

基于 .NET 的 WEB 数据库中间件的设计与实现

徐迎晖

(北京邮电大学 信息安全中心, 北京 100876)

摘要:为使多种数据库和 WEB 功能的结合更加简便统一,文中提出一种基于 .NET 平台的数据库中间件。它基于 .NET 平台,利用 ASP.NET, ADO.NET, COM 等技术,需要在服务端和客户端分别进行开发、安装和注册,包含 .NET 和 VC6 两个版本。可以在 Internet 和 Intranet 上直接实现 WEB 功能。兼容各种主流数据库,如: Oracle, SQL Server, Sybase, DB2 等,对于不同数据库不需要重新编译,内外网使用相同的类和接口。在监控 SCADA 系统中的应用表明,软件的部署和开发都得到了简化,达到了较快的访问速度,获得了较好的效果。

关键词:中间件; .NET; 组件; ADO; ASP

中图分类号: TP311.131

文献标识码: A

文章编号: 1673-629X(2006)09-0007-02

Design and Implementation of WEB Database Middleware Based on .NET

XU Ying-hui

(Information Security Center, Beijing University of Posts and Telecom., Beijing 100876, China)

Abstract: To make the integration of variant databases and WEB more convenient and consistent, a database middleware based on .NET platform is proposed. Based on .NET platform, this middleware uses many techniques such as ASP.NET, ADO.NET, COM. It requires developing, installing and registering in server and client respectively. Two versions (.NET and VC6) are included. Web function can be realized in Internet and Intranet directly. Many widely used databases are compatible, such as Oracle, SQL Server, Sybase, DB2. Compilation is not necessary for different databases, and same classes and interfaces are provided for internal and outer network. In its usage in SCADA system, the system software disposition and development are simplified, visiting speed is high, and favorable performance is exhibited in practical application softwares.

Key words: middleware; .NET; component; ADO; ASP

0 引言

在基于 Windows 的 WEB 应用中,往往涉及多种数据库,存在各种连接技术。在内网的数据库访问中,可使用各种访问方案,如: OCI, ODBC 等。这种多样性,使得许多应用软件的开发工作量大,维护困难。文中提出一种数据库中间件,基于 .NET 的 ASP.NET、ADO.NET 和组件技术,可以兼容各种主流数据库,如: Oracle, SQL Server, Sybase, DB2 等。可以在 Internet 和 Intranet 上直接实现 WEB 功能,内外网使用统一的动态库和接口,包含 .NET 和 VC6 两个版本。并达到了较快的访问速度,从而使应用软件的部署和开发得到简化。

文中介绍了中间件的原理和实现方法,包括服务端的开发和安装、客户端的开发和安装注册等,并介绍了在监控 SCADA 系统中的应用。

1 原理

Visual Studio .NET 框架增加了许多 WEB 开发、远程访问和部署功能^[1]。它提供了 WEB 应用程序模型 ASP.NET,建立在 .NET Framework 的编程类之上,包含简化 WEB 应用开发的控件集和结构。.NET 包含了基于 XML 的数据访问技术 ADO.NET,具有良好的互操作性、可维护性、可伸缩性等性能,实现了数据和数据源的分离。.NET 支持 SOAP,这是一种基于 XML 的、用于在 WEB 上交换结构化和类型信息的简单的轻量协议,该协议是模块化的,并具有很强的扩展性^[2]。COM 则是微软公司定义的二进制的对象访问的标准,应用程序或对象可通过 COM 公开其功能和接口。

如图 1 所示,在 WEB Server 程序中,使用 ADO.NET 技术访问数据库,它提供多种速度不同的访问方式,对于 SQL Server 和 Oracle,可使用快速的专用连接方法,对于不支持 OLEDB 的数据库,可采用 ODBC 连接,这样形成了包含各种主流数据库的快速统一的访问方法。在数据库的访问中,使用数据集实现数据绑定,提供了统一的访问方式,也实现了数据和数据源的分离。

收稿日期: 2005-12-19

作者简介: 徐迎晖(1977-),男,湖南长沙人,博士研究生,研究方向为网络信息安全、信息加密与隐藏;导师:杨义先,教授,研究方向为信息安全、密码学及其应用、信号与信息处理等。

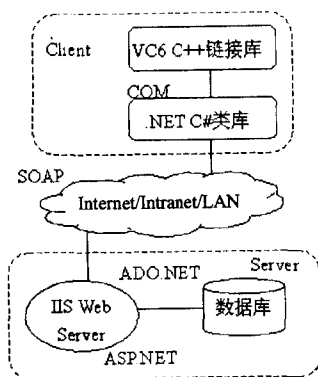


图 1 WEB 数据库中间件原理图

在使用 SOAP 进行访问中,为了适应不同大小的数据库,需要设置数据通信量的大小。使用 .NET 形成 WEB 访问的 C# 类库,同时包装成 COM 输出。然后利用 VC6 或其他 Windows 环境的 C/C++ 开发工具对此 COM 进行封装,形成动态链接库(dll)。这样得到的 dll 就可以实现以 WEB 方式的多种数据库统一访问。由于客户端和服务端之间采用 WEB 方式的连接,所以在客户端不需要安装数据库的客户端,只需要注册 COM 组件,简化了应用软件的安装,而且对于不同的数据库也不需要重新编译动态库^[3]。

