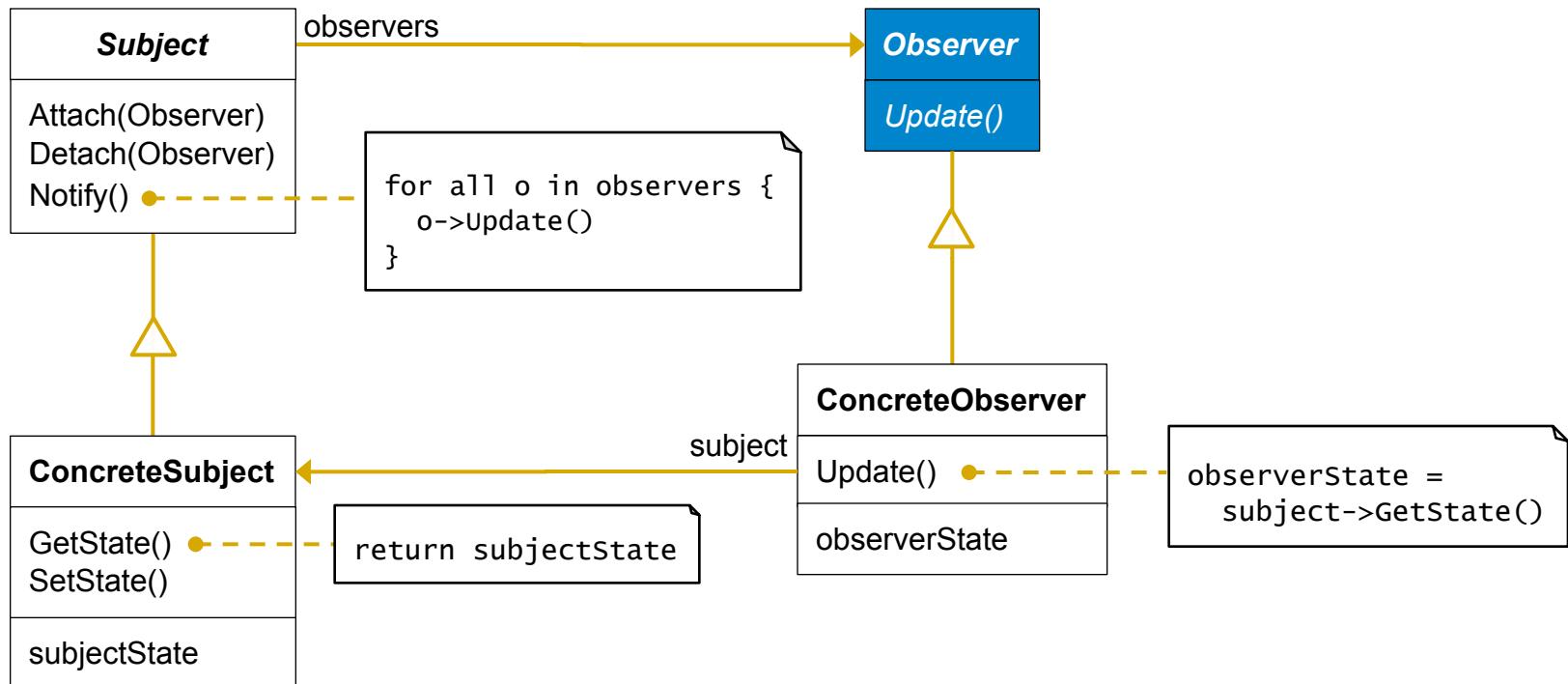
Subject

- ◆ Knows its observers. Any number of Observer objects may observe a Subject
- ◆ Provides an interface to add and remove Observer objects



