

DNA 模型下商业数据交换网的改进设计^{*}

采振祥, 徐 明

(深圳大学教育技术与信息中心, 广东 深圳 518060)

摘 要: 根据微软的 DNA 分布式互连网体系结构开发模型改进了商业数据交换网的结构设计, 讨论了三层结构模式下系统的实现原理及相关技术, 并介绍了具体的设计和实现过程。

关键词: 分布式互连网应用体系; 三层结构; 组件; 电子商务

中图分类号: TP311 **文献标识码:** A **文章编号:** 0529-6579 (2001) 03-0033-04

文献 [1] 介绍了一种利用 Internet 在商场与供应商之间实现商业数据交换的实际应用系统, 其做法是: 加盟的商场租用本地 PSTN、ISDN、DDN 等线路, 以 C/S 模式通过客户端专用软件向中心数据库提供或获取数据, 供应商通过浏览器收发与这些商场交互的数据。该系统运行一年多来, 对流通领域企业电子商务活动的开展起到了积极作用。但随着业务的发展, 系统中存在的 C/S 两层结构设计模型也暴露出一些局限性: ①系统只能为同城商场提供很好的数据传输服务, 而不能为异地商场、本地商场的异地分店提供经济可靠的服务; ②系统接收许多商场的商业数据并生成数据仓库, 而用户同类业务单据往往分布在多个数据库中, 使原有模式数据查询效率下降, 产生应用瓶颈; ③用户的需求日趋复杂, 如某些商场希望数据以加密形式与供应商交换, 但原有模式下系统相应功能的修改和扩充不够灵活; ④业务链不断延伸, 商场增加了分店, 供应商同时与多家商场建立业务联系, 数据交换需要具体到分店一级, 数据操作不可能一次或只在一个数据库中完成, 需要增加中间运算与调度环节。

在原有系统结构的基础上, 难以有效地解决上述问题, 需要对相关部分作调整和改进设计, 以适应业务发展的要求。为此, 我们根据 Microsoft DNA 分布式互连网应用结构的思想, 在原系统中增加一个应用服务器层, 利用分布式组件对象来访问数据库和完成复杂商业逻辑计算, 提升系统的运行效率和扩充系统的功能。

1 基于 DNA 模型的三层应用结构

为顺应 Internet 应用发展的潮流, Microsoft 提出了 DNA (distributed internet application architecture) 分布式互连网应用体系结构, 目标是为基于 Internet 的分布式应用提供一个规范而高效的开发框架和运行环境, 其核心是通过 COM/DCOM 技术将 Web 与三层结构开发模式结合起来^[2]。同时, Microsoft 为 DNA 模型的每一层都提供了解决方案和底层服务支持。按照 DNA 的实现原理, 结合应用实际, 我们调整了商业数据交换网的系统结构, 将一些计算复杂的、容易造成系统运行瓶颈的、具有分布式数据库操作的服务按照 DNA 模型来实施, 如图 1 所示。改进设计后的系统是以“表示层/事务逻辑层/数据服务层”三层体系结构为构架的, 利用 COM 组件对象在中间层进行事务逻辑服务, 处理各种复杂的商务逻辑计算和演算规则。

表示层: 直接面向用户, 完成应用的前端界面处理。表示层以应用请求的方式调用中间层的商务逻辑组件, 完成相应的功能。

中间层 (事务逻辑层): 整个应用的核心, 负责处理表示层的应用请求, 完成商务逻辑的计算任务, 并将处理结果返回给用户。事务逻辑层通过 COM 进行事务处理, 并由 IIS 和 MTS (microsoft transaction server) 为各种应用组件提供完善的管理。IIS 负责事务逻辑层 Web 页面的管理, MTS 负责事务逻辑层 COM 组件的管理。

数据服务层: 是系统中对数据进行管理的部分, 由原系统数据库及数据仓库组成。在 DNA

^{*} 基金项目: 科技部 2000 年国家级科技成果推广资助项目 (2000E041D7800262)

收稿日期: 2000-12-15; 作者简介: 采振祥 (1957—), 男, 副研究员。

平台上, 允许选用 SQL Server、Oracle 及任何与 OLE DB 或 ODBC 兼容的数据源。由于中间层的出现, 数据库不再和每个活动客户保持一个连