2 实现方法

2.1 概述

由于使用了 .NET 的 WEB 技术,服务端和客户端都需要单独进行开发和安装,需要设定 IP 地址,在客户端需要注册 COM 组件^[4]。

2.2 服务端的实现

为了适应不同大小的数据库,需要在 .NET 应用程序中重新设置每次访问的最大数据量,可编辑 .NET 生成的应用程序的配置文件,修改 <httpRuntime> 元素,这个过程在服务端和客户端都需要进行。

用 VC.NET 开发服务端程序时,先创建“ASP.NET WEB 服务”类型的 C# 项目,这要求服务端必须安装 Windows IIS。在 C# 项目中建立 WEB 服务,此服务将自动添加到 IIS 中。在 C# 程序中,使用 ADO.NET 连接数据库,其中,SqlConnection 用于连接 SQL Server 数据库;OracleConnection 连接 Oracle 数据库;OleDbConnection 和 OdbcConnection 分别用于连接 OLE DB 接口和 ODBC 接口的数据库,按照上述的次序选择连接方法,可以得到较快的访问速度,所以客户端进行访问时指明数据库种类可以提高性能。

服务端程序和数据库之间的数据传递通过 DataTable 进行,首先读取指定表的数据,存于一个 DataTable 实例中,这样实现了数据和数据源的分离,对 DataTable 进行数据操作后,通过其成员函数 Update() 修改数据库,完成数据库的操作。DataTable 实际上完成了对多种数据库的封装,这也是 .NET 的特点。

在服务端程序中,设计一个类封装统一接口,可以包含数据库访问的各种功能,如:获得数据库和表的信息,读记录,修改、增加和删除记录。这些接口以基于 SOAP 的 WEB 方式公开。安装程序包括 2 个部分:.NET Framework 和服务端程序。WEB 服务的安装程序通过 .NET 的“安装和部署项目”的“WEB 安装项目”开发,将上述服务端程序生成的 .asmx, .disco, .config 文件添加到 Web 安装项目中,生成 .msi 文件,完成 XML Web Service 的安装程序。使用时,需要先在服务器上安装数据库服务端或客户端软件,然后在服务器上安装 .NET Framework,接着运行上述生成的 .msi 文件,这样完成了 WEB 服务的安装。

2.3 客户端的实现

客户端的实现需要生成两个类库和一个安装程序。先用 Visual .NET 按“C# 类库”建立“C# 项目”,利用“添加 WEB 应用”包含服务端 WEB 服务的接口,直接生成 C# 接口类库 dll,然后在项目中设置注册 COM 组件,这样就生成了包含 WEB 访问接口的 COM 组件。

利用 VC6 或其他 Windows 环境的 C/C++ 开发工具可以导入上述 COM 组件,在 VC6 下可使用 #import 指令,经过接口的封装形成 dll。关于 .NET 应用程序的配置文件,可使用 Microsoft WSE(Web Service Enhancement)进行访问,dll 运行时若没有此文件则自动创建,同时可以在运行时设置服务端的 IP 地址,用于访问不同位置的数据库。

对于 VC6 或其他不直接支持 .NET 的开发工具,在使用上述 dll 前需要进行安装。安装程序主要完成如下工作:

- 1) 安装 .NET Framework;
- 2) 注册 COM 组件,使用 tlbimp 注册 C# 类库 dll 中的组件;
- 3) 拷贝 .NET 配置文件,可以设置中间件专用的环境变量,也可拷贝至 Windows 的 system 目录,可不用建立专用环境变量。

若使用 VC6 进行开发,在主程序初始化时需要判断 .NET 配置文件,访问时若指定数据库类型则可获得较快的速度,也可同时访问多个数据库实例。

3 应用

对于各种监控 SCADA 系统,有如图 2 所示的典型结构。在应用程序开发和使用,可将 WEB 数据库中间件的服务端安装于“服务器”中,然后设置每次访问最大数据量。然后在“工作站”和“远程客户机”上安装中间件的客户端,不需要安装数据库的客户端^[5]。这样内外网使用相同的类库,系统可将应用数据存于服务器的数据库中,可被远程客户机通过中间件直接访问,内网程序可用 VC6 或其他工具开发,远程客户机程序可用 .NET 开发。已经在电力调度、GPS 监控等系统中成功使用,获得了满意的效果。

