

## 陈本布衣

可以平凡，不可以平庸；可以犯错，不可以后悔！

### 公告

### 总访问量:

1 7 0 2 4

昵称: [陈本布衣](#)

园龄: [11个月](#)

粉丝: [26](#)

关注: [16](#)

[+加关注](#)

≤	2018年8月							≥
日	一	二	三	四	五	六		
29	30	31	1	2	3	4		
5	6	7	8	9	10	11		
12	13	14	15	16	17	18		
19	20	21	22	23	24	25		
26	27	28	29	30	31	1		
2	3	4	5	6	7	8		

### 搜索

找找看

谷歌搜索

### 常用链接

[我的随笔](#)

[我的评论](#)

[我的参与](#)

[最新评论](#)

[我的标签](#)

### 随笔分类

[Java操作XML\(1\)](#)

[Java基础](#)

[Linux\(2\)](#)

[Spring框架系列\(4\)](#)

[博客园](#) :: [首页](#) :: [新随笔](#) :: [联系](#) :: [订阅](#) [XML](#) :: [管理](#)

posts - 11, comments - 30, trackbacks - 0

## Spring基础篇——Spring容器和应用上下文理解

上文说到，有了Spring之后，通过依赖注入的方式，我们的业务代码不用自己管理关联对象的生命周期。业务代码只需要按照业务本身的流程，走啊走啊，走到哪里，需要另外的对象来协助了，就给Spring说，我想要个对象——于是Spring就很贴心的给你个对象。听起来似乎很简单，使用起来也不难，但是如果仅仅是这样的拿来主义，倒也洒脱，不用费什么脑子。。。可是，你就真的不关心，Spring是从哪里把对象给你的吗？

如果你想要了解Spring深一些，而不仅仅是拿来用用，那么你就应该好好思考一下上诉问题，不然，这篇博文你还看个铲铲啊。。。你可以这样去思考：Spring既然要负责应用程序中那么多对象的创建管理，就像苹果要生产那么多的手机（对象）一样，肯定有一个专门搞对象的地方。苹果生产手机的地方叫工厂，比如富士康，但放在软件开发中，对于Spring搞对象的地方我们就不叫工厂了，而叫做容器。是的，容器的概念在java中你最熟悉的莫过于Tomcat了，它正是一个运行Servlet的web容器，而Spring要想实现依赖注入功能，就离不开对象生产的容器——如果没有容器负责对象的创建管理，你的程序代码只是喊要对象了，Spring也无处给你啊。实际上，容器是Spring框架实现功能的核心，容器不只是帮我们创建了对对象那么简单，它负责了对对象整个的生命周期的管理——创建、装配、销毁。关于Spring的这个容器你最常听闻的一个术语就是IOC容器。所谓IOC，是一种叫控制反转的编程思想，网上有很通俗易懂的总结，我就不胡乱阐述了。总之一句话，我的应用程序里不用再过问对象的创建和管理对象之间的依赖关系了，都让IOC容器给代劳吧，也就是说，我把对象创建、管理的控制权都交给Spring容器，这是一种控制权的反转，所以Spring容器才能称为IOC容器。不过这里要厘清一点：并不是说只有Spring的容器才叫IOC容器，基于IOC容器的框架还有很多，并不是Spring特有的。

好了，终于把Spring的容器概念阐述的差不多了，但有什么卵用呢？光有容器你其实什么都干不了！你以为容器那么科幻，跟叮当猫面前的百宝袋一样，你想要啥它就给你啥？实

[常用插件\(2\)](#)  
[雕虫小技](#)  
[开发环境\(1\)](#)  
[日志系列](#)  
[数据库\(1\)](#)

## 随笔档案

[2018年4月 \(2\)](#)  
[2018年3月 \(1\)](#)  
[2018年2月 \(4\)](#)  
[2018年1月 \(2\)](#)  
[2017年12月 \(2\)](#)

## 积分与排名

积分 - 13308  
排名 - 31600

## 最新评论

[1. Re:Java环境变量，真的还有必要配吗？](#)

收获颇多，尤其是想问题的思路。  
感谢

--Mr·H·

[2. Re:JAXB应用实例](#)

