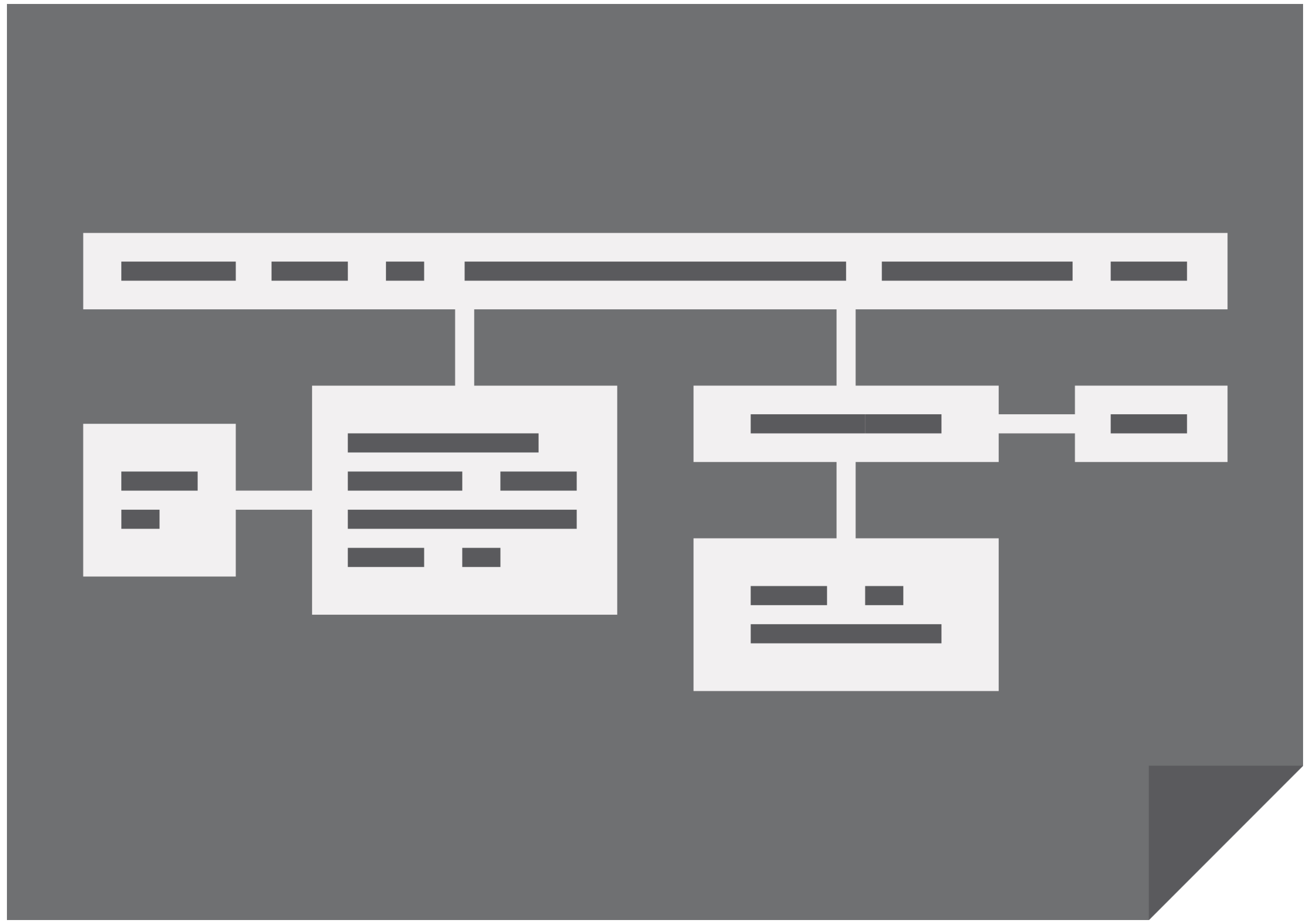
Mediator Pattern

Bryan Hansen

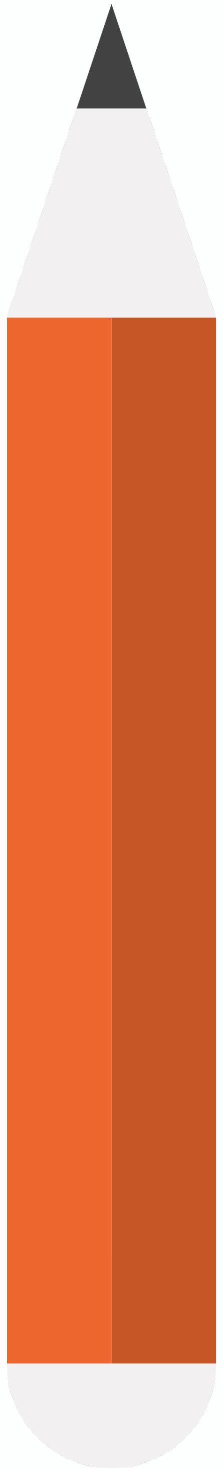
twitter: bh5k | http://www.linkedin.com/in/hansenbryan

