

PowerBuilder

快速入门

罗野 刘戈扬 周健 编著

PB



电子科技大学出版社

POWERBUILDER

快速入门

罗野 刘戈扬 周健 编著

电子科技大学出版社

内 容 提 要

PowerBuilder是基于MS-WINDOWS系统上的一种数据库开发系统,它支持客户/服务体系结构,是目前基于MS-WINDOWS的较好的数据库开发软件,在我国,该系统已广泛流行,该书以实用性为主,通过用PowerBuilder开发的一个简单应用实例,引导读者快速掌握PowerBuilder开发之精髓。全书由浅入深,重点介绍了表创建,扩展信息和数据的添加,应用对象的创建,窗口、数据窗口的创建添加,菜单的添加,数据库的维护,应用程序执行文件的创建等。本书既可作为初学者的入门读物,也可作为专业人员的参考手册。

POWERBUILDER 快速入门

罗 野 刘戈扬 周 健 编著

*

电子科技大学出版社出版发行

(中国成都建设北路二段四号) 邮编 610054

成都印刷一厂印刷

新华书店经销

*

开本 787 × 1092 1/16 印张 24.5 字数 596 千字

版次 1997年4月第一版 印次 1997年4月第一次印刷

印数 1—3000册

ISBN 7—81043—686—4/TP.277

定价: 28.00元

前 言

信息系统管理(MIS)及数据库应用系统的开发是计算机系统重要的应用,因而目前的数据库开发系统种类较多,其市场竞争也很激烈,其中 Powersoft 公司 PowerBuilder 一经推出便以其先进的思想和高性能的集成开发环境赢得了广大数据库应用软件人员的青睐。Powerbuilder 基于客户/服务的体系结构,可以和市场上各种流行的数据库系统软件相连接;Powerbuilder 采用了面向对象的先进应用程序开发方法,有利于软件的维护和继承;Powerbuilder 率先提供了可视化的数据库应用开发环境,用户可以直接通过图形化的交互操作,实现大部分的界面设计。在我国,该工具已成为多数大型数据库应用系统的公共前端开发工具。目前,我国各大城市最新启用的“火车售票系统”的前端机应用程序即是使用 PowerBuilder 3.0 开发的。

本书的作者从事了多年的数据库应用系统设计,并使用 PowerBuilder 开发多个大型的应用系统,积累丰富的经验,作者也希望通过此书能使更多的用户快速掌握这一先进的开发工作,并真诚地希望广大同行能对本书提出宝贵意见。

本书的出版离不开许多的朋友的鼓励和督促,他们是:孙林、缪世明、李仲庆、杨成伟、陈彤、侯俊、余鹏、范海生等。还要特别感谢张徐亮老师对我们的支持和帮助。

特别感谢我们的编辑周萍,没有她的帮助,本书也许永远仅仅存在于我们的脑海中。

编著者
1997.4

目 录

序章 编写本书的目的	1
0.1 本书内容简介	2
0.2 有关 Windows 的基础知识	4
0.2.1 窗口的构成	4
0.2.2 在 Windows 中启动应用程序	6
0.2.3 前台程序和后台程序	8
0.3 数据库管理系统简介	9
0.3.1 概述	9
0.3.2 数据库管理系统的体系结构	9
0.3.3 C/S 数据库前端应用程序开发的特点	12
第一章 创建表	14
1.1 打开数据库画板	14
1.1.1 启动 PowerBuilder	14
1.2 创建新的数据库	19
1.3 创建新的表	21
1.3.1 打开 Create Table 对话框	22
1.3.2 给表命名	23
1.3.3 定义 days (班期) 表列	24
1.3.4 定义 depcity (起飞城市) 表列	25
1.3.5 定义 arrcity (到达城市) 表列	26
1.3.6 定义 deptime (起飞时间) 表列	28
1.3.7 定义 arrtime (到达时间) 表列	29
1.3.8 定义 aircode (航空公司代码) 表列	29
1.3.9 定义 fltno (航班代号) 表列	31
1.3.10 定义 stop (经停) 表列	32
1.3.11 定义 class (等级) 表列	33
1.3.12 定义 a/c (飞机类型) 表列	34
1.3.13 定义 mark (注释) 表列	35
1.3.14 定义 price (价格) 表列	36
1.3.15 定义关键字	37
1.3.16 生成表	37