写得真好，学习了，感谢感谢！

--崔吕

[3. Re:Spring基础篇——Spring容器和应用上下文理解](#)

写的通俗易懂 挺好的

--xiaolin\_peter

[4. Re:Spring基础篇——Spring容器和应用上下文理解](#)

nice

--董博士的大学室友

[5. Re:拿来主义：layuiPage分页插件的使用](#)

```
实例引入页面 分页调用function
Load(cpage) {  var kssj =
$("#kssj").val();    var jssj
= $("#jssj").val();    cpage =
.....
```

--HuiSky

## 阅读排行榜

际上，容器里面什么都没有，决定容器里面放什么对象的是我们自己，决定对象之间的依赖关系的，也是我们自己，容器只是给我们提供一个管理对象的空间而已。那么，我们怎么向容器中放入我们需要容器代为管理的对象呢？这就涉及到Spring的应用上下文了。什么是应用上下文呢，你可以简单的理解成就是将你需要Spring帮你管理的对象放入容器的那么一种。。一种。。额。。一种容器对象——是的，应用上下文即是Spring容器抽象的一种实现；而我们常见的ApplicationContext本质上说就是一个维护Bean定义以及对象之间协作关系的高级接口。额，听起来是不是很抽象拗口？那你再读一遍呢。。。这里，我们必须明确，Spring的核心是容器，而容器并不唯一，框架本身就提供了很多个容器的实现，大概分为两种类型：一种是不常用的BeanFactory，这是最简单的容器，只能提供基本的DI功能；还有一种就是继承了BeanFactory后派生而来的应用上下文，其抽象接口也就是我们上面提到的ApplicationContext，它能提供更多企业级的服务，例如解析配置文本信息等等，这也是应用上下文实例对象最常见的应用场景。有了上下文对象，我们就能向容器注册需要Spring管理的对象了。对于上下文抽象接口，Spring也为我们提供了多种类型的容器实现，供我们在不同的应用场景选择——

① AnnotationConfigApplicationContext:从一个或多个基于java的配置类中加载上下文定义，适用于java注解的方式；

② ClassPathXmlApplicationContext:从类路径下的一个或多个xml配置文件中加载上下文定义，适用于xml配置的方式；

③ FileSystemXmlApplicationContext:从文件系统下的一个或多个xml配置文件中加载上下文定义，也就是说系统盘符中加载xml配置文件；

④ AnnotationConfigWebApplicationContext:专门为web应用准备的，适用于注解方式；

⑤ XmlWebApplicationContext:从web应用下的一个或多个xml配置文件加载上下文定义，适用于xml配置方式。

有了以上理解，问题就很好办了。你只要将你需要IOC容器替你管理的对象基于xml也罢，java注解也好，总之你要将需要管理的对象（Spring中我们都称之为bean）、bean之间的协作关系配置好，然后利用应用上下文对象加载进我们的Spring容器，容器就能为你的程序提供你想要的对象管理服务了。下面，还是贴一下简单的应用上下文的应用实例：

我们先采用xml配置的方式配置bean和建立bean之间的协作关系：

1. [Spring基础篇——Spring容器和应用上下文理解\(6603\)](#)
2. [JAXB应用实例\(2713\)](#)
3. [Spring基础篇——通过Java注解和XML配置装配bean\(1526\)](#)
4. [Java环境变量，真的还有必要配吗? \(1316\)](#)
5. [SQLite 带你入门\(911\)](#)

#### 评论排行榜

1. [Spring基础篇——DI/IOC和AOP原理初识\(8\)](#)
2. [SQLite 带你入门\(7\)](#)
3. [Java环境变量，真的还有必要配吗? \(4\)](#)
4. [拿来主义：layPage分页插件的使用\(3\)](#)
5. [Spring基础篇——通过Java注解和XML配置装配bean\(3\)](#)

#### 推荐排行榜

1. [Spring基础篇——通过Java注解和XML配置装配bean\(3\)](#)
2. [Spring基础篇——DI/IOC和AOP原理初识\(3\)](#)
3. [布衣之路\(二\)：虚拟机的文件共享和网络模式\(2\)](#)
4. [Java环境变量，真的还有必要配吗? \(2\)](#)
5. [拿来主义：layPage分页插件的使用\(2\)](#)



```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
                           http://www.springframework.org/schema/beans/spring-beans-3.2.xsd">
  <bean id="man" class="spring.chapter1.domain.Man">
    <constructor-arg ref="qqCar" />
  </bean>
  <bean id="qqCar" class="spring.chapter1.domain.QQCar"/>
</beans>
```



