

Delphi 7

数据库开发

完全手册

张增强
刘 成 编著



清华大学出版社

Delphi 7 数据库开发完全手册

张增强 刘成 编著

北京·清华大学出版社

(京)新登字 158 号

内 容 简 介

Delphi 是一个极为优秀的 RAD 开发工具, 在开发数据库方面有强大的功能。新推出的 Delphi 7 更是在业内产生了强大的冲击, 具有和 .NET 技术一比高低的潜力。

本书全面系统地讲述了 Delphi 开发数据库的编程方法, 深入探讨了 Delphi 7 提供的最新技术。内容包括: 数据库的基础知识、传统的 BDE 数据引擎和数据库开发环境的配置、数据集和数据感应组件的使用、ADO 和 dbGO 技术, 以及 dbExpress 数据引擎和 XML 技术。最后还详细讲解了 Delphi 7 开发 Web 数据库程序的 WebBroker 和 WebSnap 技术。

本书覆盖面广, 并具有一定的深度, 面向中级的 Delphi 用户, 十分适合于那些能够设计一般窗体程序, 正在学习使用 Delphi 进行数据库开发的读者。另外, 也可以作为培训教材及参考手册。

版权所有, 盗版必究。

本书封面贴有清华大学出版社激光防伪标签, 无标签者不得销售。

图书在版编目 (CIP) 数据

Delphi 7 数据库开发完全手册/ 张增强, 刘成编著. —北京:

清华大学出版社, 2003

ISBN 7-302-06435-0

I. D… II. ①张…②刘… III. 软件工具—程序设计—技术手册 IV. TP311.56-62

中国版本图书馆 CIP 数据核字 (2003) 第 018508 号

出版者: 清华大学出版社 (北京清华大学学研大厦, 邮编 100084)

<http://www.tup.com.cn>

印刷者: 北京市耀华印刷有限公司

发行者: 新华书店总店北京发行所

开 本: 异 16 开 印张: 29.625 字数: 647 千字

版 次: 2003 年 4 月第 1 版 2003 年 4 月第 1 次印刷

书 号: ISBN 7-302-06435-0/TP · 4849

印 数: 0001~5000

定 价: 42.00 元

前 言

Delphi 是 Borland 公司开发的可视化的快速应用程序开发工具(RAD, Rapid Application Development), 它基于视窗平台提供了强大的 VCL (可视化组件库) 组件, 不断改善的集成开发环境 (IDE), 快速的编译运行能力和良好的稳定性, 使得它在激烈的市场竞争中获得了许多程序员的钟爱, 尤其是在数据库开发方面是相当优秀的。

2001 年 5 月, Borland 公司推出了 Delphi 6。Borland 对 Delphi 6 进一步改善了开发环境, 对 XSL、SOAP、Web Service 等重要新兴行业标准都提供了支持。2002 年 8 月, Borland 公司推出了 Delphi 7, Delphi 7 是第一个支持 Windows.NET 技术的开发工具, 同时在数据库开发方面进行了改进, 增加了一系列的组件以支持网络和数据库开发。

Borland 软件公司中国大陆及香港地区总经理黄锡强先生说: “不论是初学 Windows 程序设计的学生或是开发关键性任务、大型应用系统的企业开发人员, Delphi 完整的产品线都能满足不同类型开发人员的需求, 我们希望通过 Delphi 7 为开发人员提供更方便的技术和服务, 进一步扩大 Borland 在中国的业务。”新的 Delphi 框架包括 BizSnap、WebSnap 和 DataSnap, 它们将通过一整套完全集成, 且支持 Web Service 的可视化工具、高级编译技术和可重新使用的软件组件, 帮助用户构建支持 Web Service 规范的服务器端和客户端应用程序。Delphi 7 的设计目标就是要通过简化 B2B 的集成过程, 加快企业产品进入市场的时间。

本书将专注于 Delphi 7 在开发数据库方面的强大功能。新的 Delphi 7 不仅修改了 Delphi 6 中 ADO 数据访问技术中的 bug, 还推出了数据库访问技术 dbExpress, 其对数据轻型化和快捷的访问能力, 大大增强了 Delphi 开发数据库的功能。另外, dbExpress 也是 Linux 平台上的 Kylix 数据存取引擎。此外, Delphi 7 整合了 MIDAS 和 WebBroker 推出的 DataSnap 和 WebSnap, 提供了开发多层分布式系统和 Web 程序的新环境。另外, 本书还将详细介绍 Delphi 7 对 XML 的支持, 通过本书的学习, 用户将能够轻松处理 XML 文件。没有 Web 服务的数据库系统也是不完善的, 本书还将介绍 Delphi 7 提供的 Web 程序开发技术, 尤其是 Delphi 7 提供的 WebSnap 技术。

