

# 利用 .NET 技术构建分布式人口管理信息系统

黄贵平 (中国科学院研究生院 100039)

**摘要:**本文在分析了人口数据信息特点的基础上,提出了采用分布式数据库技术实现人口数据管理系统的体系结构,并通过一实例阐述了使用 ASP.NET、ADO.NET、链接数据库技术实现对分布式数据库的访问。为了实现查询优化,系统采用了 C# 下的多线程编程技术,实现对远程分布式数据库的并行访问,从而缩短了系统的响应时间。

**关键词:**NET 分布式数据库 多线程 查询优化

## 1 前言

随着数据库技术的发展,特别是分布式数据库技术的进步,很好地满足了当今行政部门及企事业单位的业务、办公需求。对于政府行政管理部门或者那些在地理上分布较广,但管理上又相对集中的企业集团来说,基于计算机网络的分布式数据库应用系统可以将许多分散在各地的部门或单位孤立的数据联接起来,形成一个有机整体。一方面,对最终用户来说,这个系统是透明的,用户不需要关心数据的来源及分布,另一方面,作为政府各部门或企业各分支机构来说,可以继续在本地区维护自己的数据,同时在需要的时候可以很方便地、透明地访问本系统中分布在异地的数据。不但如此,B/S 开发技术及应用模式的出现,可以为最终用户提供一个简单的、统一的、免安装的、交互的、瘦身的客户端应用软件环境,大大方便了用户的使用。本文通过对国家级人口管理信息系统的分析,提出一种在 SQL SERVER 环境下的分布式数据库应用系统实施方案。

## 2 系统的总体结构设计

### 2.1 人口管理信息的分析

本文所讨论的人口信息管理系统是国家级信息系统,该系统拟对全国人口数据信息进行管理。按每个人的基本信息含项目 40-50 个,占用 400-500 字节估算,全国 13 亿人口,数据存储量将达到 580GB 左右,不仅数据量十分庞大,而且该系统地理分布广,系统接入点多,每个接入点的自治性要求较高,具有典型的分布性的特点。由上述分析可以看出,传统的集中式的

数据库应用开发技术不论从系统管理上,而是从可靠性、信息的检索速度等方面都难以胜任如此巨大的跨地域的数据库应用系统。而分布式数据库系统却能较好地解决上述问题。

### 2.2 数据库系统的分布设计

数据分片采用水平分片的模式,将全国 13 亿人口信息数据实体,按个体户口所在县(区)进行选择,从而得到互不重叠的数据片段,每个人员都属于一个特定的片段,保证了系统的完备性。数据片段的位置分配采用非冗余分配,将每一个片段放在访问量最大的地点,即相应的县(区)。

人口数据信息包括个案基本信息,如姓名、性别、年龄、民族、出生年月、户口所在地、住址等,还包括其他信息如个人的身份证信息、图像信息、生育史信息、避孕史信息、人口流动信息等。由于数据项多,在设计数据库时要分析数据表之间的约束关系。

### 2.3 人口管理信息系统的拓扑结构

系统的拓扑如图 1 所示。

如图 1 所示,整个系统采用了 B/S 模式体系结构,共分成四级,即国家-省-县-乡镇,这四级通过网络联成一个整体,构成一个分布式数据库应用系统。在国家、省、县三级均设有局域网环境及数据库服务器,但在乡镇级不设数据库,而是通过拨号方式访问县里的数据库,具体说明如下:

(1) 各行政县的人口数据存放在县局域网内的数据库服务器内,同时,在县局域网内另设一台远程访问服务器,提供本县所有乡镇的拨号联接。县局域网内的授权用户可以对本县内的人口数据信息直接进行本地访问,具有自治性,对本省内其他县人口数据的查询

访问,可以通过在本地服务器上建立省服务器的链接服务器,进行分布式查询。

(3) 省级的网络环境和县级基本一致,即省级有自身的局域网,在该局域网内安分别安装有数据库服务器,WEB 服务器,在省级设有全省的人口信息数据库,该数据库内的数据是通过 SQL SERVER 2000 的复制技术来实现。省人口信息数据库通过发布/订阅模式实现对本省内所有行政县人口信息数据库的数据复制,从而形成一个全省统一的人口信息数据库。由于人口信息的实时性要求不是很高,数据的更新也不是很快,因此,可采用快照复制技术完成。省人口信息数据库可以提供省级本地的

