

基于ASP技术实现异构数据库联合使用

北京理工大学计算机科学与工程系 刘艳梅 宋瀚涛 梁允荣

本文论述了ASP技术开发Web应用的体系结构,并进一步阐明了应用该结构进行异构数据库系统联合使用的实现。该系统的实现也为电子商务应用中异构信息源集成提供了思路。

异构数据库联合使用的意义

网络计算模式已得到广泛应用,Internet技术使全球范围信息传递易于实现,直到近期发展起来的电子商务倍受业界关注。在企业内部使用Internet技术建立的Intranet使企业能在信息基础上实现现代化运营。军事上为适应信息战,加强了信息采集与处理,无论是战术情报系统或是多兵种的战场情报系统乃至区域军事情报系统,从总体上看,都是网络计算模式。在上述应用中,网络是支撑,信息源是基础,处理与应用是精髓。就信息源而论,含有多种数据库服务器是客观现实,因历史原因,有层次、网状及关系数据库系统,还包括有近几年发展起来的对象关系数据库、多媒体数据库及面向对象数据库。对一个大的企业,各部门使用不同数据库系统也是屡见不鲜,不是人们刻意追求异构数据库,而是现实中难以避免。以一个大的公司为例,它在北京、上海、广州,甚至国外都有分公司,在不同时期,不同分公司可能安装不同的数据库管理系统,如oracle,DB2,sybase,informix,SQL server等等。即使是一个经过信息规划的企业部门,因时间推移,人事变迁,以及数据库技术的发展和数据库市场的变化,都可能造成异构数据库的局面。

数据库的异构性还体现在:

- (1)计算机平台的异构:从巨、大、中、小型机到工作站、PC以及手持机。
- (2)操作系统的异构:UNIX、WINDOWS、OS/2等
- (3)网络的异构:LAN、WAN、以太网总线结构与令牌环结构等。

异构数据库是一个既与各种数据库有密切联系,又有别于各种数据库的新的研究领域。它的研究目标是对地

理上分布的多个异质数据库(Heterogeneous Databases HDB),在尽可能少地影响其本地自治性的基础上,构造具有用户所需要的透明性的全局数据库,以支持对各数据库的全局应用和各异质数据库之间灵活的信息交换和共享。

异构数据库的研究经历了十余年的发展,在此期间取得了不少成果,也出现了一些原型系统。然而异构数据库究竟有什么样的功能,至今无统一认识,这与数据库技术不断迅速发展也有密切关系。Internet/Intranet的迅猛发展,给这一论题注入了新的活力,提供了更加先进的技术途径和更加广阔的应用前景,在电子商务领域的应用就是一例。

下面我们就北京理工大学的UUHDB(United Use of Heterogeneous Databases)系统的实施做进一步的阐述。

UUHDB集成的技术途径—ASP技术

ASP是Microsoft的Web server IIS3.0以后新增的功能,ASP具有下述优点:

- 1.ASP允许将脚本制作语言VBScript,JavaScript或Jscript与HTML结合起来制作Web页面,克服了HTML的局限性,使得Web页面从静态变为动态,这是一个飞跃。
- 2.ASP的工作机制消除了对于Web浏览器的依赖性,即客户机方的浏览器可以采用任何厂商的产品。浏览器向IIS请求ASP文档,IIS解释其中的脚本语言,并生成标准的HTML,传送给客户机浏览器。所有的处理都是在服务器上进行的。
- 3.Active Server的另一大优点是代码的安全性。页面的访问者所能看到的只是服务器方的处理结果。

4.ASP提供了许多开发者可利用的内置组件对象,可管理从变量到提交表单的所有内容。

5.ASP允许用户开发自己的COM组件来封装事务逻辑。虽然不用自制的组件,仅使用ASP也能实现绝大部分的事务处理,但相对组件而言,存在着明显的缺陷。首先,ASP脚本是通过IIS解释执行的,这要比运行一个编译为二进制代码的组件慢得多,不利于多用户环境下的快速响应;其次,脚本不能将事务逻辑分离出来,使得整个脚本结构复杂,逻辑不清晰,可读性差,编程不便,更不要说由于事务逻辑变化需要进行的系统维护了。而组件是可重用的,当多个页面需要进行相同的事务处理时,调用同一组件即可;当进行类似的系统开发,需要进行相同的事务处理时,可方便地使用已有的组件;当事务逻辑变更时,不必改变页面脚本,只需调整相应的组件,即可适应事务逻辑的改变。

