

JSP 静态网页生成技术的研究

徐 白¹, 宋 玲¹, 吴 昊²

(1. 广西大学 计算机与电子信息学院, 广西南宁 530004;

2. 广西壮族自治区电子产品监督检验所, 广西南宁 530031)

摘 要: 动态网页在服务器端以数据库为基础运行程序, 而静态网页不在服务器端执行且不访问数据, 因此静态网页在访问速度方面要快, 使用静态页面生成技术将动态网页静态化, 具有重要意义。文中详细介绍了两种静态页生成技术, 基于模板方法和基于 URL 方法, 其核心就是将需要生成的数据, 转成流输出成文件, 管理人员添加网页时, 利用后台发布程序把页面存成相应 HTML 文件。这种技术已成功应用到某门户网站的新闻浏览模块中, 可以减少服务器端访问数据的压力, 提高网页浏览速度, 提高了数据存取效率, 且生成的静态页更利于被搜索引擎收录和排名靠前。

关键词: 静态网页; JSP; 静态页生成技术; 模板; URL

中图分类号: TP39

文献标识码: A

文章编号: 1673-629X(2010)06-0175-04

Research of Techniques of Static Web Pages in JSP

XU Bai¹, SONG Ling¹, WU Hao²

(1. School of Computer and Electronics Information, Guangxi University, Nanning 530004, China;

2. Guangxi Zhuang Autonomous Region Electronic Products Supervision and Inspection Center, Nanning 530031, China)

Abstract: Dynamic web pages have to run program based on database at the server, while the static web pages neither run program at server nor visit the database. Therefore, static web - pages are of faster visiting speed. It is of great significance to make dynamic web - pages into static pages by applying the generative technology of static web page making. This paper specifically introduces two generative methods of static web - pages, one is based on model and the other is based on URL, and the core is to invert the data required to be generated into stream and output as files. When the webmaster adds the web - pages, he can store the pages in the form of HTML documents by issuing programs on the backstage. This technology has already been applied to a certain news - scanning model in a gateway website, which can reduce the stress of visiting data on the end of the server, enhance the scanning speed of web - pages, and increase the efficiency of data storage. Besides, the generated static pages are beneficial to embodying of search engines and moving ahead in the ranking list.

Key words: static web pages; JSP; static web page making; model; URL

0 引 言

随着现代计算机网络技术的飞速发展, IT 行业也由传统应用向基于 Internet 的 Web 服务模式转化, 动态网站开发技术日渐成为目前主流 Web 开发技术中使用最为广泛的, 通过以数据库为基础的动态网页, 能够降低网站的维护工作量, 并且实现类似用户注册、在线浏览、查询等功能^[1]。但是, 在浏览网站时, 经常出现长时间等待而刷新不出页面的情况。在网络带

宽一定的情况下, 如何通过一些关键性技术改善网站性能、提高浏览速度是值得讨论的问题^[2], 就此, 文中主要讨论了在 JSP(Java Server Page)技术框架下的改善方法。

1 静态页生成的优点

在服务器端运行的程序、网页、组件, 属于动态网页, 它们会随不同客户、不同时间, 返回不同的网页, 一般以数据库为基础。

动态页面技术, 它所实现的动态功能主要就是要将存储在后台数据库中的数据动态地展示给用户, 并将与用户的交互数据存储到后台的数据库中以备以后使用^[3]。也就是说网站中后台数据库存储更新比较频繁的数据, 当客户端向 Web 服务器进行请求时, 服

收稿日期: 2009-09-05; 修回日期: 2009-12-15

基金项目: 广西自然科学基金资助项目(桂科自 0991059); 广西高校人才小高地建设创新团队资助计划项目(桂教人[2007]71号)

作者简介: 徐 白(1983-), 女, 硕士研究生, 研究方向为计算机网络及性能优化; 宋 玲, 教授, 硕士生导师, 研究方向为计算机网络及并行分布式技术。

务器通过执行程序片段,这些程序片段所做操作可以是操作数据库、重新定向网页以及发送 email 等等,服务器的执行结果被重新嵌入到 HTML 代码中,然后一起发送给浏览器。所有程序操作处理都在服务器端执行,通过网络传送给客户端的仅为得到的结果,对客户浏览器的要求低,但服务器端在执行客户端请求的程序片段时往往需要从数据库中找和指令对应的数据,因此需要占用服务器资源,并且随着后台数据库存储的记录不断壮大和访问的人数变多,会明显增大服务器的压力,页面的加载速度就会变慢。

