

93-95

基于 DCOM 的 Web 与数据库集成系统的体系结构*

The Architecture of Integration of Web and Database System Based on DCOM

黄琳倩 冯玉才 吴恒山 郭 琰 TP393

TP311.13

(华中理工大学计算机科学与工程学院数据库与多媒体技术研究所 武汉 430074)

Abstract The integration of Web and database system has become the important issue in the field of database research with the development of network. In this paper, we present the implementation method of integration of Web and DBMS DM3 using Microsoft's DCOM. Utilizing the idea of component, we solve some problems of the integration, such as the performance of the whole system when the number of client increases, how to maintain the ACID characteristic of the transaction in the environment of the HTTP protocol, and so on.

Keywords Web, Database system, DCOM/COM, DM3-Web

1 引言

随着网络技术的发展, Web 日趋成为目前重要的信息系统。数据库系统如何与 Web 集成(DBMS-Web)成为数据库研究领域的重要课题。这一集成可以充分发挥 DBMS 高效的数据存储和管理能力,以 Web 为平台,将客户端融入 Web 浏览器,为 Internet 用户提供简单方便、内容丰富的服务。具体地,数据库系统与 Web 的集成具有以下优点:(1)客户端的平台独立性,客户端统一使用 Web 浏览器,从而省去了在 Client/Server 模式下为不同的平台开发客户程序的重复劳动。(2)系统的可伸缩性,易于扩充和修改。(3)对于用户来说,界面统一,易于使用。

我们自主开发的分布式关系数据库管理系统 DM3 与 Web 的集成方案 DM3-Web,以 Microsoft 的 DCOM/COM 技术为核心,实现了 Web 与 DM3 数据库系统的交互。通过对组件和访问机制的管理,改善了网络上客户数增加时的系统性能,提供了多种 Web 与数据库系统集成途径。

2 DCOM/COM 技术特点

COM 是一个标准规范,提供了客户和对象组件应遵循的一些标准,它支持组件的互操作性和二进制级的共享,允许不同的软件开发者用不同的语言、工具、甚至平台开发软件组件,这些由不同的软件开发者完成的组件,可整合在一起,部署在定制的环境中。同时

COM 也是一个面向对象的编程模型。DCOM 是在 COM 基础上的分布式网络的扩展,在 DCE RPC 上构造了对象远程调用(ORPC)层来支持远程对象。使用 DCOM/COM 技术的优势在于:

- (1)COM 组件是与语言无关的,提供了二进制级的共享,可用多种语言实现。
- (2)COM 组件具有良好的扩展性。COM 提供了一种实现同一组件不同版本的标准方法,易于系统的升级。
- (3)COM 可为用户提供透明的服务,而不必考虑服务对象的位置。
- (4)DCOM 支持多种通信协议。
- (5)将 Java 虚拟机(JVM)集成进来,使 Java Applet 能与 Internet 上的其它控件通信。

3 现有的 Web 与数据库连接的集成方法

目前,已有的 Web 与数据库集成方案大致有以下几种模型。

- (1)通用网关接口(CGI) CGI 方式是传统的 Web 与数据库系统连接的方案。CGI 是一组关于 Web 浏览器、Web 服务器、CGI 应用程序之间传递信息的规范。CGI 的优点在于:编码简单,可用多种语言实现,如 C/C++, perl, shell script 等等;程序的可移植性好,几乎所有的 Web 服务器都支持 CGI 程序。CGI 的不足在于当 Web 客户与数据库服务器进行交互时,由于 HTTP 协议的无态性,每一次请求都将经历一次连接-

*)本课题得到国防科技预研项目“基于客户/服务器方式的分布式数据库管理系统”的资助。黄琳倩 硕士生,研究方向为数据库;冯玉才 教授、博导,主要研究领域为数据库。

执行-断开的过程,系统开销大,效率低。

(2)专用 API 方式 由于 CGI 不能保持与服务器的连接,Netscape 和 Microsoft 分别推出了 NSAPI 和 ISAPI 来提高 Web 服务器与数据库服务器之间的通信效率和性能。与 CGI 不同,API 方式以 DLL 的形式存在,Web 客户与数据库服务器能保持连接,直到 Web 客户退出。API 与 Web 服务器是一个整体,即 API 依赖特定的 Web 服务器,不能跨平台。

(3)可移动代码系统(MCS) 随着 Java 的流行,出现了 MCS 模式。其主要思想是分布应用程序的代码,将它传送给客户程序,然后由客户端在用户的计算机上执行。Java 通过 JDBC 取得与数据库的连接。基于 Java 的方法提供了高度的灵活性、可扩充性、可移植性和健壮性。JDBC 提供了异构数据库系统之间的互操作性。

