

文章编号:1005-0523(2006)01-0068-04

# 网上书城系统的研究与设计

刘立月, 谢昕, 舒俭

(华东交通大学 信息工程学院, 江西 南昌 330013)

**摘要:**介绍了网上书城系统体系结构与工作原理,对系统功能模块与主要页面设计进行了阐述与分析,尤其是对系统的安全性、可靠性、方便性及数据挖掘等方面进行了深入探讨与研究。

**关键词:**网上书城系统;电子商务;安全性;数据挖掘

**中图分类号:**TP391

**文献标识码:**A

## 1 引言

因特网的迅猛发展正以前所未有的深度和广度影响和改变着人类生活的各个方面,越来越多的人开始意识到因特网所蕴含的巨大经济价值和无穷商机,并积极投身于电子商务活动。随着因特网的迅猛发展而日益流行,电子商务的应用越来越广泛,如网上书城、网上商城等。网上书城的崛起对传统的图书流通体系产生了强烈的冲击,有效地缩短了图书流通发行环节,将广大读者、图书、出版者、发行者紧密地结合在一起,大大提高了图书流通率。

以网上书城为例,首先介绍了其开发环境,体系结构及工作原理,然后对系统功能模块与主要页面设计进行了详细的阐述与分析,最后对系统的关键技术进行了深入的探讨与研究。

## 2 系统设计方案

### 2.1 开发环境

考虑到 ASP (Active Server Pages) 编程和维护的方便性,系统 Web 服务器采用 IIS 5.0 (Internet Information Server 5.0), 数据库采用 MS SQL Server 2000。

### 2.2 体系结构

系统体系结构大致可分为三层逻辑结构,即浏览器一

Web 服务器一数据库服务器,如图 1 所示,其中 Web 服务器又被细分为前端和后端两部分。三层结构将用户界面、应用逻辑和数据管理之间彻底分离,彼此独立,各司其职。用户端通过 HTTP 和 Web 服务器建立网络连接,获取 Web 服务器上的图书及相关信息;应用逻辑层则通过 ADO 和数据库服务器进行数据交换。同时为保证系统的安全性和网络的访问性能,将应用逻辑细分:普通的数据流只能到达前端 Web 服务器,网上交易的私人数据和后台管理数据则由后端 Web 服务器专门处理。后台数据库服务器采用双备份,主数据库与从数据库之间定期热备份,以确保系统重要数据安全性。

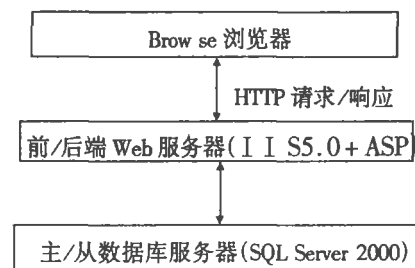


图 1 系统体系结构

### 2.3 工作原理

顾客实现一次图书购买的过程是:顾客通过分类浏览或关键字检索选择需要的图书,如果要正式购买则通过用户注册,递交图书订单,并网上支付或邮寄货款,系统管理员收到银行的划款或邮款后,通知图书配送中心,配送中心可根据具体情况选择直接邮寄或通知图书供货商邮寄给客户。系统

收稿日期:2005-08-20

作者简介:刘立月(1970-),男,讲师,硕士,研究方向:测控,网络及数据库。

工作模式如图2所示.

