

浅析实现电子商务的基于 J2EE 的框架

曾陈萍¹, 李怀杰²

(1. 西昌学院, 四川 西昌 615013 2. 西昌市开发办, 四川 西昌 615000)

【摘要】本文从 MVC 的软件设计模式出发, 具体分析了 J2EE 在企业级软件开发中的应用, 提出了一个基于 J2EE 的实现电子商务的框架。

【关键词】电子商务; J2EE; XML; Servlet; MVC

【中图分类号】TP311.52 **【文献标识码】**A **【文章编号】**1673-1891(2006)03-0078-05

引言

我们正处于一个崭新的计算时代, 一个互联网时代。随着计算机及网络技术的飞速发展, 基于 Web 的应用服务方式已经成为互联网服务的主流, 使用电子技术进行商务活动的成本不断下降, 新一代电子商务正在改变企业经营的面貌, 正逐步渗透进我们的生活。

Sun 公司推出的 J2EE 架构, 为 Web 应用找到了一种统一的具有体系结构的计算模式, 成为开发商创建电子商务应用的一种事实上的标准。本文提出的基于 J2EE 实现的电子商务框架采用了 J2EE 多层体系结构, 尽量使用现在 J2EE 系统中的许多功能, 可扩充性好, 具有良好的移植性和可维护性。

1 开发平台的选择

优秀的平台是开发高需求电子商务系统的基础, 选择平台必须分析平台的性能与发展前景, 以及考虑公司的遗留系统和规划。

J2EE 和 .NET 都是目前比较优秀的开发平台。但 .NET 在可预见的将来, 仅运行在 Windows 操作系统内。尽管微软经常宣称要再开发 Linux 上的 .NET, 但这只是一个很遥远的未知数。而 J2EE 只要遵循 Java VM(规则) 和一组平台需要的服务就可以在任何平台上工作。所有这些定义了 J2EE 平台的规程, 都已经公开发表, 并提供公众阅读。许多供应商也提供兼容产品和开发环境。因此, 在目前情况

下, 选择 J2EE 是一个较为稳妥的方案, 因为 J2EE 可以支持较多的平台, J2EE 得到较多厂商的支持并且经过实践检验已经比较成熟, 而 .NET 仅有微软支持。

2 MVC 模式

J2EE 多层体系结构的灵感来自模型/视图/控制器(Model - View - Controller, MVC)架构。MVC 模式是软件设计的典型结构, 在这种设计结构下, 一个复杂应用被分解为模型、视图和控制器三部分, 分别对应于业务逻辑和数据、用户界面、用户请求处理和数据同步。MVC 设计模式如图 1 所示:

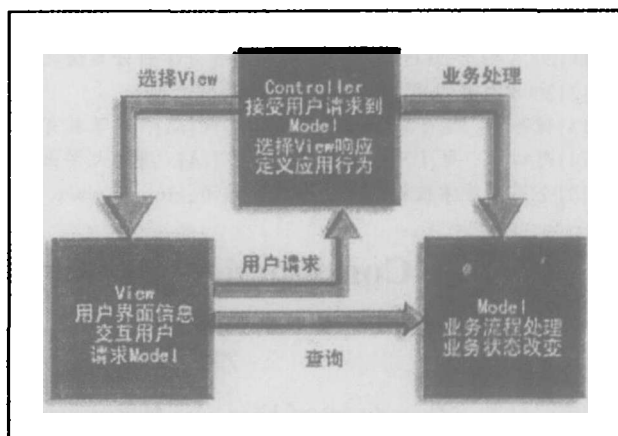


图 1 MVC 设计模型

视图 (View) 代表用户交互界面, 对于 Web 应用来说, 可以概括为 HTML 界面, 但有可能为 XHTML、XML 和 Applet。随着应用的复杂性和规模性, 界面

收稿日期: 2006-09-11

作者简介: 曾陈萍 (1968 -), 女, 计算机副教授, 四川大学软件工程硕士

的处理也变得具有挑战性。一个应用可能有很多不同的视图,MVC设计模式对于视图的处理仅限于视图上数据的采集和处理,以及用户的请求,而不包括在视图上的业务流程的处理。业务流程的处理交予模型(Model)处理。比如一个订单的视图只接受来自模型的数据并显示给用户,以及将用户界面的输入数据和请求传递给控制和模型。

