



链滴

Java Web通用架构(一) SpringMVC+Free marker

作者: [sigon](#)

原文链接: <https://ld246.com/article/1445964653339>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

<h2>

一.涉及技术

</h2>

<p>

Spring是Java阵营最重要的框架,没有之一.主要作用是解决企业应用架构的复杂性,实现优雅的分层结构和配置基本的JavaBean完成复杂的业务逻辑.

</p>

<p>

Spring MVC是Spring的MVC解决方案,相比其它MVC框架,与Spring的风格最为接近,可以无缝整合

</p>

<p>

Freemarker是轻量级的模板框架,能避免传统JSP的初始化较慢和代码嵌入的问题,也提供了类似于jst的强大脚本

</p>

<h2>

二.Spring基本配置

</h2>

<p>

首先需要在web.xml配置如下

</p>

<p>

</p>

```
<pre class="prettyprint lang-xml">    <context-param>
    <param-name>contextConfigLocation</param-name>
    <param-value>
        classpath*/applicationContext*.xml
    </param-value>
    </context-param>
</pre>
<pre>
<code class="highlight-chroma">
<span class="highlight-line">
<span class="highlight-cl">
    &lt;!--servlet-->
</span>
</span>
<span class="highlight-line">
<span class="highlight-cl">
    &lt;!--servlet
name-->springmvc</servlet-name-->
</span>
</span>
<span class="highlight-line">
<span class="highlight-cl">
    &lt;!--servlet
class-->org.springframework.web.servlet.DispatcherServlet</servlet-class-->
</span>
</span>
<span class="highlight-line">
<span class="highlight-cl">
    &lt;!--init-param-->
</span>
</span>
<span class="highlight-line">
<span class="highlight-cl">
    &lt;!--param-name-->contextConfigLocation</param-name-->
</span>
</span>
<span class="highlight-line">
<span class="highlight-cl">
    &lt;!--param-value-->classpath*/applicationContext-mvc.xml</param-value-->
</span>
</span>
<span class="highlight-line">
<span class="highlight-cl">
    &lt;!--init-param-->
</span>
</span>
<span class="highlight-line">
<span class="highlight-cl">
    &lt;!--load-on-startup-->1</load-on-startup-->
</span>
</span>
<span class="highlight-line">
<span class="highlight-cl">
    &lt;!--servlet-mapping-->
</span>
</span>
<span class="highlight-line">
<span class="highlight-cl">
    &lt;!--servlet
name-->springmvc</servlet-name-->
</span>
</span>
<span class="highlight-line">
<span class="highlight-cl">
    &lt;!--url-pat
```

```

ern&gt;*action&lt;/url-pattern&gt;
</span></span><span class="highlight-line"><span class="highlight-cl">&lt;/servlet-
apping&gt;&lt;/pre&gt;
</span></span></code></pre>
<p>
  <br>
</p>
<p>
  在工程的resources目录下applicationContext-mvc.xml
</p>
<pre><pre class="prettyprint lang-xml">&lt;?xml version="1.0" encoding="UTF-8"&gt;
&lt;beans xmlns="http://www.springframework.org/schema/beans" xmlns:xsi="http://www.w
.org/2001/XMLSchema-instance" xmlns:context="http://www.springframework.org/schema/c
ncontext"
  xmlns:mvc="http://www.springframework.org/schema/mvc"
  xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springf
amework.org/schema/beans/spring-beans-3.2.xsd
  http://www.springframework.org/schema/context http://www.springframework.org/sch
ma/context/spring-context-3.2.xsd
  http://www.springframework.org/schema/mvc http://www.springframework.org/schema
mvc/spring-mvc-3.2.xsd"
  default-lazy-init="true" default-autowire="byName"&gt;
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight
cl">&lt;!-- 公用属性文件 --&gt;
</span></span></code></pre>
<p>&lt;context:property-placeholder location="classpath*:important.properties" ignore-res
ource-not-found="true" ignore-unresolvable="true" /&gt;</p>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight
cl">&lt;!-- 指定注解扫描的包 --&gt;
</span></span></code></pre>
<p>&lt;context:component-scan base-package="org.sigon" use-default-filters="false"&gt;<b
>
&lt;context:include-filter type="annotation" expression="org.springframework.stereotype.Con
roller" /&gt;<br>
&lt;/context:component-scan&gt;</p>
<p>&lt;!-- MVC 拦截器 --&gt;<br>
&lt;mvc:interceptors&gt;<br>
&lt;mvc:interceptor&gt;<br>
&lt;mvc:mapping path="/**" /&gt;<br>
&lt;mvc:exclude-mapping path="/login/login.action" /&gt;<br>
&lt;bean id="logInterceptor" class="com.aunewtop.merchant.web.interceptor.LogInterceptor"
/&gt;<br>
&lt;/mvc:interceptor&gt;<br>
&lt;/mvc:interceptors&gt;</p>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight
cl">&lt;!-- bean id="viewResolver" class="org.springframework.web.servlet.view.freemarker
FreeMarkerViewResolver"&gt;
</span></span><span class="highlight-line"><span class="highlight-cl">  &lt;!-- proper
y name="contentType" value="text/html; charset=UTF-8" /&gt;
</span></span><span class="highlight-line"><span class="highlight-cl">  &lt;!-- proper
y name="suffix" value="{template.suffix}" /&gt;
</span></span><span class="highlight-line"><span class="highlight-cl">&lt;!-- bean&a
p;gt;
</span></span><span class="highlight-line"><span class="highlight-cl">

