دوره آموزشی اسپرینگ فریمورک

مدرس: اسماعیل صادقی

JavaTarFoundation

EsmaelSadeghijob@gmail.com

Contents

- Overview
- Spring IO Platform
- Spring Framework
- Environment Setup
- Inversion of Control
- Dependency injection
- Spring Web MVC Framework
- Spring Security

JavaTar
EsmaelSadeghi2ob@gmail.com

JavaTarFoundation

JavaTar
EsmaelSadeghi2job@gmail.com

Spring Framework

The Spring Framework provides a comprehensive programming and configuration model for modern Java-based enterprise applications - on any kind of deployment platform.

A key element of Spring is infrastructural support at the application level: Spring focuses on the "plumbing" of enterprise applications so that teams can focus on application-level business logic, without unnecessary ties to specific deployment environments.

JavaTarFoundation

JavaTar
EsmaelSadeghi2job@gmail.com

POJO

```java
package com.sample;


public class Actor {

    private String actorId;
    private String firstName;
    private String lastName;
    private java.sql.Timestamp lastUpdate;


    public String getActorId() {
        return actorId;
    }

    public void setActorId(String actorId) {
        this.actorId = actorId;
    }


    public String getFirstName() {
        return firstName;
    }

    public void setFirstName(String firstName) {
        this.firstName = firstName;
    }


    public String getLastName() {
        return lastName;
    }
```

JavaTarFoundation

JavaTar
EsmaelSadeghi2job@gmail.com

Version History | JDK Version Range

Version	Date
0.9	2002
1.0	2003
2.0	2006
3.0	2009
4.0	2013
5.0	2017

Spring Framework 5.3.x: JDK 8-17 (expected)

Spring Framework 5.2.x: JDK 8-15 (expected)

Spring Framework 5.1.x: JDK 8-12

Spring Framework 5.0.x: JDK 8-10

Spring Framework 4.3.x: JDK 6-8

Spring Framework

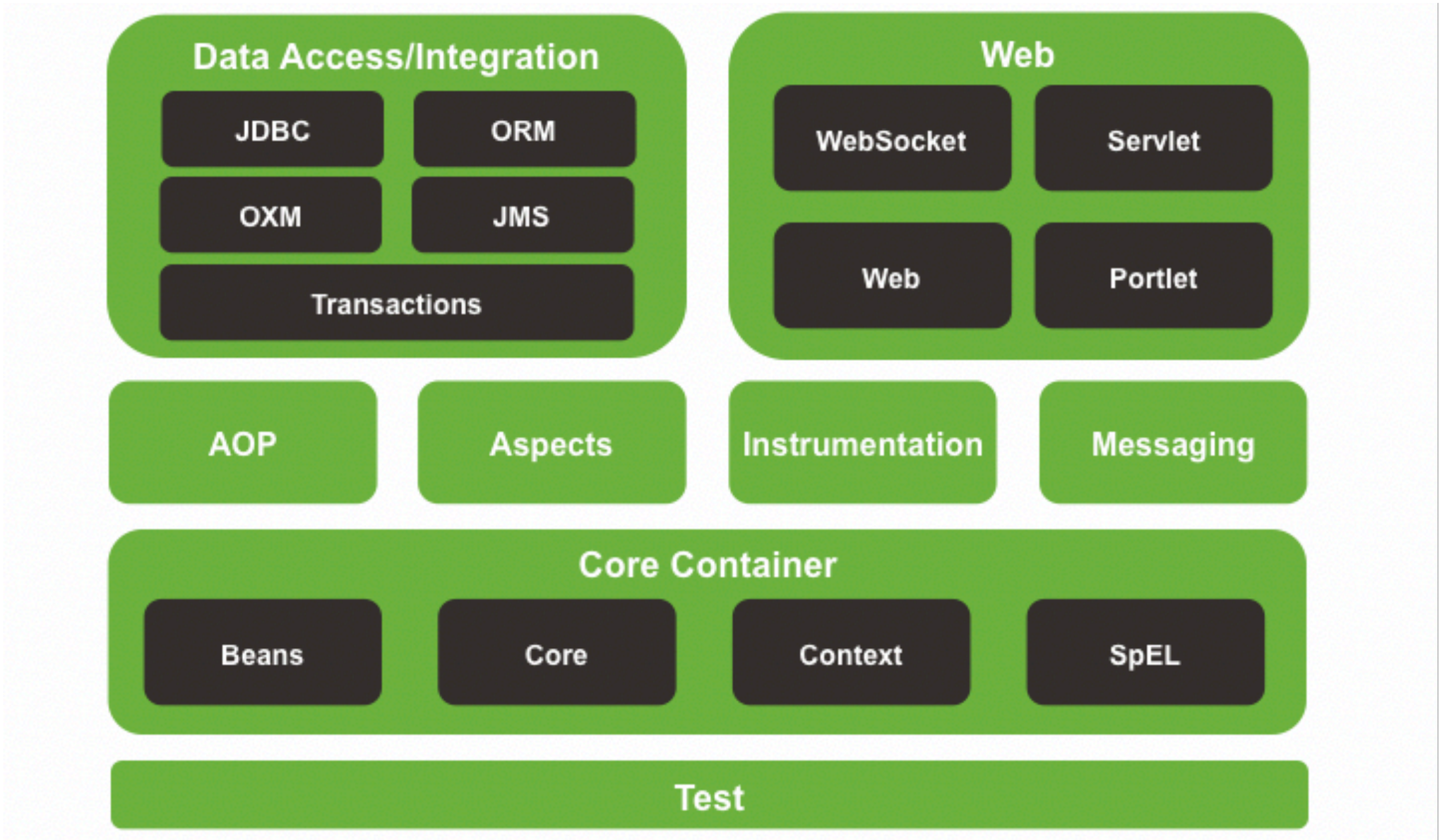
The Spring Framework provides a comprehensive programming and configuration model for modern Java-based enterprise applications - on any kind of deployment platform.

