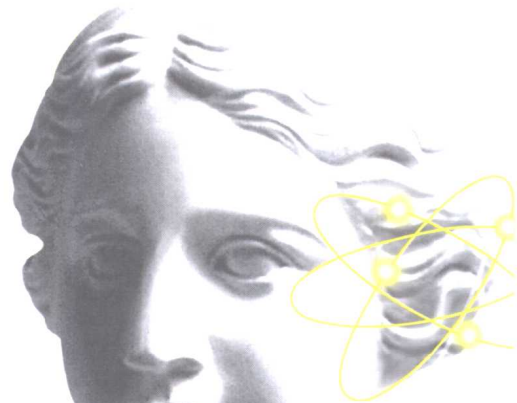


HZ BOOKS

李维 作品系列



Delphi 7

高效数据库程序设计

李维 著

Delphi 6/Kylix 1/2/3 适用

Borland新一代dbExpress提供了跨平台的高效率的数据引擎。dbExpress不但提供广泛的关系型数据库存取能力，并且适用于客户机/服务器、Web和多层的应用系统。本书不但深入讨论dbExpress的使用技巧，更辅以丰富的范例让读者完全掌握Delphi/Kylix的dbExpress和DataSnap技术。



附 赠



机械工业出版社
China Machine Press

李维 作品系列

TP311.56
123D



Delphi 7

高效数据库程序设计

李维 著

北方工业大学图书馆



00523939



机械工业出版社
China Machine Press

dbExpress是Borland公司下一代数据访问技术。本书不仅详细介绍了dbExpress的基本功能、使用技巧以及Delphi/Kylix的DataSnap技术,还详细讨论了dbExpress的实现原理以及dbExpress的未来发展趋势。本书结构清晰,讲解透彻,实例丰富。作者李维是Borland公司著名技术专家,曾著有多部Delphi名著。通过阅读本书,读者可以充分利用dbExpress的强大功能,开发出高效率的Delphi/Kylix系统。本书适合Delphi/Kylix/C++Builder程序员、数据库程序设计师以及对Delphi/Kylix感兴趣的人员阅读。

本书简体字版由李维授权机械工业出版社出版。

版权所有,侵权必究。

图书在版编目(CIP)数据

Delphi 7高效数据库程序设计, 李维著; -北京:机械工业出版社, 2003.1
(李维作品系列)

ISBN 7-111-11226-1

I. D… II. ①李… III. 软件工程-程序设计 IV. TP311.56

中国版本图书馆CIP数据核字(2002)第089979号

机械工业出版社(北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑:朱 劼

北京第二外国语学院印刷厂印刷·新华书店北京发行所发行

2003年1月第1版第1次印刷

787mm×1092mm 1/16·26.25印张

印数:0 001-6 000册

定价:78.00元(附光盘)

凡购本书,如有倒页、脱页、缺页,由本社发行部调换

序

自从Delphi推出以来，Delphi全方位的功能便让许多程序员津津乐道。而Delphi在数据库方面提供的强大又富有弹性的能力更是让许多数据库程序设计师惊喜不已。Delphi的BDE/IDAPI不但提供了合理的性能，并且能够连接各种不同的数据库，从基于文件的Paradox/dBase数据库到真正的关系型数据库（如InterBase、Oracle、Informix、DB2和MS SQL Server等），都能够让Delphi程序员使用相同的技术来访问各种不同的数据源，实在是非常方便。

但是随着软件技术的进步以及用户需求的变化，数据访问的方法也随之出现巨大的改变。从以往提供持续连接的主从结构应用系统转换成现在的批处理，而且客户端和数据库之间可能处于没有连接的状态。例如，现在的数据访问技术必须同时能够满足企业内部的主从结构式应用系统，以及能够提供Web访问的分布式应用系统。虽然BDE/IDAPI技术也可以同时在这两种应用中使用，但是BDE/IDAPI毕竟是按照基于文件和主从结构概念设计的，对于基于Web和分布式结构的应用则无法提供比较有效率和经济的使用方式。

Borland希望满足用户的新需求并向使用Borland开发工具的用户提供跨平台的能力，而且Borland推出了在Linux上执行的开发工具——Kylix，因此Borland必须认真地考虑开发下一代的数据访问技术，因为BDE/IDAPI明显已经无法满足新的应用而且无法提供跨平台的功能。于是，dbExpress应运而生。

dbExpress是Borland下一代的数据访问技术。dbExpress的设计目标是提供高效率数据访问以及提供跨平台能力的数据库访问引擎。自从Borland在Delphi 6中推出了dbExpress 1.0之后，dbExpress发展得相当迅速。目前，dbExpress已经能够访问许多关系型数据库，包括了Oracle、InterBase、Informix、DB2、MS SQL Server、MySQL、PostgreSQL等，而且在未来，Borland将持续开发Sybase等数据库的dbExpress驱动程序。此外，目前dbExpress已经能够在Windows和Linux平台上执行，未来也将移植到Microsoft的.NET上。

Delphi 7中的dbExpress已经是2.0版了。dbExpress 2.0不但提供了更多的dbExpress驱动程序，修改了许多的bug，还提供了更高的性能。由于dbExpress现在的表现已经超越BDE/IDAPI，而且Borland也已经宣布不再更新BDE/IDAPI，只是提供基本的维护。因此Delphi/Kylix/C++Builder程序员也必须使用dbExpress来开发新的数据库应用程序。