(4)分布式计算结构 这种连接模式以 OMG 的 CORBA 和 Microsoft 的 DCOM 为代表,支持异构平台上组件的无缝集成。它与以服务器为中心的信息系统相比,构造方法更为先进。利用这一模型,很容易实现客户对象与服务器对象的创建和访问的透明性。

4 DM3-Web 的体系结构

4.1 扩展的 Client/Server 结构——三层结构模型

传统的 Client/Server 结构将客户端和服务端分离,与原有的集中式结构相比,减轻了主机的负担,但是这种结构不利于管理,可扩充性不好。随着 Internet/Intranet 的发展,用户数目大大增加,对于通常的 DBMS 来说,其用户个数是有限的,这必然限制了同时进行数据库访问的用户数目,而且这种结构不支持异构数据库之间的互操作,开发者将针对不同的 DBMS 开发不同的应用程序,系统的管理、维护费用较高。因此,在 C/S 基础上,提出了其扩展模型——三层结构(图1)。

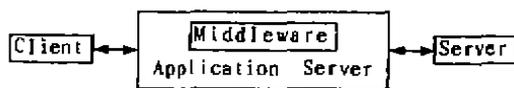


图1

在扩展的 Client/Server 三层结构中,客户端负责提供与用户交互的界面,而将一部分处理移到应用程序服务器。应用程序服务器通过中间件与客户端和数据库服务器进行交互,可以完成客户端请求的调度,执行应用程序的部分功能,进行安全控制、会话管理、故障恢复、事务控制等。

4.2 DM3-Web 的体系结构

DM3-Web 采用扩展的 Client/Server 三层结构

模型,为 DM3 数据库设计人员提供了开发从 Web 浏览器访问 DM3 的应用程序的开发工具。DM3-Web 的工作流程是:当 DM3-Web 服务启动后,由侦听器(Web-Listener)接收客户端的 URL,进行文本分析,确定对于 Web 页面来说,它是动态的还是静态的。若是静态的,则由文件系统来检索这个 HTML 文件,以静态 Web 页面的形式传给浏览器来显示;若为动态文本,则由 Web-Listener 将用户的动态请求传送给连接管理器。连接管理器根据请求的不同类型,将请求分配到各个不同的请求代理器。这些代理器提取符合 DM3 语法的 SQL 语句,交由 DBMS 服务器完成,将返回数据装配成相应的 HTML 文档,返回给客户端。DM3-Web 的体系结构如图2所示。

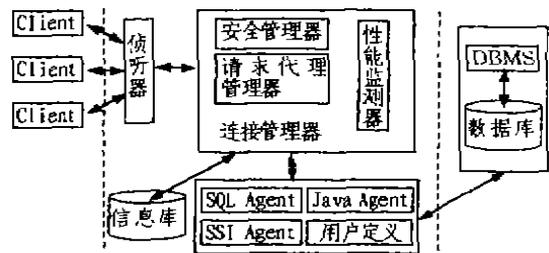


图2

(1)侦听器是 DM3-Web 服务器的主控部分,管理客户的服务请求,并将完成的 HTML 文档送回给请求者。DM3-Web 服务器允许有多个侦听器。

(2)连接管理器(Web-CM)管理客户与数据库系统之间的连接。它主要由请求代理管理器、安全管理器、性能监测器三部分构成。

请求代理管理器为用户创建一个服务线程,根据侦听器传来的消息,综合信息库中的有关缓冲区和请求代理器的配置信息,为用户请求分配一个服务请求代理器,负责对缓冲区的管理。缓冲区分为连接缓冲区、数据缓冲区和信息缓冲区。连接缓冲区保存客户的连接信息,以便实现连接组件的复用,数据缓冲区存放从数据库返回的结果集,信息缓冲区存放从信息库返回的配置数据。缓冲区的管理采用 LRU 算法。

安全管理器提供了连接访问授权检查以及独立于数据库的安全措施。

性能监测器定期监测缓冲区中的连接状态,对无法建立的连接返回错误信息。

(3)信息库存放有关缓冲区、安全和请求代理器的配置信息。信息库以磁盘文件的形式存放,由连接管理器负责调入缓冲区进行数据查询。信息库中的数据由系统管理员负责输入、配置。

(4)请求代理。不同的请求代理执行相应的处理功

能,实现了一套与 DBMS 连接的 API,它与 DBMS 进行交互,将 DBMS 返回的结果装配成 HTML 文档的形式并返回给客户端。考虑到上述的几种主流的数据库与 Web 的连接方案,我们设计的代理器有:SQL Agent,Java Agent,SSI Agent,并且允许用户自定义代理器。

SQL Agent 实际上是一个过程化的 CGI。为了克服 CGI 方式的不足,我们采用保持连接的方式来减小连接-断开所带来的开销。即在用户的一个事务内,连接都是保持的,直到用户发出退出请求或提交、回滚命令,利用 DM3 的存储过程来进行 SQL 请求,可以减轻网络压力,提高系统效率。

