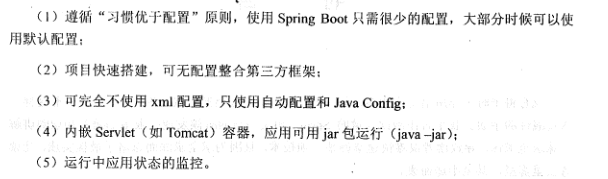
### 一、简介

1..遇到问题

1. 大量配置文件定义
2. 第三方软件整合

2.优点



#### 二、第一部分 spring4.x

1.发展

1.xml 1.x

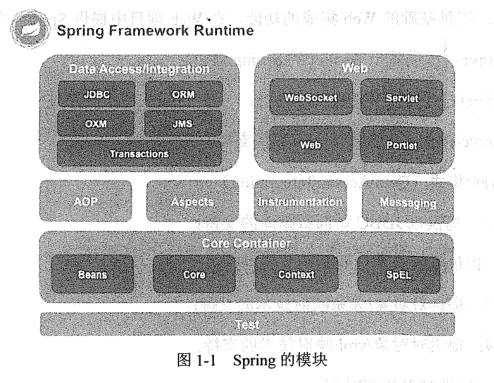
2.注解 2.x jdk5出现

1.基本配置用xml 数据库等

2.业务配置用注解

3.java配置 3.x 推荐使用

2.spring是一个轻量级的企业级开发的一站式解决方案



3.maven 项目对象模型 pom

Eclipse 使用sts插件 搭建spring

Idea 略

4.spring原则

1.使用pojo进行轻量级和最小侵入式开发

2.通过依赖注入和基于接口编程实现松耦合

3.通过aop和默认习惯进行声明式编程

4.使用aop和模板减少模块化代码

5.元数据

1.声明bean注解

[1.@Component](mailto:1.@Component)

[2.@Service](mailto:2.@Service) service

[3.@Repository](mailto:3.@Repository) dao

[4.@Controller](mailto:4.@Controller) web（mvc-springmvc使用）

2.注入bean

[1.@Autowired](mailto:1.@Autowired)

[2.@Inject](mailto:2.@Inject)

[3.@Resource](mailto:3.@Resource)

6.使用

1.编写 功能Bean 带有@声明式注解 类上

2.注入到另外一个@Service类 @Autowired 属性上等

3.配置类

@Configuration

@ComponentScan（自动扫描包下）

4.AnnotationConfigApplication(配置类.class) getBean（注入bean类） 作为spring容器

7.java配置

[1.通过@Configuration](mailto:1.通过@Configuration) 相当于xml文件

[2.@Bean](mailto:2.@Bean) 声明当前方法返回一个Bean

总结：

全局配置使用xml

业务使用注解

8.AOP面向切面编程

目的：让一组类共享相同行为

Aop注解

<http://blog.csdn.net/a352193394/article/details/7345860>

http://www.jianshu.com/p/6f40dddd71a5

步骤

1. 使用注解@Aspect来定义一个切面，在切面中定义切入点(@Pointcut),通知类型(@Before, @AfterReturning,@After,@AfterThrowing,@Around).

2. 开发需要被拦截的类。

3. 将切面配置到xml中，当然，我们也可以使用自动扫描Bean的方式。这样的话，那就交由Spring AoP容器管理。

另外需要引用 aspectJ 的 jar 包： aspectjweaver.jar aspectjrt.jar

9.bean的scope

@Scope

Singleton Prototype Request session GlobalSession

10.SpringEl和资源调用

1. @Value

http://ray-yui.iteye.com/blog/1944582

11.Bean初始化和销毁