

SAMS

北京科海培训中心

PowerBuilder 4 应用程序开发指南

[美] Kent Marsh, Bruce Praunstein 著

廖卫东 李 奇 吴洁明 等译

廖卫东 审校

111332

清华大学出版社

(京)新登字 158 号

内 容 提 要

这是一本面向 PowerBuilder 技术咨询和开发人员的技术指南。

本书全面阐述了在使用 PowerBuilder 开发应用系统的过程中应注意的各种问题;从对象命名、开发策略到客户机/服务器程序设计等,系统地介绍了 PowerBuilder 4. x 的新功能;本书最具特色的是将一些应用示例分解成简单而易于理解的过程,从而使读者能够更快地掌握 PowerBuilder 开发技术;同时还介绍了面向对象和客户机/服务器方面的开发技巧,使读者高效地完成自己的开发任务。

PowerBuilder 4 Developer's Guide

Copyright © 1995 by Sams Publishing

All rights reserved. No part of this book shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, without written permission from the publisher.

本书英文版由 Macmillan 集团属下的 SAMS 公司于 1995 年出版。版权归 SAMS 所有。本书的中文版版权由 SAMS 授予北京科海培训中心和清华大学出版社合作共同出版、发行。未经出版者书面允许不得以任何方式复制或抄袭本书内容。

版权所有,盗版必究。

本书封面贴有 PRENTICE HALL 激光防伪标志,无标志者不得销售。

书 名:PowerBuilder 4 应用程序开发指南

译 者:廖卫东 李 奇 吴洁明

出版者:清华大学出版社(北京清华大学校内,邮编 100084)

印刷者:北京门头沟胶印厂

发 行:新华书店总店北京科技发行所

开 本:16 印张:42 字数:1021 千字

版 次:1996 年 2 月第 1 版 1996 年 2 月第 1 次印刷

印 数:00001~5000

书 号:ISBN 7-302-02161-9/TP·1027

定 价:75.00 元

前言

当 Kent 第一次请我为本书写前言时,我坐下来思考这本书是否能为开发者提供他们真正需要的和成功地进行客户机/服务器开发方面的指导。我们的首次交往是在网络上进行的,那是在 CompuServe 的 PowerSoft 电子论坛,当时 Kent 正在开发一个类库。当我看到 Kent 的作品时,我知道我已经找到了我想要的东西。在首次发表的 PowerBase 中,Kent 的 PowerBuilder 库便已显示出其他类库中极难见到的精妙之处。在交谈中,我们交流了哲学与理想。在 1994 年 5 月份的 PowerSoft 用户会议上,Kent 向与会者赠发他的 PowerBase 类库,这也让我大吃一惊。在这之前,有几家大公司的代表已向我问起过他的类库。一个建立在 Pacific Northwest 基础上的小公司怎么能希望为全美提供所需要的高水平服务与支持呢? Kent 却已经做到了。

接着就有了这本书。由于已被邀为 Kent 的另外一本书 (Secrets of the PowerBuilder Masters) 作过序,曾对 Kent 的知识和能力更加敬佩。他已直接用 PowerBuilder 解决客户机/服务器系统开发方面的问题。

在现代开发的竞技场上,“客户机/服务器”远远不只是一个名词。重要用途的程序每天都在被开发,作为开发者的大公司,大多数都在被一个新的现象所困扰,那就是 RAD(快速程序开发)。

RAD 是 90 年代出现的名词,它足以使系统开发的管理者头上冒汗。一想到客户机/服务器科技展现的光芒及其所带来的可扩充性,他们的眼前便会闪烁出这种开发技术所节省的大量金钱。在说服了财政官员、用户及开发人员后(对开发人员来说由于新科技=工作保障,故他们最易接受),一些人将采取措施改变现有的系统,一些人将重新设计他们的数据库,多数人将设立一些培训项目,却有更多的人踏上 RAD 之路却无法到达顶点,通常是“快速”一词被理解为“我们可以跳过这一步,因为 RAD 不需要这一步”,对 RAD 的强调应放在整个句子“快速的程序开发”上,而不是仅强调“快速”两个字。为了达到快速的目的,需要有比平常更详尽的计划,一种与众不同的计划:RAD 分析与设计。

