

Programming Advanced Java

Spring Web MVC

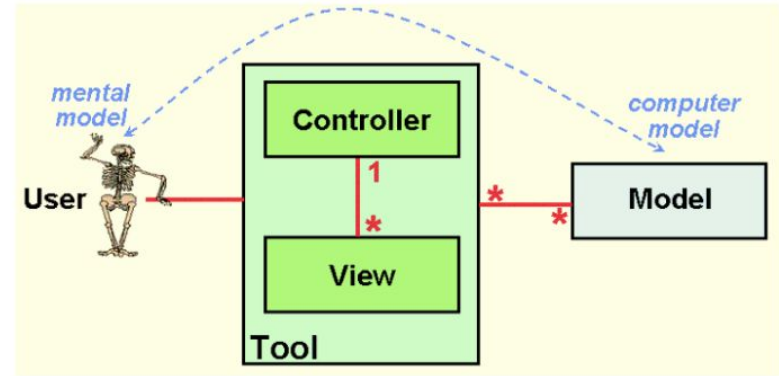
Goals



The **junior-colleague**

- can describe what Spring Web MVC is
- can explain the Model-View-Controller design pattern
- can explain the role of DispatcherServlet in Spring
- can consume Rest services with WebClient
- can explain the Model-View-Controller pattern
- can implement a simple view with Thymeleaf

The Model-View-Controller pattern



(original model of MVC by Trygve Reenskaug)

MVC can be understood by thinking of responsibilities:

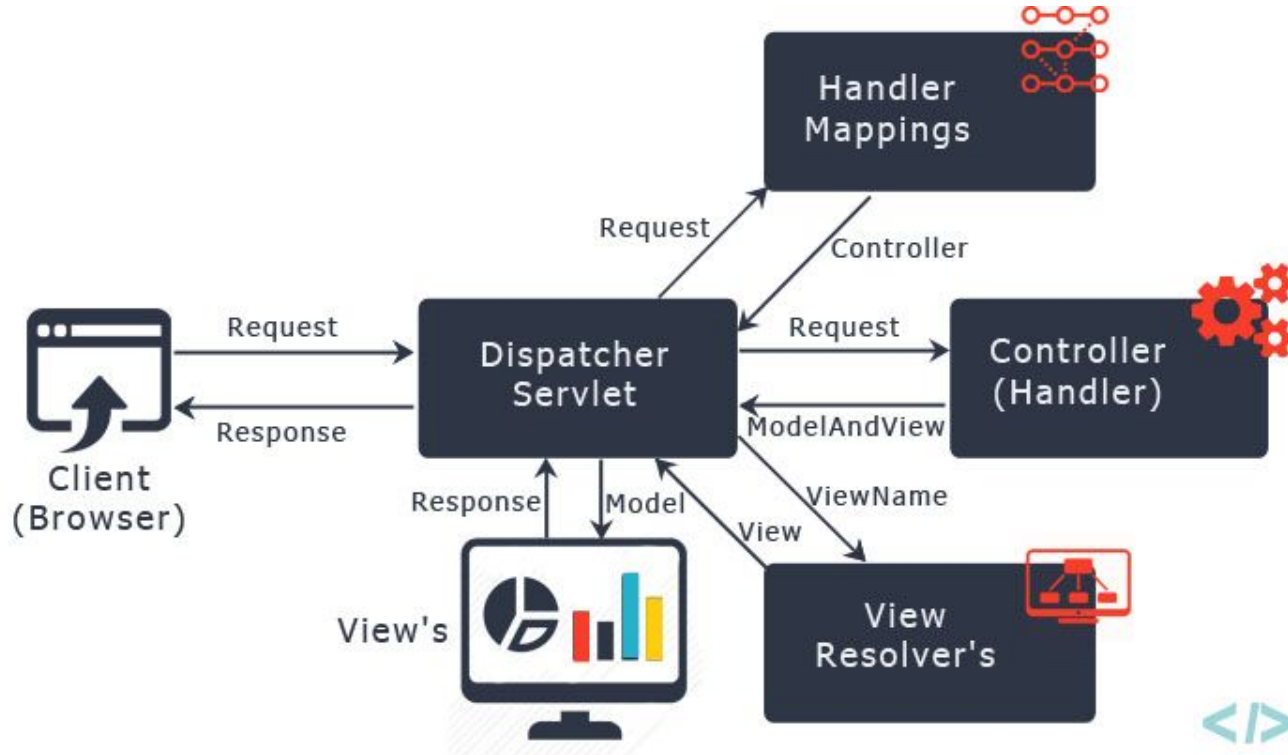
The View is not allowed to change the state of the model directly - only through the Controller. The view can still have some direct access to the Model although only for viewing (or by having a copy that isn't the official Model).

