

# The strategy For Calling Remote SQL Server DataBase

胡吉平

(济南军区司令部工程设计研究院 250002)

张淑华

(济南 山东经济学院 250014)

## 远程 SQL SERVER 数据库的访问策略

**摘要:** 本文以 VB 应用程序为例讨论了访问远程 SQL SERVER 数据库的几种流行方法, 分别论述了每种方法的内部机理, 并比较了每种方法的优点及缺点。

**关键词:** VB 远程数据库对象模型  
ActiveX OLEDB RDC RDO  
ADC ADO ODBC

### 1 前言

SQL SERVER 是微软公司推出的一套功能非常强大的 C/S (客户/服务器) 数据库系统, 其应用领域非常广泛。本文主要以 Visual Basic (简称 VB) 为例讨论访问远程 SQL SERVER 数据库的方法。VB 访问本地数据库的方法很多, 诸如用 Jet 和 Data 数据控件访问数据库, 用数据访问对象 (DAO) 访问本地数据库, 但随着计算机技术的发展, 基于单机的信息系统已经不能适合迅速变化的市场需要, 这就要求企业要有完整的信息系统, 而计算机网络技术的迅猛发展为解决这个问题提供了可能, 因而, 利用数据服务器在企业内部共享数据便应运而生了。

当今, 设计人员可以选择多种方式访问远

程 SQL SERVER 数据库, 有些技术已经淘汰, 本文着重讨论用远程数据控件 RDC (Remote Data Control)、远程数据对象 RDO (Remote Data Object)、ADO (ActiveX Data Object) 数据控件即 ADC (Advance Data Control)、ActiveX 数据对象即 ADO 四种方法访问 SQL SERVER 远程数据库。

### 2 实现方法

#### 2.1 用远程数据控件 RDC 访问远程 SQL SERVER 数据库

利用远程数据控件, 你可以很方便地创建一个基于 SQL SERVER 数据的表, 用 RDC 创建表是可视化设计的最终目标——不需任何代码, 用 RDC 创建表包括以下步骤:

(1) 为数据库创建一个 ODBC (Open DataBase Connectivity) 数据源。

(2) 向窗体中添加远程数据控件 (即 Microsoft Remote Control 6.0), 并设置 DataSourceName 和 SQL 属性。

(3) 对需要显示的项在窗体中添加控件, 设置 DataSource 和 DataField 的属性。

将远程数据控件绑定到 ODBC 的用户 DSN 或系统 DSN 上, 该控件打开与 SQL SERVER 的连接, 提交查询。②中的 SQL 属性, 我们以 SQL SERVER 自带的 "Pubs" 数据库为例, 比如要查

询 "Pubs" 数据库中的 "Publishers" 表中的字段, 可如下定义:

```
SELECT pub-id, pub-name, city, state, country  
FROM publishers
```

然后将查询的结果以你自己设计的输出或显示方式表示出来, 再利用控件提供的导航按钮在查询返回的行中浏览, 实现数据库的遍历比较简单, 如果要对数据库中的记录进行修改, 则要用到远程数据控件的结果集 Resultset 对象进行编程, Resultset 对象提供了 AddNew 和 Delete 方法向数据库的表中添加及删除行, 如 rdc.resultset.addnew 和 rdc.resultset.delete, 所要注意的是执行删除行的操作以后, 当前行变成未定义, 需要添加代码移向有效行。

远程数据控件是一个非常方便使用的控件, 如果不向表中添加或删除记录, 其基本不用编写代码, 但它对远程数据库的操作非常有限。

#### 2.2 用远程数据对象 RDO 访问远程 SQL SERVER 数据库

##### 2.2.1 关于 RDO

RDO 是 VB 提供的重要工具之一, 其对象集合框架如图 1 所示, RDO 在 ODBC API (Apply Program Interface) 上为我们提供了一个应用层, 包括一系列对象和集合, 提供了使用代码创建并控制远程 ODBC 数据系统部件的框架, 对象和集合的属性描述了数据库的特性, 也描述了用来操