按通常标准来讲,客户机/服务器技术还处在它的幼年时期,但我们不能等一项技术成熟之后才加以利用。如果那样,它也就失去了走向成熟的动力。支持这项技术的一些工具已经被开发出来。虽然工具一般比它们支持的技术还要不成熟,但其发展要快得多,因为它们更为具体,而且有卖方的利益促使他们成长。激烈的竞争使缓慢开发的产品,不管它们有多好,都会被弃置一边。

直到今天,面向对象的开发还被作为客户机/服务器的前景来招揽客户,但 OOP 的发展到头来却使很多一向性急的 I/S 公司愿望落空。经验告诉我,有一些投入 PowerBuilder 程序开发的公司招集了一周前还在搞 COBOL 的开发者,强行制订了最后期限,并雇用了一些高手做指导,但结果还是失败了。

另外一些用户采取了不同的方式:积木方式。他们对一些高层雇员重新进行客户机/服务器技术培训,它们先找一个好的顾问团体提供指导,并对系统进行周密计划,制订出了较

为宽松的时间表。这样成功就有了保障,也为将来的开发奠定了坚实的基础,并且使对步骤指导的制订与检验落在实处。除能通过详细计划节省开发费用外,用 PowerBuilder 等工具开发客户机/服务器的真正好处还在于在维护方面的省力。编写完美且说明完整的可重用程序也可以免去很多的维护方面的费用,但质量低劣的程序很少可以再利用,维护起来更是繁复。

可重用的基础就是类库。好的类库可以为所有的开发项目提供最大的可重用性,减少编码和维护的工作量,不强加任何设计决策给开发者,用起来简便易行。现在的类库大多能具备以上的某些特点,但很少具有全部。作为一个曾为某个在 Fortune 500(由《幸运》杂志评选的全球 500 家最大企业——译者注)之列的用户服务过的类库开发者,我亲历了设计、开发一个可重用类库所需付出的艰辛,而且参与过评定几个商业类库的工作。我的用户选择了从头开始建造他们自己的类库的道路。最后结果是:开发的类库为数据窗口之外的所有用过它的开发项目提供了 100% 的所需对象。为此付出的代价就是在学习道路上走了大弯路,对那些 OOP 新手来说更是如此。公司购买商业类库的秘诀就是:仔细评估现在类库,寻找能符合他们特殊需要的且有一定可重用性的对象,功能完整,资料详尽并有良好支持的类库。

在类库已选好或开发好之后,首先应在对象级对系统开发予以计划。否则,你将会在开发中步入艰难,或更糟糕地是,在开发已完成后才会知道哪里应该重用或哪里可以重用。成功地设计对象及对象之间的接口都要遵循这一原则,即使是有了这类计划,一些对象仍需要重新设计和/或再行开发,这不应被认为是计划的失败,因为即使在像 PowerBuilder 这样有力的技术下,也总能找到更有效更灵活的方法。最好的决策是随时进行重新设计与再行开发,而不要等到项目已完成(或对象已被其他几个项目重用),到那时再进行重新开发和设计就比较费事儿了。

任何项目开发组的最大财富都是细心的指导,适时地认真选择并使用恰当的技术。这样,加上采用辅导程序、培训和各种工具的应用,才能帮助开发者和 I/S 部门使他们的开发走向成功。

本书提供的技术适用于所有开发者,尤其能帮助那些急于完成项目的开发人员解燃眉之急。这些技术已在编程实践中得到验证。Kent 的 PowerBuilder 知识及面向对象的技术将有助于减轻初学者或高级开发者学习的弯路。

William Green

有关 CD-ROM

原英文版附带有一张 CD-ROM,内容包括书中正文部分涉及的所有程序的源代码,以及作者专为本书开发的类库和多媒体 ScreenCam 文件。需要者请与北京科海培训中心联系。

