

您的潜力. 我们的动力

**Microsoft**  
微软(中国)有限公司

# SQL Server 2005 SqlCeResultSet 和高性能数据访问深度探索

马宁

Windows Mobile MVP



您的潜力. 我们的动力

**Microsoft**  
微软(中国)有限公司

# 目 录

什么是SqlCeResultSet?

Visual Studio 2005中使用SqlCeResultSet

SqlCeResultSet数据绑定

高性能数据访问

# 什么是SqlCeResultSet?

您的潜力, 我们的动力

**Microsoft**  
微软(中国)有限公司

- SqlCeResultSet是一个特殊的DataReader (实现IDataReader)
- 但是不像一般的DataReader, 你可以向前或向后来查看数据
- 而且可以将修改的数据Update到数据库中
- 因为实现了IListSource, 所以支持被WinForm控件进行数据绑定
- 始终与数据库连接, 不像DataSet, 可以实时显示被其他连接改变的数据
- .NET Compact Framework 2.0支持

# SqlCeResultSet Class

您的潜力, 我们的动力

**Microsoft®**

微软(中国)有限公司

SqlCeResultSet Class

SqlCeDataReader

Properties

- ContainsListCollection
- Item (+ 1 not shown)
- ResultSetView
- Scrollable
- Sensitivity
- Updatable

Methods

- CreateRecord
- Delete
- GetBoolean
- GetByte
- GetBytes
- GetChars
- GetDateTime
- GetDecimal
- GetDouble
- GetEnumerator
- GetFloat
- GetGuid
- GetInt16
- GetInt32
- GetInt64

- GetList
- GetSqlBinary
- GetSqlBoolean
- GetSqlByte
- GetSqlDateTime
- GetSqlDecimal
- GetSqlDouble
- GetSqlGuid
- GetSqlInt16
- GetSqlInt32
- GetSqlInt64
- GetSqlMetaData
- GetSqlMoney
- GetSqlSingle
- GetSqlString
- GetString
- GetValue
- GetValues
- Insert (+ 1 overload)
- IsDBNull
- IsSetAsDefault
- OnMove
- ReadAbsolute
- ReadFirst
- ReadLast
- ReadPrevious
- ReadRelative
- SetBoolean
- SetByte
- SetBytes

- SetChar
- SetChars
- SetDateTime
- SetDecimal
- SetDefault
- SetDouble
- SetFloat
- SetGuid
- SetInt16
- SetInt32
- SetInt64
- SetObjectRef
- SetSqlBinary
- SetSqlBoolean
- SetSqlByte
- SetSqlDateTime
- SetSqlDecimal
- SetSqlDouble
- SetSqlGuid
- SetSqlInt16
- SetSqlInt32
- SetSqlInt64
- SetSqlMoney
- SetSqlSingle
- SetSqlString
- SetString
- SetValue
- SetValues
- SqlCeResultSet
- Update

您的潜力. 我们的动力

**Microsoft**  
微软(中国)有限公司

# 在Visual Studio 2005使用 Typed Result Sets

**demo**



# 使用Typed Result Sets

您的潜力. 我们的动力

**Microsoft**  
微软(中国)有限公司

- 使用Visual Studio 2005 Data Source wizard
- 将.xsd file Custom Tool 属性改为 MSResultSetGenerator
- 为了增加适应性, 通过partial classe扩展产生 typed SqlCeResultSet 代码
  - Override the constructor so calling Open is optional
  - Override Open to specify index and range or SQL statement or whatever you need
- 在Data Source页中使用数据绑定控件
- Typed ResultSet可以通过new操作符来创建

# ResultSetOptions

您的潜力. 我们的动力

**Microsoft**  
微软(中国)有限公司

- 使用SqlCeCommand.ExecuteResultSet创建对象
- 无须设置ResultSetOptions参数（对象按照默认行为执行）
- **Insensitive** –不会发现其他数据连接对数据源的改变
- **Sensitive** –察觉来自其他数据连接的改变
  - Base table cursors的属性始终为sensitive
  - TableDirect command type => base table cursor
- **Scrollable** –你可以向前或向后读取数据，使用内部的BookmarkArray
- **Updatable** –允许更新
- **None** –只读，单向浏览数据