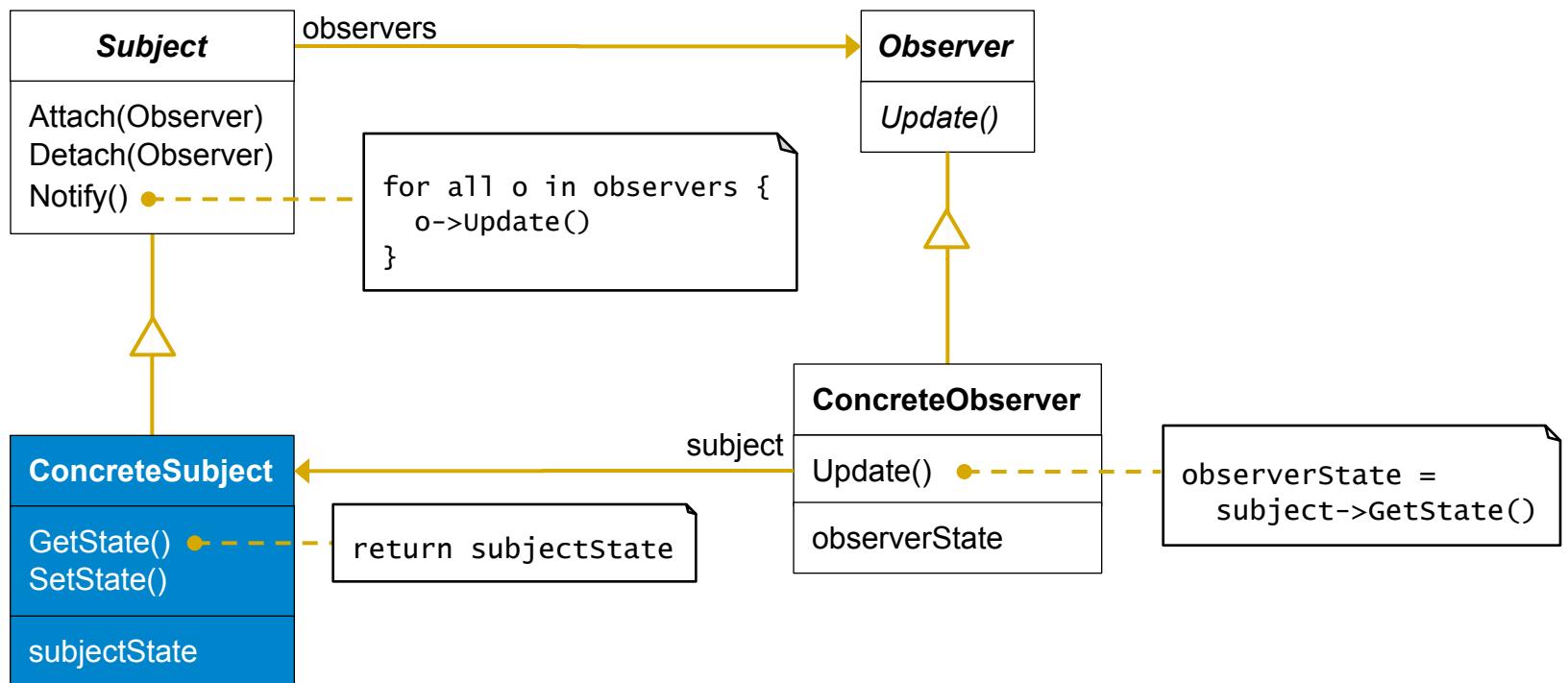
Observer

- ◆ Defines an updating interface for objects that should be notified of changes in a Subject



