
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

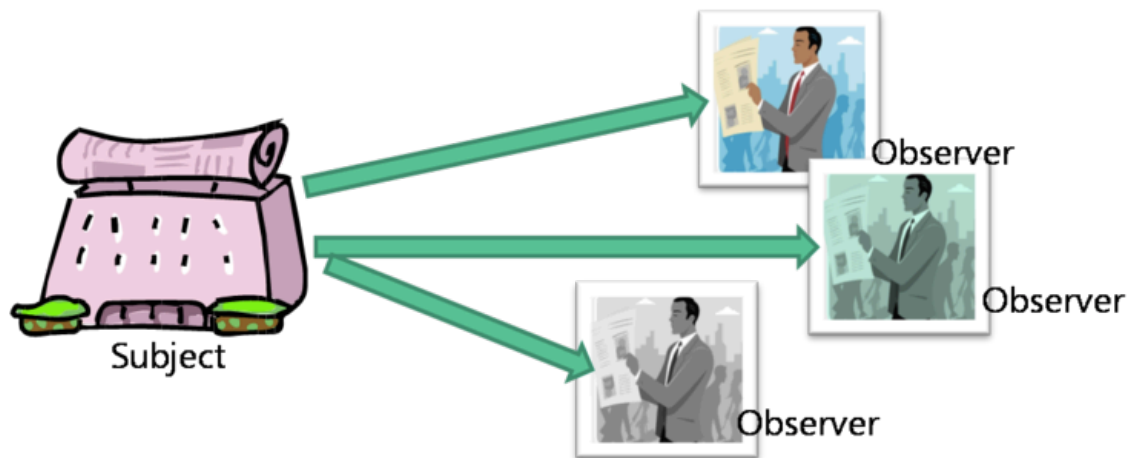
Announcements

- ◆ 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



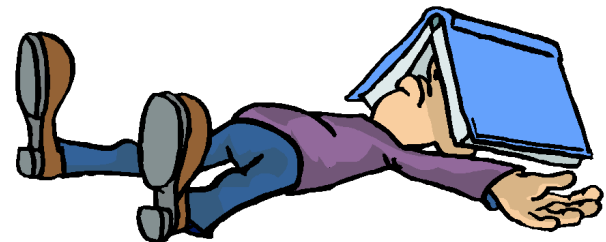
Non-Software Problem

- ◆ 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(...);  
    }  
}
```

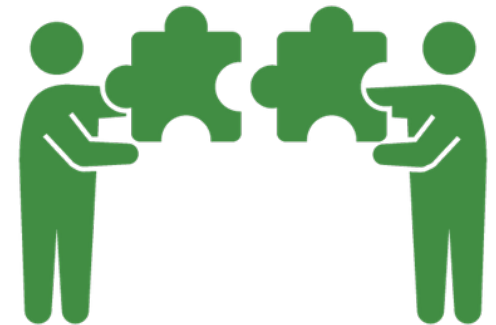
Non-Software Problem

- ◆ 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)

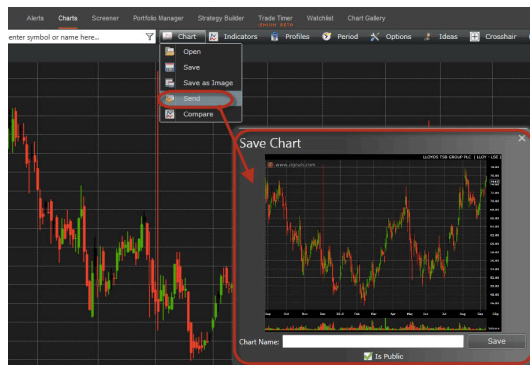


Motivation

- ◆ 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



Game Score Broadcasting



AT&T 11:05 PM

ALL GAMES SCORES **GAME TIME**

CHI 104 106 BOS

GAME 5 - FINAL

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

PLAYERS: BULLS < PLAYER STATS > CELTICS

CEL TICS		PTS	REB	AST	BLK	STL	FOU
43 K. Perkins (C)	16	19	2	7	1	0	
11 G. Davis (F)	21	6	3	0	0	2	
34 P. Pierce (F)	26	7	1	1	2	2	
20 R. Allen (G)	10	1	2	1	0	6	
9 R. Rondo (G)	28	8	11	0	2	4	
42 T. Allen	0	1	0	0	2		
8 S. Marbury	3	1	2	0	0	1	
44 B. Scalabrine	2	1	2	0	0	2	
50 E. House	0	0	0	0	0	1	

Team Totals: 106 44 24 9 5 20

SCORES SCHEDULE PLAYOFFS STANDINGS

LOS ANGELES **LAKERS** 95

PORTLAND **TRAIL BLAZERS** 100

AWAY 25:49 HOME

LF1	TOTAL GAME	RF1
95 Points	100 Points	100
39/22 (42%) Field Goals	37/11 (40%) Field Goals	4
4/15 (26%) 3 Pointers	6/19 (31%) 3 Pointers	9
13/17 (76%) Free Throws	20/25 (80%) Free Throws	4
8 Dunks	10 Dunks	2
10 Fast Break Points	12 Fast Break Points	4
50 Points in the Paint	42 Points in the Paint	9
12 Second Chance Points	32 Second Chance Points	22
29 Bench Points	32 Bench Points	9
17 Assists	22 Assists	4
10 Offensive Rebounds	9 Offensive Rebounds	10
45 Defensive Rebounds	43 Defensive Rebounds	10
4 Steals	10 Steals	7
7 Blocks	9 Blocks	11
14 (12) Turnovers (Points Off)	10 (15) Turnovers (Points Off)	7
19 Team Fouls	11 Team Fouls	10
0 Biggest Lead	7 Biggest Lead	10
25:49 Time of Possession	22:10 Time of Possession	4
2 Timeouts Remaining	4 Timeouts Remaining	

TEAM STATS

NBA GAME TIME

box score play

MAVERICKS DAL LEADS 2-1 THUNDER

FINAL CHI @ IND 84 89

CHI @ IND 43.9% | 36.5%

FINAL DAL @ POR 82 84

FINAL G3: W. Finals SAS @ MEM 88 91

MEM LEADS 2-1

FINAL G3: W. Finals OKC @ DEN 97 94

OKC LEADS 3-0

CHI WINS 4-2

CHI @ ATL

2011 NBA playoffs

Conference Finals: DAL vs. OKC

FINAL G1: W. Finals OKC @ DAL 112 121

FINAL G2: W. Finals OKC @ DAL 106 100

FINAL G3: W. Finals DAL @ OKC 93 87

3:00 PM ET G4: W. Finals DAL @ OKC

3:00 PM ET G5: W. Finals OKC @ DAL

Social News Feed Update

facebook Profile edit Friends Networks Inbox

Search

Applications edit
Photos
Like
Posted Items
Marketplace
Movies
Developer
My Flickr
more

Get more sleep.