纵它们的方法。

## 2.2.2 使用远程数据对象 RDO 访问远程 SQL SERVER 数据库步骤

首先在窗体中添加 Microsoft Remote Data Object 2.0 引用, 如果一个 ODBC 数据源已经设置好了, 还以 Pubs 数据库为例, 假设其 DSN 名称为 "pubs-example", 那么连接远程数据库的代码如下:

```
dim mcon as RDO.rdoconnection
set mcon=rdoEnvironments (0) .openConnection
("Pubs-example")
```

如果没有配置 ODBC 的 DSN, 则要建立所谓 "DSN-less" 连接, 其步骤如下:

由用户动态创建 DSN 以访问远程数据库, 这比用 ODBC 建立 DSN 省略了一步。与数据库的连接完成之后, 生成结果集, 其基本代码如下:

```
dim mrsPublishers as rdoResultset
dim sSQL as string
sSQL = " SELECT pub-id, pub-name, city, state,
country FROM publishers "
set mrsPublishers=mcon.OpenRusultset (sSQL,
rdOpenKeyset, rdConcurValues)
```

同样可以利用这个结果集对象的 AddNew 和 Delete 方法来增加、更新和删除记录, 限于篇幅, 此处不再赘述。

## 2.3 ADO 数据控件访问远程 SQL SERVER 数据库

控件来处理, 若要对记录源进行添加、修改、删除等操作, 数据控件就无能为力了, 而需要使用 ADO 数据控件的 RecordSet 属性的各种方法, 如 adodc.RecordSet.AddNew 是向缓冲区中创建一个新的空行, 而 adodc.RecordSet.Update 是把这一新行保存到数据库中, adodc.RecordSet.Delete 来删除一条记录, 删除完之后要把数据控件移到一个新的有效行上。

## 2.4 用 ActiveX 数据对象访问远程 SQL SERVER 数据库

### 2.4.1 关于 ADO 数据对象

通用数据访问 UDA (Universal Data Access) 是 Microsoft 想使开发人员能够用一种数据访问方法去访问他正在查询的任何数据源的一种方法。其实, ODBC 是向这个目标迈出的一大步, 在 ODBC 中, 应用程序只需要 API 就能访问任何数据库, 但 ODBC 的问题是它直接针对关系数据库, 其他数据源并不十分适合它的模型, 为解决这个问题, Microsoft 创建了对象连接嵌入数据接口 OLE DB (Object Linking Embedding Database), OLE DB 是一组 C++ API 函数, 就象 ODBC

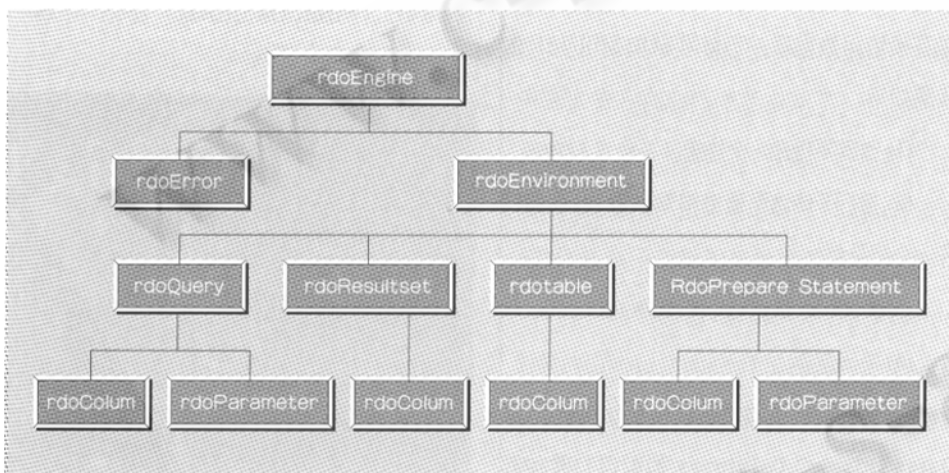


图 1 RDO 对象集合框架

(1) 在窗体中添加 Microsoft Remote Data Object 2.0 引用,

(2) 编写访问远程数据库代码:

```
dim mcon as rdoconnection
dim sConnect=" UID=sa; PWD=mypassword;
DATABASE=pubs; SERVER=myserver; " &
DRIVER={SQL SERVER}; DSN=''; "
set mcon=rdoEnvironments (0) .openConnection
(" ", rdDriverNoPrompt, False, sConnect,
rdAsyncEnable)
```