A key element of Spring is infrastructural support at the application level: Spring focuses on the "plumbing" of enterprise applications so that teams can focus on application-level business logic, without unnecessary ties to specific deployment environments.

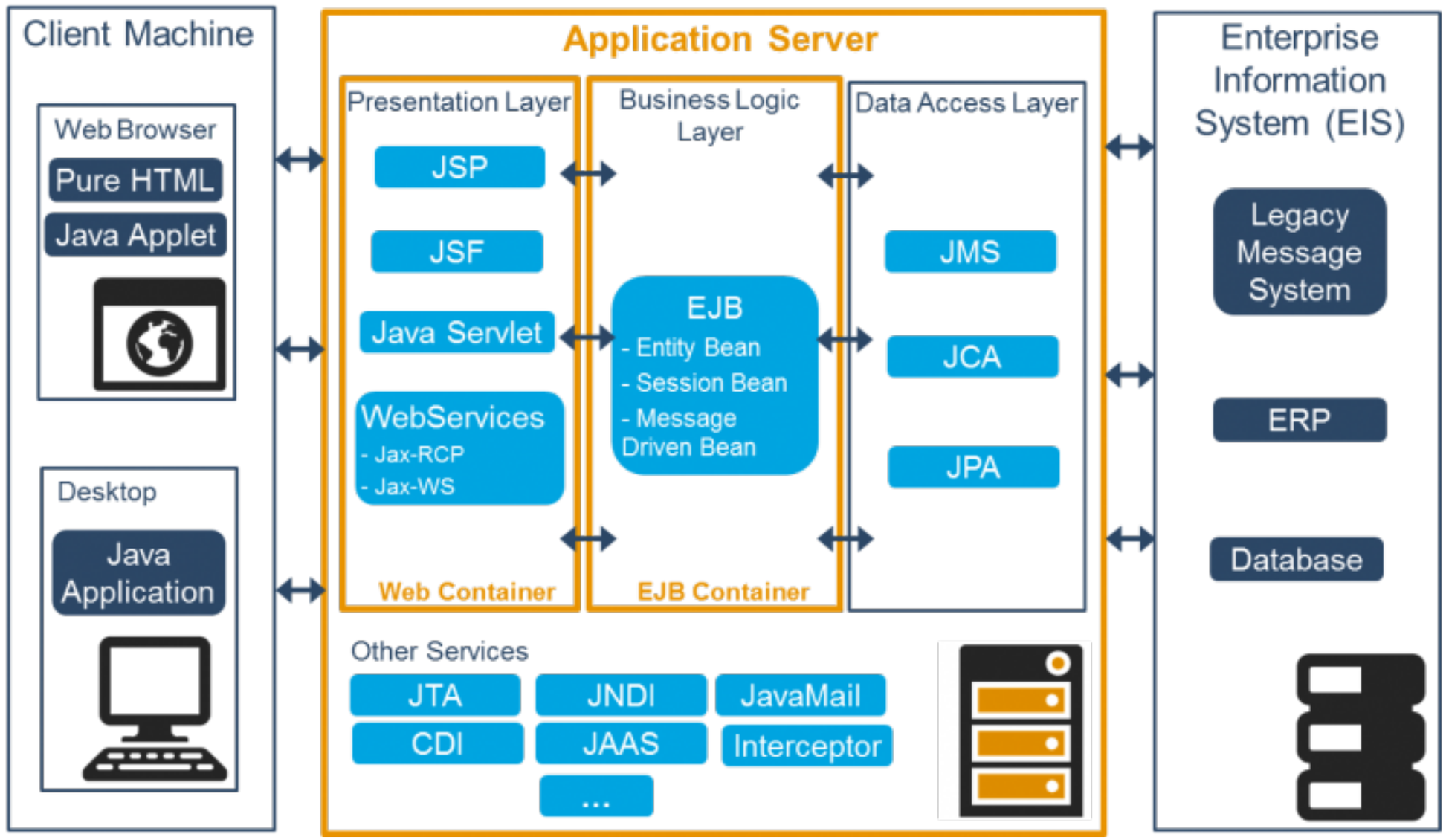
JavaTarFoundation

JavaTar
EsmaelSadeghi2job@gmail.com

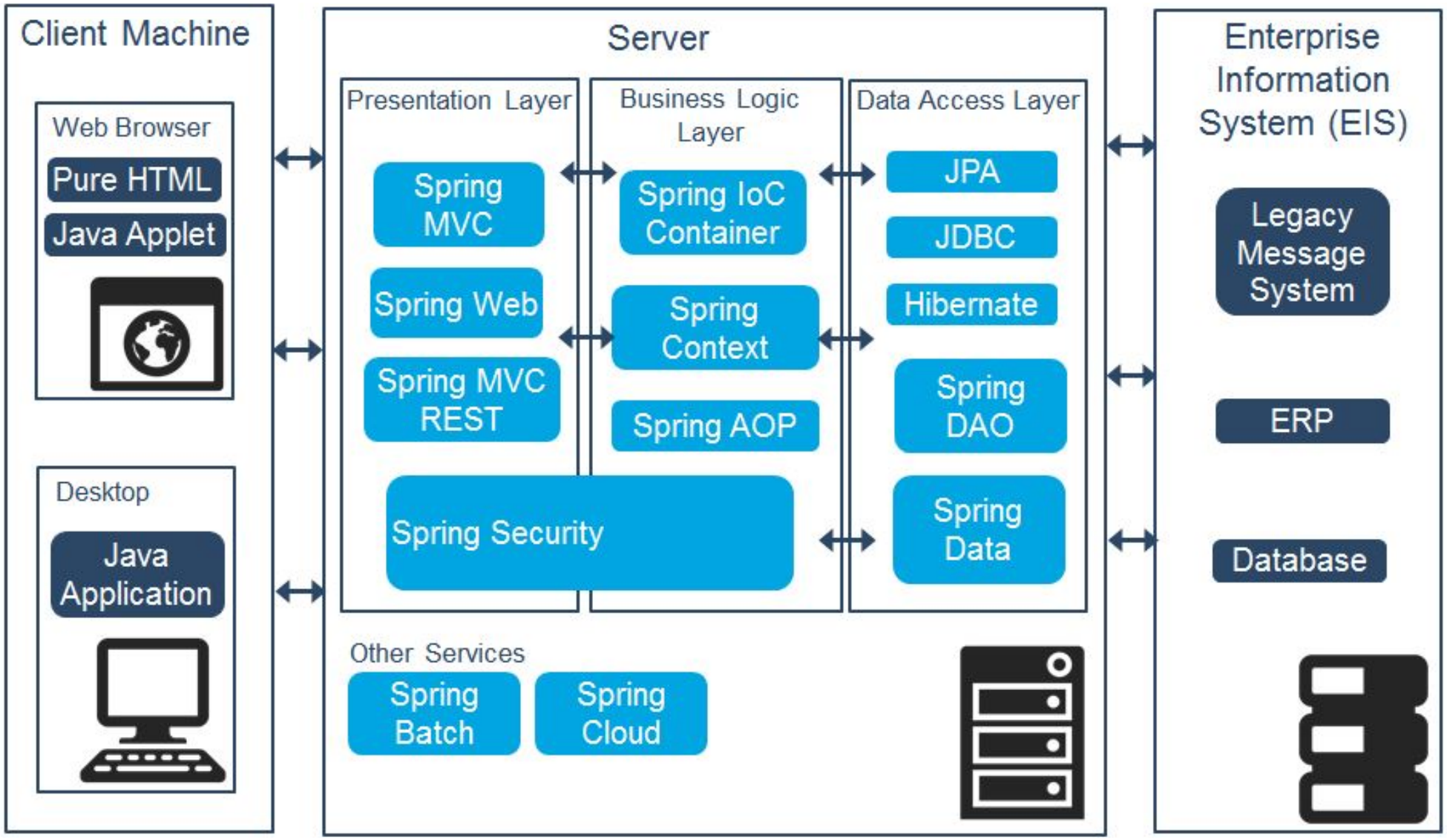
Architecture



JavaEE vs Spring Framework Architecture



JavaEE vs Spring Framework Architecture



JavaEE vs Spring Framework

#1. Architecture

Java EE	Spring
Based on three-dimensional Architectural Framework i.e. Logical Tiers, Client Tiers, and Presentation Tiers.	It is based on a layered architecture that includes many modules. These modules are made on top of its core container.

#2. Language

Java EE	Spring
It uses a high-level object-oriented language which has a certain style and syntax.	It doesn't have a certain programming model.

JavaTarFoundation

JavaTar
EsmaelSadeghi.job@gmail.com

JavaEE vs Spring Framework

#3. Interface

Java EE	Spring
It typically has a graphical user interface created from Project Swing or Abstract Window Toolkit APIs.	Syntax the same everywhere – independent of an IDE or a compiler.

#4. Dependency Injection

Java EE	Spring
Uses dependency injection.	Uses dependency injection.