Work closer to home.
Find your dream job in your neighborhood.
SEARCH LOCAL JOBS

News Feed Preferences

Robert Scoble posted a video.
Asking The Facebooks About Gnomedex by Schlomo Rabinowitz 3127 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 Kytex.tv channel for more info (it's on my Facebook profile).

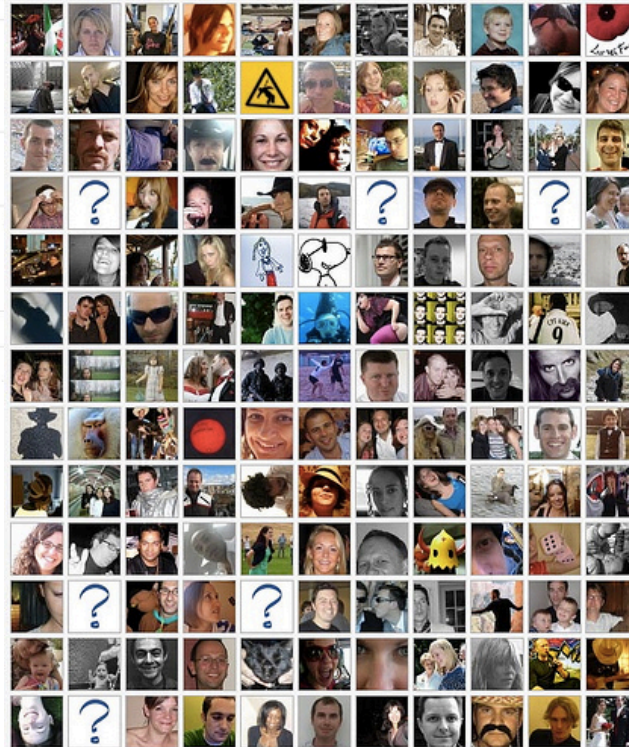
Higuel De Icaza received a gift.
Cuddle Monkey Designer: Susan Kare Give a Gift

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 Search Home Profile Account

News Feed Top news Most recent

Share Status Photo Link Video

What's on your mind?

platform:58 THE JESUS AND MARY CHAIN - HERE COMES ALICE
ALICE www.youtube.com