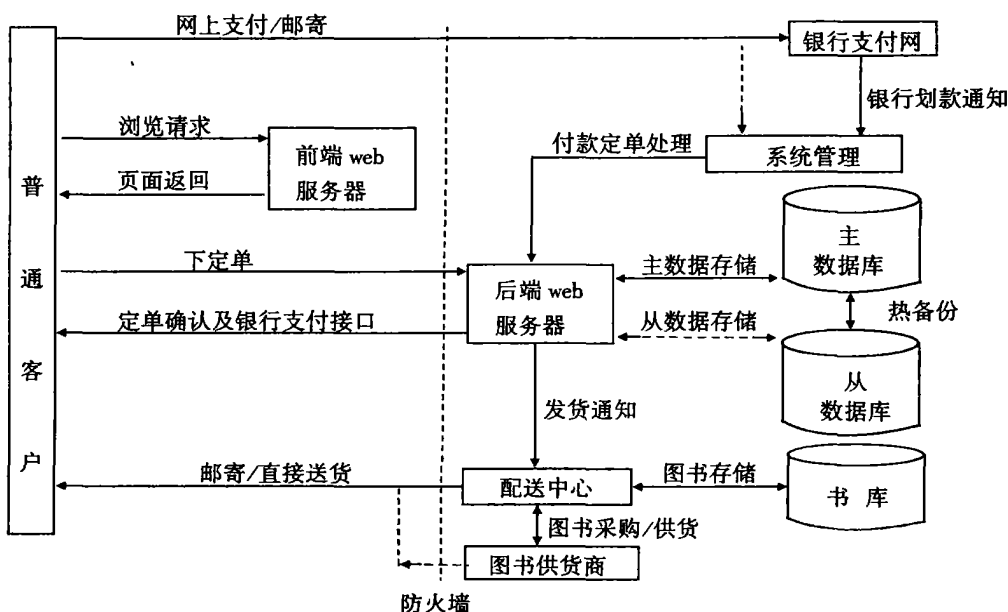


图2 系统工作模式

### 3 系统功能模块与页面设计分析

#### 3.1 功能模块

网上书城系统按功能分为两部分:从客户角度,需要实现前台网上订购功能;从商家角度,需要实现后台系统管理功能.其功能结构如图3所示.

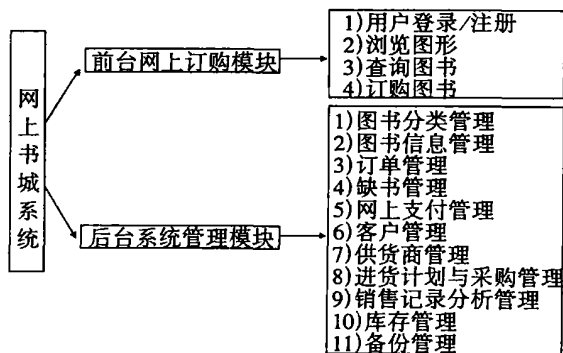


图3 系统结构功能

##### 3.1.1 前台网上订购模块

- 1) 用户登录/注册:客户可以随时进行用户登录/注册.
- 2) 浏览图书:客户通过浏览图书界面,可以浏览图书的种类目录及某一目录下的所有图书,点击某一商品后,显示该图书的详细信息,如显示图书的详细介绍、价格等.
- 3) 查询图书:可输入查询条件找出某些符合条件的图书.
- 4) 订购图书:客户选中图书后,递交购书定单及定单回

复确认,实现网上订购图书交易.

##### 3.1.2 后台系统管理模块

- 1) 图书分类管理:对图书分类进行管理,包括新增、删除、修改、查询图书等.
- 2) 图书信息管理:包括图书目录结构管理,图书信息维护、共享、展示以及、查询等.
- 3) 订单管理:订单处理情况包括标识管理、订单查询、订单统计等.
- 4) 缺货管理:对缺货登记,为制定采编计划提供依据.
- 5) 网上支付管理:以银行为中介,支付信用卡,实现现金电子化交易.
- 6) 客户管理:客户管理角色包括系统管理员,员工,会员及一般顾客.
- 7) 供货商管理:包括添加、更改、查询、删除供货商资料等.
- 8) 进货计划与采购管理:定期根据缺货登记表和书本的出售情况,供货商提供的新书目录制定进货计划;对采购情况进行记录.

9) 销售记录分析管理:对每本书籍的出售情况作详细记录.定期对这些记录进行不同角度的分析,如书本出售即时走向图,季度销售表、消费群体分析等,提供给决策部门.

10) 库存管理:根据每次进货,出货的情况,自动更新数据库中库存的信息.

11) 备份管理:图书系统资料备份.

##### 3.2 主要页面设计分析

网上书城虽然是一个虚拟的网上购物系统,但它的购物原理与现实生活的购物基本类似.在具体的网站设计中,将

购物过程转换为页面,利用页面实现购物方式.

### 3.2.1 网上书城系统主页面

首页作为网站的门面,它的设计要考虑很多方面的因素,如说功能、美观、内容等.网上书城的首页无论在功能还是界面美观都要体现对客户的友好性,即最大程度地满足客户、方便客户是要考虑的首先因素.

### 3.2.2 图书信息页面

用来显示每本书的详细信息,具体包括作者、出版社、出版日期,适合对象,内容简介,目录及相关读者或专家评论等,帮助顾客确定是否购买.