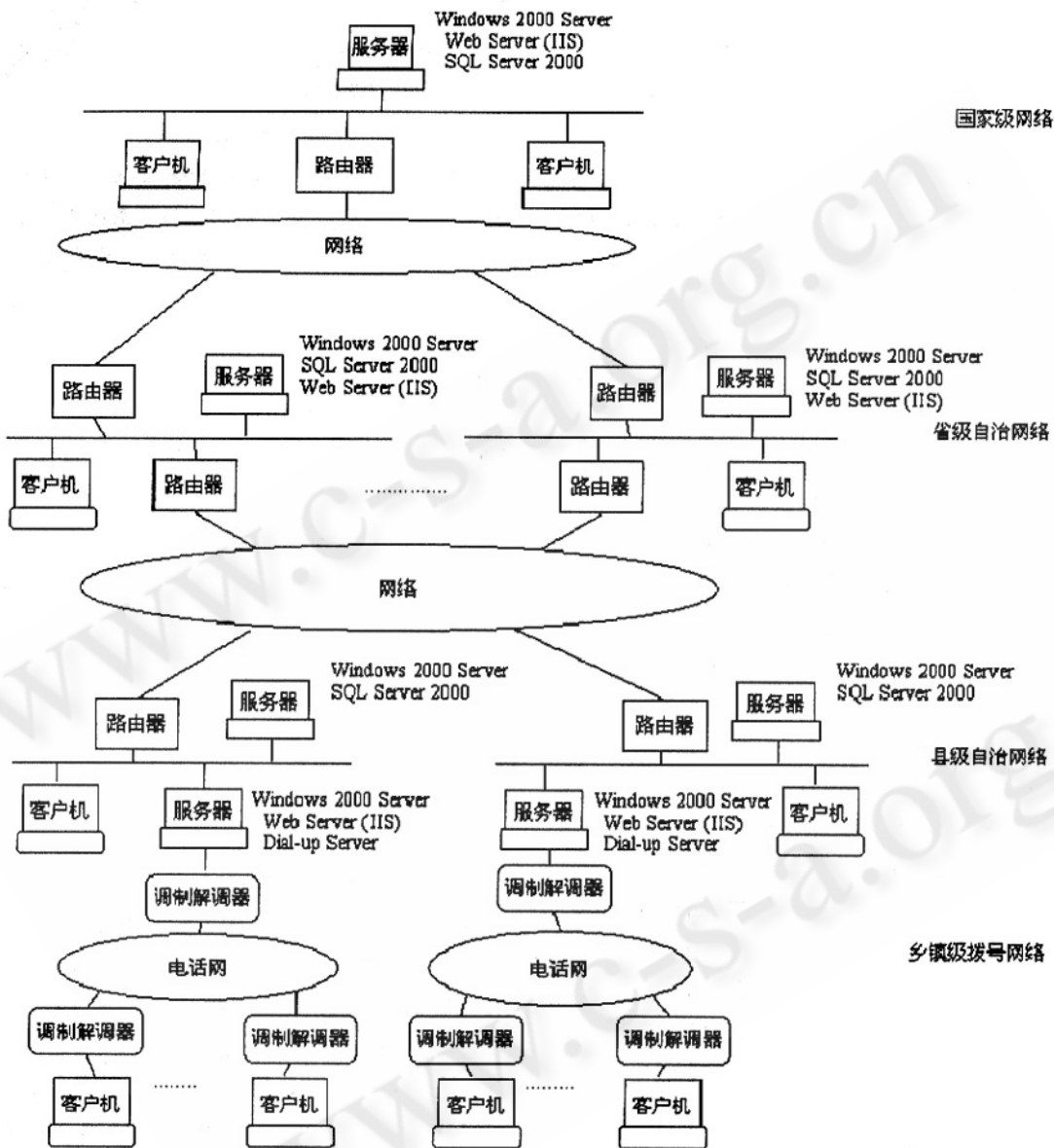


图 1

(2) 乡镇用户可以通过拨号到县里的拨号服务器来访问本乡镇的人口数据,从而可以完成对本乡镇人口数据信息的增加、更新、删除等处理任务。对外乡镇人口数据信息的查询采用创建视图的方式完成,对本省内各县人口数据的查询通过链接服务器技术实现。

数据查询,也可通过分布式技术提供外省的数据查询。

(4) 在国家级,仍然有一个局域网环境,并安装数据库服务器及 WEB 服务器,但在服务库服务器上不再设立物理的全国人口信息数据库,国家级数据库服务器的数据库中,只建立到全国各省数据库服务的链接

及访问视图,来实现对全国人口数据信息的分布式访问。

### 3 分布式数据库系统的体系结构

如图 2 所示,最终用户通过浏览器提出数据库访问请求到 WEB SERVER,WEB SERVER 通过 ADO.NET 技术建立同数据库的联接访问,如果是本地数据库,则查询结果返回用户端浏览器,完成一个交互,如果用户需要访问的数据不在本地,那么就要启动分布式查询。在 SQL SERVER 环境下可以有三种方式实现分布式查询,在本系统中采用了建立链接服务器的技术实现。如图 2 所示,通过建立链接服务器,本地服务器将查询提交到远程数据库服务器,由远程数据库服务器完成查询请求,并将查询结果返回本地数据库服务器,再由运行在本地服务器上的 ASP.NET 程序将结果返回到用户端,从而完成分布式查询任务。

