



### 相关主题

RECOMMEND ARTICLE

- 简单介绍Visual Basic中的Do循环结构
- Visual Basic Print方法基础学习
- VB.net常用函数
- 关于Visual Basic中的For循环语句
- Visual Basic中实现带预览的对话框
- 关于Visual Basic 9.0的动态标识符
- VB中运用反射原理优化程序代码
- 利用VB6.0实现五线谱作曲工具

MORE

### 推荐文章

RECOMMEND ARTICLE

- 游戏音乐制作案例之《战火 红色警戒》音效制作揭秘
- 英雄连Online 原画
- 游戏音乐制作案例之《乱武天下》
- 游戏音乐制作案例之《诛仙》
- 《鹿鼎记》最新原画
- MDP2.1规范的新特性
- 3D游戏编程入门经典(6)
- Introduction to 3d game engine design using directx-9 and c#(10)

MORE

### 热门文章

HOT ARTICLE

- [电子书下载]游戏设计 — 原理与实践
- [电子书下载]网络游戏开发
- 游戏设计全过程
- [电子书下载]游戏设计技术
- [电子书下载]游戏设计理论
- CS游戏人物模型制作教程
- CG人物插画基本流程
- [转贴]MAX高级人头教程

MORE

您的位置: VB专区



文章标题	ADO数据访问模型初学者入门		
来源:	[ ogdev ]	浏览:	[612]

ADO (ActiveData Objects) 是Microsoft提供和建议使用的新型的数据访问接口, 具体实现为Automation。这样, 程序员可以在各种支持Automation的开发环境下方便地访问ADO对象, 如Visual Basic、Visual C++、Visual J++及Delphi等。ADO被实现为OLEDB之上的一个薄层, 这使得ADO可以有更快的访问速度, 更易使用, 同时更节省资源。值得注意的是, 对于Visual Studio 6.0的用户而言, 现在提供的ADO2.0要比Visual Studio 5.0时代的ADO1.5更加完整。这意味着程序员可以更为广泛地使用ADO接口, 甚至在所有的基于Windows平台的数据访问中使用它。比如对RDS (前身是ADC) 的支持, 可以方便的构建高效的Web应用。

#### 一、对ADO对象的主要操作

对ADO对象的主要操作, 同DAO、RDO库的实现基本相同。主要包括6个方面:

1. 连接到数据源。这是可选的、通常涉及ADO的Connection对象。
2. 向数据源提交命令。通常涉及ADO的Command对象。在查询中可以与参数对象 (Parameter) 协同使用。
3. 执行命令, 比如一个SELECT脚本。
4. 如果提交的命令有结果返回, 可以通过ADO的Recordset对象对结果进行操作, 数据存储在缓存中。
5. 如果合适, 可将缓存中被修改的数据更新到物理的存储上。
6. 提供错误检测。通常涉及ADO的Error对象。

从程序员的视角来看, ADO、DAO和RDO三者的对象名称不很相同。但使用ADO对象要比DAO和RDO简单得多。最主要的一点在于, 程序员不用像在使用DAO和RDO那样要从对象模型的顶层开始一步步的创建子对象。因此, ADO提供了一种更灵活的编程方式。

#### 二、ADO中主要对象的功能

Connection对象, 表示了一个到数据源的会话。使用Connection对象的成员, 可以使用相应的属性打开到数据源的连接, 设置游标的位置, 设置默认的当前数据库, 设置将使用的OLEDBProvider, 直接提交SQL脚本等。值得注意的是, 在提交SQL脚本的任务时, 不用创建一个Command对象, 就可完成查询。另外, 对Connection对象的创建是同其他对象无关的。Command对象, 可用于查询数据库并返回结果在Recordset对象中。也可以进行批操作和操纵数据库的结构, 当然, 这需要使用的OLEDBProvider提供相应的支持。此外, 可以将一个激活的Connection对象绑定到Command对象的ActiveConnection属性, 这使得多个Command对象实例可以共用一个Connection对象。

