

文章编号: 1005-0523(2005)01-0066-05

基于 Delphi 的多层数据库应用系统连接技术的研究

吴小林¹, 蒋先刚², 高艳锦¹

(华东交通大学 1. 电气与电子学院; 2. 基础科学学院 江西 南昌 330013)

摘要: 以一个 SQL Server 数据库为例, 介绍了基于 Delphi 7 的多层数据库应用系统的几种连接方式, 并且讨论了不同连接方式的效率和应用范围。

关键词: DataSnap; BDE; dbExpress; ADO/MTS/COM+

中图分类号: TP311

文献标识码: A

1 概述

当前的数据库应用系统中, 按其结构划分为两类, 一类是两层结构的数据库应用系统, 另一类是多层结构的数据库应用系统。

两层结构的数据库应用系统有两种构造方式, 这就是客户/服务器(C/S)结构与浏览器/服务器结构(B/S)。它的特点是数据库服务器端管理数据, 而客户机或浏览器显示、访问、和更新数据。两层结构的数据库应用程序使数据和程序分离, 因此可以分别进行开发和维护, 同时也降低了数据库服务器的工作量, 减小了网络的流量, 具有较强的安全性, 但当启动的客户端程序或浏览器较多时, 服务器的性能会明显下降。同时还存在客户端程序难以分发, 运行效率较差等问题。

为了解决两层结构应用系统的不足, 人们提出了多层数据库应用系统的结构模式, 具有代表性的是三层结构, 它是在两层结构的基础上建立一个中间层——应用服务器, 将有关数据访问和存取的相关程序和动态链接库运行在应用服务器端, 因此大大减少了与数据库服务器连接的次数, 且客户端程序不必关心数据操作的细节, 企业逻辑的创建和修

改都在中间层完成, 提高了整个系统的可靠性。

应用于多层数据库应用系统的常用数据库主要有 SQL Server、InterBase、Sybase、IBM DB2、MySQL、Oracle 等数据库。SQL Server 无疑是 Windows 平台上最受欢迎的关系型数据库, 而且 MS SQL Server 逐渐向大型关系型数据库的方向发展。而 SQL Server 数据库 ADO 和 BDE 都提供了良好的数据访问技术, 且未来 dbExpress 将会提供最好的解决方案。

本文主要以一个交通图像监控系统的 TransportImage 数据库为例, 讨论多层数据库应用程序的连接技术。交通图像监控系统是一个实时系统, 它处理的信息量很大, 接入的客户端较多, 要求整个系统负载能力强, 查询效率高。

2 基于 DataSnap 的多层应用系统的前端连接技术

在多层数据库应用系统中, 客户端与应用服务器端常处于不同的物理区域, 它们之间的数据传送成了影响系统运行效率最关键的技术, 在 Delphi 中要建立与应用服务器的连接, 客户程序必须使用一个或几个 DataSnap 连接组件, 以及 TClientDataSet 等组件。其结构如图 1 所示, 首先 DataSetProvider 组件把读取的 DataSet 数据源压缩并切割成一段一段的

收稿日期: 2004-05-20

作者简介: 吴小林(1975-), 江西金溪人, 华东交通大学在读研究生, 主要研究领域: 工业测控, 数字图像处理。

数据包 (data packet), 通过各种传输协议如 DCOM、TCP/IP、HTTP 协议传给前端, 前端的 DataSnap 连接组件会把接受到的数据包再还原成 DataSet 传给 ClientDataSet 组件, Data Control 组件就可以将 ClientDataSet 作为数据源对它进行操作。

DataSnap 技术是 Delphi 开发多层数据库应用系统的核心技术, 它不但提供了基于 DCOM、TCP/IP、HTTP 三种传输协议的数据传送方式, 还通过 SimpleObjectBroker 组件使系统轻易具备容错能力和负载均衡能力, 使得 Delphi 开发的多层应用系统不但结构简洁、执行效率高、应用范围广, 而且安全稳定。而且现在 DataSnap 还能够同时在 Windows 和 Linux 上执行。