模型(Model):就是业务流程/状态的处理以及业务规则的制定。业务流程的处理过程对其它层来说是黑箱操作,模型接受视图请求的数据,并返回最终的处理结果。业务模型的设计可以说是MVC最主要的核心。目前流行的EJB模型就是一个典型的应用例子,它从应用技术实现的角度对模型做了进一步的划分,以便充分利用现有的组件,但它不能作为应用设计模型的框架。它仅仅告诉你按这种模型设计就可以利用某些技术组件,从而减少了技术上的困难。对一个开发者来说,就可以专注于业务模型的设计。MVC设计模式告诉我们,把应用的模型按一定的规则抽取出来,抽取的层次很重要,这也是判断开发人员是否优秀的设计依据。抽象与具体不能隔得太远,也不能太近。MVC并没有提供模型的设计方法,而只告诉你应该组织管理这些模型,以便于模型的重构和提高重用性。业务模型还有一个很重要的模型那就是数据模型。数据模型主要指实体对象的数据保存(持续化)。比如将一张订单保存到数据库,从数据库获取订单。我们可以将这个模型单独列出,所有有关数据库的操作只限制在该模型中。

控制(Controller)可以理解为从用户接收请求,将模型与视图匹配在一起,共同完成用户的请求。划分控制层的作用也很明显,它清楚地告诉你,它就是一个分发器,选择什么样的模型,选择什么样的视图,可以完成什么样的用户请求。控制层并不做任何的数据处理。例如,用户点击一个链接,控制层接受请求后,并不处理业务信息,它只把用户的信息传递给模型,告诉模型做什么,选择符合要求的视图返回给用户。因此,一个模型可能对应多个视图,一个视图可能对应多个模型。

3 视图表示

3.1 页面显示技术的选择

在J2EE中,JSP和Servlet都是用于实现系统的

表示逻辑的,它们就像一枚硬币的两面,各有特长,并且其特长的形成都是以其它功能的削弱为代价的。

具体的说来,JSP可以看作是嵌入了Java程序的HTML文件;而Servlet则相反,它完全用Java编写而成,并且能够产生HTML代码。一个设计良好的JSP页面甚至不用精通Java的专业人员就可以维护。Servlet则擅长于处理用户的请求,并且将这些请求转发给相应的组件,可以说是一个转发中心。同时,由于Servlet由纯Java写成,所以它能够充分Java语言的种种优势,在安全性的API扩展方面特长突出。

JSP和Servlet可以完全代替对方的工作。但是,用JSP完成复杂的请求处理和转发工作加大了JSP设计人员的工作量和工作难度,提高了对页面维护人员的要求,也不符合设计JSP的初衷。同样,用Servlet完成页面显示固然是可行的,但是工作繁琐,让程序变得复杂,不易维护。

因此我们认为很难在JSP和Servlet之间划一条严格的界限,单独使用JSP或者直接从Servlet中产生HTML代码都不是一种很理想的显示页面的方法。所以我们可以使用J2EE中的XML技术,由Servlet产生代表XML树型数据结构的DOM树,并把它转换成HTML,最终把数据送到浏览器中。

下面,将页面显示所使用的XML、XSL和DOM技术作一个简单的介绍。

3.2 XML简述

XML(Extensible Markup Language)是一种可扩展的元置标语言,它的主要设计动机是要克服HTML的种种不足,将网络上传输的文档规范化,并赋予标记一定的含义,与此同时,还要保留HTML所具有的简捷、适于网上传输和浏览的优点。因此,XML最终以SGML子集的形式出现,它集SGML和HTML的优势于一身,具有易于编辑、便于管理、适于存档、容易查询等诸多优势,成为网络发展的又一个亮点。

与HTML不同,XML的设计目的不是用于在浏览器中显示数据,而主要是描述数据。XML的目的是供程序处理数据用,便于企业应用程序相互之间传递结构数据。最常见的做法是把数据作为记录保存到数据库中。当需要时,抽取出来并把数据转换成XML进行传递。然后由Servlet或JSP读取XML并把它转换成HTML,最终把数据送到浏览器中。

Java XSLT 库恰好具备这样的功能。

3.3 XSL 概述

XSL(Extensible Stylesheet Language)是描述 XML 文档样式信息的一种语言,是由 W3C 制定的。XML 的一个优点是形式与内容相分离,这使得 XML 文档具有简洁、易读的特点。而它的样式信息都包含在称为样式单的样式文件中, XSL 就是这种样式单之一。