第二章 加入扩展	40
2.1 加入扩展属性信息	40
2.1.1 为 days 列定义(Header)题头和(Lable)标签	41
2.1.2 为 depcity 列定义(Header)题头和(Lable)标签	43
2.1.3 为 arrcity 列定义(Header)题头和(Lable)标签	44
2.1.4 为 deptime 列定义(Header)题头和(Lable)标签	44
2.1.5 为 arrtime 列定义(Header)题头和(Lable)标签	45
2.1.6 为 aircode 列定义(Header)题头和(Lable)标签	46
2.1.7 为 fltno 列定义(Header)题头和(Lable)标签	47
2.1.8 为 stop 列定义(Header)题头和(Lable)标签	48
2.1.9 为 class 列定义(Header)题头和(Lable)标签	49
2.1.10 为 a/c 列定义(Header)题头和(Lable)标签	50
2.1.11 为 mark 列定义(Header)题头和(Lable)标签	50
2.1.12 为 price 列定义(Header)题头和(Lable)标签	51
2.1.13 定义 days 列有效性检查规则	53
2.1.14 定义 fltno 列有效性检查规则	54
2.1.15 定义 price 列有效性检查规则	56
2.1.16 定义 price 显示格式	57
2.1.17 定义 deptime 编辑风格	58
2.1.18 定义 arrtime 编辑风格	59
2.2 向表里加入数据	60
第三章 创建应用对象	63
3.1 打开应用画板	63
3.2 创建和保存应用对象	64
3.3 为应用对象指定缺省图标	66
第四章 建造窗口	68
4.1 建造新的窗口	69
4.2 定义窗口风格	73
4.3 加入命令按钮	74
4.4 修改按钮属性	75
4.5 为按钮加入脚本	82
4.6 保存窗口	83
4.7 为应用程序加入脚本	86
4.8 运行程序	86
4.9 使用 C 和 C++语言建立基本窗口的例子	88
4.9.1 使用 C 语言建立窗口	89
4.9.2 使用 C++语言建立窗口	91

第五章 建造数据窗口	93
5.1 创建数据窗口	93
5.2 预览数据窗口对象	98
5.3 保存数据窗口对象	99
5.4 创建组	100
5.5 修饰数据窗口	102
第六章 PowerScript-PowerBuilder 的脚本语言	105
6.1 例子	105
6.2 过程驱动与事件驱动	109
6.3 PowerScript language 的基本知识	111
6.3.1 基本语法单位	112
6.3.2 PowerScript 语句	113
6.3.3 对控件和对象的引用	115
6.3.4 函数	116
6.4 嵌入 SQL 语句	116
6.4.1 SQL 语言简介	117
6.4.2 SQL 数据定义语言	119
6.4.3 SQL 数据操纵语言	128
6.4.4 库函数	129
6.5 编写脚本	130
6.5.1 打开脚本画板	132
6.5.2 脚本画板的编辑环境	132
6.5.3 编辑脚本会遇到以下问题	132
第七章 加入数据窗口	134
7.1 加入第一个数据窗口	135
7.1.1 在窗口里放入数据窗口控件	135
7.1.2 加入数据窗口对象	137
7.1.3 加入响应事件的脚本	138
7.1.4 给 SQLCA 赋值	141
7.1.5 运行应用程序	142
7.2 加入第一个数据窗口对象	143
7.2.1 指定数据源和风格	143
7.2.2 选择表和表列	146
7.2.3 定义参数	147
7.2.4 定义 WHILE 子句	147
7.2.5 预览数据窗口对象	149