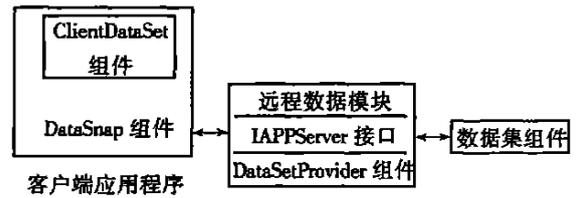


图 1 客户端与应用服务器的连接

DataSnap 分别通过 DCOMConnection、SocketConnection、WebConnection 三个组件来实现 DCOM、TCP/IP、HTTP 三种传输协议的。其中 DCOMConnection 组件提供的功能最为完善, 而且执行效率比 SocketConnection 要快 5% 以上, 另外安全性也比较好。表 1 是三种组件的主要性能对比。

表 1 DataSnap 连接组件的性能比较

	性能特点	系统配置	执行效率	适用的范围
DCOMConnection	功能比较完善, 安全性好	配置简单	执行效率较高	WIN2000 及以后的版本, 或者系统有严格安全控制的要求
SocketConnection	功能简略	配置简单	效率一般	应用范围广泛, 特别适应在 internet 或有多个路由器的网络
WebConnection	功能简略	需要的系统资源较高	效率一般	适应服务器端有防火网的系统

3 多层应用系统中应用服务器与数据库的连接技术

3.1 基于 BDE 的数据库连接技术

BDE 是 Borland 最早也是最主要的存取各种数据的引擎, 它的功能一直随着 Delphi 的进步在不断的改善和增加, 因此 BDE 支持的数据库比较完善, 但因为本身机制的问题, 在开发多层数据库应用系统的时候, 它的负载能力不强。基于 BDE 的多层数据库应用系统是把原先的 C/S 系统的前端应用程序的 BDE、SQL Links (或 ODBC)、DataModule, 转移到应用服务器上, 客户端只剩下一个执行文件及 DB-Client.dll, 而移到应用程序服务器的 Datamodule 变成了一个 COM 程序。前端程序与应用服务器上的 COM 程序通过 DCOM 的机制互相沟通, 而应用服务器的 COM 程序则是通过 BDE 与 SQL Links (或 ODBC) 传送 SQL 到后端数据库, 整个多层应用系统的结构如图 2 所示。Delphi 提供的 TDataSetProvider 组件会把读取到的 DataSet 压缩并且切割成一段一段的数据包 (Data Packet) 通过 DCOM 的机制传给前端, 前端的 MIDAS.DLL 会把接收到的数据包再还原成 DataSet, 传递给前端程序中的 TClientDataSet 组件, 前端的 Data Control 组件就可以通过 TClient-

DataSet 组件来显示这些 DataSet 数据。

使用 ODBC 连接数据库要先注册 ODBC 数据源, 建立用户 DSN。这是至关重要的一步, 否则就无法实现对数据库的访问, 本文是访问 Transport Server 服务器上 TransportImage 数据库。进入控制面板双击 ODBC 图标, 在用户 DSN 一栏中选择添加, 先选择安装数据源的驱动程序为 SQL Server, 然后设置数据源的名称为 Image1, 最后更改默认的数据库为 TransportImage, 此时按测试数据源按钮, 若连接成功, 那么打开 BDE Administrator 就可以发现, 刚刚建立的数据源 Image1 就会自动加入到 Database 一栏中, 成为 BDE 的一个别名。利用 DCOM 机制连接客户端和应用服务器端使用的是 TDCOMConnection 组件, 属性 ComputerName 指定应用服务器所在的计算机, 属性 ServerName 标志应用服务器。

3.2 基于 dbExpress 的数据库连接技术

dbExpress 是一组新的组件、技术和驱动程序, 程序员使用它可连接到各种数据源, 再配合不同的数据库连接 DLL 文件, 以处理后端数据库中的数据。dbExpress 具备了跨平台的能力, 能够同时在 Windows 和 Linux 平台以及未来的 .NET 上使用, 并且提供了快速、丰富的数据处理能力。