JavaTarFoundation

JavaTar
EsmaelSadeghi2ob@gmail.com

JavaEE vs Spring Framework

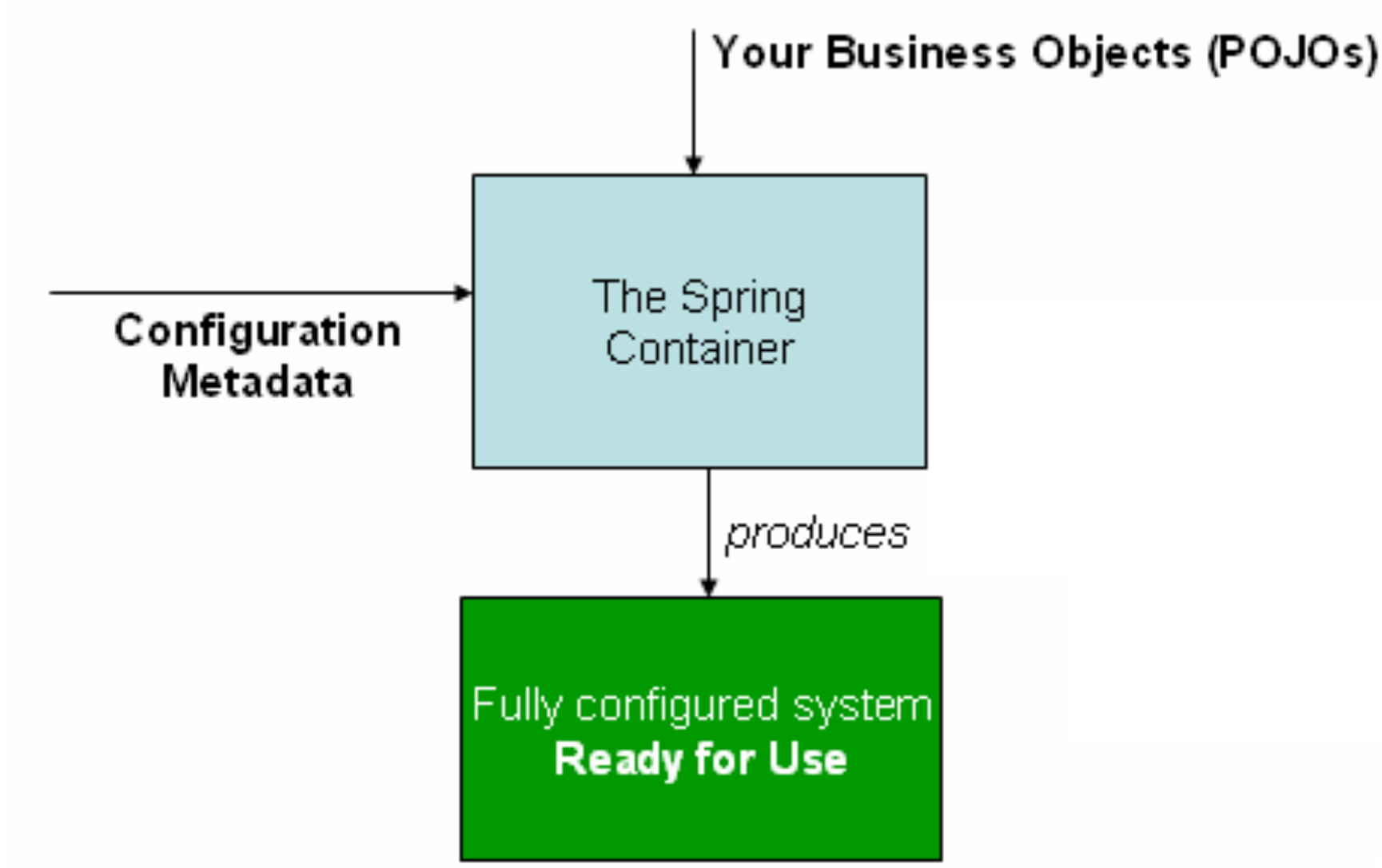
#5. Structure

Java EE	Spring
Can be web based or non-web based.	Based on almost 20 modules.

#6. Speed

Java EE	Spring
Quite good speed.	Spring is slower than Java EE.

JavaEE vs Spring Framework



Bean

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.

Property	Explained in…
class	the section called “Instantiating beans”
name	the section called “Naming beans”
scope	Section 3.5, “Bean scopes”
constructor arguments	the section called “Dependency Injection”
properties	the section called “Dependency Injection”
autowiring mode	the section called “Autowiring collaborators”
lazy-initialization mode	the section called “Lazy-initialized beans”
initialization method	the section called “Initialization callbacks”
destruction method	the section called “Destruction callbacks”

Metadata Spring Configuration

XML (XML based configuration)

Annotation

Java

JavaTar
EsmaelSadeghi2ob@gmail.com

Exp.

```java
package com.tutorialspoint;

public class HelloWorld {
   private String message;

   public void setMessage(String message){
      this.message  = message;
   }
   public void getMessage(){
      System.out.println("Your Message : " + message);
   }
}
```

1

JavaTarFoundation

JavaTar
EsmaelSadeghi2ob@gmail.com

Exp.

2

```
package com.tutorialspoint;

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

public class MainApp {
   public static void main(String[] args) {
      ApplicationContext context = new FileSystemXmlApplicationContext
         ("C:/Users/ZARA/workspace/HelloSpring/src/Beans.xml");

      HelloWorld obj = (HelloWorld) context.getBean("helloWorld");
      obj.getMessage();
   }
}
```

JavaTar
EsmaelSadeghi2ob@gmail.com

Exp.