遗憾的是，虽然dbExpress提供了良好的功能和效率，但是目前市面上几乎没有什么书籍专门讨论dbExpress技术，能够帮助程序员善加利用dbExpress的功能开发高效率的数据库应用系统。就是由于这个原因而促使笔者撰写这本书，希望本书帮助读者完全掌握dbExpress技术。本书除了讨论如何使用dbExpress以及Delphi/Kylix的DataSnap技术之外，还详细且深入地讨论了dbExpress的高级技术，包括如何调整dbExpress的性能，dbExpress的实现原理以及dbExpress未来的发展，以便让读者能够充分地利用dbExpress引擎的强大功能，开发出高效率的Delphi/Kylix

45117/01

数据库应用系统。

笔者从Delphi 6.0/Kylix 1.0开始撰写本书，虽然Delphi 7的dbExpress和DataSnap有少许的改善，不过本书的内容仍然适用于Delphi 6/Kylix 2.0以及Delphi 7/Kylix 3.0，此外笔者也在书中说明了Delphi 6和Delphi 7的不同之处。

笔者要感谢所有帮助本书出版的朋友，更要感谢读者的支持。最后希望本书的内容真的能够帮助读者快速进入dbExpress的世界，并且在阅读完毕之后成为精通Delphi/Kylix数据库应用系统设计的高手，谢谢。

李 维

2002年10月于台北，新店

目 录

序

第一部分 dbExpress基本功能篇

第1章 dbExpress组件、概念、技术和应

用程序3

1.1 dbExpress组件3

1.2 建立第一个dbExpress数据库应用程序4

1.3 使用dbExpress的概念13

1.4 使用dbExpress修改数据16

1.4.1 使用TSQLDataSet搭配TDataSet-
Provider和TClientDataSet组件17

1.4.2 使用TSimpleDataSet组件21

1.5 dbExpress驱动程序的设置25

1.6 结论28

第2章 使用dbExpress组件29

2.1 使用TSimpleDataSet组件29

2.1.1 使用动态SQL语句处理数据32

2.1.2 Data和Delta特性42

2.1.3 修改数据——多个数据表45

2.1.4 控制数据访问记录数——Packet-
Records特性50

2.2 DataSnap技术54

2.3 使用TSQLDataSet和TSQLQuery组件59

2.3.1 使用TSQLDataSet组件59

2.3.2 使用TSQLQuery组件63

2.3.3 执行SQL脚本65

2.4 使用TSQLStoredProc组件69

2.5 使用TSQLMonitor组件74

2.6 结论78

第3章 更多的dbExpress技巧81

3.1 数据排序81

3.1.1 dbExpress/DataSnap默认排序82

3.1.2 使用TSQLDataSet的排序特性83

3.1.3 在TSimpleDataSet中进行动态排序86

3.1.4 排序时考虑的因素93

3.2 内存数据表98

3.3 使用计算字段105

3.4 使用Aggregate字段109

3.5 UpdateStatus111

3.6 SavePoint113

3.7 MyBase115

3.8 TField对象的SetText和GetText事件处
理函数118

3.9 结论124

第二部分 dbExpress进阶功能篇

第4章 搜寻数据127

4.1 搜寻数据集数据127

4.1.1 Locate128

4.1.2 Lookup137

4.1.3 过滤器139

4.1.4 Range147

4.2 搜寻方法的比较148

4.3 如何快速地在数据集中搜寻数据152

4.3.1 分析Delphi/Kylix搜寻结果数据集
方法的行为153

4.3.2 数据表包含少量的数据154

4.3.3 数据表包含大量的数据154

4.3.4 快速搜寻数据155

4.4 结论168

第5章 dbExpress的高级技术171

5.1 事务管理171

5.2 数据库的TransIsolation175

5.3 错误处理177

5.4 在COM+中使用dbExpress188

5.5 结论202

第6章 使用dbExpress处理复杂的数据类型	203
6.1 TDataSetProvider组件	203
6.1.1 TDataSetProvider的重要事件处理函数	204
6.1.2 TDataSetProvider的重要特性	208
6.1.3 TDataSetProvider的范例	212
6.2 主从类型的应用	215
6.2.1 使用组件和dbExpress实现主从功能	215
6.2.2 使用程序代码实现主从功能	219
6.3 处理多数据表数据	221
6.4 结论	232
第7章 dbExpress和Web应用程序	233
7.1 Delphi的WebBroker技术	233
7.2 使用dbExpress开发Web应用程序	234
7.3 dbExpress和IntraWeb	262
7.4 结论	271

第三部分 dbExpress高级技术篇

第8章 处理二进制大型数据	275
8.1 处理图形数据	275
8.2 处理JPEG类型的图形数据	279
8.3 如何有效率地处理二进制大型数据	281
8.4 OLE Container类型的数据	290
8.5 结论	298

第四部分 深入的dbExpress实战技术

第9章 dbExpress和元数据	303
9.1 dbExpress和元数据	303
9.2 使用dbExpress处理元数据	304
9.3 观察dbExpress如何使用元数据	321
9.4 结论	331

