

Delphi 5
高级编程丛书之三

Delphi 5

高级编程

— Database 与 MIDAS 编程

徐新华 编著



人民邮电出版社
www.pptph.com.cn

Delphi 5 高级编程丛书之三

Delphi 5 高级编程

——Database 与 MIDAS 编程

徐新华 编著

人民邮电出版社

内 容 提 要

本书全面深入地介绍了 Delphi 5 的数据库编程技术,包括数据集、数据库访问链路、数据感知控件、ADO、Interbase、Client/Server、MIDAS、决策支持、BDE API、自定义数据集、QuickReport 报表、TeeChart 图表、数据库浏览器等内容。

本书内容全面而又不失简洁,范例丰富。既可以作为广大读者学习 Delphi 5 的入门指导书,也可以作为程序员编程时的参考手册。

本书读者对象为计算机程序开发人员、大专院校计算机专业师生。

JS252/09

Delphi 5 高级编程丛书之三

Delphi 5 高级编程

——Database 与 MIDAS 编程

◆ 编 著 徐新华

责任编辑 李 际

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号

北京鸿佳印刷厂印刷

新华书店总店北京发行所经销

◆ 开本:787×1092 1/16

印张:27.25

字数:675 千字

2000 年 4 月第 1 版

印数:1-6 000 册

2000 年 4 月北京第 1 次印刷

ISBN 7-115-08465-3/TP·1584

定价:43.00 元

2000.4

前 言

Delphi 5 是 Inprise 公司面向 Internet 和电子商务推出的一个战略产品。在 Internet 日益普及和众多厂商看好电子商务的今天, Delphi 5 的推出必将受到程序员的热烈欢迎。

为了帮助广大用户全面、准确地掌握 Delphi 5 的编程思想和用法, 我们编写了这套《Delphi 5 高级编程丛书》。作为《Delphi 4 高级编程丛书》的升级, 本套丛书在保留原丛书精髓的基础上, 对丛书的结构和内容作了大刀阔斧的修改, 这是考虑到很多程序员已经初步掌握了 Delphi 5 的基本用法, 现在需要的是进一步提高和精通。

本套丛书分为 4 册。第一册是《Delphi 5 高级编程——IDE 与面向对象编程》, 第二册是《Delphi 5 高级编程——GUI 编程》, 第三册是《Delphi 5 高级编程——Database 与 MIDAS 编程》, 第四册是《Delphi 5 高级编程——COM、CORBA 与 Internet 编程》。

本书是此套丛书的第三册, 全面深入地介绍了 Delphi 5 的数据库编程技术, 包括数据集、数据库访问链路、数据感知控件、ADO、Interbase、Client/Server、MIDAS、决策支持、BDE API、自定义数据集、QuickReport 报表、TeeChart 图表、数据库浏览器等内容。

由于我们的水平有限, 再加上时间很紧, 尽管我们作了严格的审核和测试, 书中还是难免会有一些错误, 敬请广大读者不吝赐教, 我们谨在此表示感谢。

为了帮助广大程序员更好地掌握这个优秀的开发工具, 北京东大阿尔发软件技术有限公司愿意为购买此书的读者提供技术咨询。

北京东大阿尔发软件技术有限公司
地址: 北京市上地信息产业基地上地村路 1 号(100085)
电话: (010)62987260 传真: (010)62985141
网址: <http://www.allfa.com.cn> 邮件: books@allfa.com.cn

徐新华
2000 年 2 月

目 录

第一章 数据集	1
1.1 TDataSet.....	2
1.2 TBDEDataSet.....	25
1.3 TDBDataSet.....	30
1.4 字段对象.....	32
1.4.1 具体的字段对象.....	32
1.4.2 动态字段对象.....	33
1.4.3 永久字段对象.....	33
1.4.4 属性集.....	37
1.4.5 TField.....	38
1.5 字段定义.....	50
1.6 Oracle8 的对象字段.....	52
1.6.1 ADT 字段.....	52
1.6.2 数组字段.....	53
1.6.3 数据集字段.....	54
1.6.4 引用字段.....	54
1.7 BLOB 字段.....	55
第二章 数据库访问链路	61
2.1 数据源.....	62
2.2 访问数据库表.....	64
2.2.1 访问数据库表的一般步骤.....	64
2.2.2 TTable.....	65
2.2.3 TIndexDef.....	78
2.2.4 TCheckConstraint.....	80
2.2.5 Master/Detail 关系.....	81
2.3 查询数据库.....	82
2.3.1 查询数据库的一般步骤.....	82
2.3.2 TQuery.....	83
2.3.3 TParam.....	90
2.3.4 TParams.....	92
2.3.5 用 SQL 进行统计.....	95
2.3.6 SQL Builder.....	96
2.4 存储过程.....	98

