Application Delivery Fundamentals 2.0: Java

Module 26: Introduction to Spring Framework

Module Objectives

At the end of this module, participants will be able to:

- Explain the advantages of the Spring Framework
- List the Spring Framework components
- Write/Modify an Application Based on Spring Core
- Describe Dependency Injection
- Identify the two distinct Spring containers:
  - BeanFactory
  - ApplicationContext

Agenda

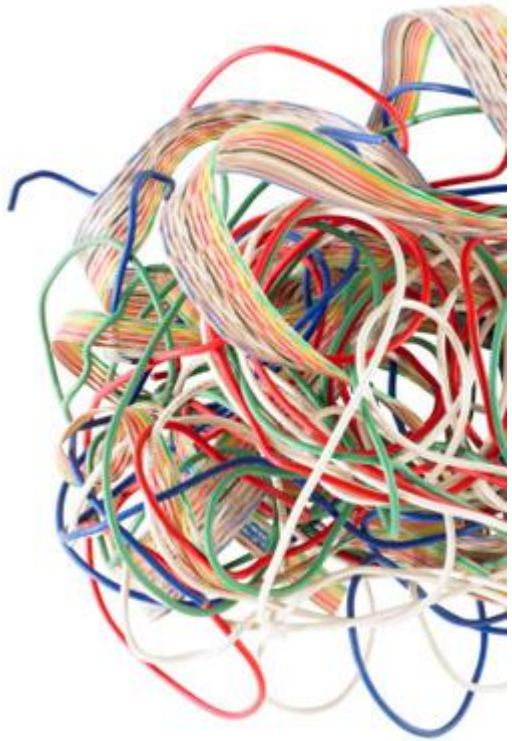
- Spring Framework Background
- Problems with Traditional Approach
- Goals of Spring Framework
- Spring Framework Components
- Spring Containers
- Spring Core
- Activity 1: Spring Core

Spring Framework Background

- Simplified development of enterprise applications in Java technologies
- Started around 2002-2003 by Rod Johnson
- Open source application framework for Java platform
- Layered architecture; allows selection of components based on requirements
- Easy way to configure and resolve dependencies using Inversion of Control (IoC)

Problems with Traditional Approach

- *Most* JavaEE applications are complex and require a lot of effort to develop.
- Specific causes of complexity and other problems in JavaEE applications:
  - Contain excessive amounts of ‘plumbing’ code
  - Difficult to unit test
  - Certain JavaEE technologies have failed in performance, for example, EJB 2.x entity beans



Goals of Spring Framework

Reduced glue code/plumbing work:

- Dependencies described in separate file (xml), rather than mixing with business logic code itself, for better control over application
- Dependencies better managed

Flexibility:

- Programmers choose modules to suit their application
- Offers integration points with several other frameworks

Spring Framework Components Overview (1 of 2)

- Spring framework consists of several components/ modules.
- Each module has a defined set of functionality.
- Each module can be used independently.
- Spring provides integration points for every module to work with other frameworks.

Spring Framework Components Overview (2 of 2)

2. Spring Co
contextual in
Context inc
internalizatio
dependency s

3. Spring AO
programming
framework,
As a result,
managed by
module prov
objects in an

4. Spring DAO: The Spring JDBC DAO abstraction layer offers a meaningful exception hierarchy for managing the exception handling and error messages thrown by different database vendors. The exception hierarchy simplifies error handling and greatly redu he

7. Spring MVC Framework: The Model-View-Controller (MVC) framework is a full-featured MVC implementation for building Web applications. The MVC framework is highly configurable via strategy interfaces. The framework accommodates numerous view technologies including JSP, Velocity, Tile and so on.

6. S
app
app
inte
task
parameters to domain objects.

Spring Framework Components
Spring Core (1 of 2)

- All Spring modules rely on core components.
- Spring Core is also referred to as an IoC container.
  - Supports Dependency Injection into Spring components through Inversion of Control mechanism
  - Provides decoupling of configuration and dependency specifics from the actual program logic

Spring Framework Components
Spring Core (2 of 2)

- Spring Core supports creation of and management of objects and other common applications services.
- Main packages include:
  - Core package: *BeanFactory*
    - Provides the basic functionality of creating beans
  - Context package: *ApplicationContext*
    - Superset of BeanFactory
    - More suitable for JavaEE applications

Spring Core Containers Overview

Spring's Container uses IoC to manage components of the application.

Spring has two distinct containers

- **Bean Factories:** (org.springframework.beans.factory.BeanFactory), provides support for Dependency Injection
- **Application contexts**: (org.springframework.context.ApplicationContext) provides application framework services

Spring Core Containers
Dependency Injection

Java classes should be as independent as possible from each other.

- To decouple classes from one another, dependencies should be injected through:
  - Constructors
  - Setters
- Spring Framework injects these dependencies via their containers.
- A class should not configure itself, IoC uses dependency injection to:
  - Configure a class correctly from outside the class
  - Wire services or components

Spring Core Containers
Configuring Beans (1 of 2)

- Piecing together all beans in the Spring Container is called **wiring**.
- Wiring can be done through xml.
- Various BeanFactories and ApplicationContext objects that support wiring are:
  - XmlBeanFactory
  - ClassPathXmlApplicationContext
  - FileSystemXmlApplicationContext
  - XmlWebApplicationContext

Spring Core Containers
Configuring Beans (2 of 2)

The beans are listed in the configuration file so that they can later be referred to by application programs.

Example:

```
<? xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE beans PUBLIC "-//SPRING//DTD BEAN// EN"
"http://www.springframework.org/dtd/spring-beans.dtd"?>
<beans>
<bean id="event"
class="com.accenture.adf.newcodington.module26.sample.Event"/>
</beans>
```

Spring Core: See It

Demonstration:

Faculty will demonstrate how to create a Spring Core application to print a greeting along with a name.

Time Allocated: 20 minutes

Environment or File: Eclipse

Steps:

1. Open the project Week2CodebaseM26_participant
2. Navigate to Java Resources/src.
3. Open package com.accenture.adf.newcodington.module26.sample.
4. Complete TODOs in
    a) HelloSeeIt.java
    b) HelloSeeItImpl.java
    c) HelloSeeItClient.java
5. Complete helloSeeIt.xml.

Spring Core: Try It

Now You Try It:

Create a Spring Core application to add two numbers and print the sum.

Time Allocated: 30 minutes

Environment or File: Eclipse

Steps:

1. Open the project Week2CodebaseM26_participant
2. Navigate to Java Resources/src.
3. Open package com.accenture.adf.newcodington.module26.sample..
4. Complete TODOs in
   a) NumberTryIt.java
   b) NumberTryItImpl.java
   c) NumberTryItClient.java
5. Complete numberTryIt.xml.

Spring Core: Solution

Your faculty will now provide you with the Solution to check and update your file.

Activity 1: Spring Core

Objective:

Write/modify an application based on Spring Core which will update and then display a record in the Zoo table.

Instructions:

- Navigate to the Module 26, Activity 1 page on the course web site.
- Follow the instructions provided on the web page to locate the codebase, launch Eclipse and complete the activity.

Module Summary

The key content points from this module are:

- Spring is a lightweight container that makes Java EE and existing technologies easy to use.
- Spring Framework has seven modules. Each module has a set of functionalities and can be used independently.
- Spring's Container uses IoC to manage components of the application.
- Spring has two distinct containers:
  - BeanFactory
  - ApplicationContext

Questions and Comments