本书的第 1、2、3 章介绍 Delphi 开发数据库的基本知识, 包括 BDE (Borland 数据库引擎) 的配置、工具的使用、数据库的一般知识; 第 4、5、6、7、8 章详细地介绍了 Delphi 7 提供的访问数据库的机制, 包括 BDE、ADO、dbExpress, 还有 MIDAS 技术, 这是本书最核心的内容; 第 9、10 章介绍了快速创建报表方法和数据仓库的使用; 第 11、



12、13、14、15 章介绍 Web 编程的相关内容，包括多层分布式系统 XML、WebBroker、WebSnap 等，其中还有 ASP 的部分内容，这部分将会升华用户对 Delphi 7 的掌握，有助于领略 Delphi 在扩展数据库系统、开发电子商务系统方面的卓越技术。我们希望本书能够帮助读者快速领悟和使用 Delphi 7 开发各种数据库应用程序。本书实例的调试环境为 Delphi 7 企业版。

本书由张增强和刘成共同执笔编写。此外，张忠东、李晓裳、范智育、王宏生、李光龙、王瑾、吴浩、李炎、刘伟、刘华刚、朱峰、赵晓燕、李晓、马苍、郝春容、韦勇、成美华、萧峰、李菊、张浩然、李欣、张浩、李想、朱大成、周卫、赵中伟、杨竞锐、王贵新、张诚华、朱丽云、程松、李毅、赵郡超、孙名等同志在整理材料方面给予了作者很大的帮助，在此表示衷心地感谢。

由于编者的水平有限，书中定会存在很多的疵漏，恳请广大读者斧正。

编者

2002 年 10 月

目 录

第 1 章 Delphi 7 概述 1

- 1.1 Delphi 简介 1
- 1.2 Delphi 的新特征 2
 - 1.2.1 编译器功能增强 2
 - 1.2.2 新增了两个网络协议组件组 Indy Intercepts 和 Indy I/O Handlers 3
 - 1.2.3 可定制化的 View 视窗 4
 - 1.2.4 可定制的代码完成功能 4
 - 1.2.5 更加强大的网络开发功能 5
 - 1.2.6 跨平台开发库 (CLX) 5
 - 1.2.7 新增了 Rave 报表制作工具 7
 - 1.2.8 数据库功能增强 7
 - 1.2.9 支持 Microsoft 公司的 .NET 技术 8
 - 1.2.10 对 Windows XP 风格的支持 8
- 1.3 Delphi 访问数据库的机制 8
 - 1.3.1 BDE 8
 - 1.3.2 ADO 9
 - 1.3.3 dbExpress 10
- 1.4 小结 10

第 2 章 数据库基础 11

- 2.1 数据库简介 11
 - 2.1.1 关系数据模型的基本概念 12
 - 2.1.2 数据库保护的基本概念 12
- 2.2 SQL 语言简介 13
 - 2.2.1 运算符和函数 14
 - 2.2.2 数据类型 15
 - 2.2.3 数据定义语言 (DDL) 的实现 15
 - 2.2.4 数据操作语言 (DML) 的实现 16

- 2.2.5 数据库管理和控制的实现 18
- 2.3 Database Desktop 的使用 19
 - 2.3.1 界面介绍 19
 - 2.3.2 建立表结构 21
 - 2.3.3 表窗口操作 28
 - 2.3.4 查询窗口操作 29
 - 2.3.5 SQL 编辑窗口操作 31
- 2.4 选择和建立数据库 33
 - 2.4.1 选择数据库 33
 - 2.4.2 设计数据库的步骤 35
- 2.5 典型的数据库 35
 - 2.5.1 Oracle 数据库 35
 - 2.5.2 SQL Server 2000 36
 - 2.5.3 MySQL 36
 - 2.5.4 InterBase 36
- 2.6 桌面数据库程序示例 37
 - 2.6.1 一个数据表的浏览程序 37
 - 2.6.2 建立 Master/Detail 关系的桌面数据库程序 38
 - 2.6.3 通过向导 Form Wizard 建立 Master/Detail 41
- 2.7 小结 45