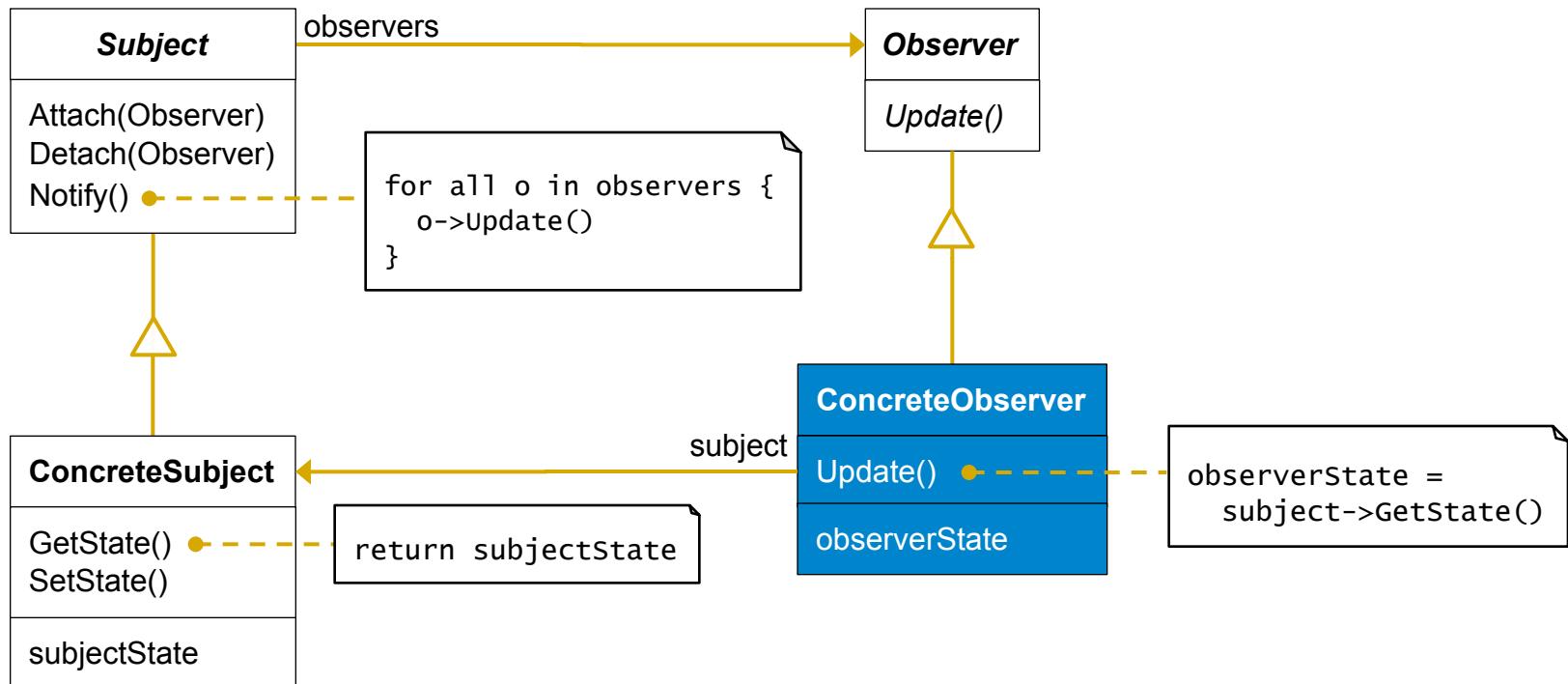
ConcreteSubject

- ◆ Stores state of interest to **ConcreteObserver** objects
- ◆ Sends a notification to its observers when its state changes



ConcreteObject

- ◆ Maintains a reference to a `ConcreteSubject` object
- ◆ Implements the `Observer` interface to keep its state consistent with the Subject