3

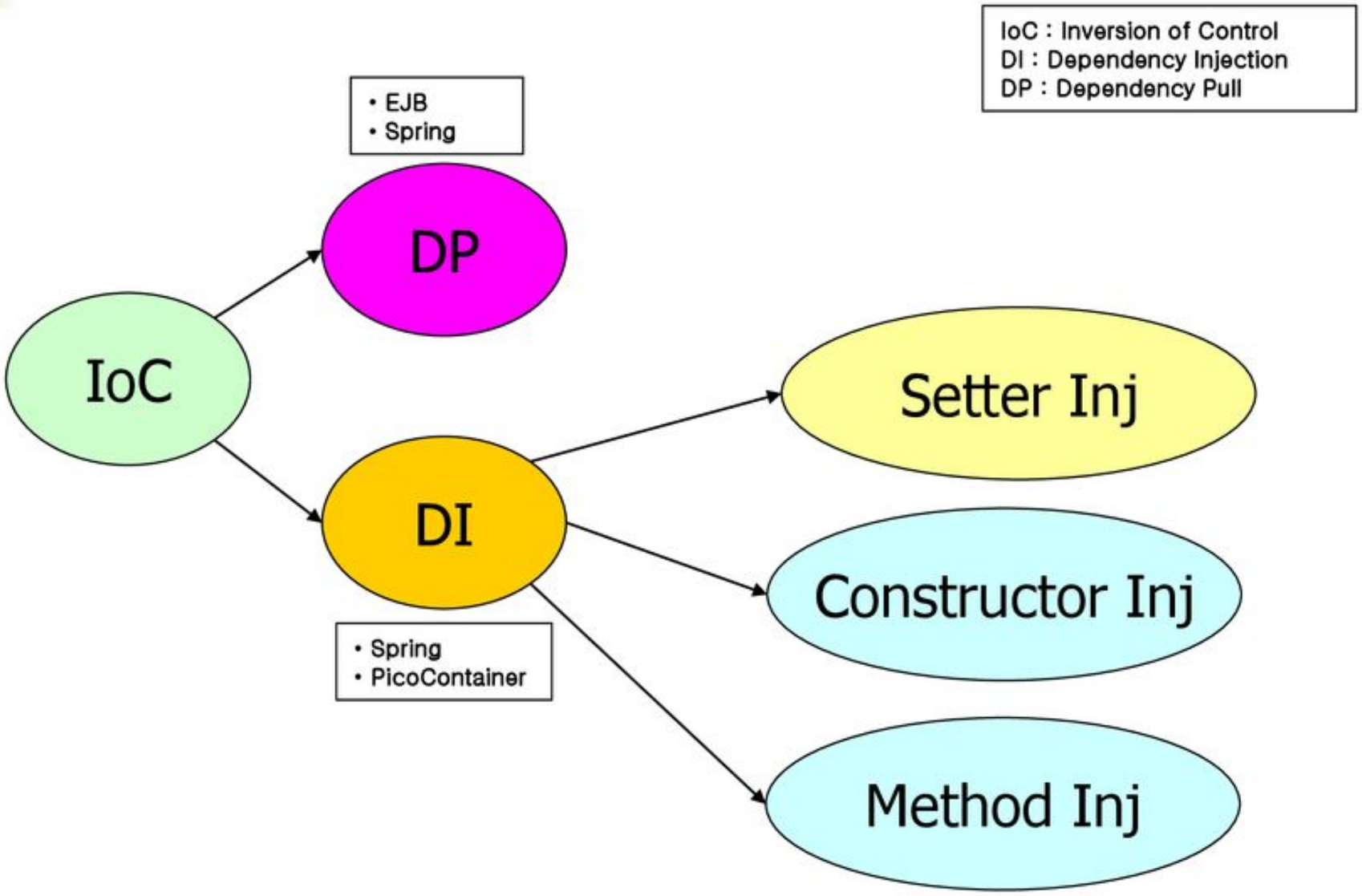
```xml
<?xml version = "1.0" encoding = "UTF-8"?>

<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-3.0.xsd">

   <bean id = "helloWorld" class = "com.tutorialspoint.HelloWorld">
      <property name = "message" value = "Hello World!"/>
   </bean>

</beans>
```

