

Servlet在图书馆Web系统中的应用

易俊刚

(中国科学院文献情报中心 北京 100080)

[摘要] 从理论和应用的角度探讨 Servlet 技术在图书馆 Web 系统建设中的作用。通过一个简单的基于 Web 的图书管理系统的设计和实现, 阐述具体利用 Servlet 技术构造 Web 系统的方法和步骤。

[关键词] Servlet Web 系统 数据库

[分类号] G250.7

The Application of Servlet to the Library's Web System

Yi Jungang

(Documentation and Information Centre, the Chinese Academy of Sciences, Beijing)

[Abstract] Discusses the role of Servlet in the construction of library's Web system. The author describes the method and process of constructing Web system with Servlet technology by designing and implementing a simple book management system.

[Keywords] Servlet Web system database

在传统的 Web 系统中, 服务器的系统结构一般都是两层, 即 Web 服务器层和数据库服务器层。这样, 在用户每次提交一个 Http/CGI 请求, 或者调用一个 CGI 程序的时候, 服务器端就需要新启用一个系统进程。这样在网站访问量大的时候, 系统资源就消耗得越来越多, 从而导致性能迅速下降, 出现死机、数据库死锁等现象。随着 Web 技术的深入发展, Web 系统出现了 3 层或多层应用模式, 其中 3 层应用模式为前端浏览器、中间层和后台数据库服务器。这种结构将数据的表示逻辑和商业逻辑独立开来, 极大地增强了系统的灵活性和可扩展性。无论是在 Microsoft 的 Windows DNA 中, 还是在 SUN 公司推出的以 Java 为核心的 J2EE 体系中, 这种 3 层或多层结构都得到了具体的体现。Servlet (简称服务端小应

用程序) 技术作为 J2EE 体系中的重要组成部分, 它的出现推动了以 Java 为核心技术的企业级多层 Web 应用的发展, 更利于开发人员开发与 Web 服务器紧密相关的多层应用。

1 Servlet 的 3 层应用结构

Servlet 技术的 3 层应用结构包含客户端 Web 浏览器、Servlet 中间层和后台数据库服务器。

1.1 客户端浏览器

Web 浏览器是 3 层应用结构的客户端, 是 Web 系统用户直接接触到的部分, 它提供了一个统一的访问系统的可视用户界面和系统运行环境, 极大地简化了客户端的安装和维护。

1.2 Servlet 中间层

中间层是运行在 Web 服务器中

的, 联系客户端浏览器与后台数据库的应用模块。一方面, Servlet 能够像 CGI 脚本一样扩展 Web 服务器的功能, 充分利用服务器端的资源, 并且同样具备“一次编写, 多平台运行”的特点。另一方面, Servlet 由 Web 服务器进行加载, 在系统性能、可靠性以及可移植性等方面均比 CGI 有很大的进步, 是目前最易实现的中间层技术。

1.3 数据库服务器

后台数据库服务器是存放企业和用户数据的地方, 中间层可以通过 ODBC (对 CGI 中间层) 或是 JDBC (对 Servlet 中间层) 来访问后台数据库。

2 3 层结构的图书管理系统

为了说明如何使用 Servlet 技术构造中间层, 我们设计了一个简单的基于

收稿日期: 2000—12—14

3层应用结构的图书管理系统。它的功能包括:

- 馆员通过前端浏览器将书目信息提交给中间层,并由中间层写入数据库。

- 读者通过浏览器查询书目信息,通过中间层从数据库中取得书目数据。

系统的前端为一般常用的浏览器(IE、Netscape等),中间层是Servlet,后台数据库采用Access。系统各层次的交互过程如下:

- 馆员在HTML表格中输入书目信息,数据被传输到在Web服务器上运行的Servlet程序(LibraryServlet)中;

- 中间层LibraryServlet对表格的数据进行分析,动态构建一个SQL语句,并与JDBC关联;

- 后台数据库执行SQL语句并将一组结果返回给中间层LibraryServlet;

- 中间层LibraryServlet对返回的结果进行处理,并用其中的数据为读者构建一个返回页面。

对系统的功能、交互过程了解之后,我们给出整个图书管理系统的实现过程:

2.1 数据库表结构的设计

为便于理解,现将我们设计的数据库表结构列于表1。

表1 书目数据表(T-Book)

字段名称	数据类型	长度	说明
B-Name	String	50	图书名
Author	String	50	图书作者
ISBN	String	50	国际图书标准书号