23 minutes ago · 3 · Like · Comment · Share

The Wood 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 Wood Room http://bit.ly/...

Win wool bedding and a sheep yogurt gaffle box - Competitions - BBC Good Food bit.ly/...

Two lucky winners will win a selection of goodies from TheWoodRoom.com and Woodlands Dairy.

ago · Like · Comment · Share

is now available in Spanish, French, German, Italian and Japanese

FourSquare Gets Translated into 3 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 3 new languages: Spanish, French, German. It

ago via HostLulu · 30 · Like · Comment · Share

why http://stringraphy.tumblr.com/

elga stringraphy stringraphy.tumblr.com
http://www.stringraphy.org/ http://www.flickr.com/photos/stringraphy http://www.stringraphy.com/stringraphy http://www.myspace.com/stringraphy See more stuff I like on Tumblr [...]

ago · 2 · Like · Comment · Share

Events View all
What are you planning?

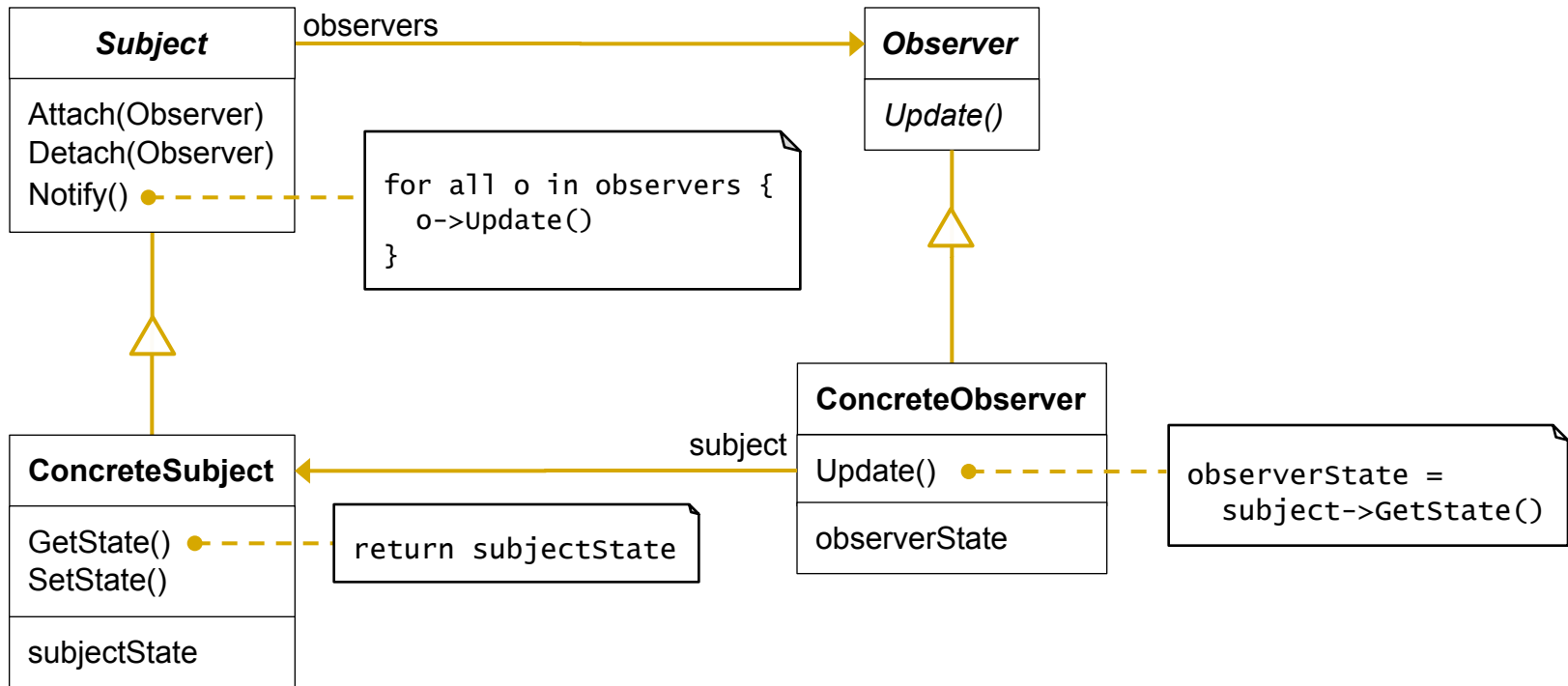
Insights Summary See all
73 Monthly Active Users
Daily New Likes
Daily Post Views
Daily Post Feedback
Insights are visible to page admin only.

Recommended Pages View all
Anthony Robbins 7 of your fans like this.
Hydrosol - advice for small business 6 of your fans like this.
Fairy Communications 6 of your fans like this.
TED 14 of your fans like this.
Robin Hood Tax 2 of your fans like this.