电 话: 2589259, 2569289

通信地址: 北京海淀路 82 号科海培训中心

邮政编码: 100080

鸣 谢

要是没有 PowerBuilder 广大开发者的支持和 PowerSoft 雇员、朋友、同事及家人的帮助,这本书是不可能问世的。

我使用 PowerBuilder 始于它的第一个版本。当时许多早期开发者正在拼命寻找编写可重用窗口的方法,大多数开发者甚至都没想到要设计应用的总体框架,当时最主要的心思在于如何把一些零散的东西组织起来,以完成有用的工作。一个英勇的 PowerSoft 雇员小组勇敢地向用户提供支持,他们不仅帮助解决 PowerBuilder 中出现的问题,而且还就如何建立对象提供有益的帮助,甚至帮助解决一些与 PowerSoft 无关的,诸如数据库、连网及 Windows 方面的问题。

我尤其不会忘记在早些年 Gregg Bjork 在 PowerSoft 支持热线上所提供的帮助。经过这许多年后,他仍然保持着对用户提供帮助时的那种不厌其烦的积极态度。即使现在,作为 Customer Services 的经理,Gregg 仍具有一种出色的服务态度。如果没有像 Gregg 那样的一批可以信赖的献身面向服务的支持人员,我们许多的早期用户可能早就放弃了一百次了。感谢你——Gregg,以及你的那些优秀的工作人员。

我还应感谢 PowerSoft 新产品开发部的经理 Bob Zureck,他对我写这本书起了促进作用。Bob 和我是在前年才认识的,但在这短短的时间内,他对我一直是一个激励。我感到我们俩有许多共同之处。Bob 特别喜欢向人们讲解 PowerBuilder 并帮助他们获得成功,而本书的目的也正在于此。我希望我的激励作用和帮助能赶上 Bob 的一半,同时我也祝 Bob 取得成功。

1994 年夏天我在俄勒冈州的 Portland Oregon 作为 PowerBuilder 技术负责人参加了 PacificCorp 项目(俄勒冈州和犹他州的电力设施)的开发,其间我认识了他们的项目经理 Tim Rilling。Tim 主要负责使工作顺利进展的一些重要事务及保护开发组免受政治及其他不利因素的干扰。Tim 对管理工作富有经验,总能在预算之内将工程按期交付。他享有“百事通”的美誉,当有什么棘手的事情必须完成时,你就去找 Tim 好了。Tim 管理项目的方式给我及其他人都留下了深刻印象。我们在人手不足的情况下,三个月就完成了任务。而完成这个项目对于大多数人员齐备的组至少需要用六个月的时间。在编写本书第 22 章“管理 PowerBuilder 开发项目”时,我向 Tim 去请教了管理者应有的远见卓识。Tim 又一次被证实为“百事通”,他的贡献对那一章极有价值。

PowerBuilder 第 4 版发表于 1994 年 12 月,作为 PowerBuilder 编码伙伴,我得以参加他们最初的程序设计,为这本书提供范例时,我使用的大多数是 β 版本,但它惊人地可靠,PowerSoft 开发者的出色成果使我能用其 β 版本的软件编写了几个程序,而这对我尽早使用第 4 版本的新特点及尽快获得用这些新特点编程的经验大有帮助。尤其感激 Karen Berault,她负责对我的问题予以答复。若没有她对一些问题的研究,本书将失去一些重要而丰富的事实。

由于既要写书,要进行 β 测试,还要对付许多其他的职责,我没有时间将版本 4 中所有

的新特点一一查清。多亏了 PowerBuilder 的一个开发伙伴及朋友 Fred Reiner 的帮助。我在此对他的卓越的工作及贡献表示感谢。

本书有一些章节涉及到了面向对象的开发技术。我的尊敬的同事及朋友 William Green, 是 PowerSoft 组的成员, 本身又是一个面向对象的开发者。当我向他说起我需要获得一些别人对面向对象开发的见解时, 他在百忙之中抽空帮了我的忙。他花了相当的时间和心血把他的面向对象开发经验贡献给本书, 使本书的内容大大丰富。

虽然他们都没有在本书中写一个字, 但正因为有他们的爱和支持才能有这本书的形成。他们就是我的妻子 Sherie 以及我们的孩子 Ryan, Rebecca 和 Rosalind。我呆在屋子里一写就是好几个钟头, 但从没听到他们为老见不到丈夫或爸爸而抱怨过。他们肯定为我能完成这本书而做出过许多个人的牺牲, 而且其中大多数我或许永远不会知道。感谢你们, 我的家人。