第 3 章 数据库环境的设置和 BDE 工具

- 46
- 3.1 BDE 简介 46
- 3.2 BDE Administrator 47
 - 3.2.1 界面介绍 47
 - 3.2.2 建立一个本地数据库别名 48

3.2.3 建立网络数据库别名	49	4.4.2 比较 TQuery 和 TTable	104
3.2.4 用 ODBC 连接 Access 数据库	51	4.4.3 返回可以修改的数据集	105
3.3 SQL Explorer	52	4.4.4 设置静态的 SQL 语句	106
3.4 SQL Monitor	54	4.4.5 用 TQuery 建立一个数据表的浏览 程序	107
3.5 Data Pump	55	4.4.6 设置动态的 SQL 语句	108
3.6 小结	58	4.4.7 Prepared 属性、Prepare 方法和 UnPrepare 方法	110
第 4 章 数据集	59	4.4.8 设置查询参数	110
4.1 TDataSet	59	4.4.9 一个数据库浏览程序	113
4.2 深入研究 TTable	65	4.4.10 从数据集中获取 SQL 参数	117
4.2.1 TBDEDataSet	66	4.5 深入研究 TStoredProc	118
4.2.2 TDBDataSet	67	4.5.1 TStoredProc 组件的属性、方法和 事件	119
4.2.3 TTable 的属性和方法	68	4.5.2 TStoredProc 的使用步骤	119
4.2.4 打开和关闭数据集	70	4.5.3 使用 TParam 类	120
4.2.5 连接一个数据表的实例	71	4.5.4 使用 TStoredProc 调用存储过程的 实例	122
4.2.6 建立数据表	72	4.5.5 使用 TQuery 组件调用存储过程	125
4.2.7 浏览数据表	78	4.6 小结	126
4.2.8 书签	81	第 5 章 数据访问组件	127
4.2.9 查询记录	82	5.1 数据访问组件简介	127
4.2.10 检索数据表	84	5.2 TDataSource 组件	128
4.2.11 记录的编辑	86	5.3 TDatabase 组件	129
4.2.12 数据调和	89	5.3.1 TDatabase 组件的属性、方法和 事件	129
4.2.13 过滤器	89	5.3.2 建立应用程序级别的数据库别名	131
4.2.14 TTable 的其他功能	91	5.3.3 实现登录过程	133
4.3 深入研究 TField	93	5.4 TSession 组件	137
4.3.1 访问字段的值	94	5.4.1 TSession 组件的属性、方法和事件	138
4.3.2 字段名和编号的转换	96	5.4.2 处理别名	140
4.3.3 字段编辑器	98		
4.3.4 建立计算字段	99		
4.3.5 建立查询字段	99		
4.3.6 建立 Aggregate 字段	100		
4.3.7 字段的其他属性	101		
4.4 深入研究 TQuery	103		
4.4.1 TQuery 组件的属性、方法和事件	103		

5.4.3 处理驱动	143	7.2 建立一个 ADO 应用程序	178
5.4.4 获取数据库的对象	144	7.3 建立 ADO 连接字符串	181
5.4.5 控制数据库组件和连接	144	7.3.1 使用 ConnectionString 编辑器	181
5.4.6 管理 Paradox 表的密码	145	7.3.2 建立和使用 UDL 文件	183
5.4.7 一个浏览 BDE 的 Alias 和 Driver 的实例	146	7.3.3 手工编辑 ConnectionString	183
5.5 TBatchMove 组件	148	7.4 TADOConnection 组件	184
5.5.1 TBatchMove 组件的属性和方法	149	7.4.1 TADOConnection 组件的属性、 方法和事件	184
5.5.2 使用 TBatchMove 组件	150	7.4.2 处理 ADO 对象	186
5.6 TUpgradeSQL 组件	151	7.4.3 连接控制	188
5.6.1 TUpgradeSQL 组件的属性和方法	151	7.4.4 在应用程序中建立事件的日志	190
5.6.2 使用 TUpgradeSQL 组件	152	7.4.5 获取数据库的信息	192
5.7 TNestedTable 组件	154	7.4.6 事务处理	192
5.8 数据模块 DataModule	154	7.4.7 直通查询方式	194
5.9 小结	157	7.5 TADOCommand 组件	195
第 6 章 数据感应组件	158	7.5.1 TADOCommand 组件的属性和 方法	196
6.1 数据感应组件概览	158	7.5.2 直通模式	196
6.2 数据感应组件的基本特征	159	7.5.3 TParameter 和 TParameters	198
6.3 TDBGrid 组件	160	7.6 TADODataSet 组件	200
6.3.1 TDBGrid 的属性和设计	160	7.6.1 TADODataSet 组件的属性、方法 和事件	200
6.3.2 TDBGrid 组件的事件和使用	164	7.6.2 确定 TADODataSet 执行的操作	205
6.4 TDBNavigator 组件	166	7.6.3 批处理 (BatchUpdate)	205
6.5 其他的数据感应组件	167	7.6.4 记录集的操作	207
6.6 小结	171	7.6.5 事件的使用	210
第 7 章 ADO 与 dbGO	172	7.6.6 数据的调和	212
7.1 ADO 技术	172	7.7 TADOQuery 组件和 TADOTable 组件	212
7.1.1 ADO 的发展	172	7.8 TRDSCONNECTION 组件	213
7.1.2 ADO 对象	174	7.9 公文包模型	216
7.1.3 使用 ADO	175	7.10 一个 dbGO 应用程序	221
7.1.4 ADO 的游标 (Cursor)	175	7.11 小结	230
7.1.5 ADO 组件	177		

