[Bài đọc] Giới thiệu về Spring MVC

*Nguồn:*[*Spring MVC: A Tutorial (Second Edition)*](https://books.google.com.vn/books?id=O1LzCwAAQBAJ&printsec=frontcover#v=onepage&q&f=false)*, Paul Deck*

Spring MVC là một framework giúp các nhà phát triển viết các ứng dụng MVC nhanh hơn.

Trong phần này, chúng ta sẽ thảo luận về những lợi ích của Spring MVC. Chúng ta cũng sẽ thảo luận về một số thành phần cơ bản của Spring MVC, chẳng hạn như Dispatcher Servlet (trình điều phối), và cách viết các controller kiểu cũ (cách để viết các controller trong các phiên bản cũ của Spring trước phiên bản 2.5). Một loại controller khác cũng được trình bày, đó là “Annotation-based Controllers”. [Bài đọc] Giới thiệu về Spring MVC

cũ được thảo luận ở đây vì bạn vẫn có thể phải làm việc với mã kế thừa được viết bằng các phiên bản cũ của Spring. Tuy nhiên, đối với những mã mới phát triển, bạn nên sử dụng “annotation-based controllers”.

Phần này cũng sẽ giới thiệu cấu hình Spring MVC. Hầu hết các ứng dụng Spring MVC sẽ có một tệp XML để khai báo các bean khác nhau được sử dụng trong ứng dụng.

**Lợi ích của Spring MVC**

Khi viết một ứng dụng MVC, bạn thường phải viết một trình điều phối (servlet dispatcher) và các bộ điều khiển (controller). Trình điều phối của bạn phải có khả năng làm những việc sau:

* Xác định từ URI hành động nào được gọi.
* Khởi tạo đúng controller.
* Tạo một bean mẫu với các giá trị tham số yêu cầu.
* Gọi đúng phương thức trong controller.
* Chuyển điều khiển đến một view (trang JSP).

Spring MVC là một framework MVC sử dụng một trình điều phối để gọi các phương thức trong controller và chuyển tiếp điều khiển đến một view (khung nhìn). Đây là lợi ích đầu tiên của việc sử dụng Spring MVC: Bạn không cần viết servlet điều phối của riêng mình. Dưới đây là danh sách các tính năng mà Spring MVC được trang bị để phát triển nhanh hơn:

Controller kiểu cũ được thảo luận ở đây vì bạn vẫn có thể phải làm việc với mã kế thừa được viết bằng các phiên bản cũ của Spring. Tuy nhiên, đối với những mã mới phát triển, bạn nên sử dụng “annotation-based controllers”.

Phần này cũng sẽ giới thiệu cấu hình Spring MVC. Hầu hết các ứng dụng Spring MVC sẽ có một tệp XML để khai báo các bean khác nhau được sử dụng trong ứng dụng.

**Lợi ích của Spring MVC**

Khi viết một ứng dụng MVC, bạn thường phải viết một trình điều phối (servlet dispatcher) và các bộ điều khiển (controller). Trình điều phối của bạn phải có khả năng làm những việc sau:

* Xác định từ URI hành động nào được gọi.
* Khởi tạo đúng controller.
* Tạo một bean mẫu với các giá trị tham số yêu cầu.
* Gọi đúng phương thức trong controller.
* Chuyển điều khiển đến một view (trang JSP).

Spring MVC là một framework MVC sử dụng một trình điều phối để gọi các phương thức trong controller và chuyển tiếp điều khiển đến một view (khung nhìn). Đây là lợi ích đầu tiên của việc sử dụng Spring MVC: Bạn không cần viết servlet điều phối của riêng mình. Dưới đây là danh sách các tính năng mà Spring MVC được trang bị để phát triển nhanh hơn:

Spring MVC cung cấp một trình điều phối, lưu bản ghi của bạn.

* Spring MVC sử dụng tệp cấu hình bằng XML mà bạn có thể chỉnh sửa mà không cần biên dịch lại ứng dụng.
* Spring MVC khởi tạo các lớp controller và gắn các bean với các đầu vào của người dùng.
* Spring MVC tự động liên kết đầu vào của người dùng với đúng kiểu dữ liệu. Ví dụ, Spring MVC có thể tự động phân tích cú pháp một chuỗi và đặt thuộc tính kiểu float hoặc thập phân.
* Spring MVC xác thực đầu vào của người dùng và chuyển hướng người dùng trở lại form đầu vào nếu xác thực không thành công. Xác thực đầu vào là tùy chọn và có thể được thực hiện theo chương trình hoặc theo khai báo. Trước đó, Spring MVC cung cấp trình xác thực tích hợp cho hầu hết các tác vụ bạn có thể gặp phải khi xây dựng một ứng dụng web.
* Spring MVC là một phần của Spring framework. Bạn nhận được mọi thứ mà Spring cung cấp.
* Spring MVC hỗ trợ quốc tế hóa (*internationalization*) và bản địa hoá (*localization*). Điều này có nghĩa là bạn có thể hiển thị tin nhắn bằng nhiều ngôn ngữ tùy thuộc vào ngôn ngữ người dùng.
* Spring MVC hỗ trợ nhiều công nghệ hiển thị (view). Hầu hết trường hợp bạn sẽ sử dụng JSP, nhưng các công nghệ khác, bao gồm Velocity và FreeMarker, cũng được hỗ trợ.