然后通过应用上下文将配置加载到IOC容器，让Spring替我们管理对象，待我们需要使用对象的时候，再从容器中获取bean就ok了：



```
public class Test {
    public static void main(String[] args) {
        //加载项目中的spring配置文件到容器
        // ApplicationContext context = new
        ClassPathXmlApplicationContext("resources/applicationContext.xml");
        //加载系统盘中的配置文件到容器
        ApplicationContext context = new
        FileSystemXmlApplicationContext("E:/Spring/applicationContext.xml");
        //从容器中获取对象实例
```

```
Man man = context.getBean(Man.class);  
man.driveCar();  
  
}  
  
}
```



以上测试中，我将配置文件applicationContext.xml分别放在项目中和任意的系统盘符下，我只需要使用相应的上下文对象去加载配置文件，最后的结果是完全一样的。当然，现在项目中越来越多的使用java注解，所以注解的方式必不可少：



//同xml一样描述bean以及bean之间的依赖关系

```
@Configuration  
  
public class ManConfig {  
  
    @Bean  
    public Man man() {  
        return new Man(car());  
    }  
  
    @Bean  
    public Car car() {  
        return new QQCar();  
    }  
  
}
```



```
public class Test {  
    public static void main(String[] args) {  
        //从java注解的配置中加载配置到容器  
        ApplicationContext context = new AnnotationConfigApplicationContext(ManConfig.class);  
        //从容器中获取对象实例  
        Man man = context.getBean(Man.class);  
        man.driveCar();  
    }  
}
```



自此，Spring容器和应用上下文就算阐述的差不多了，具体的技能点在今后的博文中会慢慢的给大家奉上。博主功力也不深厚，博文有阐述欠妥或者不准确的地方，欢迎大神大仙大侠们指正，陈某不胜感激。

作者: [陈本布衣](#)

出处: <http://www.cnblogs.com/chenbenbuyi>

本文版权归作者和博客园共有，欢迎转载分享，但必须保留此段声明，且在文章页面明显位置给出原文链接，否则保留追究法律责任的权利。

分类: [Spring框架系列](#)

好文要顶

关注我

收藏该文



陈本布衣

[关注 - 16](#)

[粉丝 - 26](#)

[+加关注](#)

1

0

« 上一篇: [Spring基础篇——DI/IOC和AOP原理初识](#)

» 下一篇: [JAXB应用实例](#)

posted on 2018-01-04 09:20 [陈本布衣](#) 阅读(6603) 评论(3) [编辑](#) [收藏](#)

## Feedback

[#1楼](#)

2018-02-28 08:11 | [TINGBUQING.COM](#)

很好, 简单易懂

支持(0) 反对(0)

[#2楼](#)

2018-06-22 14:29 | [董博士的大学室友](#)

nice

支持(0) 反对(0)

[#3楼](#)

2018-06-24 16:32 | [XIAOLIN PETER](#)

写的通俗易懂 挺好的

支持(0) 反对(0)

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论, 请 [登录](#) 或 [注册](#), [访问网站首页](#)。

**最新IT新闻:**

- [前微软工程师为勒索软件服务而被判入狱](#)
  - [红芯浏览器成多家政府部门标配？国资委、中车否认](#)
  - [恒大向中科院6个项目投16.5亿元 包括全球最快超算](#)
  - [号称“站在巨人肩膀上创新”的红芯浏览器，创始人履历也造假](#)
  - [微软开源驱动程序模块框架 让开发者轻松编写Windows驱动程序](#)
- » [更多新闻...](#)

**最新知识库文章:**

- [被踢出去的用户](#)
  - [成为一个有目标的学习者](#)
  - [历史转折中的“杭派工程师”](#)
  - [如何提高代码质量？](#)
  - [在腾讯的八年，我的职业思考](#)
- » [更多知识库文章...](#)

Powered by:

[博客园](#)

Copyright ©2018 陈本布衣