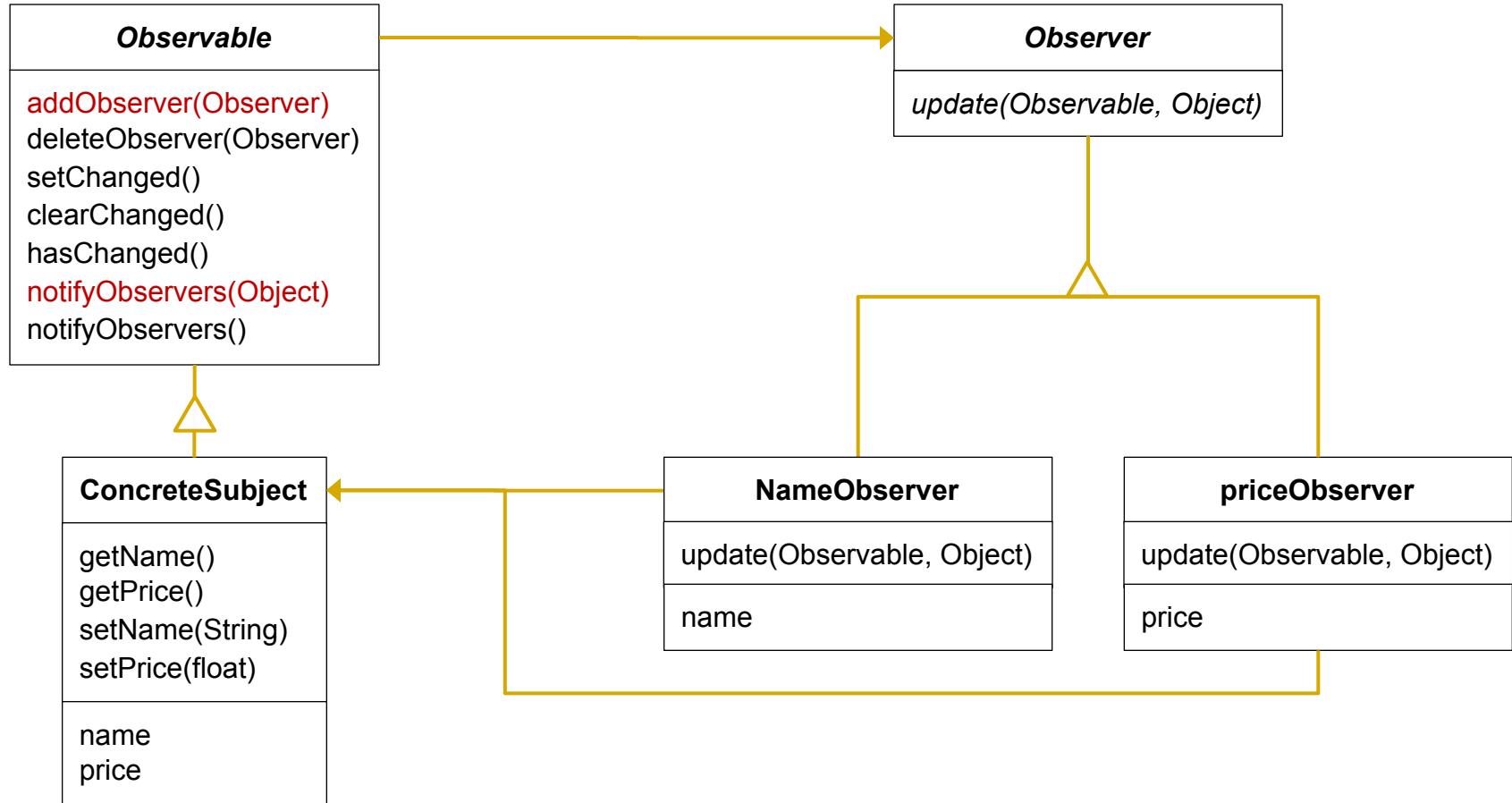
Applicability

- ◆ When a change to one object requires changing others
- ◆ When an object should be able to notify other objects without making assumptions about those objects

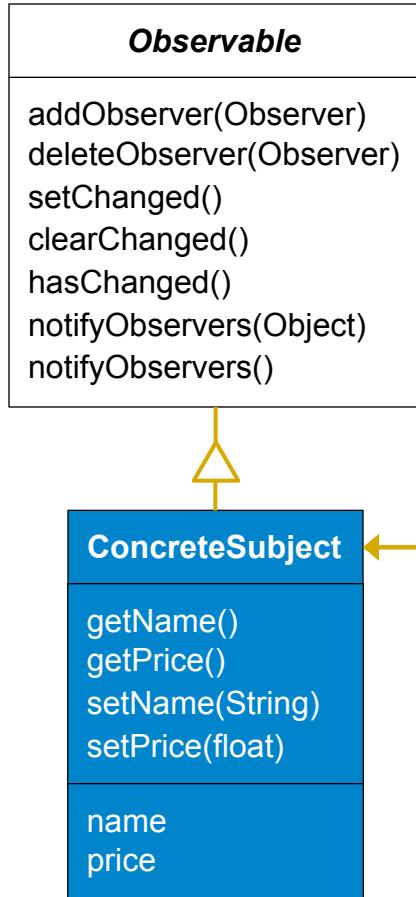
Example with Java's Observable/Observer

- ◆ We could implement the pattern “from scratch” in Java
- ◆ But Java provides built-in Observable/Observer classes
- ◆ `java.util.Observable` class is the base Subject class
 - ◆ Any class that wants to be observed extends this class
 - ◆ Provides methods to add/delete observers
 - ◆ Provides methods to notify all observers
- ◆ `java.util.Observer` interface is the Observer interface
 - ◆ It must be implemented by any observer class

Name and Price Observers



ConcreteSubject



```
public class ConcreteSubject extends Observable {
    private String name;
    private float price;

    public ConcreteSubject(String name, float price) {
        this.name = name;
        this.price = price;
    }

    public String getName() {return name;}
    public float getPrice() {return price;}

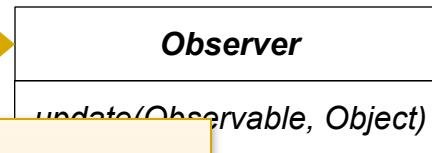
    public void setName(String name) {
        this.name = name;
        setChanged();
        notifyObservers(name);
    }

    public void setPrice(float price) {
        this.price = price;
        setChanged();
        notifyObservers(new Float(price));
    }
}
```

PriceObserver



```
public class PriceObserver implements Observer {  
    private float price;  
  
    public NameObserver() {  
        price = 0;  
    }  
  
    public void update(Observable obj, Object arg) {  
        if (arg instanceof Float) {  
            price = ((Float)arg).floatvalue();  
            System.out.println("Price Observer: Price changed to "  
                + price);  
        } else {  
            System.out.println("PriceObserver: Some other change "  
                + " to subject!");  
        }  
    }  
}
```