**Trình điều phối Spring MVC (DispatcherServlet)**

Spring MVC đi kèm với một trình điều phối (servlet dispatcher) mà bạn có thể sử dụng ngay lập tức. Tên đầy đủ của nó là *org.springframework.web.servlte.DispatcherServlet*.

Để sử dụng được servlet này, bạn cần cấu hình nó trong tệp tin web.xml bằng cách sử dụng các phần tử servlet và servlet-mapping, cụ thể như sau:

<servlet>  
 <servlet-name>springmvc</servlet-name>  
 <servlet-class  
 org.springframework.web.servlet.DispatcherServlet  
 </servlet-class>  
 <load-on-startup>1</load-on-startup>  
</servlet>  
  
<servlet-mapping>  
 <servlet-name>springmvc</servlet-name>  
 <!-- map all requests to the DispatcherServlet -->  
 <url-pattern>/</url-pattern>  
</servlet-mapping>

Phần tử <load-on-startup> trong <servlet> là tùy chọn. Nếu nó hiện diện, nó sẽ nạp servlet và gọi phương thức init khi ứng dụng được khởi động. Nếu không có phần tử <load-on-startup>, servlet sẽ được nạp khi nó được yêu cầu lần đầu tiên.

Trình điều phối (servlet dispatcher) sẽ sử dụng nhiều thành phần mặc định đi kèm với Spring MVC. Ngoài ra, lúc khởi tạo, nó sẽ tìm kiếm một tệp cấu hình trong thư mục WEB-INF của ứng dụng. Tên tệp XML phải phù hợp với mẫu sau đây:

servletName-servlet.xml

Trong đó, servletName là tên được đặt cho servlet Spring dispatcher trong bộ mô tả triển khai (deployment descriptor). Nếu bạn đã cung cấp cho servlet tên springmvc, bạn sẽ cần phải có tệp springmvc-servlet.xml trong thư mục WEB-INF của thư mục ứng dụng của bạn.

Tuy nhiên, bạn có thể đặt tập tin cấu hình Spring MVC của bạn bất cứ nơi nào trong thư mục ứng dụng của bạn miễn là bạn nói với trình điều phối nơi để tìm nó. Bạn làm điều này bằng cách sử dụng một init-paramelement dưới khai báo servlet. Phần tử init-param sẽ có phần tử param-name có giá trị contextConfigLocation. Nó cũng có phần tử param-value chứa đường dẫn đến tệp cấu hình của bạn. Ví dụ: bạn có thể thay đổi tên và vị trí mặc định của tệp cấu hình thành /WEB-INF/config/simple-config.xml bằng cách sử dụng thẻ init-param:

<servlet>  
 <servlet-name>springmvc</servlet-name>  
 <servlet-class>  
 org.springframework.web.servlet.DispatcherServlet  
 </servlet-class>  
 <init-param>  
 <param-name>contextConfigLocation</param-name>  
 <param-value>/WEB-INF/config/simple-config.xml</param-value>  
 </init-param>  
 <load-on-startup>1</load-on-startup>  
</servlet>

**Controller Interface**

Trước Spring 2.5, cách duy nhất để viết một controller là triển khai giao diện *org.springframework.web.servlet.mvc.Controller*. Giao diện này cho thấy phương thức handleRequest phải được ghi đè (override) bằng cách triển khai các lớp. Đây là nguyên mẫu của phương thức:

ModelAndView handleRequest(HttpServletRequest request, HttpServletResponse response)

Một implementation (triển khai) có quyền truy cập vào HttpServletRequest và HttpServletResponse của yêu cầu tương ứng. Một implementation cũng phải trả về một ModelAndView có chứa một đường dẫn view hoặc một đường dẫn view và một model.

Một controller triển khai controller interface chỉ có thể xử lý một hành động duy nhất. Trái lại, một annnotation-based controller có thể chứa nhiều phương thức xử lý và không phải triển khai bất kỳ interface nào.

Last modified: Friday, 2 October 2020, 9:18 AM