第 8 章 dbExpress	231	9.6 Rave 组件	270
8.1 Delphi 7 和 dbExpress	231	9.6.1 Rave 组件简介	270
8.1.1 dbExpress 简介	231	9.6.2 使用 Rave 组件	273
8.1.2 dbExpress 的接口	232	9.7 小结	278
8.1.3 dbExpress 的数据类型	232	第 10 章 Decision Cube 数据仓库 ..	279
8.1.4 Delphi 7 的 dbExpress 组件	233	10.1 Decision Cube 简介	279
8.2 TSQLConnection 组件	233	10.2 TDecisionQuery 组件	280
8.2.1 TSQLConnection 组件的属性、 方法和事件	233	10.3 TDecisionCube 组件	282
8.2.2 连接的配置	235	10.4 TDecisionSource 组件	286
8.2.3 控制执行的语句数目	237	10.5 TDecisionGrid 组件	288
8.2.4 事务处理	238	10.6 TDecisionPivot 组件	292
8.2.5 执行 SQL 命令	240	10.7 TDecisionGraph 组件	293
8.3 SQLDataSet 组件	241	10.8 小结	295
8.3.1 SQLDataSet 组件的属性、方法 和事件	241	第 11 章 多层分布式应用程序系统 .	296
8.3.2 SQLDataSet 组件的使用	243	11.1 数据库应用程序的发展	296
8.3.3 项目 1: 浏览单向数据集	243	11.1.1 从桌面数据库程序到客户/服务器	296
8.3.4 项目 2: 缓存单向数据集	245	11.1.2 比较客户/服务器和桌面数据库 开发	297
8.4 TSimpleDataSet 组件	249	11.1.3 客户/服务器系统开发	298
8.5 小结	251	11.1.4 多层分布式系统	299
第 9 章 快速创建报表	252	11.2 开发多层分布式应用程序系统	300
9.1 QReport 概述	252	11.2.1 多层分布式应用程序体系结构	300
9.2 报表向导	253	11.2.2 多层分布式应用程序系统开发 示例	302
9.3 报表模板	255	11.3 应用程序服务器组件 TRemoteDataModule	307
9.3.1 QuickReport Labels 模板	256	11.3.1 TRemoteDataModule 的基本属性 和方法	307
9.3.2 QuickReport List 模板	258	11.3.2 为 RemoteDataModule 添加属性 和方法	308
9.3.3 QuickReport Master/Detail 模板	258	11.3.3 Instancing (实例) 模型	310
9.4 QReport 组件介绍	260	11.3.4 Threading Model (线程模型) ...	311
9.4.1 TQuickRep 组件	260		
9.4.2 TQRBand 组件	264		
9.4.3 TQRExpr 组件	265		
9.4.4 其他 QReport 组件的介绍	266		
9.5 安装 QReport 组件面板	269		

11.3.5 配置 DCOM 应用程序服务器.....	312
11.3.6 删除一个 DCOM 服务器	313
11.4 数据连接组件	313
11.4.1 TDispatchConnection 组件	313
11.4.2 TDCOMConnection 组件.....	314
11.4.3 TSocketConnection 组件.....	314
11.4.4 TWebConnection 组件	315
11.4.5 TCORBAConnection 组件.....	316
11.4.6 TSimpleObjectBroker 组件.....	316
11.4.7 TSharedConnection 组件	317
11.4.8 TLocalConnection 组件	317
11.4.9 开始连接	318
11.4.10 断开连接	318
11.5 小结	318