7.2.6 保存数据窗口对象	150
7.2.7 修改编辑风格	150
7.2.8 在窗口中加入数据窗口控件和数据窗口对象	153
7.3 为第一个窗口加入脚本	155
7.4 运行程序	156
第八章 加入多行编辑控件	158
8.1 加入静态文字控件	158
8.2 加入多行编辑控件	159
8.3 加入多行编辑的脚本	161
8.4 运行程序	163
第九章 增加查阅及维护功能	165
9.1 加入查阅按钮	165
9.1.1 加入按钮	165
9.1.2 为“首项”按钮加入脚本	168
9.1.3 为“前翻”按钮加入脚本	168
9.1.4 为“后翻”按钮加入脚本	168
9.1.5 为“末项”按钮加入脚本	168
9.1.6 运行程序	169
9.2 增加数据库维护窗口	170
9.2.1 画一个新的窗口	170
9.2.2 定义窗口风格	173
9.3 创建及修饰新的数据窗口	175
9.3.1 创建新的数据窗口	175
9.3.2 预览数据窗口对象	179
9.3.3 保存数据窗口对象	180
9.3.4 修饰数据窗口对象	180
9.4 在维护窗口中加入数据窗口	183
9.4.1 在窗口里放入数据窗口控件	183
9.4.2 加入数据窗口对象	184
9.5 为维护数据库窗口加入脚本	186
9.6 为维护数据库窗口加入维护按钮	192
9.6.1 加入图片按钮	192
9.6.2 为“插入”按钮加入脚本	198
9.6.3 为“删除”按钮加入脚本	199
9.6.4 为“更新”按钮加入脚本	199
9.6.5 为“退出”按钮加入脚本	200
9.6.6 运行程序	200

第十章 加入菜单	203
10.1 打开菜单画板	203
10.2 加入菜单项	205
10.3 给菜单加入脚本	211
10.4 加入第二个菜单	213
10.5 把菜单加入到窗口	216
10.6 运行程序	219
第十一章 创建独立于 PowerBuilder 环境的应用程序	222
11.1 创建 EXE 文件	222
11.1.1 动态连接库	224
11.2 执行 EXE 文件	225
第十二章 PowerBuilder 函数快速参考	230
附录 程序源码	351

序章 编写本书的目的

搞软件的人好辛苦!知识老化速度太快,需要学习的新东西层出不穷。就拿软件工作者人人都会一点的编程语言来讲,谁能数清楚究竟有多少种呢?

数清楚不容易,要想掌握当然更难。但是,针对不同的问题,我们不得不努力钻研永远学不完的编程语言。昨天,我们谈论 C++、VB,今天我们研究 PowerBuilder,明天可能是 Java。显然,按部就班,依照传统学习语言的方法,已经不能跟上语言更新的节奏了。

出版科技类图书异常困难的今天,计算机类书籍却是例外。大大小小的出版社争先恐后出版各种与计算机有关的书籍,页数动辄数百页的各类大全占了其中很大部分。

我也读过很多大而全的书籍,特别是当软件的大趋势由 DOS 转向 Windows 时,面对书店里蜂拥而至的各种由美国人编写有关 Windows 编程的(××Bible)翻译而来的(××大全),由于崇洋媚外心理作祟,采用“通吃”的手法,不考虑经济承受能力而买了不少。

结果花了两个多星期时间,消灭了一本 500 多页的《 Borland C++ 编写 Windows 程序大全》,看得头晕脑胀,回头一总结——竟然没搞懂 Windows 编程为何物!这决不是夸张。

愤然弃 Borland C++ 而学 Visual Basic,又是一本 500 多页的大全《 Visual Basic 编程指南》,又花了两个多星期时间,又弄得头晕脑胀,又回头一总结——还是一塌糊涂!