运行于客户端的程序、网页、插件、组件,属于静态网页,静态网页不在服务器端执行,不访问数据。浏览器在访问这样的页面时,直接从服务器端把这些静态文件中的 HTML 代码传递到客户端浏览器即可,所以从浏览速度来讲,相比动态网页,静态网页无须访问数据库,节省了最大的系统消耗。因此对服务器的压力小^[4],因为每一次显示都是一个直接的网页。同时如果使用了静态网页,网站不需要经过数据库就可以打开,所以运行速度更快。可以提高页面打开浏览速度。

从网站优化来分析,搜索引擎更喜欢静态的网页。更便于抓取,更容易被搜索引擎收录和排名靠前^[5]。

从安全角度讲,静态网页可以提高网站安全性,防止不良代码注入,可减少攻击,防 SQL 注入^[6]。也就是说静态网页更不容易受到黑客的攻击。

从网站稳定性来讲,程序或数据库出错时,静态网页不影响网站正常访问^[7],不会因为程序等,而损失网站数据,影响正常打开,损失用户体验,影响网站信任度。

因此,通过某些技术实现网页静态化是具有重要的实际意义和应用价值的。文中讨论的是静态页面生成技术方案。

2 静态页生成技术的实现

静态页生成技术简单来说就是“动态更新内容,静态浏览内容”。将动态页面按一定规则创建相应的纯静态页面,将网页的动态信息和彼此之间的关系存入数据库,根据特定的规则生成网页的各级静态页面包括首页、子页,以静态页面的形式发布,最终完成网页的自动生成。文中以新闻浏览发布系统为例,参照了文献[8~10],介绍两种静态页生成方法。

2.1 模板生成方法

实现思想:提前把网页的公共部分写好,做成一个模板生成。先定义模板文件,用 HTML 格式,其中需要从数据库中提取的部分用标记代替,如 \$ Title \$ 代表在这个位置显示数据库中的 Title 字段,当需要产生

时即信息提交后,用程序去读取模板,把模板文件读入到字符串中,再用提交上来的值替换模板里面的标记,这样可以尽量避免在 java 类中生成 HTML 代码,修改页面的风格直接改模板文件就可以,真正的网页存在网站的目录下^[2,11,12]。

实现的主要步骤如下:

首先设计模板,模板页面可以自己在 dreamweaver 中设计,其中只要将表达的名称、内容、时间用“# name #”,# content #, # time # 代替即可。模板设计好后存放在一定目录下,该目录也是下面的生成函数中用到的参数——模板所在路径。

在本文例中用的模板页代码为:

```
<TABLE> <TBODY>
<TR>
<TD>标题: # name # </TD>
</TR>
<TR>
<TD>发布时间为: # time # </TD>
</TR>
<TR>
<TD>详细内容: # content # </TD>
</TR>
</TBODY></TABLE>
```

其次进行特殊字符处理,因为利用模板方案生成页面时,特殊字符“\$”是不能被写到静态页的。所以在添加或修改记录时需要进行特殊字符处理^[13],实际应用中,用函数将字符串中出现“\$”符号的位置用“\ \$”代替即可。

再次在服务器端编写生成函数:

参数 Fid, name, content, time, sPath, toPath 分别为需要生成动态页面的编号,名称,内容,时间,模板所在路径,生成的静态页所保存的路径,且该静态页以 Fid. htm 为名称。

具体生成的函数如下:

```
public void tofile(String Fid, String name, String content, String
time, String sPath, String toPath)
{
    try
    {
        String templateContent;
        FileInputStream fileInputStream = new FileInputStream(sPath);
        int length = fileInputStream.available(); //available()返回可以不受阻塞地从此文件输入流中读取的字节数
        byte bytes[] = new byte[length];
        fileInputStream.read(bytes); //read(byte[] b)从此输入流中将最多 b.length 个字节的数据读入一个字节数组中
        fileInputStream.close();
```

```

templateContent = new String(bytes);
templateContent = templateContent.replaceAll("# name #",
name);
templateContent = templateContent.replaceAll("# content #",
content);
templateContent = templateContent.replaceAll("# time #",
time);
templateContent = templateContent.replaceAll("# Fid #",
Fid); //替换掉模块中相应的地方
String filename = Fid + ".html";
filename = toPath + filename;
FileOutputStream fileoutputstream = new FileOutputStream
(filename); //建立文件输出流
byte tag_bytes[] = templateContent.getBytes();
fileoutputstream.write(tag_bytes);
fileoutputstream.close();
}
catch(Exception e)
{
e.printStackTrace();
}
}

