

Spring 高级框架（讲师：应癡）

ssh = spring + struts + hibernate

ssm = spring + springmvc + mybatis

spring全家桶：脚手架框架springboot、微服务框架springcloud等等

主要课程内容

- Spring 概述（基本情况）
 - 核心思想 IoC 和 AOP
 - 手写实现 IoC 和 AOP（自定义spring框架）
 - Spring IoC 高级应用
 - 基础知识
 - 高级特性
 - Spring IoC 源码深度剖析
 - 设计非常优雅
 - 设计模式
 - 注意：原则、方法和技巧
 - Spring AOP 高级应用
 - 声明式事务控制
 - Spring AOP 源码深度剖析
- 必要的笔记、必要的图、通俗易懂的语言化解知识难点

第一部分 Spring 概述

第1节 Spring 简介

Spring 是分层的 full-stack（全栈）轻量级开源框架，以 IoC 和 AOP 为内核，提供了展现层 Spring MVC 和业务层事务管理等众多的企业级应用技术，还能整合开源世界众多著名的第三方框架和类库，已经成为使用最多的 Java EE 企业应用开源框架。

Spring 官方网址：<http://spring.io/>

我们经常说的 Spring 其实指的是Spring Framework（spring 框架）。

第2节 Spring 发展历程

- 1997年 IBM 提出了EJB的思想； 1998年，SUN 制定开发标准规范EJB1.0； 1999年，EJB 1.1发布； 2001年，EJB 2.0发布； 2003年，EJB 2.1发布； 2006年，EJB 3.0发布；
- Rod Johnson（spring之父）

- Expert One-to-One J2EE Design and Development(2002) 阐述了J2EE使用EJB开发设计的优点及解决方案
- Expert One-to-One J2EE Development without EJB(2004) 阐述了J2EE开发不使用EJB的解决方案（Spring雏形）

2017 年 9 月份发布了 Spring 的最新版本 Spring 5.0 通用版（GA）

第3节 Spring 的优势

整个 Spring 优势，传达出一个信号，Spring 是一个综合性，且有很强的思想性框架，每学习一天，就能体会到它的一些优势。

- **方便解耦，简化开发**

通过Spring提供的IoC容器，可以将对象间的依赖关系交由Spring进行控制，避免硬编码所造成的过度程序耦合。用户也不必再为单例模式类、属性文件解析等这些很底层的需求编写代码，可以更专注于上层的应用。