第 12 章 分布式应用程序中的数据

数据集.....	319
12.1 TDataSetProvider 组件	319
12.1.1 TDataSetProvider 组件的属性和 方法	319
12.1.2 TDataSetProvider 组件提交更新 的方式	321
12.1.3 TDataSetProvider 组件定位记录 的方式	321
12.1.4 TDataSetProvider 组件的事件和 使用.....	323
12.2 TClientDataSet 组件	327
12.2.1 TClientDataSet 组件的属性、 方法和事件.....	328
12.2.2 使用 TClientDataSet 组件的事件	331
12.2.3 数据的封装机制	334
12.2.4 获取本地的数据	335
12.2.5 获取应用程序服务器的数据	337

12.2.6 调用 IAppServer 接口.....	338
12.2.7 TClientDataSet 作为嵌套数据集	339
12.2.8 传递 SQL 命令.....	340
12.2.9 调和数据	340
12.3 小结	342

第 13 章 Delphi 7 对 XML 的支持 ... 343

13.1 Delphi 7 和 XML	343
13.2 XML Mapper 工具	344
13.2.1 建立对应 XML 结构的系列表.....	344
13.2.2 XML Mapper 工具的使用	345
13.3 TXMLTransform 组件.....	353
13.3.1 TXMLTransform 组件的属性.....	353
13.3.2 TXMLTransform 组件的方法.....	354
13.3.3 TXMLTransform 组件的事件.....	354
13.3.4 TXMLTransform 组件的使用.....	356
13.4 TXMLTransformProvider 组件.....	359
13.4.1 TXMLTransformProvider 组件的 属性.....	359
13.4.2 TXMLTransformProvider 组件的 方法	359
13.4.3 TXMLTransformProvider 组件的 使用	359
13.4.4 浏览和修改 XML 文件的程序.....	360
13.5 XMLTransformClient 组件	363
13.5.1 XMLTransformClient 组件的属性	363
13.5.2 XMLTransformClient 组件的方法	363
13.5.3 使用 XMLTransformClient 组件	364
13.6 XMLDocument 组件.....	368
13.6.1 XMLDocument 组件的属性.....	368
13.6.2 XMLDocument 组件的方法.....	369

13.6.3 XMLDocument 组件的事件.....	370	14.4.6 进行安全验证	409
13.6.4 使用 DOM 接口操作 XML 文件	371	14.4.7 设置 Cookie.....	410
13.6.5 使用 IXMLNode	372	14.5 页面生成器组件	413
13.6.6 使用 IXMLNode 转变数据集为 XML 文件	374	14.5.1 TPageProducer 组件.....	413
13.7 利用向导生成和使用 XML 接口.....	376	14.5.2 TDataSetPageProducer 组件	415
13.7.1 生成 XML 文件接口	376	14.5.3 TDSTableProducer 组件.....	419
13.7.2 使用生成的 XML 文件接口	378	14.6 使用 Web App Debugger.....	421
13.8 XML 简介	381	14.7 开发和使用 Apache Module	425
13.8.1 XML 语法简介	382	14.8 小结	427
13.8.2 XML Schemas	384	第 15 章 WebSnap	428
13.9 小结	385	15.1 Active Server Pages.....	428
第 14 章 WebBroker	386	15.1.1 Delphi 和 ASP	428
14.1 Web 程序.....	386	15.1.2 利用 Delphi 建立 ASP 对象	429
14.1.1 网页.....	386	15.2 开发 WebSnap 程序	432
14.1.2 CGI.....	388	15.2.1 一个最简单的 WebSnap 程序	433
14.1.3 ISAPI.....	390	15.2.2 WebSnap 程序的组件和脚本	435
14.1.4 Apache Modules	391	15.2.3 设计多页面	438
14.1.5 使用 Web 服务器	391	15.2.4 Adapter 组件	442
14.2 WebBroker 概述.....	392	15.2.5 使用 Adapter 和 AdapterPage-Producer 设计页面内容	445
14.3 DB Web Application 向导.....	394	15.2.6 硬编码和脚本代码	449
14.4 TWebModule 组件.....	399	15.2.7 WebSnap 和数据库程序	450
14.4.1 TCustomWebDispatcher 组件	400	15.2.8 建立 Master/Detail 关系的 WebSnap 程序	456
14.4.2 动作项 (TWebActionItem)	401	15.2.9 会话管理	458
14.4.3 返回内容	403	15.2.10 安全认证	460
14.4.4 传送流	405	15.3 小结	464
14.4.5 获取客户输入的数据	407		