XSL 是通过 XML 进行定义的,遵从 XML 的语法规则,是 XML 的一种具体应用。它两大部分组成:第一部分描述了如何将一个 XML 文档进行转换,转换为可浏览或可输出的格式;第二部分则定义了格式对象 FO(formatted object)。在输出时,首先根据 XML 文档构造源树,然后根据给定的 XSL 将这个源树转换为可以显示的结果树,这个过程称作树转换,最后再按照 FO 解释结果树,产生一个可以在屏幕上、纸上、语音设备或其它媒体中输出的结果,这个过程称为格式化。

到目前为止, W3C 还未能出台一个得到多方认可的 FO,但是描述树转换的这一部分协议却日趋成熟,已从 XSL 中分离出来,另取名为 XSLT(XSL Transformations),现在一般所说的 XSL 大都指的是 XSLT。

使用 XSL 显示 XML 的基本思想是通过定义模板将 XML 源文档转换为带样式信息的可浏览文档。XSLT 将一个没有形式表现的 XML 内容文档作为一个源树,将其转换为一个有样式信息的结果树。在 XSLT 文档中定义了与 XML 文档中各个逻辑成分相匹配的模板,以及模板匹配和转换方式。值得一提的是,尽管制定 XSLT 规范的初衷只是利用它来进行 XML 文档与可格式化对象之间的转换,但它的巨大潜力却表现在它可以很好地描述从 XML 文档向任何一个其他格式的文档作转换的方法,例如转换为另一个逻辑结构的 XML 文档、HTML 文档、XHTML 文档、VRML 文档甚至 SVG 文档。

3.4 DOM

DOM 的全称是 Document Object Model,也即文档对象模型。在应用程序中,基于 DOM 的 XML 分析器将一个 XML 文档转换成了一个对象模型的集合(这个集合通常被称为 DOM 树),应用程序可以通过对该对象模型的操作,实现对 XML 文档中数据的操作。通过 DOM 接口,应用程序可以在任何时候访问 XML 文档中的任何一部分数据,因此,这种利用

DOM 接口的机制也称做随机访问机制。而且通过 DOM 接口,应用程序不仅可以对 XML 文档中的数据进行访问,还可以对 XML 文档中的数据进行修改、移动、删除和插入等操作。

DOM 接口提供了一种通过分层对象模型来访问 XML 文档中信息的方式,这些分层对象模型依据 XML 文档的结构形成了一棵节点树。应用程序正是通过与该节点树的交互,来访问 XML 文档中的信息。

作为 W3C 的接口标准,DOM 实际上就是 HTML 文档和 XML 文档的应用程序接口(API)。它定义了 HTML 文档和 XML 文档的逻辑,给出了一种访问和处理 HTML 文档和 XML 文档的方法。利用 DOM,程序开发人员可以动态地创建文档,遍历文档结构,添加、修改、删除文档内容,改变文档的显示方式等等。可以这样说,文档代表的是数据,而 DOM 则代表了如何去处理这些数据。

3.5 页面显示类的设计框架

以设计实现页面显示的一个 Java 类 ShowPage 为例。

代码的第一部分是 import 语句,引入 XML 包、I/O 包和 W3C 所制定的 DOM 接口。

```
import javax.xml.parsers. DocumentBuilder;
import javax.xml.parsers. DocumentBuilderFactory;
import javax.xml.transform. *;
import javax.xml.transform. stream. *;
import javax.xml.parsers. *;
import javax.xml.transform. dom. *;
import org. w3c. dom. *;
import java. io. *;
```

类 ShowPage 包括两个静态的方法,其中第一个方法是获取初始文档对象。文档对象是一种内存中生成的树型数据结构, DOM API 提供了某些接口,允许把任何数据存储到这种树型数据结构中。因此我们可以从数据库中查询数据,并可以在进行逻辑处理后存储到文档对象中。然后我们就可以采用第二个方法将处理后的文档对象结合 XSL 文件转换为 HTML 文件格式显示输出。

```
public class ShowPage
{
    //获取初始文档对象
    public static Document getDocument()
    {
        . . . . .
    }
}
```

//将 XSL 文件转换为 HTML 文件格式显示输出的方法

```
public static void viewPage
(HttpServletRequest request,
HttpServletResponse response,
Document doc,String filePath)
{.....}
}
```