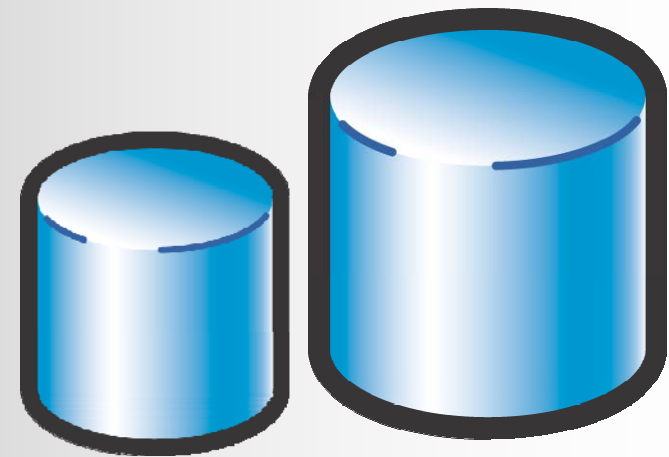
# ResultSet加载数据

您的潜力, 我们的动力

**Microsoft**  
微软(中国)有限公司

## 从数据库中加载数据

- 使用`sqlCeResultSet`
- 使用`TableDirect`
- 如果需要修改数据库, 则使用  
`ResultSetType.Updatable`





您的潜力. 我们的动力

**Microsoft**  
微软(中国)有限公司

# Data Binding 和 SqlCeResultSet

**demo**

# 使用BindingSource

您的潜力, 我们的动力

Microsoft®  
微软(中国)有限公司

- BindingSource控件(new in Microsoft .NET 2.0)
- 提供数据源与窗体控件间的间接逻辑层
- 为数据集合提供strong-typed数据绑定
- 将BindingSource.DataSource设置为ResultSetView中, 然后将控件的DataSource属性设置为BindingSource
  - Visual Studio中的大部分控件可以用于数据绑定
  - 控件被绑定到同一个BindingSource上, 可以保持一致

使用SqlCeResultSet.Bind方法绑定BindingSource

**Bind**

使用BindingSource方法来移动SqlCeResultSets

**MoveFirst**

**MoveLast**

**MoveNext**

**MovePrevious**

# BindingSource更新

您的潜力. 我们的动力

Microsoft®  
微软(中国)有限公司

在data bound控件中的任何改变, 可以更新到数据库中

- 调用EndEdit()
- 移动到另外的数据行时(在内部调用EndEdit)





# ResultSetView数据绑定

您的潜力, 我们的动力

**Microsoft**  
微软(中国)有限公司

- 不需要使用BindingSource, 也可以进行数据绑定

```
DataGrid.DataSource = your ResultSetView
```

- 简单control(例如TextBox) 使用ResultSetView 作为data source参数

```
TextBox.DataBindings.Add("text", ResultSet.ResultSetView,  
"Customer ID", true, DataSourceUpdateMode.Never);
```

- List controls (ListBox, ComboBox)

```
ListBox.DataSource = your view
```

```
ListBox.DisplayMember = view.Columns[n]
```

```
ListBox.ValueMember = view.Columns[n]
```



您的潜力. 我们的动力

**Microsoft**  
微软(中国)有限公司

# 高性能数据访问

Base Table Cursor

Seek/SetRange方法

Data Reader与Data Set

SqlCeResultSet的效率

您的潜力. 我们的动力

**Microsoft**  
微软(中国)有限公司

# DataSet, DataReader和 SqlCeResultSet

## demo

# Demo 测试结果

您的潜力. 我们的动力

**Microsoft**  
微软(中国)有限公司

	Load 10000	Read 1000	Read Store	Update 4000
<b>SQL Syntax</b>			DataSet	
SQL statement	47500		1366	1900
TableDirect			1140	
<b>Data Reader</b>			List<T>	
SQL statement		30		
TableDirect		6	700	
<b>Result Set</b>			List<T>	
SQL statement		350		1900
TableDirect	20000	7*	730	6000



# Base Table Cursor

- 可以设置index, 用于查询指定值、排序、设置范围
- 获取一个数据行中的所有项, 不能根据Select语句获取数据子集
- 从数据表中读取所有列的速度最快

Example:

```
// Create and execute SqlCommand
SqlCommand cmd = new SqlCommand("Authors", cnn);
cmd.CommandType = CommandType.TableDirect;
SqlCeDataReader dr = cmd.ExecuteReader();
// Retrieve Results
while(dr.Read())
{
    MessageBox.Show("Name = " + dr["au_lname"]);
}
dr.Close();
dr.Dispose();
// Create Result Set
SqlCeResultSet rs =
cmd.ExecuteResultSet(ResultSetOptions.Scrollable |
ResultSetOptions.Updatable);
```

