



页面布局Sitemesh3.0和Tiles3.0差异

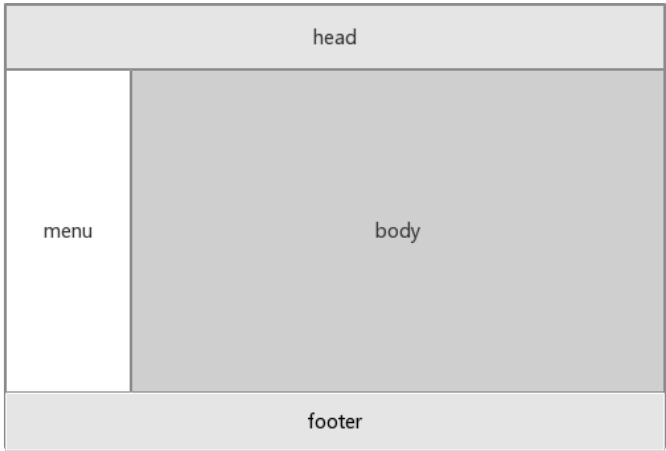
2015-10-09 00:00 it168网站 原创 作者: 周亚培 编辑: 崔月

0条评论

【IT168 技术】在页面开发过程中，经常会遇到页面组合情况，即一个页面有几个部分组成，某些部分内容是固定不变的，某些部分是可以进行替换或者页面跳转的，常常要求同一站点的所有Web页面保持一致的外观，有相同的布局、页头、页尾和菜单。如果不使用页面布局框架单独开发每个页面，既浪费时间又容易出现风格不统一的情况，若使用Iframe实现，则要面临页面控制的复杂操作，为了实现页面的可重用性，提高开发效率，使用页面布局框架Sitemesh或者Apache Tiles是个很好的选择。但是很多开发人员对于使用哪种布局框架更适合自己的项目存在困惑，本文基于Sitemesh和Tiles的实现原理来讲解两个框架有何异同，来剖析两个框架的优劣以帮助开发人员做最适合的抉择。

一、Sitemesh和Tiles的介绍

很多应用程序的布局如下图，在开发过程中，如果每个页面单独组合自己，会耗费大量的时间来做重复性的开发，因此使用页面组合框架，减少页面开发的复杂度和开发时间，是较好的选择，最常用的就是Sitemesh和Tiles。



Sitemesh通过过滤器(filter)来拦截页面访问,根据被访问页面的URL找到合适的装饰模板页面,提取被访问页面的内容，放到装饰模板中合适的位置,最终将装饰后的页面发送给客户端。在sitemesh中，页面分为两种：装饰模板和普通页面。装饰模板，是指用于修饰其它页面的页面。普通页面，一般指各种应用页面，这些逻辑来自一组共享的(或者自定义的)插件。它基于Web页面布局、装饰以及与现存Web应用整合的框架。它能帮助我们在由大量页面构成的项目中创建一致的页面布局和外观，如一致的导航条，一致的banner，一致的版权，等等。它不仅仅能处理动态的内容，如jsp，php，asp等产生的内容，它也能处理静态的内容，如html的内容，使得它的内容也符合你的页面结构的要求。甚至于它能将HTML文件象include那样将该文件作为一个面板的形式嵌入到别的文件中。Sitemesh是Decorator模式的最生动的实现，它是由java语言来实现的，能与其他语言的Web应用很好地集成。

广告



IT168企业级

IT168文库

文库文集
小白必读：PHP菜鸟入门教程
Linux服务器入门基础知识集锦！
Django从入门到精通系列教程
Java框架型项目从入门到装逼
Oracle DBA 手记1
阮一峰 React 技术栈系列教程
CTDC 2017 首席技术官领袖峰会
ArchSummit全球架构师峰会 2017
IBM面试题集合
基于openpower的解决方案——软硬件

一周热文
于企业应用程序而言，Go比Java更明智！
三大角度PK，Go语言和Node.js谁胜谁负？
40位软件名人榜单出炉，60后占半壁江山
MBT测试思想在苏宁蛙测的运用实践分享
亚马逊CTO:我们为何要提供这么多的数据

解析：数据库开发如何向DevOps模式转换

青云QingCloud正式推出SD-WAN服务 实现

Tiles框架为创建Web页面提供了一种模板机制，它可将网页的布局和内容分离。Tiles模板是一种描述页面布局的JSP页面，它只定义了Web页面的样式，而不指定内容，然后在运行时动态地将内容插入到模板中。其中的模版和tiles组件均具有重复可用性，并且各组件可以像搭积木一样组合成新的组件。