4 数据库连接池

数据库处理往往是应用程序的整个业务处理中最耗时的步骤,其中数据库的连接和释放往往又是特别耗时的。在应用服务器系统中,一般都采用数据库连接池(Connection Pool)技术,即在系统初次启动或者初次使用时,进行数据库的连接,而后并不释放这些连接。当应用程序需要连接数据库时,就从数据库连接池中取出一个连接;当不再需要数据库连接时,就把这个连接归还连接池。这样,就可以反复使用这些已经建立的连接。这种方式可以大大减少数据库的处理时间,有利于提高系统的整体性能,因此被广泛地应用在各种应用服务器产品中。

下面,我们以一个 Java 类 ConnectionPoolManager 为例来了解如何实现数据库连接池的功能。代码如下:

```
import javax.naming.*;
import java.sql.*;
import javax.sql.*;
public class ConnectionPoolManager
{ //获取数据库连接
    public Connection getConnection(String jndi)
        throws Exception
    { Connection cnn = null;
        try
        {
            Context ctx = new InitialContext();
            if(ctx == null)
                throw new Exception("没有匹配的环境");
            //获取连接池对象
            DataSource ds =
```

```
(DataSource) ctx.lookup("java: comp/
env/jdbc/" + jndi);
            if(ds == null)
                throw new Exception("没有匹配数据库");
            //获取一个连接
            cnn = ds.getConnection();
            if (cnn == null)
                throw new Exception();
            return cnn;
        }
        catch (Exception e)
        {
            e.printStackTrace();
            throw new Exception("连接池连接获取异常");
        }
    }
    //释放数据库连接
    public void freeConnection(Connection cnn)
    { try
        { if (cnn != null)
            { //将连接放回连接池
                cnn.close();
            }
        }
        catch (Exception e)
        { e.printStackTrace();
        }
    }
}
```

类 ConnectionPoolManager 中实现了两个方法。第一个方法是获取一个数据库连接,参数 java 是命名与目录接口。第二个方法是将一个不再使用的连接归还到数据库连接池中。当然,并不是只要我们在以后应用程序中使用这个类就能获得数据库连接,我们还需要对服务器进行连接池设置。

5 结束语

J2EE 是目前进行电子商务开发的一个重要平台。利用这种基于 J2EE 的电子商务框架,可以快速、迅捷地实现电子商务,改善系统的可重用性和可维护性,还能将来系统的扩展打好良好的基础。J2EE 具有广阔的发展前景。

参考文献:

- [1] Bruce Eckel[美]著,侯捷译. Java 编程思想[M]. 机械工业出版社,2002.

- [2] Peter van der Linden[美]著,邢国庆等译. Java2 教程[M]. 电子工业出版社,2003.
- [3] Johon Bell[美]等著,马树奇等译. Java Servlet 2.3 编程指南[M]. 电子工业出版社,2002.
- [4] XML 中国论坛,XML 实用进阶教程[M],清华大学出版社,2001.
- [5] 安洪余. 基于 J2EE 技术搭建电子商务平台[J],计算机应用研究,2003, (11).

Analysis of Framework Based on J2EE of Realizing E – Commerce

ZENG Chen – ping¹, LI Huai – jie²

(1. College Xichang, Xichang Sichuan 615013. 2. The development office of Xichang, Xichang Sichuan 615000)

Abstract: This paper discusses software design mode of MVC, makes a concrete analysis of the application in the software development of the enterprise layer of J2EE, puts forward a frame based on e – commerce of realization of J2EE.

Key words: E – Commerce; J2EE; XML; Servlet; MVC

(责任编辑:张荣萍)

(上接 67 页)

To Perform Drawing Sheet Management through Integrating AutoCAD and Database

WU Di

(Xichang College, Xichang Sichuan 615013)

Abstract: We can transform attribute information among database software and Excel based on mechanical assembly simulation with AutoCAD. So we can add language explain for assembly simulating system easily, which in turn helps us perform drawing sheet management.

Key words: AutoCAD; Attribute; Database; Drawing sheet management

(责任编辑:张荣萍)

(上接 70 页)

which now exist in training the qualified personnel of software of colleges and universities. It also gives some tactics and makes some suggestions on how to train the qualified personnel of software and how to improve the quality of the training of the qualified personnel of software.

Key words: The qualified personnel of software; Train

(责任编辑:张荣萍)