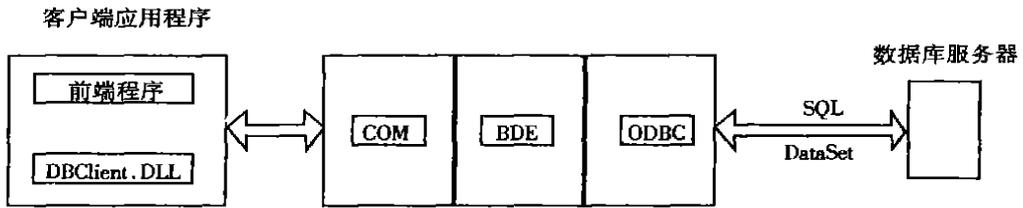


图2 基于 dbExpress 的多层系统结构

但 dbExpress 使用单向游标访问数据,而且 dbExpress 本身没有内建任何缓冲机制,它只有和 DataSnap 技术结合才能对数据进行有效的处理.因此使用 dbExpress 组件来开发数据库应用系统时,一般要用到四、五个组件来达到数据访问和编程功能,结构如图 3 所示.一个是 SQLConnection 组件,用来为数据库提供一个到 dbExpress 驱动程序连接,另一个就是 dbExpress DataSet 组件,通过执行一个 SQL 语句或调用一个存储过程来提供数据,还有 DataSnap 组件、DataSetProvider 组件、ClientDataSet 组

件等,应用程序通过 dbExpress 使用 SQL 语句从数据源中取得数据,当数据源根据 SQL 语句产生结果数据集并且返回给 dbExpress 时,dbExpress 会把结果数据集交由 DataSnap 技术管理,再由 DataSnap 来管理和修改数据.通过 DataSet 取得的数据事实上就是由 DataSnap 在缓冲内存中的数据,只有当程序员真的需要把缓冲内存中的数据更新回数据源时,DataSnap 才会通过 SQL 语句把经过修改的数据更新回数据库.

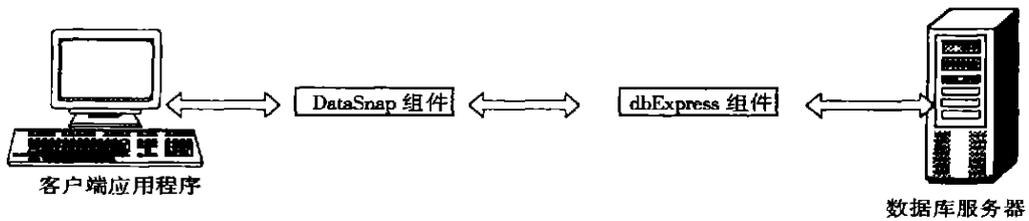


图3 基于 dbExpress 的多层系统结构

dbExpress 的核心是由数个简洁的接口 (Interface) 组成的,这些接口定义了如何跟数据库厂商接口沟通的 ISQLDrive、如何连接数据库的 ISQLConnection,如何对数据库下达命令的 ISQLCommand 如何控制 Cursor 的 ISQLCursor,以及存取数据库 Metadata 的 ISQLMetadata.这些接口定义的目标是达到简易、高效率、它们和 JAVA 的 JDBC 有非常类似的概念,但 Borland 提供了 DataSnap 来巧妙的结合这些接口,因此提供了比 JDBC 高数倍的效率,正因为 dbExpress 使用单向游标访问数据,而且本身没有内建任何缓冲机制,因此访问数据也特别快,它和 DataSnap 技术的完美结合,也特别适合开发多层数据库应用系统,并且可以轻易地移植到 Linux 上,这些优势都是传统的开发方式无法轻易达成的.当然目前 dbExpress 支持的数据库还非常有限,它的许多功能还在不断的完善中.

它通过封装 OLE-DB 来存取各种不同的数据. OLE-DB 定义了统一的 COM 接口作为存取各类异质数据源的标准,藉由 OLE-DB,程序员就可以使用一致的方式来存取各种数据.但 OLE-DB 太底层化,在使用上非常复杂.而 ADO 组件以 COM 技术封装 OLE-DB 为 ADO 对象,大量简化了程序员开发数据存取的工作.同时许多厂商也推出了 ADO 驱动程序,使得 ADO 应用日趋广泛.

3.3 基于 ADO/MIS/COM+ 的数据库连接技术

MIS/COM+ 作为应用系统的中介软件,已经成为 Microsoft 的核心技术之一. MIS/COM+ 不但能够稳定地执行应用系统的企业对象,以服务客户端的请求,更能够提供在分布式环境和异质数据库之间保护数据的事务管理能力,还能够有效地利用各种系统资源,增加中介软件的执行效率. MIS/COM+ 提供的主要功能有事务管理、资源管理、安全机制、连接池等技术.虽然 MIS/COM+ 也可以通过 BDE 来存取数据,但是性能还是比 MIS/COM+ 通过 ADO 要差一些.基于 ADO/MIS/COM+ 的多层应用系统

ADO 是 Microsoft 存取通用数据源的标准引擎.