本文在SpringMVC中针对如上图的情况来分析两种框架的实现。

二、Sitemesh3.0的配置实现

Sitemesh通过以下步骤实现上述图片的功能。

1)①下载最新siteMesh.jar文件放到项目lib包中：<https://github.com/sitemesh/sitemesh3/downloads>

② 若使用maven,在pom.xml文件配置(也可以放在maven私有库中)：

```
<dependency>
<groupId>org.sitemesh</groupId>
<artifactId>sitemesh</artifactId>
<version>3.0.0</version>
</dependency>
```

2)web.xml文件中添加 Sitemesh Filter

```
<web-app>
<filter>
<filter-name>sitemesh</filter-name>
<filter-class>org.sitemesh.config.ConfigurableSiteMeshFilter</filter-class>
</filter>
<filter-mapping>
<filter-name>sitemesh</filter-name>
<url-pattern>/*</url-pattern>
</filter-mapping>
</web-app>
```

3)添加/WEB-INF/decorators.xml，其中可以定义多个模板文件，使用不同的规则装饰指定类型的页面

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- 模板文件存放路径 -->
<decorators defaultdir="/WEB-INF/views">
<!-- 装饰页面, 在需要装饰的页面增加<meta name="decorator" content="mydecorator"/> -->
<decorator name="mydecorator" page="layouts/mydecorator.html" />
<!-- 第二个模板文件和匹配使用的被装饰文件 -->
<decorator name="mydecorator2" page="layouts/myTest.html">
<pattern>/showinfo.html*</pattern>
</decorator>
</decorators>
```

4)准备页面

①mydecorator.html-“装饰页面”，应用中每个页面都包含的样式页面,div中的页面可以使用css来控制div的样式实现布局的位置。

```
<!DOCTYPE html>
<head>
  <title></title>
</head>
<div>
  <div>Header</div>
  <hr />
<div>Menu部分的页面内容</div>
  <hr />
  <!-- 变化部分页面内容在这个位置 -->
  <decorator:div />
  <hr />
  <div>Footer部分的页面内容</b></div>
</div>
</html>
```

②Body.html - “被装饰的页面”，实际要呈现的内容页。

```
<!DOCTYPE html>
<html>
<head>
  <meta name="decorator" content="mydecorator"/>
</head>
<div>
  内容页的div部分，这部分内容会被加载到mydecorator.html的 <decorator:div />部分
</div>
</html>
```

三、Tiles的配置实现

Tiles通过以下步骤实现上述图片的功能。

1)①所需要jar包为tiles-api.jar,tiles-core.jar,tiles-test.jar,tiles-jsp.jar， Tiles3.0需要JDK6.0以上版本才支持，根据JDK版本选择相应版本的jar包放在lib文件夹下，下载地址：<http://tiles.apache.org/download.html>。

②若使用Maven,在pom文件中配置如下：

```
<dependency>
  <groupId>org.apache.tiles</groupId>
  <artifactId>tiles-extras</artifactId>
  <version>3.0.1</version>
</dependency>
```

2)在Spring mvc 中配置Tiles框架

要配置Tiles，在spring-servlet.xml中配置一个TilesConfig的<bean>。

```
<bean id="tilesConfigurer" class="org.springframework.web.servlet.view.tiles3.TilesConfigure
r">
    <property name="definitions">
        <list>
            <value>/WEB-INF/tiles/general.xml</value>
        </list>
    </property>
</bean>
<bean id="viewResolver" class="org.springframework.web.servlet.view.UrlBasedViewResolv
er">
    <property name="order" value="1" />
    <property name="viewClass" value="org.springframework.web.servlet.view.tiles3.TilesView
"></property>
</bean>
<bean
    class="org.springframework.web.servlet.view.InternalResourceViewResolver">
    <property name="prefix" value="/WEB-INF/views/" />
    <property name="suffix" value=".jsp" />
    <property name="order" value="2" />
</bean>
```

3) 在WEB-INF下配置"tiles.xml",该文件定义了Tiles模板

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE tiles-definitions PUBLIC "-//Apache Software Foundation//DTD Tiles Configuratio
n 3.0//EN" "http://tiles.apache.org/dtds/tiles-config_3_0.dtd">
<tiles-definitions>
    <!-- 前端页面模板 -->
    <definition name="template_page" template="/WEB-INF/templates/layout.jsp">
<put-attribute name="header" value="/WEB-INF/views/header.jsp" />
<put-attribute name="menu" value="/WEB-INF/views/menu.jsp" />
<put-attribute name="div" value="" />
<put-attribute name="footer" value="/WEB-INF/views/footer.jsp" />
    </definition>
    <!-- 引用模板 视图名 + .page -->
    <definition name="*.page" extends="template_page" >
        <put-attribute name="div" value="/WEB-INF/views/{1}.jsp" />
    </definition>
```

