spring
by Pivotal™

Plan

- Annotations
- Reflection
- IoC Container
- Spring Framework (Overview)

Reflection

- Reflection is commonly used by programs which require the ability to examine or modify the runtime behavior of applications running in the Java virtual machine.
- This feature is extremely powerful and has no equivalent in other conventional languages such as C, C++, Fortran, or Pascal.

```java
Method[] methods = Object.class.getMethods();
for(Method method : methods){
	System.out.println("method = " + method.getName());
}
```

Annotation

Annotations is a form of syntactic metadata. Annotations have no direct effect on the operation of the code they annotate.

- **Information for the compiler** — Annotations can be used by the compiler to detect errors or suppress warnings.
- **Compile-time and deployment-time processing** — Software tools can process annotation information to generate code, XML files, and so forth.
- **Runtime processing** — Some annotations are available to be examined at runtime.

- Built-in annotations:
  - @Override
  - @Deprecated
  - @SuppressWarnings
  - @Retention
  - @Target
  - …

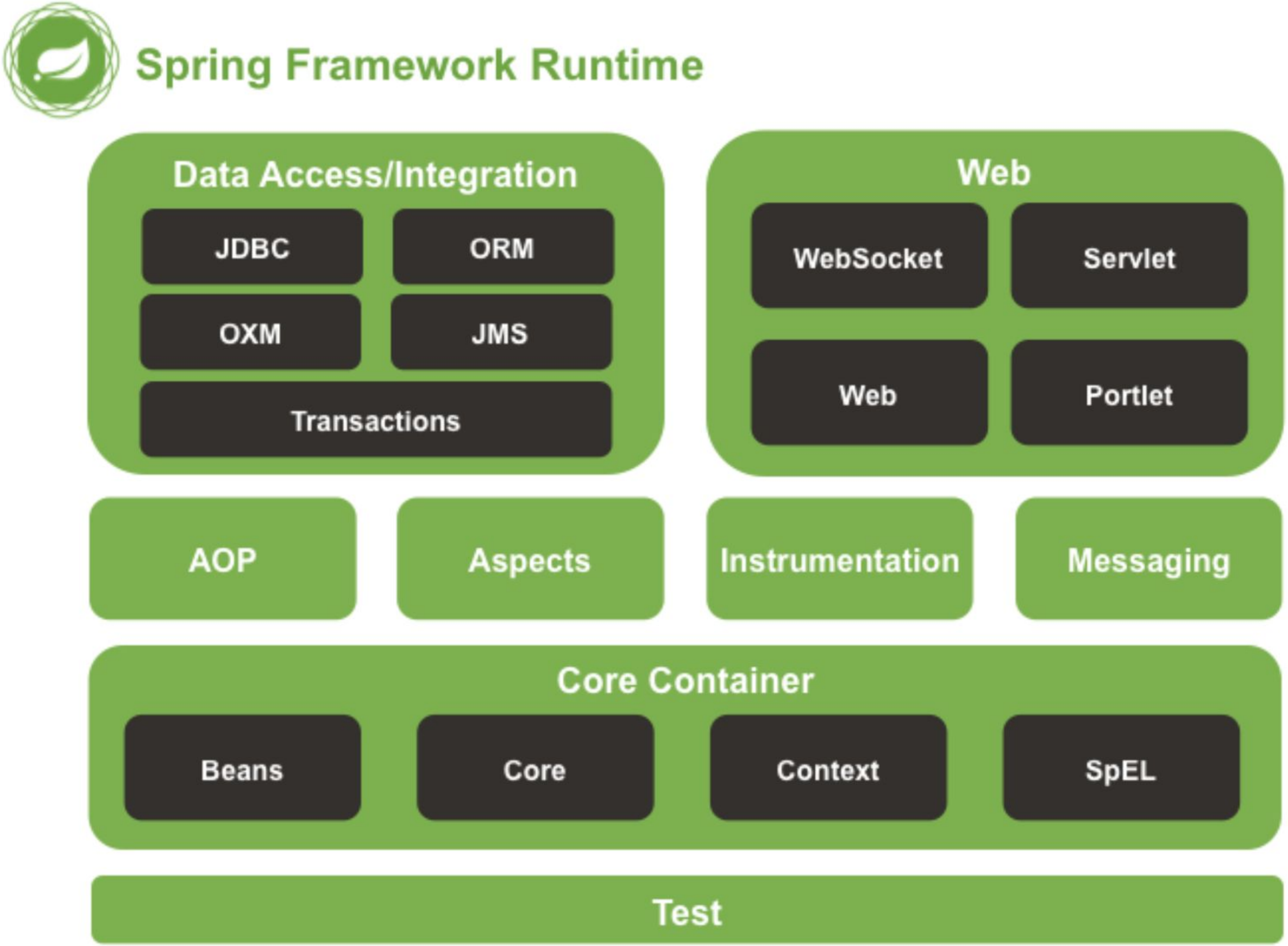
For example

```
@Target(ElementType.FIELD)
@Retention(RetentionPolicy.RUNTIME)
public @interface Qualifier {

   String name();
}
```

Spring Framework

The Spring Framework is a Java platform that provides comprehensive infrastructure support for developing Java applications.