接, 而是若干个客户通过共享商务逻辑组件共享数据库的连接。

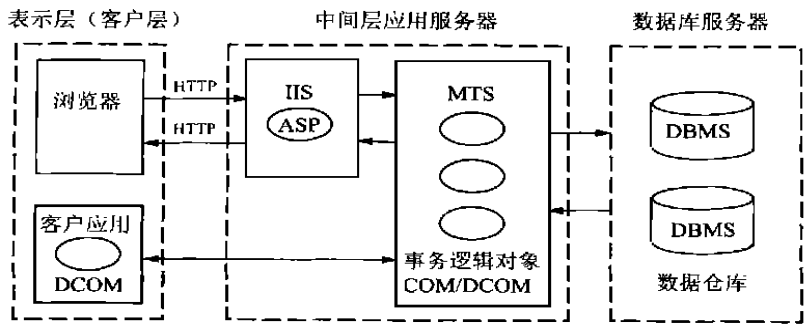


图 1 商业数据交换网的三层结构模型图

Fig 1 A 3-tier application architecture model of the business data interchange network system

2 组件对象的设计和调用

COM 组件是遵循 COM 技术规范编写的、以 Win32 动态链接库或可执行文件的形式发布的可执行二进制代码, 有很高的运行效率。COM 规范包括 COM 核心、结构化存储、统一数据传输、智能命名和系统级的实现等。常用的 COM 组件有两类: 用于中间层处理商务逻辑的 ActiveX DLL 和用于客户端表现结果的 ActiveX 控件。

DCOM 是一种分布式组件对象模型, 它是 COM 在分布式计算方面的延续, 为分布在网络不同节点的两个 COM 组件提供了互操作的基础结构。DCOM 支持 TCP/IP、HTTP 等多种网络通信规程, 能自动建立连接、传输信息并返回来自远程组件的答复^[3]。

通过系统分析, 我们可以确定系统需要多少个组件, 并划分每一个组件的功能, 然后根据组件对象的技术规范, 设计在本系统中使用的组件。这些组件一般包含两部分功能, 一是商务逻辑计算, 二是访问数据库/数据仓库。COM/DCOM 的设计与开发工具语言无关, 我们可以选择自己熟悉的编程工具, 如 VC、VB、C++ Builder 等常用工具。组件的调用方法比较简便, 以创建对象实例的方法进行, 既可以在 ASP 脚本中直接引用, 也可以在另一个 ActiveX DLL 中引用。例如, 在 ASP 页面中通过使用 Server 对象的 CreateObject 方法, 可以在服务器上启动 ActiveX 对象例程。

3 MTS 与组件的管理

MTS 是 DNA 结构中的组件管理器, 它为构建和分发各种组件提供了一个服务器端的运行环境。MTS 是一个基于组件的事务处理系统, 用于开发配置与管理高性能的 Internet 及 Intranet 企业级应用程序, 提供数据连接缓冲、线程缓冲、锁管理、事务服务等多种自动服务, 解决了多客户端利用组件频繁访问后台数据库等一系列问题^[4]。中间事务逻辑层的组件都是在 MTS 的控制下运行的并通过 MTS 访问各种数据源。

在如图 1 所示的结构中, IIS 担负页面的服务和管理的工作, MTS 负责所有商务逻辑组件 (包含页面所嵌入组件) 的服务和管理的工作, 两者协同工作组成中间层的应用服务器。在商业数据交换网中, MTS 所起的具体作用主要表现在: ①管理数据库连接。MTS 架设在应用程序和数据库之间, 使数据库只保持一个和 MTS 的连接而不是若干个和应用客户的连接; 应用程序只需一个和 MTS 的连接而不用因访问多个数据库而建立多个数据库连接, 使数据库响应更快, 从而提高系统的性能; ②管理分布式对象。MTS 允许将 ActiveX DLL 组件分配到网上的不同服务器上, MTS 记录下每个组件的位置, 负责组件与应用程序的通信; ③事务协调。MTS 提供了对多个对象及多种数据库进行事务协调控制的能力, 支持事务性组件、事务性 ASP 及对多个数据库的事务性处理。

