

PowerBuilder 10

企业应用程序开发系列丛书



余金山 冯星红 李 肖 编著

PowerBuilder 10

数据库开发高级实例



科学出版社

www.sciencep.com

PowerBuilder 10 企业应用程序开发系列丛书

PowerBuilder 10 数据库 开发高级实例

余金山 冯星红 李 肖 编著

科 学 出 版 社

北 京

内 容 简 介

本书是“PowerBuilder 10 企业应用程序开发系列丛书”的工程实例篇。

本书以企业固定资产管理系统为背景,结合软件工程的思想,从需求分析、系统设计到编码实现,完整地介绍了如何使用 PowerBuilder 开发企业应用系统。

本书总结了作者多年从事工程应用开发的经验,对从事数据库开发的人员具有较高的参考价值,是一本不可多得的参考读物。本书也可作为大专院校计算机专业以及相关培训机构的教材或参考书。本书源代码请读者登录<http://www.sciencep.com>查询。

图书在版编目(CIP)数据

PowerBulider 10 数据库开发高级实例/余金山等编著. —北京:科学出版社, 2005

(PowerBulider 10 企业应用程序开发系列丛书)

ISBN 7-03-015004-X

I. P… II. 余… III. 数据库系统-软件工具,PowerBuilder 10
IV.TP311.56

中国版本图书馆 CIP 数据核字(2005)第 010851 号

责任编辑:李娜 马琳/责任校对:都岚

责任印制:吕春珉/封面设计:耕者设计室

科学出版社 出版

北京东黄城根北街16号

邮政编码:100717

<http://www.sciencep.com>

双音印刷厂 印刷

科学出版社发行 各地新华书店经销

*

2005 年 2 月第 一 版 开本:787×1092 1/16

2005 年 2 月第一次印刷 印张:25 3/4

印数:1—3 000 字数:597 000

定价:39.00 元

(如有印装质量问题,我社负责调换<路通>)

销售部电话:010-62136131 编辑部电话:010-62138978-8004

前 言

PowerBuilder 10 是 Sybase 公司推出的新一代可视化应用开发工具,支持面向对象的程序开发,对于开发基于 Windows、分布式计算和 Web 应用系统均有出色的表现。PowerBuilder 开发工具的一个特点是易于学习,能够轻松上手,为新的企业应用开发需求提供支撑。

本书是“PowerBuilder 10 企业应用程序开发系列丛书”的工程实例篇。

本书以企业固定资产管理系统为背景,结合软件工程的思想,从需求分析、系统设计到编码实现,完整地介绍了如何使用 PowerBuilder 开发企业应用系统。书中浓缩了作者多年从事工程应用开发的经验和教训,相信对读者具有很大的参考价值。

在本书学习的过程中,需要注意以下几个方面。

- 本书介绍的实例以 PowerBuilder 10 作为数据库前端开发工具,Oracle 作为后台数据库。因此,读者模拟实例操作前需要先安装 PowerBuilder 和 Oracle 数据库,同时配置好与数据库的连接。

- 因为篇幅的限制,某些模块的实现以源代码的形式提供,请读者登录 <http://www.sciencep.com> 查询源代码并自行配置,体会系统的实现。

- 本书用实例体现了多种报表的设计,这对于需要使用 PowerBuilder 设计高级报表的开发人员具有很大的帮助。

- 读者在学习创建系统的时候,需要注意系统的整体框架结构以及各个模块的具体实现,从而加深对整个系统设计的理解。

参与本书编写的人员有李自强、李枫、唐家峰、朱名选、方程、刘小山、程工、吴权海、刘比新、朱海涛、方新成、吴凯、唐务明、付松新、杨峰、孟海明、赵林尚、李科、成自权、周海兴等。

