

PowerBuilder 应用程序的项目开发

师海峰 杨家军 刘丽霞 (哈尔滨工业大学 150001)

PowerBuilder 5.0 作为客户/服务器模式中客户端的开发工具得到了开发人员的普遍好评,该开发工具不仅功能强大,而且在方便软件人员的开发工作和工程管理方面做出了杰出的贡献。

开发人员作出选用 PowerBuilder 5.0 作为其前端开发工具的决策前,首先需要了解它的特点以便于充分利用它所提供的功能,本文从开发前、编码阶段和后期打包三个阶段阐述了 PowerBuilder 5.0 的基本特点。

一、开发前的准备工作

1. 定义标准和规范

在一个软件的开发工作开始时定义一系列的标准及规范将非常有助于整个系统的维护或升级。通常来说,标准及规范包括界面风格、命名规则、编程风格和文档规范的定义。PowerBuilder 5.0 在这几方面都建议软件开发人员尽量遵循着一定的标准。

在命名规则方面,如果能保证各元素的名称直观将大大提高程序的可维护性和容易阅读。PowerBuilder 5.0 中每个元素的名称能达到 40 个字符长,同时 PowerBuilder 5.0 为其每个元素的命名提供一个缺省的前缀,通过前缀能够清楚地辨别出该元素是对象、控件或者变量以及是什么对象,是什么控件,如果是变量,能够表明出它的范围和数据类型;

在编程风格方面,PowerBuilder 5.0 在编程人员编写代码时自动检测当前语句是否为循环语句、分枝语句,为此提供了自动缩进;PowerBuilder 5.0 为变量的定义提供了全局(Global)、实例(Instance)、共享(Share)和本地(Local)等几种类型,能使变量在最小范围内使用,同时变量可在代码头一起定义,也可放在代码中最靠近第一次使用的地方定义。

在界面风格方面,开发者可采用多文档界面,也可采用单文档界面或者它们的结合,可采用菜单工具条型,也可采用命令按钮型。PowerBuilder 5.0 提供了各种控件、多种颜色和多种字体来支持用户的选择,窗口的

类型也有主窗口(main)、响应窗口(response)、弹出窗口(popup)和子窗口(child)四种供选择。

在文档规范方面,PowerBuilder 5.0 为开发者所创建的各种对象,所建立的表和字段,所编写的代码提供注释的机会,同时能够使软件开发人员很方便地为最终用户提供联机帮助。

2. 仓库

在利用一个软件时,首先看一看该软件有哪些附带的元件是相当有益的,PowerBuilder 5.0 提供了三个仓库,即应用程序仓库、艺术仓库和对象仓库。应用程序仓库中包括了许多在 PowerBuilder 5.0 中设计的例程。另外,PowerBuilder 5.0 提供的 The Code examples 和 The Anchor Bay Nut Company(ABNC) Order Entry Application 两个例子覆盖了极大范围的编程技术;艺术仓库则包括了一些位图和像标,可充分应用于界面中表达思想;对象仓库中包括了许多 OCX 控件。这三个仓库都可在安装时装入。

3. 设计、定义数据库

我们可以利用服务器端的数据库管理系统所带的工具进行数据库的设计和创建。PowerBuilder 5.0 也提供了对后端服务器中数据库创建的功能齐全的工具,它们是:Database Painter,负责创建和管理表和视图;Table Painter 定义或修改表的细节及字段、主键索引等普通特性和扩展特性,扩展特性包括字段的格式、有效性规则、初始值、显示标题、对准方式、长宽等;Data Manipulation Painter 实现表中数据的增删改查功能;Database Administration Painter 定义表的安全性;Data Piping Painter 实现把一个表的数据拷到另一个表中的功能。

二、开发阶段

1. 库的管理

PowerBuilder 5.0 库文件(.PBL)存储着开发人员开发的所有对象,为了系统的性能或使用方便,一个库文件最好小于 800K,内放的对象应小于 50 或 60 个,同时

应按照各个库文件被访问的频率为顺序安排它们的访问路径,由于搜索路径长将影响效率,因此库文件的数量不宜过多。

库文件中的对象有两种存放方式:按照对象的类型或按照功能模块划分。PowerBuilder 5.0 为其库文件的管理提供了极其方便的工具,在其 Library 工具中可以拷贝移植和删除各对象,可以访问或升级以前的版本,也可以当某一对象的祖先发生改变时重新编译该对象(PowerBuilder 5.0 库文件中存放着每一对象的两种格式:一种为字符描述,另一种为解释性的编译代码—PCODE型);另外,如果你经常对某一库文件的对象做改动,其内部存放空间变得零散,将影响运行效率,此时你可以运用 Library 工具中的优化功能。