有高手指点:这是因为这两种语言太难,现在都流行学 PowerBuilder,人称 VB 杀手,杀手, killer,懂么?喜新厌旧的品性又一次暴露出来,毫不犹豫地扔掉 VB,转而投向 PowerBuilder 的怀抱,又是一本令人心惊胆颤的大全。

这时候,有人邀请我参加一个任务很紧的网络数据库系统的开发工作,开发软件正是 PowerBuilder。显然,通过读大全来学会使用 PowerBuilder 时间已经不够了,幸好我发现了一本极好的教材——PowerBuilder 的用户手册《 Getting Start》(开始),全书引导初学者一步一步编写第一个 PowerBuilder 程序,按照书上的说法,学会编写第一个程序只需要按部就班执行如书所示 4 个小时的操作而已。事实的确如此,我很快了解到 PowerBuilder 编程的大概方法,并且利用学到的知识,愉快地与人合作完成了网络数据库系统的开发。应该承认,在开发过程中,《 Getting Start》传授的知识是不够的,我仍然不断查阅大全。但此时,这本大全已经不那么面目可憎,因为我对 PowerBuilder 已经有了大概了解,能够知

道自己不明白的地方应该查阅大全的哪一部分。

阿兰·拉肯提出了著名的“80/20”定律：如果所有的工作项目都是根据价值的大小来排列，那么，80%的价值来自只占20%的项目，其余20%的价值来自80%的项目。一本大全几乎囊括了PowerBuilder所有的内容，一本《Getting Start》只提供了其中的20%。但是，对于我来说，这20%的内容可以解决编程中80%的问题。

好了，我们现在问：究竟应该用100%的时间去学习100%的内容，还是只用20%的时间学习其中80%的时间会用到的内容？

作为编程的基础课程《C语言》，我们学得肯定够细够微，回想一下，有几次编程你用到了C语言的指针呢？而在学习指针花费的时间几乎是其他内容的总和。也许你要说了，不用指针是因为水平不够，那我要告诉你，作为下一代语言的代表Java，几乎等同没有指针的C++语言，哎，学习指针的努力似乎白费了。

尽管PowerBuilder做得如此之好，但是我从不认为PowerBuilder所有的内容对我们都有用。我们有必要对所有的问题进行研究吗？

在学习语言方面我们从来就是追根究底，循序渐进，所以你也许很难接受如下观点：在变化太快的软件界，使用20%的时间学习80%常用的东西是最优的学习方法。

如果你接受这个观点，谢谢，并且恭喜：你接受了一种只需耗费很少时间(例如3天)，学会实用的PowerBuilder编程的方法，就算你从未接触过Windows编程。

短时间(3天)学会PowerBuilder编程，在许多人看来都不太可能。不管你同样心存疑虑还是坚信不疑，请务必按书的要求，具体上机操作。否则，几天下来，你很有可能破口大骂，因为你一无所获。

0.1 本书内容简介

作为一名初学者，在后面的内容中你很快会发现对自己最有用的东西——完全照着章节的安排，也就是依葫芦画瓢，你的第一个Power Builder实用程序很快开发成功。如果你还有些编程基础，比如曾经编制过超过二十行的C语言小程序，学完本书后，下次看见类似“招聘软件工程师，要求熟练使用Power Builder，月薪三千”的广告，你完全可以大大方方走到招聘处，说：“嗨，算我一个！”(当然，你完全有可能像我一样，并不是第一次就会成功)。

我们以一个有关民航航班的程序为例，一步一步了解PowerBuilder编程的基本方法，也是PowerBuilder编程的主要内容。程序类似于一本航班时刻表。它可以录入、查询、删除、更新民航航班时刻信息。程序主要由以下几个窗口组成。