Observer

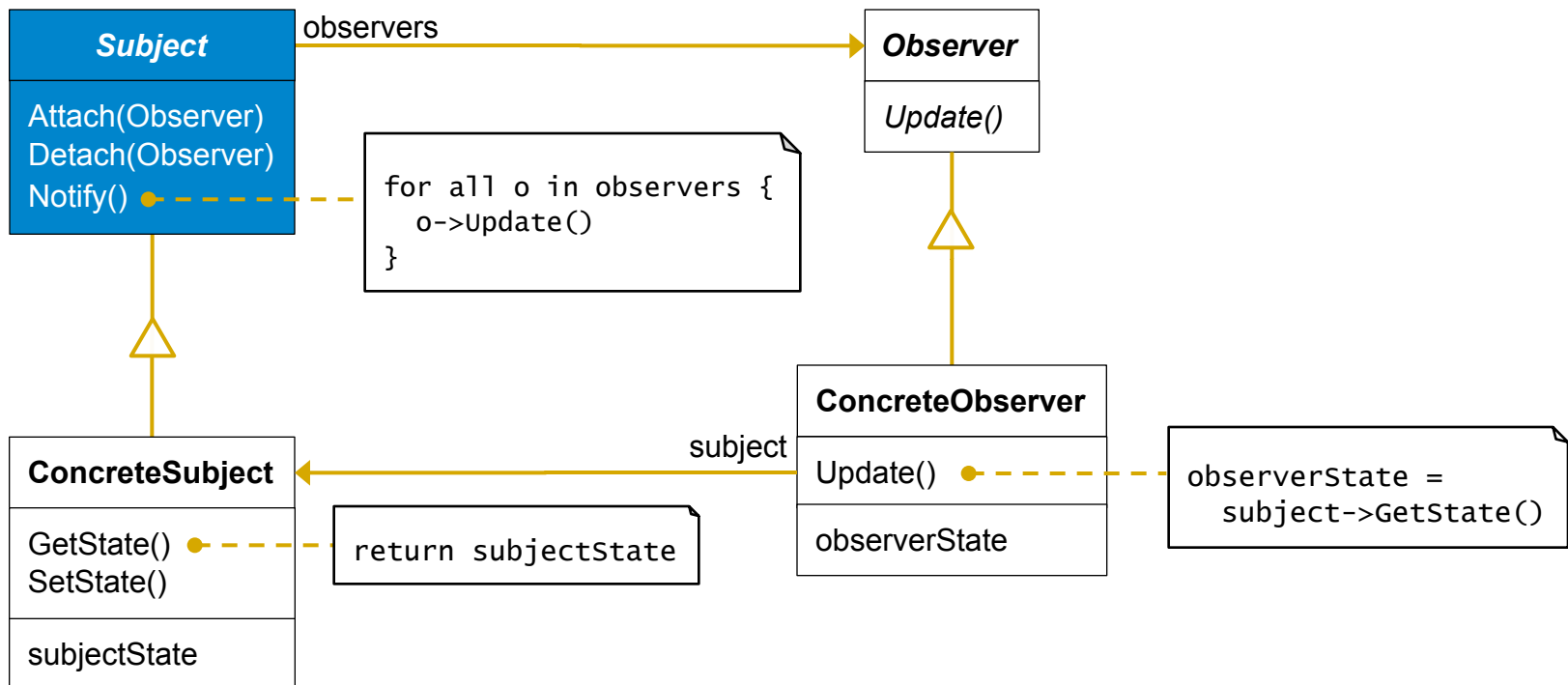
- ◆ A *publish-subscribe* mechanism
 - ◆ A **Subject** (object being observed) maintains a list of observers
 - ◆ **Observers** get added to that list by subscribing to the subject
 - ◆ **Observers** implement a common interface (an **update()** method)
 - ◆ When a change occurs, the **Subject** iterates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

Structure



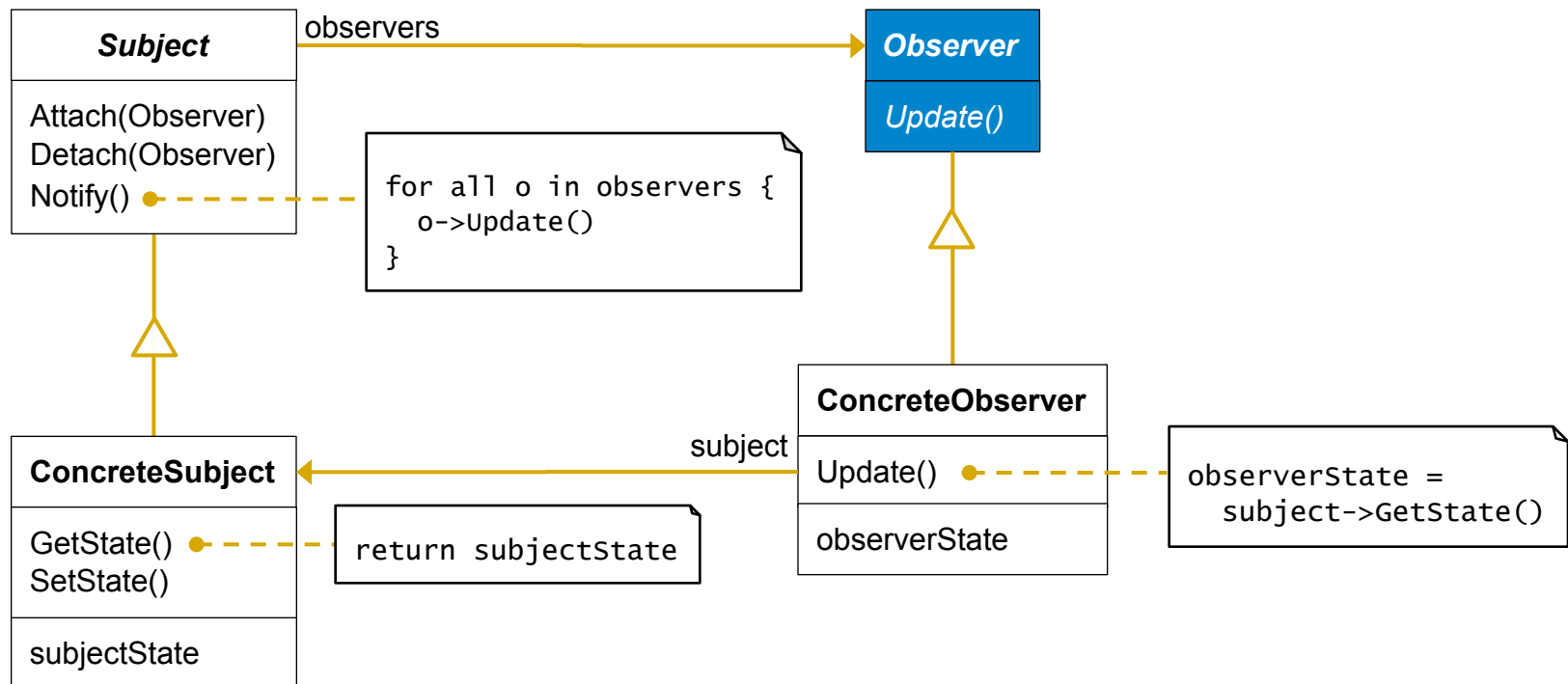
Subject

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



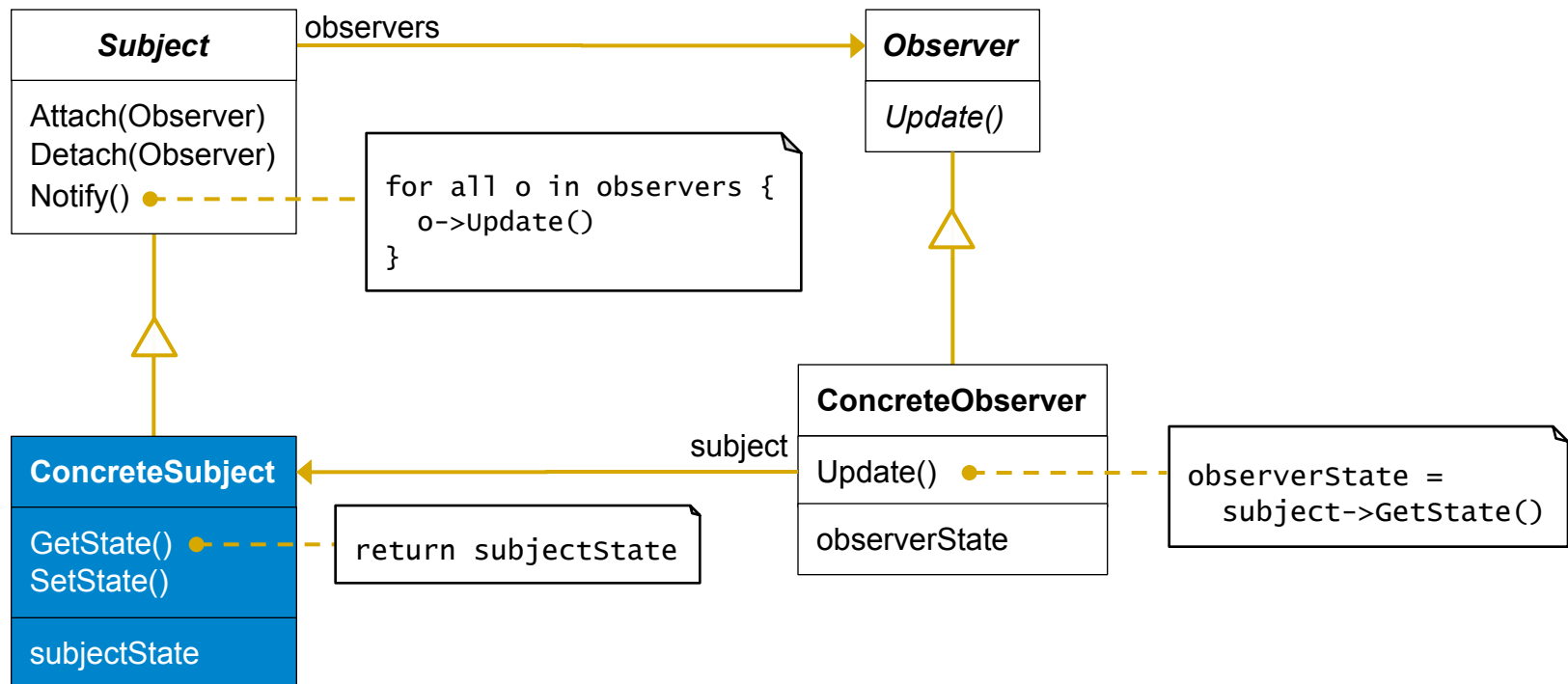
Observer

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



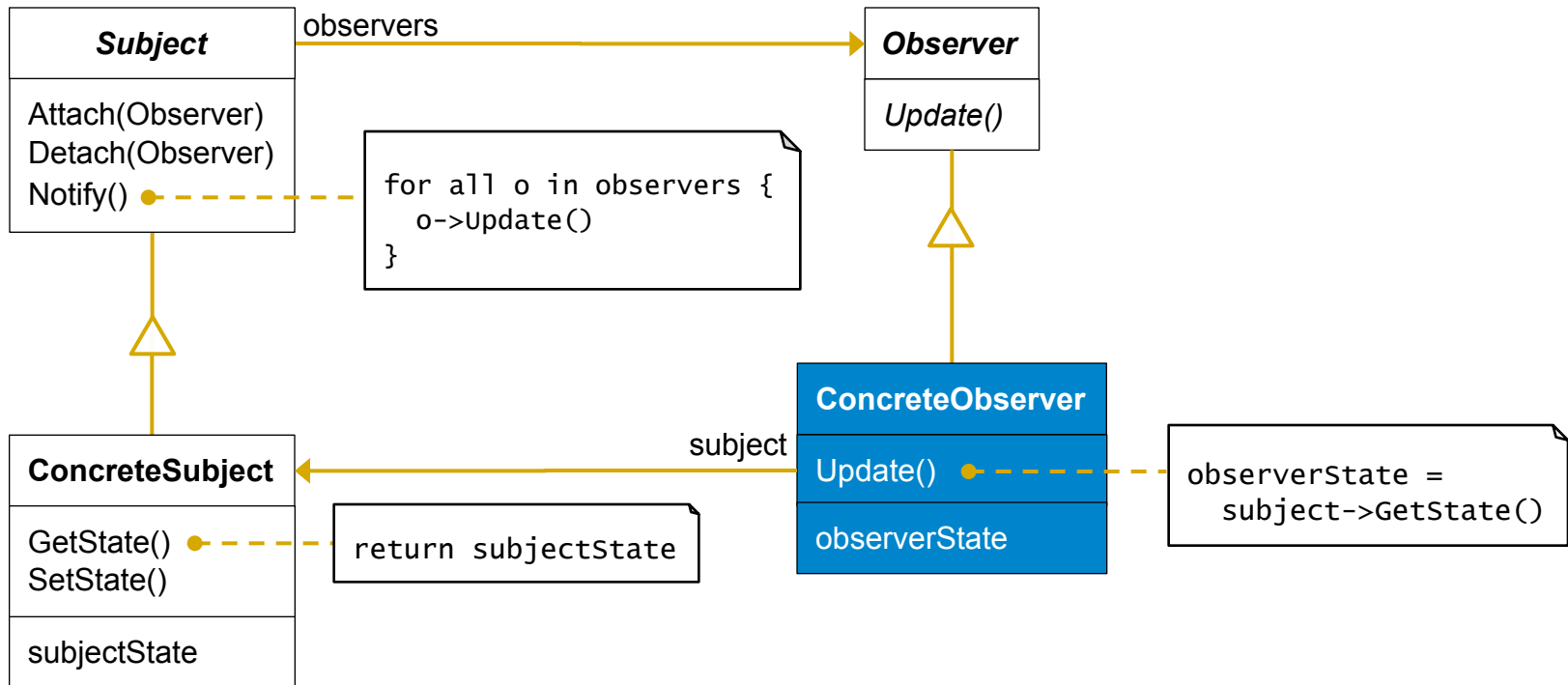