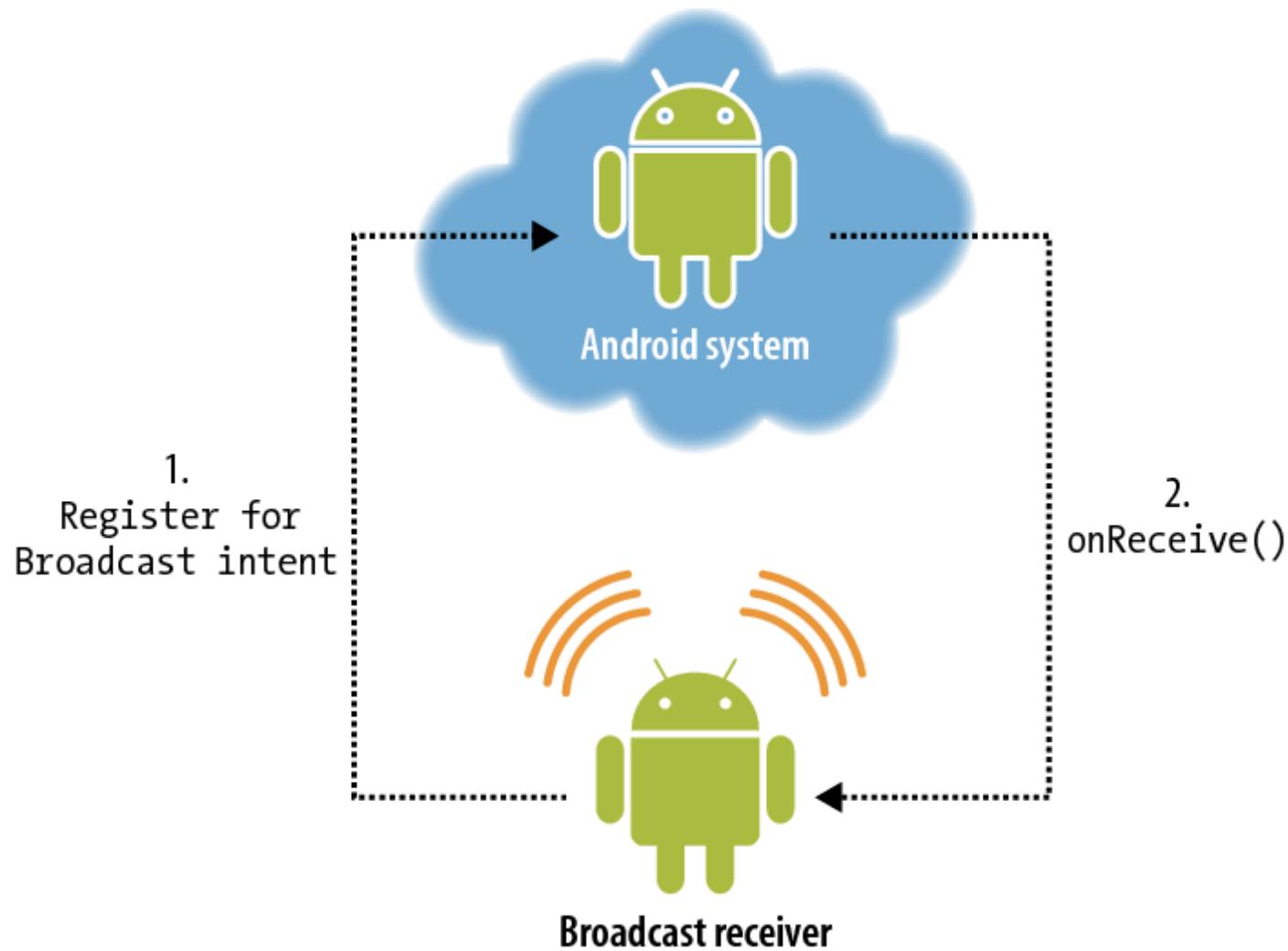
TestObservers

```
public class Testobservers {  
    public static void main(String args[]) {  
        // Create the Subject and Observers.  
        ConcreteSubject s = new ConcreteSubject("Corn Pops", 1.29f);  
        NameObserver nameObs = new NameObserver();  
        PriceObserver priceObs = new PriceObserver();  
  
        // Add those observers!  
        s.addobserver(nameObs);  
        s.addobserver(priceObs);  
  
        // Make changes to the Subject.  
        s.setName("Frosted Flakes");  
        s.setPrice(4.57f);  
        s.setPrice(9.22f);  
        s.setName("Sugar Crispies");  
    }  
}
```