### 3.2.3 购物车页面

用来存放顾客选购的各种图书,它简要列出所购图书的名称、数量、定价等信息,用户可以随时查看和修改购物车中的图书.

### 3.2.4 购书清单页面

每次购买完后,都以定单回复确认方式向顾客显示其购买清单,以便于顾客对图书种类、数量及金额的查看与核实.

### 3.2.5 顾客信息页面

它用来让用户输入自己的姓名、地址、电话、Email、送货日期,选择自己的付款方式等必要信息,以便顾客与商家(书城)联系,如发货等.

### 3.2.6 客户数据库设计

客户数据库中至少需建立三个表:客户购书信息表、客户信息表及订单信息表.

## 4 系统设计中关键技术探讨

### 4.1 系统安全性方面

安全性问题在网上书城系统中的重要性是不言而喻的,轻者网站停止运行,重者网站瘫痪,甚至重要数据丢失,造成严重的经济损失.

考虑到 Web 服务器经常要与后台数据库进行数据交换,容易受到黑客攻击,可将其放置在防火墙内.但防火墙的安全检查又会造成数据访问流量的瓶颈,因此可以对系统的体系结构作适当的改进,即将 Web 服务器分成前端和后端服务器两部分.前端服务器置于防火墙外,负责处理普通的浏览业务;后端服务器置于防火墙内,仅当启动一个网上交易时才访问它,负责存取数据库数据,处理定单,验证身份,发送图书通知等后台管理工作.这样就既保护了数据的安全性,网络的整体性能也不会有很大损失.

后台数据库服务器存放重要的数据信息,为确保数据安全,采用双备份,主数据库与从数据库之间定期热备份.当主数据库出现故障时,从数据库立即成为主数据库,确保系统的正常运行,同时也保证了数据的安全性.

### 4.2 支付可靠性方面

网上书城系统中电子支付是网上购书与网上支付的重要保证,网上支付必须得到高度电子化银行等金融机构的

支持.网上书城系统中的顾客持有信用卡,在发卡行设信用卡帐号,以便接收顾客的网上支付.在利用信用卡或支票交易中,敏感信息被交换,在信息网上传送中,必须经过加密及数字签名处理,以确保网上支付交易的可靠性.

著名的 CyberCash 和 VISA/Mastercard 的 SET 就是基于数字信用卡的典型支付系统.SET 主要是为了解决用户、商家和银行之间通过信用卡支付的交易而设计的,以保证支付信息的机密、支付过程的完整、商户及持卡人的合法身份,以及可操作性.

在 SET 协议中,加密技术构成了 SET 网上安全运行的基础,主要加密技术有:对称密钥加密、公钥加密、哈希算法、数字签名、数字信封和数字证书.从用途来看,对称密钥加密、数字信封用于信息的保密,哈希算法用于达到交易的不可抵赖的目的;数字证书、公钥加密确保交易方的身份认定,而数字签名用于确认发送事实和发送者的身份,从而保证传递信息的完整性.进行数字签名首先需要将发送信息利用哈希算法生成信息摘要,对于同一信息使用哈希运算得到的摘要是唯一的.然后再对它用私人密钥加密形成摘要密文,信息接受者在收到加密信息和摘要密文后,首先用发送者的公开密钥解密信息摘要,与被解密后的信息原文使用哈希算法产生的信息摘要进行比较,确认真伪.因为只有私人密钥的拥有者才能对原文用私钥加密,所以数字签名可以证实信息是否来源于发送者,从而达到信息的不可抵赖性的要求.

系统可采用基于 SET 协议的网上交易,其支付流程如图 4 所示.

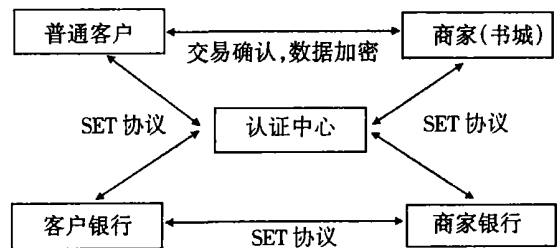


图 4 基于 SET 协议的网上支付流程

### 4.3 方便客户购书方面

一个成功的网上书城系统不仅要便于商家(书城)管理,更重要的是方便客户购书.