# Seek/SetRange: 基本用法

您的潜力. 我们的动力

**Microsoft**  
微软(中国)有限公司

- 用于打开表的index
- 选择值的范围的最快方法

## Example:

```
cmd.CommandType = CommandType.TableDirect;  
cmd.CommandText = "Orders";
```

```
//Assume: Index contains one column [datetime]
```

```
cmd.IndexName = "SomeIndex";  
object[] start = new object[1];  
object[] end = new object[1];  
start[0] = new SqlDateTime(2001, 1, 1);  
end[0] = new SqlDateTime(2002, 2, 3);
```

```
cmd.SetRange(DbRangeOptions.Match, start, end);  
SqlCeDataReader dr = cmd.ExecuteReader();  
dr.Seek(DbSeekOptions.FirstEqual,  
    new SqlDateTime(2001,3,4));
```

```
while(dr.Read()) { } // Read data in the usual way
```



# Seek/SetRange 与Scrollability

您的潜力. 我们的动力

**Microsoft**  
微软(中国)有限公司

- SqlCeDataReader提供有限的scrolling功能
- SqlCeResultSet
  - 可以使用base-table cursor (TableDirect)
  - 可以将SqlCeCommandBuilder与base-table cursor一起使用

**Example:**

```
// Set the index range
cmd.SetRange(DbRangeOptions.InclusiveStart, start, end);
SqlCeDataReader dr = cmd.ExecuteReader();
SqlCeResultSet rs = cmd.ExecuteResultSet(ResultSetOptions.Scrollable);

// Seek to a value (customer name)
dr.Seek(DbSeekOptions.FirstEqual, "Shaffer");
dr.Read();
// Process the row in the usual way

// Seek to another value (customer name)
rs.Seek(DbSeekOptions.LastEqual, "Smith");
rs.Read();
rs.ReadPrevious();
// Process the row in the usual way
```

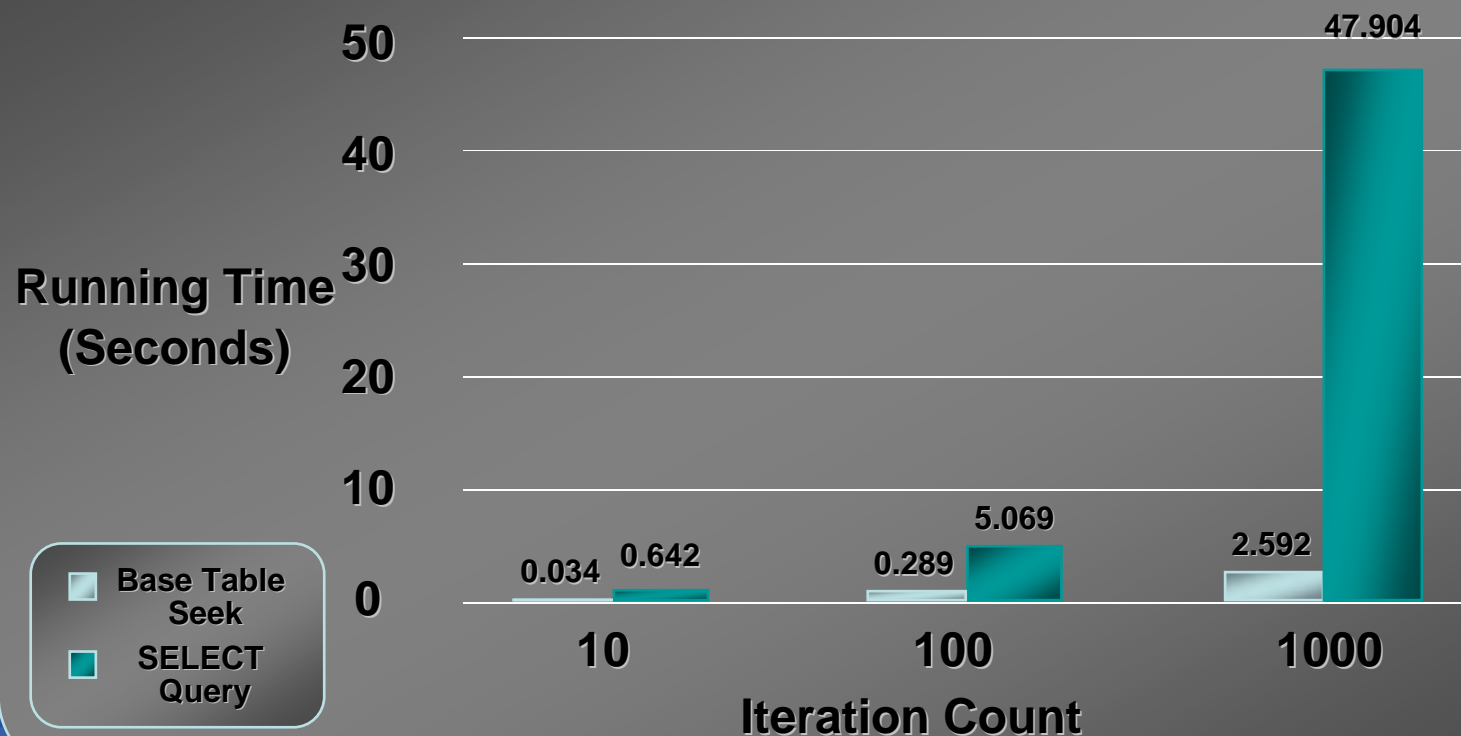
# Use Queries?