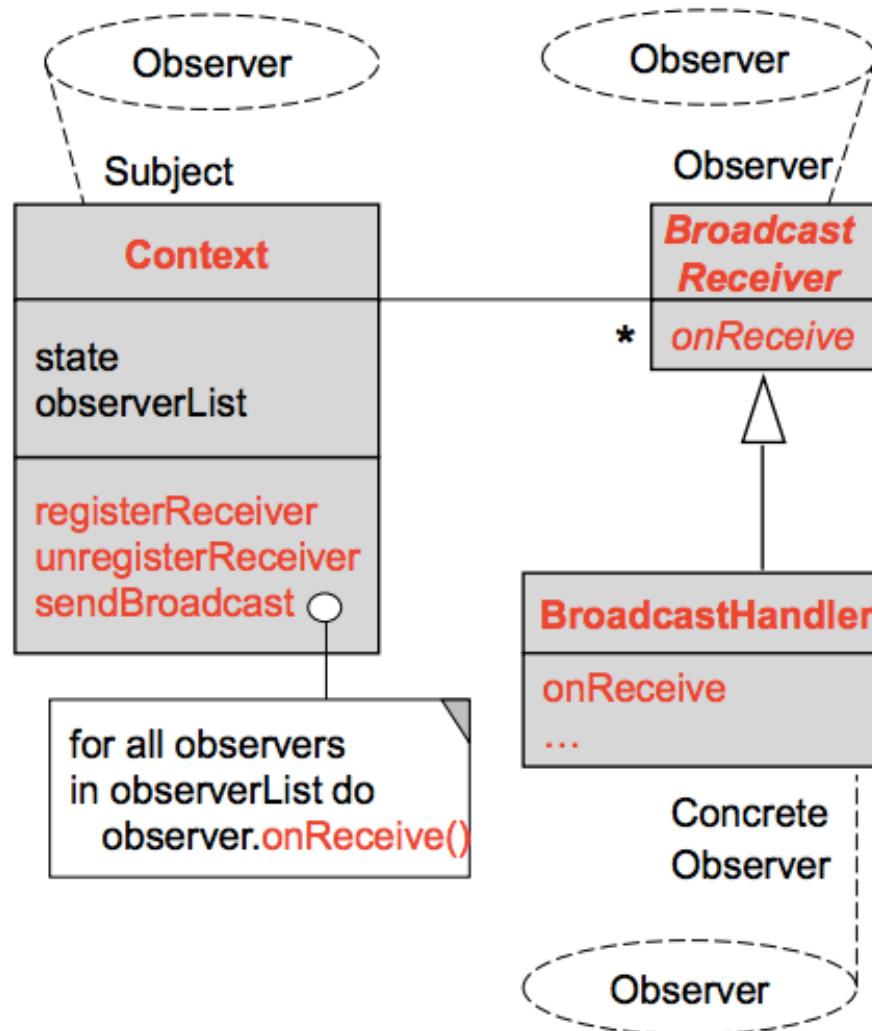
Output:

```
PriceObserver: Some other change to subject!  
NameObserver: Name changed to Frosted Flakes  
PriceObserver: Price changed to 4.57  
NameObserver: Some other change to subject!  
PriceObserver: Price changed to 9.22  
NameObserver: Some other change to subject!  
PriceObserver: Some other change to subject!  
NameObserver: Name changed to Sugar Crispies
```

Android Intent Broadcasting



Android Intent Broadcasting



Twitter



A global community of friends and strangers answering one simple question: **What are you doing?**

Biz Stone | **Invite** | Public | Codes | Setting

Characters available: 82

!thing for her birthday

But who will get your updates? Invite some of your friends to Twitter!

Or make your account public and let everyone read them.

1 new friend request!

102 Friends

Recent Public Updates



TC DM: "The Things You Said" - perfect walking beat pace less than 10 seconds ago from im

Please Sign In!

Email or Mobile Number

Password [Forgot?](#)

Remember me

Sign In!

Want an account?

Join for Free!

It's fast and easy!