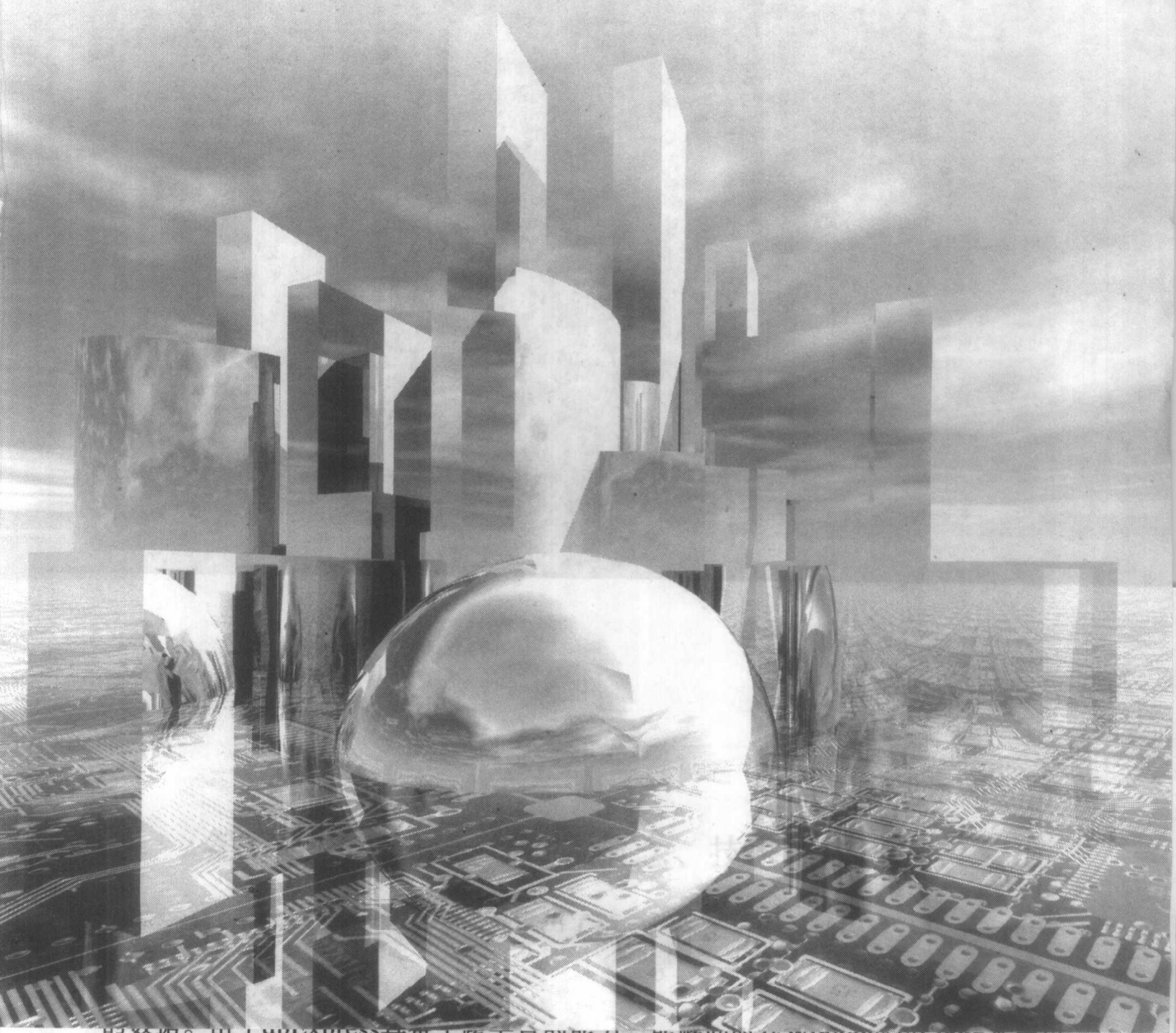
第五部分 性能篇

第10章 开发高效率的数据库应用系统335

10.1	从测试dbExpress、BDE/IDAPI和dbExpress开始	335
10.1.1	观察dbExpress的执行行为之一	338
10.1.2	观察dbExpress的执行行为之二	340
10.2	dbExpress和BDE/IDAPI的性能比较	345
10.2.1	连接数据库的速度	345
10.2.2	访问大量数据的速度	346
10.2.3	添加大量数据	347
10.3	调整dbExpress访问数据的方式	348
10.3.1	调整PacketRecords特性值	348
10.3.2	改变dbExpress处理数据的行为	349
10.4	快速查询数据	354
10.5	Delphi 7之后的TSimpleDataSet	361
10.6	结论	361
第11章	动动脑, 快乐一下	363
11.1	从一个看似简单的场景开始	363
11.2	开始动动脑吧	374
11.3	第二个问题	374
11.4	结论	376
第12章	数据访问技术	379
12.1	dbExpress的发展	379
12.2	BDE的状况	380
12.3	ADO	380
12.4	可选用的数据库	380
12.5	几个数据库及数据访问技术	384
12.6	数据库和组件模型	384
12.7	结论	385
第13章	dbExpress的实现和未来的发展	387
13.1	dbExpress的实现技术	387
13.2	模拟dbExpress的工作流程	399
13.3	有关dbExpress实现的异同	409
13.4	dbExpress未来的实现开发	410
13.5	结论	411

