



Spring Framework

Module 1 - Introduction

Evgeniy Krivosheev
Andrey Stukalenko
Vyacheslav Yakovenko
Last update: Feb, 2012

Spring Framework :: Introduction

- Spring is a light-weighted, but at the same time flexible and universal framework used for creating Java SE and Java EE applications;
- Spring is a framework with an open source code;
- Spring is an application framework, not a layer framework;
- Spring includes several separate frameworks;

Spring Framework :: Introduction

- Rod Johnson created Spring in 2003;
- Spring took its rise from books
Expert One-on-One Java J2EE Design and Development and *J2EE Development Without EJB*;
- The basic idea behind Spring is to simplify traditional approach to designing J2EE applications;

Spring Framework :: Introduction

- At the moment Spring Framework is developed under the sponsorship of SpringSource, which is a division of VMWare(vmware.com)

Spring Framework :: Introduction

- Main Resources:
 - springsource.com
 - blog.springsource.com
 - [@SpringSource](https://twitter.com/#!/SpringSource) (<https://twitter.com/#!/SpringSource>)
 - infoq.com/spring

Spring Framework :: Introduction

Spring helps to:

- Create logical architecture;
- Simplify designing J2EE applications;
- Combine frameworks of different levels;
- Effectively organize middle-tier objects both with and without EJB;
- Reduce number of singletons and property-files;
- Facilitate testing;
- Use various application servers / Servlet containers (WebLogic, Tomcat, Resin, JBoss, Jetty, Geronimo, WebSphere);

Spring Framework :: Introduction

Benefits of Spring:

- It is non-invasive framework: you can easily migrate to another container or higher Spring version;
- It is possible to develop without linking to framework classes and use libraries that don't use Spring;
- Application server is not required;
- It promotes code reuse;
- It facilitates OOD in J2EE applications, involves Design to Interface as well as DDD and DMD;
- It encourages you to think of an application as named services (don't confuse with SOA);
- It allows to rework service implementation without impacting the rest of the application;

Spring Framework :: Introduction

Benefits of Spring:

- It facilitates the extraction of configuration values into xml or properties files;
- POJO classes can be tested separately without a heavyweight container;
- It uses good ready-made solutions (ORM, logging abstraction, connection pool, remoting protocols, etc.);
- It simplifies using these technologies and combines them to create consistent architecture.

Spring Framework :: Introduction

Problems with the Traditional Approach to J2EE :

- EJB is overused;
 - Distribution should not be built into the basic component model;
- Many J2EE design patterns are workarounds;
- J2EE applications are hard to test;

Advantages of Traditional Approach to J2EE:

- EJB is a standard;
- Supported by many large vendors;
- Wide spreading among programmers;
- Great number of development aids;

Spring Framework :: Introduction

Comparison between EJB and Spring:

- Both Spring and EJB suit well the majority of J2EE applications
- EJB distributed transactions are better, but one can use Spring + JTA