第 1 章 Delphi 7 概述

Delphi 是一个强大的全方位面向对象的可视化开发工具，本章将介绍 Delphi 7 的 IDE 和它的新特点，以及 Delphi 对数据库的支持和新功能。

1.1 Delphi 简介

Delphi 程序使用的是 Object Pascal 语言。Borland 推出 Delphi 其核心本身就是 Pascal 编译器。Pascal 语言向来以编译速度快著称，当初推出的 Turbo Pascal 编译器，便具有了稳定性、快速编译的能力。即使在 Borland 不断升级其编译器功能时，其编译速度仍然是快速而稳定的，这在以往一直是微软的 Visual C++ 的痛处。另外 Object Pascal 语言在稳定性和复杂性方面作了平衡，限制了一些可用的功能，如多重继承，使得语言本身逻辑性好、组织严谨。

Delphi 是面向对象的可视化开发工具，它提供了一个丰富强大的可视化组件库（VCL, Visual Component Library），这些组件也是用 Object Pascal 语言写的，本身就是对象，拥有属性、事件和方法。VCL 组件体现了面向对象技术，它封装了系统行为和许多底层的实现方法。使得程序开发者可以快速地获取对象，修改属性，建立事件和进行程序设计。当然，程序员也可以自己编写组件。

由于 Delphi 的可视化开发环境的编辑器和窗体设计器协同工作，这使得程序员在设计窗体实体时，后台会同步地在编辑器中自动地生成组件的代码。另外，Delphi 采用了事件驱动的程序运行机制，当一个事件作用在目标对象上时，就可以触发一个事件处理程序，或是发布一个消息。程序员通过设置对象的属性、方法和消息处理过程，就可以构造出响应程序来。因此有人说，Delphi 的程序结构像一套相互作用的组件对象，程序员要做的就是搭建组件，然后用代码把组件对象粘合起来，当然这只是 Delphi 简单程序设计的模式。可以说 Delphi 几乎实现了面向对象编程的所有技术。

从 Delphi 1 到 Delphi 7，Delphi 在同 Visual Basic 和 PowerBuilder 的竞争中，一直保持着自己的优势，获得了许多程序员的钟爱。Borland 不断致力于改善开发环境，使 Delphi 更易于使用，在技术上也保持着领先。Delphi 4 发展了 MIDAS、DCOM 和 CORBA，Delphi 5 引入了技术支持 ADO、XML。Delphi 7 则是完全支持所有重要的新兴行业标准（XML、SOAP、WSDL 和 XSL）的惟一 RAD（快速应用程序开发）环境，它在提供企业和 Web 开发人员所



需的升级性与可靠性的同时，又支持以新兴的 Web Services 为基础的供应商平台，如 Microsoft .NET、BizTalk 以及 Sun Microsystems 的 ONE（开放网络环境）等。

Delphi 7 是正当 Microsoft 全力发展 .NET 虚拟环境，而 Borland 本身也往 Linux 平台上开发 RAD 工具 Kylix 的时候，由 Borland 视窗开发小组所研发的原生视窗开发工具。有人不禁担忧 VC++ 和 Delphi 的前途问题。所以 Delphi 7 的出现确实是值得许多视窗程序员好好研究的。

Borland 公司为了满足不同层次用户的需要，历来把 Delphi 分为三个版本，如 Delphi 7 就分为 Delphi 7 标准版、Delphi 7 专业版、Delphi 7 企业版。

个人版（Delphi 7 Personal）是入门级的学习版本，功能有限，能编写高性能非商务系统的程序。

专业版（Delphi 7 Professional）在标准版的基础上加入了更多的功能，能够实现跨平台，与 Linux 下的 Kylix 程序员共同开发。共享程序设计的技巧和代码库。

企业版（Delphi 7 Enterprise）提供了完整的功能和技术，包括适合于建立网络服务器程序，电子商务系统的最新技术，如 Web Services、BizSnap WebServices，通过建立 XML/SOAP Web Services 和建立集成有用用户和厂商的商务连接，简化了开发电子商务的平台。作者建议使用该版本开发数据库工程和 Web 程序。

