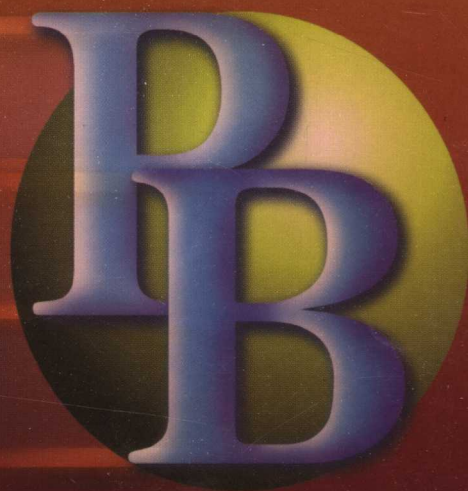


21 世纪工程应用计算机技术丛书

PowerBuilder

应用工程软件开发指南

张英杰 曹 岩 曾高强 编著

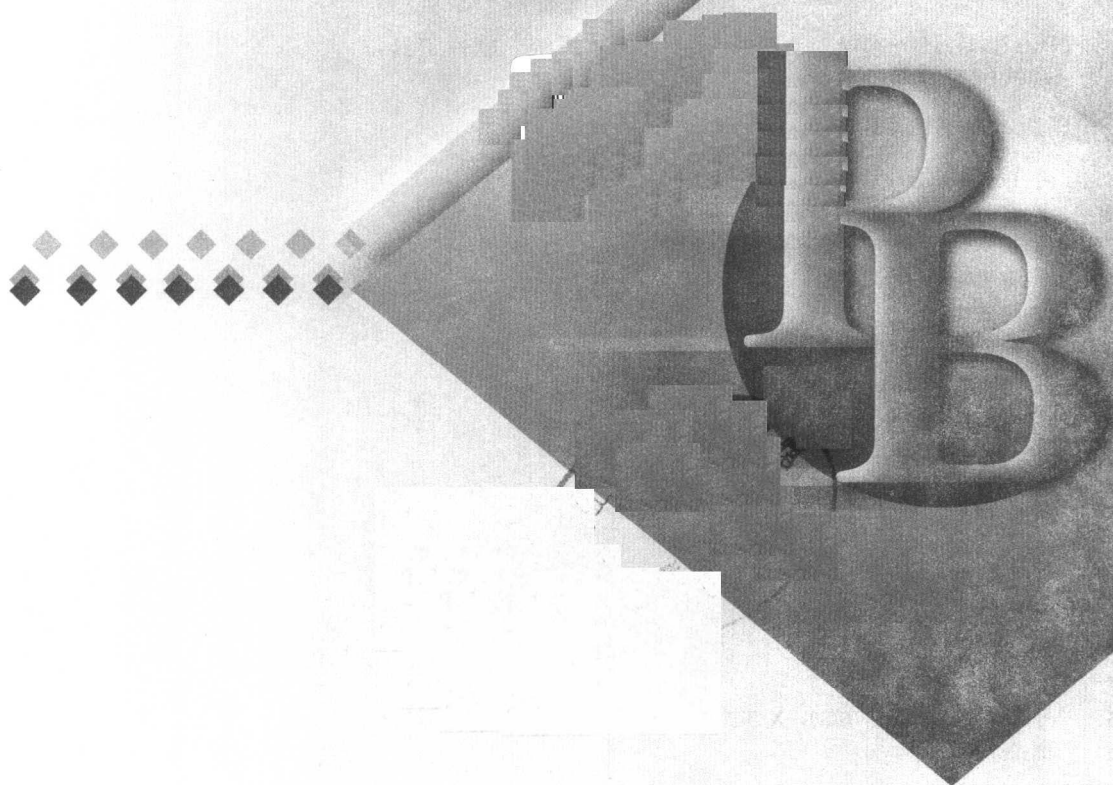


21 世纪工程应用计算机技术丛书

PowerBuilder

应用工程软件开发指南

张英杰 曹 岩 曾高强 编著



西安交通大学出版社
XI'AN JIAOTONG UNIVERSITY PRESS

内 容 提 要

本书以一个具体的工程应用软件的开发和实施过程为主线,详细地介绍了 PowerBuilder 软件对数据库的操作和各种控件的使用方法,基本涵盖了 PowerBuilder 应用开发的全过程。对开发中经常会碰到或程序员比较关心的问题(如界面制作、并发控制、ASA8 数据库实现 C/S 应用等)、当前比较热的一些技术(如 Jaguar CTS、Web 应用)、一般的 PowerBuilder 工具书上没有或论述较少的技术问题(如 API 应用、编译与发行)等进行详细深入地论述,并详细介绍了应用 PowerBuilder 软件进行网络程序设计开发的方法和技巧。

本书注重概念与实际应用相结合,既适合于学习 PowerBuilder 的初学者的和中级读者,更适合于用 PowerBuilder 进行应用软件开发的高级用户,可供工程开发人员深入了解 PowerBuilder 软件之用,也可供大专学生和工程软件开发人员使用。

图书在版编目(CIP)数据

PowerBuilder 应用工程软件开发指南/张英杰,曹岩,曾高强,
编著. —西安:西安交通大学出版社,2005. 2

(21 世纪工程应用计算机技术丛书)

ISBN 7-5605-1880-X

I. P… II. ①张… ②曹… ③曾… III. 数据库系统—软件工具,
PowerBuilder—指南 IV. TP311.56-62

中国版本图书馆 CIP 数据核字(2004)第 078498 号

书 名:PowerBuilder 应用工程软件开发指南

编 著:张英杰 曹岩 曾高强

责任编辑:贺峰涛 屈晓燕

文字编辑:曹银强

出版发行:西安交通大学出版社

地 址:西安市兴庆南路 25 号(邮编:710049)

网 址:<http://unit.xjtu.edu.cn/unit/jtupress>

电 话:(029)82668357,82667874(发行部)

(029)82668315,82669096(总编部)

电子信箱:eibooks@163.com

印 刷:陕西友盛印务有限责任公司

经 销:各地新华书店

版 次:2005 年 2 月第 1 版 2005 年 2 月第 1 次印刷

开 本:787mm×1 092mm 1/16

印 张:22

字 数:537 千字

书 号:ISBN 7-5605-1880-X/TP·385

定 价:28.00 元

版权所有 侵权必究!