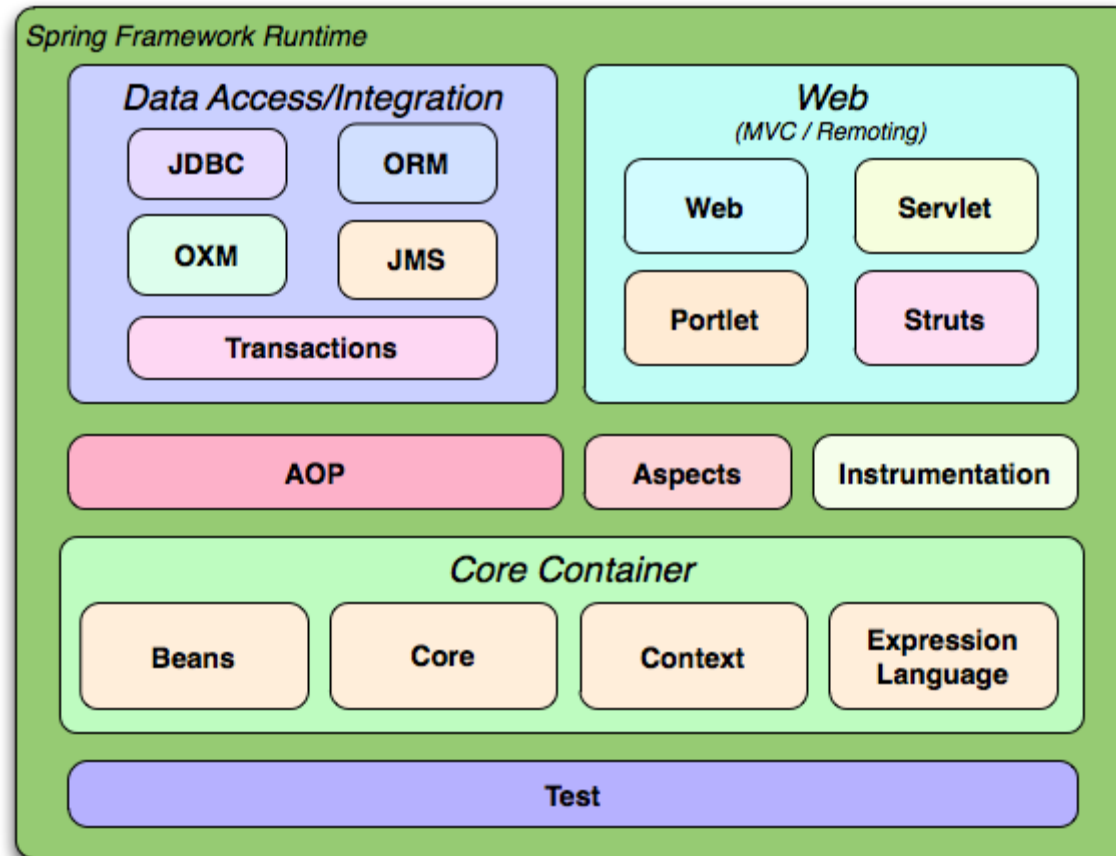
Spring Framework :: Introduction

JDK	Spring 1.2.x	Spring 2.0.x	Spring 2.5.x	Spring 3.x	Spring 3.1
1.3	Yes	Yes, but some features of JDK 5 may not be available	No	No	No
1.4	Yes	Yes, but some features of JDK 5 may not be available			No
5	N/A	Supported, but not used in full	Fuller use	Yes	Yes
6	N/A	No	Yes	Yes	Yes
7	No	No	No	No	Yes

Spring Framework :: Framework Structure

- SpringSource releases the whole range of products:
 - **Spring Framework** is the main framework providing core functionality;
 - **Spring Security** is a powerful customizable framework that facilitates authorization implementation and access control, and allows for declarative security;
 - **Spring WebServices (Spring WS)** is a framework that facilitates creating document-driven web services using SOAP;
 - **Spring Mobile** is an extension to Spring MVC (Core part) used in development of mobile web-applications;
 - **Spring Data** supports data access technologies;
 - **Spring Batch** is a framework facilitating batch processing;
 - **Spring WebFlow** is a framework aimed to enable the development of rich, process-aware web applications;
 - **Spring BlazeDS Integration** is a framework that facilitates building of RIA and integration with Adobe FLEX;
 - etc.

Spring Framework :: Framework Structure



Spring Framework :: Core

■ Core + Beans

- Inversion of Control (IoC) and Dependency Injection (DI);
- Dependencies description;
- Implementation of the IoC container;
- Removes the need for programmatic singletons;
- Allows you to decouple the configuration of dependencies from program logic;

■ Context

- Access to objects;
- Internationalization - I18N;
- Event-propagation;
- Resource-loading;

■ Expression Language

- It is a scripting language which allows access to Java components through JSP;
- It is an extension of JSP 2.1 specification;

Spring Framework :: Data Access

■ JDBC

- Wrapper facilitating work with JDBC;
- Uniform exceptions handling for all DBMS;
- DAO support (Data Access Objects);

■ ORM

- Integration with popular ORM API: Hibernate, JPA, JDO, iBatis , etc.;
- It is possible to use full functionality of relevant ORM jointly with Spring functionality (for instance, declarative transaction management);

■ Transactions

- Supports programmatic and declarative transaction management;
- Support for POJO transaction management;

■ OXM is a universal support for various implementations of Object/XML mapping (JAXB, XMLBeans, Xstream, etc.);

■ JMS is a support for Java Messaging Service

Spring Framework :: AOP

- AOP
 - Implementation of Aspect-Oriented Programming;
 - Cross-cutting feature for several classes and modules;
 - Complements OOP by enriching it with another type of modularity that make it possible to localize implementation code of cross-cutting logic in one module;
 - Can be applied to POJO;
 - Corresponds to AOP Alliance;
- Aspects – integration with AspectJ;
- Instrumentation is a set of various tools and classloader implementations for specific application servers;
- Examples:
 - Application logging;
 - Context-sensitive error handling;
 - Optimization of program execution;

Spring Framework :: Web

- Web Core
 - Multipart file upload (Servlet 3.0 specification will make adjustments in the nearest time);
 - Initialization of IoC container using servlet listeners;
 - Implementation of web-oriented application context;
- Web-Servlet
 - Model-View-Controller Implementation for web applications
- Web-Portlet
 - MVC implementation to be used in portlet environment
 - Mirrors the functionality of Web Servlet module
- Web-Struts
 - Allows for integration with web layer of classical Struts
 - It is not recommended to be used as of Spring 3.0
 - It is recommended to migrate to Struts 2.0 and its Spring integration or to Spring MVC solution

Spring Framework :: Test support

- Supports the testing Spring components with Junit or TestNG;
- It provides consistent loading of Application Context when testing;
- It provides Mock objects that can be used to test the code in isolation;

Spring Framework :: Framework Structure

- Spring in Managed environment:
 - Declarative security management;
 - Declarative transactions;
 - Database connection management;
 - Examples: JBoss, BEA WebLogic, IBM WebSphere;

- Spring in Non-managed environment:
 - Application has to track all resources on it's own
 - Examples: Jetty, Tomcat, J2SE application;

- Spring allows to work in non-managed environment like in managed, without implementing heavy weighted solutions (particularly those using JTA);

Exercises

- №1: Installing and adjusting run-time environment:
 - 15 min, if using virtual machine;
 - 45 min, if adjusting environment without assistance;
- №2: Installing and adjusting auxiliaries:
 - 30 min

Any questions!?

