

基于中间件技术的图书管理系统的设计

姚妮, 李红婵

(郑州轻工业学院 计算机与通信工程学院, 河南 郑州 450002)

摘要: 针对目前图书管理系统维护性和延展性差的问题, 将应用服务器加入到系统的分布式结构中, 给出了应用服务器、远程数据模块和本地数据模块的设计。运行结果表明, 基于中间件技术的图书管理系统降低了系统的开发和维护成本, 提高了系统的延展性。

关键词: 图书管理系统; 中间件; 应用服务器; 分布式系统

中图分类号: TP301 **文献标志码:** A

Design of book management system based on middleware technology

YAO Ni, LI Hong-chan

(College of Comp. and Com. Eng. Zhengzhou Univ. of Light Ind. Zhengzhou 450002, China)

Abstract: Aiming at the poor maintenance and scalability of the library management system, the application server was added to the system in distributed structure. Application server, remote data module and a local data module were designed. The running results showed that library management system based on middleware technology could reduce the cost of development and maintenance and improve the system scalability.

Key words: book management system; middleware; application server; distributed system

0 引言

随着信息技术的迅猛发展, 馆藏的图书越来越多, 图书的管理工作也越来越复杂。要想使图书的管理便捷化, 使读者检索图书高效化, 就要求配备功能强大、应用方便的图书管理系统。但由于资金的投入和信息系统建设的渐进性以及产品服役的周期性, 不可避免地形成了许多厂商产品并存的局面。如何屏蔽不同厂商产品的差异, 如何减少应用软件系统集成的工作复杂度, 则成为软件工程和系统集成的重大技术问题^[1-2]。这些问题解决不好会大大降低图书管理系统的效率。

中间件是一种屏蔽异构和分布所导致集成和

共享的各种复杂技术细节而使技术问题简单化的软件工程技术和系统集成技术^[3]。它将应用软件所面临的基于平台的共性问题进行提炼和抽象, 在操作系统及数据库管理系统之上再形成一个可复用的结构或结构群, 供成千上万的不同层面、不同类型的、不同领域的应用软件复用。中间件一般分为消息中间件、交易中间件、对象中间件、应用服务器、安全中间件、应用集成服务器 6 类^[4]。

本文将采用中间件技术来解决传统图书管理系统设计方法的不足。

1 分布式结构

图书管理系统通常采用 C/S 结构。在 C/S 结构

收稿日期: 2011 - 04 - 25

基金项目: 河南省基础与前沿技术研究计划项目(102300410266); 郑州轻工业学院博士科研基金资助项目

作者简介: 姚妮(1978—), 女, 湖南省张家界市人, 郑州轻工业学院助理实验师, 主要研究方向为智能信息处理、地理信息系统。

的系统中,应用程序通常由客户端执行,并通过与后端数据库服务连接来获取系统所需要的数据。对于一般的管理信息系统(MIS),采用C/S结构十分适合,只要应用系统客户端数目不超过200个并且分布在同一个区域,那么C/S结构在执行MIS时便已足够。然而,在此时的MIS中,客户端的应用程序通常包含了大量的企业逻辑。一旦应用系统发生改变,客户端中包含的应用程序也需做相应改变,就会造成维护成本急剧升高。如果把业务逻辑封装到数据库中,由于许多应用逻辑并不适合在数据库中执行,否则数据库的执行效率就会大大降低^[5-6]。

为了解决这些问题,可以把应用程序服务器加到分布式结构中。应用程序服务器是一种应用程序,它具有众多企业逻辑。软件开发人员能够采用某种组件形态,把企业逻辑程序代码封装起来,这些的经过封装的对象称为“企业对象”。这些企业对象可以被部署到应用程序服务器上。此时系统具有众多优点,不但能够增加企业对象的可复用的程度,还能降低整个系统的开发和维护成本。又由于分布式应用系统被分成多个不同的部分并执行在不同的且互相独立的机器中,这使得整个应用系统的延展性大大提高^[7-9]。

2 系统设计

由于数据库设计以及客户端的设计不是本文重点,所以本文只给出应用服务器、远程数据模块以及本地数据模块的开发与设计。

本文将传统的图书管理系统改造为一个分布式系统。图书管理系统后台使用Server 2000数据库,中间部分采用应用服务器,前台使用Delphi 7创建客户端^[10-11],其系统结构如图1所示。

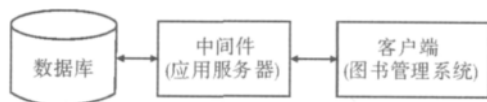


图1 系统结构图

要利用应用服务器开发多层应用系统,必须遵照以下步骤:

- 1) 开发远程服务器,负责提供数据或服务给客户端应用程序;
- 2) 开发客户端应用程序并向远程服务器提出数据或服务要求;
- 3) 连接客户端应用程序和远程服务器。