如图 4 所示

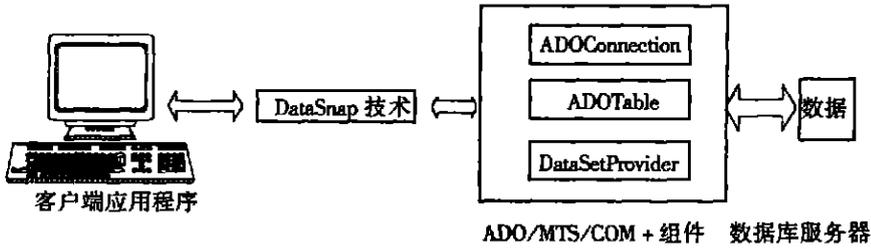


图 4 基于 ADO/MTS/COM+ 的系统结构

开发存取 Transport - Server 服务器的 TransportImage 库 VehicleImage 表的多层应用系统首先要在应用服务器端点选 File/New/Other/ActiveX/ActiveXLibrary, 建立一个新的 ActiveXLibrary 项目, 然后再点选 File/New/Other/Multitier/Transaction DataModule 在此项目中建立一个新的 MTS/COM+ 数据模块 MTSImage

在 MTS/COM+ 数据模块中放入 ADO 组件 ImageADOConnection、ImageADOQuery、ImageDataSetProvider, 来连接数据库服务器, 设置如下

```
ImageADOConnection.connectionstring := "Provider =SQLOLEDB.1;Persist Security Info=False;User ID=sa;Initial Catalog=TransportImage;Data Source=Transport - Server;Use Procedure for Prepare = 1;Auto Translate = True;Packet Size = 4096;Workstation ID = Transport - Server;Use Encryption for Data = False;Tag with column collation when possible = False"; //连接 Transport - Server 服务器数据库 transportimage;
```

```
ImageADOQuery.connection := ImageADOConnection; //
```

```
ImageADOQuery.SQL := "select * from VehicleImage"; // 连接 VehicleImage 数据表;
```

```
ImageDataSetProvider.Dataset := ImageADOQuery;
```

接着点选 Run/Install MTS Objects... 菜单安装数据模块 MTSImage 到套件组件中。

在客户端应用程序主窗体中放置 ImageDCOMConnection, ImageClientDataSet, ImageDBGrid, ImageDataSource 等组件来读取数据, 设置如下

```
ImageDCOMConnection.ComputerName := Transport - Server; //定位应用服务器
```

```
ImageDCOMConnection.ServerName := MTSImage; //标志应用服务器
```

```
ImageDCOMConnection.Connected := true;
```

```
ImageClientDataSet.RemoteServer := ImageDCOMConnection;
```

```
ImageClientDataSet.providerName := ImageDataSetProvider; //连接应用服务器的 ImageDataSetProvider 组件
```

```
ImageDataSource.DataSet := ImageClientDataSet;
```

```
ImageDBGrid.DataSource := ImageDataSource;
```

当将 ImageClientDataSet 的属性 Active 设为 True 时, 应用程序便可取得 VehicleImage 的数据。

3.4 三种不同数据库连接技术的比较

BDE 是从 Paradox 引擎演化来的, BDE 访问关系数据库的状态下, 由于采用混合机制而导致一些资源的浪费, 例如在 BDE 中至少保存了 4 份的元数据, 其搜寻过程中经常使用双向游标, 还使用了 3 层的缓冲内存, 另外 BDE 也会自动产生 SQL 语句, 并且使用复杂的配置设置, 这导致了 BDE 在多层应用系统的执行效率较低, 稳定性差。

而 dbExpress 基本上是一个以处理 SQL 命令的数据访问引擎, 它和 BDE 源自 Paradox 访问引擎, 可以说是完全不同的设计方式, dbExpress 在连接数据库时只使用单向游标, 且在内部不提供任何缓冲内存, 而是直接使用客户端的 DataSnap 的缓冲内存以避免缓冲内存的浪费。另外 dbExpress 在执行时才根据需要访问元数据以避免无谓的数据访问, 此外 dbExpress 不会自动产生 SQL 语句, 而只会执行客户端赋予它的 SQL 语句, 这样客户端可以使用最有效率的 SQL 语句而不会受到 dbExpress 的干扰, 另外 DataSnap 提供了根据数据自动产生 SQL 语句的能力, 因此 dbExpress 不需要在重复提供这个功能。