Concepts

- Loose coupling
- Well-defined, but complex
- Reusable components
- Hub / Router
- Examples:
  - java.util.Timer
  - java.lang.reflect.Method#invoke()



Design



Interface based

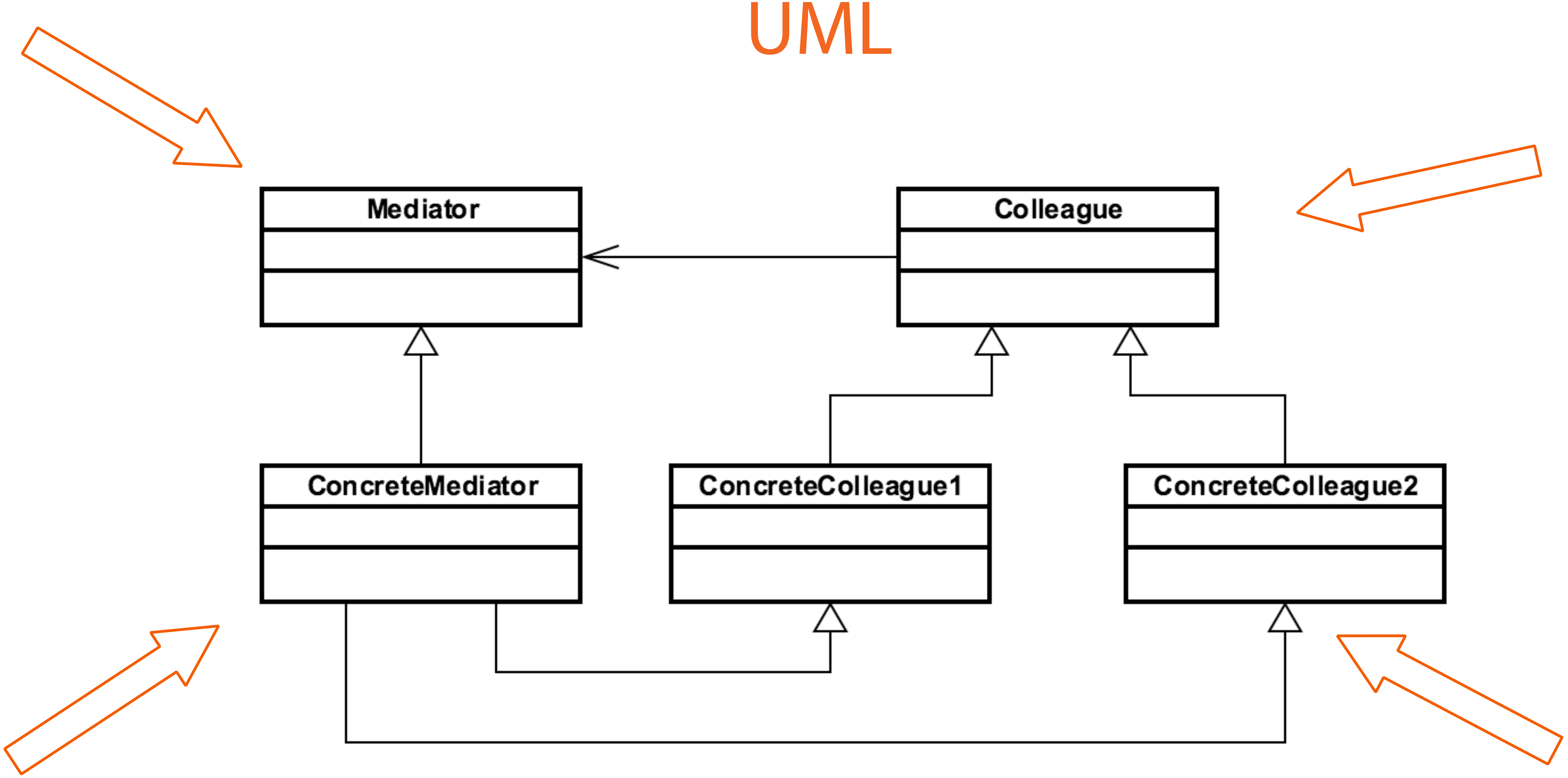
Concrete class

Minimizes inheritance

Mediator knows about colleagues

Mediator, ConcreteMediator

UML


Mediator
Colleague
ConcreteMediator
ConcreteColleague1
ConcreteColleague2

Everyday Example - Timer

```
public MediatorEverydayDemo(int seconds) {
   toolkit = Toolkit.getDefaultToolkit();
   timer = new Timer();
   timer.schedule(new RemindTask(), seconds * 1000);
   timer.schedule(new RemindTaskWithoutBeep(), seconds * 2 * 1000);
}

class RemindTask extends TimerTask {
   public void run() {
      System.out.println("Time's up!");
      toolkit.beep();
   }
}
```

Exercise Mediator

Mediator, Colleague

Command

Light

Pitfalls

- Deity object
- Limits subclassing
- Over or with Command

Contrast

Mediator

- Defines Interaction
- Object decoupling
- More specific

Observer

- One-to-Many
- Object decoupling
- More generic

Mediator Summary

- Loose coupling
- Simplified communication
- Mediator complexity
- Use with Command