

文章编号:1000-582X(2006)02-0076-03

ASP.NET 在异构数据源转换中的应用*

陈静^{1,2}, 高川^{1,3}, 申建军¹

(重庆大学 1. 自动化学院, 2. 教务处, 重庆 400030; 3. 重庆邮电学院, 重庆 400065)

摘要:为了实现异构数据源的统一访问,针对异构数据源转换问题,借助 ASP.NET 中的数据库访问技术,提出了一种采用 ASP.NET 技术在互联网上进行异构数据源相互转换的通用模型和实现方法. 具有如下特点:属于 B/S 模式,与用户平台无关,数据可分布处理,对客户端要求较低;最后给出将非 DBMS 的数据源转换为 SQL 数据源的实例,说明该方法是可行和有效的.

关键词:统一数据访问;ASP.NET;DBMS

中图分类号:TP311

文献标识码:A

目前数以亿计的个人计算机和商业计算机里存在着以不同的格式存储的数据和文档,如果这些数据能够让其拥有者以一种方便和规范的方式提交到在 INTERNET 上的数据库服务器里给人们共享,实现异构数据源的统一访问^[1],这无疑对信息的检索和交流有着及其重要的意义,文中利用 ASP.NET 技术在这方面进行了一些探索,并取得了一定的实际应用效果,实例说明该方法是可行和有效的.

1 ASP.NET 实现异构数据源转换的原理

ASP.NET 是一种建立在通用语言上的程序构架,可用于在 Web 服务器上建立强大的 Web 应用程序. ASP.NET 中的 ADO.NET 提供了强大的 Web 开发模式和更强的数据库访问能力. 在 ASP.NET 提供的所有 NameSpace 中,数据库与数据设计及操作有关的有: System. Data, System. Data. SqlClient 和 system. Data. OleDb^[2],根据 ASP.NET 的数据库操作原理和特点,设计了如图 1 所示的数据库访问模式.

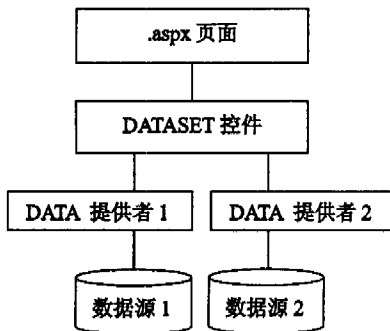


图 1 异构数据源转换原理图

由于 DATASET 控件是一个与具体物理数据库无关的内存数据库,对数据的任何操作均与数据库类型无关系,只须数据提供将不同数据库或数据源的物理数据提取到 DATASET 中,再利用数据提供者 2 就可将数据转换成数据源 2 的数据格式,因此可以非常方便地实现异构数据源的转换、存储与管理,并且在实际应用中顺利实现转换^[3].

2 异构数据源转换实现

对于异构数据源之间的转换,传统的方法受限较多. 而上述方案由于在 ASP.NET 编程中采用了 ADO.NET 技术,实现了在 Web 上更为广泛的应用,超出了数据库的范围局限和地域局限. 本方案具有以下特点: 1) 属于 B/S 模式; 2) 与用户平台无关; 3) 数据可分布处理; 4) 对客户端要求较低.

文中以 Microsoft Excel 数据转换为 SQL 数据表为例来说明实现异构数据源转换的方法和一般过程,图 2 是待转换的 Microsoft Excel 数据表. 由于要实现 2 种数据源之间的转换,所以实现方案为 2 个部分. 第 1 部分读出源数据源里的数据;在读源数据时,用户先将源数据文件提交到 Web 服务器,然后利用 ASP.NET 技术进行读出处理;第 2 部分利用 ASP.NET 技术将读出的数据写入到目的数据源. 具体实现步骤如下:

2.1 源数据访问实现

为用户提供交互式 Web 界面;让用户选择待转换的数据源类型,并输入转换后的数据要存储的数据表名,用户提交的数据源文件通过计算机网络(或 Internet)传送到服务器(图 3 所示).

* 收稿日期:2005-09-13

作者简介:陈静(1972-),女,四川中江人,重庆大学硕士研究生,主要从事计算机网络、自动控制研究.