(下转第 11 页)

```

procedure Tgps. Map2MouseDown ( Sender: TObject; Button:
TMouseButton; Shift: TShiftState; X, Y: Integer);
//将鹰眼图中的点作为主图的中心
var
  Screanx, Screany: Single;
  Mapx: Double;
  Mapy: Double;
begin
  Screanx := x;
  Screany := y;
  Map2. ConvertCoord ( Screanx, Screany, Mapx, Mapy, miScreen-
ToMap);
  Map1. CenterX := Mapx;
  Map1. CenterY := Mapy;
end;
procedure Tgps. Map1MapViewChanged (Sender: TObject);
var
  tempFea: CMapXFeature;
  tempStyle: CMapXStyle;
begin
  if m_. Layer. AllFeatures. Count = 0 Then
//创建初始的矩形
  begin
    tempStyle := CoStyle. create;
    tempStyle. RegionPattern := miPatternNoFill;
    tempStyle. RegionBorderColor := 255;
    tempStyle. RegionBorderWidth := 2;
    tempFea := Map2. FeatureFactory. CreateRegion
      (Map1. Bounds, tempStyle);
    m_. Fea := m_. Layer. AddFeature(tempFea, EmptyParam);
  end
  else
    begin //根据主图的范围变化矩形的大小

```

```

    m_. Fea. Parts. Item[1]. RemoveAll;
    m_. Fea. Parts. Item[1]. AddXY(Map1. Bounds. XMin, Map1.
Bounds. YMin, EmptyParam);
    m_. Fea. Parts. Item[1]. AddXY(Map1. Bounds. XMax, Map1.
Bounds. YMin, EmptyParam);
    m_. Fea. Parts. Item[1]. AddXY(Map1. Bounds. XMax, Map1.
Bounds. YMax, EmptyParam);
    m_. Fea. Parts. Item[1]. AddXY(Map1. Bounds. XMin, Map1.
Bounds. YMax, EmptyParam);
    m_. Fea. Update(EmptyParam, EmptyParam)
  end;
end;

```

4 小 结

组件式 GIS 的出现带来许多传统 GIS 工具无法比拟的优点,把 GPS 行业通用的功能提取出来进行封装,做成构件,在以后的开发过程中,开发者可以直接调用相应的构件,只需针对相应的功能简单地将属性进行设置,这将会大大提高开发的效率及开发软件的质量。

参考文献:

- [1] 李慕华. 分布式 WebGIS 构件化的研究和实现[D]. 北京: 北京大学, 2000.
- [2] 林 春, 王 波, 贺贵明. 利用 MapX 实现的组件式 GIS 技术[J]. 计算机工程, 2002(5): 217-218.
- [3] 段凌宇, 鲍远律, 张旺生. GPS Vehicle Navigation System [J]. 南京航空航天大学学报(英文版), 1998(2): 172-178.
- [4] 夏红霞, 周 宏, 杨红云, 等. 基于 GIS/GPS 车辆监控系统实现及关键技术[J]. 微机发展, 2004, 14(8): 100-102.
- [5] 飞思科技研发中心. Delphi7 组件与分布式应用开发[M]. 北京: 电子工业出版社, 2003.

(上接第 8 页)

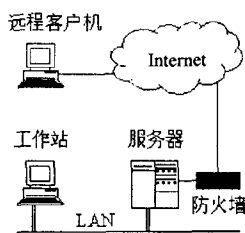


图2 WEB数据库中间件应用示意图

4 结束语

为使多种数据库和 WEB 功能可以高效简便地集成,在 .NET 基础上,利用 ASP.NET, ADO.NET, COM 等技术,形成了基于 WEB 的数据库中间件,它需要在服务端和客户端分别进行开发、安装和注册。兼容各种主流数据库,对于不同数据库不需要重新编译,内外网使用相同的

类和接口,包含 .NET 和 VC6 两个版本,并达到了较快的访问速度。在电力调度和 GPS 监控等实际应用中表明,软件的部署和开发都得到了简化,获得了较好的效果。

参考文献:

- [1] 孙 卿, 滕远道, 徐明毅. Microsoft .NET 和 J2EE 的构成技术与比较[J]. 微机发展, 2004, 14(11): 90-93.
- [2] 马建红, 王万森, 卫权岗. 基于 SOAP 协议的 Agent 通信语言的研究[J]. 微机发展, 2005, 15(6): 39-41.
- [3] 美国微软公司. Visual Studio .NET MSDN[Z]. 2003.
- [4] 李 飞. Web 服务异步调用的研究与实现[J]. 微机发展, 2005, 15(11): 96-98.
- [5] 管在林, 王艳红, 周 微, 等. Web 服务及在区域性网络化制造中的应用研究[J]. 计算机应用研究, 2005, 22(4): 210-212.