第一部分

dbExpress基本功能篇



的效站。由于dbExpress具备了跨平台的能力，能够同时在Windows和Linux平台以

第1章 dbExpress组件、概念、技术和应用程序

Delphi 6/7的重要功能之一便是推出了新一代跨平台的数据访问引擎——dbExpress。dbExpress是一组新的组件、技术和驱动程序，以允许程序员使用它连接到各种数据源，再配合不同的数据库连接DLL文件，让程序员可以处理后端数据库中的数据。由于dbExpress具备了跨平台的能力，能够同时在Windows和Linux平台以及未来的.NET上使用，更提供了快速的数据处理能力，让程序员能够开发出更有效率的数据库应用程序，因此势必成为以后C++Builder/Delphi和Kylix的核心数据访问技术。

这些新的dbExpress组件除了新进入Delphi世界的程序员需要学习之外，即使是使用Delphi已经有一段时间的有经验的Delphi程序员也需要上手的时间。此外，dbExpress提供了强大的功能，允许Delphi程序员微调它的执行行为，并且访问低层的核心信息。同时dbExpress也能够与DataSnap技术（这是Delphi 6/7中的名称，在旧的Delphi版本中称为MIDAS）结合以便让程序员能够同时开发单机、Briefcase、主从结构和瘦客户类型的数据库应用程序，让程序员能够使用一组组件和技术同时开发数种类型的应用系统。

本书将会讨论dbExpress的所有强大功能，不过万丈高楼平地起，本章将从说明如何使用这些新的dbExpress组件开始，一步一步地带领各位学习到dbExpress最精髓的核心技术。

1.1 dbExpress组件

在Delphi 7中，dbExpress组件组包含了7个组件，这些组件的功能就是让应用程序连接后端数据库，访问数据表中的数据，把修改的数据更新回数据库中以及让程序员观察dbExpress向后端数据库下达的命令等。简单地说，这些组件涵盖了数据库应用程序所有必要的功能。图1-1是dbExpress组件面板中的所有dbExpress组件。



图1-1 Delphi 7的dbExpress组件组

下面的表格概要地说明了每一个dbExpress组件的基本功能，在稍后的章节中将会详细讨论如何使用每一个dbExpress组件。

组件名称	功 能
TSQLConnection	与后端数据库建立连接的组件
TSQLDataSet	可以用来执行SQL语句、执行存储过程或是直接连接数据库中特定数据表的组件。它是一个通用的组件。TSQLDataSet同时具备了类似于TSQLQuery、TSQLStoredProc和TSQLTable组件的能力
TSQLQuery	用来执行SQL语句的组件
TSQLStoredProc	用来执行数据库中的存储过程（Stored Procedure）的组件
TSQLTable	用来连接数据库中特定的数据表的组件，类似于BDE中的TTable组件
TSQLMonitor	可以观看和检视客户端向后端数据源发出的SQL语句的组件。程序员可以使用它来调试或是调整应用程序的性能
TSimpleDataSet	允许dbExpress修改数据的组件，可以结合Delphi的数据感知组件以访问数据

在Delphi 7.0和Kylix 3.0中，dbExpress目前正式支持6种数据库，这些支持的数据库以及对应的dbExpress驱动程序总结在下面的表格中。

数据库名称	dbExpress驱动程序
InterBase 6.5/6.x	DBEXPINT.DLL
DB2 7.2	DBEXPDB2.DLL
Oracle 9i	DBEXPORA.DLL
MySQL 3.23.49	DBEXPMYS.DLL
MS SQL Server 2000	DBEXPMSS.DLL
Informix SE	DBEXPINF.DLL

虽然目前dbExpress只支持6种数据库，但是Borland已经声明将继续提供其他数据库的dbExpress驱动程序，例如Sybase System 12等。在新的dbExpress驱动程序开发完毕之后，Borland会公布在网站上让程序员下载，并且包含在随后版本的Delphi和Kylix中。

学习dbExpress组件最好的方法就是直接使用它来开发数据库应用程序，在下一个小节中，我们将立刻开始学习dbExpress组件组，让程序员快速地学习如何使用这些新的组件开发应用系统。

1.2 建立第一个dbExpress数据库应用程序

现在就让我们快速地使用dbExpress来开发一个数据库应用程序，学习如何使用dbExpress组件来访问数据。要连接数据库并且从其中访问数据，程序员可以使用下面的三个步骤来完成这个工作：

- 1) 使用TSQLConnection组件连接数据库。
- 2) 使用TSQLDataSet组件访问数据。
- 3) 在数据感知组件中显示数据。

现在就让我们一步一步地完成上面的三个步骤。

步骤1：使用TSQLConnection组件连接数据库

首先点击Delphi的File|New|Application菜单建立一个新的Delphi应用程序，接着点击组件面板中的dbExpress选项卡，选择第一个组件TSQLConnection并且将它放入到应用程序的主窗体，如图1-2所示。有了TSQLConnection组件之后，现在我们需要让它连接到数据库服务器，在这个范例中是使用InterBase，读者可以使用其他数据库，例如Oracle或MySQL等。



图1-2 在主窗体中放入TSQLConnection组件以连接数据库

要使用TSQLConnection连接数据库，请使用鼠标双击TSQLConnection，此时会出现TSQLConnection的组件编辑器，如图1-3所示。在图1-3中列出了目前dbExpress内置的连接或是用户新增的连接，由于现在本范例要使用InterBase作为连接的数据库，因此请使用鼠标点击上方的“+”按钮以建立一个新的连接。