Kent Marsh

作者简介

1. Kent Marsh 是 Millennium 公司总裁/CEO。自从 PowerBuilder Version 1 问世以来,他一直是 PowerBuilder 开发者、顾问及指导。他是一名持证 PowerBuilder 开发者 (CPD), 且有很长的用客户机/服务器技术成功开发商业软件的历史, 本书介绍了他开发的 PowerBase 类库。Millennium 公司是 PowerSoft 的 CODE 伙伴、顾问伙伴并负责 PowerBuilder 的技术培训。
2. Bruce Braunstein 是《PowerSoft 软件开发(PAD)》杂志的编辑, 曾出版不少科技书和科幻书。他以前还在《FoxMasters》杂志及 Pinnacle Publications 的 DBF 杂志当过编辑。现在致力于 PAD 杂志作为 PowerSoft 协会为用户及时提供 PowerSoft 的技术信息。

导 引

很久以来我就觉得应该有一本全面介绍如何开发 PowerBuilder 软件的专著。在 1991 年 PowerBuilder 第一版发行后我就开始了 PowerBuilder 软件开发的专职工作,还记得第一次坐在终端前想用 PowerBuilder 编写东西时的情景。尽管我已用客户机/服务器技术及关系数据库进行了好几年的软件开发工作,而且也学习了 PowerBuilder 的初级课程,但那一刻我突然意识到我的知识是多么的欠缺,对 PowerBuilder 应用开发的支持是多么的稀少。那时,PowerBuilder 刚刚问世,它的开发者极少,而且大都没有经验。我认识的人中没有一个人编出过成功的 PowerBuilder 应用软件。人们都在不断的试验、出错、创新并不断返工。我当时多么希望有人能写一本实用的关于如何开发 PowerBuilder 应用程序的书啊,但我从没想到几年之后我自己能有机会做这项工作。

这本书集中了我自己在开发 PowerBuilder 应用系统的过程中积累的经验。书中介绍的方法确实管用,这已经被众多艰巨的项目开发所验证。读者可以直接利用其中的很多东西,或对其加以改进。这些年中,我一直把每个新项目都作为巩固现有技术和开创新技术的机会,这使我开发 PowerBuilder 软件的水平不断提高。我可以告诉读者怎样快速开发出可靠、易于维护且高效的 PowerBuilder 软件。

本书介绍的是如何成功地开发应用软件。在我看来,成功的应用软件应达到以下要求:

- 能满足项目的功能需求。任何项目的基本目标都是要首先满足项目赖以生存的商业要求,然而许多项目过于注重枝节问题。开发者浪费了时间、金钱和情感,却忘了他们最初的目标。
- 在合理的时间内完成。造成项目不能及时完成的责任无疑在于项目管理的缺陷。缺乏计划、用人不当、开发人员各持己见、设备不齐、资料不全、培训不佳等等责任全在管理者。懂得如何开发 PowerBuilder 应用软件可以提高一个原本很普通的管理者的形象,但即使天赋最高的人也不一定能做到管理有方。
- 占用合理的资金投入。具体怎样的投入才算合理可以讨论。然而,这本书中阐述的开发方法可帮助你迅速、高效地完成任何应用系统的功能需求,因此可以减少开发者的人数,并降低开发费用。当然对“一个顶三个”的开发者要比“三个顶一”的开发者支付更高的报酬。
- 所开发的应用软件必须高效、易维护且受用户赞扬。要达到技术上的高效且易于维护意味着这两方面均不能低于用户的期望。这两者不仅都是可以达到的,而且都是应该努力达到的。
- 结果要比预期的好。我在接受每个项目时都向自己挑战:要在低于预定时间内做出比预期多的成果。PowerBuilder 开发经常会达到比预期要好的结果,因为开发平台提供了那么多大多数设计者所意想不到的有力支持。此外,要达到比预期好的结果通常还需要开发者花费时间和精力去全面地了解开发项目。
- 富有情趣。人人都承认项目开发是艰巨的工作,但这并不意味着其间毫无乐趣可言。