2.4.1 访问存储过程的一般步骤.....	99
2.4.2 存储过程的参数.....	99
2.4.3 TStoredProc.....	100
2.5 连接数据库.....	102
2.6 BDE 会话期.....	111
2.6.1 TSession.....	111
2.6.2 TSessionList.....	122
2.6.3 动态创建 TDatabase 和 TSession.....	123
2.7 批量移动数据.....	125
2.8 缓存更新.....	128
2.9 访问嵌套表.....	132
2.10 数据模块.....	132
2.10.1 为什么要使用数据模块.....	132
2.10.2 加入数据模块.....	133
2.10.3 命名.....	133
2.10.4 重用.....	134
2.11 访问文本文件.....	134
2.12 用 ODBC 连接.....	137
第三章 数据感知控件.....	141
3.1 显示和编辑数据的一般步骤.....	142
3.2 TDBGrid.....	142
3.2.1 TDBGrid.....	143
3.2.2 TDBGridColumns.....	148
3.2.3 在设计期设置列的属性.....	150
3.2.4 在运行期操纵列的属性.....	150
3.3 TDBNavigator.....	154
3.4 TDBText.....	158
3.5 TDBEdit.....	159
3.6 TDBMemo.....	159
3.7 TDBImage.....	161
3.8 TDBListBox.....	162
3.9 TDBComboBox.....	163
3.10 TDBCheckBox.....	163
3.11 TDBRadioGroup.....	164
3.12 TDBLookupListBox.....	165
3.13 TDBLookupComboBox.....	166
3.14 TDBRichEdit.....	167
3.15 TDBCtrlGrid.....	168

第四章 ADO	172
4.1 关于 ADO 的概述	173
4.2 连接 ADO 数据库	174
4.2.1 用 TADOConnection 来连接	174
4.2.2 异步连接.....	176
4.2.3 连接属性.....	177
4.2.4 超时.....	177
4.2.5 登录.....	178
4.2.6 访问依附于连接的数据集.....	179
4.2.7 访问依附于连接的命令.....	180
4.2.8 列出所有可用的表.....	180
4.2.9 列出可用的存储过程.....	181
4.2.10 事务.....	181
4.3 ADO 数据集.....	182
4.3.1 连接 ADO 数据库.....	182
4.3.2 显示和编辑数据.....	182
4.3.3 批量更新.....	183
4.3.4 从文件中存取数据.....	184
4.3.5 在命令中使用参数.....	185
4.4 使用 TADODataset.....	186
4.5 使用 TADOTable.....	187
4.6 使用 TADOQuery	188
4.7 使用 TADOStoredProc.....	188
4.8 执行命令	190
第五章 访问 Interbase	192
5.1 关于 InterBase 的概述.....	193
5.2 配置 InterBase 别名.....	195
5.3 创建数据库	196
5.4 创建表	197
5.5 用 WISQL 连接和浏览数据库.....	197
5.6 事务	199
5.7 安全性	201
5.8 备份表	202
5.9 Interbase Express	203
5.10 TIBCustomDataSet	203
5.11 TIBTable.....	208
5.12 TIBQuery	213

5.13 TIBStoredProc	217
5.14 TIBDatabase.....	219
5.15 TIBTransaction	224
5.16 TIBUpdateSQL	228
5.17 TIBDataSet.....	229
5.18 TIBSQL.....	231
5.19 TIBDatabaseInfo.....	235
5.20 TIBSQLMonitor.....	237
5.21 TIBEvents	237
第六章 Client/Server.....	239
6.1 什么情况下采用 Client/Server 模式.....	240
6.2 Client/Server 的体系结构	240
6.3 与桌面数据库开发的比较	242
6.3.1 面向子集与面向记录.....	242
6.3.2 数据安全性.....	242
6.3.3 数据完整性.....	242
6.3.4 记录锁定的方式.....	243
6.3.5 面向事务.....	243
6.4 服务器：设计后端	243
6.4.1 定义数据库表.....	244
6.4.2 域	246
6.4.3 视图.....	247
6.4.4 存储过程.....	247
6.4.5 触发器.....	251
6.4.6 对数据库对象的访问权限.....	252
6.5 登录	253
第七章 MIDAS	257
7.1 应用服务器	258
7.2 TDataSetProvider.....	261
7.3 “瘦”客户	270
7.4 TCustomRemoteServer	272
7.5 TDispatchConnection.....	274
7.6 TCOMConnection.....	276
7.7 TDCOMConnection.....	276
7.8 TOLEnterpriseConnection	276
7.9 TStreamedConnection.....	277
7.10 TSocketConnection.....	278

