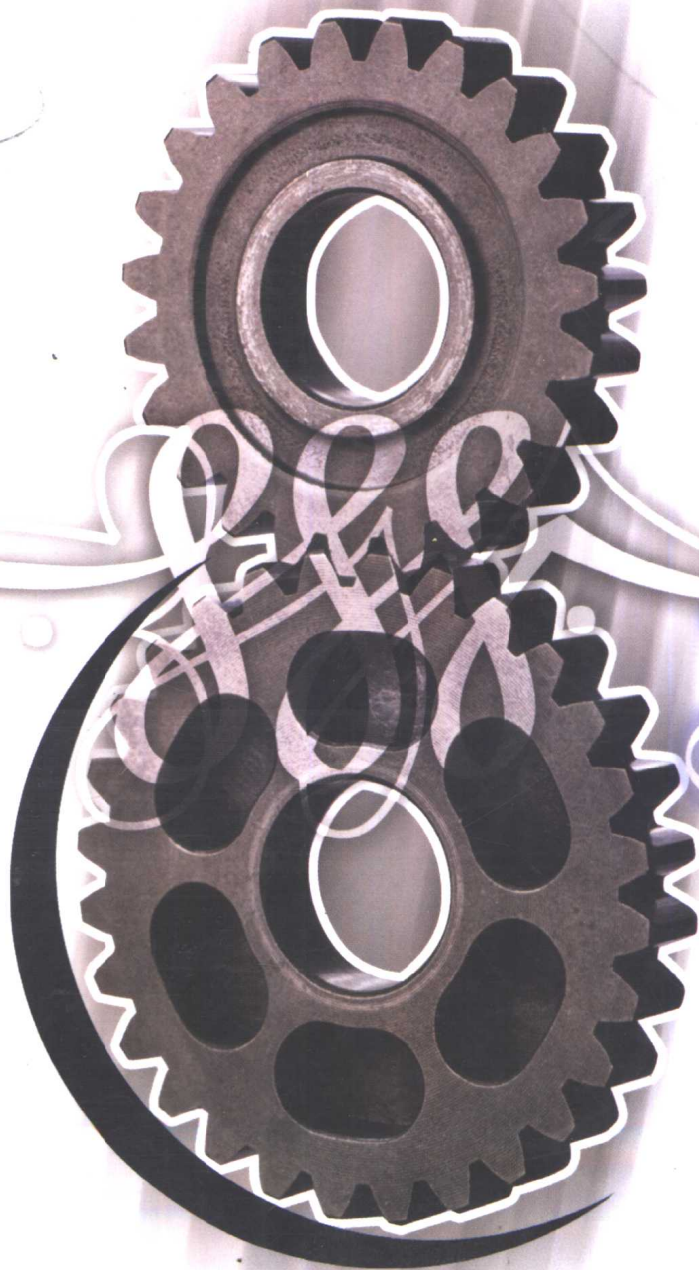


数据库应用开发丛书

PowerBuilder 8 实例

中



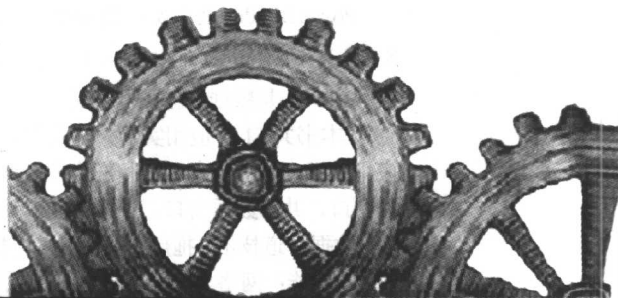
龙腾 杨俊 张勤 杨飞 于世江 编著

国防科技大学出版社

TP311.56

117D

:2



数据库应用开发丛书

PowerBuilder 8

PowerBuilder 8

实例

中

龙腾 杨俊 张勤 杨飞 于世江 编著

国防科技大学出版社

6254/02

北方工业大学图书馆



00521466

内 容 简 介

PowerBuilder 是 Sybase 公司推出的目前最流行的数据库前端开发工具之一,PowerBuilder 8.0 是 PowerBuilder 的最新版本,它全面支持分布式计算和 Web 应用。《PowerBuilder 8 实例(中)》是数据库应用开发丛书之一。

本书共 51 个应用实例,全面深入讲解了 PowerBuilder 8.0 的高级设计开发技术、网络应用技术,全书分高级类和网络类。高级类包括相关知识概述、动态数据窗口技术、下拉数据窗口、共享数据窗口、数据存储、动态 SQL 语句、存储过程、游标、DDE 技术、OLE 技术、数据管道技术、拖放技术、多媒体技术、多线程、多参数传递、窗口消息和 PowerBuilder 8 扩展功能;网络类包括邮件技术、网络文件传输技术、Socket 应用、连接 Jaguar 服务器、C/S 应用实例、Jaguar CTS 应用实例、MTS 应用实例、EAServer 和 MTS 组件技术、Web Targets、Window Plug-in 技术、Window ActiveX 技术、DataWindow Plug-in 和 Web 数据窗口技术等内容。全书并配有实例光盘,便于学习和实践。