```

静态页生成函数需要使用 java.io 包,它的实质就是将要生成的数据转成流输出成文件,要求在数据添加初期就生成 html 文件。具体应用中,在后台管理的增加修改或删除记录时,随之进行记录的页面生成操作,按定义好的模板生成 HTML 文件,该文件名按添加记录的 ID 号命名,调用前面步骤中的生成函数,在调用生成函数前,先调用字符过滤函数进行特殊字符处理。添加的信息写入到数据库中。

在客户端浏览器,用户点击相应的信息标题时,就可以通过链接地址找到静态页面的路径,若选择某条记录名称后,期望显示该记录对应的详细信息,该信息作为子页被生成静态页,只需读取生成好的静态页面就行。

2.2 URL(Uniform Resource Locator)方法

这种方法实现思想为:先使用 java.net 包的 URL 类,把动态页面 xxx.jsp? yyy = yy&zzz = zz 获取下来转成流^[14],然后用 java.io.File 方法^[15]把获取的流输出成 html 文件。

核心程序如下:

函数中参数 sourcePath 代表需要访问的动态页面的 URL 地址,参数 destPath 代表需保存的生成好的 HTML 页面的路径加名称;

```

public static boolean produceFromURL(String sourcePath, String
destPath)
{

```

```

RandomAccessFile destFile = null;
InputStream inStream = null;
try {
destFile = new RandomAccessFile(destPath, "rw");
URL url = new java.net.URL(sourcePath);
URLConnection conn = url.openConnection();
inStream = conn.getInputStream();
byte[] buffer = new byte[512];
int lineRead = 0;
while((lineRead = inStream.read(buffer, 0, 512)) != -1) {
destFile.write(buffer, 0, lineRead);
}
}
catch (Exception e) {
e.printStackTrace();
return false;
}
finally {
try {
if (destFile != null) {
destFile.close();
}
}
catch (Exception e) {
}
try {
if (inStream != null) {
inStream.close();
}
}
catch (Exception e) {
}
}
return true;
}

```

该函数需要引用 java.io 包和 java.net 包^[15],调用时按如下例调用:

若需要生成首页,参数 sourcePath 值取首页的 URL 地址,参数 destPath 值取需保存静态页面的名称,该名称是带路径的。例如本测试实验中,将主页生成后放在原来主页所在路径,具体调用函数如下:

```

produceFromURL("http://localhost:8081/myprj/default.jsp", request.
getRealPath("/default.html"))

```

若需要生成的为子页面,参数 sourcePath 值取子页的 URL 地址,参数 destPath 值取需保存静态页面的名称,该名称是带路径的,并能唯一标识该子页。例如本测试实验中,将生成子页的静态页也就是生成某条记录的静态页,生成后放在指定的路径下。则调用函

数可写为:

```
produceFromURL("http://localhost:8081/myprj/ShowDetail.jsp?
id=166",request.getRealPath("/JT/WX/Detail166"))
```

项目中实际应用中,在后台管理增加修改或删除记录时,随之进行记录的页面生成操作,特别地,为了顺利生成某条记录的静态页,在服务端后台增加或修改记录时,当时获取记录的 id,便写入 sourcePath 和 destPath 参数,此时便能生成具体的某条记录的静态页面。

例如,在增加或修改记录时只需这样调用:

```
produceFromURL("http://localhost:8081/myprj/
ShowDetail.jsp?id=" + id,request.getRealPath("/JT/
WX/Detail") + id + ".htm"),便生成了该记录的静态
页面。
```

在浏览器客户端,若用户浏览首页,直接链接到首页静态页的地址就可以;若用户点击相应的信息标题,需要通过链接地址找到静态页面的路径,在首页中信息标题的链接设置为相应子页所在路径,例:<A href ="/JT/WX/Detail< % = id% > .html">,便能链接到相应的静态子页。

3 结束语

现在,静态页生成技术已经很成熟,应用也日益广泛,许多大型网站都相应采用这种技术。在设计网站时,在 JSP 页面时可以进行处理,先检测是否存在相应的静态页面,如果有则转向,没有则立即生成,然后转向,这个检测的频度可以用临时文件处理。这样的话,即使后台有些页面没有实现或者不是很好实现静态页面与实时数据的影响问题,也可以生成静态页面,从而起到提高网页访问速度的目的。

参考文献:

- [1] Morrison M, Morrison J, Anthony. Keys Integrating Web Sites

And Database[J]. Communications of the ACM, 2002, 45(9): 81 - 86.