当一群有才干的、专心致志的开发人员在管理完善、不受无理限制的环境下工作时，他们将超常发挥，自得其乐。一个成功的开发组中通常士气高涨。

本书假设你已学完了 PowerBuilder 初级课程或者拥有相当的经验。你在这方面的经验越丰富，书中的介绍和范例对你就越有用。你应已接触过 PowerBuilder 环境并已用过它的 Painter。本书还假设你已经具有 SQL 语言方面的基本知识，并且对开数据窗口和创建数据库表之类的事情不感陌生。

本书大多数章节讲述如何编写 PowerBuilder 程序，它不是 PowerBuilder 用户手册的重复，也不涉及那些可以在其他书中找到答案的肤浅问题，如面向对象的分析、数据库分析、需求定义、项目管理等。本书是一本有关 PowerBuilder 开发的实用指导书，整本书中，针对每个相关题目都给出了一些建议和提示，以丰富读者的阅读。

最近，在用类库进行 PowerBuilder 应用开发方面取得了很大成绩。从 1992 年春季开始，我就一直用自己的类库编写 PowerBuilder 程序。记得是 1992 年，在从达拉斯 PowerBuilder 会议飞回家的途中，我开始草拟我所知道的第一个 PowerBuilder 类库。从那以后，我已为不同的开发组编写了很多版本的类库，并应用于大型项目的开发。这些版本的基本概念和优点没有什么区别，但每次改写都增加了大量的新功能。我的主要目的在于创建一个拥有所有各种优点并且最易于使用的类库，以作为 PowerBuilder 的得力助手和补充。

没有比粗劣的类库更能使程序开发陷入困境的了。不幸的是，当今市场上竟有一批出奇粗劣，然而却被广为宣传的类库，这些类库只能使开发过程更复杂，而系统性能更为糟糕，使用它们只能给 PowerBuilder 项目开发带来巨大损害。我在这本书中包含了 PowerBase，目的之一就是告诉读者有的类库确实好用。不是所有的类库都难掌握，也不是所有的类库都会使程序运行缓慢。理想的类库应该使开发人员不必再为父类对象中已解决过的一般性问题再伤脑筋，而让他们把精力集中在项目的特殊要求上。

PowerBase 类库性能全面，Millennium 公司的开发人员已用它开发了不少重大项目。本书第 11 章和第 12 章中的练习使用了 PowerBase 库，这使你不用考虑许多关于应用的结构框架方面的问题（此问题全书都有论述），从而迅速地建立对象。不管在编程中是否用 PowerBase 作为基础库，这个信息对你都是有用的。

在写这本书的过程中，我把这个库的拷贝送给几个 PowerBase 新手，以测试第 11、12 章中的练习。信不信由你，其中有几个测试者后来竟把这个库用作他们产品开发的基础。他们掌握之快让我吃惊，他们也非常高兴能有如此实用的工具。我希望读者也能有类似的体验。

做一个 PowerBuilder 开发人员，对我来说是一个非常收获的经历。PowerBuilder 产品的功能越来越强，同时也越来越复杂。因而对每一个开发者来说，要想掌握它的每一个方面即便不是不可能，也是难上加难了。我希望这本书对你的 PowerBuilder 开发工作有所帮助，也希望这本关于如何开发 PowerBuilder 应用程序的实用书能从整体上增加 PowerBuilder 应用系统的数量并提高它们的质量。

Kent Marsh

目 录

第1部分 入门

前言.....	(1)
鸣谢.....	(3)
作者简介.....	(5)
导引.....	(6)
第1章 PowerBuilder 开发过程介绍	(1)
1.1 小结	(3)
第2章 成为有能力的开发人员	(4)
2.1 聪明的学习方法——边用边学	(4)
2.2 越过学习曲线	(4)
2.3 快捷键和热键	(6)
2.4 合理的设置配备	(8)
2.5 理想的工作环境	(9)
2.6 成功的人员组合	(9)
2.7 其他资料	(10)
2.8 降低效率的因素	(10)
2.9 小结	(11)
第3章 事件驱动程序的设计方法	(12)
3.1 目标	(12)
3.2 设计窗口对象	(14)
3.2.1 窗口属性	(19)
3.2.2 窗口事件	(26)
3.2.3 窗口函数	(35)
3.2.4 多态性	(45)
3.2.5 窗口控制	(46)
3.3 编码数据窗口控制	(49)
3.3.1 编辑框	(51)
3.3.2 数据窗口属性	(53)
3.3.3 数据窗口事件	(55)
3.3.4 数据窗口函数	(59)
3.4 PowerSoft 程序设计	(65)
3.4.1 保留字	(65)
3.4.2 变量声明	(67)