前 言

PowerBuilder 是一种优秀的数据库应用前端开发工具,以其交互式、快捷性、简便性、易学习、高效率 and 先进性等优点,从问世以来一直受到程序开发人员的青睐。从 1991 年 Sybase 公司发布 PowerBuilder 1.0 至今,已经根据客户的需求和对其他技术的支持,不断开发和升级新的版本。为了满足网络技术发展对开发软件提出的新要求,2003 年 8 月, Sybase 公司发布了最新的 PowerBuilder 9.0 版本,新版的 PowerBuilder 除了对传统客户端/服务器(Client/Server;C/S)网络结构应用开发功能的增强以外,重点扩展和增强了 PowerBuilder 在分布式应用开发、基于组件应用开发和 Web 应用开发等方面的功能。

本书为了方便读者快速理解和应用 PowerBuilder 软件,从软件工程的角度出发,以具体工程应用实例的完整开发过程为主线,详细讲解了应用 PowerBuilder 进行数据库应用系统开发时所涉及的知识、技术和所遇到的问题。

本书由 11 章组成,第 1 章简要介绍了 PowerBuilder 的特点、集成开发环境以及应用 PowerBuilder 进行软件开发的一般过程;第 2 章详细介绍了数据库基础知识以及在 PowerBuilder 环境下对数据库的连接和操作过程;第 3 章详细讲解了 PowerBuilder 脚本语言——PowerScript 语言;第 4 章重点介绍了 PowerBuilder 中采用 SQL 语句进行各种数据库操作的方法;第 5、6、7 章分别对 PowerBuilder 中常用控件、对象的应用、OLE 技术以及 PowerBuilder 的专利技术——DataWindow 技术做了详细介绍;第 8 章介绍了 PowerBuilder 的部分实用的高级编程技术(如多媒体应用开发、动态库混合编程、WindowsAPI 编程等);第 9、10 章分别介绍了应用 PowerBuilder 进行分布式应用开发和 Web 开发的详细过程;第 11 章结合实例详细介绍了数据库应用系统的后期编译和打包发行过程。附录中为读者列出了 PowerBuilder 中的保留字和大量事件 ID 的含义解析,方便读者查询。

本书的编写具有以下几个特点:

(1) 内容涉及广泛。本书从软件工程开发的整体角度出发,介绍了应用 PowerBuilder 进行软件开发的一般过程,涉及比较全面的数据库应用开发知识,并通过实例加以说明。让读者在学习 PowerBuilder 软件的同时,掌握应用该软件进行应用开发的完整过程;

(2) 实用性好。本书是作者在总结长期的数据库应用开发经验,并参考大量资料后总结和积累起来的,对于部分知识(如控件、对象的使用、报表设计、应用程序打包发行等)的讲解融入了作者本人的经验和理解,使读者易于理解和应用;

(3) 理论与实例紧密结合。本书在介绍各部分理论知识时,都采用具有针对性的实例进行详细介绍,通过实例有助于读者对该知识或技术的理解与应用;

(4) 基础知识与高级技术结合。本书详细讲解了实际开发中常用的大量 PowerBuilder 基础知识,同时根据对开发经验的总结,对 PowerBuilder 的部分实用的高级开发技术进行了详细介绍,并通过大量的实例说明了在开发过程中的使用方法和技巧。

本书从数据库应用开发出发,注重概念理论与实际应用的结合,精心安排了大量具有针对性的实例对相关理论知识和开发技术进行说明,从软件开发整体角度对 PowerBuilder 软件的功能、使用技巧以及高级开发技术做了系统全面的介绍。本书既适合于 PowerBuilder 的初学者选作基础入门教材;更适合于 PowerBuilder 熟练者和爱好者加深理解时使用;也可供数据库应用开发人员在解决实际应用开发中的问题时查阅。

本书第 1,2,4 章由曹岩编写,第 6,9,10 章由张英杰编写,第 3 章由张英杰和曾高强共同编写,第 5,7,8,11 章以及本书附录部分由曾高强编写。

尽管本书在编写过程中,查阅了大量资料,并通过网络收集和总结了应用 PowerBuilder 进行软件开发中的大量开发技巧,同时融入了作者在软件开发中的经验。但由于作者水平有限,错误之处在所难免,望各位读者不吝赐教,作者在此深表感谢!