由于编者水平有限,书中错漏、不妥之处在所难免,恳请广大读者批评指正。

编 者

2004 年 6 月

目 录

第 1 章 PowerBuilder 10 概述	1
1.1 使用 PowerBuilder 创建应用程序概述	1
1.2 使用 PowerBuilder 10 开发第一个应用程序	2
1.3 定制 PowerBuilder 10	5
1.3.1 设置启动选项	5
1.3.2 定制系统树	6
1.3.3 定制工具栏	8
1.3.4 定制键盘快捷键	9
1.3.5 定制颜色	10
第 2 章 应用系统需求分析与建模基础	12
2.1 软件工程基本思想	12
2.2 需求分析	13
2.3 UML 建模基础	14
2.3.1 用例与用例图	15
2.3.2 静态图	15
2.3.3 行为图	15
2.3.4 交互图	16
2.3.5 实现图	16
2.3.6 配置图	16
2.4 数据建模	16
2.4.1 概念数据库建模	17
2.4.2 实体—关系建模	18
2.4.3 面向对象的建模	18
2.5 设计数据库	18
2.5.1 创建数据库计划	19
2.5.2 联机事务处理与决策支持	20
2.5.3 数据完整性	22
2.5.4 数据安全性	23
2.5.5 性能	23
2.5.6 维护	24
2.6 使用 PowerBuilder 开发应用	24
2.6.1 需求分析	24
2.6.2 系统设计	28
2.6.3 界面设计	31

2.6.4	菜单设计	31
2.6.5	报表设计	31
2.6.6	应用程序编译、测试与发布	32
第 3 章	系统需求分析与设计	33
3.1	需求分析	33
3.2	系统设计	35
3.3	数据字典	38
第 4 章	企业固定资产管理系统	47
4.1	系统概述	47
4.2	创建应用	47
4.3	系统准备	48
4.3.1	建立同数据库的连接	49
4.3.2	创建库	50
4.3.3	创建祖先窗口	51
4.3.4	创建复用的用户对象 u_datawindow	65
4.3.5	创建父用户对象 u_dw	86
4.4	创建系统登录模块	91
4.5	创建主窗口	103
4.6	系统主菜单设计	104
4.7	运行系统	111
第 5 章	权限管理子系统设计	112
5.1	概述	112
5.1.1	设计目标	112
5.1.2	系统组成	112
5.1.3	授权模式	112
5.1.4	权限说明	113
5.1.5	数据量控制	113
5.2	用户组管理	113
5.2.1	创建新增用户组窗口	113
5.2.2	用户组查看窗口	117
5.2.3	创建分配用户组用户窗口	124
5.3	用户管理	129
5.3.1	创建新增用户窗口	129
5.3.2	用户信息查看窗口	129
5.3.3	用户授权管理	130
5.3.4	口令管理	131
5.3.5	用户管理主窗口	132
第 6 章	基本档案管理子系统	153
6.1	资产存档模块设计	153

6.1.1	资产存档主窗口设计	155
6.1.2	附属设备录入窗口设计	193
6.2	资产编辑模块设计	197
6.3	分类代码管理模块设计	216
6.4	代码查询模块设计	228
第 7 章	固定资产信息查询子系统	229
7.1	系统概述	229
7.2	固定资产查询	229
7.3	报废资产查询	243
7.4	闲置资产查询	246
第 8 章	固定资产动态管理	249
8.1	资产调拨	249
8.1.1	产品部内部调拨	249
8.1.2	公司内部调拨	269
8.1.3	公司对外调拨	270
8.1.4	固定资产调拨审批	272
8.2	报废管理	273
8.2.1	资产报废申请	273
8.2.2	资产报废审批	290
8.2.3	报废资产保管	291
8.3	更新计划	291
8.3.1	更新计划编辑	292
8.3.2	更新计划汇总	294
8.3.3	更新计划审批	295
8.4	闲置管理	296
8.4.1	资产闲置申请	296
8.4.2	资产闲置审批	297
8.4.3	闲置资产保管	297
8.5	复活管理	298
8.5.1	资产复活申请	298
8.5.2	资产复活审批	300
第 9 章	固定资产统计分析子系统设计	301
9.1	系统概述	301
9.2	变动记录	302
9.3	资产清查	310
9.3.1	清查表管理	310
9.3.2	清查表汇总	323
9.4	资产核算	336
9.5	资产统计	351