在tiles.xml中定义模板tiles-definitions。该布局包含的属性，如头，标题，主体，菜单和页脚。然后扩展和布局的新关联页面，使用我们定义的页面覆盖tiles默认的布局改变Body和Title的内容。

4)创建布局页面layout.jsp和各个属性页面，用div的css样式控制header、menu、div和footer页面的位置。Tiles3.0可以定义能够重用的布局子页面，也可以只定义可重用的布局样式，然后在不同页面的相同位置加载不同的页面内容。

```
<%@ taglib uri="http://tiles.apache.org/tags-tiles" prefix="tiles"%>
```

```
<!DOCTYPE HTML>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

</head>

<div>

<div><tiles:insertAttribute name="header" /> </div>

<div><tiles:insertAttribute name="menu" /></div>

<div><tiles:insertAttribute name="div" /></div>

<div><tiles:insertAttribute name="footer" /> </div>

</div>

</html>
```

5)Tiles2.0和Tiles3.0有了较大的区别，在3.0以前的版本中，配置文件的配置方法比较固定繁琐，需要对每一个页面都进行模板配置，而在3.0中，可以使用通配符来实现多个页面的相同配置调用，该功能已经类似了Sitemesh的配置方法，使用起来更加快捷方便。

四、Sitemesh和Tiles的实现原理

1)Sitemesh实现原理：

一个请求到服务器后，如果该请求需要sitemesh装饰，服务器先解释被请求的资源，然后根据配置文件获得用于该请求的装饰器，最后用装饰器装饰被请求资源，将结果一同返回给客户端浏览器。

SiteMesh采用filter实现，实现了下面功能：

对于页面被SiteMesh包装输出的页面代码，sitemesh进行解析，然后根据配置进行代码修改。提取网页中重复相同的html、js、css等代码。SiteMesh提取web页面的公共部分，把页面的公共代码提取出来，可以灵活配置，方便页面的开发、修改。

对于页面根据环境的不同，自动的进行定制修改，可以根据浏览器类型、用户语言、搜索引擎、操作系统等进行不同的页面装饰。

SiteMesh要先在web.xml中配置filter，让Sitemesh可以过滤页面，然后需要写一个配置文件sitemesh.xml，从上面的文件内容可以看出，该文件主要配置哪些页面需要做装饰，哪些页面不需要装饰。

SiteMesh的主要处理过程如下：

Servlet Container进行正常的页面处理，各种filter执行

在执行到sitemesh的filter时，根据配置确定是否需要执行

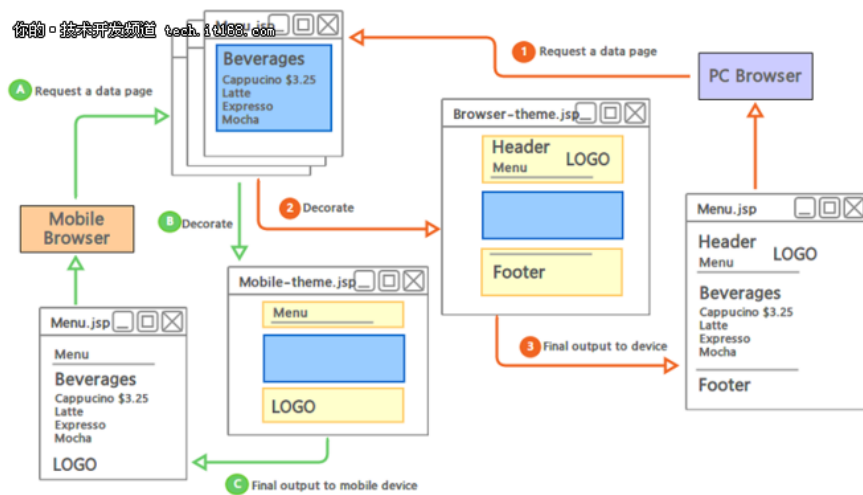
sitemesh构造PageResponseWrapper，然后执行之后的各种filter