作者

2004 年 8 月

目 录

第 1 章 PowerBuilder 概述	(1)	3.1.3 语句分隔符(;)	(45)
1.1 PowerBuilder 特点及其开发环境	(1)	3.1.4 标识符	(45)
1.1.1 PowerBuilder 介绍	(1)	3.1.5 标号	(46)
1.1.2 PowerBuilder 的特点	(1)	3.1.6 特殊的 ASCII 字符	(46)
1.1.3 PowerBuilder 9.0 的新特性	(3)	3.1.7 空值(NULL)	(47)
1.2 PowerBuilder 集成开发环境	(4)	3.1.8 代词	(48)
1.2.1 PowerBuilder 安装	(4)	3.1.9 引用方法	(52)
1.2.2 PowerBuilder 9.0 集成开发环境	(5)	3.1.10 PowerBuilder 中的保留字	(52)
1.2.3 PowerBuilder 中的基本概念	(6)	3.2 数据类型	(54)
1.3 PowerBuilder 与面向对象编程	(7)	3.2.1 标准数据类型	(54)
1.3.1 面向对象的基本概念	(7)	3.2.2 Any 数据类型	(57)
1.3.2 PowerBuilder 中面向对象方法的应用	(8)	3.2.3 系统对象的数据类型	(60)
1.4 PowerBuilder 应用程序开发	(9)	3.2.4 枚举数据类型	(61)
1.4.1 PowerBuilder 适用开发范围	(9)	3.3 声明	(62)
1.4.2 应用程序开发的一般流程	(10)	3.3.1 变量的声明	(62)
第 2 章 数据库的连接与操作	(12)	3.3.2 常量的声明	(70)
2.1 数据库基础知识	(12)	3.3.3 数组的声明	(70)
2.1.1 数据库的基本概念	(12)	3.3.4 外部函数的声明	(75)
2.1.2 数据库的连接原理	(14)	3.3.5 存储过程的声明	(76)
2.2 数据库的连接方式	(15)	3.3.6 结构的创建与结构变量的声明	(77)
2.2.1 数据库的连接接口	(16)	3.3.7 小结	(78)
2.2.2 通过 ODBC 接口连接数据库	(17)	3.4 操作符与表达式	(78)
2.2.3 通过专用数据库接口连接数据库	(27)	3.4.1 算术运算符	(79)
2.3 事务对象与数据库的连接	(29)	3.4.2 关系运算符	(80)
2.3.1 事务对象的属性	(29)	3.4.3 逻辑运算符	(80)
2.3.2 事务对象的操作与管理	(30)	3.4.4 连接运算符	(81)
2.3.3 数据库连接参数的设置	(31)	3.4.5 圆点操作符	(82)
2.3.4 数据库连接实例	(33)	3.4.6 操作符的优先级别	(82)
2.4 数据库的操作	(35)	3.4.7 表达式及其数据类型	(84)
2.5 数据库建立与连接实例	(36)	3.5 PowerScript 语句	(85)
本章小结	(42)	3.5.1 赋值语句	(85)
第 3 章 PowerScript 语言	(43)	3.5.2 条件语句	(85)
3.1 PowerScript 语言基础	(43)	3.5.3 分支语句	(87)
3.1.1 注释	(43)	3.5.4 循环控制语句	(88)
3.1.2 语句连接符(&)	(44)	3.5.5 对象创建与释放语句	(93)
		3.5.6 程序终止与返回语句	(95)
		3.5.7 调用语句(CALL)	(96)

3.6 事件和函数	(97)	5.3.1 菜单对象的建立	(142)
3.6.1 PowerBuilder 中的事件	(97)	5.3.2 菜单对象的操作	(142)
3.6.2 PowerBuilder 中的函数	(98)	5.3.3 菜单对象的引用	(144)
3.6.3 函数和事件的调用	(100)	5.4 列表框类控件	(146)
3.7 PowerBuilder 常用函数	(101)	5.4.1 四种列表框类控件的比较	(146)
3.7.1 数值函数	(101)	5.4.2 列表框类控件的主要属性	(147)
3.7.2 字符(串)操作函数	(102)	5.4.3 列表框类控件的主要事件	(149)
3.7.3 日期/时间以及定时函数	(103)	5.4.4 列表框类控件的主要函数	(149)
3.7.4 数据类型检测与转换函数	(104)	5.5 视图类控件	(151)
3.7.5 文件与目录操作函数	(105)	5.5.1 ListView 控件	(152)
3.7.6 注册表操作函数	(109)	5.5.2 TreeView 控件	(157)
3.7.7 初始化文件函数	(111)	5.6 Tab 控件	(162)
3.7.8 其它函数	(112)	5.6.1 Tab 控件主要属性和函数	(162)
本章小结	(113)	5.6.2 Tab 控件的使用实例	(164)
第 4 章 SQL 语句及数据库操作	(114)	第 6 章 OLE 技术与 OLE 控件的使用	(168)
4.1 数据查询操作	(114)	6.1 OLE 概述	(168)
4.1.1 SELECT 查询语句	(114)	6.2 OLE 控件的定义	(169)
4.1.2 利用游标(CURSOR)查询	(115)	6.3 OLE 控件的控制	(171)
4.1.3 利用存储过程查询	(119)	6.3.1 OLE 控件的属性	(171)
4.2 其它操作语句	(120)	6.3.2 OLE 对象的激活	(173)
4.2.1 数据插入语句(Insert)	(120)	6.3.3 OLE 对象的编辑与修改	(173)
4.2.2 数据删除语句>Delete)	(121)	6.3.4 OLE 对象的关联方式	(173)
4.2.3 数据更新(Update)	(121)	6.4 OLE 控件的函数	(174)
4.3 Blob 数据的查询和更新	(121)	6.5 OLE 应用实例	(179)
4.3.1 SELECTBLOB 语句	(122)	第 7 章 DataWindow 与报表设计	(185)
4.3.2 UPDATEBLOB 语句	(122)	7.1 DataWindow 对象	(185)
4.4 PowerBuilder 中的动态 SQL 语句	(123)	7.1.1 数据窗口对象的创建	(186)
4.5 数据操作实例	(129)	7.1.2 数据窗口对象的显示风格	(192)
4.5.1 游标查询实例	(129)	7.1.3 数据窗口对象的设计	(197)
4.5.2 Blob 数据操作实例	(130)	7.1.4 数据列的设计	(204)
本章小结	(132)	7.2 复杂报表设计	(207)
第 5 章 典型对象和控件的应用	(133)	7.2.1 复合报表	(207)
5.1 应用对象	(133)	7.2.2 交叉报表	(210)
5.1.1 应用对象的建立	(133)	7.2.3 图形报表	(211)
5.1.2 应用对象的属性设置	(135)	7.3 DataWindow 控件和 DataStore 对象	(216)
5.1.3 应用对象编程	(136)	7.3.1 数据窗口控件的主要属性	(216)
5.2 窗口对象	(137)	7.3.2 数据窗口控件的事件	(217)
5.2.1 窗口的创建	(137)	7.3.3 数据窗口控件的使用	(217)
5.2.2 窗口的属性	(139)	7.3.4 DataStore 对象	(218)
5.2.3 窗口的类型	(140)	7.3.5 数据查询	(220)
5.3 菜单对象	(141)	7.4 创建动态数据窗口	(226)