该书适合于对 PowerBuilder 比较熟悉的高级用户使用,也适合于 PowerBuilder 8.0 集成开发环境进行网络应用程序设计、网络数据库系统设计和开发及维护人员,同时作为大、中专和高等院校及相关专业教材使用。

图书在版编目(CIP)数据

PowerBuilder 8 实例(中)/龙腾等编著.
—长沙:国防科技大学出版社,2002.6
(数据库应用开发丛书)
ISBN 7-81024-868-5

I. P… II. 龙… III. 数据库系统—软件工具,
PowerBuilder 8.0 IV. TP311.56

中国版本图书馆 CIP 数据核字(2002)第 041723 号

PowerBuilder 8 实例(中)

(数据库应用开发丛书)

龙腾等编著

国防科技大学出版社

电话:0731-4572640 邮政编码:410073

E-mail: gfkdcbs@public.cs.hn.cn

策 划:陆魁玉 卢天颢

责任编辑:卢天颢 曹泽文

全国各地书店经销

湖南省地质测绘印刷厂印刷

*

787×1092 毫米 16 开本 37.5 印张 909 千字

2002 年 7 月第 1 版 2002 年 7 月第 1 次印刷

定价:60.00 元(带光盘)

编委会名单

主 编：明德祥

编 委：（按姓氏笔画顺序排列）

于世江 于 晶 王东阳

王继东 龙 腾 朱文献

刘 靓 李新梅 何晓明

杨 磊 原 雪 谢松县

韩 旭

序

PowerBuilder 8.0 的卓越性能

PowerBuilder 8.0 是 PowerBuilder 的最新版本。相对于以前版本, PowerBuilder 8.0 除了在整体上进行增强以外, 还通过诸多创新特性以及与应用服务器的无缝集成, 提供了基于客户机/服务器、Web 以及多层结构的快速开发环境。

同时, PowerBuilder 8.0 与 EAServer 的紧密集成给用户进行应用程序开发提供了极大的便利, PowerBuilder 8.0 卓越的应用开发效率及强大的 Web 功能为新版本增色不少。PowerBuilder 8.0 可以用来设计、建立高性能的基于分布式、客户/服务器体系结构的网络应用系统, 同时全面支持分布式计算和 Web 环境。具体表现在以下几个方面:

- ◇ PowerBuilder 8.0 集成了 PowerSite 功能, 利用它可以建立、管理和提交 Web 应用。
- ◇ 使用 DataWindow 技术, 开发人员可以利用现有的 PowerBuilder 逻辑, 把它们重新提交到瘦客户端或 HTML JavaScript 客户端。
- ◇ 容易地集成标准的 Web 组件和业务逻辑。
- ◇ 它与 Enterprise Application Server 紧密集成, 便于开发、管理、调试和提交。

总之, 在数据库开发工具领域, PowerBuilder 全面支持面向对象编程, 内置包括数据窗口 (DataWindow) 在内的多种对象类, 是集成功能强大并已使用的第四代编程语言 (4GL)。PowerBuilder 8.0 为建立企业 Web 应用提供了一个全面的、集成的 Web 开发环境。

丛书内容简介及适用对象

本套丛书共三册, 配有精心设计制作的实例光盘, 通过近 200 个应用实例与技巧, 由浅入深、由单机到网络、由基本应用到分布式计算, 全面讲述了 PowerBuilder 8.0 的应用技术与开发技巧。本套丛书是 PowerBuilder 程序设计开发人员和数据库维护人员进行数据库开发和系统设计的必备工具书, 适合于不同层次的 PowerBuilder 用户学习使用, 同时可作为大、中专及高等院校相关专业的教材参考书。

《PowerBuilder 8 实例 (上)》分基础类和提高类。基础类包括 PowerBuilder 8.0 的基本操作、工作空间管理、基本语法、应用对象、数据库创建与管理, ASA7.0 的使用、简单数据库连接、应用程序界面、常用控件的使用、数据窗口等内容; 提高类包括相关知识概述、事务对象、连接数据库、高级控件使用、数据窗口技术及其进一步应用、报表技术、MDI 应用程序设计、创建和使用帮助文件、用户对象、用户事件以及自定义函数等内容。该书适合于 PowerBuilder 8.0 初级用户学习的工具书。

《PowerBuilder 8 实例 (中)》分高级类和网络类。高级类包括相关知识概述、动态数据窗口技术、下拉数据窗口、共享数据窗口、数据存储、动态 SQL 语句、存储过程、游标、DDE 技术、OLE 技术、数据管道技术、拖放技术、多媒体技术、多线程、多参数传递、窗口消息和 PowerBuilder 8.0 扩展功能等内容; 网络类包括相关知识概述、邮件技术、网络文件传输技术、Socket 应用、连接 Jaguar 服务器、C/S 应用实例、Jaguar CTS 应用实例、MTS 应用实例、EAServer 和 MTS 组件技术、Web Targets、Window Plug-in 技术、Window ActiveX 技术、DataWindow Plug-in 和 Web 数据窗口技术等内容。该书适合于对 PowerBuilder 8.0 比较熟悉的高级用户使用, 也适合于 PowerBuilder 8.0 集成开发环境进行网络应用程序设计、网络数据库系统设计和开发及维护人员。

