

4-3

ActionDispatcher View Page應用架構



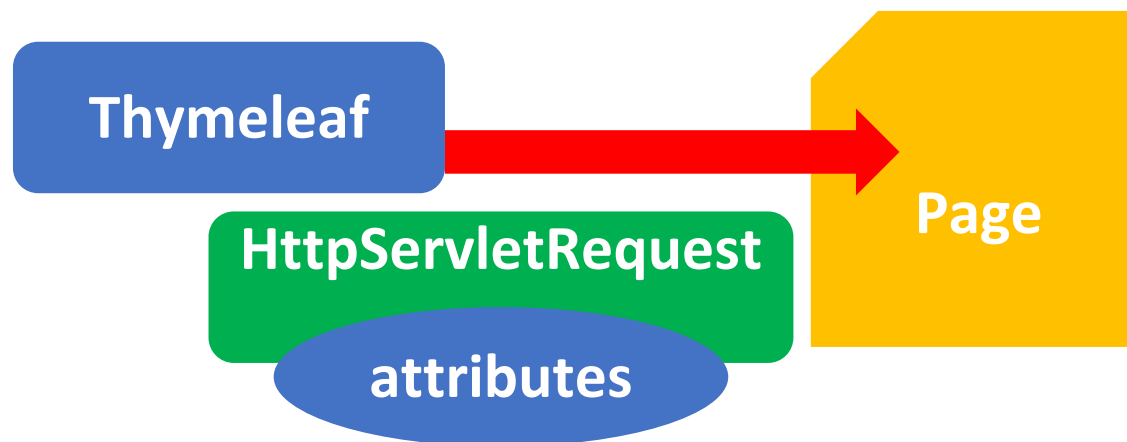
Thymeleaf engine調用頁面
仍使用Request範圍進行派送Dispatcher

狀態管理預設定位為Request Scope

控制項Action注入Java Servlet API
HttpServletRequest物件

進行狀態管理設定

網頁使用thymeleaf expression渲染



Action採用參數注入Servlet API-HttpServletRequest物件

直接action參數注入HttpServletRequest物件

設定Request範圍狀態管理

使用thymeleaf運算頁面 渲染出狀態內容

```
//採用參數注入HttpServletRequest物件
@RequestMapping(path= {"/hello/scope"})
public String passingHello(HttpServletRequest request) {
    //進行Request範圍狀態管理
    request.setAttribute("hello", "世界和平");
    return "whohello";
}
```

Thymeleaf Expression

- 採用th:attribute

使用{xxxx}配合指令\$運算變數內容

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>thymeleaf 調用頁面狀態</title>
</head>
<body>
  <!--使用thymeleaf expression動態渲染-->
  <h1 th:text="${hello}"></h1>
</body>
</html>
```





總結：4-3 ActionDispatcher View Page應用架構

簡單部署第一個Controller，與使用thymeleaf進行
頁面調用，接下來我們正式進入MVC Controller 細
部規劃與設計