7.4.1 创建动态数据窗口相关函数	226	9.3.1 建立 EAServer 客户端应用的步骤	289
7.4.2 创建动态数据窗口的一般过程	227	9.3.2 与 EAServer 服务器的连接	290
7.4.3 动态数据窗口对象的使用	227	9.3.3 生成 EAServer 代理对象	293
7.4.4 动态数据窗口创建实例	228	9.3.4 调用 EAServer 服务器组件方法	294
7.5 报表打印控制	229	9.3.5 客户端应用的其它工作	295
第 8 章 PowerBuilder 高级编程技术	238	第 10 章 PowerBuilder 的 Web 开发	296
8.1 PowerBuilder 中的文件目录操作	238	10.1 Web 应用概述	296
8.1.1 目录操作	239	10.1.1 Web 应用介绍	296
8.1.2 文件操作	240	10.1.2 Web 应用开发环境	298
8.2 操作注册表	245	10.1.3 Web 目标开发工具介绍	299
8.3 INI 文件的使用	250	10.2 开发 Web 应用	303
8.3.1 INI 文件格式	250	10.2.1 Web 开发基本步骤	303
8.3.2 INI 文件使用实例	250	10.2.2 Web 目标的创建	303
8.4 多媒体开发应用	253	10.2.3 建立调试配置文件	304
8.4.1 OLE 技术实现视频播放	253	10.2.4 定义连接配置文件	306
8.4.2 Windows API 编程实现多媒体开发	259	10.2.5 Web 目标的操作设计	307
8.5 动态连接库的调用与混合编程	259	10.2.6 Web 应用的构建与生成	309
8.5.1 DLL 简介	259	10.3 Web 开发总结	311
8.5.2 PowerBuilder 中 DLL 的使用	260	第 11 章 编译和发行可执行应用	312
8.5.3 DLL 调用编程实例	261	11.1 创建工程应用对象	312
8.6 API 函数编程实例	263	11.2 编译生成可执行应用	315
8.6.1 API 编程概述	263	11.2.1 准备工作	315
8.6.2 获取计算机信息	265	11.2.2 创建资源文件	316
8.6.3 隐藏/显示 Windows 任务栏	270	11.2.3 选择编码格式	317
8.6.4 目录获取	272	11.2.4 选择打包模型	318
8.6.5 设置窗口显示位置和顺序	273	11.2.5 编译生成可执行应用	319
8.6.6 定时关机程序	273	11.3 支撑文件及测试	320
8.6.7 控制应用程序只启动一次	277	11.3.1 可执行应用的支撑文件	321
第 9 章 分布式应用简介	278	11.3.2 可执行应用的测试	325
9.1 分布式应用与 EAServer	278	11.4 制作安装程序	326
9.1.1 结构体系概念	278	11.4.1 应用程序打包准备	327
9.1.2 EAServer 介绍	280	11.4.2 创建工程	327
9.1.3 PowerBuilder 对分布式应用的支持	282	11.4.3 添加文件及组件安装	332
9.2 建立 EAServer 服务器组件	284	11.4.4 编译生成安装文件	337
9.2.1 建立 Profile 文件	284	附录 1 PowerBuilder 中的保留字	338
9.2.2 建立 EAServer 组件	285	附录 2 PowerBuilder 中部分事件 ID(Event ID)	339
9.3 建立 EAServer 客户端	289	含义解析	339
		参考文献	343

1.1 PowerBuilder 特点及其开发环境

1.1.1 PowerBuilder 介绍

PowerBuilder 是美国 Sybase 公司的 PowerSoft 子公司(1995 年被 Sybase 公司收购)在 1990 年开发的客户/服务器(Client/Server;C/S)结构应用程序的前端开发工具。从 1991 年发布 PowerBuilder 1.0 至今,根据客户的需求和对其它技术的支持,不断开发和升级新的版本。2003 年 8 月,Sybase 公司已发布了最新的 PowerBuilder 9.0 版本。

PowerBuilder 是一种面向对象的、可视化、交互式数据库应用程序前台开发工具,一直被认为是一个易于使用、可升级、开发时间短且效率高的开发工具。PowerBuilder 用于开发 C/S 结构数据库应用程序的快捷性、简便性、高效率 and 先进性等优点一直受到软件开发者的赞赏。如今,随着 PowerBuilder 版本的不断升级,PowerBuilder 除了为用户提供更方便、更完善的开发环境以外,同时扩展和加强了 PowerBuilder 在分布式应用和 Web 应用方面的功能,让用户可以在熟悉的 PowerBuilder 环境下,使用相关技术创建功能强大的 Web 应用和分布式应用。

随着网络技术的不断发展和应用,对软件提出了新的要求。PowerBuilder 面对这种新的要求和市场环境,在不断完善作为 C/S 结构开发中各种实用功能的同时,重点增强了 PowerBuilder 在分布式应用开发、基于组件应用开发和 Web 应用开发方面的功能。从 PowerBuilder 7.0 以后的版本中,这些方面的功能都得到了不断的增强。

在不断追求效率和质量商业竞争环境中,商品更新速度不断加快。软件的更新周期更短,所以软件开发中,选择合适的开发工具对于提高开发效率和质量,并保证下一代产品的顺利更新是至关重要的。PowerBuilder 为数据库开发人员提供了很好的开发环境,程序开发人员可以在很短的时间里掌握开发工具的应用,在 PowerBuilder 所提供的各种先进的技术支持下,能以极高的开发效率完成优质的产品开发任务。

1.1.2 PowerBuilder 的特点

PowerBuilder 之所以具有强大的开发功能、极高的开发效率、广泛的应用范围等优势,就是因为它的诸多特点决定的。

1. 可视化开发环境

PowerBuilder 为开发人员提供直观可视化开发环境,随时可以进行方便的程序调试,在 OutPut 窗口可以详细查看调试时的出错信息提示,具有很强的人机交互功能。开发人员在使用 PowerBuilder 开发软件时,大大减小了开发的复杂程度和难度。