7.11 TWebConnection.....	279
7.12 TCorbaConnection.....	280
7.13 TSimpleObjectBroker.....	281
7.14 TClientDataSet.....	284
7.15 “公文包”模式.....	309
7.16 创建 ActiveForm 形式的“瘦”客户.....	309
第八章 决策支持.....	311
8.1 创建决策支持程序的一般步骤.....	312
8.2 引入数据集.....	313
8.3 建立数据仓库.....	314
8.4 决策方编辑器.....	320
8.5 决策源.....	322
8.6 数据透视表.....	329
8.7 决策栅格.....	331
第九章 调用 BDE API.....	336
9.1 Check().....	337
9.2 游标和句柄.....	339
9.3 dBASE 表.....	339
9.3.1 物理记录编号.....	339
9.3.2 显示被软删除的记录.....	341
9.3.3 测试被软删除的记录.....	341
9.3.4 恢复被软删除的记录.....	342
9.3.5 紧缩.....	342
9.4 Paradox 表.....	343
9.4.1 获取记录的序号.....	343
9.4.2 紧凑.....	344
9.5 限制查询结果.....	351
9.6 快速进行表复制.....	352
9.7 列出所有的用户.....	353
第十章 自定义数据集.....	355
10.1 创建自定义数据集.....	356
10.2 打开、初始化和关闭数据集.....	359
10.3 管理记录缓冲区.....	361
10.4 书签管理.....	364
10.5 记录导航.....	365
10.6 编辑.....	366

10.7 记录编号	367
10.8 异常处理	368
10.9 TMyDataSet 的完整代码	368
第十一章 QuickReport 报表	382
11.1 QuickReport 概述	383
11.2 建立报表的一般步骤	383
11.2.1 一个最简单的报表	384
11.2.2 一个稍复杂的报表	384
11.2.3 基于自定义的文本建立报表	385
11.2.4 自定义预览窗口	385
11.3 TQuickRep	386
11.4 TQRSubDetail	392
11.5 TQRBand	393
11.6 TQRChildBand	395
11.7 TQRGroup	396
11.8 TQRLabel	396
11.9 TQRDBText	397
11.10 TQRExpr	397
11.11 TQRSysData	398
11.12 TQRMemo	399
11.13 TQRRichText	400
11.14 TQRDBRichText	400
11.15 TQRShape	400
11.16 TQRImage、TQRDBImage	400
11.17 TQRCompositeReport	400
11.18 TQRPreview	401
11.19 TQRPrinter	401
11.20 TQRTextFilter 和 TQRCSVFilter	404
11.21 TQRHTMLFilter	405
第十二章 TeeChart 图表	406
12.1 制作 TeeChart 图表的一般步骤	407
12.2 TeeChart 向导	408
12.3 图表编辑器	410
12.4 引出图表	410
12.5 预览和打印图表	411
12.6 创建数据库图表	412
12.7 在 QuickReport 报表上创建图表	413

12.8 创建决策图表	414
第十三章 数据库浏览器	415
13.1 数据库浏览器的窗口	416
13.2 建立和维护数据库别名	417
13.3 信息窗格	417
13.4 访问数据库表	420
13.5 数据字典	420

第一章 数 据 集

本章有以下内容:

- TDataSet;
- TBDEDataSet;
- TDBDataSet;
- 字段对象;
- 字段定义;
- Oracle 8 的对象字段;
- BLOB 字段。

数据集是一组离散的数据记录的集合。在 Delphi 5 中,数据集有 3 种表现形式:表、查询、存储过程,这 3 种形式的数据集分别用 TTable、TQuery、TStoredProc 来操纵。

TTable、TQuery、TStoredProc 的直接上级是 TDBDataSet, TDBDataSet 是从 TBDEDataSet 派生的,而 TBDEDataSet 又是从 TDataSet 派生的。

