



1.Spring框架是什么

2.Spring框架优点

- 2.1.轻量级
- 2.2.针对接口编程，解耦合
- 2.3.AOP 编程的支持
- 2.4.声明式事务的支持
- 2.5.方便集成各种优秀框架
- 2.6.降低JavaEE API的使用难度

3.Spring框架体系结构

- 3.1.核心容器
- 3.2.AOP和Instrumentation
- 3.3.消息
- 3.4.数据访问/集成
- 3.5.Web（也就是Spring MVC框架）
- 3.6.测试

4.Spring版本与下载

- 4.1.Spring版本
- 4.2.下载地址

1.Spring框架是什么

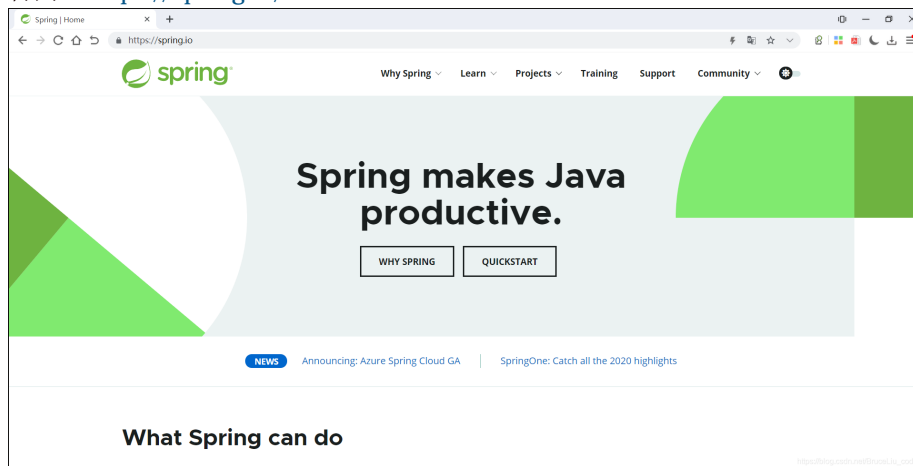


Spring是一个开源框架，Spring是于2003 年兴起的一个轻量级的Java 开发框架，由Rod Johnson 在其著作Expert One-On-One J2EE Development and Design中阐述的部分理念和原型衍生而来。它是为了解决企业应用开发的复杂性而创建的。框架的主要优势之一就是其分层架构，分层架构允许使用者选择使用哪一个组件，同时为 J2EE 应用程序开发提供集成的框架。Spring使用基本的JavaBean来完成以前只可能由EJB完成的事情。然而，Spring的用途不仅限于服务器端的开发。从简单性、可测试性和松耦合的角度而言，任何Java应用都可以从Spring中受益。Spring的核心是控制反转（IoC）和面向切面（AOP）。简单来说，Spring是一个分层的JavaSE/EE full-stack(一站式) 轻量级开源框架。

Spring的主要作用就是为代码“解耦”，降低代码间的耦合度。就是让对象和对象（模块和模块）之间关系不是使用代码关联，而是通过配置来说明。即在Spring中说明对象（模块）的关系。

Spring根据代码的功能特点，使用Ioc降低业务对象之间耦合度。IoC使得主业务在相互调用过程中，不用再自己维护关系了，即不用再自己创建要使用的对象了。而是由Spring容器统一管理，自动“注入”，注入即赋值。而AOP使得系统级服务得到了最大复用，且不用再由程序员手工将系统级服务“混杂”到主业务逻辑中了，而是由Spring容器统一完成“织入”。

官网：<https://spring.io/>



2.Spring框架优点

Spring 是一个框架，是一个半成品的软件。有 20个模块组成。它是一个容器管理对象，容器是装东西的，Spring 容器不装文本，数字。装的是对象。Spring 是存储对象的容器。

2.1.轻量级

Spring 框架使用的 jar 都比较小，一般在 1M 以下或者几百 kb。Spring 核心功能的所需的 jar 总共在 3M 左右。

Spring 框架运行占用的资源少，运行效率高。不依赖其他 jar

2.2.针对接口编程，解耦合

Spring 提供了 Ioc 控制反转，由容器管理对象，对象的依赖关系。原来在程序代码中的对象创建方式，现在由容器完成。对象之间的依赖解耦合。

2.3.AOP 编程的支持

通过 Spring 提供的 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

在Spring 中，开发人员可以从繁杂的事务管理代码中解脱出来，通过声明式方式灵活地进行事务的管理，提高开发效率和质量。

2.4.声明式事务的支持

只需要通过配置就可以完成对事务的管理，而无需手动编程。

■ *SSM SSH 项目中事务管理 不需要人工处理 Spring 声明式事务！*

2.5.方便集成各种优秀框架

Spring 不排斥各种优秀的开源框架，相反 Spring 可以降低各种框架的使用难度，Spring提供了对各种优秀框架（如 Struts,Hibernate、MyBatis）等的直接支持。简化框架的使用。