2. 面向对象编程

PowerBuilder 具有面向对象编程(Object Oriented Programming; OOP)的各种特性。可以利用面向对象编程技术的多态性、封装和继承等特点来提高系统的可重用性和可扩展性。例如窗口、菜单等对象的可继承性为程序开发人员减少了大量重复工作。

3. 事件驱动

PowerBuilder 采用事件(Event)驱动的开发方法,程序并不顺序执行,而是通过用户操作请求或者对系统的响应来完成程序的运行的。比如在窗口 Open 事件中编写的程序代码,将在该窗口打开时触发并执行程序;Clicked 事件将在鼠标点击该对象(或控件等)时触发后执行;应用对象或窗口对象的 Timer 事件则是响应系统时钟触发的事件等。PowerBuilder 为对象和控件都定义了大量的事件,一般情况下,这些事件可以满足开发人员的需要;PowerBuilder 还为用户提供了大量的事件 ID(Event ID),用户可以为它们定义用户事件(包括带 Event ID 的用户事件和不带 Event ID 的用户事件),为程序开发带来了很大的灵活性。

4. 功能强大的开发语言

PowerBuilder 采用 PowerScript 编程语言,其语法和其它语言类似,初学者易学易掌握。同时,PowerScript 还为程序开发人员提供了大量系统内置函数,在编写时可以直接调用,如窗口的打开(Open)和关闭(Close)函数,系统时间日期函数,注册表操作函数等。用户也可以根据需要采用该语言定义自己的函数。

在 PowerScript 语言中,提供 SQL 语句的嵌入式使用,为用户开发数据库系统时,对数据库的操作提供了极大的方便。

在 PowerBuilder 中,还提供了外部动态链接库的调用,可以在 PowerScript 语言中声明外部 DLL 库文件中的函数并加以调用,这大大扩展了 PowerBuilder 与其它语言实现混合编程的功能。同时,采用这种方法可以调用 Windows API 函数进行编程,这无疑给 PowerBuilder 在应用程序开发中增添了不少色彩。

5. 支持跨平台开发

PowerBuilder 是一种基于多平台的开发工具,可以在不同的系统平台下进行软件开发和发布。对于同一个项目,也可以在不同平台上完成不同功能模块的开发,最后实现各模块的集成。

6. 对多种数据库系统的支持

作为前台开发工具,PowerBuilder 可以支持几乎所有的后台数据库管理系统(如 Oracle、Sybase、DB2、MSSQL Server、Access 等)。可以使用标准的数据库接口(ODBC、JDBC 或 OLE DB 等)与数据库建立连接。同时,对于部分大中型数据库系统(如 MSSQL Server、Oracle 等)还提供了专用数据库接口,通过专用接口驱动程序建立起数据库与应用系统之间的直接连接。而 DataWindow(数据窗口)作为 PowerBuilder 的专利技术,为程序中实现与数据库

的数据交换提供了高效的解决方案。

7. 对多种开发方案的支持

PowerBuilder 工具一直被认为是开发传统客户/服务器(C/S)结构的最佳选择,其高效率、低成本、开发周期短等好处一直被人们所赞赏。随着 PowerBuilder 的发展和版本升级,在分布式应用、基于组件开发应用和 Web 应用方面的功能也不断得到增强。

1.1.3 PowerBuilder 9.0 的新特性

新推出的 PowerBuilder 9.0 在原版本的基础上,在很多方面都有了明显提高和改进。概括起来主要表现在以下几个方面:

1. 新增 JSP 快速应用开发目标

PowerBuilder 9.0 包括一个便捷的开发环境,用于编写 JSP Web 应用程序,它能够使许多开发工作自动化,并且允许配置 Web 应用程序。程序开发人员可以在 PowerBuilder 中利用向导或者其它 RAD(Rapid Application Development:快速应用开发)开发工具进行快速开发和配置一套 JSP(JavaServer Page)网络应用系统。

2. Web 服务的增强

开发人员不需具备深厚的 SOAP 协议和 WSDL 方面的知识基础,即可在 Microsoft .Net 或者其它框架结构下,创建、发布和完成客户端 Web 服务系统。

3. XML 数据窗口

可以直接从一个 XML 文档中导入或者保存数据,或把从任何数据来源中恢复的数据存储为完全的用户化 XML 文档,也可以把这些数据存储为 PDF 或 XSL 格式对象的文档或者字符串。

4. PowerBuilder 本地接口

PowerBuilder 捆绑了一个 C++ 类作为扩展类,在 PowerBuilder 应用中可以调用该类的方法,极大地扩展了 PowerBuilder 应用开发的功能。通过 Java 本地接口(Java Native Interface:JNI),Java 应用程序也可以与 PowerBuilder 虚拟机(PowerBuilder Virtual Machine:PBVM)进行通讯。

5. 对第三方应用服务器的支持

除了对现有的 Sybase EAServer 的支持以外,PowerBuilder 9.0 还可实现对 BEA Weblogic Server TM,IBM WebSphere 以及其它 J2EE 应用服务器的支持。通过 Wizard 为企业 Java Beans(EJBs)产生代理对象,同时通过代理对象可以从 PowerBuilder 中调用在 J2EE 服务器上的 EJBs。

6. PowerBuilder 文档对象模型

利用该文档模型可以定义怎样存取和操作 XML 文档。

Sybase 产品可存储或操作 XML 文件的文档对象模块(Document Object Model:DOM)。这项功能(PB DOM)用于从 PowerScript 源码中读取、编写、并操作标准格式的 XML。

7. PDF 格式报表的生成

PowerBuilder 9.0 可以提供两种方式保存 DataWindow 对象, 以使其能够直接生成 PDF 格式的报表。这种方式以默认的方式应用, 并可保存所有类型的 DataWindow 对象。

8. 源代码控制的提高

资源控制的增强可以帮助开发人员高效的管理大型的 PowerBuilder 工程, 并实现与 Rational ClearCase 和 Merant PVCS 的紧密集成。