《PowerBuilder 8 实例 (下)》分技巧类和系统类。技巧类包括系统设计理论基础、常

用数据库管理系统介绍、PowerBuilder 8.0 与数据库的各种连接技术、两个实际的 ASA7.0 数据库应用系统、PowerBuilder 8.0 应用程序调试与分发以及安装文件的制作等内容；系统类包括相关知识概述、控件技术、拖放技术、数据窗口技术、应用程序界面设计、DDE 技术、OLE 技术以及网络应用小技巧等。该书适合于对 PowerBuilder 比较熟悉的高级用户使用，也适合于 PowerBuilder 8.0 集成开发环境进行网络应用程序设计、网络数据库系统设计和开发及维护人员。

丛书的特点

本套丛书经过精心的策划和设计，以崭新的面貌、全面的内容、科学合理的设计与安排为广大用户提供一套实用而全面的技术实例参考书。它是目前国内出版的 PowerBuilder 书籍中提供实例最多、最具代表性和实用价值的丛书。概括起来具有以下 4 个主要特色：

完整的体系结构

该套丛书在内容结构上进行了精心设计，以应用实例和技巧为基础，贯穿 PowerBuilder 8.0 各个方面的内容。该丛书自成为一套完整体系，同时，每一本书在内容和结构上又相对独立。

强大的编者阵容

该套丛书的主要作者都是计算机专家和博士，都是多年来一直从事应用 PowerBuilder 和 Sybase Internet 产品的数据库应用专家和高手。

超值的实例源码

该套丛书推出的经典实例近 200 个，是作者们多年来使用 PowerBuilder 进行数据库系统设计开发中的经验总结和长期积累，其中精选出的大量实例源码已在实际应用中得到验证。

全面丰富的内容

该套丛书在内容的选材上覆盖了 PowerBuilder 8.0 的基本操作、系统应用、高级开发和网络技术（Socket 程序、Web 技术、C/S 系统、Jaguar CTS 应用及分布式计算）等。

致谢与其他

本套丛书由长期从事计算机软件设计和开发工作的明德祥博士担任主编，并负责全套丛书的定稿、审核工作。在编著过程中参阅了国内外部分 PowerBuilder 文献并征求了国防科技大学和湖南大学许多专家意见并经过多次论证部分源码后，由国防科技大学出版社组织和出版。长沙芦雨科技图书发行有限公司也为本书的编写和出版及发行给予了极大的关心和支持，并付出了艰辛的劳动，在此一并表示真诚的感谢。

由于时间仓促和水平有限，书中难免会有不足和错误之处，敬请广大读者提出宝贵的意见和建议，阅读时只作参考。以便我们在下一次再版时进行修订。

前 言

本书是针对那些对 PowerBuilder 开发环境有了深入了解,能够使用 PowerBuilder 提供的各种窗口、画板、工具、PowerScript 语言、各种控件实现一定水平的应用程序设计以及具有一定网络应用基础的初中级用户,围绕 PowerBuilder 应用程序设计开发、高级技术和网络程序应用,引导读者进一步掌握和设计开发 PowerBuilder 高级应用程序和网络应用程序的方法。

本书分为高级类和网络类,高级类包括概述、高级数据窗口技术、数据存储与动态 SQL 语句、拖放技术、多媒体技术、DDE 和 OLE 技术、数据管道技术和其他应用技术等 8 个部分,共有 33 个应用实例;网络类包括概述、一般网络技术、Socket 应用技术、Web 应用技术和分布式应用等 5 个部分,共有 18 个网络应用实例。

高级类首先讲解本类应用实例中要使用到的相关知识,内容包括动态数据窗口技术、下拉数据窗口、数据窗口共享、数据存储、动态 SQL 语句、存储过程、游标、DDE 技术、OLE 技术、数据管道技术、拖放技术、窗口之间多参数传递、窗口消息以及高级 API 调用等主要基本知识。同时,这一章的内容也是在进入本类的实例内容学习前的重要基础知识准备。如果用户对 PowerBuilder 8.0 这些高级技术应用已有了一定的基础,可以跳过这一章而直接进入第 2 章开始实例的学习。高级类中的 33 个应用实例深入浅出地介绍了 PowerBuilder 8.0 进行系统开发中的相关高级应用技术,内容涉及高级数据窗口技术、数据存储、动态 SQL 语句、DDE 技术、OLE 技术、数据管道技术、拖放技术、窗口之间多参数传递等。

网络类首先对本类应用实例中要使用到的相关知识进行概述,内容包括分布式计算基本概念、分布式应用的体系结构、基本网络应用技术、Sybase 的 Internet 计算模型、PowerBuilder 分布式结构以及分布式应用设计等主要相关知识。同时,这一章的内容也是在进入本类的实例内容学习前的重要基础知识准备。如果用户对这些网络技术应用已有了一定的基础,可以跳过这一章而直接进入第 10 章开始实例的学习。网络类中的 18 个应用实例全面而深入地介绍了 PowerBuilder 8.0 进行网络应用系统开发中的相关技术,内容涉及邮件技术、网络文件传输技术、Socket 应用、连接 Jaguar 服务器、C/S 应用实例、Jaguar CTS 应用实例、MTS 应用实例、EAServer 和 MTS 组件技术、Web Targets、Window Plug-in 技术、Window ActiveX 技术、DataWindow Plug-in 技术和 Web 数据窗口技术等。