上述编码中, DSN 指定为空, 程序运行之后,

ADO 数据控件访问远程 SQL SERVER 数据库步骤为:

(1) 先加载组件 Microsoft ADO Data Control 6.0, 然后在窗体上画 ADO 数据控件, 再设置其 ConnectingString (包括 DSN, User ID 和 Password) 和 RecordSource 属性。

(2) 在窗体中添加几个控件并设置它们的 DataSource 和 DataField 属性以显示记录源中的字段。

只需简单的两步, 不要一行代码, 就可以连接并查询数据库, 整个数据访问全由 ADO 数据



## The strategy of Accessing the Remote SQL Server DataBase

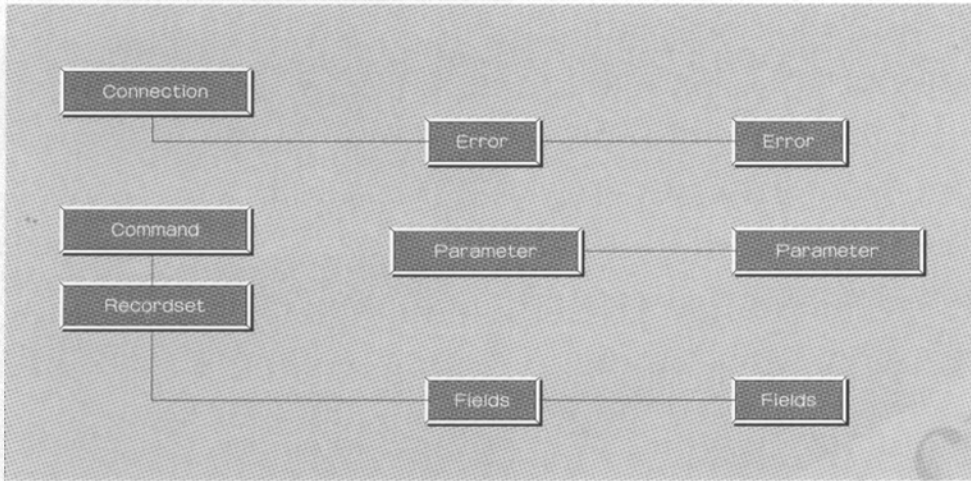


图 2 ADO 对象模型

标准中的 ODBC API 一样,不同的是 OLE DB 的 API 是符合 COM (Component Object Model) 标准,基于对象的 API, ODBC API 则是简单的 C API。这样就不用去试图向 ODBC 中添加功能以便使它能象处理关系数据库一样去处理其他的数据源了。OLE DB 的数据提供者可以是关系数据库、电子表格、文件系统或任何其他数据源, Microsoft 在 OLE DB 上创建了易于使用的 ActiveX Data Object (ADO), 它是进入 OLE DB 的接口, 借助 ADO, VB 就能利用 OLE DB 的优点。ADO 对象模型如图 2 所示, 从模型图中可以看出, ADO 与 RDO 模型的层次结构不同 (RDO 有一个 rdoRecordset 对象, rdoRecordset 是 rdoConnection 子对象, rdoConnection 是 rdoEnvironment 的子对象, rdoEnvironment 是 rdoEngine 的子对象), ADO 基本上是一种平板结构, 从一个数据源中获取记录只需要一个连接和一个记录集, Command 和 RecordSet 与 Connection 三者之间没有上、下层的联系, 这种设计主要是为了适应 Internet 应用开发的需要, 因为在 Internet 上, 象在局域网内那样维护一个永久性的连接, 然后在连接的基础上执行查询基本上是不可能的。