9. OrcaScript 的应用

PowerBuilder 9.0 可以用类似英文的自然语言调用 Oracle 应用程序。新的源码控制集成能根据源控件直接生成目标对象。

10. PowerBuilder 与 PowerDesigner 的集成

可以快速产生一个应用及其所有代码的可视化继承, 利用拖放编程技术完成 N 层体系结构的重构, 并在 PowerBuilder 集成开发环境里有相应的组件支持。

1.2 PowerBuilder 集成开发环境

在介绍 PowerBuilder 开发功能之前, 本小节先介绍一下 PowerBuilder 工具的安装过程, 并对其开发环境作简要介绍。

1.2.1 PowerBuilder 安装

在 PowerBuilder 9.0 企业版的安装过程中, 将出现“Select Components”的对话框, 如图 1.1 所示。在该对话框中选择要安装的组件。

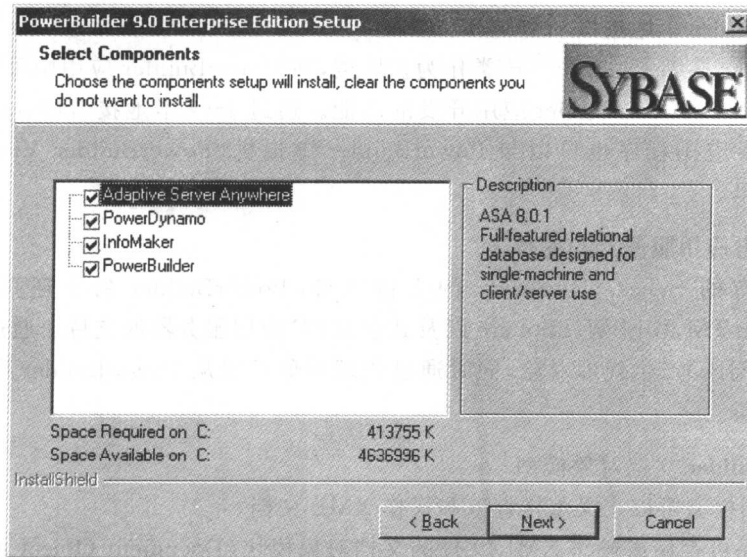


图 1.1 PowerBuilder 9.0 企业版中的组件

用户可选择的包括以下四个组件或模块：

(1) Adaptive Server Anywhere 8.0

ASA(Adaptive Server Anywhere)是安装 PowerBuilder 时可以选择安装的一个数据库, PowerBuilder 9.0 所带的 ASA 数据库为 8.0 版本,当选择安装 ASA 数据接口后,在 PowerBuilder 的运行环境中就可以完成 ASA 数据库的建立、数据维护和操作;

(2) PowerDynamo:可以生成“瘦客户机”Web 应用的动态网页服务器;

(3) InfoMaker:可以完成数据访问、管理和报表的工具软件;

(4) PowerBuilder:PowerBuilder 集成开发环境;

在该对话框中选择需要安装的组件,然后按照安装向导执行安装过程就可以完成 PowerBuilder 的安装。其安装过程简单,在这里不做详细介绍。

1.2.2 PowerBuilder 9.0 集成开发环境

安装 PowerBuilder 后,在程序中启动 PowerBuilder 集成开发环境,如图 1.2 所示。

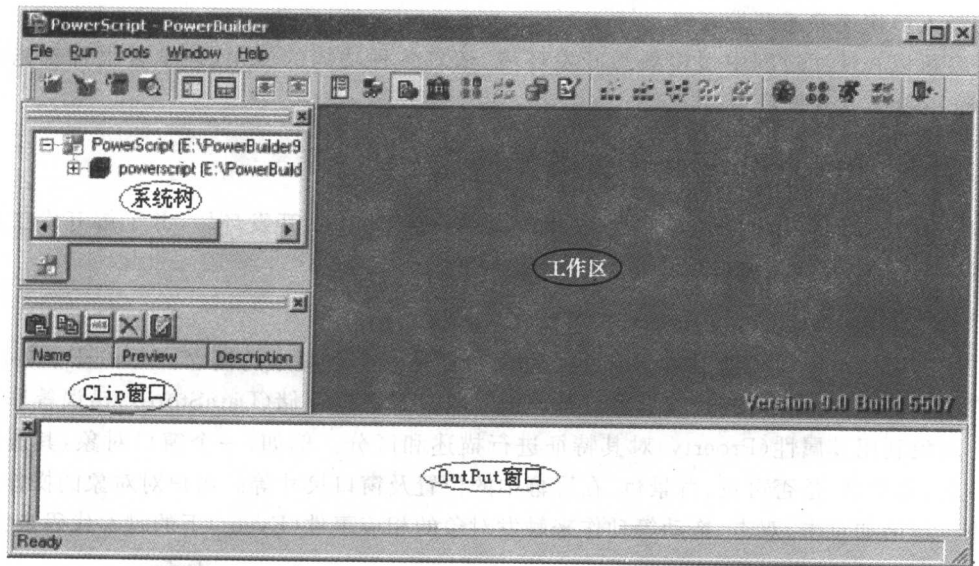


图 1.2 PowerBuilder 9.0 集成开发环境

其开发环境和 PowerBuilder 8.0 基本一样。在 PowerBuilder 开发环境下,可以打开工作区、系统树、Output 窗口和 Clip 窗口等几个子窗口。其中系统树中列出了当前工作空间中的 PowerBuilder 应用,以及各应用下的 PowerBuilder 库(PBL),各 PBL 下的对象以及对象的属性等;工作区用于打开各对象、属性等画板,供开发人员编程使用;通常在 Output 子窗口显示程序编译信息以及错误提示等信息;Clip 子窗口用来暂时存储经常使用的代码。

在工作区不打开任何画板时,PowerBuilder 环境中只有:File、Run、Tools、Window 和 Help 几个主菜单项,环境中的菜单栏随着打开对象画板的不同而不同,图 1.3 为打开窗口对象画板时的菜单。