结合实例讲解知识要点,不仅有助于读者理解相关的知识和网络应用设计开发技巧,而且在实际应用中具有较大的可移植性和可参考性。实例中是许多作者多年开发总结的设计开发技巧,具有很大的阅读和参考价值。

另外,本书配有精心设计制作的实例光盘,包含本书全部技巧实例的源程序。这些源程序在 Windows 2000 和 PowerBuilder 8.0 下全部调试通过,并有相应的源程序注释信息。

本书由龙腾、杨俊、张勤、杨飞、于世江编写,明德祥负责最后的校正和统稿工作。参加本书编写、整理、资料搜集工作的人员有:王东阳、邓波、杨磊、谢松县、李新梅、于晶、何晓明等。

由于作者水平所限以及时间仓促,书中难免会有不足和错误之处,欢迎广大读者朋友批评指正,阅读时只作参考。

作 者

目 录

高 级 类

第 1 章 概述.....	1
第 2 章 高级数据窗口技术实例.....	27
2.1 动态创建数据窗口	28
2.2 动态读取和改变数据窗口的风格.....	37
2.3 动态改变数据窗口行和列的数据状态.....	53
2.4 数据窗口的缓冲区	59
2.5 数据窗口的数据有效性检验.....	69
2.6 数据窗口若干实用技术	75
2.7 PowerBuilder 中的下拉数据窗口	81
2.8 PowerBuilder 中实现两个数据窗口的共享.....	87
2.9 创建和使用 Rich Text 数据窗口	94
2.10 实现 RichTextEdit 控件与数据窗口共享	103
2.11 使用 Query Mode 查询模式	108
2.12 动态修改数据窗口的查询条件.....	114
2.13 控制数据窗口列的修改属性.....	119
2.14 实现数据窗口的多表更新.....	125
第 3 章 数据存储与动态 SQL 语句实例	133
3.1 创建和使用数据存储 (DataStore)	134
3.2 动态 SQL 语句、存储过程和游标.....	142
第 4 章 拖放技术实例.....	153
4.1 拖放技术与 TreeView 控件.....	154
第 5 章 多媒体技术实例.....	163
5.1 PowerBuilder 的动画技术	164
5.2 PowerBuilder 多媒体技术	170
第 6 章 DDE 和 OLE 技术实例.....	179
6.1 PowerBuilder 中的 DDE 技术	180
6.2 通过 DDE 的两种链接方式与 Excel 通讯.....	189
6.3 PowerBuilder 中的 OLE 技术.....	198
6.4 通过 OLE 实现与 WORD 的通讯.....	212
6.5 通过 OLE 实现与 PowerPoint 的通讯	217
6.6 通过 OLE 实现与 Excel 的通讯.....	221
6.7 通过 OLE 实现存取 BLOB 类型数据	225

6.8 通过 OLE 实现串口通讯.....	233
第 7 章 数据管道技术实例.....	241
7.1 在 PowerBuilder 环境下创建和执行数据管道.....	242
7.2 在应用程序中通过编程执行数据管道.....	253
第 8 章 其他应用技术实例.....	267
8.1 Win32 API 函数与 PowerBuilder 高级编程.....	268
8.2 窗口之间多参数传递.....	280
8.3 PowerBuilder 窗口消息.....	289
8.4 PowerBuilder 编写多线程应用程序.....	296

网 络 类

第 9 章 概述.....	305
第 10 章 一般网络技术实例.....	335
10.1 利用 PowerBuilder 实现远程拨号.....	336
10.2 通过 FTP 实现文件在网络中的上传和下载.....	342
10.3 如何开发邮件应用程序.....	348
10.4 Inet 对象的使用.....	364
10.5 串行通讯在 PowerBuilder 中的实现.....	370
10.6 PowerBuilder 8.0 中的 Socket 编程.....	382
第 11 章 Web 应用技术实例.....	401
11.1 PowerBuilder 8.0 中的 Window Plug-in 技术.....	402
11.2 PowerBuilder 中的 Window ActiveX 技术.....	416
11.3 DataWindow Plug-in 技术.....	434
11.4 利用 Jaguar CTS 实现 Web 数据窗口.....	439
11.5 利用 ASP 与 MTS 实现 Web 数据窗口.....	454
第 12 章 分布式应用技术实例.....	465
12.1 在 PowerBuilder 8.0 中创建 EAServer 组件.....	466
12.2 在 EAServer 组件中访问数据库.....	482
12.3 创建和使用 EAServer 客户.....	493
12.4 创建和使用 COM/MTS 组件.....	513
12.5 创建和使用 COM/MTS 客户.....	534
12.6 基于 JaguarCTS 的文件上传下载系统.....	539
12.7 Web Service 技术.....	570