JavaTarFoundation

JavaTar
EsmaelSadeghi2ob@gmail.com

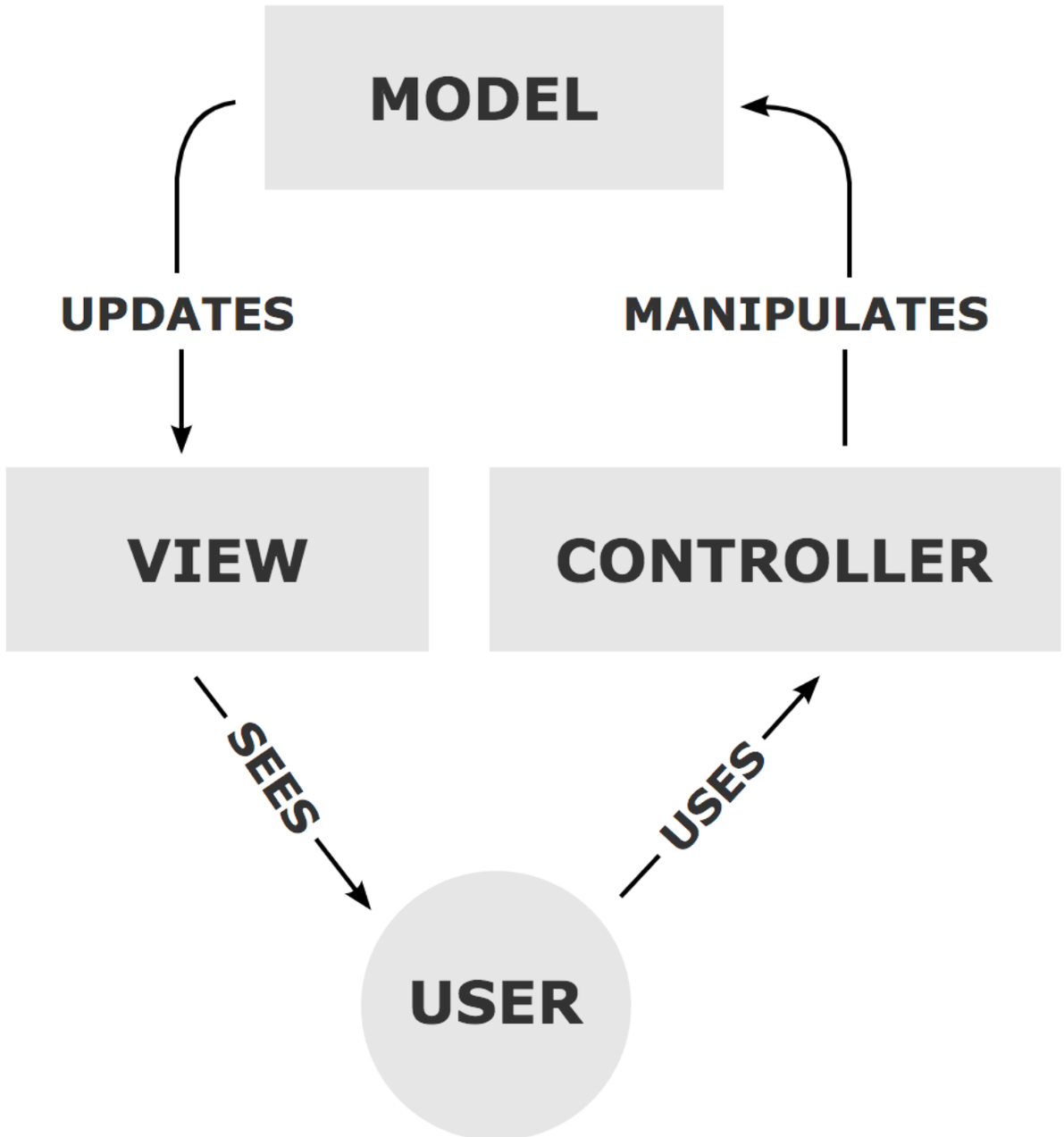
Dependencies

Dependency Injection



JavaTar
EsmaelSadeghi2job@gmail.com

MVC



MVC

Model directly manages the data, logic and rules of the application

View can be any output representation of information

Controller, accepts input and converts it to commands for the model or view

JavaTarFoundation

JavaTar
EsmaelSadeghi2job@gmail.com

Spring Web MVC Framework

The Spring Web MVC framework provides model-view-controller architecture and ready components that can be used to develop flexible and loosely coupled web applications.

The Model encapsulates the application data and in general they will consist of POJO.

The View is responsible for rendering the model data and in general it generates HTML output that the client's browser can interpret.

The Controller is responsible for processing user requests and building appropriate model and passes it to the view for rendering.

JavaTar
EsmaelSadeghi2job@gmail.com

Spring Web MVC Framework

The Spring Web MVC framework provides model-view-controller architecture and ready components that can be used to develop flexible and loosely coupled web applications.

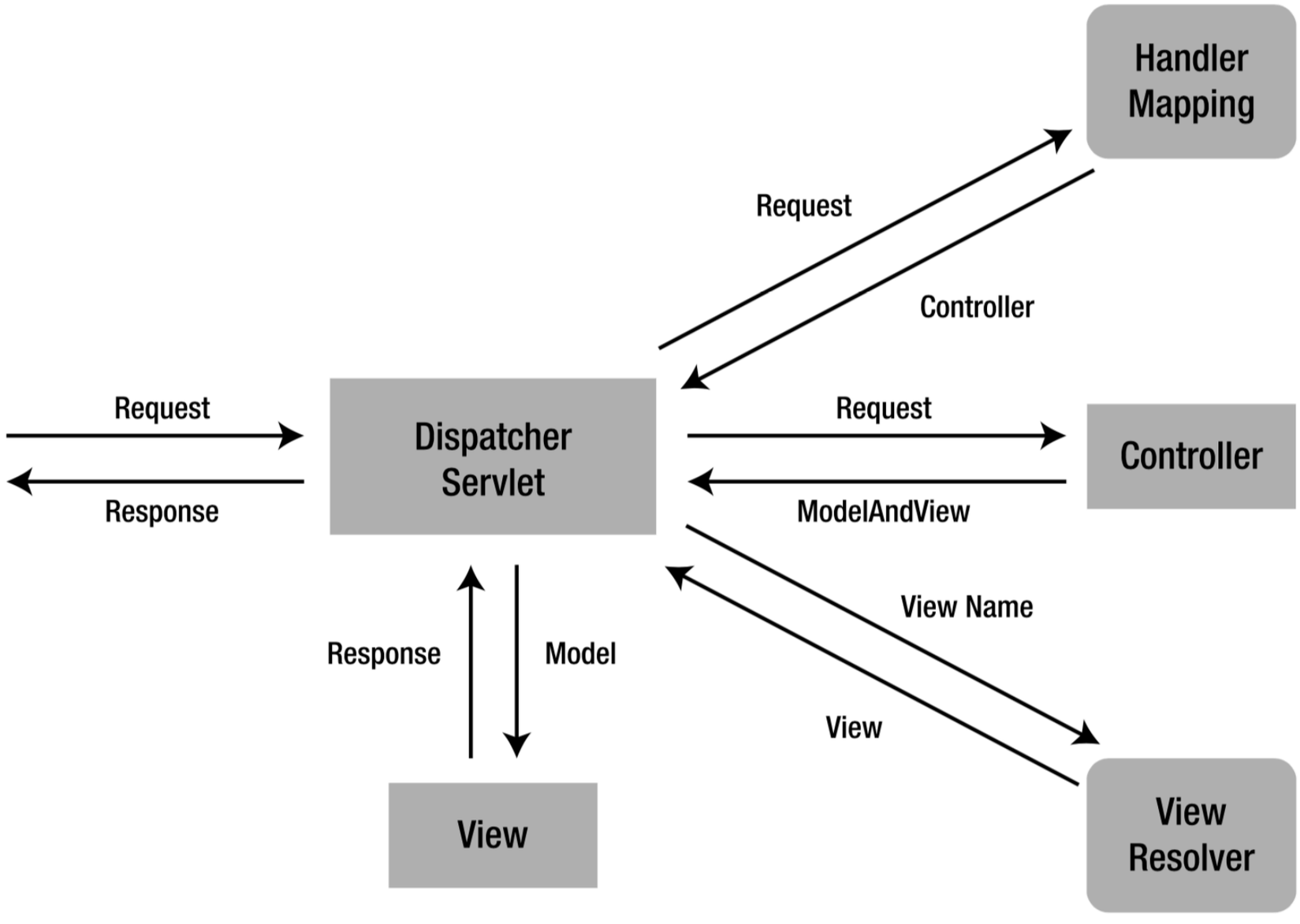
The Model encapsulates the application data and in general they will consist of POJO.