3.4.3 变量辖域	(70)
3.5 小结	(78)

第2部分 建立 PowerBuilder 应用程序

第4章 使用应用框架进行面向对象的开发

(80)

4.1 基础库	(80)
4.2 接口父类库	(81)
4.3 具体类库	(82)
4.4 一个针对应用对象的基础库	(82)
4.5 接口父类库	(84)
4.6 具体类库	(86)
4.7 服务器对象及协作类	(88)
4.8 面向对象开发的益处	(88)
4.9 小结	(89)

第5章 总体设计

(90)

5.1 MDI 与 SDI	(90)
5.2 应用安全性	(91)
5.2.1 控制应用的访问权限	(92)
5.2.2 如何控制进程的访问权限	(93)
5.2.3 控制数据访问	(94)
5.2.4 协同应用和数据库版本	(96)
5.3 用户界面标准	(99)
5.3.1 直观的应用	(100)
5.3.2 一个安全的应用	(104)
5.4 报表需求	(107)
5.4.1 报表分类	(108)
5.4.2 报表规范表格	(110)
5.4.3 报表的创建	(111)
5.4.4 报表展示	(111)
5.4.5 报表组织	(113)
5.4.7 报表的权限	(113)
5.4.6 报表的权限	(114)
5.5 系统需求	(114)
5.6 菜单系统设计	(115)
5.7 企业范围的考虑	(116)
5.8 小结	(117)

第6章 命名和编程指南

(118)

6.1 两字母应用辖域标识(TLASID)	(118)
-----------------------------	-------

6.2 命名数据库对象	(119)
6.2.1 表和视图	(120)
6.2.2 字段	(121)
6.2.3 触发器	(122)
6.2.4 存储过程	(124)
6.3 命名 PowerBuilder 对象	(124)
6.3.1 库的命名	(125)
6.3.2 窗口命名	(125)
6.3.3 数据窗口对象命名	(126)
6.3.4 菜单命名	(127)
6.3.5 用户对象命名	(128)
6.4 PowerScript 编程准则	(129)
6.4.1 变量的命名	(129)
6.4.2 函数的命名	(132)
6.4.3 script 程序头	(133)
6.4.4 变量声明	(135)
6.4.5 变量访问声明	(135)
6.4.6 编码标准	(137)
6.5 扩充属性	(140)
6.6 小结	(140)
第7章 设计数据库接口	(141)
7.1 锁定	(142)
7.1.1 不锁定	(142)
7.1.2 悲观锁定	(142)
7.1.3 乐观锁定	(142)
7.1.4 乐观锁校验字段	(143)
7.2 事务处理	(145)
7.3 错误处理	(147)
7.3.1 乐观锁定错误	(147)
7.3.2 悲观锁定错误	(148)
7.3.3 数据完整性错误	(148)
7.3.4 数据约束错误	(148)
7.4 触发器的使用	(149)
7.5 触发器存在的问题	(149)
7.6 突发性表错误	(150)
7.7 使用存储过程	(150)
7.8 实施数据库约束	(151)
7.9 数据库角色的使用	(152)
7.10 小结	(153)
第8章 PowerBuilder 中面向对象的构件块	(154)
8.1 子类	(157)