第1章

概述

内 容 提 要

- ➡ 动态数据窗口技术
- ➡ 数据窗口共享技术
- ➡ 数据存储
- ➡ 动态SQL语句、存储过程与游标
- ➡ DDE 和 OLE 技术
- ➡ 数据管道技术
- ➡ 拖放技术
- ➡ 外部 API 函数调用

本书包括的主要知识要点有动态数据窗口技术、下拉数据窗口、数据窗口共享、数据存储、动态 SQL 语句、存储过程、游标、DDE 技术、OLE 技术、数据管道技术、拖放技术、窗口之间多参数传递、窗口消息以及 API 调用等。

本章主要讲解本书后面的实例中要使用到的相关知识。同时, 本章内容也是进入本书后面的实例内容学习前的重要基础知识。如果用户对 PowerBuilder 8.0 的这些高级技术应用已有了一定的基础, 可以跳过本章而直接进入第 2 章开始实例的学习与实践。

1.1 动态数据窗口技术

数据窗口对象和其中的所有对象(如列、文本、图表和图形等)都有一个属性集, 这些属性可以在运行时通过使用数据窗口函数或属性表达式进行查看和修改, 也可以在运行时创建数据窗口对象, 在 PowerBuilder 8.0 中, 属性表达式使用“.”符号操作属性。

在运行时被修改或创建的数据窗口对象称为动态数据窗口对象。利用这种动态特性, 可以允许用户改变数据窗口对象的外观(如改变颜色和文本的字体等), 或者通过重新定义数据源来创建查询。在创建完一个动态数据窗口对象且满足用户对外观和数据显示的要求后, 用户就可以以报表形式打印其中的内容。

1.1.1 使用 Create 函数

数据窗口对象的创建一般都是使用 DataWindow 画板完成, 但在很多情况, 数据窗口画板中不能完成系统的需求, 这样往往就需要通过 Create 函数创建数据窗口或者是动态修改数据窗口中的对象。因为这样比较简单和直观。

在运行时, 可以使用 Create 函数动态地创建一个数据窗口对象。Create 使用指定的源代码生成一个数据窗口对象。它用新的数据窗口对象取代当前在数据窗口控件中指定的数据窗口对象。

使用 Create 函数创建数据窗口对象时, 数据窗口控件和事务对象之间的连接将被删除, 因此, 在调用 Create 后, 必须通过调用 SetTransObject 或 SetTrans 函数来重新设置数据窗口控件的事务对象。

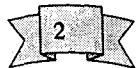
在 JavaBeans 组件和 Web 控件中, 如果使用的是不需要调用 SetTransObject 或 SetTrans 函数的连接技术, 则不必在 Create 后调用它们。

在 PowerBuilder 8.0 中, 可以:

- 使用事务对象的 SyntaxFromSQL 函数。
- 使用 PowerScript 函数 LibraryExport。
- 使用 DataWindow 对象的 DataWindow.Syntax 属性。
- 自己创建语法。

对于大多数的动态数据窗口对象, 都可以使用 SyntaxFromSQL 来创建语法。如果使用 SyntaxFromSQL, 则所有要做的就是提供 SELECT 语句和表现风格。

在 PowerBuilder 8.0 中, SyntaxFromSQL 是事务对象的函数, 当调用该函数时, 必须先



连接该事务对象。如果使用专用数据库接口程序,则表名会自动地赋以所有者的名字,要在程序中实现相同的功能,则必须设置事务对象的 USERID 属性以使表名能赋以所有者的名字,并可以使用其扩展属性。

SyntaxFromSQL 有三个必需的参数:

- 包含 DataWindow 对象的 SELECT 语句的字符串。
- 指定表现风格和其他设置的字符串。
- 保存任何错误信息的字符串名字。

SyntaxFromSQL 返回使用指定的 SELECT 语句建立的数据窗口对象的完整的语法。

如果使用的 DBMS 是 SQL Server,则在调用 SyntaxFromSQL 之前,请设置事务对象的 AutoCommit 属性为 TRUE。这就确保了创建的数据窗口对象中包含了标签顺序和更新能力。

在 PowerBuilder 中,可以使用 PowerScript 函数 LibraryExport 导出数据窗口对象的语法并存储在一个字符串中,接着,可以在 Create 中使用导出的语法(或修改后的语法)来创建一个数据窗口对象。

可以使用 DataWindow.Syntax 属性来获得一个现有的数据窗口对象的语法源代码,使用此源代码可以作为语法的模板或稍作修改形成一个新的语法。源代码语法中的许多值都是同数据窗口对象的属性相关联的。

供给 Create 函数的数据窗口语法源代码是非常复杂的。要看数据窗口对象语法的例子,可在 PowerBuilder 的 Library 画板中,把一个数据窗口对象导出为一个文本文件,接着在编辑器中就可以查看该文件中的语法。

1.1.2 使用 Modify 函数

在运行时,可以使用 Modify 函数或属性表达式来设置属性的值,这样就可以改变在数据窗口画板中通常所指定的一些值。

在修改某属性之前,可能要获得属性当前的值并把它存在一个变量中,以便以后能恢复该属性的初始值。要获得数据窗口对象或其中一个元素的当前属性的信息,可以使用 Describe 函数或属性表达式。