ConcreteSubject

- ◆ 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



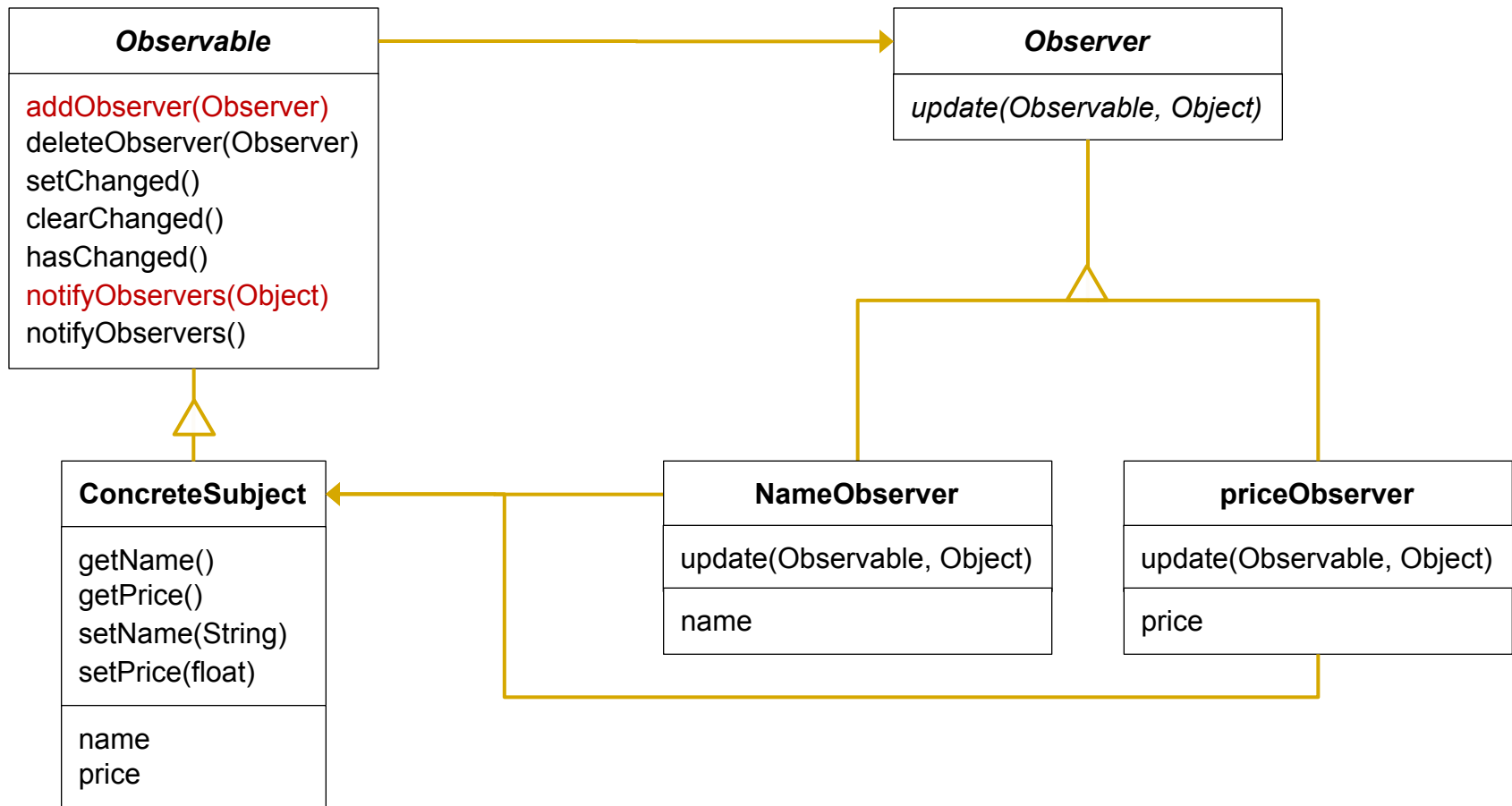
Applicability

- ◆ 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

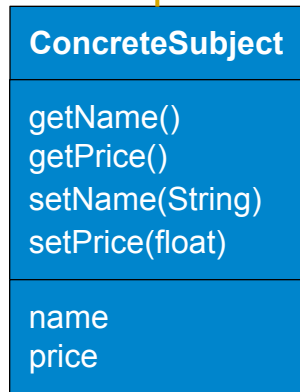
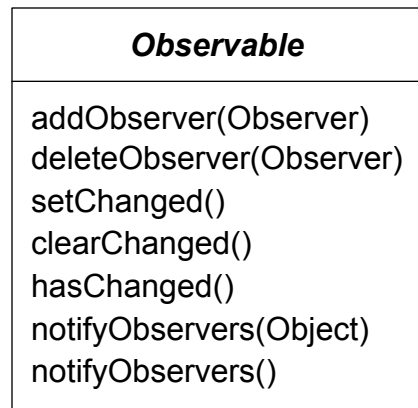
Example with Java's Observable/Observer

- ◆ 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;

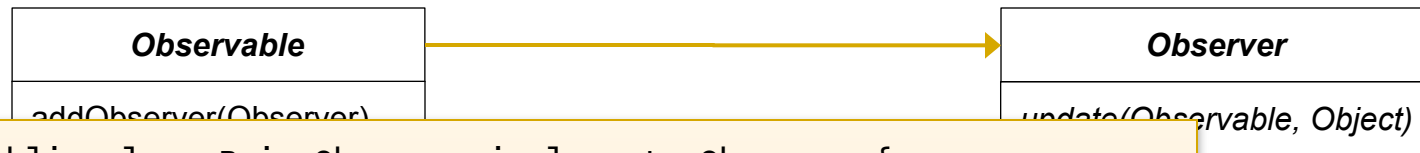
    public ConcreteSubject(String name, float price) {
        this.name = name;
        this.price = price;
    }

    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));
    }
}
```

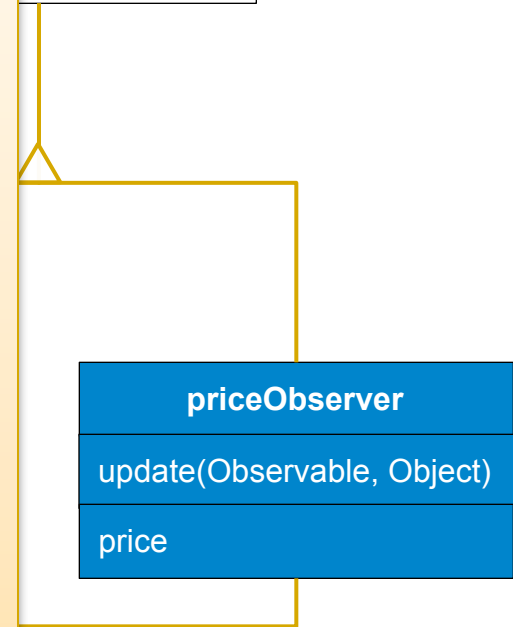
PriceObserver



```