Featured!

tina

SMITH Magazine

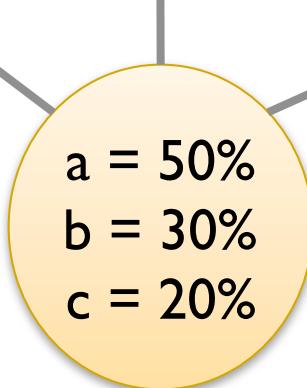
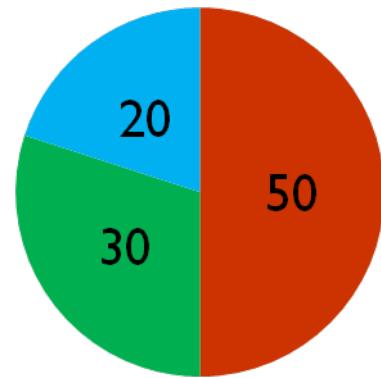
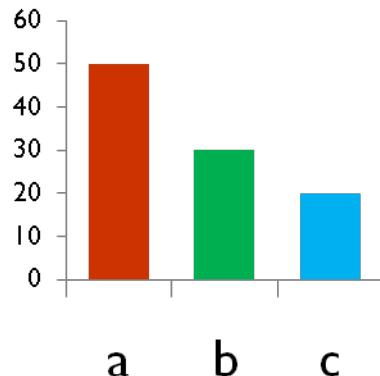
telene

phopkins

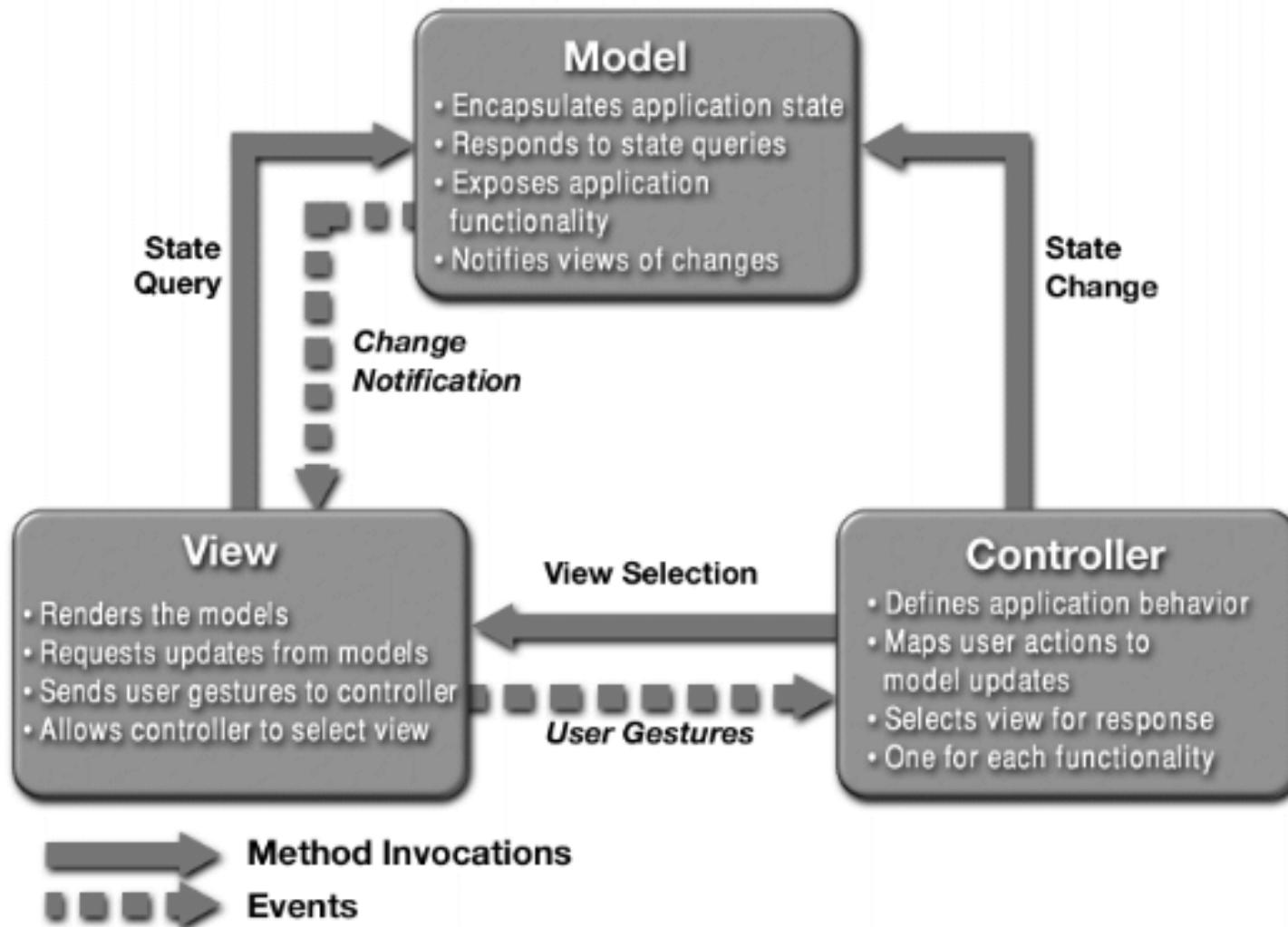
asuriso

Model–View–Controller (MVC)