```

```

</span></span><span class="highlight-line"><span class="highlight-cl">&lt;bean id=
multipartResolver" class="org.springframework.web.multipart.commons.CommonsMultipartR
solver"&lt;/span></span><span class="highlight-line"><span class="highlight-cl"> &lt;lt;proper
y name="defaultEncoding" value="UTF-8" /&lt;/span></span><span class="highlight-line"><span class="highlight-cl"> &lt;lt;proper
y name="maxUploadSize"&lt;/span></span><span class="highlight-line"><span class="highlight-cl"> &lt;lt;valu
&lt;10485760&lt;/value&lt;/span></span>&lt;!-- 文件大小限为10M --&lt;/span></span><span class="highlight-line"><span class="highlight-cl"> &lt;lt;/prope
ty&lt;/span></span><span class="highlight-line"><span class="highlight-cl">&lt;/bean&a
p&lt;/span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">&lt;bean id=
exceptionResolver" class="org.sigon.demo.web.exception.ExceptionHandler"&lt;/span></span><span class="highlight-line"><span class="highlight-cl"> &lt;lt;proper
y name="defaultErrorView" value="/msg/error"/&lt;/span></span><span class="highlight-line"><span class="highlight-cl"> &lt;lt;proper
y name="defaultStatusCode" value="500"/&lt;/span></span><span class="highlight-line"><span class="highlight-cl"> &lt;lt;proper
y name="statusCodes"&lt;/span></span>&lt;!-- 配置多个statusCode --&lt;/span></span><span class="highlight-line"><span class="highlight-cl"> &lt;lt;prop
&lt;/span></span><span class="highlight-line"><span class="highlight-cl"> &lt;lt;p
op key="/msg/maxUploadSize"&lt;/span></span>500&lt;/prop&lt;/span></span>&lt;!-- error.jsp --&lt;
mp&lt;/span></span><span class="highlight-line"><span class="highlight-cl"> &lt;lt;p
op key="/msg/error"&lt;/span></span>404&lt;/prop&lt;/span></span>&lt;!-- error1.jsp --&lt;/span></span><span class="highlight-line"><span class="highlight-cl"> &lt;lt;/pro
s&lt;/span></span><span class="highlight-line"><span class="highlight-cl"> &lt;lt;/prope
ty&lt;/span></span><span class="highlight-line"><span class="highlight-cl"> &lt;lt;proper
y name="exceptionMappings"&lt;/span></span><span class="highlight-line"><span class="highlight-cl"> &lt;lt;prop
&lt;/span></span><span class="highlight-line"><span class="highlight-cl"> &lt;lt;p
op key="org.springframework.web.multipart.MaxUploadSizeExceededException"&lt;/span></span>ms
/maxUploadSize&lt;/prop&lt;/span></span><span class="highlight-line"><span class="highlight-cl"> &lt;lt;p
op key="java.lang.exception"&lt;/span></span>msg/error&lt;/prop&lt;/span></span><span class="highlight-line"><span class="highlight-cl"> &lt;lt;/pro
s&lt;/span></span><span class="highlight-line"><span class="highlight-cl"> &lt;lt;/prope
ty&lt;/span></span><span class="highlight-line"><span class="highlight-cl">&lt;/bean&a
p&lt;/span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">&lt;mvc:anno
ation-driven&lt;/span></span><span class="highlight-line"><span class="highlight-cl"> &lt;lt;mvc:m
essage-converters&lt;/span></span><span class="highlight-line"><span class="highlight-cl"> &lt;lt;bean