第 10 章 创建打印子系统	352
10.1 打印子系统概述	352
10.2 创建模块所需要的结构和用户对象	353
10.3 创建打印和打印预览比例设定窗口	379
10.4 创建排序窗口	380
10.5 创建等待窗口	388
10.6 创建录入报表新标题窗口	389
10.7 创建打印设置窗口	390
10.8 菜单设计	395
10.9 创建打印子系统主窗口	397
参考文献	404

第 1 章 PowerBuilder 10 概述

PowerBuilder 是由多个功能模块构成的可视化集成开发环境，是面向对象的开发工具，可以用于创建传统的 C/S 应用和基于 Windows 的分布式数据库应用。PowerBuilder 为开发人员提供了强大的集成开发特性，各个功能模块协同工作，完成对应用的创建、编码、测试、发布和部署等工作。PowerBuilder 还提供了丰富的面板和开发向导，可视化应用程序开发。本章主要介绍 PowerBuilder 的一些基础知识，为开发本书实例做些准备。

本章主要内容：

- 使用 PowerBuilder 创建应用程序概述。
- 使用 PowerBuilder 10 开发第一个应用程序。
- 定制 PowerBuilder 10。

1.1 使用 PowerBuilder 创建应用程序概述

当开发一个典型的客户机/服务器应用程序的时候，需要执行一些基本的步骤，完成第一步以后，就可以按照任何顺序执行开发任务。也就是说，开发人员可以按照应用需求定义使用的对象，完成对系统的开发任务。创建一个典型的客户机/服务器应用的基本步骤如下。

1. 创建应用（使用 New 向导），定义应用的库列表

当使用 Start 向导的时候，开发人员创建 Application 对象，它是应用的入口点，应用对象包括应用名和应用层的脚本。

2. 创建窗口

将控件放在窗口中，定义在事件触发的时候，如何操作的脚本语句。

3. 创建菜单

在窗口中的菜单包括菜单栏、下拉菜单、层叠菜单和浮动菜单，开发人员可以确定菜单项，编写菜单项的脚本。

4. 创建用户对象

如果希望能够重用在窗口中的构件，可以将它们定义为用户对象，然后保存到库中。当创建一个窗口的时候，只需要将用户对象放在窗口中即可，不需要重新定义它们。

5. 创建函数、结构和事件

为了支持自己定义的脚本，开发人员可以定义函数、执行应用的处理、定义结构、保存相关数据。同时，开发人员还可以定义自己的用户事件。

6. 创建数据窗口对象

使用这些数据窗口对象从数据库中提取数据、格式化和验证数据、通过图表分析数据、更新数据库。

7. 测试和调试应用

开发人员可以在任何时候执行应用程序，如果发现问题，需要通过设置断点、单步执行，查看执行过程中的变量值来调试应用。用户在运行应用和使用概要工具分析一个应用的性能和逻辑流的时候，可以创建一个跟踪文件。

8. 准备可执行文件

当完成了应用编码并测试通过以后，可以创建应用程序的一个可执行版本，分发给用户。

1.2 使用 PowerBuilder 10 开发第一个应用程序

在 PowerBuilder 中执行任何开发任务之前，都必须创建或者打开一个工作区。创建一个新的工作区的步骤如下。

1) 单击 PowerBar 中的 New 图标 ，或者选择 File | New 菜单选项，也可以在系统树中的工作区标签中，右击工作区名，从浮动菜单中选择 New 菜单选项，打开 New 对话框，如图 1.1 所示。

