

您的潜力, 我们的动力

Microsoft
微软(中国)有限公司

ADO.NET 2.0 开发实践系列课程

用 **ADO.NET 2.0** 进行数据库应用程序开发 (下)

徐长龙
微软 **MSDN** 讲师

本系列课程简介

- **ADO.NET 2.0 开发实践系列课程**

2005.9.7	用 ADO.NET 2.0 进行数据库应用程序开发 (上)	本讲为 ADO.NET 入门课程, 包括 ADO.NET 2.0 介绍及数据访问的新功能介绍。
2005.9.21 (today)	用 ADO.NET 2.0 进行数据库应用程序开发 (下)	重点介绍 ADO.NET 2.0 的新特性, 以及何时如何使用这些新特性。

2005年9月7日课程回顾

- **ADO.NET 2.0 简介**
- 基于基类的提供者模型
- **Drag-Once 数据绑定**
- **Binding to Objects**
- 数据绑定的相关组件(控件)介绍
- 数据集设计器(DataSet Designer)
- 表适配器(TableAdapters)
- 保存连接字符串到程序配置文件

议程

- **DataSet 改进**
- 异步处理(Asynchronous Processing)
- 大批量数据操作(Bulk Copy Operations)
- Multiple Active Result Sets

DataSet 改进-新的索引引擎

Microsoft
微软(中国)有限公司

- 新的索引引擎
 - ADO.NET 2.0为DataTable重写了新的索引引擎
 - 针对大数据量的DataSet性能有明显的改良
 - 使新增、更新、删除的速度更快
 - 更快的Fill和Merge操作

DataSet 改进-新的索引引擎

Microsoft
微软(中国)有限公司

- 往一个 **DataTable** 中添加 **500,000** 笔数据行, 共两个字段, 一个索引字段
 - **ADO.NET 1.1** 需要 **127** 秒
 - **ADO.NET 2.0** 仅需要 **9** 秒
- 数据量越大差异越大

```
for (i = 1; i <= 500000; i++) {  
    try {  
        intValue = rand.Next();  
        dr = ds.Tables[0].NewRow();  
        dr["ID"] = intValue;  
        dr["Value"] = intValue;  
        ds.Tables[0].Rows.Add(dr);  
    } catch {}  
}
```

DataSet改进- 二进制序列化选项

您的潜力. 我们的动力

Microsoft
微软(中国)有限公司

- 二进制序列化选项(Binary Serialization Option)
 - 在ADO.NET 1.0/1.1中, DataSet序列化为XML, 无论是使用二进制方式还是Soap方式
 - 优点: 两种格式都支持Web Service方式传递, 真正的文本流
 - 缺点: 序列化后的文件体积庞大
 - ADO.NET 2.0支持真正的二进制序列化
 - 优点: 以二进制格式序列化后的文件体积小
 - 缺点: 不支持Web Service方式传递, .NET Remoting可以采用

```
ds.RemotingFormat = SerializationFormat.Binary;
```

DataSet改进- 二进制序列化选项

您的潜力. 我们的动力

Microsoft
微软(中国)有限公司

```
DataSet ds = new DataSet();  
SqlDataAdapter dadpt = new SqlDataAdapter("select * from [order details]", connstr);  
dadpt.Fill(ds);  
BinaryFormatter bf = new BinaryFormatter();  
FileStream fs = new FileStream(@"c:\xml.txt", FileMode.CreateNew);  
bf.Serialize(fs, ds);
```

- 序列化后文件大小是389K

```
DataSet ds = new DataSet();  
SqlDataAdapter dadpt = new SqlDataAdapter("select * from [order details]", connstr);  
dadpt.Fill(ds);  
BinaryFormatter bf = new BinaryFormatter();  
FileStream fs = new FileStream(@"c:\xml.txt", FileMode.CreateNew);  
ds.RemotingFormat = SerializationFormat.Binary;  
bf.Serialize(fs, ds);
```

- 序列化后文件大小是59K

DataSet改进- 更独立的DataTable

您的潜力, 我们的动力

Microsoft
微软(中国)有限公司

- 更独立的**DataTable**
 - 很多时候是一个**DataSet**仅包含一个**DataTable**对象
 - 在读取XML, 或存储为XML时, **ADO.NET 1.0/1.1**不得不将**DataTable**放入**DataSet**才能达成以上目的。
 - **ADO.NET 2.0**的**DataTable**可以直接操作XML, 像**DataSet**对象一样
 - ReadXML
 - ReadXMLSchema
 - WriteXML
 - WriteXMLSchema

DataSet改进- 更独立的DataTable

您的潜力. 我们的动力

Microsoft
微软(中国)有限公司

- Load方法
 - 从DataReader对象载入到DataTable
 - Load(IDataReader)
- Merge方法
 - 合并DataTable
 - Merge(DataTable)

议程

- DataSet 改进
- 异步处理(**Asynchronous Processing**)
- 大批量数据操作(**Bulk Copy Operations**)
- **Multiple Active Result Sets**

异步处理 (Asynchronous Processing)

您的潜力. 我们的动力

Microsoft
微软(中国)有限公司

- Visual Studio.Net 2005目前支持异步处理的提供程序有：
 - System.Data.SqlClient
- 其他提供程序也可以通过实现异步API实现异步处理
- 在针对大批量数据插入或更新时，使用异步处理方式可以不用等待所有数据更新完毕才能操作界面或进行下一步处理，改善用户体验
- SqlCommand对象方法如下：