4 具体设计及实现

原系统设计流程如图 2 所示. 在系统改进设计工作中, 最重要的是确定中间事务逻辑层的功能单位, 一方面要调整原来两层结构中各层的部分功能, 将一部分功能划分到中间层进行; 另一方面要根据市场的需求, 设计一些新功能, 发挥 DNA 结构的优势, 提高系统原有功能的执行效率, 增加一些在原系统中实现困难的、含有复杂商业逻辑的新功能; 其次是 COM 组件的编程实现和客户层上应用程序及动态 Web 页面源代码的改变. 整个设计过程如图 3 所示. 改进设计的核心工作是 COM 组件的划分及组件内部商务逻辑的确定, 我们的做法是: 客户对数据库的请求一般通过 COM 组件实现, 以提高数据访问的执行性能; 将某一个或某一类相似的应用请求划分为一个组件, 以方便组件共享和提高组件的可重用性; 组件的计算逻辑可以较为复杂, 但功能要单一, 以免影响组件的通用性. 原系统中大部分业务规则保存在数据库的存储过程 (Procedure)、触发器和商场端应用程序中, 如客户分组管理、数据分检与转发等^[1], 在本次设计中, 将其中很多业务规则移植到中间层执行, 我们重点设计或完善了以下功能:

(1) 供求撮合: 在原系统中, 已经设计有一定的商品供求信息撮合匹配功能, 是通过预定义撮合规则的办法, 如根据价格区间、类型、产地等内容寻找符合条件记录, 并通知信息需求者. 实现方法是将上述算法设计成存储过程, 在数据库系统后台执行, 该方法的缺点是撮合适应性差、智能性差, 占用了数据库系统开销, 影响系统性能. 新的设计通过组件将用户提交的多项条件进行智能组合, 实时生成撮合匹配条件, 查询供求信息数据库, 得到符合条件的记录, 并返回给用户.

(2) 单据管理: 将原系统的订单、销售单、结算对帐等单据查询功能改为通过中间层组件进行. 由于供应商一般均与多家商场有业务关系, 业务单据的查询实际涉及到分布式多数据库访问. 在各个单据查询组件中, 接受供应商用户请求, 利用原系统的客户分组管理功能, 得到用户与商场的业务关系, 分别查询来自这些商场的数