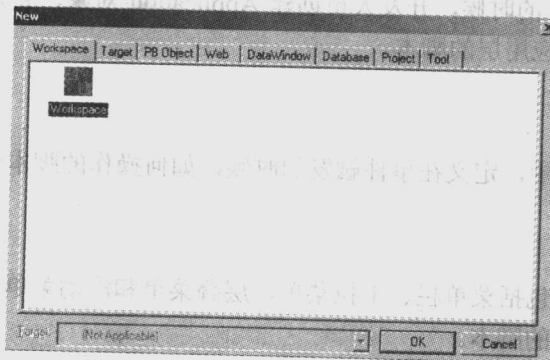


图 1.1 新建工作区对话框

2) 单击 OK 按钮，定义工作区的名称和保存目录位置，输入文件名为 gz，单击“保存”按钮即可，如图 1.2 所示。

这样，就创建了一个新的工作区，在 PowerBuilder 标题栏中显示出来。如图 1.3 所示。



图 1.2 定义工作区的名称和存储文件夹

当下一次启动 PowerBuilder 的时候，上次打开的工作区将自动打开，可以在 System Options 对话框中的 Workspace 标签页中或者在 Welcome to PowerBuilder 屏幕中修改 Reopen workspace on startup 选项，取消自动打开工作区的行为，如图 1.4 和图 1.5 所示。



图 1.3 新建的工作区

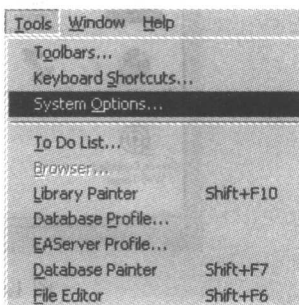


图 1.4 选择 Tools | System Options 菜单

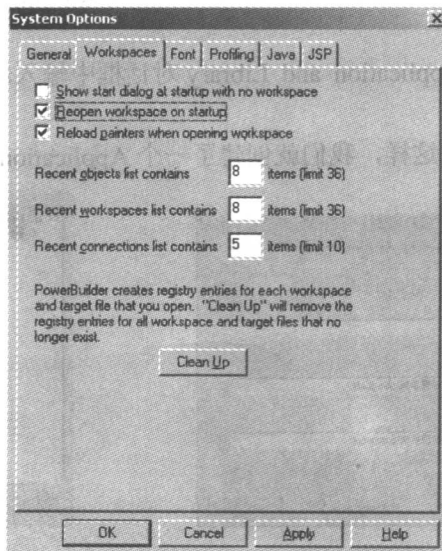


图 1.5 Workspaces 标签页

如果需要修改到最近使用的工作区,可以单击 **File | Recent Workspace** 中的某个工作区。创建了工作区之后,需要创建一个目标, **PowerBuilder** 提供了多种应用目标。

① **PowerScript** 应用目标: 应用目标向导、模板应用目标向导和已有目标向导。

② **PowerScript** 构件目标: **EAServer** 构件目标目标向导、**COM/MTS/COM+** 构件目标向导和自动服务器目标向导。

③ **Web** 目标: **JSP** 目标向导、**Dynamo web** 站点目标向导、**Web** 站点目标向导、移植 **PowerSite 7.1** 项目向导和源代码控制 **Web** 目标向导。

当创建一个目标时, **PowerBuilder** 将提示开发人员输入目标名和位置。目标文件可以是文本文件, 包含目标的信息。为了创建一个新的目标, 执行下面的步骤。

1) 单击 **PowerBar** 中的 **New** 图标, 或者选择 **File | New** 菜单选项, 也可以在系统树中的工作区标签中, 右击工作区名, 从浮动菜单中选择 **New** 菜单选项, 打开 **New** 对话框, 如图 1.6 所示。

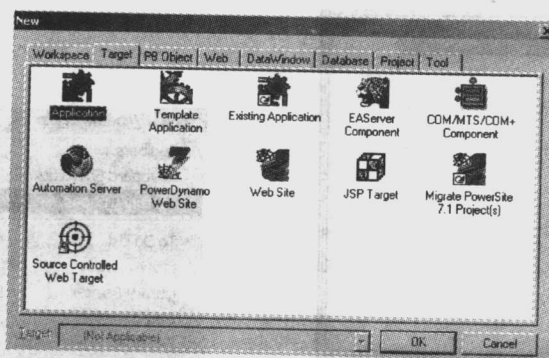


图 1.6 新建目标对话框

2) 在 **Target** 标签页中, 选择一个 **Target** 向导。在此, 我们选择 **Application** 向导, 单击 **OK** 按钮, 打开向导。