程序主窗口，如图 0.1.1。

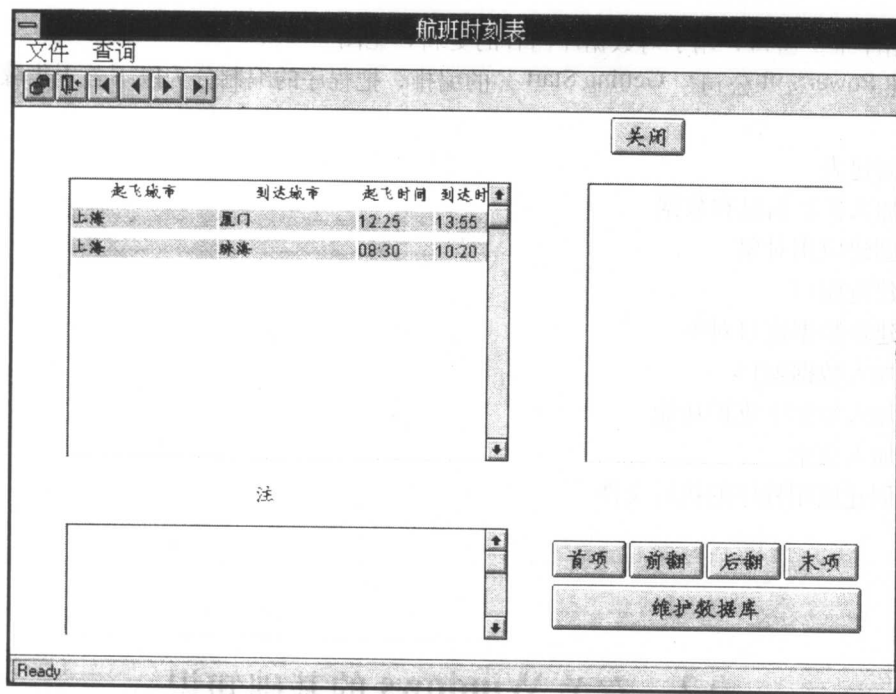


图 0.1.1

窗口最上面的菜单，包含了主窗口所有的功能选项。例如：录入、查询、删除、前翻、后翻，等等。

菜单栏下面是带有 MicroHelp 的小按钮，它们与 PowerBuilder 工具栏的图标一致，也是一些菜单选项的快捷按钮。



图 0.1.2.

窗口中的 DataWindow(数据窗口)是本窗口的焦点,所有的航班时刻信息都可以从中查找到

数据库维护窗口,用于对数据库内容的更新,见图 0.1.2。

参照 PowerSoft 公司《 Getting Start 》的编排,把程序的编制分为以下若干步骤:

1. 创建表
2. 加入扩展信息和数据
3. 创建应用对象
4. 建造窗口
5. 建造数据窗口对象
6. 加入数据窗口
7. 加入数据库维护功能
8. 加入菜单
9. 创建应用程序的执行文件

0.2 有关 Windows 的基础知识

假如你对 Windows 知之甚少,以下一些有关 Windows 的基础内容也许对你有所帮助。

0.2.1 窗口的构成

窗口是构成 Windows 程序的基本元素。按照 Microsoft 的说法,一个窗口包括以下全部或部分成分:

窗口边界	工作区	控制菜单框	控制菜单	菜单条
水平滚动条	垂直滚动条	最大框	最小框	标题栏

如图 0.2.1

1. 窗口边界

窗口边界包围着一个窗口的外沿,它包括三个基本内容: 1.窗口的四角,可以用来同时改变窗口水平和垂直方向的大小; 2.垂直边,可以改变窗口水平方向的大小; 3.水平边,可以改变窗口垂直方向的大小。