Spring 像插线板一样，其他框架是插头，可以容易的组合到一起。需要使用哪个框架，就把这个插头放入插线板。不需要可以轻易的移除。

■ *SSM 框架整合 Redis RabbitMQ Solr ES Quartz...*

2.6.降低JavaEE API的使用难度

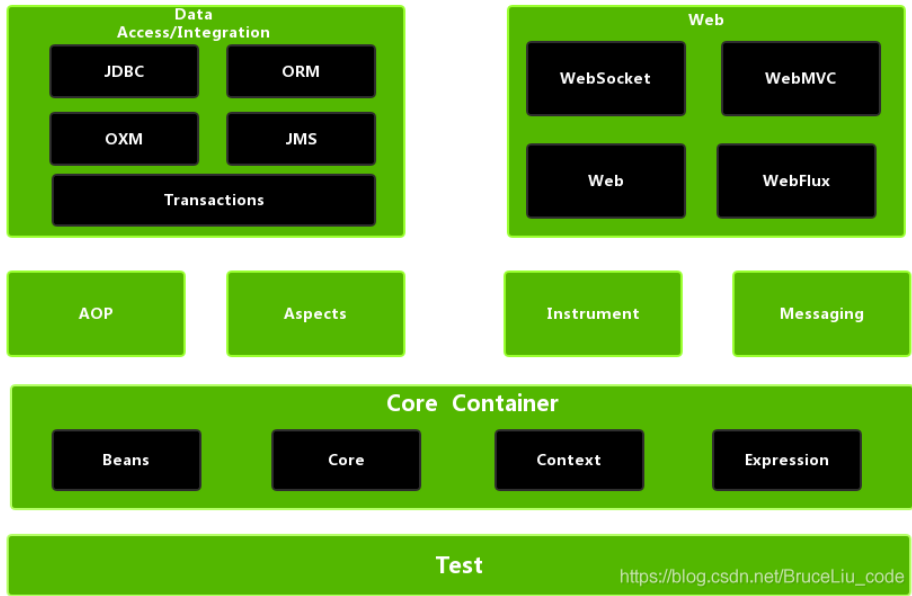
Spring对JavaEE开发中非常难用的一些API（JDBC、JavaMail、远程调用等），都提供了封装，使这些API应用难度大大降低。

3.Spring框架体系结构

Spring框架至今已集成了20多个模块，这些模块分布在以下模块中：

- 核心容器（Core Container）
- 数据访问/集成（Data Access/Integration）层
- Web层
- AOP（Aspect Oriented Programming）模块
- 植入（Instrumentation）模块
- 消息传输（Messaging）
- 测试（Test）模块

Spring体系结构如下图：



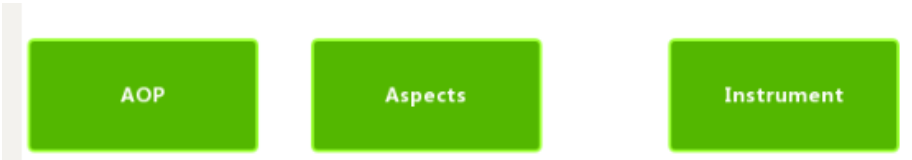
3.1.核心容器



Spring的核心容器是其他模块建立的基础，有Spring-core、Spring-beans、Spring-context、Spring-context-support和Spring-expression（String表达式语言）等模块组成。