3) 在 **Specify New Application and Library** 对话框中输入应用名 **gdzcg1**, 单击 **Next** 按钮后, 如图 1.7 所示。

4) 单击 **Finish** 按钮。这样, 我们就创建了一个 **Application** 的目标, 如图 1.8 所示。

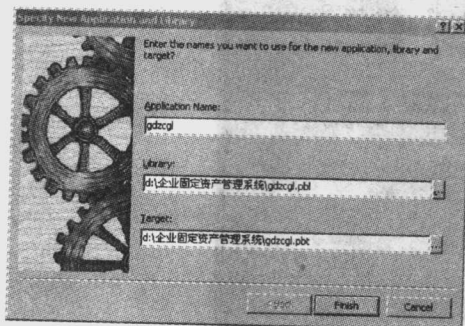


图 1.7 定义应用目标名

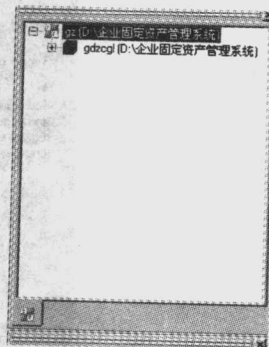


图 1.8 创建完成后在系统树中的应用程序目标

1.3 定制 PowerBuilder 10

在开发应用程序的时候，可能需要自定义 PowerBuilder 集成开发环境。下面我们介绍如何自定义 PowerBuilder 集成开发环境，满足应用开发的需要，并且能够获得 PowerBuilder 的大多数特性，主要内容如下。

- ① 设置启动选项。
- ② 定制系统树。
- ③ 定制工具栏。
- ④ 定制键盘快捷键。
- ⑤ 定制颜色。

1.3.1 设置启动选项

当启动 PowerBuilder 的时候，可能需要在已有的项目上继续工作，为此，可以让 PowerBuilder 在启动的时候自动打开上次使用的工作区和面板。

在系统选项对话框中的 Workspaces 标签页包含三个选项，可以用来确定在启动 PowerBuilder 的时候显示的内容。如果需要打开系统选项对话框，可以单击 Tools | System Options 选项，如图 1.5 所示。

这个对话框中包含三个选项。

① Show start dialog at startup with no workspace: 在启动的时候显示开始对话框，但是不打开工作区。

② Reopen workspace on startup: 在启动的时候打开工作区。

③ Reload painters when opening workspace: 当打开工作区的时候打开面板。

开发人员可以从命令行启动 PowerBuilder，打开一个工作区、目标和面板。能够打开的面板如下。

- ① Application painter: 应用面板。
- ② Database painter: 数据库面板。
- ③ Data Pipeline painter: 数据管道面板。
- ④ DataWindow painter: 数据窗口面板。
- ⑤ Debugger: 调试器。
- ⑥ File Editor: 文件编辑器。
- ⑦ Function painter: 函数面板。
- ⑧ Library painter: 库面板。
- ⑨ Menu painter: 菜单面板。
- ⑩ Query painter: 查询面板。
- ⑪ Structure painter: 结构面板。
- ⑫ User Object painter: 用户对象面板。
- ⑬ Window painter: 窗口面板。

语法如下：

```
directory\pb10.exe (/workspace workspacepath) (/target targetpath)
(/painter paintername) (/output outputpath)
```

其中 *directory* 包含 PowerBuilder 的全路径名。

同时，可以在 */painter paintername* 添加选项到命令行，打开特定的对象，或者创建新的对象，语法如下：

```
{/library libraryname} (/object objectname) (/inherit objectname)
{/new} (/run) (/runonly) (/argument arguments)
```

