



Guide To Spring Modules

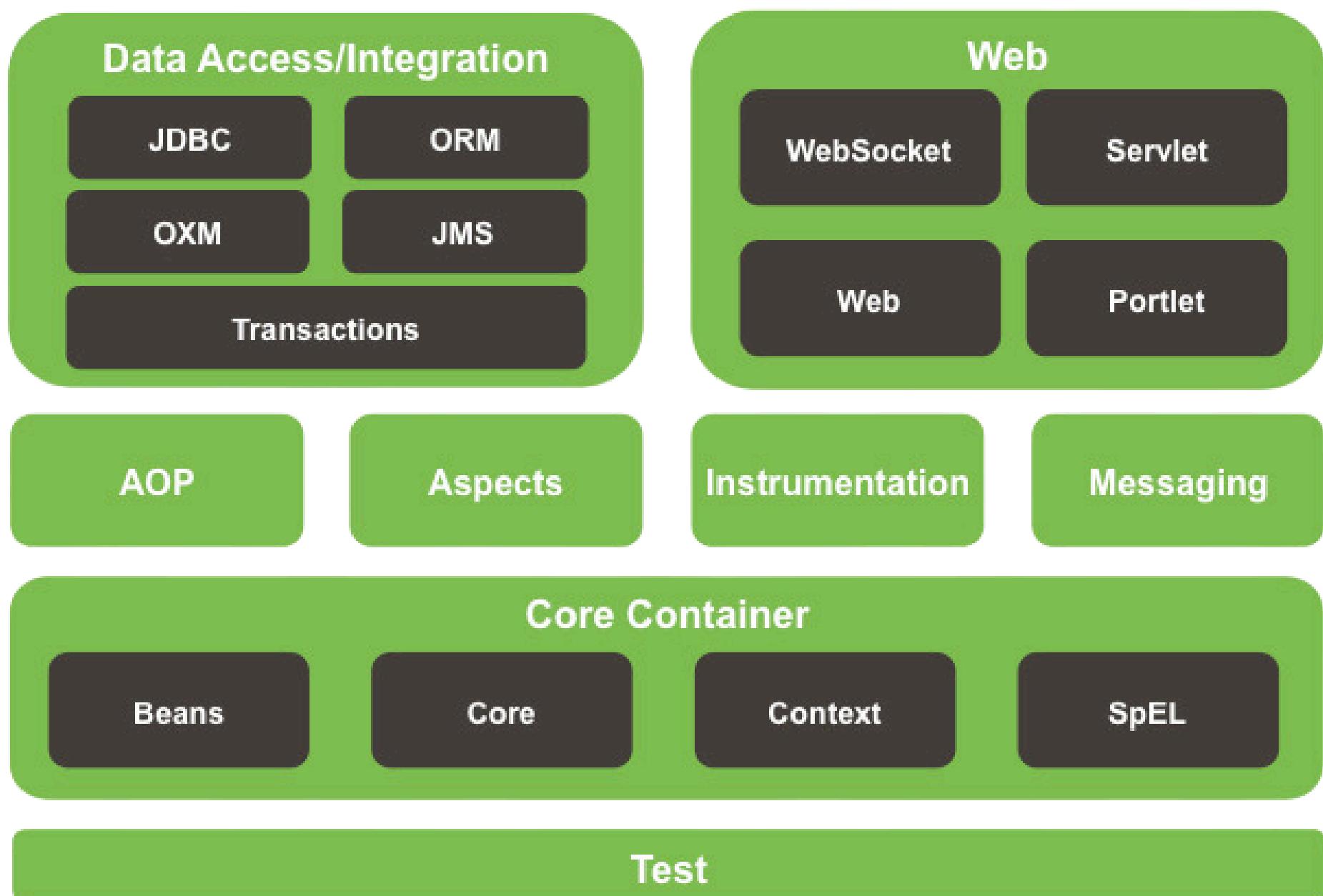


**LAHIRU
LIYANAPATHIRANA**



Spring Modules Overview

Spring Framework provides a modular architecture that enhances flexibility and component reusability, ease of integration with other components among other benefits.



Spring Modules Overview

Spring Framework contains about 20 main modules.

These modules are grouped into the following:

- Core Container
- AOP (Aspect Oriented Programming)
- Aspects
- Instrumentation
- Messaging
- Data Access/Integration
- Web
- Testing



Core Container

spring-core:

The spring-core module contains the fundamental parts and the core utilities other spring modules use.

spring-beans:

The spring-beans module provides support for beans. Beans are objects managed by the IoC (Inversion of Control) Container.

spring-context:

The spring-context and spring-beans modules provide the basis for the IoC container and Dependency Injection functionalities.

It is also responsible for instantiating, configuring, and assembling the beans.



Core Container

spring-context-support:

The `spring-context-support` module handles classes for integrating common third-party libraries into a Spring application context.

spring-expression:

The `spring-expression` module provides the Spring Expression Language (SpEL) which is an expression language that supports querying and manipulating an object graph at runtime.



AOP and Aspects

spring-aop:

The spring-aop module provides the implementation of aspect-oriented programming concepts into Spring.

AOP enables modularization by separating cross-cutting concerns (like logging and transaction management) into distinct aspects that can be applied across multiple classes.

spring-aspects:

The spring-aspects module provides the integration mechanisms for AspectJ based aspects.



Instrumentation

spring-instrument:

The spring-instrument module provides class instrumentation and classloader implementations for application servers, enabling features like hot swapping.

spring-instrument-tomcat:

The spring-instrument-tomcat provides an extension of Tomcat's default class loader, enabling instrumentation of loaded classes

Messaging

spring-messaging:

The spring-messaging provides support for messaging architectures and protocols.



Data Access/Integration

spring-jdbc:

The spring-jdbc provides an abstraction layer for JDBC which handles resource management, exception handling, and the execution of SQL queries.

spring-orm:

The spring-orm module provides integration layers for object-relational mapping APIs, including JPA, JDO, and Hibernate.

spring-oxm:

The spring-oxm module provides an abstraction layer that supports Object/XML mapping implementations such as JAXB, Castor, XMLBeans, JiBX and XStream.



Data Access/Integration

spring-tx:

The `spring-tx` module provides programmatic and declarative transaction management capabilities to Spring.

It simplifies transaction management across various data access technologies, ensuring data integrity and consistency.

spring-jms:

The `spring-jms` module provides support for Java Messaging Service (JMS) features and includes helper classes to send/receive JMS messages.

It also provides integrations with the `spring-messaging` module.



Web

spring-web:

The spring-web module provides essential features for building web applications integration with servlets and web contexts. It also provides HTTP client functionalities

spring-webmvc:

The spring-webmvc module contains Spring's model-view-controller (MVC) and REST Web Services implementation for web applications.

It also provides a clean separation between domain model code and web forms and integrates with all other features of the Spring Framework.



Web

spring-websocket:

The `spring-websocket` module provides support for WebSocket and SockJS infrastructure, including STOMP messaging support.

spring-webmvc-portlet:

The `spring-webmvc-portlet` module provides the MVC implementation for the Portlet environment and it is similar in the functionality of the Servlet-based `spring-webmvc`.



Testing

spring-test:

The spring-test module provides support for the unit testing and integration testing for Spring components with JUnit/TestNG.

It also provides mocking of objects using Mockito.



**Did You Find This
Post Useful?**

**Stay Tuned for More
Spring Related Posts
Like This**