Recordset对象, 用来封装查询的结果, 可称为结果集。

Field对象, 用来表达一行结果中各子段的类型和值。

Error对象, 用来检测和判断在数据库操作中出现的错误, 比如连接失败。在ADO中, 许多对象名后多了一个"s", 比如Error->Errors, Field->Fields等等。添加"s"意味着是相应对象的Collection (集合) 对象, 比如Errors是Error对象的Collection对象。Collection有点像数组 (Array), 但不同的是, Collection可以以不同类型的数据或对象作为自己的元素, 而数组中的各元素通常都是相同类型的。所以, 在看到一个对象名最后是"s", 通常表明这是一个Collection对象, 比如Errors中的各元素是由Error对象的实例组成的。

#### 三、简单的例子

现在, 我们在经过了前面的铺垫后, 终于可以进入写代码的过程了。下面是一系列在Visual Basic中编写的代码片段, 其中对ADO的操纵完全用代码来实现。在开始前, 请不熟悉ActiveXAutomation的读者牢记, 我们正在使用的, 是被实现为ActiveXAutomation的ADO组件。1. 打开Connection对象

打开一个到数据源的连接, 即Connection对象的VB代码如下:

```
Dim cn As ADODB.Connection
' 声明ADODB.Connection对象变量
Dim strCN As String
' 声明存放连接串的字符串变量
Set cn= New Connection
' 实例化Connection对象
' 生成连接串 (ConnectionString)
strCN="Provider=Microsoft.Jet.Oledb.3.51;
UserID=Admin; "&_
"DataSource=D:\Microsoft\Visual
Studio\VB98\Nwind.mdb;"
cn.Open strCN ' 调用Connection对象的方法Open连接数据源
```

以上代码在访问ADO对象时, 使用了前绑定 (Early-bind), 实现前绑定需要从Visual Basic的Project菜单下选择

Reference菜单项, 并选择MicrosoftActiveXDataObjects2.0Library。在程序中声明的strCN变量中, 连接串属性Provider标识了OLEDBProvider为OLEDBProviderforMicrosoftJet, 因为我们访问的是MicrosoftAccess数据文件, 使用MicrosoftJetEngine可以获得比ODBC更好的性能。在试验以上代码时有两个地方要注意。首先, 要根据系统安装的OLEDBProviderforMicrosoftJet服务选择相应版本, 可能是3.51, 也可能是4.0。在本例中使用的是3.51版本。如果使用4.0版本, 首先需要将Provider属性改为"Microsoft.Jet.Oledb.4.0"; 其次, DataSource属性标识了所要访问的数据文件的路径, 要根据自己的安装情况作出适当的调整。文件Nwind.mdb通常被VisualStudio默认安装, 一般放在VisualBasic的工作目录下。

除了OLEDBProviderforMicrosoftJet以外, VisualStudio6.0还提供了以下的OLEDBProvider, 如下所示。

OLEDB Provider类型 数据源类型

Microsoft OLEDB Provider for ODBC databases  
Microsoft OLEDB Provider for Microsoft Index Server Microsoft(r) Index Server  
Microsoft OLEDB Provider for Microsoft Active Directory Service Microsoft(r) Active Directory Service  
Microsoft OLEDB Provider for SQLServer Microsoft(r) SQLServer Microsoft OLEDB Provider for Oracle  
Oracle databases

这意味着使用ADO接口可以方便地访问以上的各种数据源, 要做的就是选择适当的OLEDB Provider。除了使用Connection对象的Open方法以外, 还可通过Recordset对象的Open方法快捷地创建到数据源的连接。这充分体现了ADO的灵活性。

## 2. 创建Command对象

```
Dim cmd As ADODB.Command
```

```
Set cmd=New Command' 实例化Command对象
```

## 3. 执行查询

以Command对象为例。

```
Dim rs As New ADODB.Recordset
```

```
Set cmd.ActiveConnection=cn' 绑定激活的Connection对象实例
```

```
cmd.CommandText="SELECT*from Customers"' 生成SQL脚本
```

```
Set rs=cmd.Execute' 执行查询
```