ADO 是 Microsoft 主推的数据访问引擎, 它通过封装 OLE - DB 来存取不同的数据, 能够有效地配合 MTS/COM+ 数据库缓冲池技术, 以及分布式事务管理功能。

下面一组数据是在 WIN/98 系统下试验 100 次得出的平均数据连接与读取的运行时间:

表 2 三种数据引擎运行时间比较

	连接数据库的时间(s)	读取 20 个记录的时间(s)	读取 1000 个记录的时间(s)
BDE	1.925	0.042	3.571
dbExpress	0.639	0.067	1.326
ADO	0.859	0.051	3.310

通过试验可以看出, dbExpress 总是比 BDE 更快地连接数据库, 大致上 dbExpress 提供了比 BDE 快 30%

表 3 三种数据引擎主要性能比较

	性能特点	执行效率	稳定性	应用范围
ADO	易于使用、低内存支出和占用磁盘空间较小, 查询能力强, 能有效发挥 MTS/COM+ 的功能, 具有较强的延展性,	较好	最好	适合存取 Microsoft 的数据库, 适合应用于 Web 等延展性较强的分布式应用系统
dbExpress	访问引擎简单, 体积小又有效率, 数据处理功能丰富、灵活, 但支持的数据库有限, 许多功能还在不断完善中	效率最高	一般	比较适合访问关系型数据库, 适合于跨平台的应用系统
BDE	功能比较完善, 支持的数据库较全面, 应用范围广, 但配置比较复杂, 负载能力有限	效率一般	较差	适合访问基于文件的数据库, 如 Paradox、dBase 等

4 结论

本文以一个实时的交通监控系统的 SQL Server 数据库 TransportImage 为例讨论了基于 Delphi 7 的多层数据库应用系统的连接技术. 连接和运用数据库的策略是非常重要的工作, 选择使用适当的数据访问引擎和技术对于应用系统来说是非常重要的, 这将决定应用系统能否有效地处理数据, 甚至会影响应用系统的稳定性, 但同时我们也发现, 先进的技术还要进行适当的设置, 例如在试验中我们发现当 dbExpress、ADO 设置不好时, 其效率反而远远低

以上的连接速度, 在连接 SQL Server 2000 的试验过程中, dbExpress 要比 BDE 快一倍多时间. 而且在处理比较大的数据量的时候, dbExpress 表现得更为优秀一点. 但我们从整个连接试验过程发现, ADO 每次得出的运行时间相差不大, 而 BDE、dbExpress 的运行时间则相差比较大, 尤其 BDE 相差更大, 这说明 ADO 在稳定性方面表现的更好一点.

BDE/dbExpress/ADO 三种数据引擎主要性能比较如表 3 所示.

于 BDE 访问数据库的效率. 因此对 dbExpress/ADO/BDE 访问引擎来说, 恰当的系统设置也是影响访问效率一个关键的因素.

参考文献:

- [1] 李维. Delphi 7 高效数据库程序设计[M]. 北京: 机械工业出版社, 2003.
- [2] 蒋先刚. 基于数据库的电子商务网站的 Delphi 实现技术[J]. 计算机工程与应用, 2002.
- [3] 李维. Delphi 5. x ADO/MTS/COM+ 高级程序设计篇[M]. 北京: 机械工业出版社, 2000.
- [4] 陈灿煌, 陈周造. Delphi 5 彻底研究[M]. 北京: 中国铁道出版社, 2001.

Study of Connected Technology in Multi-Tier Application System Based on Delphi

WU Xiao-lin, JIANG Xian-gang, GAO Yan-jin

(1. School of Electrical and Electronic Eng., East China Jiaotong Univ., Nanchang 330013; 2. School of Natural Science, East China Jiaotong Univ., Nanchang 330013)

Abstract: Taking a database of SQL Server for example, different connected methods of multi-tier application system are introduced by Delphi in this paper. This paper also analyzes each method's efficiency and performance.

Key words: DataSnap; BDE; dbExpress; ADO/MTS/COM+