Java Agent 实现了通过 Java 语言访问数据库的方式。连接管理器激活 Java Agent 中的 Java 解释器,执行 Java 程序。Java 语言通过 JDBC 与数据库进行交互。目前,Java Agent 还不支持 Java Applet,

SSI Agent 实现了服务器隐含 (SSI) 方式,可实现以多种方式创建 Web 页面。通过内部代理桥 (Inter Agent Bridge),SSI 方式可利用 SQL Agent 访问数据库。

5 技术关键

5.1 请求代理组件的设计

每个请求代理是一个 DCOM 组件对象。我们将 DM3 提供的 API 函数用 DCOM 规范包装起来,提供对外统一的接口。对三种不同的请求代理,分别设计了不同的组件。请求代理组件的接口定义如下:

```
ISQL->Execute(BSTR sql-statement, bool bSuccess)
    // 执行 SQL 语句
    PrintHTML(BSTR sql-result, BSTR HTML-result)
    // 生成 HTML 文档
    Result-return(BSTR HTML-result) // 返回结果
Ijava->Invoke() // 激活 Java 解释器
    Result-return(BSTR HTML-result) // 返回结果
ISSI->Retrieve() // 检索扩展的 HTML 文档
    Execute(BSTR Command, bool bSuccess)
    // 执行服务器端隐含的程序
    PrintHTML(BSTR sql-result, BSTR HTML-result)
    // 生成 HTML 文档
    Result-return(BSTR HTML-result) // 返回结果
```

5.2 组件复用

DCOM 机制可以使一个组件为多个客户提供服务,缓解了客户数量受到数据库用户数量限制造成的问题。DM3-Web 以缓冲池的方式管理各个请求代理组件,减轻了连接-断开数据库时的开销。组件以四元组的形式表示为(组件类型 ID,数据库名,用户 ID, PASSWORD)。组件对外提供一个逻辑连接接口,由连接管理器分配内部 ID。当用户访问服务器时,若连接管理器分析出该用户具有相同逻辑接口,即上述四元组的各项均相同,则可以共享一个请求代理组件。这种多个客户共享一个请求代理组件的原则是:(1)用户在访问数据库时,在逻辑上是相同的,这样才能维护通过

组件操作的数据库不会产生逻辑错误。(2)基于事务级处理。事务是数据库对外提供服务的基本单位。Web 与数据库系统集成的一个困难之处就在于 HTTP 协议的无态性与数据库系统的 Client/Server 模式的有态性的矛盾,这一矛盾给保证事务的 ACID 特性带来了困难。因此,在一个事务期间,保持客户与数据库之间连接至关重要。一个客户在单个事务内必须独占一个请求代理组件,此时不能与其它用户共享。连接管理器中的请求代理组件的分配算法如下:

```
if(用户连接请求)
{
    if(存在内部 ID 相同的空闲组件)
        将该组件标志为 BUSY;
    else if(存在空闲组件)
        该组件建立与数据库的新连接,并将该组件标志为 BUSY;
    else if(连接缓冲池中已无可用组件)
    {
        向客户返回消息:连接请求过多,无法响应;
        return;
    }
    else
    {
        分配一个新的组件,建立与数据库的连接,将该组件标志为 BUSY;
    }
    将组件挂在 BUSY-组件队列中,返回组件接口指针
}
else
{
    if(断开请求)
        将组件标志为 IDLE,插入到 IDLE-组件队列中;
    else if(事务终止请求,如 COMMIT,ROLL BACK)
        将组件标志为 IDLE,插入到 IDLE-组件队列中;
}
```

结束语 DM3-Web 提供了一套 Web 与数据库系统连接的解决方案。利用 DCOM 技术,解决了多个客户共享数据库资源,并在基于 HTTP 协议的环境中保证了事务的 ACID 特性。同时采用组件模型使得系统具有可扩展性,良好的兼容性。概括起来,DM3-Web 实现了以下功能:(1)提供了 Web 与数据库系统的多种连接方式,如改进的 CGI 方式,Java 方式,SSI 方式。(2)提供了开放接口,允许用户插入自定义的请求代理器。(3)采用了组件复用,改善了当用户数增加时的系统性能。

参考文献

- 1 赵洪彪,周立柱.数据库系统与 WWW 的集成途径[J].计算机科学,1997,24(6)
- 2 陈羽中,麦中凡.基于分布式对象的软件构件[J].计算机科学1997,24(4)
- 3 Papaj R, Bursleson D 著,王燕,崔雨柏译.Oracle Web 数据库开发指南[M].机械工业出版社,1998
- 4 Stonebraker Michael, Betting On ORDBMS[J], BYTE, APRIL, 1998
- 5 陈澄,徐宏炳,王能斌. DBMS Web 支撑框架研究[J].计算机研究与发展,1998,35(6)