The View is responsible for rendering the model data and in general it generates HTML output that the client's browser can interpret.

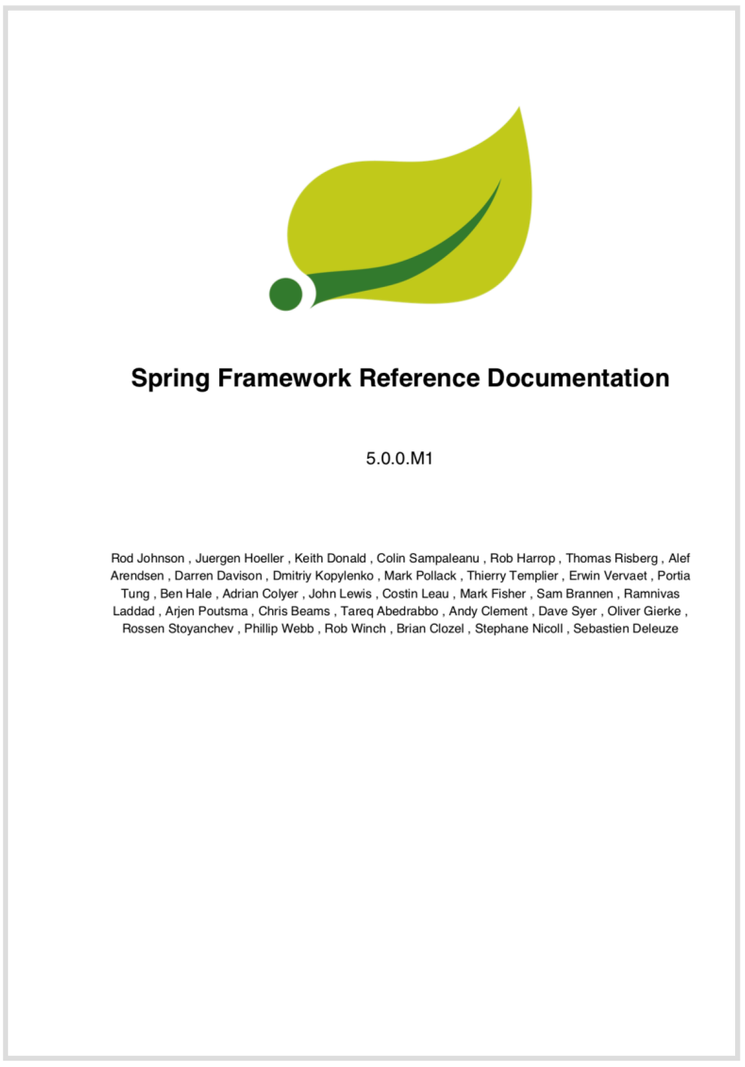
The Controller is responsible for processing user requests and building appropriate model and passes it to the view for rendering.