此时Delphi会显示图1-4所示的新连接对话框，请在这个对话框中选择使用InterBase驱动程序，并且输入一个连接名称。在这个范例中使用了CHINESEDEMO作为本范例的连接名称。

接着Delphi会显示图1-5所示的对话框，要求输入CHINESEDEMO真正的数据库

路径和名称信息，请按照图1-5那样输入数据库的实际位置。由于CHINESEDEMO使用的InterBase数据库是使用BIG5内码建立的，因此请读者记得将ServerCharSet的特性值改为BIG5。在输入了数据库实际位置之后，读者可以点击对话框上方的“~”按钮，以测试是否可以正确地连接到数据库。最后确定一切正确之后，点击OK按钮以完成设置TSQLConnection组件的步骤。

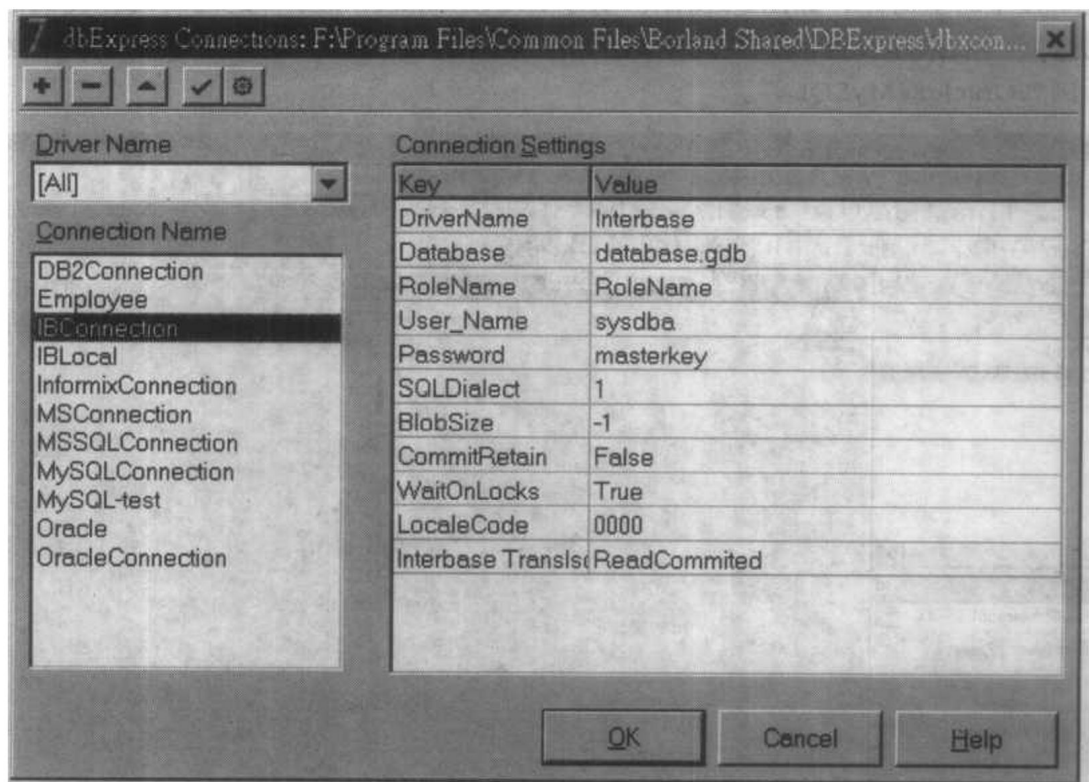


图1-3 TSQLConnection组件的组件编辑器

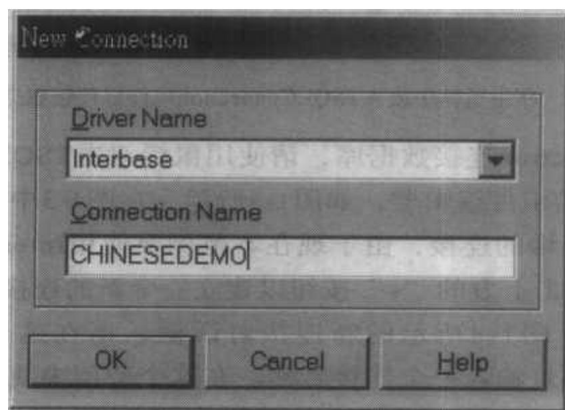


图1-4 dbExpress的新数据库连接对话框

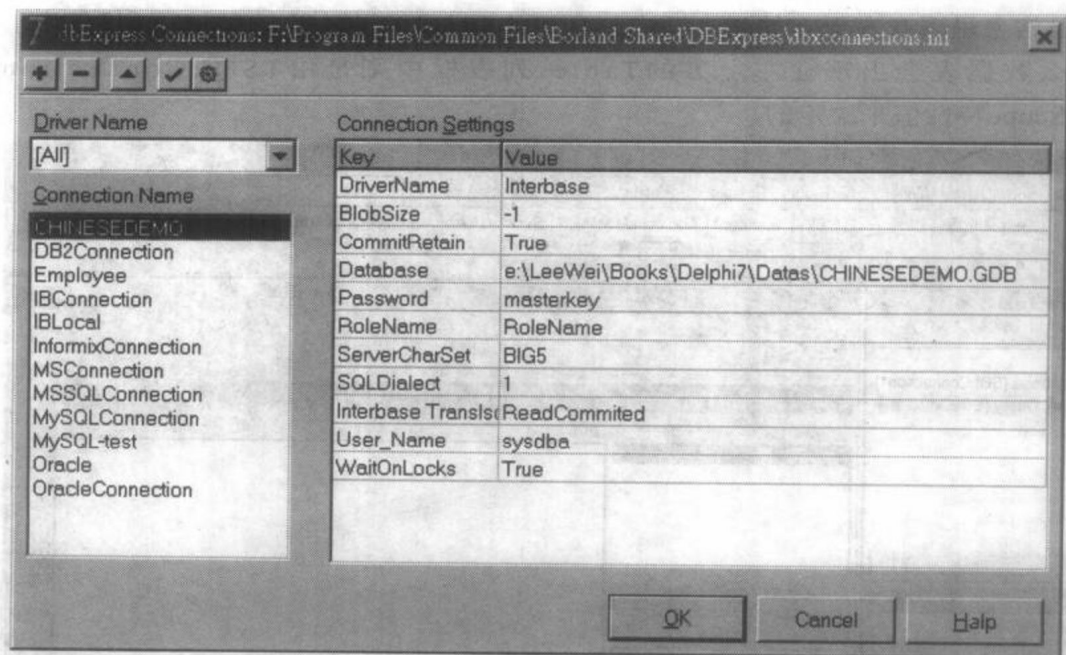


图1-5 输入CHINESEDEMO要连接的InterBase数据库

读者可以在本书的光盘中找到CHINESEDEMO.GDB这个InterBase数据库。