在ASP中调用组件的代码如下:

```
<%
Dim property1
set Obj_instance_name=Server.createObject
("Component_name.Obj_name")
Obj_instance_name.method_name(parm1,parm2,...)
property1=Obj_instance_name.property_name1
...
Obj_instance_name.close
Set Obj_instance_name=Nothing
%>
```

可见,在ASP中使用组件是非常容易的。只有把HTML、脚本语言和组件等结合在一起,才能建立一个动态的、功能强大的Web应用系统。ASP对自制组件的支持,使得它获得了无限的可扩展性。可以肯定,组件对象与ASP结合将成为构造Web应用系统的最常用方式。

异构数据库的联合使用是一个极为复杂的应用,并且随着信息技术的发展,不断会有新的情况出现,因此,开发功能各异的COM组件,将使异构数据库的集成变得简单、灵活,更具动态性。

另外,ASP还可借助于MTS进行自制COM组件的管理。MTS是一个基于组件的事务处理系统,提供了包括数据连接缓冲、线程管理、事务服务等多项服务,很好地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。当多个用户同时访问主页时,MTS将组件放入线程池,自动进行管理,从而避免网络拥塞。

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应用客户的直接连接,减轻了数据库的负载,加快了响应。

MTS允许将一系列组件分配到网络上的不同应用服务器上,并记录下每个组件的位置,负责这些组件与应用程序的通信。还可以将某个组件复制后分布到多台服务器上,以便MTS在各个副本间平衡负载,使应用得到最优化的响应。这一功能是因为MTS是建立在DCOM基础上的。COM使得同一机器中的组件能够互操作,而DCOM则更进一步地让网络中不同机器中的组件互相沟通。DCOM构造于RPC(Remote Procedure Call)技术之上,并且使用TCP/IP作为通信协议。

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。这一点对异构数据库中的事务级处理是非常重要的。

图1即是一个ASP应用系统的框架,以浏览器作为客户端,IIS负责应用逻辑层Web页面的管理,MTS负责分布在网络中的事务逻辑COM组件的管理,可以根据COM组件负载的情况,增减应用服务器的数量。

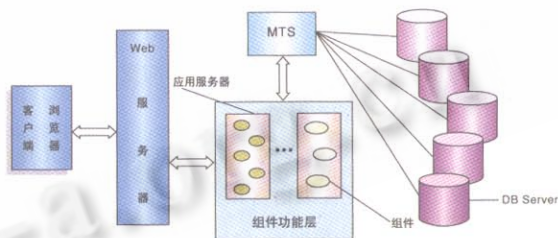


图1 ASP数据库应用系统结构

这是一个N-Tier结构的系统框架,以浏览器作为客户端,IIS负责应用逻辑层Web页面的管理,MTS负责分布在网络中的事务逻辑COM组件的管理,可以根据COM组件负载的情况,增减应用服务器的数量。

UUHDB系统功能组件及其实现

如图2所示,B/S结构的UUHDB系统采用Web页面作为用户界面来收集用户输入的GSQL语句。我们可以使用目前流行的页面开发工具如FrontPage98等进行页面设计,生成原始的HTML。随后,对生成的HTML进行编辑,加入脚本语言,如收集页面中的GSQL语句。提取

