

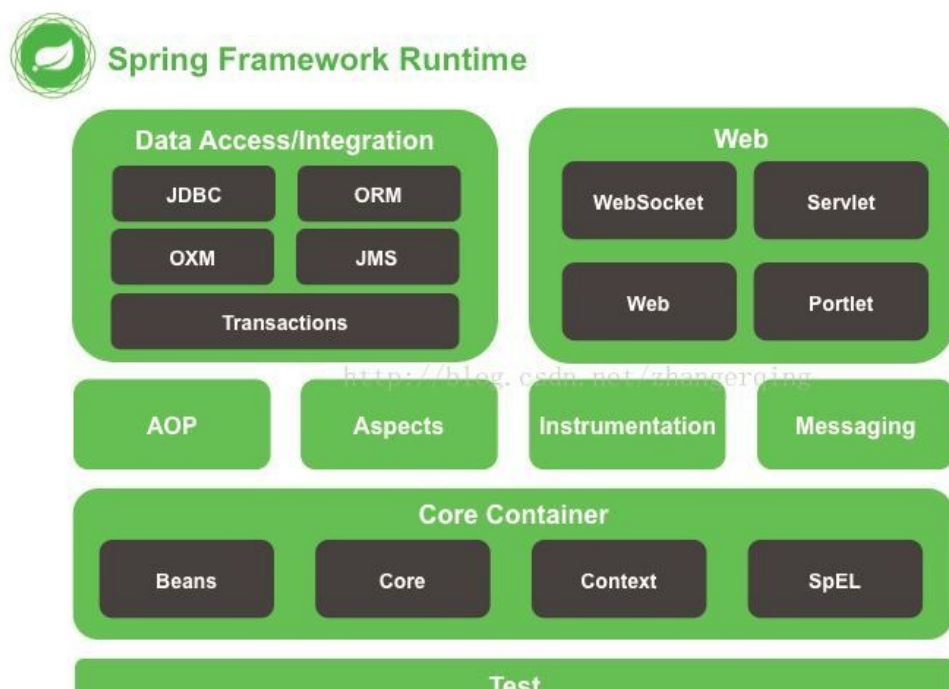
Java之美[从菜鸟到高手演变]之spring框架初识

初识轻量级Java开源框架 --- Spring

作者: egg

spring是一个轻量级Java框架，其核心思想就是DI（Dependency Injection，即依赖注入）和IoC（Inversion of Control，即控制反转），因为其开源、低侵入性，现在已经席卷了很大一部分市场，其最大竞争对手乃是JavaEE框架EJB。EJB3.0以前，由于其笨重以及使用繁琐，被人们无奈弃用，取而代之的就是spring，但是spring也有它的局限性，就是spring对分布式的支持不好，但是EJB在这方面很有优势，而且当前EJB3.0简化了操作，使用不再繁琐，竞争力进一步提升。所以，当前轻量级的JavaWeb项目，很多都是使用的spring，我们有必要把它学好！本章系[Java之美\[从菜鸟到高手演变\]系列](#)之spring框架初识，希望大家和我一起，再次回顾下Spring的基础知识，积极提出宝贵的意见和建议！本次学习使用的是Spring4.1 版本。

Modules



从这幅图中我们可以看出来Spring的整体架构，主要分为六大模块

Core Container

核心部分分为4大块，spring-core, spring-beans, spring-context, spring-expression. 其中core和bean是整个框架的核心，提供了基础的DI和IoC功能。Context建立在core和beans模块之上，提供一种类似JNDI且以框架的方式来操作对象的方式。Context模块从beans模块继承它的功能同时增加了国际化支持，如资源绑定等，同时，Context模块也支持JavaEE功能，如EJB，JMX和基本的远程调用。ApplicationContext接口是context模块的焦点。expression是一种很强大的expression language，支持在运行时查询和操作对象的属性，我们会在后面的文章中举些例子来说明spring expression language的用法。

AOP and instrumentation

Aop模块提供了面向切面编程的实现，和AspectJ集成。

Messaging

Messaging是spring4新增加的模块，包含了一部分主要的基于message的应用的实现。

Data Access/Integration

Data access顾名思义，是spring对数据层提供的支持，是功能比较丰富的模块。提供了包括JDBC，事物，ORM，JMS等一系列实现。

Web

Web模块主要提供面向web的一些实现，例如多文件上传，servlet监听器以及spring mvc方面的支持。

Test

Test模块主要是针对spring的各个模块做各种各样的测试，包括单元测试、集成测试等等。

关于各个模块具体的解释及作用我们会在后面的文章中一一提到，也请有兴趣的读者自行去查阅spring官方文档，获得更详细的说明。

第一个Spring小程序

新建一个User类，一个UserDao类以及一个测试类：

User.java

```
package com.adam.java.spring.po;

public class User {
    private String name;
    private int age;
    private int score;
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public int getAge() {
        return age;
    }
    public void setAge(int age) {
        this.age = age;
    }
    public int getScore() {
        return score;
    }
    public void setScore(int score) {
        this.score = score;
    }

    public String toString(){
        return "toString from user";
    }
}
```

UserDao.java

```

package com.adam.java.spring.dao;

import com.adam.java.spring.po.User;

public class UserDao {
    private User user;

    public void add(){
        System.out.println("add from userdao");
        System.out.println(user.toString());
    }

    public User getUser() {
        return user;
    }

    public void setUser(User user) {
        this.user = user;
    }

}

```

DITest.java

```

package com.adam.java.spring;

import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;

import com.adam.java.spring.dao.UserDao;

public class DITest {

    @SuppressWarnings("resource")
    public static void main(String[] args) {
        ApplicationContext atx = new ClassPathXmlApplicationContext("beans.xml");
        UserDao userDao = (UserDao) atx.getBean("userDao");
        userDao.add();
    }

}

```

一个配置文件beans.xml

```

<?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.xsd">

    <bean id="user" class="com.adam.java.spring.po.User"/>
    <bean id="userDao" class="com.adam.java.spring.dao.UserDao">
        <property name="user" ref="user"/>
    </bean>
</beans>

```

执行测试类，得出如下输出：

add from userdao
tostring from user