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();

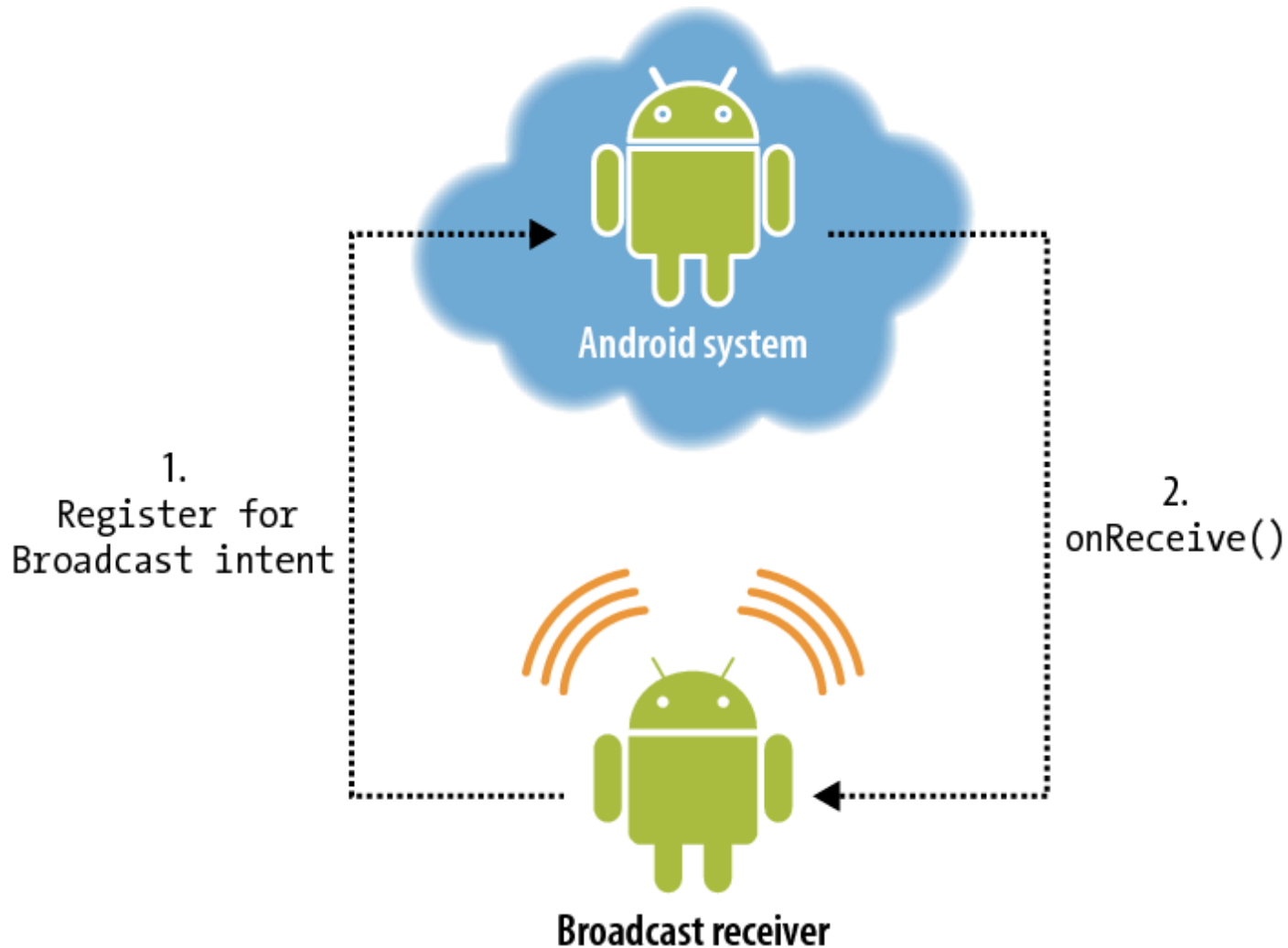
        // Add those Observers!
        s.addObserver(nameObs);
        s.addObserver(priceObs);

        // Make changes to the Subject.
        s.setName("Frosted Flakes");
        s.setPrice(4.57f);
        s.setPrice(9.22f);
        s.setName("Sugar Crispies");
    }
}
```

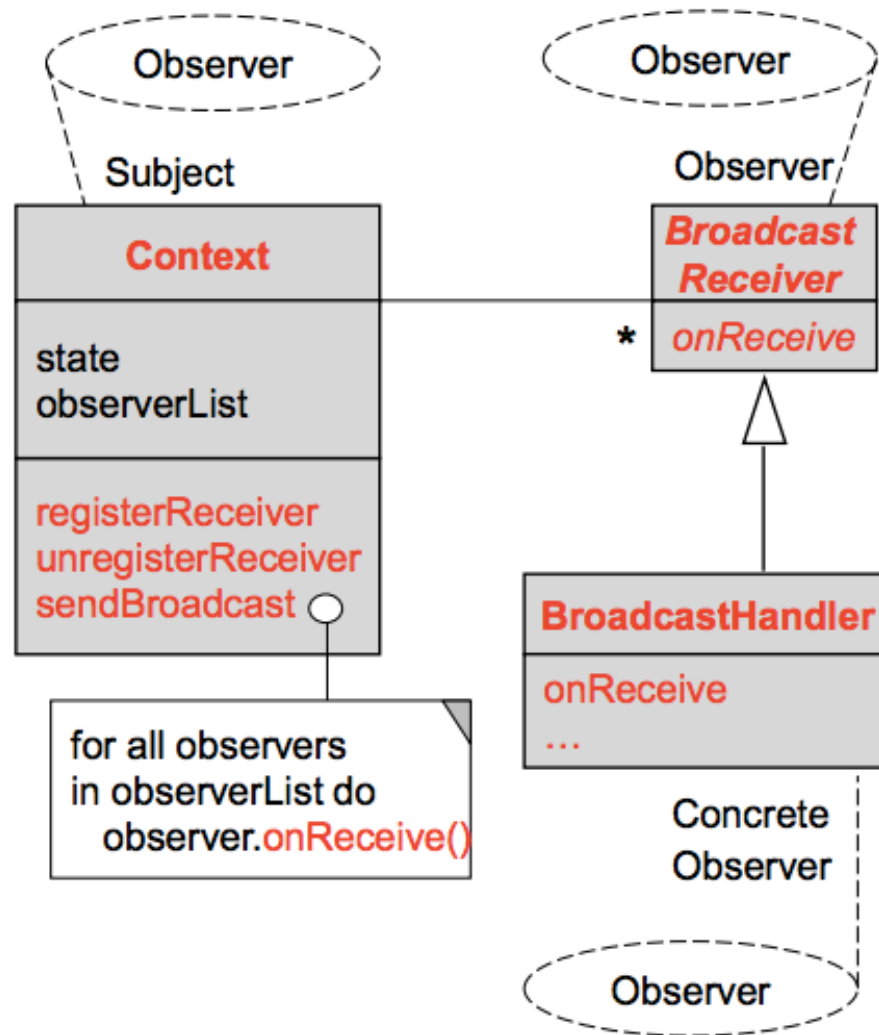
Output:

```
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

twitter

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



Recent Public Updates



TÇ 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

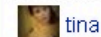
Sign In!