姓名	性别	年龄	体重	身高
张立文	男	28	65	172
李伟	男	28	74	177
王小明	男	28	66	176
刘志强	男	28	84	177
陈强	男	28	85	178
周伟	男	28	84	177
吴强	男	28	74	176
陈强	男	28	85	177
周伟	男	28	75	176
吴强	男	28	65	172
刘志强	男	28	60	170
陈强	男	28	85	177
周伟	男	28	77	172
吴强	男	28	69	172
陈强	男	28	84	173
周伟	男	28	71	172
吴强	男	28	92	178
陈强	男	28	13	173
周伟	男	28	89	172
吴强	男	28	61	171
陈强	男	28	47	161
周伟	男	28	82	170

图 2 待转换的 Microsoft Excel 数据表

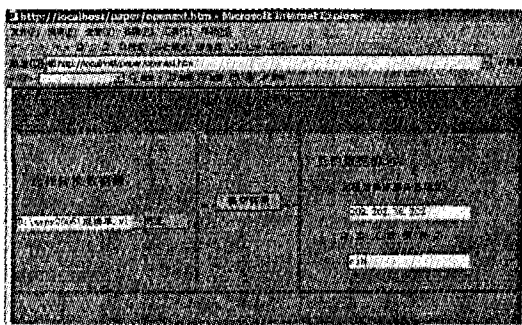


图 3 交互式 Web 界面

2.2 服务器端程序编写

2.2.1 上传数据

编写服务器端程序,实现用户数据源上传到 Web 服务器^[4]。代码如下:

```

<% If ( Request. Files. Counts > 0 ) Then
Dim FileColl As System. Web. HttpFileCollection = Request. Files
Dim PostedFile As HttpPostedFile = FileColl. Get(0)
Dim FileName As String = PostedFile. FileName
Dim ObjDir As System. IO. Directory = New System. IO. Directory("c:\upload")
MyPostedFile. SaveAs ( ObjDir. FullName + "\ " + FileName)
End If

```

2.2.2 利用 ADO.NET 技术读取数据

采用 ADO.NET Managed Provider,该方式可以连接到任何 ODBC(Open Database Connectivity 开放式数据库连接)或 OLEDB(Object Linking and Embedding Database 对象连接和嵌入式数据库)的数据中心,以下是创建读取数据源的 VB 代码例程。

①创建数据链路^[5]

```

<%@ Page Language = "VB" %>
<%@ Import Namespace = "System. Data" %>
<%@ Import Namespace = "System. Data. ADO" %>
<Script Language = "VB" RunAt = "Server" >
VB 语言描述
Sub Page_Load

```

```

Dim ExCnStr As String = " DBQ = " & fileName & "; DefaultDir = " & sDir & _
"; Driver = { Driver do Microsoft Excel( *. xls ) }; DriverId = 790"

```

fileName 是用户输入的待转换的数据源,sDir 是待转换数据源的路径

创建连接对象

```

Dim ndbConn As New ADOConnection( ExCnStr)
ndbConn. Open(); 打开数据链路
.....

```

End Sub

②创建数据集^[6]

‘使用 ADODataSetCommand 将数据写入到 DataSet(数据集)中。

```

sqlstr = "SELECT * FROM " & sDriver & ": \ \" & Right(sDir, locate - 3) & sFile & ". Sheet1 $"
MyCommand = New ADODataSetCommand(sqlstr, ndbConn)

```

```

MyDataset = New DataSet
MyCommand. FillDataSet( MyDataset, "Excel")

```

③创建数据表

```

MyTable = New DataTable
MyTable = MyDataset. Tables("Excel")

```

到此,Excel 数据表中的数据已经读取,并放在内存 MyTable 对象中,于是人们就可以根据要求比较自由地处理这些数据,文中作者把这些数据写入到一个指定的 DBMS 数据库中。

2.3 将读出的数据写入到目的数据源

以 SQL Server 数据库为例,要写入其它数据源,只须改变相应的连接字符串即可。

2.3.1 创建数据链路

下面代码创建一个到目的数据源(SQL Server 数据库)的连接^[7]

```

目的数据源(SQL Server 数据库)的连接字符串
Dim dbcnStr As String = " Provider = SQLOLEDB; Data Sourced = 202. 202. 38. 202; " & _ "& " Integrated Security = SSPI; Initial Catalog = pubs"

```

创建 OleDbConnection 连接对象

```

Dim dbConn As New ADOConnection(dbcnStr)
打开数据链路
dbConn. Open();

```

2.3.2 创建数据表

由于目的数据源中还没有相应的数据表,所以必须在指定的数据库中创建一个可存储源数据的数据表,该数据表的结构按照源数据的标题建立。

‘根据源数据建立目的(SQL Server)数据表

```

Dim dbcol As DataColumn = New DataColumn
strSQL = "create table &" dbfName & "(" & dbfName;
用户希望建立的 DBMS 数据表