- **AOP编程的支持**

通过Spring的AOP功能，方便进行面向切面的编程，许多不容易用传统OOP实现的功能可以通过AOP轻松应付。

- **声明式事务的支持**

@Transactional

可以将我们从单调烦闷的事务管理代码中解脱出来，通过声明式方式灵活的进行事务的管理，提高开发效率和质量。

- **方便程序的测试**

可以用非容器依赖的编程方式进行几乎所有的测试工作，测试不再是昂贵的操作，而是随手可做的事情。

- **方便集成各种优秀框架**

Spring可以降低各种框架的使用难度，提供了对各种优秀框架（Struts、Hibernate、Hessian、Quartz等）的直接支持。

- **降低JavaEE API的使用难度**

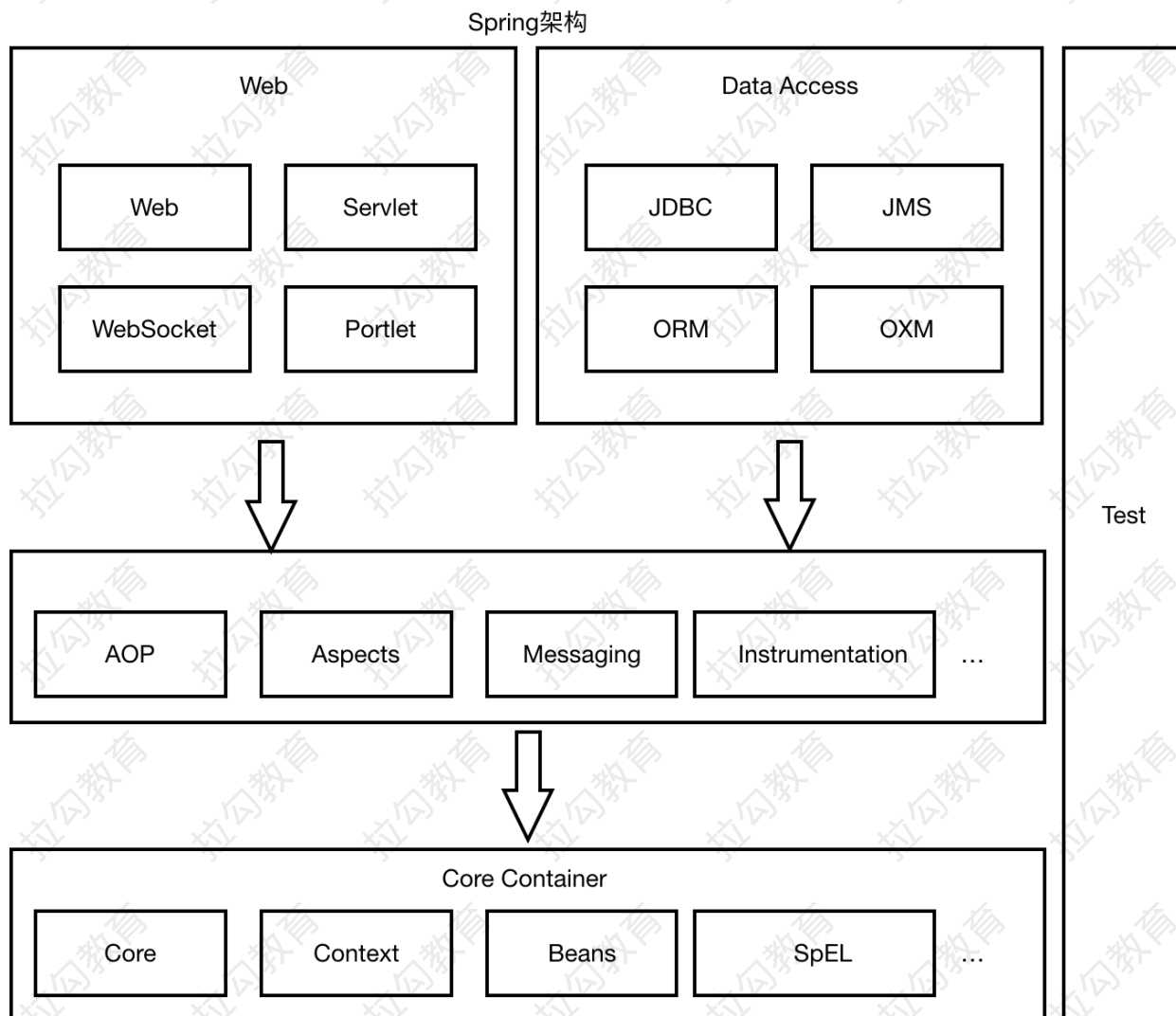
Spring对JavaEE API（如JDBC、JavaMail、远程调用等）进行了薄薄的封装层，使这些API的使用难度大为降低。

- **源码是经典的 Java 学习范例**

Spring的源代码设计精妙、结构清晰、匠心独用，处处体现着大师对Java设计模式灵活运用以及对Java技术的高深造诣。它的源代码无意是Java技术的最佳实践的范例。

第4节 Spring 的核心结构

Spring是一个分层非常清晰并且依赖关系、职责定位非常明确的轻量级框架，主要包括几个大模块：数据处理模块、Web模块、AOP（Aspect Oriented Programming）/Aspects模块、Core Container模块和 Test 模块，如下图所示，Spring依靠这些基本模块，实现了一个令人愉悦的融合了现有解决方案的零侵入的轻量级框架。



- Spring核心容器（Core Container） 容器是Spring框架最核心的部分，它管理着Spring应用中bean的创建、配置和管理。在该模块中，包括了Spring bean工厂，它为Spring提供了DI的功能。基于bean工厂，我们还会发现有多种Spring应用上下文的实现。所有的Spring模块都构建于核心容器之上。
- 面向切面编程（AOP）/Aspects Spring对面向切面编程提供了丰富的支持。这个模块是Spring应用系统中开发切面的基础，与DI一样，AOP可以帮助应用对象解耦。
- 数据访问与集成（Data Access/Integration）
Spring的JDBC和DAO模块封装了大量样板代码，这样可以使得数据库代码变得简洁，也可以更专注于我们的业务，还可以避免数据库资源释放失败而引起的问题。另外，Spring AOP为数据访问提供了事务管理服务，同时Spring还对ORM进行了集成，如Hibernate、MyBatis等。该模块由JDBC、Transactions、ORM、OXM 和 JMS 等模块组成。
- Web 该模块提供了SpringMVC框架给Web应用，还提供了多种构建和其它应用交互的远程调用方案。SpringMVC框架在Web层提升了应用的松耦合水平。
- Test 为了使得开发者能够很方便的进行测试，Spring提供了测试模块以致力于Spring应用的测试。通过该模块，Spring为使用Servlet、JNDI等编写单元测试提供了一系列的mock对象实现。

第5节 Spring 框架版本