8.1.1 实例	(157)
8.2 搜索窗口	(158)
8.2.1 选取列表	(159)
8.2.2 查阅窗口	(161)
8.2.3 查询窗口	(161)
8.2.4 向下分级(Drill Down)	(163)
8.3 数据录入窗口	(164)
8.3.1 基本录入窗口	(167)
8.3.2 子集录入窗口	(169)
8.3.3 纲要/明细窗口	(169)
8.3.4 匹配程序窗口	(171)
8.3.5 多表录入窗口	(171)
8.3.6 转换器窗口	(173)
8.3.7 分布式更新	(173)
8.4 导航和选项窗口	(175)
8.4.1 导航窗口	(175)
8.4.2 用户参数选择	(176)
8.4.3 选项窗口	(176)
8.5 特殊窗口	(177)
8.5.1 Splash 窗口	(177)
8.5.2 About 窗口	(178)
8.5.3 登录窗口	(182)
8.6 全局函数	(183)
8.6.1 有效性验证	(186)
8.6.2 过程	(188)
8.6.3 计算	(189)
8.6.4 进程	(190)
8.7 用户对象	(190)
8.7.1 标准可视用户对象	(191)
8.7.2 自定义的可视用户对象	(191)
8.7.3 标准类	(191)
8.7.4 自定义类	(191)
8.8 应用报表	(192)
8.9 菜单	(193)
8.10 小结	(195)
第9章 自底向上创建应用程序	(196)
9.1 数据库对象	(196)
9.1.1 开发人员和数据管理员	(197)
9.1.2 表	(198)
9.1.3 约束	(202)
9.2 触发器与存储过程	(206)
9.2.1 触发器	(206)

9.2.2 存储过程	(207)
9.3 索引	(207)
9.4 优化器(Optimizers)	(208)
9.4.1 基于规则的优化器	(209)
9.4.2 基于代价的优化器	(209)
9.5 扩充属性	(210)
9.5.1 编辑风格	(211)
9.5.2 显示格式	(213)
9.5.3 有效性验证	(214)
9.6 应用程序的库结构	(218)
9.6.1 库内对象的命名	(219)
9.7 小结	(221)
第10章 辅助数据窗口	(223)
10.1 下拉数据窗口	(223)
10.1.1 一个简单的下拉数据窗口对象	(224)
10.1.2 创建一个下拉数据窗口编辑风格	(229)
10.1.3 其他更复杂的数据窗口	(234)
10.2 编码表维护数据窗口	(239)
10.3 管理表数据窗口	(241)
10.3.1 建立一个管理数据窗口对象	(242)
10.4 小结	(251)

第3部分 应用程序范例

第11章 一个公司/名册应用系统	(252)
11.1 详细设计	(253)
11.1.1 规格说明分析	(254)
11.1.2 对象仓库	(260)
11.1.3 对象索引	(260)
11.1.4 可提交的详细设计报告	(261)
11.2 创建阶段的开始	(261)
11.3 实现进度安排	(263)
11.4 创建高级对象	(264)
11.5 应用库/对象/MDI 框架/MDI 菜单	(265)
11.6 建立扩充属性	(271)
11.7 显示格式	(271)
11.8 编辑风格	(272)
11.8.1 有效性验证	(281)
11.9 创建数据窗口对象	(283)
11.10 创建接口父类对象	(285)

11.11	状态码录入窗口——w_state_de	(289)
11.12	类型码录入窗口——w_type_de	(293)
11.13	公司数据录入窗口——w_comp_de	(295)
11.14	名册录入窗口——w_cont_de	(304)
11.15	把名册录入窗口连接至公司录入窗口——w_comp_de	(307)
11.16	公司记录查询窗口——w_comp_lu	(324)
11.17	增加屏幕打印报表	(335)
11.18	数据窗口屏幕打印报表	(336)
11.19	合成数据窗口屏幕打印报表	(336)
11.20	“Print Screen”屏幕打印报表	(336)
11.21	小结	(341)
第 12 章	销售订单录入应用	(344)
12.1	建立销售订单录入应用的框架	(346)
12.1.1	设置销售订单录入应用环境	(347)
12.1.2	建立“应用打开”事件的对象	(351)
12.2	可选的多表录入窗口	(369)
12.3	优化子数据窗口	(376)
12.3.1	bw_DropDowns 的函数	(377)
12.4	雇员数据录入	(379)
12.5	建立一个用于继承的 Master/Detail 数据录入窗口	(391)
12.6	小结	(424)
第 13 章	应用报表	(427)
13.1	创建报表的完整过程	(427)
13.1.1	建立 SQL 语句	(427)
13.1.2	设计报表布局	(429)
13.1.3	预览(测试)报表	(431)
13.2	报表的组织	(432)
13.3	报表命名法	(433)
13.4	PowerBuilder 4 新增的报表性能	(434)
13.4.1	嵌套报表	(435)
13.4.2	合成数据窗口报表	(440)
13.4.3	存储报表文件	(443)
13.5	提高基于 SQL 的报表的效率	(446)
13.5.1	如何管理系统代价高的报表	(446)
13.5.2	一个高效率的数据库环境	(447)
13.5.3	优化程序的使用	(447)
13.5.4	优化连接操作	(448)
13.5.5	优化“并”操作	(448)
13.6	如何把报表结合进应用程序	(450)
13.6.1	报表选择	(450)