```

```

class="org.springframework.http.converter.StringHttpMessageConverter"&gt;>
</span></span><span class="highlight-line"><span class="highlight-cl">      &lt;it;p
operty name="supportedMediaTypes"&gt;>
</span></span><span class="highlight-line"><span class="highlight-cl">      &lt;it;
ist&gt;>
</span></span><span class="highlight-line"><span class="highlight-cl">      &lt;it;
lt;value&gt;>text/plain;charset=UTF-8&lt;/value&gt;>
</span></span><span class="highlight-line"><span class="highlight-cl">      &lt;it;
lt;value&gt;>text/html;charset=UTF-8&lt;/value&gt;>
</span></span><span class="highlight-line"><span class="highlight-cl">      &lt;it;
list&gt;>
</span></span><span class="highlight-line"><span class="highlight-cl">      &lt;it;/p
operty&gt;>
</span></span><span class="highlight-line"><span class="highlight-cl">      &lt;it;/bea
&gt;>
</span></span><span class="highlight-line"><span class="highlight-cl">      &lt;it;/mvc:
essage-converters&gt;>
</span></span><span class="highlight-line"><span class="highlight-cl">      &lt;it;/mvc:ann
tation-driven&gt;>
</span></span></code></pre>
<p>&lt;bean id="freeMarkerConfigurer" class="org.springframework.web.servlet.view.freema
ker.FreeMarkerConfigurer"&gt;<br>
&lt;property name="templateLoaderPaths" value="{template.loader_path}" /&gt;<br>
&lt;property name="freemarkerSettings"&gt;<br>
&lt;props&gt;<br>
&lt;prop key="defaultEncoding"&gt;{template.encoding}&lt;/prop&gt;<br>
&lt;prop key="tag_syntax"&gt;auto_detect&lt;/prop&gt;<br>
&lt;prop key="whitespace_stripping"&gt>true&lt;/prop&gt;<br>
&lt;prop key="classic_compatible"&gt>true&lt;/prop&gt;<br>
&lt;prop key="template_update_delay"&gt;{template.update_delay}&lt;/prop&gt;<br>
&lt;prop key="object_wrapper"&gt;freemarker.ext.beans.BeansWrapper&lt;/prop&gt;<br>
&lt;prop key="datetime_format"&gt;{template.datetime_format}&lt;/prop&gt;<br>
&lt;prop key="date_format"&gt;{template.date_format}&lt;/prop&gt;<br>
&lt;prop key="time_format"&gt;{template.time_format}&lt;/prop&gt;<br>
&lt;prop key="number_format"&gt;{template.number_format}&lt;/prop&gt;<br>
&lt;prop key="boolean_format"&gt;{template.boolean_format}&lt;/prop&gt;<br>
&lt;prop key="auto_import"&gt;include/spring.ftl as spring&lt;/prop&gt;<br>
&lt;/props&gt;<br>
&lt;/property&gt;<br>
&lt;property name="freemarkerVariables"&gt;<br>
&lt;map&gt;<br>
&lt;entry key="systemName" value="{system.project_name}" /&gt;<br>
&lt;entry key="systemVersion" value="{system.version}" /&gt;<br>
&lt;entry key="systemDescription" value="{system.description}" /&gt;<br>
&lt;entry key="locale" value="{locale}" /&gt;<br>
&lt;!-&lt;entry key="base" value="{serverside.domain.url}" /&gt;--&gt;<br>
&lt;entry key="base" value="" /&gt;<br>
&lt;/map&gt;<br>
&lt;/property&gt;<br>
&lt;/bean&gt;<br>
&lt;/beans&gt;</p></pre><br>
</pre><p></p>
<h2>

