DispatcherServlet

The DispatcherServlet

HTTP Request → **DispatcherServlet** → **HTTP Response**

① Handler Mapping
② Controller
③ View Resolver
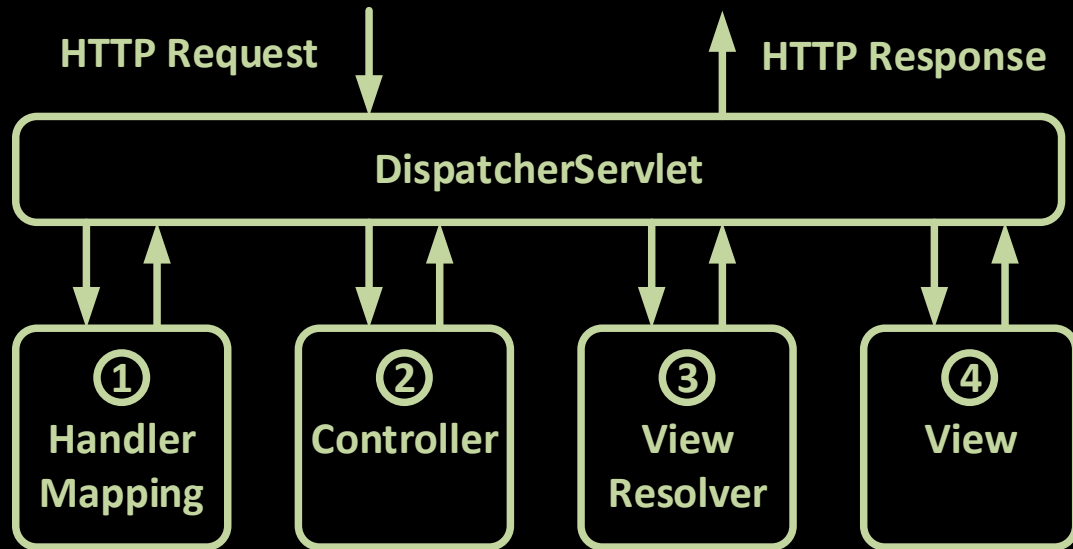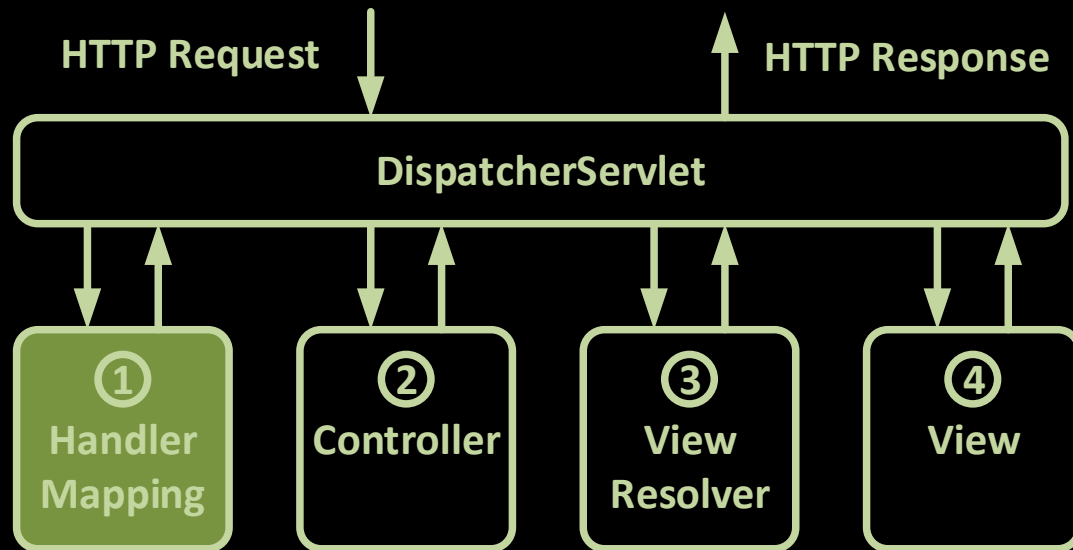④ View