同步方法	异步方法	
	"Begin" part	"End" part
ExecuteNonQuery	BeginExecuteNonQuery	EndExecuteNonQuery
ExecuteReader	BeginExecuteReader	EndExecuteReader
ExecuteXmlReader	BeginExecuteXmlReader	EndExecuteXmlReader
ExecuteScalar	/	/

异步处理

(Asynchronous Processing)

您的潜力. 我们的动力

Microsoft
微软(中国)有限公司

- **Begin**前缀的方法传入输入参数
- **End**前缀的方法返回输出参数和返回值

```
IAsyncResult ar = command.BeginExecuteReader();  
// ...  
// do other processing  
// ...  
SqlDataReader r = command.EndExecuteReader(ar);  
// use the reader and then close the reader and the connection
```

- **Begin**前缀的方法返回的是**IAsyncResult**
 - 用于追踪异步方法的执行状态

异步处理 (Asynchronous Processing)

您的潜力. 我们的动力

Microsoft
微软(中国)有限公司

- **IAsyncResult**
 - **AsnycState**
 - 用户自定义的状态对象
 - **AsyncWaitHandle**
 - 呼叫代码的等待形式，是等其中一个异步方法完成还是全部方法完成
 - **CompletedSynchronously**
 - 获取所有异步方法是否同时完成
 - **IsCompleted**
 - 是否执行完毕。呼叫代码可以根据此属性进行下一步动作

异步处理 (Asynchronous Processing)

您的潜力. 我们的动力

Microsoft
微软(中国)有限公司

- 连接字符串加入“**async=true**”
 - 如果所有的命令都是同步的建议在连接字符串中显式加入“**async=false**”
 - 如果有一部分命令是异步执行，又有一部分是同步执行，建议分别建立两个连接对象
 - 如果“**async=true**”时，也可以执行同步命令，但会耗损一些资源。

议程

- DataSet 改进
- 异步处理(Asynchronous Processing)
- 大批量数据操作(**Bulk Copy Operations**)
- Multiple Active Result Sets

大批量数据操作 (Bulk Copy Operations)

您的潜力. 我们的动力

Microsoft
微软(中国)有限公司

- 可以利用**SqlBulkCopy**类快速写入大批数据
 - 针对**SQL Server**的优化
 - 可写入**DataRow**数组, **DataTable**, **DataReader**

```
DataSet ds = new DataSet();  
SqlConnection conn = new SqlConnection(connstr);  
SqlDataAdapter dadp = new SqlDataAdapter("select * from customers", conn);  
dadp.Fill(ds);  
// Copy the Data to SqlServer  
SqlBulkCopy bcp = new SqlBulkCopy(connstr);  
bcp.DestinationTableName = "customers1";  
bcp.WriteToServer(ds.Tables[0]);
```

大批量数据操作 (Bulk Copy Operations)

您的潜力. 我们的动力

Microsoft
微软(中国)有限公司

- **WriteToServer(DataRow[])**
 - 批次写入数据行
- **WriteToServer(DataTable)**
 - 写入数据表
- **WriteToServer(DataTable, DataRowState)**
 - 按行状态写入数据表
- **WriteToServer(IDataReader)**
 - 写入DataReader对象

议程

- DataSet 改进
- 异步处理(Asynchronous Processing)
- 大批量数据操作(Bulk Copy Operations)
- **Multiple Active Result Sets**

Multiple Active Result Sets (MARS)

您的潜力. 我们的动力

Microsoft
微软(中国)有限公司

- ADO.NET 1.0/1.1中一个Command对象只能同时打开一个DataReader, 不能同时打开多个DataReader
- ADO.NET 2.0针对SQL Server 2005可以在同一个Command对象上同时打开多个DataReader.

```
SqlConnection conn = new SqlConnection(connstr);
conn.Open();
SqlCommand cmd1 = new SqlCommand("select * from customers", conn);
SqlCommand cmd2 = new SqlCommand("select * from orders", conn);
SqlDataReader rdr1 = cmd1.ExecuteReader();
// next statement causes an error prior to SQL Server 2005
SqlDataReader rdr2 = cmd2.ExecuteReader();
// now you can reader from rdr1 and rdr2 at the same time.
```

DEMO-Full Screen:Ctrl+H

您的潜力. 我们的动力

Microsoft
微软(中国)有限公司

ADO.NET 2.0 开发实践系列课程

用 ADO.NET 2.0 进行数据库应用程序开发 (下)

Add 500,000 records to a DataTable

Binary Serialization Option

DataTable Write XML

DataTable Read XML

DataTable Load DataReader

Async Command

label1

label2

label3

MARS


SqlBulkCopy

参考

- MSDN数据访问开发中心（英文）
 - <http://msdn.microsoft.com/data/>
- ADO.NET 2.0专栏（英文）
 - <http://msdn.microsoft.com/data/DataAccess/Whidbey/default.aspx>
- MSDN中文专栏（中文）
 - <http://www.microsoft.com/china/MSDN/library/data/default.mspx>



Question & Answer

如需提出问题，请单击“提问”按钮并在随后显示的浮动面板中输入问题内容。一旦完成问题输入后，请单击“提问”按钮。

 **问题和解答 (无问题)** ▲ ×

在此会议中尚未解答任何问题。

要向演示者提问，请在此处键入问

提问(A)

删除(D)

问题管理器(Q)

您的潜力. 我们的动力

Microsoft®
微软(中国)有限公司

Microsoft®

msdn


MSDN Webcasts