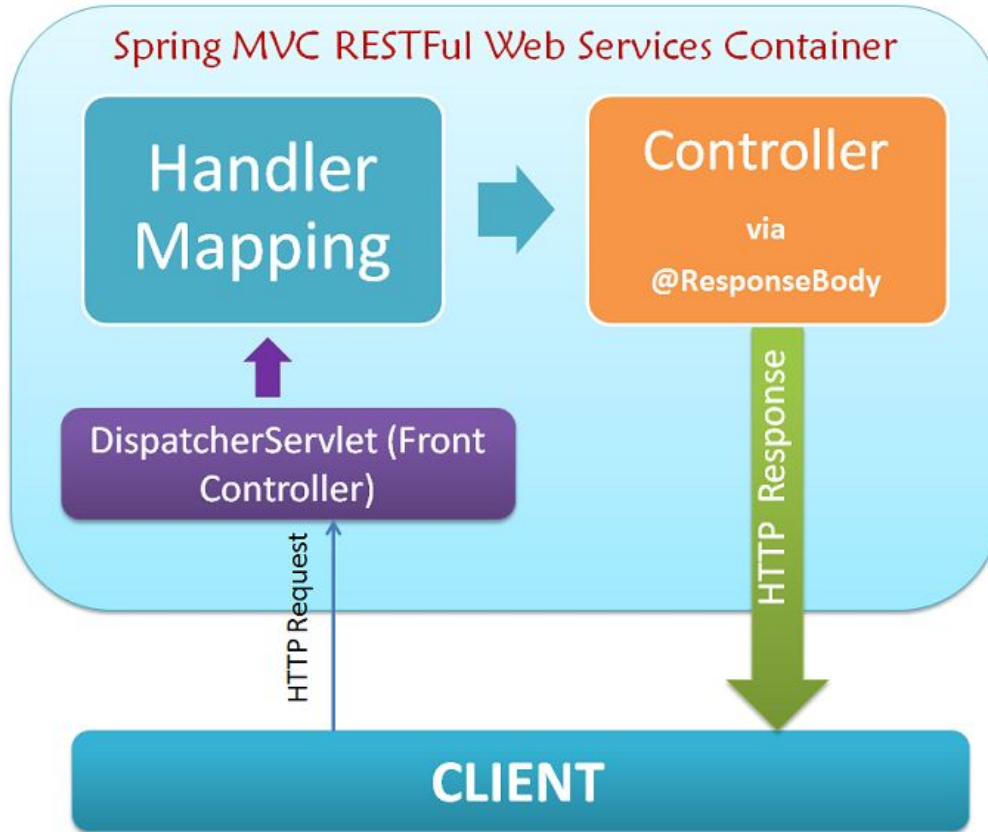
The Model should live in its own universe and not have any reference to controllers or to views.

The Controller controls the state and access to the Model.

<https://stackoverflow.com/questions/2663674/correct-implementation-of-mvc-architecture/2663768>

Spring MVC





Template engines

Given that concerns in a Spring MVC application are cleanly separated switching from one view technology to another is primarily a matter of configuration.

Traditional technology: *Java Server Pages*

Main template engines that can be used with Spring: *Thymeleaf, Groovy, FreeMarker, Jade.*

Demo

https://github.com/custersnele/PAJ_servlets



Thymeleaf

Exercises

- <http://itutorial.thymeleaf.org/>
- See textbook