Delphi 5 用 TClientDataSet 来实现和操纵分布式数据集,而 TClientDataSet 也是从 TDataSet 派生的。

Delphi 5 还允许通过 ActiveX Data Objects(ADO)来访问数据,而 TADODataSet 或更具体地说是 TADOTable 和 TADOQuery 也是从 TDataSet 派生的。

可见,不管是基于 BDE 的数据集,还是分布式数据集,或者是 ADO 数据集, TDataSet 是它们的共同基类。这些数据集之间的继承关系如图 1.1 所示。

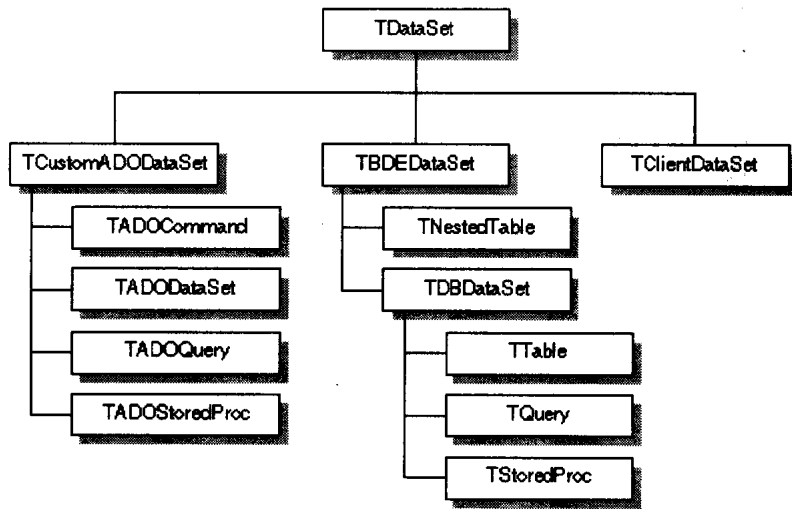


图 1.1 数据集的继承关系

1.1 TDataSet

TDataSet 是所有数据集的虚拟基类。不能直接创建它的对象实例,也不能直接访问它的特性和方法,因为这些特性和方法大部分是虚拟的或抽象的。

如果从功能上划分, TDataSet 的成员可以分为这么几大块:打开和关闭数据集、浏览记录、编辑数据、书签管理、控制连接、访问字段、记录缓冲区管理、过滤、事件。

Active 特性

声明: property Active: Boolean;

如果这个特性设为 True,相当于调用 Open()来打开数据集;如果这个特性设为 False,相当于调用 Close()来关闭数据集。

您也可以通过 Active 特性来判断数据集是否打开。程序示例如下:

```
Table1.Active := False;
```

```

try
    Table1.DatabaseName := 'Delphi_Demos';
    Table1.Active := True;
except
    on EDatabaseError do
        Table1.DatabaseName := 'c:\delphi\demos\database';
        Table1.Active := True;
end;

```

提示: 如果要访问 SQL 服务器, 在打开数据库中的一个数据集之前, 先要连接数据库。当关闭了数据库中的最后一个数据集时, 连接就会终止。连接或断开数据库会有一些开销, 因此, 最好用 TDatabase 元件来管理数据库的连接。

AutoCalcFields 特性

声明: property AutoCalcFields: Boolean;

当程序从数据集中检索记录时, 将触发 OnCalcFields 事件。如果 AutoCalcFields 特性设为 True, 当计算字段的值被更新时, 也将触发这个事件。

注意: 如果 AutoCalcFields 设为 True, 在处理 OnCalcFields 事件的句柄中不能再修改数据集; 否则, 将老是触发这个事件, 从而导致无限循环。

BlockReadSize 特性

声明: property BlockReadSize: Integer;

如果这个特性设为 0, 每当调用 Next() 时, 关联的数据感知控件将刷新。为了提高应用程序的性能, 可以把 BlockReadSize 特性设为一个大于 0 的数, 这样, 只有当调用了 Next() 若干次后, 关联的数据感知控件才刷新一次。

BOF 特性

声明: property BOF: Boolean;

如果当前记录是数据集的第一条记录, 则这个只读的特性将返回 True。当程序第一次打开一个数据集或者调用了 First() 后, 当前记录就是第一条记录。程序示例如下:

```

procedure TForm1.Table1DataChange(Sender: TObject; Field: TField);
begin
    if Table1.BOF then
        CopyData.Enabled := False
    else
        CopyData.Enabled := True;
end;