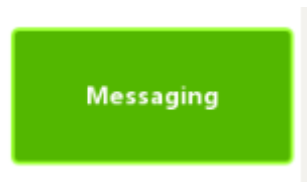
- **Spring-core**模块：提供了框架的基本组成部分，包括控制反转（Inversion of Control, IOC）和依赖注入（Dependency Injection, DI）功能。
- **Spring-beans**模块：提供了BeanFactory，是工厂模式的一个经典实现，Spring将管理对象称为Bean。
- **Spring-context**模块：建立在Core和Beans模块的基础之上，提供一个框架式的bean对象访问方式，是访问定义和配置的任何对象的媒介。ApplicationContext接口是Context模块的焦点。
- **Spring-context-support**模块：支持整合第三方库到Spring应用程序上下文，特别是用于高速缓存（EhCache、JCache）和任务调度（CommonJ、Quartz）的支持。
- **Spring-expression**模块：提供了强大的表达式语言去支持运行时查询和操作对象图。这是对JSP2.1规范中规定的统一表达式语言（Unified EL）的扩展。该语言支持设置和获取属性值、属性分配、方法调用、访问数组、集合和索引器的内容、逻辑和算术运算、变量命名以及从Spring的IOC容器中以名称检索对象。它还支持列表投影、选择以及常用的列表聚合。

3.2.AOP和Instrumentation



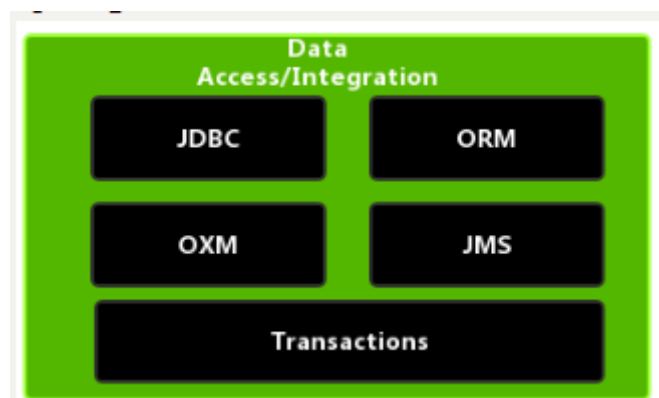
- **Spring-aop模块**：提供了一个符合AOP要求的面向切面的编程实现，允许定义方法拦截器和切入点，将代码按照功能进行分离，以便干净地解耦。
- **Spring-aspects模块**：提供了与AspectJ的集成功能，AspectJ是一个功能强大且成熟的AOP框架。
- **Spring-instrument模块**：提供了类植入（Instrumentation）支持和类加载器的实现，可以在特定的应用服务器中使用。

3.3.消息



Spring4.0以后新增了消息（Spring-messaging）模块，该模块提供了对消息传递体系结构和协议的支持。

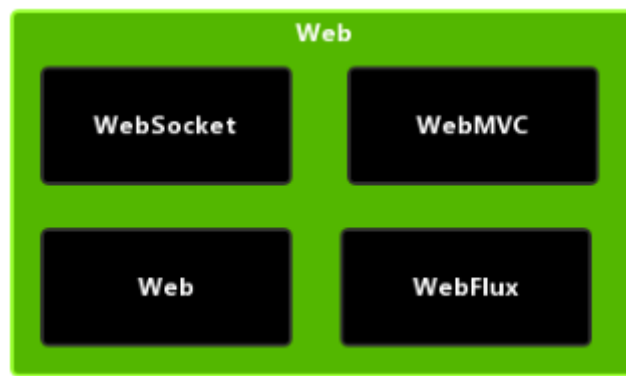
3.4.数据访问/集成



数据访问/集成层由JDBC、ORM、OXM、JMS和事务模块组成。

- **Spring-jdbc模块**：提供了一个JDBC的抽象层，消除了烦琐的JDBC编码和数据库厂商特有的错误代码解析。
- **Spring-orm模块**：为流行的对象关系映射（Object-Relational Mapping）API提供集成层，包括JPA和Hibernate。使用Spring-orm模块可以将这些O/R映射框架与Spring提供的所有其他功能结合使用，例如声明式事务管理功能。
- **Spring-oxm模块**：提供了一个支持对象/XML映射的抽象层实现，例如JAXB、Castor、JiBX和XStream。
- **Spring-jms模块（Java Messaging Service）**：指Java消息传递服务，包含用于生产和使用消息的功能。自Spring4.1以后，提供了与Spring-messaging模块的集成。
- **Spring-tx模块（事务模块）**：支持用于实现特殊接口和所有POJO（普通Java对象）类的编程和声明式事务管理。

3.5.Web（也就是Spring MVC框架）



Web层由Spring-web、Spring-webmvc、Spring-websocket和Portlet模块组成。

- **Spring-web**模块：提供了基本的Web开发集成功能，例如多文件上传功能、使用Servlet监听器初始化一个IOC容器以及Web应用上下文。
- **Spring-webmvc**模块：也称为Servlet模块，包含用于web应用程序的Spring MVC和REST Web Services实现。Spring MVC框架提供了领域模型代码和Web表单之间的清晰分离，并与Spring Framework的所有其他功能集成。
- **Spring-websocket**模块：Spring4.0以后新增的模块，它提供了WebSocket和SocketJS的实现。
- **WebFlux**模块：也称为类Portlet，似于Servlet模块的功能，提供了Portlet环境下的MVC实现。