2.2 设计客户端HTML页面

为馆员输入书目数据以及为读者查询书目信息分别设计客户端HTML页面,为所有馆员和读者提供一个统一的系统访问接口。

2.3 设计和实现中间层LibraryServlet

Servlet中间层在浏览器和数据库服务器之间起着“联系人”的作用,它根

据馆员或读者提交的信息构造合适的SQL语句,并将SQL语句提交给后台数据库服务器,然后根据数据库返回的结果产生HTML页面再返回给浏览器。

下面以LibraryServlet为例具体说明如何实现Servlet中间层的。

2.3.1 构造LibraryServlet

下面这段代码主要完成Servlet与后台数据库的连接的建立。

```
public class LibraryServlet extends HttpServlet {
    protected String b-Name, b-Author, b-ISBN;
    protected Connection conn;
    protected PreparedStatement readSql, writeSql;
    //Library是系统DSN名
    protected String dbUrl = "jdbc:odbc:Library";
    public void init(ServletConfig config) throws ServletException { try {
        //加载JDBC-ODBC驱动程序
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        //建立连接
        conn = DriverManager.getConnection(dbUrl, "", "");
    } catch (Exception e) {
        System.out.println("Can not Connect database");
    }
}
```

当一个服务器加载Servlet时,它调用Servlet的init()方法。该方法不能被反复调用,直到服务器调用Servlet的destroy()方法卸载Servlet后才能再调用,一旦调用就等于再一次加载Servlet。在LibraryServlet的init()方法中我们利用JDBC的几个数据库连接类建起了与书目数据库的连接。

2.3.2 对用户请求提供服务

在服务器装载初始化Servlet后,Servlet就可以开始处理客户端的请求。用service方法可以做到这一点。每个客户端有它自己的service()方法:这些方法接收客户端的请求,并且作出相应的响应。在service()中我们必须实现中间层的所有功能。部分代码如下:

```
public void service(HttpServletRequest req, HttpServletResponse res) throws ServletException, IOException {
    b-Name = req.getParameter("B-Name");
    b-Author = req.getParameter("B-Author");
    b-ISBN = req.getParameter("B-ISBN");
    if (b-Name == null && B-Author == null && b-ISBN == null)
        doQuery(req, res);
    else
        doInsert(req, res);
}
```

```
public void doQuery (HttpServletRequest req, HttpServletResponse res) {
    try {
        readSql = conn.prepareStatement("select from T-Book");
        String PageHead = "<html> <head> <title> 查询返回结果 </title> </head> \n";
        String PageBody = "<body> " + ;
        ResultSet rs = readSql.executeQuery();
        while(rs.next()) {
            String Name = rs.getString("B-Name");
            String Author = rs.getString("B-Author");
            String ISBN = rs.getString("B-ISBN");
            PageBody += Name + " " + Author + ISBN + "\n";
        }
        PageBody += "</body> " + </html> "
        PrintWriter out = new Print Writer
    }
}
```

HttpServletRequest接口可以获得到这样一些信息,如由客户端传送的参数名称,客户端正在使用的协议,产生请求并且接收请求的服务器远端主机名。我们通过它来取得馆员或读者的输入数据,然后根据他们的需要进行书目库查询或更新操作。

2.3.3 查询书目数据

查询书目库数据首先要构建一个查询SQL语句,然后通过executeQuery()方法向后台数据库服务器提交查询请求。取得执行结果后,Servlet通过HttpServletResponse生成结果页面返回给读者。

```
public void doQuery (HttpServletRequest req, HttpServletResponse res) {
    try {
        readSql = conn.prepareStatement("select from T-Book");
        String PageHead = "<html> <head> <title> 查询返回结果 </title> </head> \n";
        String PageBody = "<body> " + ;
        ResultSet rs = readSql.executeQuery();
        while(rs.next()) {
            String Name = rs.getString("B-Name");
            String Author = rs.getString("B-Author");
            String ISBN = rs.getString("B-ISBN");
            PageBody += Name + " " + Author + ISBN + "\n";
        }
        PageBody += "</body> " + </html> "
        PrintWriter out = new Print Writer
    }
}
```

(下转第60页)

1
快进
科研
Inter
教育
开始
通了
础上
些小
角洲
长送
具备
来看

化。我国图书馆历来“重藏轻用”，不注重社会效益，缺乏主动服务、不断创新的积极精神；另外，我国长期以来奉行闭关自守、独善其身的传统文化，全国图书馆实行条块分割、纵向管理的体制，各系统之间各自为政，互不往来，各方面的重复建设都很严重，这对作为发展中国家的我国来说，是极大的浪费。