JavaTar
EsmaelSadeghi2job@gmail.com

Primary flow of request handling in Spring MVC



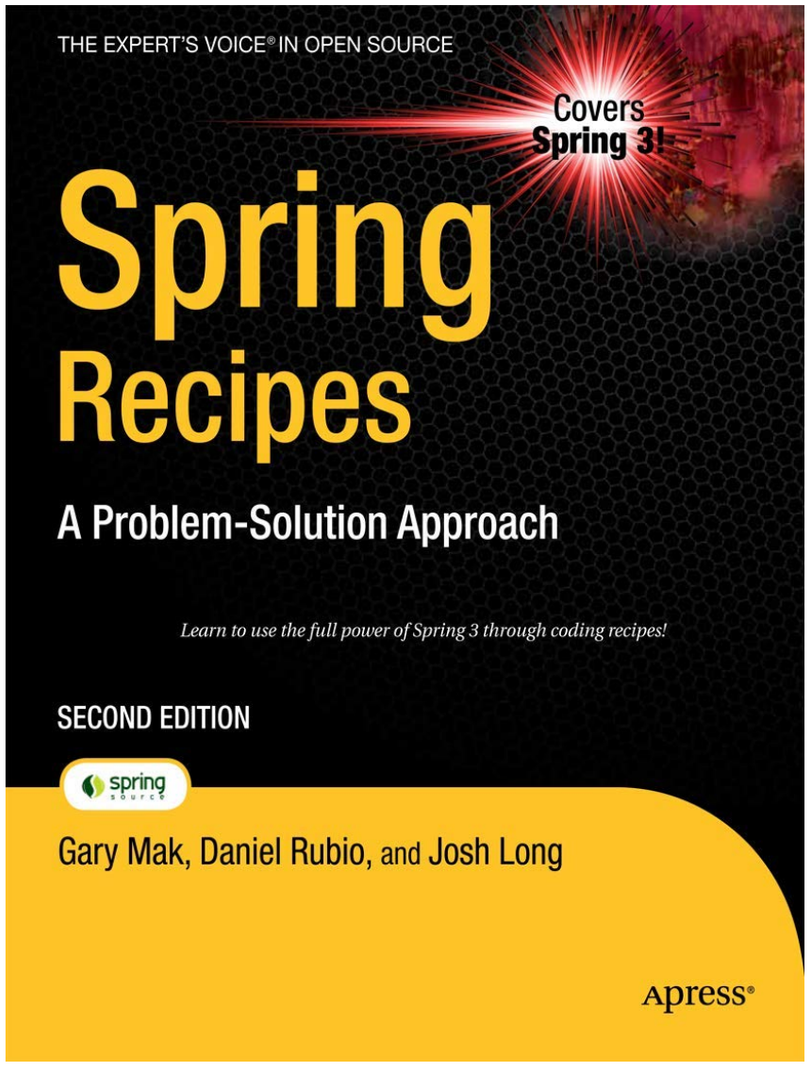
Book

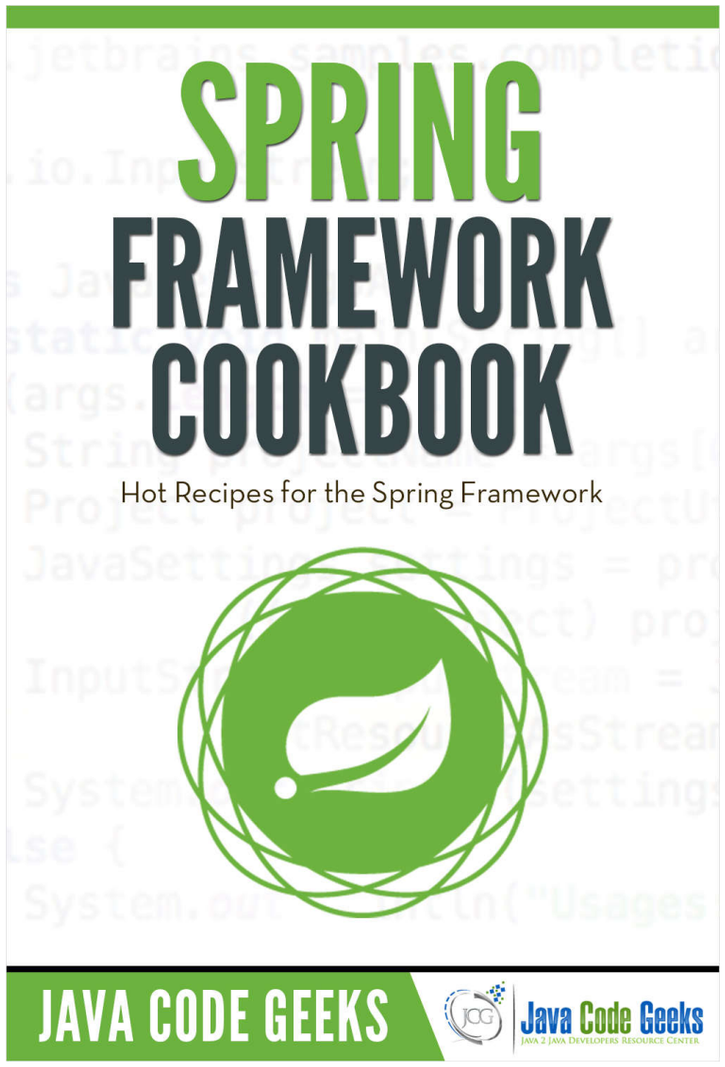


Spring Framework Reference Documentation

5.0.0.M1

Rod Johnson , Juergen Hoeller , Keith Donald , Colin Sampaleanu , Rob Harrop , Thomas Risberg , Alef Arendsen , Darren Davison , Dmitriy Kopylenko , Mark Pollack , Thierry Templier , Erwin Vervaet , Portia Tung , Ben Hale , Adrian Colyer , John Lewis , Costin Leau , Mark Fisher , Sam Brannen , Ramnivas Laddad , Arjen Poutsma , Chris Beams , Tareq Abedrabbo , Andy Clement , Dave Syer , Oliver Gierke , Rossen Stoyanchev , Phillip Webb , Rob Winch , Brian Clozel , Stephane Nicoll , Sebastien Deleuze





JavaTarFoundation

JavaTar

EsmaelSadeghizjob@gmail.com

JavaTarFoundation

JavaTar

EsmaelSadeghijob@gmail.com