上述的代码仅仅是一种查询途径, 此外, ADO的Connection对象的Execute方法和Recordset对象的Open方法也提供了查询能力。返回的结果可以被保存在一个Recordset对象实例中以便后续的数据处理和操纵。例如:

```
Dim rs As New ADODB.Recordset
```

```
rs.Open cmd, cn, adOpenDynamic, adLockBatchOptimistic
```

另外在SQL脚本的生成方式上, 通常可以借助Parameters/Parameter对象来完成。

## 4. 显示和操纵数据

查询结果由Recordset对象封装。对数据的操纵可以通过Recordset对象提供的成员(属性和方法)来完成。

```
rs.MoveFirst
```

```
Do While Not rs.EOF' 判断EOF标记属性(EndOfFile)
```

```
Debug.Print rs!CustomerID & vbTab & rs!CompanyName & _
```

```
vbTab & rs!ContactName & vbTab & rs!ContactTitle
```

```
rs.MoveNext' 将游标指针移到下一条记录
```

```
Loop
```

上述代码将Recordset中的各行记录打印在VisualBasic的Immediate Window中。为了看到打印的结果, 可以在上述代码之后, 增加一条Stop语句, 以便进入Debug状态。

## 5. 更新记录

使用Recordset对象来完成Update操作。

```
rs.Close' 关闭之前建立的结果集
```

```
' 打开新的结果集, 具有写操作权限
```

```
rs.Open"Customers", cn, adOpenDynamic, adLockOptimistic, adCmdTable
```

```
rs.MoveFirst' 虽无必要, 但可能是一个良好的习惯
```

```
rs!CompanyName="Microsoft"' 对相应字段赋予新值
```

```
rs.Update' 在物理存储上生效
```

除了Update以外, 写操作还包括AddNew(添加一条新记录)和Delete(删除一条新记录)。另外, Recordset支持批处理, 但这些特征需要由低层的OLEDBProvider支持。当写操作出现错误时, 可根据错误描述(访问ADO的Error或VisualBasic的Err对象)来判断原因。另一个需要注意的是, 当使用Recordset对象完成写操作时, 需要预先指定Recordset对象实例非只读。

## 6. 收尾工作

在这个阶段应该显式的释放相应的资源, 如果不做的话, 通常VisualBasic会自动释放和回收资源。但对于一个有良好编程习惯的程序员来说, 应该主动地做收尾工作, 就像下面的代码一样。

rs. Close

Set rs=Nothing






Set cmd=Nothing

cn. Close

Set cn=Nothing

本栏目登载此文出于传递信息之目的，如有任何的问题请及时和我们联系！

无任何评论！

<p><b>请您注意：</b></p> <ul style="list-style-type: none"><li><input checked="" type="checkbox"/> 尊重网上道德，遵守《全国人大常委会关于维护互联网安全的决定》及中华人民共和国其他各项有关法律法规</li><li><input checked="" type="checkbox"/> 尊重网上道德，遵守中华人民共和国的各项有关法律法规</li><li><input checked="" type="checkbox"/> 承担一切因您的行为而直接或间接导致的民事或刑事法律责任</li><li><input checked="" type="checkbox"/> 中国网游研发中心新闻留言板管理人员有权保留或删除其管辖留言中的任意内容</li><li><input checked="" type="checkbox"/> 您在中国网游研发中心留言板发表的作品，中国网游研发中心有权在网站内转载或引用</li><li><input checked="" type="checkbox"/> 参与本留言即表明您已经阅读并接受上述条款</li></ul>	<p><b>发表评论：</b></p> <p>昵 称： <input type="text"/> <input type="button" value="GO"/></p> <p>联系EMAIL: <input type="text"/></p> <p>    </p> <p>j&lt; j&lt; j&lt; j&lt; j&lt; j&lt;</p> <div style="border: 1px solid gray; height: 60px; width: 100%;"></div>
--	--

[关于我们](#) - [免责声明](#) - [联络热线](#) - [申请链接](#) - [站点地图](#) - [网站帮助](#)

Copyright © 2004-2007 盛趣信息技术（上海）有限公司 All rights reserved.

OGDEV.NET -- 网络游戏研发网 最佳分辨率 1024×768