通过PageResponseWrapper得到需要进行装饰的页面对象，PageResponseWrapper调用配置的Parser进行页面解析

调用applyDecorator进行页面装饰处理，使用freemarker或者velocity模版作为装饰外框

把处理结果写入到response

其工作原理如下图：



2)Tiles实现原理

用户通过访问一个页面的Apache Tiles定义名， 就可以访问一个由定义文件中包含的页面组成的最终的JSP。

如以下一个Apache Tiles定义文件：

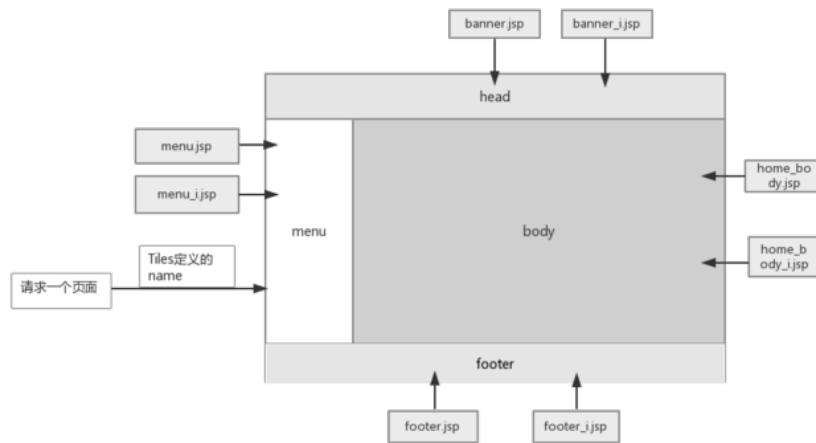
```

<tiles-definitions>
<definition name="myTitles" template="/layouts/classic.jsp">
  <put-attribute name="title" value="Tiles page" />
  <put-attribute name="header" value="/tiles/banner.jsp" />
  <put-attribute name="menu" value="/tiles/menu.jsp" />
  <put-attribute name="div" value="/tiles/home_div.jsp" />
  <put-attribute name="footer" value="/tiles/footer.jsp" />
</definition>
</tiles-definitions>
    
```

其中首先告诉我们， 这个定义的虚拟页面叫做myTitles.用户通过http://site:port/webname/myTitles就能访问到一个页面。这个页面来自的模板是：/layouts/classic.jsp而模板中各个部位真正的值， 则是下面的那些jsp页面。Tiles通过一个组装器， 在页面被请求后开始工作， 按照页面顺序把Value中对应的页面加载完毕， 然后返回， 网站文件中也许根本没有这个请求的资源， 而是临时按照定义拼凑起来的。

其实现原理图如下：

其中head、 menu、 div和footer部分有模板默认的JSP页面， 如果要使用统一风格的模板， 无需再定义这些页面， 如果在这些位置使用的页面风格或者内容不同， 可以使用其他的页面替代这些默认页面。



五、Sitemesh和Tiles的优缺点

使用页面布局框架的出现让我们有更多的时间去关注底层业务逻辑，而不是页面的风格和结构。它让我们摆脱了大量用include方式复用页面尴尬局面，也避免了页面开发在团队开发中的复杂度，它还提供了很大的灵活性以及给我们提供了整合异构Web系统页面的一种方案

sitemesh最令人诟病的一点可能就是它的性能问题，由于使用了装饰器模式，要去解析html代码，会产生很多垃圾对象，所以会频繁的引起GC，CPU的负担会加大。而Tiles基本上没有这样的问题。Sitemesh的测试见：<http://www.iteye.com/topic/715100>。

使用Sitemesh虽然也至少需要两个文件，一个布局文件，一个装饰器映射文件，在映射中也需要配置匹配的URL路径，实际使用上比Tiles简单不了多少。Tiles2.X的版本配置确实复杂，但是通过上面的例子我们可以看见自从Tiles3.0引入通配后，配置简单了很多。

在JSP页面的编写上，使用Sitemesh非常简单，完全是无侵入的，开发人员根本不知道是否用了sitemesh。而使用Tiles的开发人员要注意由于<HTML>这样的标签都定义在了布局jsp中，所以每个tile页面是不能包含它们的。

动态内容的编辑，比如每个页面的title，sitemesh提供了<decorator:title />用来很方便的为你插入title内容。而Tiles则必须单独定义在定义文件中，或者使用通配从controller返回的视图名中解析出来，感觉相当不好。