如果读者仔细观察图1-5所示的对话框，便会发现刚才设置的CHINESEDEMO这个InterBase连接信息事实上是存储在\Borland Shared\DBExpress目录下的dbxconnections.ini文件中，这是一个文本文件，读者也可以使用文字编辑器，例如NotePad或是Delphi的编辑器来修改其中的内容。

现在请点击对象检视器，设置TSQLConnection的LoginPrompt特性值为False，以避免出现登录对话框，最后再把Connected特性值设置为True。如此一来我们就成功地连接到InterBase服务器了（当然，读者的InterBase必须正在执行）。

步骤2：使用TSQLDataSet组件访问数据

现在再从dbExpress选项卡中选择第二个组件TSQLDataSet，并且将它放到主窗体中。先在对象检视器中将它的SQLConnection特性值设置为步骤1放入的SQLConnection1，再点击它的CommandText特性值。此时Delphi便会显示CommandText特性值的特性值编辑器，让程序员可以使用可视化的方式下达SQL命令。图1-6便是激活CommandText特性值编辑器的画面。

当CommandText特性值编辑器出现时，它会自动地从连接的数据库中取得所有目前可以被看到的数据表名称，放入到左上方的Tables列表框中，而把选择的数据表

的字段信息呈现在左下方的列表框中，真正的SQL命令则在右边的Memo控件中。至于什么数据表会出现在左上方的Tables列表框中则是由TSQLConnection的TableScope特性值来决定的。