2.1 建立应用服务器

Delphi 7提供建立的远程服务器是一个EXE形态的服务器,称为MIDAS应用程序服务器。建立该服务器的步骤如下。

1) 建立应用程序。由于Delphi 7的服务端应用程序必须是一个执行在Win NT的EXE应用程序。所以在开发服务端应用程序时,必须建立一个Delphi 7应用程序。点击Delphi 7中File|New菜单下的New Application可建立Delphi 7应用程序,这时Delphi 7就会产生一个空白的窗体以及初始化的程序单元。

2) 建立远程数据模块。应用程序服务器向客户端应用程序提供存取远程数据模块的接口,这使得编写应用程序服务器成为使用Delphi 7开发多层应用系统的最重要的工作之一。应用逻辑组件可以放在该服务器上让所有的客户端应用程序调用。Delphi 7利用VCL组件建立应用程序服务器,该服务器是一个OLE Automation服务器,客户端应用程序可以利用TCP/IP,OLE Enterprise和它沟通。客户端应用程序调用服务器的IAppServer接口,以便存取远程数据模块所连接到的数据库中的数据。

要建立远程数据模块,只需在步骤1)的基础上激活New Items对话框,然后点击Multitier中的Remote Data Module。本次设计的远程数据模块的名字是ZHDTUSHUGUANLI,其他选项使用默认项即可。然后点击OK,Delphi 7就会自动建好一个空白的远程数据模块以及相应的程序单元。程序员可以在这个模块中加入任何非可视化的VCL组件(如Tdatabase,TQuery等)连接到后端的数据库。

3) 远程数据库的连接。使用数据集组件连接远程数据库并存取其中的数据,其过程如下。

首先在空白的远程数据模块中置入一个Tdatabase组件,设定其AliasName属性为图书管理系统,HandleShared属性为True,DatabaseName属性为图书管理系统,Connected属性为True,其余都为默认值。这样就可以连接到图书管理系统数据库了。

然后依次置入4个TQuery组件,设定它们的DatabaseName属性都为图书管理系统。以Query1为例继续设置以下属性:把鼠标放在Query1上点右键激活SQL Builder,选择图书信息表,这表示Query1从图书信息表中取数据;在显示的字段信息中选择所有字段后离开SQL Builder,这时它会询问是否要把自动产生的SQL叙述存储到Query1中,若回答

Yes, SQL Builder 就会把产生的 SQL 叙述存储到 Query1 的 SQL 属性之中;最后设定 Query1 的 Active 的属性为 True, 这样 Query1 就可以从数据库中取得图书信息. Query2, Query3 和 Query4 也做如上设置, 只不过它们取的数据分别为学生信息表、学生借阅记录表和借阅历史记录表中的数据.

4) 连接数据集组件. TDataSetProvider 组件提供一个接口让客户端与应用服务器沟通. 接口首先与远程数据模块中的数据相连接, 然后再让客户端连接到它上面, 这样客户端就可以存取远程数据. 本设计中依次置入 4 个 TDataSetProvider: DataSetProvider1, DataSetProvider2, DataSetProvider3, DataSetProvider4 的 Dataset 属性值分别为 Query1, Query2, Query3, Query4.

最后在应用程序服务器的主窗体内放入一个 TTable 组件, 并设定它的 Caption 属性为“ZHD-TUSHUGUANLI 应用服务器”, 以便在稍后开发客户端应用程序时把它区分出来.

5) 注册应用程序服务器. 开发完应用程序服务器之后, 必须注册这个应用程序服务器以便 COM 系统可以自动激活这个应用程序服务器. 要注册这个服务器, 只要执行一次应用程序服务器, 就会自动在系统中注册.

应用程序服务器开发完之后, 把它保存为 Project1. 接下来就可以开发客户端的应用程序, 以便客户端连接到此应用程序服务器处理远程数据.

2.2 开发客户端应用程序

多层客户端应用程序的开发与一般的两层数据库应用程序的开发区别仅在于: 使用组件 TClientDataSet 而不使用一般的数据集组件. 客户端应用程序开发步骤如下.

1) 建立一个新的 Delphi 7 的应用程序. 点击 File|New, 在对话框中选择建立 Project Group 从而建立一个新项目组. 加入上述建立的应用程序服务器即 Project1, 再于 Project Manager 窗口中点击上方的 New 按钮, 选择建立一个新的应用程序. 其中 Project2 就是一个完整的图书管理系统.

2) 建立一个新的数据模块. 分 4 步进行.

第 1 步, 点击 File|New, 在对话框中选择建立数据模块. 在刚才建立的空白数据模块中置入 TDCOMConnection 组件, 设定它的 ComputerName 属性值为 ZHU, ServerName 的属性为 Project1. ZHD-TUSHUGUANLI, Connected 属性值为 True, 这样就可

以真正地把客户端连接到应用程序服务器上了, COM/DCOM 会自动把这个应用程序服务器激活. 经过上述步骤后, 就能够利用 TClientDataSet 对远程数据进行存取.