13.6.2 报表检索参数录入	(453)
13.6.3 报表“展示”选项	(455)
13.7 PSR 浏览程序	(455)
13.8 应用报表实例	(456)
13.9 标准报表标题对象	(456)
13.10 小结	(458)

第4部分 高级话题

第14章 DBMS 事务管理 (460)

14.1 DBMS 事务对象	(460)
14.1.1 DBMS 连接	(460)
14.1.2 SQL 执行的返回数据	(461)
14.2 PowerScript 的事务管理语句	(463)
14.2.1 COMMIT 和 ROLLBACK	(464)
14.2.2 CONNECT 和 DISCONNECT	(465)
14.3 多个事务对象的处理	(465)
14.3.1 数据窗口的事务函数	(467)
14.4 逻辑事务	(474)
14.4.1 更新多个数据窗口	(475)
14.5 并发控制	(477)
14.5.1 乐观锁与悲观锁的比较	(477)
14.5.2 使用一个乐观锁定字段	(478)
14.6 多窗口事务	(480)
14.6.1 多数据窗口	(480)
14.6.2 子窗口和弹出式窗口	(480)
14.6.3 共享的数据窗口	(480)
14.6.4 嵌套的响应窗口	(480)
14.7 小结	(480)

第15章 客户机/服务器计算模式 (482)

15.1 经济因素	(482)
15.2 技术因素	(482)
15.3 企业方面的要求	(483)
15.4 数据处理方面的要求	(483)
15.5 可用性方面的要求	(484)
15.6 客户机/服务器程序设计	(484)
15.6.1 减少数据库请求的次数	(485)
15.6.2 尽可能减少返回结果	(485)
15.6.3 利用存储过程来优化性能	(486)
15.6.4 集中实施企业规则	(486)

15.7 独立于数据库的应用程序.....	(489)
15.8 小结.....	(491)
第16章 数据有效性验证	(492)
16.1 利用控制来限制选择.....	(492)
16.2 编辑风格.....	(494)
16.2.1 键入式编辑风格	(494)
16.3 数据窗口字段的有效性验证.....	(495)
16.3.1 在输入时验证数据的有效性	(495)
16.3.2 在更新前验证数据的有效性	(496)
16.3.3 必备域	(496)
16.3.4 自定义有效性验证函数	(497)
16.4 数据库有效性验证.....	(499)
16.5 小结.....	(501)
第17章 数据窗口编码的高级技术	(502)
17.1 SQL 预览	(502)
17.2 仅取所需.....	(503)
17.3 数据窗口缓冲区.....	(504)
17.4 记录/字段的状态	(506)
17.5 使用子数据窗口.....	(507)
17.6 动态数据窗口	(509)
17.6.1 创建动态数据窗口	(509)
17.6.2 动态数据窗口的缺省设置	(511)
17.7 Modify()和 Describe()函数	(512)
17.7.1 多个 Modify()和 Describe()语句的组合	(512)
17.7.2 描述字段的数据类型	(513)
17.7.3 动态地创建并撤消数据窗口里的对象	(513)
17.7.4 数据窗口标题颜色	(515)
17.7.5 改变正文属性	(515)
17.7.6 隐藏/显示字段和标号	(516)
17.7.7 修改字段的制表键顺序	(517)
17.7.8 改变数据窗口的 SQL 语句	(519)
17.8 把数据源由 SQL SELECT 改变为存储过程	(522)
17.9 查询模式	(523)
17.9.1 查询模式编辑风格	(525)
17.9.2 显示查询结果	(525)
17.9.3 关闭查询模式	(525)
17.9.4 在别的数据窗口里显示结果	(526)
17.9.5 查询模式下的 QuerySort 选项	(526)
17.10 数据窗口层次	(526)
17.11 将 Enter 键用作 Tab 键	(528)