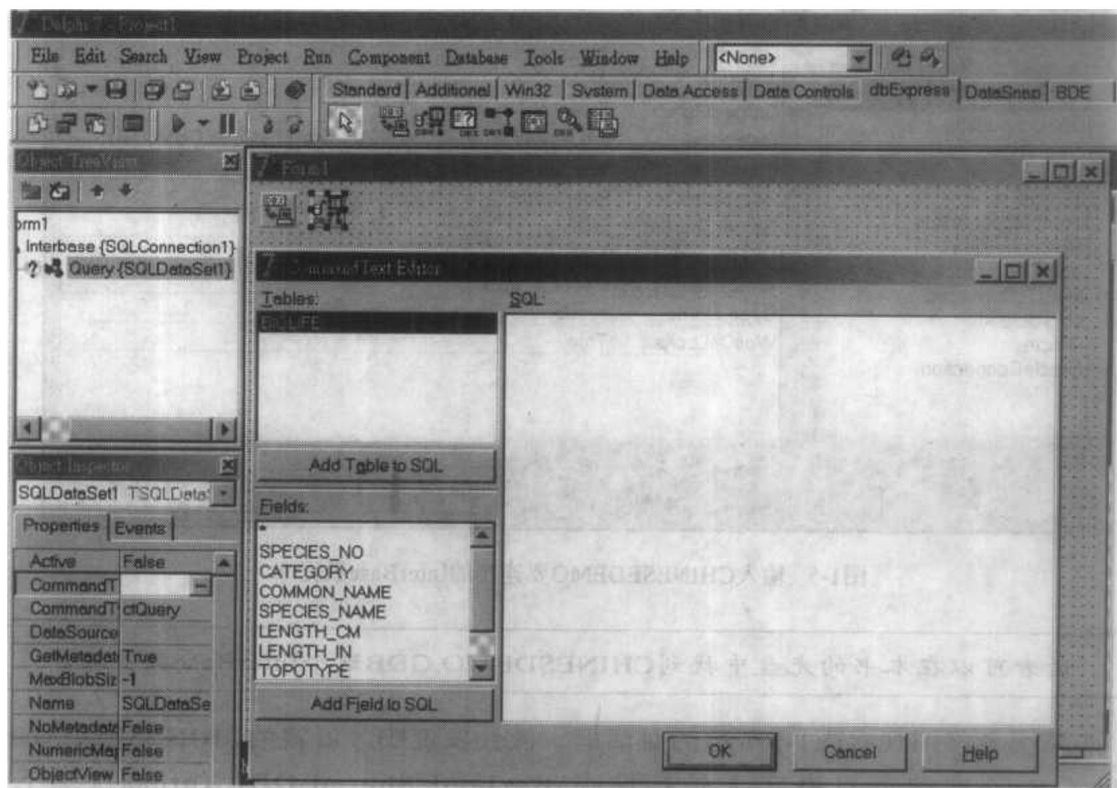


图1-6 TSQLDataSet的CommandText特性值编辑器

由于CHINESEDEMO中只有一个数据表BIOLIFE，因此请使用鼠标点击左上方的BIOLIFE，再点击左下方的“*”以代表要从BIOLIFE数据表中取得所有字段的数据，如此一来CommandText特性值编辑器便会在右边的SQL窗口中自动产生Select * from BIOLIFE的SQL命令，如图1-7所示。

请点击OK按钮，Delphi便会把刚才完成的SQL命令存储在TSQLDataSet组件的CommandText特性值中。接着请在主窗体中放入一个TDataSource组件，将它的数据Set特性值设置为刚才的TSQLDataSet组件，再放入一个TDBNavigator组件，将它的数据Source特性值设置为刚放入的TDataSource组件。

现在我们就可以通过把TSQLDataSet的Active特性值设置为True将数据从BIOLIFE数据表取到应用程序中。接着让我们激活TSQLDataSet的字段编辑器，把代表BIOLIFE每一个字段的字段对象加入应用程序中。把字段对象加入应用程序中的好处是程序员可以使用Delphi的对象检视器来设置字段对象的特性，例如字段显示的中文名称，或是设置字段显示的栏宽等。

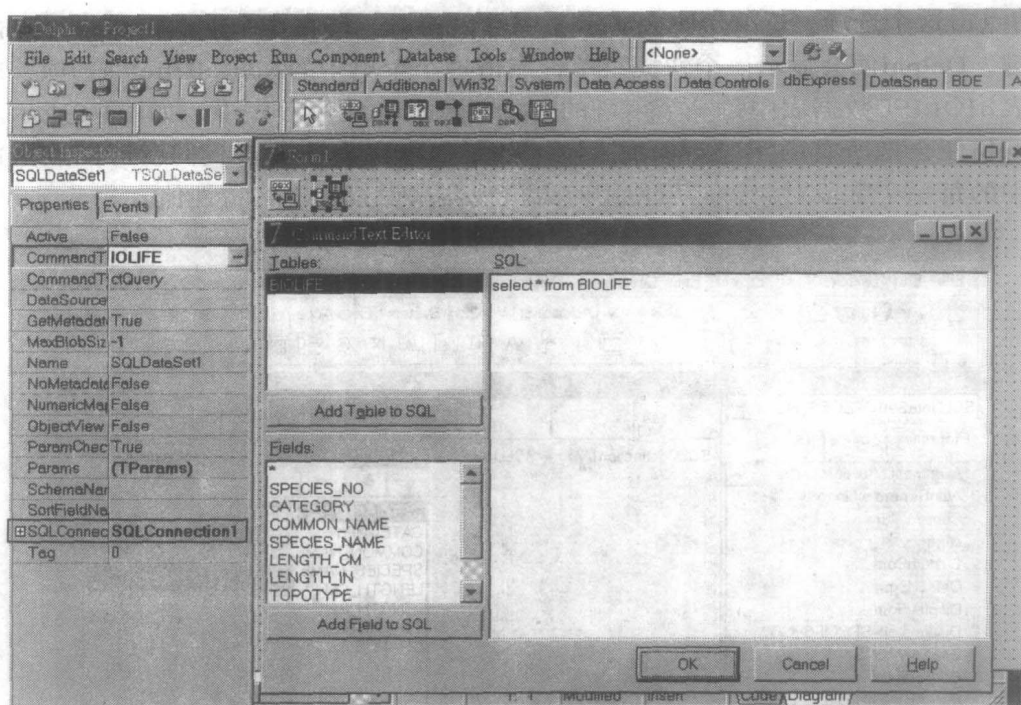


图1-7 使用CommandText特性值编辑器完成SQL命令

请点击主窗体中的TSQLDataSet组件，按下鼠标右键，Delphi会显示一个快捷菜单，菜单中的第一个选项Fields Editor...便可以激活字段编辑器，如图1-8所示。