```

```

for i = 0 to MyTable. Columns. Count - 2

```

```

strSQL = strSQL & MyTable . dbcol (i). column-

```

```
Name & " varchar ( " &MyTable. ndbcol ( i ).
ActualSize&" ), "
next
strSQL = strSQL&MyTable. dbcol ( i ). columnName&"
varchar("&MyTable(i). dbcol(i). ActualSize&" )"
MyCommand = New ADODatasetCommand( strSQL, db-
Conn)
```

2.3.3 将读出的数据写入到数据表

将数据写入到新建的目的(SQL Server)数据表

```
numrows = MyTable. Rows. Count
If numrows = 0 then
    Response. Write( " <p>无数据. </p>" )
Else
    For loop1 = 0 To numrows - 1
        strSQL = "INSERT INTO " &dbfName&" ( "
            for i = 0 to MyTable. Columns. Count - 2
                strSQL = strSQL&MyTable. dbcol
(i). columnName &" , "
                next i
                strSQL = strSQL&MyTable. dbcol ( i ).
columnName&" ) VALUES ( "
                for i = 0 to MyTable. Columns. Count - 2
                    strSQL = strSQL&MyTable. Rows
(loop1). Column(i) &" ; "
                next i
                strSQL = strSQL&MyTable. Rows ( loop1 ).
Column(i) &" )"
            MyCommand = New OleDbDataAdapter ( strSQL, db-
Conn)
            Next loop1
            Response. Write( " <p>祝贺你! 数据转换成功.
</p>" )
    End if
```

运行上述程序后,用 SQL Server Manager 打开新创建的 DBMS 数据表 cjb(见图 4),可以看到转换后的 SQL 数据表内容和 Microsoft Excel 数据表一致。

id	name	sex	age	address	phone
1	张三	男	25	重庆	13800000000
2	李四	女	22	成都	13800000000
3	王五	男	28	北京	13800000000
4	赵六	女	20	上海	13800000000
5	孙七	男	23	天津	13800000000
6	周八	女	21	武汉	13800000000
7	吴九	男	26	广州	13800000000
8	郑十	女	19	深圳	13800000000
9	陈十一	男	24	西安	13800000000
10	冯十二	女	27	南京	13800000000

图 4 转换成功后的 SQL 数据表

3 结束语

笔者以 ASP.NET 的常用数据访问组件 ADO.NET 实现了异构数据源间的转换,具有方便易行,实现简单,不受地域限制,对用户机器要求不高等特点,具有较高的应用价值和实际意义。

参考文献:

- [1] 寇继虹. 统一数据访问技术及应用[J]. 武汉大学学报(工学版), 2002, (12): 104-107.
- [2] 廖信彦. ASP.NET 技术参考[M]. 北京:中国铁道出版社, 2001.
- [3] NET Framework Developers Guide[DL/OL]. <http://msdn.microsoft.com/library/>, 2003-01-11.
- [4] 刘立群,刘宏. 基于 ASP.Net 实现文件上传[J]. 现代计算机, 2003, (2): 54-56.
- [5] 孙冬卫,张冀红,车声别. ASP.NET 的数据库访问技术研究[J]. 现代电子技术, 2002, (7): 36.
- [6] 杨兴凯,刘宏. 基于 Web Services 跨平台存取数据库[J]. 电脑学习, 2003, (12): 27-28.
- [7] [美] ANDREW DUTHIE G. Microsoft ASP.NET 程序设计[M]. 北京:清华大学出版社, 2002.

Application of Transformation Among Heterogeneous Data Sources Based on the ASP.NET Technology

CHEN Jing^{1,2}, GAO Chuan^{1,3}, SHEN Jian-jun¹

(1. College of Automation;

2. Educational Administration Department, Chongqing University, Chongqing 400030, China;

3. Chongqing University of Post and Telecommunication, Chongqing 400065, China)

Abstract: The paper analyzes and utilizes the ASP.NET technology and puts forward a kind of model and method based on the ASP.NET technology, which realizes the transformation among heterogeneous data sources. The scheme has the following characteristics: B/S mode, being independent of and user's platform, data distributed processing. Finally, it takes an example that converts non-DBMS data source into SQL data source which proves the method is feasible and effective.

Key words: UDA; ASP.NET; DBMS