GSQQL后,则转入GSQO的处理。GSQQL的功能是由自定制的COM组件完成的。本系统的大部分工作集中在COM组件的开发上。

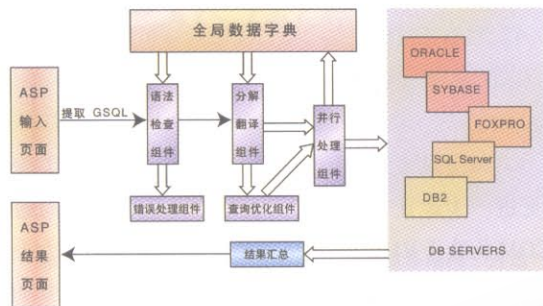


图 2 UUHDB 系统结构图

首先, UUHDB 系统拥有全局数据模式(Global Data Model), 并将纳入联合使用的各异构数据库(LDB)映射到全局模式上, 使用户感觉就象使用一个单一数据库一样, 从而为用户透明地使用异构数据库打下基础。全局数据模式是通过全局数据字典(Global Data Dictionary)实现的。全局数据字典描述整个网上数据库的定义及分布情况, 如全局表对应的局部表名、全局表中字段对应的局部表字段、局部表的存在结点名等, 它是全局查询的基础。该字典是动态刷新的, 如路由表一样, 所以, 由字典监控器同各数据库通信, 将与字典有关的各数据库的变化情况反映在字典中, 做到操作的实时性。

其次, UUHDB 系统提供全局语言(Global SQL)包括数据定义语言、数据操纵语言和查询语言。最终用户将通过全局语言操纵异构数据库。全局语言的定义我们参照了SQL92 标准。全局语言区别于通常的数据库语言之处在于全局语言所涉及的表由两级模式构成, 即逻辑上的全局表和物理上的局部表。全局语言包括下列功能:

- (1)全局数据定义语言。包括创建、修改、删除全局表。
- (2)全局数据操纵语言。包括插入、删除和更新全局表中的数据。
- (3)全局查询语言。全局查询语言在格式上同SQL92 完全一致, 形式上查询的是全局表, 实质上是对跨平台的异构表的操作, 因此它的实现是极其复杂的。

(4)事务处理。事务这一概念在数据库中是十分重要的, 它意味着“全部或全不”, 是数据库操作正确的保证机制, 对于异构数据库尤为重要。UUHDB 系统中的事务处理是通过GSQQL 结合MTS 的事务管理完成的。

在设计完成上述GSQQL功能的组件时,我们注意遵循以下原则:

- (1)组件粒度不宜过大,使每个组件完成的事务逻辑相对单一,有助于组件的重用。
- (2)将对于数据库的操作封装在组件中,保证数据库的安全性。

· 语法检查组件。用户的GSQQL 语句首先需要进行语法检查确认正确后方能进行下一步处理。该组件实现了check 方法, err_str 属性, 调用该方法可以输入GSQQL 进行关键字析出和语法判别, 并对应全局数据字典检查涉及的名和字段名称。出错将错误信息写入err_str 属性。

· 分解翻译组件。parsing 方法将正确的GSQQL 语句分解为面向各局部数据库的SQL 语句, 包括表名、列名、相应函数的替换, 并将各局部SQL 写入相应的属性中。如涉及异表连接, 还需调用查询优化组件。

· 并行处理组件。各局部数据库在接收到分解翻译后的SQL 语句后, 其执行应是互不干扰的并行进行, 但在任何一个发出多个局部SQL 的应用程序中, 这种处理却是串行的, 即只有在一个局部数据库完成其局部SQL 的执行后, 另一局部数据库方才开始工作。这必然导致系统资源的浪费和效率的降低。因此, 设计多线程的多数据库并行处理组件是解决此问题的关键。

Delphi 提供了一个多线程对象 ThreadObject, 它是调用Windows32 API CreateThread 来建立一个操作系统的多任务对象。通过重载Execute 方法, 完成此组件功能。

· 查询优化组件。网上传输的速度是制约查询效率的关键因素, 而各LDBMS 的查询方式、速度等的不一致又使处理变得更为复杂, 特别是对涉及多个异构数据库的连接查询提高处理效率尤为重要。该组件功能为采用启发式方法寻找一个最优的连接半连接执行序列, 以减少网上传输量为目的。

最后, 本系统还提供安全控制机制(Security Control)用于保护数据库中的数据不被非法用户访问, 安全机制由三个层次构成: UUHDB 系统级别的用户管理, 通过COM 组件访问数据库的权限管理, MTS 对组件用户的权限管理。

结论

采用ASP 技术, 结合COM/DCOM 开发事务逻辑组件, 使用MTS 进行组件管理, UUHDB 系统为异构数据库在未来电子商务领域的应用奠定了坚实的基础。事实证明, 这一方案是可行的、高效的。■