### 2.4.2 用 ActiveX 数据对象访问远程 SQL SERVER 数据库步骤

ADO 的 Connection 和 RecordSet 对象可以直接访问数据库, 它访问 SQL SERVER 的步骤如下:

- (1) 创建并打开一个连接。
- (2) 把打开的这个连接作为参数, 创建并打开一个记录集。
- (3) 关闭记录集。
- (4) 关闭连接。

首先在窗体中添加 Microsoft ActiveX Data Object 2.0 Library 的引用, 并加入 Microsoft Windows Common Control 6.0 组件, 一般在窗体加载事件中打开 Connection 对象和 RecordSet 对象, 如下:

```
Private Sub Form-load()
Dim adocn as ADO.DB.Connection
Dimadors as ADO.DB.RecordSet
Set adocn = New ADO.DB.Connection
```

以下三种方法连接到数据库或数据源:

方法一: 直接用连接字符串连接到数据库 pubs:

```
adocn.open "Provider=SQLOLEDB.1;PersistSecurity Info=False;User ID=sa;" &
```

```
_"+ " Initial Catalog=pubs;Data Source=myserver"
```

方法二: 如果有一个 DSN 数据源, 则用连接字符串连接到 ODBC 的 DSN 数据源 pubs-example:

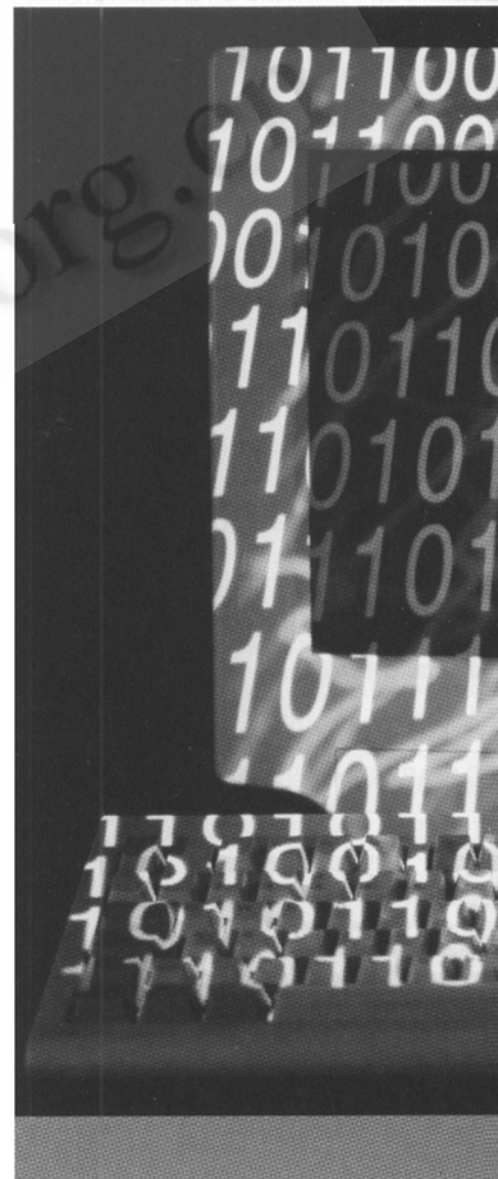
```
'adocn.Open "Provider=MSDASQL.1;Persist Security Info=False;" &
```

```
_"+ " User ID=sa;Data Source=pubs-example"
```

方法三: 如果没有 DSN 数据源, 就象 RDO 一样, 我们同样可以在数据环境窗口当中创建一个 DNS-less 的连接字符串:

```
'adocn.open "Provider=MSDASQL.1;Persist Security Info=False; User ID=sa;" &
```

```
_"+ " ExtendedProperties=' DRIVER=SQL Serve; SERVER=WINBOOK;UID=sa;" &
```



```

_+ 'APP=Visual Basic;WSID=WINBOOK;
DATABASE=pubs';Initial Catalog=pubs' Set
adors=New ADO.DB.RecordSetadors.open
'Publishers'.cn, adOpenForwardOnly,
adLockReadOnly