这个语法语句显示了选项名的表单，参数说明如表 1.1 所示。

表 1.1 选项含义

选 项	说 明
<i>/W workspacepath</i>	打开工作区 <i>workspacepath</i> ，如果已经在系统选项对话框中选择了重新打开工作区，默认打开最近使用的对话框
<i>/T targetpath</i>	打开目标 <i>targetpath</i>
<i>/P paintername</i>	打开面板 <i>paintername</i> 。当开始一个新的 PowerBuilder 会话时，默认显示的是窗口。面板名必须唯一标识面板。不需要输入整个名称，例如，可以使用 <i>q</i> 打开查询面板， <i>datab</i> 打开数据库面板。如果输入全面，在名称中删除任何空格。面板名不区分大小写。为了打开文件编辑器，可以设置 <i>paintername</i> 为 <i>F1</i> 或者 <i>fileeditor</i>
<i>/OU outputpath</i>	记录输出窗口的内容到 <i>outputpath</i>
<i>/L libraryname</i>	表示包含希望打开的对象库。默认为 PowerBuilder 初始化文件中的 <i>PB</i> 项中的 <i>Deflib</i> 变量中定义的库
<i>/O objectname</i>	表示希望打开的对象，如数据窗口对象或者窗口
<i>/I objectname</i>	表示希望继承的对象
<i>/N</i>	创建新的数据窗口对象
<i>/R</i>	运行在 <i>/O</i> 中定义的数据窗口对象，允许设计
<i>/RO</i>	运行在 <i>/O</i> 中定义的数据窗口对象，但是不允许设计
<i>/A arguments</i>	为特定数据窗口对象提供参数

下面的示例假定 Powerbuilder 可执行文件的位置位于系统路径中。它在 Client PBL 中打开窗口面板。输出的会话发送到名为 *math.log* 的文件中。工作区文件、PBL 和日志文件都在当前目录中。

```
Pb10 /W Math.pbw /L Client.pbl /P window /oumath.log
```

输入命令启动 PowerBuilder，打开数据窗口对象，称为 *d_emp_report*。

```
Pb10 /W D:\pbws\Emp.pbw /P dataw /O d_emp_report
```

1.3.2 定制系统树

可以使用几种方式修改 PowerBuilder 的主窗口的框架。

- ① 显示和隐藏系统树、输出窗口、剪贴窗口、修改它们的位置。
- ② 显示和隐藏面板中的视图，修改它们的位置。

同时，我们还可以显示或者隐藏工具栏、改变它们的位置、添加自定义按钮。

1. 安排系统树、输出窗口和剪贴窗口

系统树、输出窗口、剪贴窗口在任何时候都可以隐藏，只需要单击 PowerBar 上的按钮。例如，隐藏系统树窗口之前的界面如图 1.9 所示。当单击系统树按钮 System Tree 隐藏系统树窗口的界面如图 1.10 所示。