对于一些数据窗口属性,可以通过一个表达式来赋值,该表达式是在运行时获得值,而不是直接指定一个值。例如,下面的语句就是在运行时,如果工资小于¥1,000 则以红色显示工资,反之则以黑色显示:

```
dw_1.Modify("salary.Color='0~t if salary<1000,255,0')")
```

在数据窗口画板中指定的表达式语法和在代码中的表达式的语法是不相同的。

也可以使用 Modify 函数进行如下的操作:

- 在数据窗口对象中创建新的对象。
- 这就可以动态地增加数据窗口对象的数据窗口元素(如文本、位图和图表对象等)。
- 在数据窗口对象中删除对象。

这可以动态地删除不需要的数据窗口元素。

1.1.3 使用 Describe 函数

此函数的功能是返回数据窗口控件结构方面的指定信息, 包括数据窗口对象以及数据窗口对象中其他对象的属性取值。在 Describe() 函数的参数中可以通过字符串指定要得到其值的一组属性, 该函数将会把它们值作为一个字符串返回。

该函数的语法为: dwcontrol.Describe (propertylist)。其中参数 dwcontrol 为数据窗口控件名, propertylist 为属性列表, 是个字符串, 由要查看其取值的一个或多个属性或表达式组成, 属性以及表达式之间使用空格分隔。这里的属性指数据窗口对象的属性。

该函数返回一个字符串, 这个字符串由对应于 propertylist 中各属性的具体取值构成, 各属性值之间使用换行字符 (~n) 分隔。如果属性列表中包含了无效的项 (比如拼错了属性名或写错了表达式), 那么 Describe() 函数在相应项的位置返回一个惊叹号 (!), 并忽略随后的所有项。当某个属性没有值时, Describe() 函数在相应位置返回一个问号 (?)。如果某个属性的值是个问号 (?), 那么 Describe() 函数将在该项位置返回问号, 并把问号放置到引号中 (通常为单引号)。如果任何参数的值为 NULL, 则 Describe() 函数返回 NULL。

使用 Describe() 函数可以深入理解数据窗口的结构。例如, 我们可以计算表达式的值; 表达式中可以使用指定行/列的值; 可以查到数据窗口都具备哪些区域 (也有称作带 (Band) 的叫法), 各列的数据类型是什么等。使用 Describe() 函数也能够得到数据窗口当前的属性取值, 之后可以使用数据窗口控件 Modify() 函数进行所需的修改。

应用程序中经常使用 Describe() 函数得到数据窗口当前的 SELECT 语句, 然后根据需要进行修改, 比如增加 WHERE 子句, 修改排序方式等。

1.1.4 DW Syntax 工具

在 PowerBuilder 8.0 中提供了一个 DW Syntax 工具, 利用它可以容易地查看建立属性表达式、Describe、Modify 和 SyntaxFromSQL 语句的正确语法。在 DW Syntax 中, 单击按钮来指定要使用 DataWindow 的哪一个属性, 则 DW Syntax 会自动地建立该属性的正确语法, 建立完后可以把它复制和粘贴到应用程序的代码中, 要使用 DW Syntax, 选择菜单 “File/New” 并选择 Tool 标签页中的 DataWindowSyntax, 如图 1-1 和图 1-2 所示。

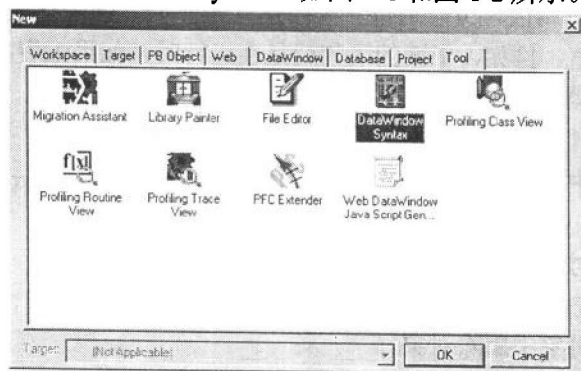


图 1-1 选择 Tool 标签中的 DataWindowSyntax

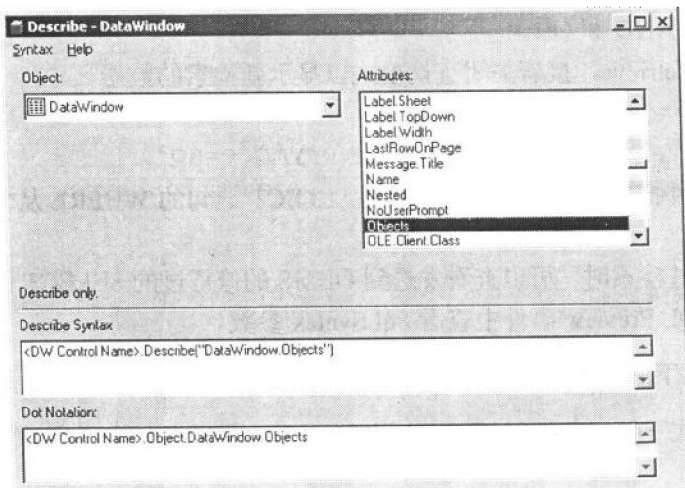


图 1-2 DataWindow Syntax

也可以使用 PowerBuilder Browser 来获得数据窗口的属性列表：在数据窗口标签页中，选择左边面板中的一个数据窗口对象，并在右边面板中选择 Properties。要查看数据窗口对象内的控件的属性，可以双击数据窗口对象名字，并选择要查看的控件即可。