The Spring Web model-view-controller (MVC) framework is designed around a *DispatcherServlet* that handles all the HTTP requests and responses. The request processing workflow of the Spring Web MVC *DispatcherServlet* is illustrated in the following diagram.
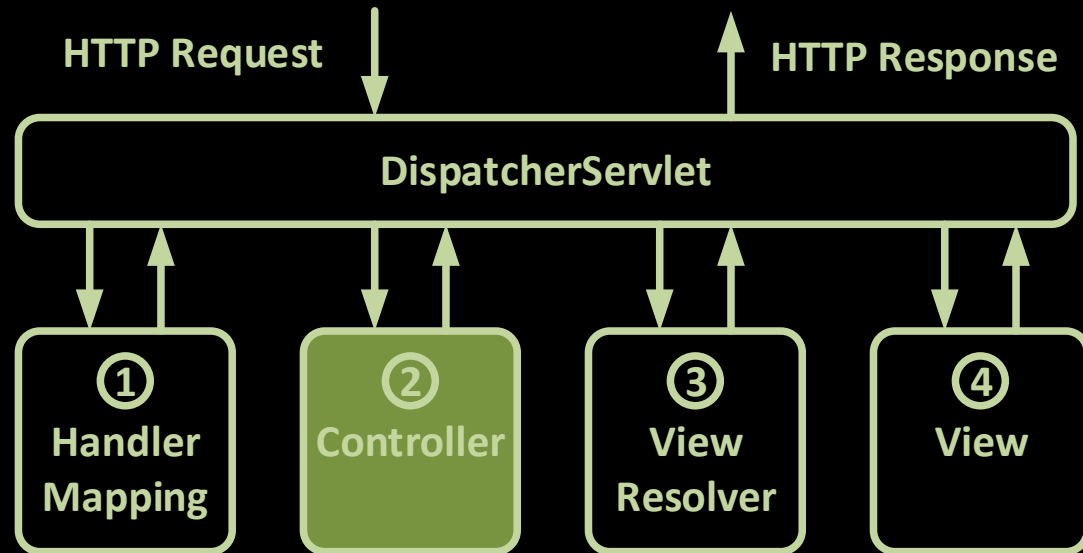
All the above-mentioned components, i.e. HandlerMapping, Controller, and ViewResolver are parts of *WebApplicationContext* which is an extension of the plain *ApplicationContext* with some extra features necessary for web applications.

Sequence of events corresponding to an incoming HTTP request to DispatcherServlet

After receiving an HTTP request, *DispatcherServlet* consults the *HandlerMapping* to call the appropriate *Controller*.

HTTP Request

HTTP Response

DispatcherServlet

① Handler Mapping

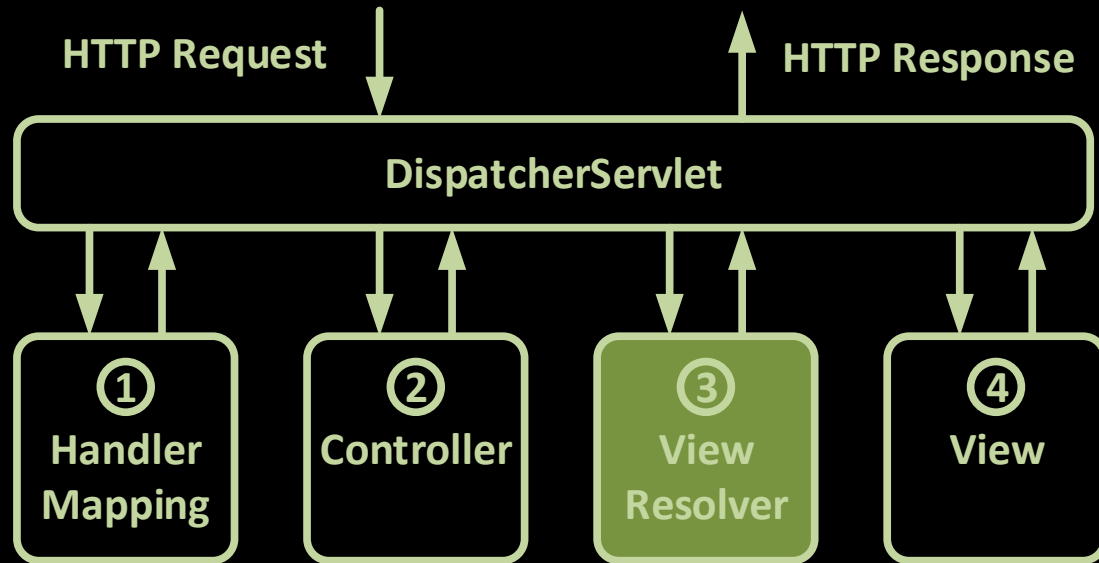② Controller

③ View Resolver

④ View

Sequence of events corresponding to an incoming HTTP request to DispatcherServlet

The *Controller* takes the request and calls the appropriate service methods based on used GET or POST method. The service method will set model data based on defined business logic and returns view name to the *DispatcherServlet*.

HTTP Request

HTTP Response

DispatcherServlet

① Handler Mapping

② Controller

③ View Resolver

④ View

Sequence of events corresponding to an incoming HTTP request to DispatcherServlet

The *DispatcherServlet* will take help from *ViewResolver* to pickup the defined view for the request.

Sequence of events corresponding to an incoming HTTP request to DispatcherServlet

Once view is finalized,
The *DispatcherServlet* passes the model data to the view which is finally rendered on the browser.

HTTP Request

HTTP Response

DispatcherServlet

① **Handler Mapping**

② **Controller**

③ **View Resolver**

④ **View**