

CS520 Web Programming

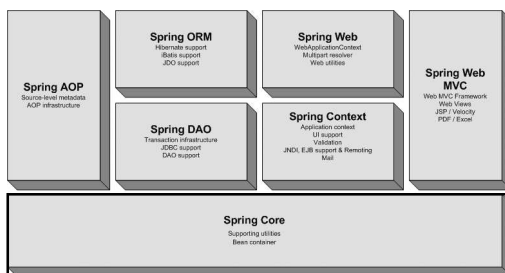
Spring – Bean Container

Chengyu Sun
California State University, Los Angeles

Background

- ◆ Developed by Rod Johnson
- ◆ Described in *Expert One-on-One: J2EE Design and Development (2002)*
- ◆ Addresses many problems of EJB
 - Overly complex
 - Invasive
 - Hard to test
 - Entity beans

Spring Framework



Bean Container

- ◆ Bean == POJO
- ◆ Container ??

Example: Hello World

- ◆ Message is a POJO managed by the Spring container
 - Created by the container
 - Property is set by the container

Bean Configuration File

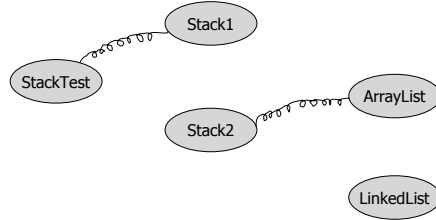
```
<beans>  
  
  <bean id="msgBean" class="rex.hello.Message">  
    <property name="message" value="Hello World!" />  
  </bean>  
  
</beans>
```

- ◆ The string "Hello World" is injected to the bean msgBean

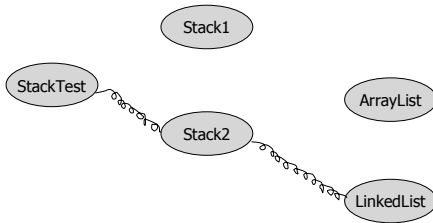
Dependency Injection

- ◆ Methods of injection
 - via Setters
 - via Constructors
- ◆ Objects that can be injected
 - Simple types: strings and numbers
 - Collection types: list, set, and maps
 - Other beans

Wiring



Wiring



Auto Wiring

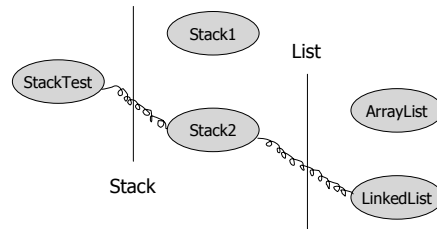
- ◆ `<bean autowire="autowire type"/>`
- ◆ `<beans default-autowire="autowire type">`
- ◆ Auto wire types
 - byName
 - byType
 - constructor
 - autodetect

Inversion of Control (IoC)

- ◆ Another name for Dependency Injection
- ◆ Usually application objects are responsible for acquiring their own dependencies
 - E.g `Stack stack = new Stack1();`
- ◆ *Inversion of control* means the container injects the dependencies into the application objects

Why IoC is Good?

- ◆ Interface vs. Implementation



More Readings

- ◆ *Spring in Action* by Craig Walls and Ryan Breidenbach
- ◆ *Professional Java Development with the Spring Framework* by Rod Johnson, Juergen Hoeller, Aref Arendsen, Thomas Risberg, and Colin Sampaleanu
- ◆ Spring documentation at <http://www.springframework.org/documentation>