Today

- Inversion of Control
- Dependency Injection
- The Adapter Pattern
Don't Call Us, We'll Call You

Inversion of Control
a.k.a.
The Hollywood Principle

Ashwin J Mathew          Spring 2008
School of Information, UC Berkeley
Examples of IoC

- **Widget**
  - +call()
  - #callTemplate()

- **JumpyWidget**
  - #callTemplate()

- **DeafWidget**
  - #callTemplate()

- **Strategy**
  - +doSomething()

- **Observer**
  - +observers: Collection
  - +callObservers()
  - +addObserver(observer: Observer)

- **StrategyFoo**
  - +doSomething()

- **StrategyBar**
  - +doSomething()
Service Locater

- Components explicitly look up services
- All components reference Service Locater
- Service Locater is a central registry of services
Service Locator in Waffle

**WaffleServlet**

- `init()`
- `getComponentRegistry()`: ComponentRegistry

**ComponentRegistry**

- `locateByKey(key:Object)`
- `locateByClass(class:Class)`
- `getActionMethodExecutor()`: ActionMethodExecutor
- `getActionMethodResponseHandler()`: ActionMethodResponseHandler
- `getBindMonitor()`: BindMonitor
- `getControllerDefinitionFactory()`: ControllerDefinitionFactory
- `getViewDispatcher()`: ViewDispatcher
Dependency Injection

- IoC in locating component implementations
- Inject dependent components prior to execution by “auto-wiring”
- Forms of Dependency Injection
  - Constructor Injection
  - Setter Injection
  - Interface Injection
DI in Waffle

```
WaffleServlet
+controllerDefinitionFactory: ControllerDefinitionFactory
+init()
-getComponentRegistry(): ComponentRegistry
+service(request: HttpServletRequest, response: HttpServletResponse)

PicoComponentRegistry
+picoContainer: MutablePicoContainer
+PicoComponentRegistry(context: ServletContext)

MutablePicoContainer
+addComponent(componentKey: Object, componentImplementationOrInstance: Object, parameters: Parameter ...): MutablePicoContainer

ControllerDefinitionFactory

ContextControllerDefinitionFactory
+ContextControllerDefinitionFactory(methodDefinitionFinder: MethodDefinitionFinder, controllerNameResolver: ControllerNameResolver, controllerMonitor: ControllerMonitor)
```
Adapter

- adaptee: Adaptee
  + foo()

AdapterInterface
  + foo()

Adaptee
  + bar()
Adapter in Waffle

WaffleServlet
- actionMethodExecutor: ActionMethodExecutor
+ service(request: HttpServletRequest, response: HttpServletResponse)

ActionMethodExecutor
+ execute(actionMethodResponse: ActionMethodResponse, controllerDefinition: ControllerDefinition)

InterceptingActionMethodExecutor
+ execute(actionMethodResponse: ActionMethodResponse, controllerDefinition: ControllerDefinition)
- handleInvocation(controllerDefinition: ControllerDefinition): Object

MethodDefinition
- method: Method
- arguments: List
+ getMethod(): Method
+ getMethodArguments(): List

MethodInvokingMethodInterceptor
+ intercept(controllerDefinition: ControllerDefinition, method: Method, chain: InterceptorChain, arguments: Object ...): Object
DI Cures Code Smells!

- Propagating Dependency
- Long Constructor Argument List
- Container Dependency
- Tests Use Container
- Singleton