```

……(此处是对数据集进行操作)

```

adors.close
adocn.close
set adors=nothing
set adocn=nothing
End Sub

```

上述实际上是用三种方式连接到数据库。如图3所示，一种是用连接字符串的方式，即 adocn.open …… 另一种方法是用ODBC的

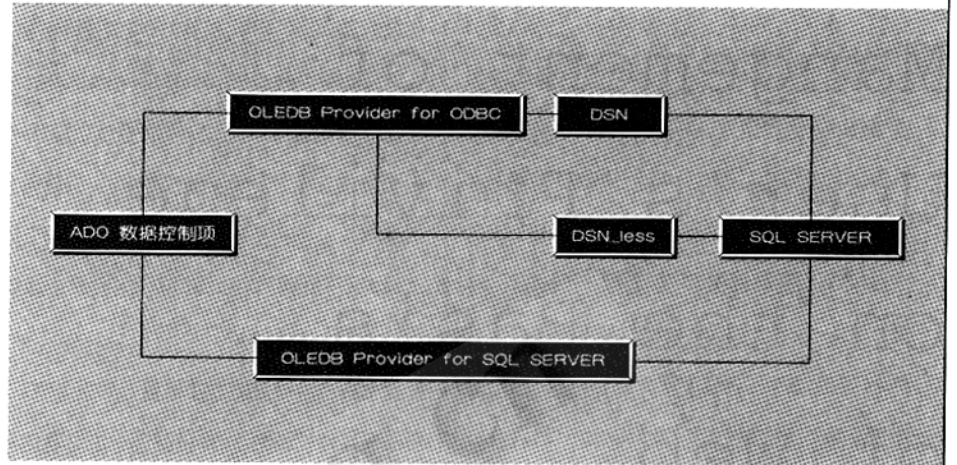


图3 ADO访问SQL SERVER的三种方式

“DSN数据库源进行连接，第三种方法就是用 DSN-less 方法，即不创建 DSN，而由程序自动创建。通过 Provider 部分我们已经明显看到由 SQLOLEDB.1 变成了 MSDASQL.1，这是有着本质上的区别。然后只用了一个向前的只读游标来显示数据，如果要对记录集进行修改，则要应用优化锁定的键集 (Keyset) 游标。ADO 提供了两种改变数据的基本方法，即 Connection 对象的 Execute 方法和 Recordset 对象的 Update、Addnew 和 Delete 方法，Execute 方法还可以使用 SQL 查询和诸如 UPDATE、INSERT、DELETE 这样的命令。限于篇幅，此处不再赘述。

源，而 OLE DB 的对象则是范围更为广泛的任何数据存储。从这个意义上说，ADO 涉及的范围更广。所以，具体使用哪种方法访问远程数据库，在很大程度上依赖于用户的应用程序的具体情况，依笔者之见，如果你感觉方便，还是尽早转移到 ADO。

参考文献

- 1 顾斌、杨德斌译，Visual Basic6.0数据库开发，清华大学出版社，1999。
- 2 熊贵喜、蔡铁岭等译，SQL Server7.0开发指南，清华大学出版社，2000。

5 结束语

上述四种访问远程 SQL SERVER 数据库的方法各有其特点，RDO 方法是基于对象的，因而便于使用，由于它是通过 ODBC API 来访问数据库，所以其通用性好，允许最强交互操作性，编程简单，但速度慢于 ADO，而 ADO 具有真正的文件驱动及错误处理能力，由于 ADO 是通过 OLE DB 数据接口完全支持异步处理、游标和计算列等，ADO 的编程比 RDO 更加简单，更加方便。由于 ODBC 标准的对象是基于 SQL 的数据