Want an account?

Join for Free!

It's fast and easy!

Featured!



tina



SMITH Magazine



telene



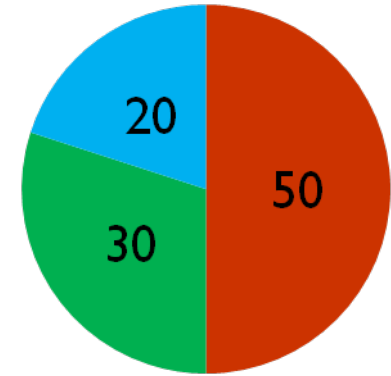
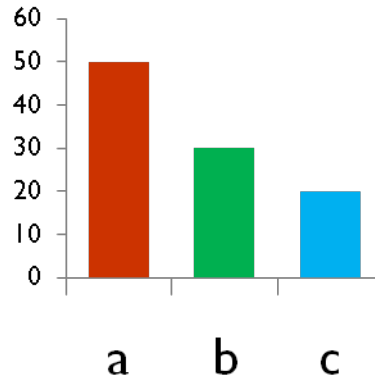
phopkins



...

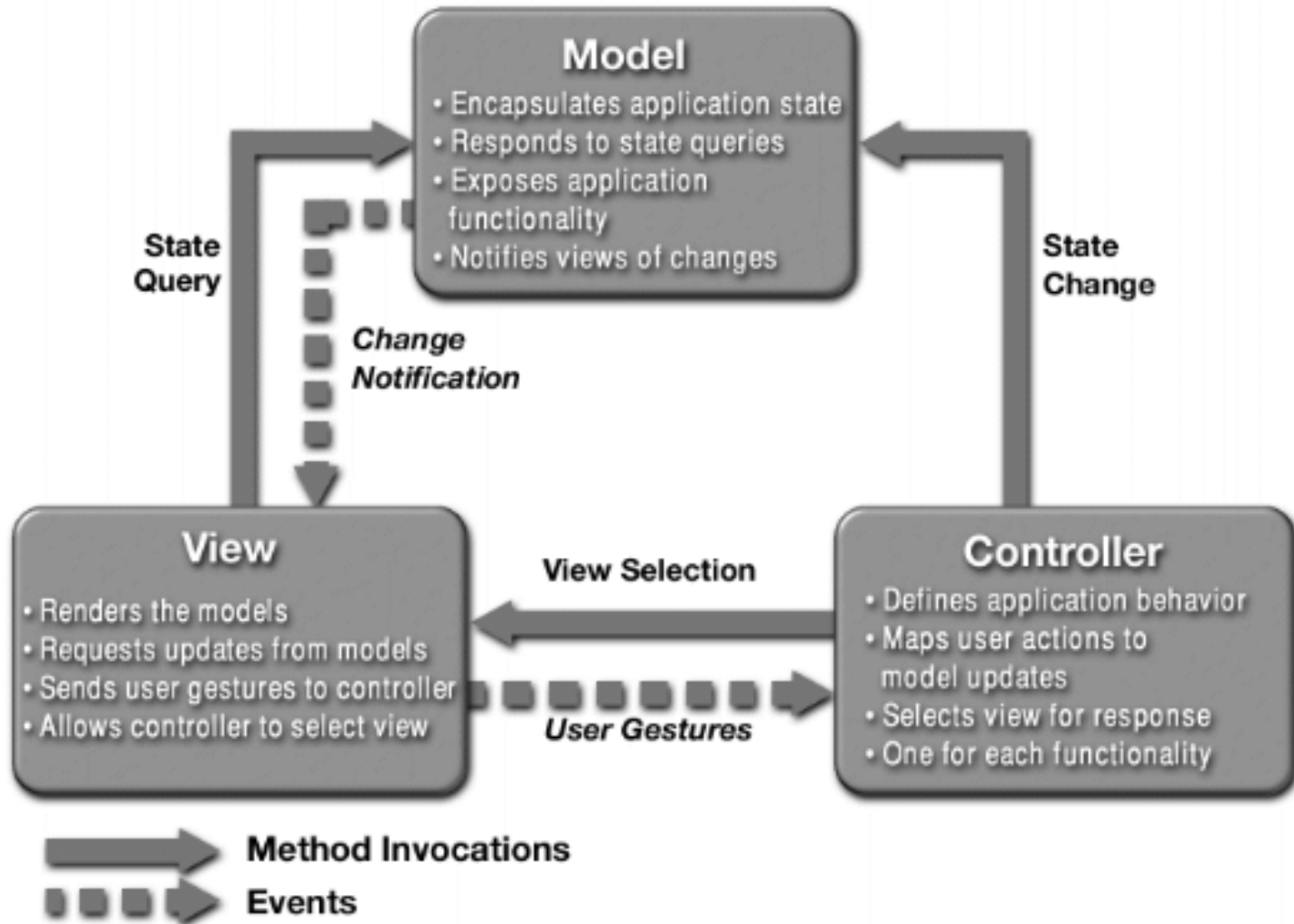
Model-View-Controller (MVC)

a	b	c
50	30	20



a = 50%
b = 30%
c = 20%

MVC Architecture



MVC Example - AngularJS