本书的实例都是在 Delphi 7 企业版上设计和调试过的。

1.2 Delphi 的新特征

Borland 公司每推出 Delphi 一个新版本，技术上都会体现在以下三个方面：

- 改善开发环境和编译器，使得 Delphi 更加适合于快速开发，增强运行的性能。
- 修正了一些 bug，稳定原有的技术。
- 支持新技术，保持自己的技术优势。

Delphi 7 在前两项的技术提高是肯定的，对于第三项则是关系到 Delphi 产品系列的发展策略的问题，更让人关注。下面介绍 Delphi 7 几个显著的新特征。

1.2.1 编译器功能增强

Delphi 的编译器一直是快速开发工具中速度最快的，Delphi 7 专门为程序员提供了两个实用的功能：

- 选择 View | Additional Message Info 菜单项，将会出现如图 1-1 所示的 Message Hints 窗口，在该窗口中可以取得编译器的最新信息。

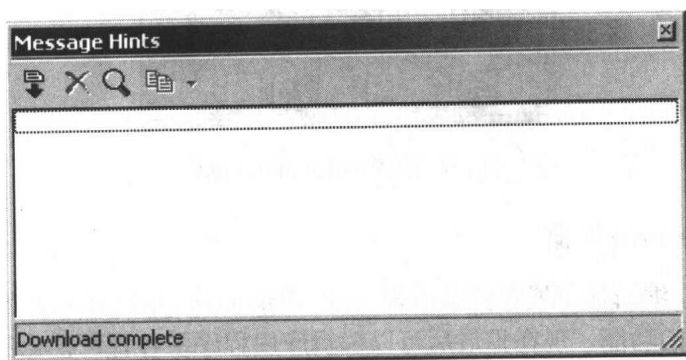


图 1-1 Message Hints 窗口

- 选择 **Projects | Options** 菜单项，将会出现如图 1-2 所示的编译器设置对话框，在该对话框中可以对编译器进行各种控制，以便 Delphi 能够更好地工作。

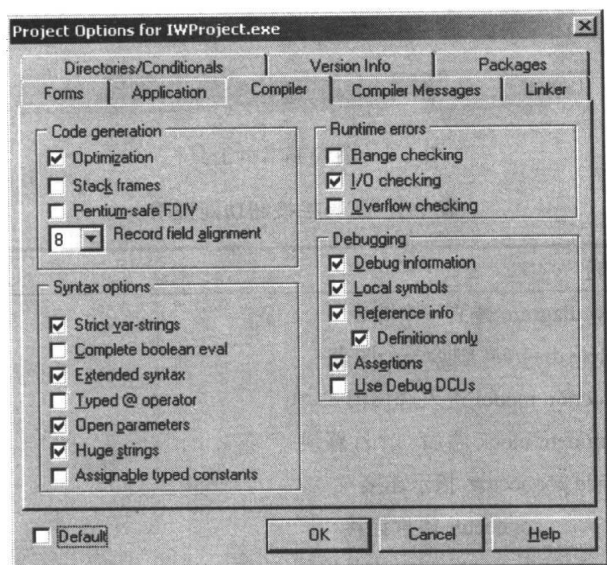


图 1-2 编译器设置对话框

1.2.2 新增了两个网络协议组件组 Indy Intercepts 和 Indy I/O Handlers

在 Delphi 7 的组件面板中新增加了两个标签页 **Indy Intercepts** 和 **Indy I/O Handlers**，如图 1-3 所示。这两个标签页主要实现了 TCP/IP 协议集中的各种协议，通过封装的方法为程序员提供接口，使网络开发更加简单。



图 1-3 新增加的两个标签页

1.2.3 可定制化的 View 视图

在 Delphi 7 中, 编辑器新增了图表编辑页面 (diagram page)。在以前版本的 DataModule 编辑页中就有相同的功能, 现在扩展后可以编辑应用程序中各种可视化的组件了。

该编辑页用箱 (box) 代表各可视和不可视对象, 用线 (line) 表示它们间的逻辑关系, 这样就能显示项目中各个对象的关系。若要观察项目的逻辑组织结构, 只要把项目对象从对象树视图 (Object TreeView) 中拖入图表编辑页面 (diagram page) 下即可。

在页面上还有工具栏 (见图 1-4)。通过这些工具, 能够创建和编辑图表, 对对象间的关系进行编辑。在左边的下拉列表框中可以为图表设定名称和输入说明文字。工具栏中各按钮的功能说明如表 1-1 所示。