1.1.5 查询模式 (QueryMode)

当调用数据窗口控件的 Retrieve 函数时，检索的是在数据窗口对象的 SELECT 语句中指定的行。通过把数据窗口设置成查询模式，可以在运行时允许用户指定检索哪些行。要实现此功能，可以使用 Modify 函数或属性表达式。

如果数据窗口对象的数据源被修改并包含有 UNION 关键字或嵌套的 SELECT 语句，则该数据窗口对象不能使用查询模式。

1. 查询模式的工作原理

一旦数据窗口处于查询模式，用户就可以使用查询来指定查询条件。就像使用 Quick Select 定义数据源一样，当定义完条件后，在下次数据检索时它们就会加到 SELECT 语句的 WHERE 从句中。

2. 查询模式的使用方法

(1) 通过代码打开查询模式

```
dw_1.Modify("datawindow.querymode=yes")
```

所有显示在数据窗口中的数据都被清空（虽然它们仍然在数据窗口控件的 Primary 缓冲区中），用户可在其中输入相应的查询条件。

(2) 在数据窗口中指定查询条件

就像使用 QuickSelect 定义数据窗口对象的数据源一样。在一行中输入的条件将用“AND”连在一起，在不同行中输入的条件用“OR”连在一起。有效的运算符是=、<>、<、>、<=、

>=、LIKE、IN、AND 和 OR。

(3) 调用 Retrieve, 接着关闭查询模式以显示新检索的数据

```
dw_1.Retrieve()
dw_1.Modify("datawindow.querymode=no")
```

数据窗口控件把新建立的查询条件加到 SELECT 语句的 WHERE 从句中, 接着检索和显示指定的行。

当数据有条件检索时。可以查看发送到 DBMS 的修订过的 SELECT 语句。方法是在数据窗口控件的 SQL Preview 事件中查看 Sql Syntax 参数。

3. 查询模式下的数据排序

当在查询模式下指定条件时, 可以使用 QuerySort 属性来允许用户对数据窗口中的数据进行排序。

下面的语句使数据窗口中的第一行成为排序条件, 在 PowerBuilder 8.0 中:

```
dw_1.Modify("datawindow.querysort=yes")
```

默认的情况下, 查询模式使用编辑风格和列的其他定义。如果想在查询模式下覆盖这些属性且为列提供一个标准的编辑控件, 则使用每列的 Criteria.Override_Edit 属性:

```
dw_1.Modify("mycolumn.criteria.override_edit=yes")
```

也可以在数据窗口画板中, 从列的弹出菜单上选择 “Query Criteria/Equality Required” 项来指定上面的该属性。如果为 yes, 则用户可以在一个单元中指定任何数量的字符。

可以通过如下代码, 强迫用户在查询模式下为一个列指定条件:

```
dw_1.Modify("mycolumn.criteria.required=yes")
```

也可以在数据窗口画板中, 从列的弹出菜单中选择 “Query Criteria/Equality Required” 项来指定上面的属性。这就确保用户必须为该列指定条件, 且条件使用的运算符是 “=” 而不是其他的运算符, 如 “<” 或 “>” 等。

1.2 下拉数据窗口

在用数据窗口进行数据录入时, 经常要用到代码录入, 即对经常输入的某些值设置代码, 录入时直接录入代码即可。例如, 有两个表, 主表 main_table 和代码表 code_table, 录入时主表中需要代码表中的数据, 但不允许录入不在代码表中的数据。这时就需要使用下拉数据窗口 (DropDownDatawindow) 技术。

下拉数据窗口作为一种编辑风格, 在数据窗口中有着广泛的应用, 它使数据窗口更灵活、更方便。PowerBuilder 中通过 GetChild()、IsChild() 等函数可以对下拉数据窗口对象进行操作和控制

1. GetChild() 函数

该函数的主要功能是得到指定子数据窗口的引用并将其保存到变量中, 然后应用程序就可以使用该变量来操作子数据窗口对象。这里所说的子数据窗口是指数据窗口对象中的下拉数据窗口。

当应用程序需要以显式方式提取子数据窗口对象的数据时,先使用 GetChild()函数得到该子数据窗口对象的引用,然后像操作普通数据窗口控件那样检索子数据窗口对象的数据。一般来说,在显示主数据窗口时,PowerBuilder 会自动提取出子数据窗口对象的数据,但是,当子数据窗口对象使用了提取参数或检索条件发生变化且需要检索新的数据时,应用程序就需要自己编写代码来实现数据提取工作了。

当用户在主数据窗口中插入一行或提取主数据窗口的数据时,PowerBuilder 也自动地提取出子数据窗口的数据。但是,如果子数据窗口带有提取参数,那么 PowerBuilder 将显示一个对话框,让用户输入提取参数的值。要想不让 PowerBuilder 显示该对话框,应用程序中可以在修改主数据窗口之前先显式地提取出子数据窗口的数据。