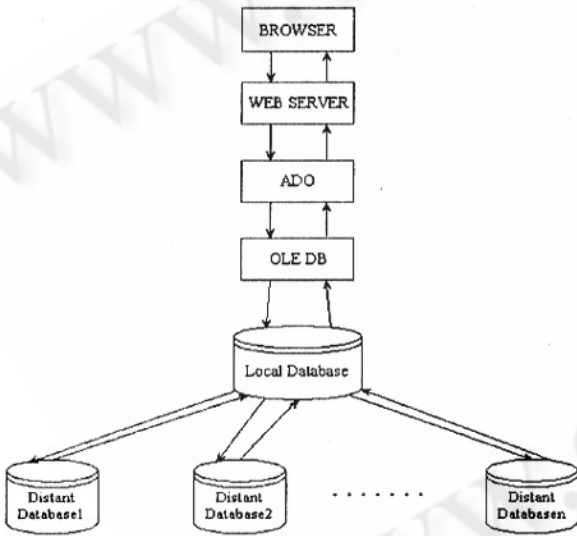


图 2

### 4 数据库分布式查询的实现及优化技术

在访问远程数据库之前,首先要建立访问链路即建立到远程数据库服务器的链接服务器,然后在 ASP.NET 页面脚本中通过 ADO.NET 技术完成对远程被链接的数据库服务器的查询访问。在这里,对远程链接服务器的定义可以通过系统来完成,也可通过 T-SQL 来

实现,但不管采用何种方式,对最终用户来说是透明的,即最终用户不知道也不必要知道自己所要查询的数据是在本地还是在网络的其它地方。

在本系统中,由于每个独立的分片数据库的数据量也较大,而且分片数据库数量较多,如果采用传统的串行查询,则等待时间会严重影响系统的反映时间,因此,为了提高系统的查询反映时间,对查询进行了优化,采取了同时对多个分片数据库并行查询的方式,即 WEB 服务器接到用户的请求后,可以启动多个线程,同时对多个相关的分片数据库进行并行访问。用 .NET 技术及 C# 语言实现查询及查询优化,主要的代码如下:

//窗口类中的建立线程方法

```
private void ThreadSetup( string sql)
```

```
{
    GetData getData1 = new GetData (); //新建数据库访问对象
```

```
    getData1.Sqlstring = sql; //给数据库对象赋值,即查询字符串
```

```
    Thread thread1 = new Thread ( new ThreadStart ( getData1. VisitingDataBase ) ); //新建线程对象
```

```
    thread1. Start (); //以线程方式启动数据库对象中的方法 VisitingDataBase,对数据库进行访问
```

```
}
//对数据库访问进行类的封装
```

```
public class GetData
```

```
{
    private string sqlstring;
    private SqlDataReader sdr;
```

```
//以下为设置类的属性
```

```
public string Sqlstring
```

```
{
    get
```

```
{
    return sqlstring;
```

```
}
```

```
set
```

```
{
    sqlstring = value;
```

```
}
```

```
}  
public SqlDataReader Sdr  
{  
    get  
    {  
        return sdr;  
    }  
}  
//以下为数据库访问方法  
public void VisitingDataBase( )  
{  
    SqlConnection con = new_  
    SqlConnection( " server = 10. 10. 11. 12 \SQLSERVER1 ;  
    database = BEIJING ; uid = sa ; pwd = hgp ; " ) ;  
    con. Open ( ) ; //上述两个语句用于建立同远程数  
    据库服务器实例 SQLSERVER1 的连接, 数据库名为 BEI-  
    JING, 用户名 sa, 为密码为 hgp, 在执行此语句时, 应  
    置好链接服务器  
    SqlCommand cmd1 = new SqlCommand( sqlstring ,  
    con ) ;  
    sdr = cmd1. ExecuteReader ( ) ;  
}
```

## 5 结论

本文在分析了人口数据信息特点的基础上, 提出了基于 B/S 体系结构的分布式人口管理信息系统的实现技术, 是对目前常见的基于 C/S 模式的分布式数据库应用系统的补充和完善。整个系统采用了基于微软的软件产品及数据库访问技术, 如 WINDOWS SERVER 2000, IIS, SQL SERVER 2000 产品, 以及 ASP. NET 和 ADO. NET 编程技术, 利用 C# 语言实现, 数据库的设计上采用了水平分片的模式, 在数据库查询优化技术上, 采用了 .NET 下的多线程并行访问模式, 解决了分布式数据库系统中串行访问的等待时间。

## 参考文献

- 1 苏贵洋, ASP. NET 网络编程从基础到实践, 电子工业出版社, 2005. 10。
- 2 顾兵, SQL SERVER 2000 网络数据库技术与应用, 华中科技大学出版社, 2006. 1。
- 3 邵佩英, 分布式数据库系统及其应用, 科学出版社, 2005. 3。
- 4 邹建峰等, C# 企业级开发案例精解, 人民邮电出版社, 2006. 1。