定位鼠标于窗口的边界,注意鼠标形状的变化。当鼠标形状变为双箭头时,按鼠标左键不放,并拖动鼠标,窗口形状发生变化。

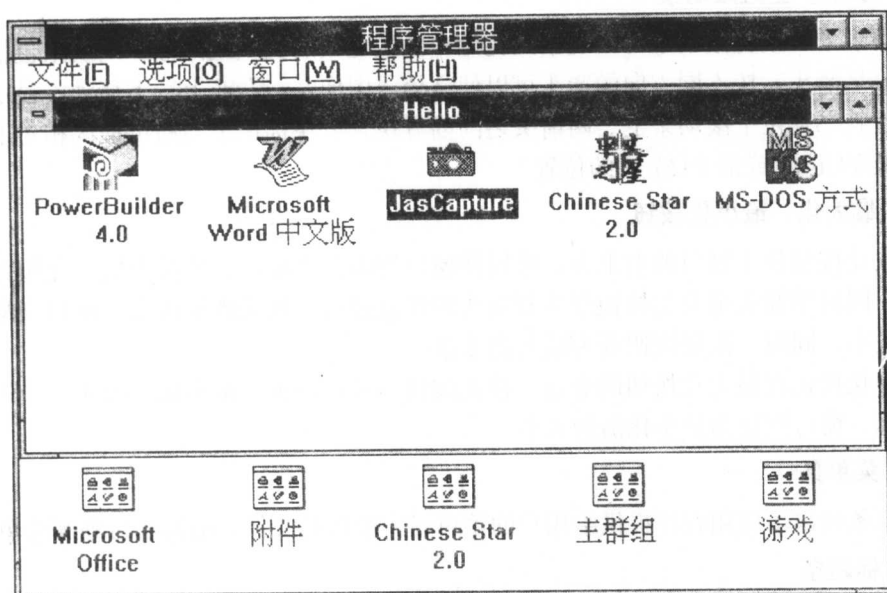


图 0.2.1

2. 工作区

工作区又叫用户区，指窗口内，未被菜单条、滚动条、边界或其他成分所占据的部分。

3. 控制菜单框、控制菜单

控制菜单框位于窗口的左上角，通过控制菜单框，可以打开或关闭控制菜单，如图 0.2.2。

控制菜单一般包括还原、移动、大小、最大化等选项。如果选项为灰白色，表示该选项当前不可执行。控制菜单提供了对窗口的基本操作。

还原(R)	CTRL+F5
移动(M)	CTRL+F7
大小(S)	CTRL+F8
最小化(N)	
最大化(X)	CTRL+F10
关闭(C)	CTRL+W
下一窗口(O)	CTRL+F6

图 0.2.2

4. 水平、垂直滚动条

移动水平、垂直滚动条可以观察到比用户区域更大的文档或图像。滚动条包括两个元素，其一是箭头，按不同方向的箭头可以使文档或图像左、右或上、下移动，显示被遮挡的部分，其二是位于滚动条上、两箭头之间的方块，方块标识了当前可视区相对于文件或图像的最左(最右)或最上(最下)的位置。

5. 最大化、最小化按钮

最大化按钮位于窗口的右上方，可以使窗口充满整个屏幕。当按下最大化按钮，窗口最大化，同时单箭头最大化按钮变为双箭头的恢复按钮；按动恢复按钮，窗口恢复为最大化前的大小，同时，恢复按钮变为最大化按钮。

最小化按钮在最大化按钮的旁边。按此按钮，可以使窗口最小化，缩为一个图标。双击该图标，窗口恢复为最小化前的大小。

6. 菜单条

菜单条列出了应用程序提供给用户的菜单。一般包括文件、编辑和帮助等菜单。

7. 标题条

标题条列出程序的名字。标题条通过颜色或亮度区分窗口是当前窗口，或者不是。

0.2.2 在 Windows 中启动应用程序

在 MS-DOS 提示符下键入 win，按回车键。一会儿，Windows 徽标出现，紧接着，Windows 的某个外壳程序(shell)——一般是 Program Manager(程序管理器)出现，此时，Windows 被打开了，如图 0.2.3。我们在编写本书的示例程序时，使用的 Windows 是中文 Windows 3.2。所以，本书插图均取自中文 Windows 3.2，与西文 Windows 3.1 略有不同。