使用数据窗口控件的对象函数 Modify()修改数据窗口的某些属性后,先前执行 GetChild()函数得到的子数据窗口对象引用可能不再有效,此时,需要重新调用 GetChild()函数来得到有效的子数据窗口对象引用。

2. IsChild()函数

该函数的主要功能是判断一个数据窗口是否为子数据窗口类型。IsChild()函数提供了一种区别子数据窗口和普通数据窗口控件或数据存储的方法。在应用程序的设计开发中,有时要调用 IsChild()函数来判断一个数据窗口是否为子数据窗口类型,然后根据返回值,确定对该数据窗口的进一步操作。

1.3 数据窗口共享技术

1.3.1 数据窗口共享简介

在两个或者更多的数据窗口控件之间共享数据,就是指允许在不同的数据窗口中使用相同的数据,从而不必在数据窗口之间复制数据,也不用担心怎样从两个数据窗口中更新数据。

数据窗口中的数据一般有如下三种类型:

- Static data (静态数据): 通常由硬编码值在你的应用程序中使用,可以作为常数、预定义列表框值、复选框状态值等。
- Occasionally modified data (偶尔修改的数据): 包含系统代码,通常与数据库查询表相联系。
- Volatile data (易变数据): 每天数值多次修改或添加的数据(例如发货单号)。

第二种和第三种类型数据对于一个下拉式 DataWindow 是理想的候选者,但是只有第二种类型才是真正值得共享的。这是因为在用户使用共享的主数据窗口时,易变数据或许已经改变了。如果数据处于高速缓冲区中,以备应用程序广域使用,而且应用程序长时间的处于开放状态,这一点就尤为突出了。

有两种类型的数据窗口控件处于一种共享的关系:主数据窗口控件和次数据窗口控件。主数据窗口控件实际上包含它的缓冲区中的数据。次数据窗口控件被赋予了对主数据窗口

中数据的访问权。

在数据窗口之间只有数据是共享的, 包含每个缓冲区以及当前的过滤器和排序。对于每个数据窗口来说, 格式化、列布局以及相应的内容都是独立的。由于数据本身是共享的, 使用下面任何一个函数 (或者适当的直接访问语法) 做出的修改, 都影响到主数据窗口和相关的所有数据窗口, 这些函数是: DeleteRow()、Filter()、ImportClipBoard()、ImportFile()、ImportString()、InsertRow()、Retrieve()、Reset()、RowsCopy()、RowsDiscrd()、RowsMove()、SetFilter、SetSort()、Sort() 以及 Update()。换句话说, 这些函数被重新定向于主数据窗口。这也意味着与这些函数相联系的所有事件只能从主数据窗口中进行触发。

在共享数据之前, 必须进行慎重的考虑分析。主数据窗口和次数据窗口必须具有完全相同的结果集说明。这意味着主数据窗口和次数据窗口之间必须有相同的数据类型、列名以及列顺序。SELECT 语句在表中和 WHERE 子句中可能变化, 但是仍然可以共享。只要列被定义成期望的结果, 你甚至可以和一个内部数据源数据窗口进行共享。

注意: 对于内部数据源数据窗口, 你需要使用适当的数值数据类型, 否则共享失败。

1.3.2 数据窗口共享方法

要实现数据共享, 需要使用两个重要的函数: ShareData() 函数和 ShareDataOff() 函数。

1. ShareData() 函数

ShareData() 函数的功能是在主数据窗口控件与从数据窗口控件之间共享数据。这种共享并不共享显示格式, 只是共享数据, 包括主缓冲区、删除缓冲区以及过滤缓冲区中的数据, 同时也共享排序顺序。使用 ShareData() 函数时, 要求主数据窗口控件与从数据窗口控件中的数据窗口对象的列必须相同, 但各自使用的 SELECT 语句可以不一样。

如果有多个从数据窗口需要共享主数据窗口的数据, 那么, 对每一个从数据窗口, 调用主数据窗口的 ShareData() 对象函数。

主数据窗口与从数据窗口共享数据后, 无论应用程序调用主数据窗口的对象函数修改数据还是调用从数据窗口的对象函数修改数据, PowerBuilder 都将针对主数据窗口完成修改, 并且将修改反映到所有从数据窗口中。

2. ShareDataOff() 函数

要终止数据窗口之间的数据共享, 需要调用 ShareDataOff() 函数。当调用主数据窗口的 ShareDataOff() 函数时, 终止所有从数据窗口与该主数据窗口之间的数据共享。当调用从数据窗口的 ShareDataOff() 函数时, 终止该从数据窗口与主数据窗口之间的数据共享。终止了数据共享后, 从数据窗口中的数据将消失。

从数据窗口只能得到主数据窗口中已有的数据。如果你在从数据窗口中增加了一个计算域 (Computed Field, 与计算列不同), 而主数据窗口中并没有相应的计算域, 那么程序运行时从数据窗口的计算域中不会显示数据。要想在从数据窗口中显示计算域的数据, 只有在主数据窗口中加上相同的计算域。

共享数据时, 从数据窗口不能进入查询模式。任何设置从数据窗口 QueryMode 或