各菜单项下有多个子菜单项,可以辅助开发人员进行快速开发。使用过 PowerBuilder 开

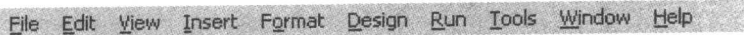


图 1.3 PowerBuilder 菜单

发环境的开发人员对这些菜单项的使用都非常熟悉,在这里不做详细介绍。

PowerBuilder 工具栏如图 1.4 所示。



图 1.4 PowerBuilder 中的工具栏

PowerBuilder 中的工具栏与菜单项相关,工具栏为开发人员提供更方便快捷的操作方式。在运行环境中,将鼠标移动到工具栏按钮上作短暂的停留后,将显示该按钮的提示信息,帮助初学人员快速熟悉开发环境。

读者可以在使用过程中慢慢熟悉开发环境,随着各种功能的使用,初学者很快就能掌握 PowerBuilder 9.0 开发环境。

1.2.3 PowerBuilder 中的基本概念

在应用 PowerBuilder 开发应用之前,先熟悉 PowerBuilder 开发环境,并了解开发工具中的一些基本概念,在以后的开发中会有所帮助。

1. 对象(Object)、属性(Property)、事件(Event)

在所有面向对象的编程工具中,对象(Object)都是一个最基本的概念。PowerBuilder 9.0 中提供了诸如窗口、命令按钮、数据窗口等可视对象以及数据存储(DataStore)、错误等非可视对象。对象利用其属性(Property)对其特征进行描述和区分。例如,一个窗口对象,其属性包括窗口名、菜单名、是否可见、背景色、在屏幕中的位置及窗口尺寸等。用户对对象的操作通过对对象进行诸如点击、双击、拖动等动作来触发对象的相应事件(Event)下的脚本代码来实现。

2. 窗口(Window)和控件(Control)

窗口是应用中所有可视对象的载体或容器,也是用户与应用程序直接交互的工作界面。PowerBuilder 9.0 中提供了丰富的控件,打开窗口画板后,可以在工具栏的控件箱中调用这些控件。如图 1.5 所示(鼠标在按钮上短暂停留,将出现控件按钮提示信息)。

按照这些控件各自的特性和功能,可以分为以下五种类型:

(1) 激活动作控件:包括 CommandButton(命令按钮)、PictureButton(图像按钮)和 PictureHyperLink(图像超链接控件)、StaticHyperLink(静态超链接控件)。

(2) 数据输入与显示控件:包括 DataWindow(数据窗口)、DropDownListBox(下拉列表框)、EditMask(编辑屏蔽)、Graph(图表)、ListBox(列表框)、List-

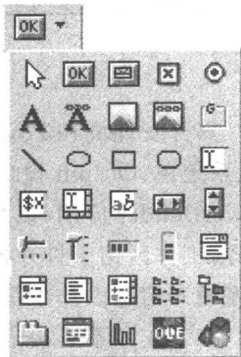


图 1.5 PowerBuilder 中的控件

View(列表视图)、MultiLineEdit(多行编辑框)、PictureListBox(图片列表框)、RichTextEdit(大文本编辑框)、SingleLineEdit(单行编辑框)、StaticText(静态文本)和 TreeView(树形视图)。

(3) 指示选择控件:包括 RadioButton(单选按钮)和 CheckBox(复选框)。

(4) 装饰界面控件:包括 GroupBox(组框)、Line(线)、Picture(图片)、Rectangle(矩形)、RoundRectangle(圆角矩形)、Oval(椭圆)。

(5) 其它辅助控件:包括 HprogressBar(水平进度条)、HscrollBar(水平滚动条)、HtrackBar(水平轨道条)、OLE (OLE 控件)、Tab(标签页)、VprogressBar(垂直进度条)、VscrollBar(垂直滚动条)和 VtrackBar(垂直轨道条)等。

PowerBuilder 为各控件定义了部分事件,用户也可以根据需要在控件上自定义若干事件。

3. 工作区(Workspace)和目标(Target)

在建立 PowerBuilder 应用之前,可以利用向导先建立起一个工作区,在建立起来的工作区里,可以同时打开多个 PowerBuilder 应用,使你可以同时在几个应用(Application)或目标(Target)上进行工作。用户每次只能打开一个工作区,但每个工作区中都可以加入任意多个目标。可以同时建立并展开多个目标,对其中的对象进行编辑。工作区文件和目标文件的后缀名分别为“pbw”和“pbt”。

PowerBuilder 目标有两种类型:

(1) PowerScript 目标:是一个 C/S 的(或多层的)可执行应用或一个服务器组件。

(2) Web 目标:是一个 Web 应用,包括 HTML 文件、脚本、图片、下载组件等建立一个 Web 站点所需要的一切元素。

4. 库文件(Library)

每个目标是由一个或多个库文件组成的,所有编译过的对象都被储存在库文件中。当执行应用时,PowerBuilder 9.0 从库文件中取得相应的对象进行处理,库文件的后缀名为“pbl”。

1.3 PowerBuilder 与面向对象编程

1.3.1 面向对象的基本概念

在面向对象编程(Object Oriented Programming:OOP)中,对象就是将状态(属性)与操作(行为)封装在一起形成的统一体,是包括数据和与之联系的方法的自包含模块。在面向对象编程中,利用可重用的类(Class)来提高开发效率。类具有属性和方法,类可以进行实例化并应用该实例执行应用处理逻辑。在面向对象编程中,有类、属性和方法等概念。

类(Class):是具有共同特征的所有对象的集合。在 PowerBuilder 中,类通常被称为对象,包括可视对象(如窗口对象、菜单对象、可视用户对象以及窗口中的控件等)和非可视对象(如 DataStore、Message、Transaction、Error 等对象)。

PowerBuilder 中的对象也被分为系统对象和用户自定义对象两种。系统对象由 PowerBuilder 定义,从基类 PowerObject 继承而来,而且它们是所有自定义对象的祖先,可以使用对象浏览器查看系统对象继承关系;用户自定义对象由用户在 Window、Menu 等不同画板中定义,它们都是从系统对象或另一个自定义对象继承而来的。

属性(Property):是描述对象性质的属性。在 PowerBuilder 中,属性包括对象性质描述(如窗口名、控件尺寸、颜色等)和在对象里定义的实例变量(实例变量可以看着是用户为对象定义的新的属性)。