值得称道的是 PowerBuilder 5.0 为多个开发者参与的项目管理提供了一种模式。该模式对库文件的管理分为三层—产品层、质量保证层和开发层。产品层为最终的版本,开发者不对产品层中的库文件进行修改和增加;如果需要则将其拷到质量保证层中。每一个开发者对其工作的库文件在本地机建立一个开发层,开发层中的对象都从建在服务器上的质量保证层中“抽出”,该“抽出”的对象将被打上标记,其他开发者只可读不能写,当完成任务后,可将该对象放回质量保证层中。质量保证层中的各对象都从服务器上的产品层中拷入和拷出,PowerBuilder 5.0 提供了抽出和放回的功能,同时还提供了版本控制来跟踪和管理某一对象的变更情况。

2. 编码工作

在编码中编码人员经常需要调用某些对象的属性、事件,调用函数过程等。在 PowerBuilder 5.0 的 PowerScript 工具中编码,一方面利用各种菜单功能可以迅速地查看各种对象所具有哪些属性以及各个变量结构等,另一方面此工具还提供了各种粘贴功能,对各种条件转移语句、循环语句和多种功能的内嵌 SQL 语句、各种对象的属性和事件、全局变量和参数都能顺手拈来,使程序员免受查找和输入之苦,大大提高了编程速度。

PowerBuilder 5.0 还为新建的应用程序提供了一个应用框架,开发者可以在这个框架的基础上进行开发。

三、后期工作

1. 编译和测试

PowerBuilder 5.0 提供了两种编译代码的类型:即

Pcode 型(Pseudocode 的缩写)和机器代码型(适用于 32 位)。Pcode 型实际上是一种伪编译,它是一种解释性语言。两者相比较,Pcode 型生成代码快,但运行速度慢,同时生成代码尺寸小,兼容性强。

在一个 PowerBuilder 5.0 应用程序中通常包括着一个可执行文件、多个动态库文件和资源文件,你可以不要动态库文件,即把所有对象均放入一个可执行文件中,但当可执行文件的大小超过 1.5 兆时,最好将它分为多个动态库。因为只有需要在需要时才调入内存,利用动态库可以提高程序运行的效率,另外还可以提高程序的重用性和灵活性。

PowerBuilder 5.0 在创建可执行文件时并不自动的把各种资源(比如位图和像标等)装入可执行文件。要想达到这一目的只需利用一个文本编辑器创建一个包含着(一行一个)所有资源名的资源文件(PBR 文件)放入工程文件即可。如果不想装入,你可以在安装软件时拷贝到应用程序所在目录即可,但这样做将降低程序的运行效率。

在 PowerBuilder 5.0 开发环境中测试软件,首先可以预览你所创建的各种界面,也可以利用 Run Window 功能单独运行某一特定的窗口,还可以利用 PowerBuilder 5.0 提供的数据库管理功能完成与数据库相关的测试。

在可执行文件运行时,可以打开 PowerBuilder 5.0 的跟踪功能,它将对象的创建和释放,事件和函数的执行记录在一个文本文件中供查看。

2. 制作安装盘

为了让应用程序独立于开发环境运行,需要在用户的机器上安装 PowerBuilder 5.0 所有的 .DLL 文件、开发者的可执行文件、动态库文件和资源文件,有时还有 .INI 文件和 .HLP 文件。

利用 PowerBuilder 5.0 提供的 Install Builder 可以制作优美的安装程序。它具有创建目录、拷贝文件、修改配置(Autoexec. bat, config. sys 和 .ini 文件)、(按磁盘大小)分块压缩解压和创建程序组等功能,它能够根据用户需要修改安装界面。它提供常见的三种安装模式,可以将整个软件分块选择安装。

由上可见,作为 PowerBuilder 5.0 集成开发环境的一个重要组成部分,PowerBuilder 5.0 所提供的各种功能保证了开发质量,减少了开发者的额外负担,适应了开发者的需要,必会大大加快开发速度。

(来稿时间:1997年4月)