Tiles的功能更强大一些，比如可以直接在servlet中渲染视图，在运行时改变定义之类的，而且有更好的个性化扩展性，相对来说sitemesh就没那么多功能了。Tiles的学习难度会比sitemesh高一些，sitemesh入手更简单一些。

sitemesh的配置简单，方便快捷，适合的项目应该是主要以jsp作为主控制器的项目，比如项目是以jsp为主去开发，相信sitemesh对这样的项目会如虎添翼!或者你直接采用struts-sitemesh这样的第三方组件，也是可行的。缺点是对底层支持、api支持暂时不够。Tiles在国际化、性能、灵活性方面更胜一筹。

六、总结

总的来说，Tiles和Sitemesh都是非常好的布局框架，具体选择哪一个应该根据项目来定。如果对性能要求比较高，并且对于页面的组合比较灵活的话，还是选择Tiles稳妥些。如果需要开发的页面很多，并且对性能的要求不是特别高，个人认为还是Sitemesh比较适合。

参考资料

- 1.<http://wiki.sitemesh.org/wiki/display/sitemesh/Home>
- 2.<http://tiles.apache.org/index.html>
- 3.<http://seanhe.iteye.com/blog/715100>

作者简介

周亚培（软件开发工程师）

任职于某外资企业，主要从事J2EE开发工作。

标签：[开发](#)，[Tiles](#)，[框架](#)，[Sitemesh](#)

分享到：

0 个人觉得赞

您可能感兴趣的文章

百度智荐

被玩坏的语音交互 叮咚2 这7个开源监控工具 你的 Google Daydream Lab 别具一格 微软Surface L
代智能音箱评测 网络值得拥有 s用ARCore做了什么？ aptop笔记本一周使用体

安全公司炼石网络完成3 使用500天不卡顿 荣耀V MySQL面试题集锦，据 航海王重磅来袭 松下新
千万元Pre-A轮融资 9现货火爆热销中 说知名互联网公司都用 旗舰空净引领新国标

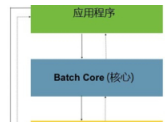
推荐文章

开发经验分享：电商促销后台的逻辑详述



电商所谓营销，归根结底都是订单金额的变化;如果我们清楚的知道订单金额的计算流程是怎样的，那么我们只需 [详情]

一文轻松搞定批处理框架 Spring Batch



在现代企业应用当中，面对复杂的业务以及海量的数据，除了通过庞杂的人机交互界面进行各种处理外，还有一类 [详情]

嵌入式开发下的单元测试问题--框架解析



单元测试是软件开发的重要一环，尤其对嵌入式开发。因为嵌入式开发受限于开发环境、调试工具等因素，不能和 [详情]

分享:GPU云服务器深度学习性能模型初探



本文根据实测数据，初步探讨了在弹性GPU云服务器上深度学习的性能模型，可帮助科学选择C++深度学习框架。

1 2 3 4 5 下一页

大神整理：一篇文章教你读懂微服务架构



资料来源：有架构给我的一些资料，以及自己百度和论坛、社区找来的一些资料，权当做一个总结式的简介。。。[详情]

Hyperledger发布区块链框架Sawtooth 1.0

畅言一下



开源协作区块链Hyperledger宣布发布Sawtooth 1.0，这是该组织的第二个区块链框架，[详情]

还没有评论，快来抢沙发吧！

评论框由畅言提供

编辑推荐



试客发言区:PS调色其实并不难，学好这几步你也是“头号玩家”



试客发言区:论一个资深魅友是如何养成的 盘点用过的那些魅族产品



试客发言区：7年姥爷笔记本升级改造！装上win10再战三年可否？



试客发言区:机械+三模，办公室新宠不为过 雷柏MT700机械键盘



每周都有免费试用

[盛拓传媒简介](#)
[关于IT168](#)
[广告服务](#)
[使用条款](#)
[投稿指南](#)
[诚聘英才](#)
[联系我们](#)
[网站导航](#)
[往日回顾](#)

北京盛拓优讯信息技术有限公司. 版权所有 中华人民共和国增值电信业务经营许可证 编号：京B2-20170206 北京市公安局海淀分局网监中心备案编号：11010802020118

广播电视节目制作经营许可证:(京) 字第07177号 信息安全等级保护备案：11010813655-00001 测绘资质证书:乙测资字11005067 网络文化经营许可证:京网文(2018) 1456-138号