

基于 Socket 的 数据库网关系统



赵明喜 费耀平 (长沙中南大学铁道校区网络中心 410075)

| | |
|-----|---|
| 摘要 | 在电子商务网站的建立中,最重要的是对数据库的访问。由于电子商务的实时性,使得数据库的访问响应速度要求很高。本文利用SOCKET的机制和数据库的存取进程建立起一套数据库访问的网关系统。 |
| 关键词 | SOCKET Store Procedure OLE DB COM 线程池和线程同步 数据库网关 |

1 系统概括

在任何网站系统中数据库是必不可少的数据存储系统,数据库的吞吐量以及响应速度直接影响着网站的速度。除去数据库本身的性能,作为数据库的使用者,如何提高数据库的速度基本上有两大类方案:一是数据库的逻辑设计,另一个就是应用程序访问数据库的路径和方法。前者在此不做讨论。我们知道应用程序访问数据库有很多种方法,对于特定的数据库供应商基本上都提供了对自己的数据库的一套特定语言的访问接口,像 Oracle 的 OCI、MS SQL 的 DB-Library 等等,但是使用这些接口进行应用程序的直接开发,则是很烦琐,对开发人员的技术素质要求很高。另外我们知道现在进行网站开发用的都是使用脚本语言的 ASP、JSP 等,这些程序都有一个共同点:上下文的无连接 (stateless)。如果使用通用的数据库访问办法,那么每个要求调用数据库的页面,都必须和数据库进行一次连接,每次连接都会消耗掉系统的很多资源,当网站的访问量达到一定程度后,数据库性能会直线下降或者死

掉。针对这种情况,作者设计出一个中间件 (Middleware),作为访问数据库的一个网关,特别适合于日访问量教大的中小型网站。

2 系统结构

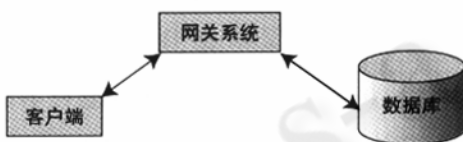


图 1 框架图

下面是系统的总体说明。

系统由客户端、网关系统和数据库三大块组成。客户端是最终用户的最终实现,它只要遵循下文所示的表 1 和表 2 的协议就可以以自己认为方便的方式对系统进行封装,本系统只实现了其 COM 形式,实现了 IDispatch 接口,因此可以用于 ASP 中。由于篇幅的原因本文不对其做详细的说明。网关系统是整个系统的核心。作为一个中间层从严格意义上讲,它只是一个通道,但是正是因为它的存在使得系统变为一个 3- 层的结构,其意义是不言而喻的。它起着

访问数据库、响应客户端的请求以及系统调度的作用。数据库部分和客户端一样都是最终用户二次开发要使用的东西。以本系统为基础的所有数据库操作都必须通过存储过程来实现。客户端和网关系统是通过 SOCKET 的字符流的形式来实现通信的。网关系统访问数据库则通过特定的数据库所提供的访问方法来实现的,如: SQL SERVER 的 OLE DB 和 DB-Library (Microsoft 已经宣称对 7.0 后的版本不提供进一步的支持) 以及 Oracle 的 OCI (Oracle Call Interface)。之所以用这些方法是因为它们可以极大地提高访问的效率。

3 网关系统技术实现

网关系统是整个系统的核心。它起着访问数据库、响应客户端的请求以及系统调度的作用。下面对它的实现做详细的说明。

3.1 系统流程

系统在启动的时候将寻找用户设置的配置文件,系统根据配置文件 (包含数据库访问信息以及每个数据库包含的用户存储过程及其全

局唯一编号,参见后文)启动访问数据库的线程,并且启动 SOCKET 的监听端口以监听客户端的访问请求。当客户端有请求时,系统的处理流程如图 3 所示。

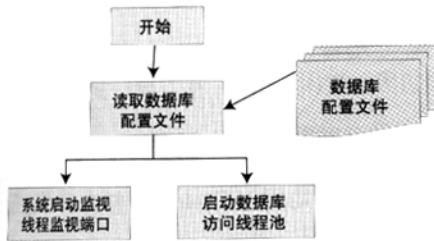


图 2 系统初始化

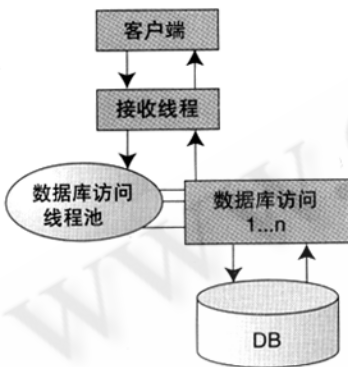


图 3 系统处理流程

系统在监听到有客户端的请求后,将会启动一个接收线程以处理和客户端的通信,接收线程分析请求指令,并且将数据库请求发送到特定的线程池,由线程池来安排特定的数据库访问线程来处理该请求,当数据库访问线程处理完数据请求后,将处理结果放到特定的缓存并且以事件的方式通知接收线程数据已经准备好。接收线程在等待到数据准备好的事件后,从缓存中取出数据并且发送给客户端,最后该线程自动销毁。从而完成一次客户端的请求处理。

3.2 类的结构

线程基础类和数据库访问类是所有工作线程类的基类。线程基础类由 C 语言的线程处理函数(_beginthread-

ex 等)进行包装,以提供线程类的基本功能。数据库访问类是用于访问数据库的基础类,它通过数据库访问线程类来实现线程级的访问。监听线程类、接收线程类和线程池处理类以及数据库访问线程类都是从线程基础类派生而来。接收线程类用于和客户端通信以及将访问请求提交给线程池处理类。线程池处理类是建立在 I/O 完成端口上的线程类,它通过 WINDOWS 2000 的线程池处理函数对数据库访问线程进行调度。