Spring Framework Runtime

Data Access/Integration

JDBC

ORM

OXM

JMS

Transactions

Web

WebSocket

Servlet

Web

Portlet

AOP

Aspects

Instrumentation

Messaging

Core Container

Beans

Core

Context

SpEL

Test

Core Container

The Core consists of the spring-core, spring-beans, spring-context, spring- context-support, and spring-expression (Spring Expression Language) modules.

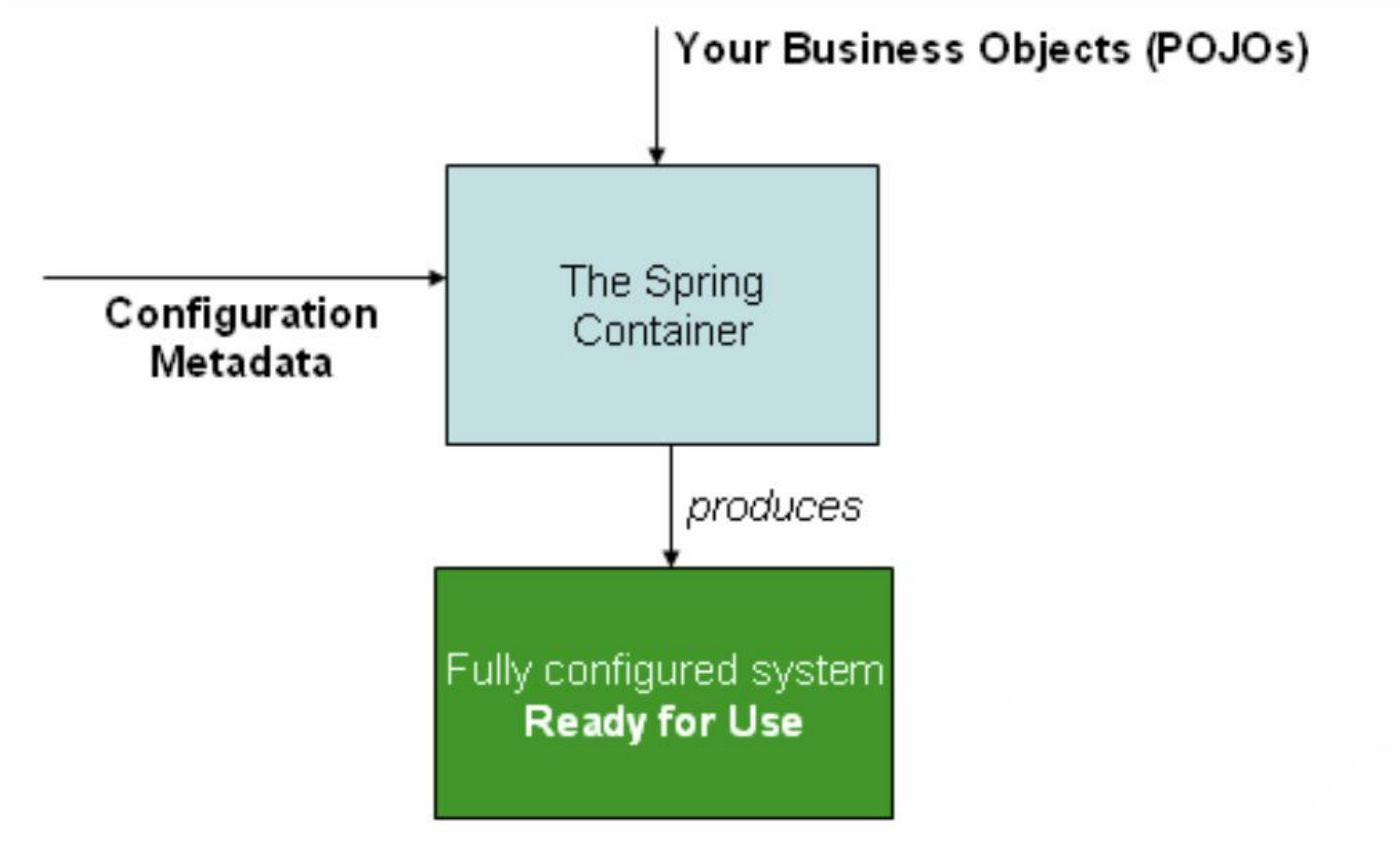
- The spring-core and spring-beans modules provide the fundamental parts of the framework, including the IoC and Dependency Injection features.
- The spring-context module builds on the solid base provided by the Core and Beans modules: it is a means to access objects in a framework-style manner that is similar to a JNDI registry.

Container overview

The interface *org.springframework.context.ApplicationContext* represents the Spring IoC container and is responsible for instantiating, configuring, and assembling a beans.

The container gets its instructions on what objects to instantiate, configure, and assemble by reading configuration metadata.

A Spring IoC container manages one or more beans. These beans are created with the configuration metadata that you supply to the container, for example, in the form of XML <bean/> definitions.


Your Business Objects (POJOs)
Configuration
Metadata
The Spring
Container
produces
Fully configured system
Ready for Use