当前我国正处在大力发展信息服务行业的关键时期，图书馆必须抓住这一大好时机，做好以下几项工作：

——加强图书馆自动化、网络化建设。由于大多数图书馆设备陈旧、落后，缺乏必要的办公自动化条件，因此无法建立相应的网络，这种现状严重地制约了图书馆的发展。未来的图书馆信息服务必须以先进的网络系统作为支撑，因此，图书馆应多方争取资金，自行开发或引进先进技术，建立图书馆的局域网、广域网，最终加入 Internet，通过网络提供全面便捷的服务。

——成立全国性图书情报管理机构。在全国网络建设、馆藏信息资源建设、引进、数据库开发乃至一些技术研发方面，要彻底打破现有的封闭格局，实行统筹规划。各系统可以本系统最大或最权威的图书馆为主形成一个系统中心，该中心馆应在系统内部的各项建设和合作服务方面发挥牵头作用。在此基础上，由各中心馆本着平等自愿的方针，联合组建一个全国性图书情报管理机构，负责制定全国网络和服务发展目标、方案、制度，统一标准，并协调各方面的关系，对内促进各单位的共建共享，对外争取政策与经费支持。

——培养高素质的人员队伍。信息社会中的图书馆信息服务水平归根结底取决于人的素质，开展多样化的现代信

[作者简介] 易飞，女，1972年生，硕士研究生，发表论文1篇。

息服务需求大量懂得现代通讯网络技术、图书情报专业、掌握一门以上外语，了解市场、公关学等知识，并具有一定创新精神的人才。现在，许多图书馆存在着人员素质低、队伍老化、知识陈旧的问题，因此应当对现有人员开展技术培训和在职教育，同时大力引进各方面的人才，使人员知识结构向多层次、综合化方向发展。

参考文献：

- 1 张树华.中国图书馆读者服务工作百年回眸.中国图书馆学报,1999(6):70~75
- 2 顾霖.美国图书馆事业概况.北京图书馆馆刊,1999(3):125~130
- 3 李嘉琳.中美图书馆自动化发展比较研究.图书情报工作,1999(1):32~36
- 4 董健民.国外图书馆网络化发展与信息服务.四川图书馆学报,1997(1):21~26
- 5 刘省泉,曹臻.网络环境下我国图书馆文献信息资源建设与共享.世纪之交图书馆事业回顾与展望,北京:北京图书馆出版社,1999:53~57
- 6 胡秋民.浅论国家图书馆的文献提供服务.国家图书馆学刊,2000(2):2~5
- 7 黄凯卿.美国俄亥俄州高校网的信息服及其对我国的启示,1997(4):45~47
- 8 谢新洲,一凡.欧美数据库产业的发展现状.情报学报,1997(6):434~442
- 9 K. Wayne Smith, ed. OCLC 1967-1997: Thirty Years of Furthering Access to the World's Information. New York: The Haworth Press, Inc, 1998

(上接第56页)

```
(res.getOutputStream());  
res.setContentType("text/html");  
out.println(PageHead + PageBody + .....);  
} catch(Exception e) { ..... }  
}
```

2.3.4 提交书目数据到书目数据库 馆员提交书目数据到后台数据库与读者查询操作相似，只是构造的 SQL 语句为：

```
String writeSql = "insert into T-Book (B-Name, Author, ISBN)  
Values( + b-Name + ", + b-Author + ", + b-ISBN + ")";
```

3 结束语

通过上文的图书管理系统的设计和实现，可以看到采用 Servlet 技术实现中间层是十分简单的。利用 Servlet 可以快

[作者简介] 易俊刚，男，1975年生，硕士研究生。

速、方便地在现有的数据库上创建 Web 接口。而且 Servlet 是在一个 Web 服务器的进程中共享和分离线程，不像 CGI 那样在每个用户提出请求时要创建一个新的进程。同时，由于 Servlet 是用 Java 语言进行编写的，其可移植性、扩展性、安全性也超过 CGI。因此，在未来的 Web 系统开发中，Java Servlet 将会有更加广阔的应用前景。

参考文献：

- 1 齐永强,熊家军等.JDBC连接数据库.重庆邮电学院学报,1999(1):63~66
- 2 赵文静.Java语言中的数据库访问技术.微机发展,1999(5):53~55
- 3 <http://www.javart.com>.
- 4 SUN公司.Java Servlet Development Kit 2.0 联机文档.1999