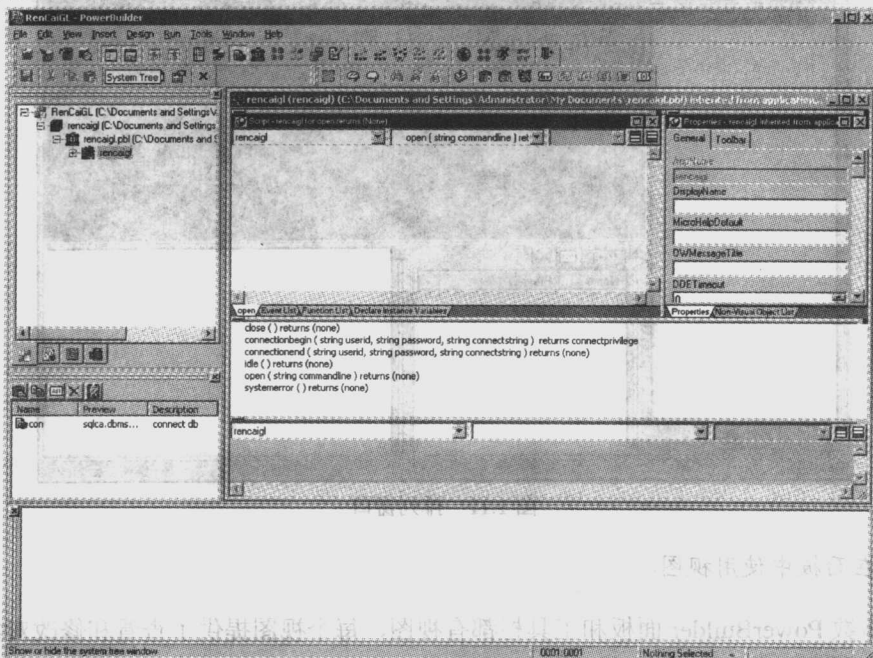


图 1.9 隐藏系统树窗口之前

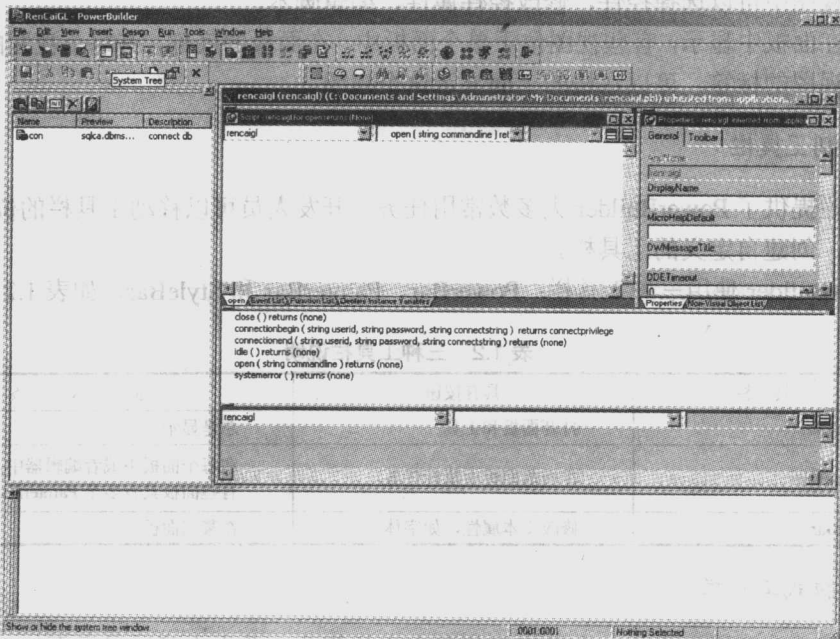


图 1.10 隐藏系统树窗口

我们可以将主窗口打开为全宽度和高度的状态，在窗口的下面显示了剪贴窗口和输出窗口，如图 1.11 所示。

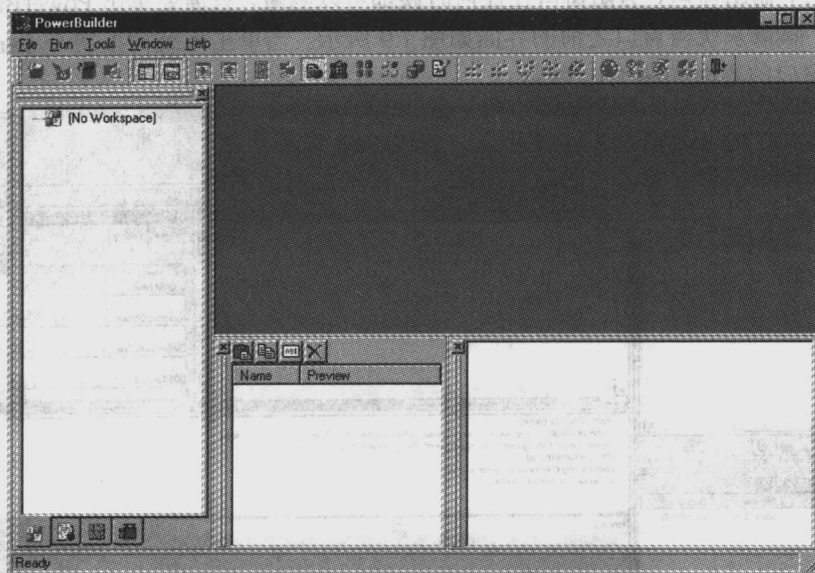


图 1.11 排列窗口

2. 在面板中使用视图

大多数 PowerBuilder 面板和工具栏都有视图，每个视图提供了查看和修改对象的方式。在面板窗口中，让所有的视图可用意味着一次可以执行多个任务。例如，在面板窗口的框架视图中可以选择控件，修改控件属性，编辑脚本。