- [2] 白金牛,李慧萍,王培吉. ASP.NET 下利用动态网页技术生成静态 HTML 页面的方法[J]. 计算机应用与软件, 2008 (1): 79 - 81.
- [3] Eldridge L. Dynamic vs Static web Pages[EB/OL]. 2001 - 07 - 05. <http://www.loriswebs.com/dynamicstatic.html>.
- [4] Knight J. Knightnet Site Design - Static vs Dynamic Web Pages[EB/OL]. 2008 - 07. <http://www.knightnet.org.uk/site-design/static-vs-dynamic-pages.htm>.
- [5] Scanlon V. Dynamic Or Static Web Pages, Which Way Should You Go [EB/OL]. 2006 - 11 - 30. <http://ezinearticles.com/?Dynamic-Or-Static-Web-Pages,-Which-Way-Should-You-Go?&id=373266>.
- [6] 董 斌. 静态页面生成的网站系统研究[J]. 福建电脑, 2009(8): 160 - 161.
- [7] ExSite Webware Content Management. Static vs. Dynamic Content [EB/OL]. 2008 - 12 - 24. <http://support.exsitewebware.com/cgi/page.cgi/articles.html/Content-Management/Static-vs-Dynamic-Content>.
- [8] 马 强,宋 玲. 基于 Web 的社保新闻发布系统的设计与实现[J]. 计算机技术与发展, 2007, 17(12): 31 - 33.
- [9] 高 翔,何立军,李国兴. JSP 动态网站开发技术与实践[M]. 北京:电子工业出版社, 2007: 245 - 293.
- [10] 汪孝宜,刘中兵,徐佳晶,等. JSP 数据库开发实例精粹[M]. 北京:电子工业出版社, 2005: 145 - 209.
- [11] 曾春华,江南雨. 动态生成静态网页技术探索[J]. 计算机与网络, 2008(24): 511 - 512.
- [12] 许冀伟,李广霞,傅王月. 一种基于 ASP.NET 技术生成新闻静态页的方法[J]. 计算机与网络, 2007(33): 214 - 214.
- [13] Decoder. JSP 技术揭秘[M]. 北京:清华大学出版社, 2001: 140 - 156.
- [14] 荣钦科技. JSP 动态网站开发与实例[M]. 第 3 版. 北京:清华大学出版社, 2006: 279 - 287.
- [15] Liang Y D. Java 编程原理与实践[M]. 第 4 版. 北京:清华大学出版社, 2005: 665 - 716.

(上接第 132 页)

识逻辑检测[J]. 农业网络信息, 2008(9): 138 - 141.

- [3] 吉 喆,李绍稳,张友华,等. 基于本体的茶虫害诊断系统构建的研究[J]. 农业网络信息, 2008(9): 112 - 116.
- [4] OWL Web Ontology Language Reference[EB/OL]. 2004 - 10. <http://www.w3.org/TR/owl-ref/>.
- [5] 王晓东,张 合,王红涛. 基于 Ontology 的语义信息检索模型研究[J]. 计算机工程与设计, 2008, 29(11): 2939 - 2941.
- [6] 钱 平,郑业鲁. 农业本体论研究与应用[M]. 北京:中国农业科学技术出版社, 2006: 89 - 98.
- [7] 鲜国建,孟宪学,常 春. 基于农业本体的智能检索原型系统设计与实现[J]. 中国农学通报, 2008, 24(6): 470 - 474.
- [8] XMLDOM, XPath[EB/OL]. 2008 - 10. [http://msdn.microsoft.com/zh-cn/87274khy\(VS.80\).aspx](http://msdn.microsoft.com/zh-cn/87274khy(VS.80).aspx).
- [9] Google Ajax Search API[EB/OL]. 2008 - 04. <http://code.google.com/intl/zh-CN/apis/ajaxsearch/>.