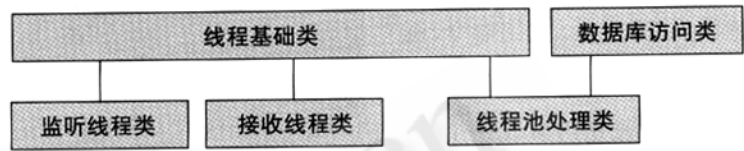


图 4 系统的工作处理类结构

3.3 线程调度和同步

线程调度主要是由系统的线程池来完成。程序创建一个 I/O 完成端口并且创建一个等待该端口的线程池(CreateIoCompletionPort),同时程序根据系统配置参数启动若干个数据库访问线程,数据库访问线程检测前面的 I/O 端口(根据端口句柄),如果检测到有数据库请求则分配一个线程处理,从而完成对线程的调度。这是一种非常出色的结构,它使少数线程能够有效地处理大量的工作项目,同时它使得程序设计人员可以节省大量的时间和精力。由于本系统对线程的处理特别频繁,线程间的同步以及对共享变量的保护显得尤其必要。在本程序中线程间的同步主要采用事件内核对象,它包括普通的事件内核对象以及 SOCKET 事件,数据的保护采用临界区系统内核对象,需要保护的有对日志文件的写以及记录系统运行状态的变量。

3.4 通信及通信协议

网关系统和客户端的通信是建立在 SOCKET 基础上的。系统在使用 SOCKET 的时候使用的都是 Windows Sockets 2 扩展库函数库。底层的传输协议采用 TCP/IP 协议。采用重叠(Overlapped) I/O 的方式来发送和接收 SOCKET,采用事件内核对象进行线程同步以及 SOCKET 事件的触发。SOCKET 的字符流是通信的唯一方式,客户端和网关系统采用自定义的协议,其格式如下:

表 1 用户请求指令数据定义

| 数据头 | 分隔符号 | 存储编号 | 参数 | 结束符号 |
|-------|------|------|-----------------------|------|
| 总数据长度 | 0x2 | nnnn | 0x2 @para1.@para2.... | 0x0 |

表 2 结果数据定义

| 数据头 | 数据行 1 | | | | | | 集结束符 | |
|-------|-------|------|-----|------|-----|-------|------|-----|
| 总数据长度 | 0x3 | 数据 1 | 0x2 | 数据 n | 0x1 | 数据行 n | 0x3 | 0x0 |

由于通信是以字符流的方式，在不是很特殊的情况下用非打印字符0X1, 0X2, 0X3作为分割符号是可行的。另外系统可以处理多结果集的情况。

3.5 数据库访问类的实现

数据库访问类封装了访问数据库的基本功能：建立连接、发出命令、接收数据处理结果以及错误处理。数据库访问类之所以独立封装成一个类，是因为针对不同的数据库其最有效率的访问方法不是唯一的，它是为了更好地进行程序移植。在本文的实现中采用的是OLE DB的数据库访问办法，OLE DB是Microsoft推出的旨在统一数据库访问的一种高效的对不同类型的数据进行访问的访问方式，其分为两个部分Provider和Consumer。SQL Server和Oracle都有Provider。在本文中对Provider没有涉及，使用的只是Consumer。由于OLE DB是基于COM的，因此数据库访问类相当于一个COM的客户端系统，它使用Consumer的一系列接口创建数据库连接、会话和命令以执行对数据库的存储过程的调用。

3.6 系统的文件系统

系统的文件系统分成两个大的系统：输入文件和日志文件。输入文件是系统的参数配置文件，在该文件中保存有连接数据库的线程数、通信地址及端口以及所要访问的数据库的登陆参数和想要访问的所有的存储过程名称和编号，该编号在整个系统中是唯一的。下面是一段配置参数的样例：

[数据库通信参数]

IP=192.168.0.1

Port=1976

记录日志=0



[数据库管理1]

数据库连接数=1

服务器=pcguy

用户名=sa

密码=

数据库=pubs

存储过程1编号=2001

存储过程1名称=GetStatic

存储过程2编号=2002

存储过程2名称=GetMy

日志文件是系统在处理请求过程中所产生的文件，它包括每次请求所产生的日志，记载有访问的存储过程编号、访问结果的状态、访问的时间（包括进入系统时间、在线程池中等待的时间以及访问数据库所用的时间），另外还有一个日志文件就是系统错误文件包括数据库产生的错误，如数据库不在运行、对应存储过程不存在或者参数错等等。这些日志文件对于诊断系统故障以及优化存储过程都有重要的作用。

3.7 系统的安全

由于本网关系统可以运行在一个LAN中，因此对其在安全方面没有做很大的设计。不过在客户端向

网关系统发送连接请求的时候，必须提供一个用户名和密码以对其进行授权访问。由于处理速度方面的原因，系统对数据的字符流并没有进行加密处理。同时由于笔者在这方面知识的欠缺也是导致系统在这里不完善的地方。

4 实现及改善的方向

系统采用VC++6.0作为开发工具，网关系统以Win2000作为系统平台。客户端采用ATL包装可以用于Win9x平台以及NT平台。进一步的完善可以以EJB的形式实现JAVA访问的客户端工具。虽然该系统已经实现并且可以满足日常的系统的需求，但是在系统的安全和负载均衡上还有很多改善之处。■

参考文献

- 1 Anthony Jones, Jim Ohlund Windows网络编程技术, 机械工业出版社, 2000.
- 2 Jeffery, Windows核心编程, 机械工业出版社, 2000.