外壳程序是 Windows 的主控程序。Windows 用它来启动、中止 Windows 的运行；由它启动各种应用程序完成所需的工作。Windows 有不同风格的外壳程序，适应不同操作者的工作习惯。

Program Manager 是一种图形化的 Windows 外壳，它使用图标(icon)表示应用程序。在代表程序的图标上连续按两次，可以启动一个应用程序。

太多的图标分散了使用者的注意力，Program Manager 提供了“程序组”(program group)管理图标。一个程序组包括了相关的各种图标，见图 0.2.4，标题条为“主群组”的就是一个程序组，其中的各个图标代表不同的应用程序。

程序管理器可以有多个程序组，一个程序组可以包含多个程序。程序管理器同样使用图标代表程序组。与代表程序的图标不同，双击程序组图标仅仅打开了程序组窗口，不会运行任何应用程序。

在 Program Manager 里启动一个程序的过程如下：

1. 打开包含有需要运行的程序图标的程序组；
2. 双击代表程序的图标。

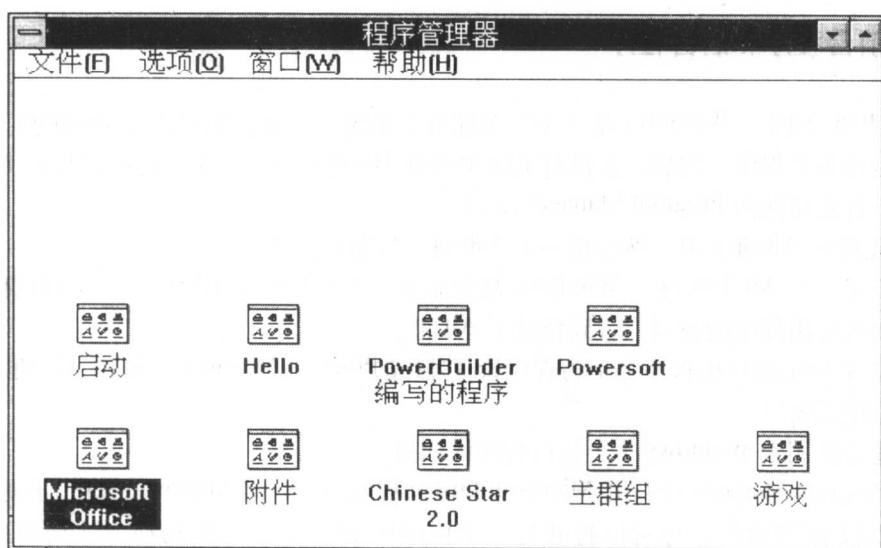


图 0.2.3

比如，我们启动 Power Builder：

1. 双击打开 PowerSoft 程序组；
2. 双击 Power Builder 图标，程序启动。

比较在 DOS 命令行输入程序名启动程序的方式，是不是感觉太轻松愉快了？

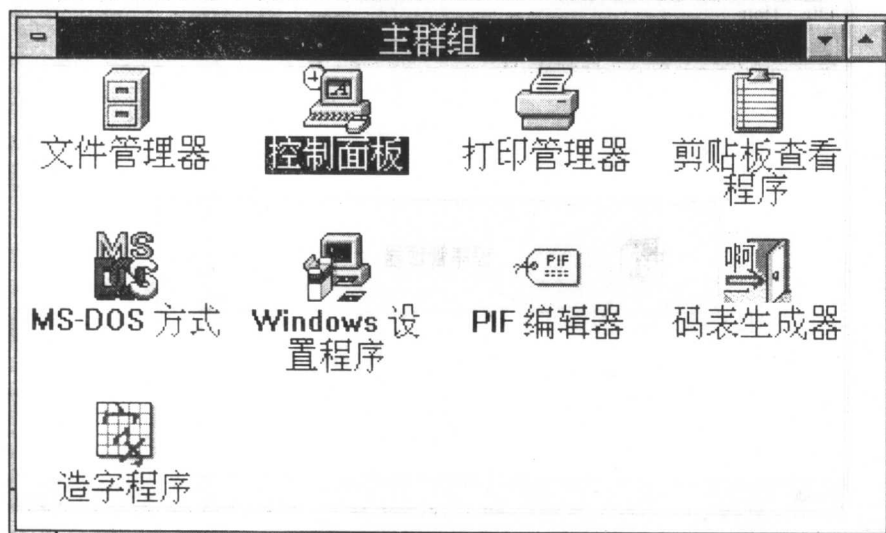


图 0.2.4

0.2.3 前台程序和后台程序