您的潜力. 我们的动力

**Microsoft**  
微软(中国)有限公司

SELECT "Customer ID" FROM Orders WHERE "Order ID" = 10123  
Versus Base Table

Base Table Seek vs. SELECT Query Performance Comparison



# Data Reader与Data Set

您的潜力. 我们的动力

**Microsoft**  
微软(中国)有限公司

- 尽可能使用DataReader
- DataReader比DataSet更快, 尤其是使用大的结果集时 (在某些情况下可以快5倍)
- 直接访问 SQL Server CE/SQL Mobile
- 没有提供对数据的缓存
- DataSet可以更新所有的数据列
- Data Reader的缺点
  - 不能与控件进行绑定
  - 不能更新
  - 单向浏览



# Data Reader: 最佳实践

- 读取完整的数据行时, 使用GetValues()要优于多次调用GetXXX
- 使用强类型比使用GetValue()更好
- 引用数据列时, 使用序号比使用名称更快
- 使用SqlType的系统开销大于CLR类型

## Example:

```
object[] row = new object[rdr.FieldCount];
while (rdr.Read())
{
    // Fewer function calls inside the data provider
    rdr.GetValues(row);
    // Process the row here
}
```

您的潜力. 我们的动力

**Microsoft**  
微软(中国)有限公司

# SqlCeResultSet

- 支持CF2.0/SQLMobile
  - 双向滚动访问
  - 可更新
  - 随机访问
- 目的:
  - 提高效率
  - 控件绑定
  - 快速开发



# SqlCeResultSet

## Example:

```
private SqlCeResultSet resultSet = null;
private ResultSetView view1 = null;
///
private void _bindData()
{
    this.command.CommandText = "SELECT * FROM Orders";
    ResultSetOptions options = ResultSetOptions.Scrollable |
        ResultSetOptions.Updatable;

    this.resultSet = this.command.ExecuteResultSet(options);

    this.view1 = this.resultSet.ResultSetView;
    int[] ordinals = new int[] { 1,3,5,8 };
    this.view1.OrdinalIndex = ordinals;

    this.dataGrid.DataSource = view1;
}
```

# ResultSet使用原则

您的潜力, 我们的动力

**Microsoft**  
微软(中国)有限公司

从数据库中读取数据

SqlCeResultSet 几乎与 SqlCeDataReader一样快

不需要对数据进行缓存, 因为随时可以通过  
SqlCeResultSet直接从数据库中获取所需要的  
数据

可以直接更新数据

# ResultSet的效率

您的潜力. 我们的动力

**Microsoft**  
微软(中国)有限公司

- 更新批量数据, SQL语句通常是最快的
- 更新单独的数据, 仍然是  
`SqlCeResultSet`比较快
- 插入新记录时, `SqlCeResultSet`最快

# Summary

您的潜力. 我们的动力

**Microsoft**  
微软(中国)有限公司

对于SQL Server Everywhere Edition的数据  
SqlCeResultSet可以提供很大的性能改进

Visual Studio 2005的data source 向导和数据绑定  
Control可以让创建typed  
SqlCeResultSets更加快捷有效




# 获取更多MSDN资源

- **MSDN中文网站**  
<http://www.microsoft.com/china/msdn>
- **MSDN中文网络广播**  
<http://www.msdnwebcast.com.cn>
- **MSDN Flash**  
<http://www.microsoft.com/china/newsletter/case/msdn.aspx>
- **MSDN开发中心**  
<http://www.microsoft.com/china/msdn/DeveloperCenter/default.msp>





# 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**