第 2 步, 在数据模块中置入 4 个 TClientDataSet, 它们的 RemoteServer 属性值都设为 DCOMConnection1, ClientDataSet1, ClientDataSet2, ClientDataSet3, ClientDataSet4 的 ProviderName 属性值分别设为 DataSetProvider2, DataSetProvider3, DataSetProvider4. 设定完后, 就可以准备连接应用程序服务器并取得数据.

第 3 步, 把这 3 个组件的 Active 属性值都设为 True. 在设定 Active 属性值设为 True 之前, 可以先调整它们的 PacketRecord 的属性值. PacketRecord 代表 TClientDataSet 一次要从应用程序服务器上得到多少数据. TClientDataSet 的 PacketRecord 默认值为 -1, 这代表客户端应用程序一次会把所有的数据取到客户端. 如果应用程序服务器上的数据很多, 这样设定不仅会浪费很多时间和内存, 而且还会造成应用程序执行错误, 所以程序员应根据具体的情况来设定这个属性值. 本次设计把 PacketRecord 属性值设置为 10, 这表示客户端应用程序一次从服务器取得 10 条记录, 设定之后, 再把它们的 Active 属性值都设为 True.

第 4 步, 往数据模块中置入 4 个 TDataSource, 分别设置它们的 DataSet 属性值为 ClientDataSet1, ClientDataSet2, ClientDataSet3, ClientDataSet4. 这样客户端应用程序就可以从应用程序服务器中获取数据了. 此时, 整个本地数据模块就设计完毕, 如图 2 所示.

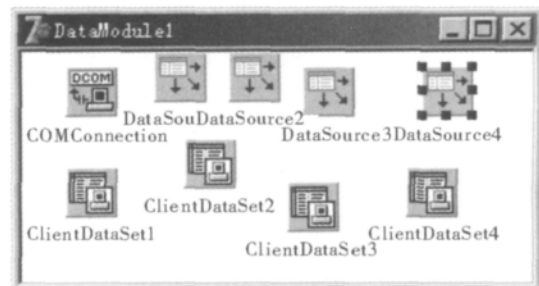


图 2 本地数据模块

其余的模块如系统登录模块、修改图书目录功能模块、修改学生记录功能模块、学生借阅历史记录功能模块、修改密码设置功能模块、图书查询与

借阅功能模块、还书功能模块等,这些模块就是从数据库中提取信息,十分简单.

3 结论

本文采用应用服务器开发分布式图书管理系统,给出了基于 Delphi 7 的应用服务器、远程数据模块和本地数据模块的设计.运行结果表明,该系统界面友好、延展性强、维护成本低,拥有分布式应用系统的大部分优点.

参考文献:

- [1] 符春. 中间件技术的现状及其发展[J]. 软件导刊, 2009, 8(9): 7.
- [2] Jingyong L, Yong Z, Yong C, et al. Middleware-based distributed systems software process [C]//Proc of Int Conf on Convergence and Hybrid Infor Tech 2009 (ICHIT 2009), Korea: Daejeon, 2009: 345 - 348.
- [3] 郑英姿. 浅谈中间件技术[J]. 中国高新技术企业, 2010, 168(33): 31.
- [4] Schiefer A, Gruhn V, Hrushchak R. VESBA: a middleware oriented architecture for virtualized embedded systems [C]//Proc of the 1st Workshop on Critical Automotive Applications: Robustness and Safety, Spain: Valencia, 2010: 47 - 50.
- [5] 王晓鹏, 苏亮, 韩伟红, 等. 大型分布式系统部署一致性的研究与实现 [C]//2006 年全国开放式分布与并行计算机学术会议, 西安 [s. n.] 2006: 12 - 14.
- [6] 别文群. 分布式计算机体系结构及中间件技术探讨 [J]. 郑州轻工业学院学报: 自然科学版, 2003, 18(3): 75.
- [7] 杨珺. 基于虚拟植物生长模型的分布式农业专家系统研究 [C]//2006 年全国开放式分布与并行计算机学术会议, 西安 [s. n.] 2006: 229 - 231.
- [8] 王启发. 面向数据库的中间件技术研究 [J]. 数学技术与应用, 2011(6): 194.
- [9] 李海波. 应用服务器中间件标准研制 [J]. 信息技术与标准化, 2008(5): 46.
- [10] 飞思科技产品研发中心. Delphi 7 基础编程 [M]. 北京: 电子工业出版社, 2003.
- [11] 朱汉民. Delphi 7 高级应用开发教程 [M]. 北京: 科学出版社, 2006.