如何最大程度地满足客户、方便客户是设计一个网上书城系统要考虑的重要因素.可借鉴超市的成功经验,系统为每个客户分配一个购物手推车,客户可一边浏览图书,一边选购.做法是,当客户发现有自己中意的图书时,只需输入购买的数量,然后按加入购物车图标或按钮,相应数量的图书便自动加入购物车.提交定单前,用户还可对购物车中的图书进行修改,增加或减少某些图书的选购数量,以及去除某些图书等.

另一个重要方面是如何方便顾客浏览商品.由于一个网上书城中的图书种类可能非常繁多,不可能要求用户一个一

个地浏览.根据图书分类特点安排布局,照顾顾客浏览习惯,简化购书流程,可将目录与图书分门别类有层次地列在不同的页面中,用户可通过层层访问找寻自己需要的图书.同时为了能够使客户迅速找到所需图书,设立“快速查询”、“高级查询”等方式,真正做到提供“简洁,高效,流畅”的购书环境.

#### 4.4 数据挖掘方面

对于任何基于数据库的电子商务系统,大量数据的产生和收集都会导致信息爆炸.虽然现在有了更强大的存储和检索系统,但是在分析和使用所拥有的信息方面还是变得越来越困难.数据挖掘技术应用,使得能够从大量繁杂的数据中找出真正有价值的信息和知识.

R. Agrawal 的关联规则发现算法——Apriori 算法是数据挖掘技术的典型应用之一.该算法首先对数据库进行搜索,找出所有的频繁项目,重复查找,直到找出所有的大项集.其中,挖掘产生布尔关联规则所需频繁项集是 Apriori 算法最关键的一步,为提高频繁项集逐层产生的效率,用 Apriori 性质(即频繁项集的所有非空子集都必须也是频繁的)来压缩搜索空间.目前该算法已经被证明在大型数据库中挖掘关联规则的有效工具,但由于该算法只用支持度和置信度这两个标准来衡量关联规则,在实际应用中往往会生成大量冗余的、虚假的和用户不感兴趣的关联规则,其结果可能产生大量的候选集,以及可能需要重复扫描数据库,针对其不足,已有相关在 Apriori 基础上的改进算法.

在该系统中,数据挖掘技术主要应用在对书本出售记录的分析上,系统可采用 Apriori 算法,以便从交易项目的内容中发现有价值的信息或知识,如:影响 11 月份销售情况的因素是什么?购买计算机类书籍的人同时又购买外语类书

籍的比例有多大?他们一般还会购买哪类书籍?等等.这些信息可以用作页面布置、销售目录设计、采购安排、针对性的促销活动等商业策略制定的参考,提供给书城上层决策部门使其能以合理的代价取得有效的决策支持.

## 5 结束语

网上购物系统虽然是以网上书城模型而设计,但它的设计思想和主要模块仍然适用于其它的以交易为中心的电子商务系统.尤其在网上市城系统的工作原理及将数据挖掘技术的 Apriori 算法应用在对图书销售分析方面的研究,对其它电子商务系统的设计具有重要的应用与参考价值.

#### 参考文献:

- [1] SET Secure Electronic Transaction Specification, Book1: Business Description. Version 1.0, 1997
- [2] R Agrawal, T Imielinski, A Swami Mining Association Rules between Sets of Items in Large Database [C]. Proceedings of the ACE SIGMOD Conference on Management of Data, 1993: 207—216.
- [3] 常晓燕,李家武.用 J2EE 实现网上购物系统[J].微机发展. 2003, 13(9): 54—56.
- [4] 陈健,朱蔚恒,印鉴.具有数据挖掘功能的网上购书系统设计与实现[J].计算机工程与应用. 2002, (10): 224—226.
- [5] 张中秋,肖冬荣,张辉,等.网上购物系统的设计与实施[J].计算机应用研究. 2002, (8): 111—113.

## Research and Design of an on-line Bookstore System

LIU Li-yue, XIE Xin, SHU Jian

(School of Information Engineering, East China Jiaotong University, Nanchang 330013 China)

**Abstract:** The paper describes system structure and work's principle of On-line Bookstore System, introduces functions of system modules and design of the main web pages in details. Especially, system security, reliability, convenience, and data mining are studied and discussed concretely.

**Key words:** on-line bookstore system; e-business; security; data Mining.