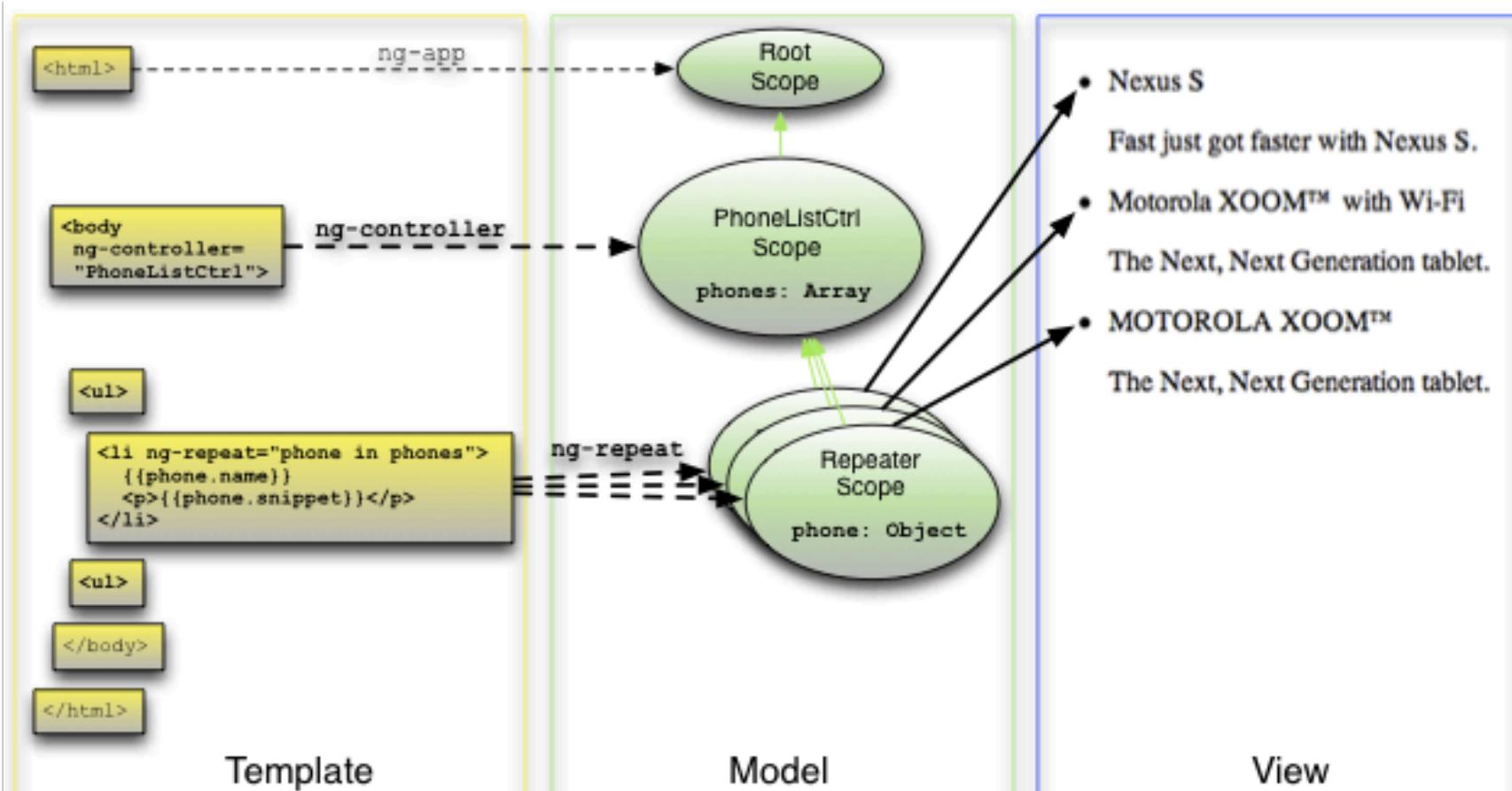
a	b	c
50	30	20



MVC Architecture



MVC Example - AngularJS



-----► Implicit Scope Declaration

→ Scope Inheritance

↔ Model / View Data-binding

· <https://angularjs.org/>