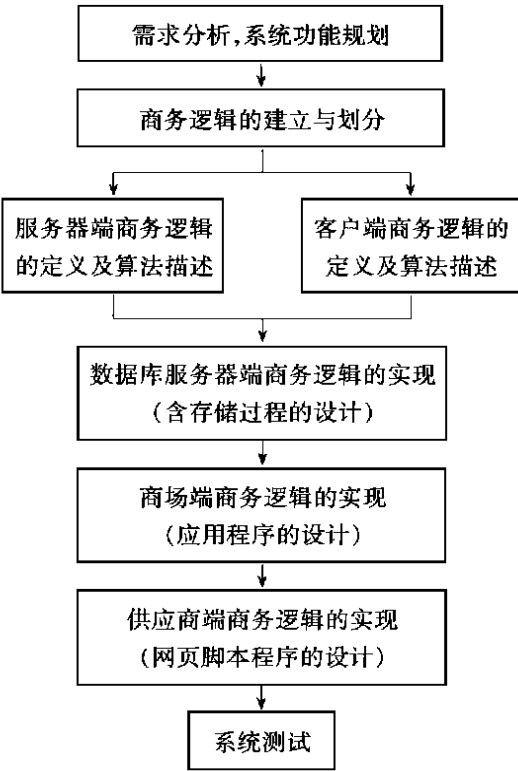


图 2 原系统设计步骤示意图
Fig.2 Schematic diagram of the original system design process

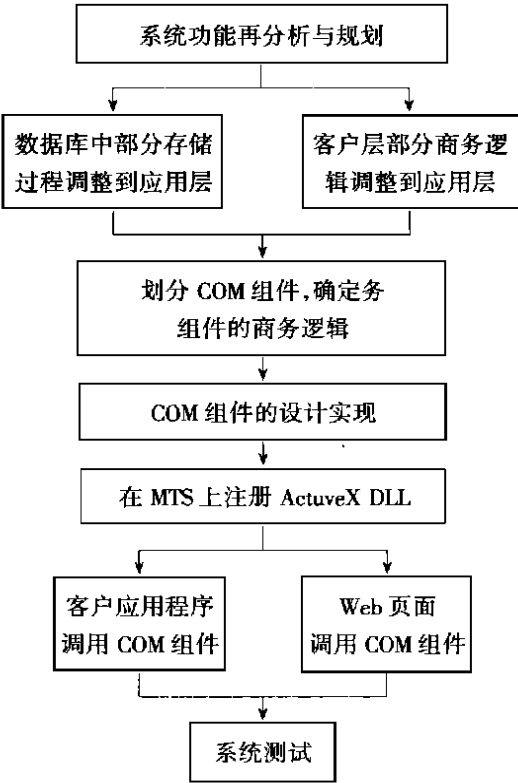


图 3 改进设计步骤示意图
Fig.3 Schematic diagram of the improved system design process

(3) 库存 (含分店) 查询: 供应商一般均与多家商场有供货关系, 供应商需要查询他在每一家商场的库存; 商场一般均开设多家分店, 供应商需要查询他在同一商场的多家分店中的库存情况. 库存组件利用原系统的客户分组管理功能, 得到供应商、商场及其分店的对应关系, 分别查询对应商场及分店的数据, 并将综合库存结果返回给用户.

(4) 数据加密与还原: 商场端定义加密算法, 数据加密后存储在数据库中. 设计解密组件, 包含与商场端对应的解密算法, 从数据库读取加密数据, 并进行解密工作. 供应商通过访问 Web 页面, 调用解密组件从而完成数据还原显示.

5 结 语

按照 DNA 模型来设计或改造电子商务系统, 能收到良好的效果, 不仅优化了系统性能, 而且能够实现以往 Web 系统中难以实现的复杂功能, 很有实用价值. 我们在实际设计过程中, 充分体会到基于 DNA 模型的电子商务系统具有以下一些突出特点:

(1) 系统性能提高. 在 DNA 模式下, MTS 在多线程支持下工作, 实现对 COM 组件的高性能管理. 若干客户通过共享组件和数据库连接,

降低了运算资源的开销, 减轻了数据库的负担, 提高了系统的吞吐能力及性能.

(2) 系统具有良好的可扩充性和自适应性. 当业务逻辑发生变化时, 不必改变任何 Web 页面或应用程序源代码, 只需更新事务服务器上的业务逻辑组件; 当需要增加新的业务功能时, 只要在原系统的基础上增加新的表示逻辑 (客户层) 和新的业务逻辑组件即可.

(3) 系统安全性大大提高. 业务逻辑放在事务服务器上集中统一管理, 避免了以往业务逻辑在客户端容易泄露的危险; 客户端不直接同数据库连接, 而是在 MTS 的管理控制下通过组件访问数据库, 保证了数据库的安全性; 通过增加数据加密组件的办法, 可以实现数据的加密存储和传输, 提高了数据的安全性.

参考文献:

- [1] 采振祥, 徐明. 商业电子数据交换网的设计与研究 [J]. 中山大学学报 (自然科学版), 2000, 39(3): 44-48.
- [2] Abemethy. COM/DCOM Unleashed [M]. 北京: 电子工业出版社, 2000.
- [3] 楼伟进. COM/DCOM/COM+ 组件技术 [J]. 计算机应用, 2000(4): 31-33.
- [4] <http://www.microsoft.com/com/tech/mts.asp>. [EB]

Improving the Design of Business Data Interchange System in Distributed Internet Application Architecture

CAI Zheng xiang, XU Ming

(Education Technology and Information Center, Shenzhen University, Guangdong Shenzhen 518060, China)

Abstract: An improved design of a business data interchange network system in the distributed internet application architecture released by microsoft is developed. The operational principles and relative techniques in 3 tier application architecture are discussed, and the design and implementation process is introduced.

Keywords: distributed internet application system; 3 tier; COM/DCOM; EC