```

三.Java 控制器

```

</h2>
<p>
  <br>
</p>
<pre class="prettyprint lang-java">package org.sigon.demo.web.action;
<p>import org.springframework.beans.factory.annotation.Autowired;<br>
import org.springframework.context.annotation.Scope;<br>
import org.springframework.stereotype.Controller;<br>
import org.springframework.ui.ModelMap;<br>
import org.springframework.web.bind.annotation.PathVariable;<br>
import org.springframework.web.bind.annotation.RequestMapping;<br>
import org.springframework.web.bind.annotation.RequestMethod;<br>
import org.springframework.web.bind.annotation.ResponseBody;</p>
<p>import javax.annotation.Resource;<br>
import javax.servlet.http.HttpServletRequest;</p>
<p>/**</p>
<ul>
<li>
<p>注解：控制器名，映射路径，原型实例</p>
</li>
<li>
<p>User: Sguang</p>
</li>
<li>
<p>Date: 15-10-26</p>
</li>
<li>
<p>Time: 下午 11:17</p>
</li>
<li>
<p>To change this template use File | Settings | File Templates.<br>
*/<br>
@Controller("testController")<br>
@RequestMapping("/test")<br>
@Scope(value = "prototype")<br>
public class TestController extends BaseController {</p>
<p>@Resource(name = "testServiceImpl")<br>
private TestService testService;</p>
<p>/**</p>
<ul>
<li>@param id 路径参数</li>
<li>@param model 给 Freemarker 模板封装的 Map</li>
<li>@param request 需要使用 request 的话,只需要在参数中加上就可以</li>
<li>@return 返回值是 Freemarker 模板的相对路径<br>
**/<br>
@RequestMapping(value = "/method/{id}", method = RequestMethod.GET)<br>
public String list(@PathVariable Long id, ModelMap model, HttpServletRequest request) {<br>

model.addAttribute("str", "SPRING MVC");<br>
model.addAttribute("id", id);<br>
model.addAttribute("testValue", testService.getTestValue());<br>
return "/test/method";<br>
}</li>
</ul>

```

```
<p>/**</p>
<ul>
<li>使用 ResponseBody 注解,会直接返回值,适合 ajax 时返回 json 数据使用<br>
**/<br>
@ResponseBody<br>
@RequestMapping(value = "/userAgent", method = RequestMethod.GET)<br>
public String userAgent(){<br>
return request.getHeader("User-Agent");<br>
}<br>
}</li></ul></li></ul></pre>
```

```
<p>
<br>
</p>
<pre><span>SpringMVC的Controller设计非常优雅灵活,能随心所欲的配置参数和返回值</span>
/pre>
<h2>
```

四.Freemarker例子

```
</h2>
```

```
<p>
```

对应的Freemarker模板就很简单了

```
</p>
```

```
<pre><pre class="prettyprint lang-html">${str} ${id} ${testValue}
[#list [1,2,3] as item]
${item}
[/#list]</pre>
```

```
<br>
```

```
</pre>
```

```
<p>
```

```
<br>
```

示例里演示了从Controller传下来的model属性的调用方法

还有List的迭代器,更多Freemarker的用法可以去百度


```
<br>
```

总结:这套整合配置是非常简单的,使用注解的方式省了写属性访问器.也更方便阅读.

```
</p>
```

```
</pre>
```