方法(Method):在 PowerBuilder 程序开发中,方法包括事件和函数。事件可以是系统事件和用户事件;函数可以是系统函数、用户自定义函数以及外部函数。

用户可以在窗口画板、菜单画板、应用程序画板和用户对象等画板中定义对象。也以在窗口中添加控件,给对象属性赋初值,说明实例变量(Instance),定义对象的方法(事件和函数),并为事件和函数编写脚本。

举一个例子说明面向对象思想。对于汽车,都包括一些基本信息(方向盘、轮胎、发动机等),只是这些信息的参数不同而已,把所有这些共同的信息提取出来,作为汽车的一个对象。在以后对汽车的定义中,只需要使用一些描述信息在已定义好的“汽车对象”基础上,对其特征做具体描述就可以产生新的汽车描述信息。而在这个过程中,抽取出来的共同信息组成了“汽车对象”,这些信息则是“汽车对象”的属性,在利用这个对象进行新的汽车描述过程则是对象实例化过程,描述新的汽车中采用的描述信息则是对象的方法。

1.3.2 PowerBuilder 中面向对象方法的应用

PowerBuilder 作为支持面向对象的编程工具支持面向对象的三个基本原则,即继承性、封装性和多态性。

1. 继承性(Inheritance)

继承性表示对象可以从已有的对象类派生出新的对象,新对象继承了原对象(祖先)的所有特征(对象的属性和方法)。在 PowerBuilder 中包括可视控件、数据和程序代码。利用面向对象的继承性,可以节省编写代码的时间,提高程序代码的可重用性和程序模块的标准化。在 PowerBuilder 中,可以利用 Inherit 画板从指定的祖先对象(已建立的可视对象,如窗口、菜单、可视用户对象等)继承建立新的对象。如图 1.6 所示。

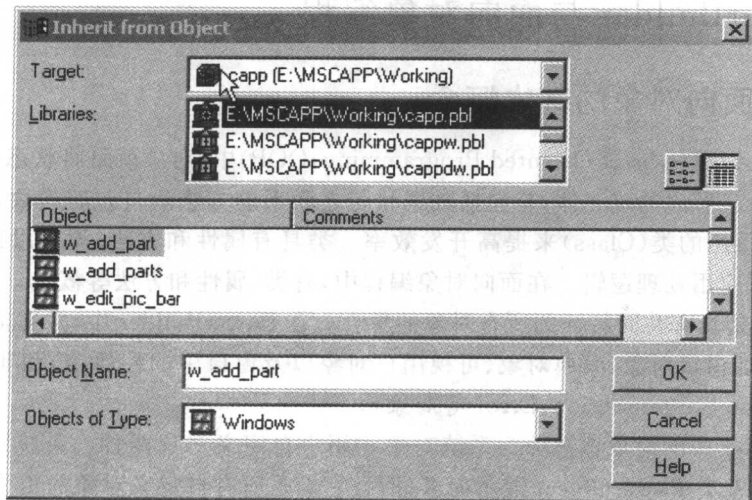


图 1.6 对象继承画板

2. 封装性(Encapsulation)

在面向对象中,一个对象本身含有它自己的数据和程序代码,某些属性或代码允许对象以外的其它对象访问和存取,而某些属性或代码是不允许使用的。这个特性也被称之为信息隐藏。换句话说,该对象定义了控制外部对象对其调用的方式。在 PowerBuilder 中,通过对数据存取权限(Access)和变量、函数等的使用范围(如全局变量、局部变量、实例变量、全局函数)的控制来实现封装。

3. 多态性(Polymorphism)

多态性是指同一名字的函数在不同的调用对象中具有不同的功能。在 PowerBuilder 中,不相关的对象可以定义同名函数,函数在各自的对象里执行相应任务;同时,还可以在同一对象中定义同名函数(PowerBuilder 中的部分系统函数也是如此),函数可以具有不同的参数,根据调用函数时给定的参数完成指定的任务。

1.4 PowerBuilder 应用程序开发

在选择开发工具进行应用程序开发之前,需要分析要开发的软件需要达到的开发目标,根据开发目标选择合适的工具、合适的开发模式。同时,应用程序开发流程也是软件工程中应该了解和掌握的。

1.4.1 PowerBuilder 适用开发范围

在进行软件开发之前必须对要开发的软件要实现的功能进行分析,然后根据要达到的目标去选择合适的开发工具,选择合适的开发工具对于提高开发效率、提高软件的维护性和可继承性都有极大的好处。因此,我们在利用选择 PowerBuilder 作为开发工具进行软件开发时也有必要了解它的适用开发范围。

PowerBuilder 的主要功能就是完成数据库系统的前台应用程序开发,对数据库的强大支持功能使得 PowerBuilder 具有极大的优势。

利用 PowerBuilder 主要可以开发三种体系结构的应用程序,即传统的客户/服务器(C/S)体系结构、分布式体系结构和 Web 体系结构。

1. C/S 结构应用程序

传统的 C/S 结构通常看成两层结构,数据库和 DBMS 存放在数据库服务器上;用户接口(表示层)和业务逻辑作为客户应用程序一同驻留在客户机上;客户端应用程序通过数据库接口(标准接口和专用接口)与数据库服务器完成数据交换。利用 PowerBuilder 开发这类应用程序非常方便,其开发过程也很简单,加上它为用户提供的数据库窗口技术可以大大提高应用程序的开发效率,缩短开发周期。

这种体系结构的应用程序在开发完成后,在服务器端安装数据库软件,设置数据库服务器,而在客户端分别安装用户软件,共同访问数据库服务器上的数据。随着客户端用户的增加,数据访问和传输速度将受到很大的影响,难以扩展至大型企业广域网或互联网(Internet)。同时,这种结构由于把用户处理逻辑和显示都放在客户端运行,增加了客户端工作载荷,也存在数据安全性和维护困难等问题。因此通常比较适合在小范围使用环境(比如一个部门的局