与 DOS 不同，Windows 是一个“多任务”系统，因此我们可以在 Program Manager 中同时运行多个程序。例如，在没有关闭 Power Builder 的情况下，还可以打开 Microsoft Word。首先切换回 Program Manager 窗口：

1. 先按住 Alt 键不放，然后按一下 Tab 键，屏幕如图 0.2.5。

每按钮一次 Alt-Tab 键，Windows 都会显示一个不同的应用程序的名字和图标，但不会切换到该应用程序的窗口，直到释放了 Alt 键。

2. 如果显示的应用程序名字为程序管理器(或 Program Manager)，释放 Alt 键，屏幕变为程序管理器窗口。

上述方法用于 Windows 多个运行程序的切换。

在 Program Manager 中打开 Microsoft Office 程序组，启动 Microsoft Word 程序(如果你的机器上没有 Word，你可以打开另一个程序)。使用上面讲的切换方法切换回 Power Builder，然后再次切换回 Word。怎么样，两个程序都没有停止运行？

只要内存允许，Windows 可以同时运行数个应用程序。但每时每刻只能有一个前台程序，其他的都是后台程序。前台程序的窗口是活动的，它一般占有最多的 CPU 时间。我们通过 Alt-Tab 键，可以随时把前台程序切换为后台程序，同时把一个后台程序切换为前台程序。一般情况下，Windows 的打印程序都在后台运行，这样，在漫长打印的同时，我们还可以进行其他的工作。

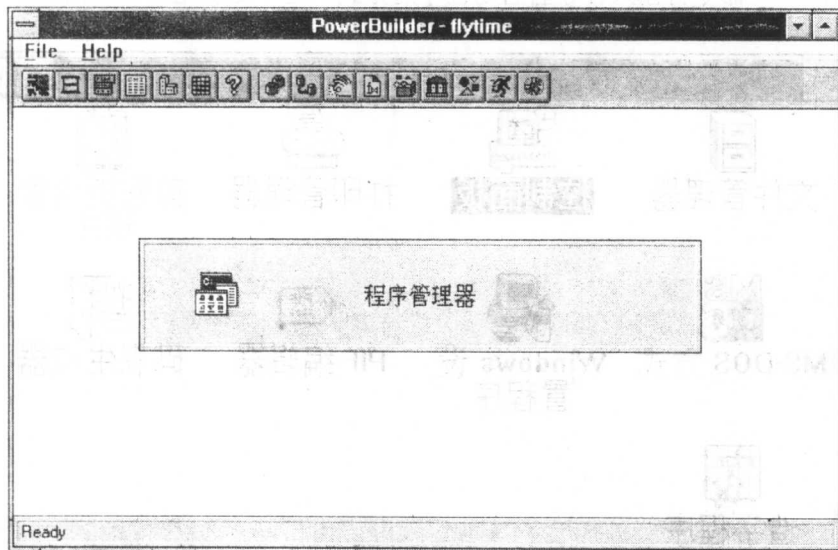


图 0.2.5