3.6.测试



Spring-test模块支持使用JUnit或TestNG对Spring组件进行单元测试和集成测试。

4.Spring版本与下载

4.1.Spring版本

地址:<https://spring.io/projects/spring-framework#learn>

Spring Boot

Spring Framework

Spring Data

Spring Cloud

Spring Cloud Data Flow

Spring Security

Spring Session

Spring Integration

Spring HATEOAS

Spring REST Docs

Spring Batch

Spring AMQP

Spring for Android

Spring CredHub

Spring Flo

Spring for Apache Kafka

Spring Framework 5.3.2

OVERVIEW

LEARN

Documentation

Each **Spring project** has its own; it explains in great details how you can use **project features** and what you can achieve with them.

5.3.2	CURRENT	GA	Reference Doc.	API Doc.
5.3.3-SNAPSHOT	SNAPSHOT		Reference Doc.	API Doc.
5.2.13.BUILD-SNAPSHOT		GA	Reference Doc.	API Doc.
5.2.12.RELEASE		GA	Reference Doc.	API Doc.
5.1.20.RELEASE		GA	Reference Doc.	API Doc.
5.0.20.RELEASE		GA	Reference Doc.	API Doc.
4.3.30.RELEASE		GA	Reference Doc.	API Doc.

4.2.下载地址

<https://repo.spring.io/release/org/springframework/spring/>

Index of release/org/springfram: X

←

→

↻

🏠

🔍

https://repo.spring.io/release/org/springframework/spring/

5.0.17.RELEASE/	28-Apr-2020 09:08	-
5.0.18.RELEASE/	21-Jul-2020 07:05	-
5.0.19.RELEASE/	15-Sep-2020 09:41	-
5.0.2.RELEASE/	30-Sep-2019 11:06	-
5.0.20.RELEASE/	09-Dec-2020 08:10	-
5.0.3.RELEASE/	30-Sep-2019 11:06	-
5.0.4.RELEASE/	30-Sep-2019 11:06	-
5.0.5.RELEASE/	30-Sep-2019 11:06	-
5.0.6.RELEASE/	30-Sep-2019 11:06	-
5.0.7.RELEASE/	30-Sep-2019 11:06	-
5.0.8.RELEASE/	30-Sep-2019 11:06	-
5.0.9.RELEASE/	30-Sep-2019 11:06	-
5.1.0.RELEASE/	30-Sep-2019 11:06	-
5.1.1.RELEASE/	30-Sep-2019 11:06	-
5.1.10.RELEASE/	30-Sep-2019 11:06	-
5.1.11.RELEASE/	02-Nov-2019 07:16	-
5.1.12.RELEASE/	03-Dec-2019 08:05	-
5.1.13.RELEASE/	14-Jan-2020 07:37	-
5.1.14.RELEASE/	25-Feb-2020 16:22	-
5.1.15.RELEASE/	28-Apr-2020 07:29	-
5.1.16.RELEASE/	09-Jun-2020 06:04	-
5.1.17.RELEASE/	21-Jul-2020 07:09	-
5.1.18.RELEASE/	15-Sep-2020 09:02	-
5.1.19.RELEASE/	27-Oct-2020 13:26	-
5.1.2.RELEASE/	30-Sep-2019 11:06	-
5.1.20.RELEASE/	09-Dec-2020 07:22	-
5.1.3.RELEASE/	30-Sep-2019 11:06	-
5.1.4.RELEASE/	30-Sep-2019 11:06	-
5.1.5.RELEASE/	30-Sep-2019 11:06	-
5.1.6.RELEASE/	30-Sep-2019 11:06	-
5.1.7.RELEASE/	30-Sep-2019 11:06	-
5.1.8.RELEASE/	30-Sep-2019 11:06	-
5.1.9.RELEASE/	30-Sep-2019 11:06	-
5.2.0.RELEASE/	30-Sep-2019 09:06	-
5.2.1.RELEASE/	02-Nov-2019 08:19	-
5.2.10.RELEASE/	27-Oct-2020 13:48	-
5.2.11.RELEASE/	10-Nov-2020 07:09	-

<https://repo.spring.io/release/org/springframework/spring/>