图 1-4 图表编辑页工具栏

表 1-1 工具栏按钮功能说明

按钮	功能
	New diagram 建立一个图表
	Delete diagram 删除一个图表
	Selection model 进入选择模式
	Comment block 添加一个注释块
	allude connector 指示连接
	Property connector 属性连接
	Master/Detail connector 主从连接
	Lookup connector 查询连接

1.2.4 可定制的代码完成功能

Delphi 的代码自动完成功能一直为程序员所津津乐道。Delphi 7 在这方面又做了新的努力。程序员可以修改提示代码的颜色, 以便能够更快地编写程序。

单击 Tools | Editor Options 菜单项, 将会出现如图 1-5 所示的编辑器属性 (Editor Properties) 对话框, 选择其中的 Code Insight 标签页, 在其中可以对各项属性进行设置。

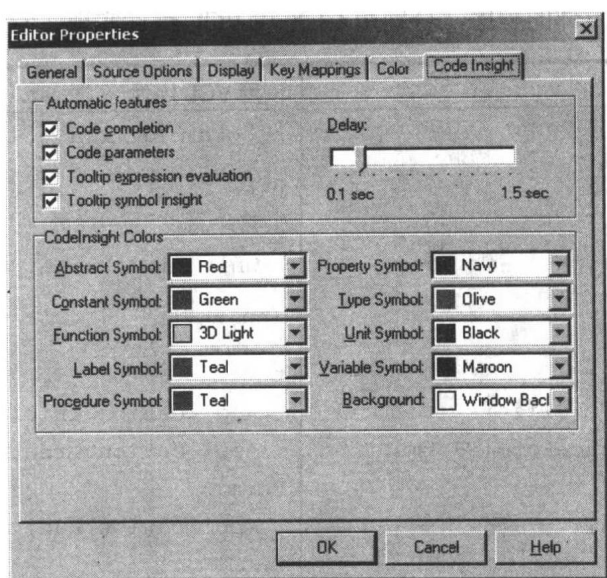


图 1-5 设置代码颜色

1.2.5 更加强大的网络开发功能

Delphi 7 中包含了 AtoZed 公司的 IntraWeb 控件包, 在该控件包中提供了一系列的 Web 开发控件, 使用该控件包可以开发出跨平台的网络应用程序。Delphi 7 提供的 IntraWeb 控件包如图 1-6 所示。

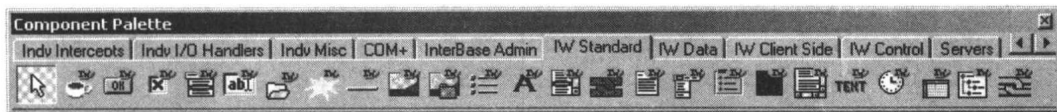


图 1-6 IntraWeb 控件包

1.2.6 跨平台开发库 (CLX)

在 Delphi 中集成了 CLX(Borland Component Library for Cross-Platform), 这样在 Delphi IDE 环境下开发的程序能够在 Windows 或者 Linux 中编译, 然后在 Linux 下运行, 为 Windows 程序员快速开发 Linux 程序提供了极大的方便。

CLX 同 VCL (可视化组件库) 相似, 许多对象具有和 VCL 库对象相同的命名, 相同的属性、方法和事件。利用 Delphi 的 VCL 和 CLX, 可以方便地设计跨平台运行于 Windows 和 Linux 下的程序。要建立 CLX 的程序, 选择 File 下拉菜单的 CLX Application 选项, 而组件栏会动态地把 VCL 组件改为 CLX 组件。



一个只含有一个窗体的程序，分别用 CLX 和 VCL 组件实现，它们的代码对比如下：

使用 CLX 组件	使用 VCL 组件
<pre>program Project1; uses QForms, Unit1 in 'Unit1.pas' {Form1}; {\$R *.res} begin Application.Initialize; Application.CreateForm(TForm1, Form1); Application.Run; end. unit Unit1; interface uses SysUtils, Types, Classes, QGraphics, QControls, QForms, QDialogs, QStdCtrls; type TForm1 = class(TForm) private { Private declarations } public { Public declarations } end; var Form1: TForm1; implementation {\$R *.xfm} end.</pre>	<pre>program Project2; uses Forms, Unit1 in 'Unit1.pas' {Form1}; {\$R *.res} begin Application.Initialize; Application.CreateForm(TForm1, Form1); Application.Run; end. unit Unit1; interface uses Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms, Dialogs; type TForm1 = class(TForm) private { Private declarations } public { Public declarations } end; var Form1: TForm1; implementation {\$R *.dfm} end.</pre>