视图在面板中显示，有些视图位于单个面板中。在面板的底部，有每个视图的标签，单击这个视图的标签，可以在栈的上面显示这个视图。

1.3.3 定制工具栏

工具栏提供了 PowerBuilder 大多数常用任务，开发人员可以移动工具栏的位置，自定义工具，创建自定义的工具栏。

PowerBuilder 使用三种工具栏：PowerBar、PainterBar 和 StyleBar，如表 1.2 所示。

表 1.2 三种工具栏说明

工 具 栏	具有按钮	显 示
PowerBar	打开面板和工具	总是显示
PainterBar	在当前面板中执行任务	在每个面板中或者编辑器中，有些面板具有多个 PainterBar
StyleBar	修改文本属性，如字体	在某些面板中

1. 下拉式工具栏

为了减少工具栏的大小，有些工具栏具有下拉箭头，单击它可以显示下拉工具栏，包含相关按钮。例如，在数据窗口面板中单击“文本”按钮的下拉箭头，显示控件工具

栏，它包含了能够在数据窗口对象中使用的每个控件对象，如图 1.12 所示。

2. 控制工具栏的显示

开发人员可以控制以，以下操作。

- ① 是否显示单个工具栏和显示的位置。
- ② 是否在按钮上显示文本。
- ③ 是否显示 PowerTips。

为了使用弹出式菜单控制工具栏，执行下面的操作。

- ① 指针放在工具栏上面，显示弹出式菜单。
- ② 单击按钮。

为了使用工具栏对话框控制工具栏，执行下面的操作。

- ① 从菜单栏选择 **Tools | ToolBar**，显示工具栏对话框。
- ② 单击希望启用的工具栏和选项。

PowerBuilder 为开发人员提供了自定义工具栏的特性，开发人员可以在任何工具栏中添加、移动和删除按钮，使用按钮激活其他应用，如时钟和文件处理器。添加一个按钮到工具栏的步骤如下。

- ① 在工具栏中定位指针，显示浮动菜单。
- ② 选择自定义，显示自定义对话框，如图 1.13 所示。
- ③ 在 **Select palette** 组对话框中，单击按钮的调色板。
- ④ 从 **Selected palette** 框架中选择一个按钮，将它拖到当前工具栏中即可。

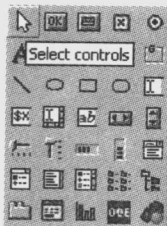


图 1.12 控件工具栏

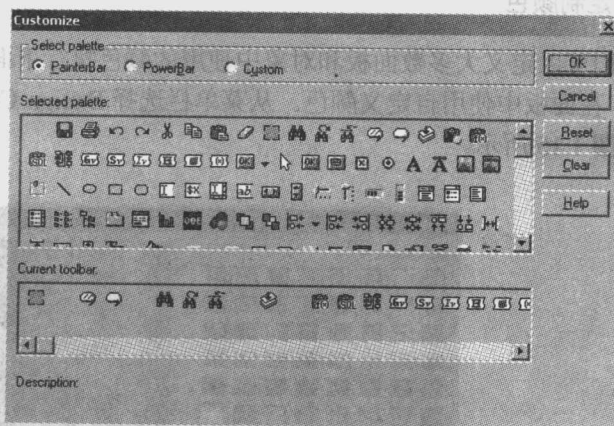


图 1.13 自定义工具栏

1.3.4 定制键盘快捷键

用户可以为 PowerBuilder 菜单项目定义自己的键盘快捷键。例如，如果希望使用另一个调试器，可以自定义特定的函数件或者键的组合调试函数。开发人员可以修改默认的快捷键，将 PowerBuilder 的调试器同某个动作关联起来。

为了将键盘快捷键同菜单项目关联起来，执行下面的操作。