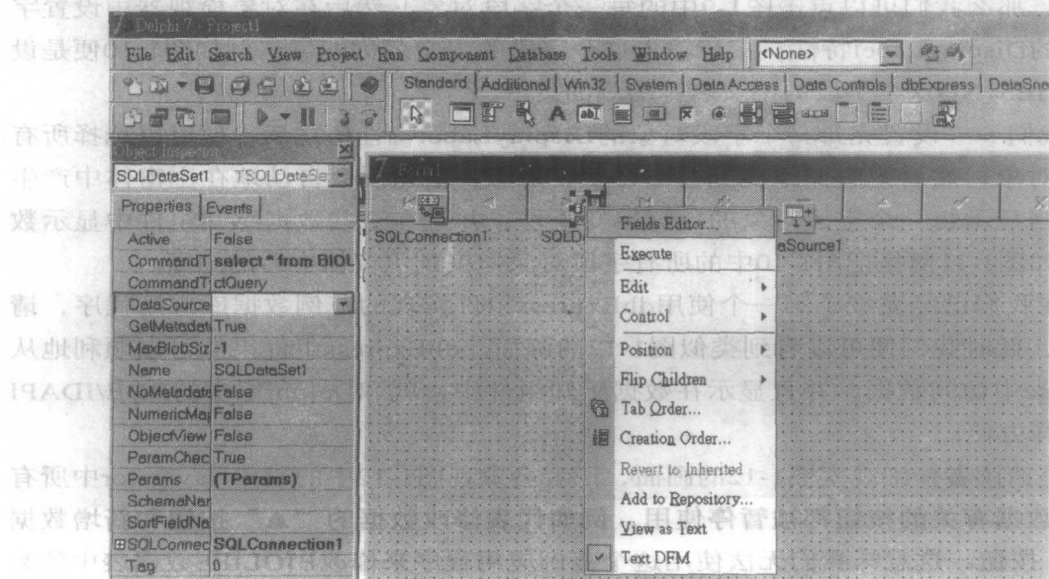


图1-8 激活TSQLDataSet的字段编辑器

请点击这个选项，Delphi便会显示一个空白窗口，请在这个空白窗口中再点击鼠标右键，Delphi便会显示一个快捷菜单，点击这个菜单中的Add all fields选项，以便将代表BIOLIFE数据表中每一个字段的字段对象加入应用程序中，如图1-9所示。接着Delphi便会把所有的字段对象加入刚才的空白窗口中，现在我们就可以点击这个窗口中的每一个字段对象，然后使用对象检视器来设置它们的特性值。

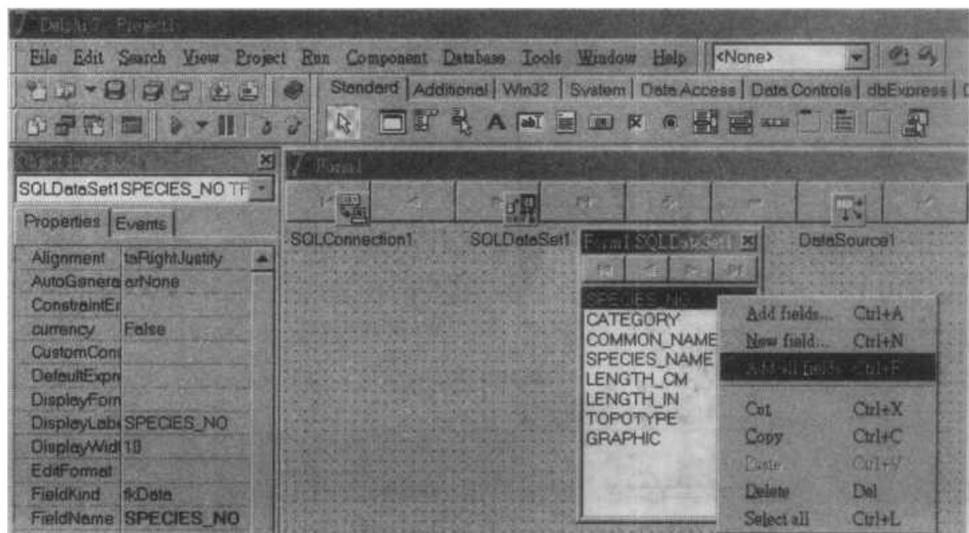


图1-9 加入所有的字段对象

现在我们希望应用程序在执行时显示中文的字段名称，而不是数据表中字段的英文名称，那么我们可以点击图1-9中的每一个字段对象，然后在对象检视器中设置字段对象的DisplayLabel特性，接着输入这个字段的中文名称即可。例如图1-10便是设置SPECIES_NO这个字段的中文名称的画面。

在我们一一设置完每一个字段对象的DisplayLabel特性值之后，便可以选择所有的字段对象，然后把它们拖曳到主窗体中，那么Delphi便会自动地在主窗体中产生能够适当显示每一个字段对象的数据感知组件，并且在这些数据感知组件中显示数据。例如图1-11便是把图1-10中的所有字段对象拖曳到主窗体中后的画面。

现在我们已经完成了第一个使用dbExpress组件实现的范例数据库应用程序，请执行它，此时我们便可以看到类似图1-12的画面。dbExpress组件果然可以顺利地从中访问数据，并且显示在数据感知组件中，就和Delphi原本的BDE/IDAPI组件一样方便。

但是请读者仔细观察图1-12的画面，读者会发现图1-12中的TDBNavigator中所有与修改数据有关的按钮都被暂停使用，例如代表修改数据的“▲”按钮和新增数据的“+”按钮。这意味我们无法使用这个范例应用程序来修改BIOLIFE数据表中的数据，也代表由dbExpress组件开发的数据库应用程序是无法修改的。