```

Bookmark 特性

声明: property Bookmark: TBookmarkStr;

书签用于保存数据集中某个记录的位置。这样,就可以方便地回到书签所标记的地方。

如果读这个特性的话,这个特性将返回当前记录的书签(如果有的话)。也可以写这个特性来指定一个已有的书签,使该书签标记的记录成为当前记录。

假设要标记当前记录的位置,以便日后访问。程序示例如下:

```
var
    BM: TBookmarkStr;
begin
    BM := Table1.Bookmark;
    ...
end;
```

上述代码读取了 Bookmark 特性,从而取得了书签;要回到书签标记的地方,可以这样调用:

```
Table1.Bookmark := BM;
```

TBookmarkStr 实际上是 AnsiString 类型的别名,因此,分配给它的内存是自动管理的,不需要手工地释放它们。如果想删除一个书签,只要用一个空字符串对它赋值,例如:

```
BM := ' ';
```

注意: Delphi 5 仍然支持 GetBookmark()、GotoBookmark()和 FreeBookmark()。不过,最好不要用这些方法,除非要保持对 16 位程序的兼容。

CanModify 特性

声明: property CanModify: Boolean;

如果这个只读的特性返回 False,表示数据集是只读的,不能修改。不过,即使 CanModify 特性返回 True,也并不意味着数据集一定能修改。例如,要修改 SQL 表,需要有访问权限。

如果 ReadOnly 特性设为 True, CanModify 特性肯定返回 False。但如果 ReadOnly 特性设为 False,只有当获得了对数据集的修改权限后, CanModify 特性才返回 True。

Constraints 特性

声明: property Constraints: TCheckConstraints;

这个特性用于给数据集建立纠错规则(仅限于记录级)。在运行期,可以通过 TCheckConstraint 对象和 TCheckConstraints 对象来操纵每一个纠错规则。

DataSetField 特性

声明: property DataSetField: TDataSetField;

这个特性是一个 TDataSetField 对象,它代表一个嵌套的数据集。

DefaultFields 特性

声明: property DefaultFields Boolean;

如果数据集中包含运行期动态生成的字段(TField 对象), 这个只读的特性将返回 True。如果用字段编辑器创建了永久的字段, 则这个特性将返回 False。

EOF 特性

声明: property EOF: Boolean;

如果当前记录是数据集的最后一条记录, 则这个只读的特性将返回 True。当程序打开一个空的数据集或者调用了 Last()时, 当前记录就是最后一条记录。程序示例如下:

```
with CustTable do
begin
  DisableControls;
  try
    First;
    while not Eof do
    begin
      { Process each record here }
      Next;
    end;
  finally
    EnableControls;
  end;
end;
```

FieldCount 特性

声明: property FieldCount: Integer;

这个只读的特性返回数据集中的字段数。由于数据集中可能包含动态生成的字段, 因此, 这个特性返回的字段数与物理字段数可能不同。程序示例如下:

```
var
  I: Integer;
  Info: String;
begin
  Info := 'The fields of table ' + Table1.TableName + ' are: '#13#10#13#10;
  for I := 0 to Table1.FieldCount - 1 do
    Info := Info + Table1.Fields[I].FieldName + '#13#10;
  ShowMessage(Info);
end;
```


FieldDefs 特性

声明: property FieldDefs: TFieldDefs;

Delphi 5 用 TFieldDef 对象来操纵数据集中的每一个字段(计算字段除外)。

一个数据集中有几个字段,就有几个 TFieldDef 对象。这些 TFieldDef 对象由 FieldDefs 特性(TFieldDefs 对象)来管理。

一般情况下,不需要修改字段的定义,除非要在运行期动态创建一个表或字段。下面的代码动态创建了一个表:

```
if not Table1.Exists then
begin
  with Table1 do
  begin
    Active := False;
    DatabaseName := 'DBDEMOS';
    TableType := ttParadox;
    TableName := 'CustInfo';
    with FieldDefs do
    begin
      Clear;
      with AddFieldDef do
      begin
        Name := 'Field1';
        DataType := ftInteger;
        Required := True;
      end;
      with AddFieldDef do
      begin
        Name := 'Field2';
        DataType := ftString;
        Size := 30;
      end;
    end;
  end;
  with IndexDefs do
  begin
    Clear;
    with AddIndexDef do
    begin
      Name := '';
      Fields := 'Field1';
      Options := [ixPrimary];
    end;
  end;
end;
```