

<<PowerBuilder 7.0应用技>>

图书基本信息

书名：<<PowerBuilder 7.0应用技术>>

13位ISBN编号：9787302008163

10位ISBN编号：7302008167

出版时间：2000-1

出版时间：清华大学出版社

作者：崔巍

页数：320

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<PowerBuilder 7.0应用技>>

内容概要

PowerBuilder作为一种软件工具由于其强大的开发功能及具备垮平台的应用系统开发能力，越来越受到数据库开发人员的青睐。

本书从多个角度介绍了PowerBuilder 7.0的应用技术和高级使用方法。

绪论和第1章是序篇，简单介绍面向对象的程序设计方法和快速学习PowerBuilder 7.0的捷径；第2~12章则分别介绍一些PowerBuilder 7.0的高级用法，包括：MDI应用、多窗口实例、数据管道、统计图、RichText、OLE和DDE等程序设计技术，及一些高级控件的使用和方法；第13~21章则重点介绍了PowerBuilder 7.0的分布式开发方法，主要包括Jaguar组件和客户端的开发、COM/MTS组件和客户端的开发方法，以及分布式PowerBuilder服务器和客户端的开发等。

本书适合于具有一定PowerBuilder 7.0(Power Builder)的使用和开发基础、而又想进一步提高的读者或开发人员阅读。

<<PowerBuilder 7.0应用技>>

书籍目录

绪论 面向对象程序设计的实现	0.1 面向对象术语	0.2 面向对象技术在PowerBuilder中的实现
0.2.1 继承	0.2.2 封装	0.2.3 多态性
PowerScript语言使用的几个专题	0.3.1 常量说明	0.3.2 控制对实例变量的存取
0.3.3 解决命名冲突	0.3.4 祖先对象程序的返回值	0.3.5 函数和事件
的参数类型	0.3.6 祖先和子孙变量	第1章 快速捷径
1.2 运行样本程序	1.2.1 浏览和运行样本程序	1.1 样本程序
1.2.2 查找自己想要的內容	2.1 什么是MDI	2.1.1 MDI窗口的构成
第2章 MDI应用程序	2.2 建立MDI窗口	2.3 MDI窗口的MicroHelp
1.2 标准MDI窗口和定制MDI窗口	2.3.1 菜单项的MicroHelp	2.4 MDI窗口的工具栏
2.3.1 菜单项的MicroHelp	2.3.2 控件的MicroHelp	2.4.1 MDI窗口工具栏的一些概念
2.4.1 MDI窗口工具栏的一些概念	2.4.2 在Menu画板中设置工具栏的属性	2.4.3 建立工具栏中的下拉工具栏
2.4.3 建立工具栏中的下拉工具栏	2.4.4 在Window画板中设置工具栏的属性	2.4.5 在Application画板中设置工具栏的属性
2.4.5 在Application画板中设置工具栏的属性	2.4.6 在MDI应用程序中灵活使用工具栏	2.5 调整客户区域的尺寸
2.5 调整客户区域的尺寸	第3章 多窗口实例	3.1 概述
多窗口实例的使用	3.2 窗口数组	3.2.1 窗口数组的说明和使用
3.2 窗口数组	3.3.1 窗口数组的说明和使用	3.3.2 混合窗口数组
3.3.1 窗口数组的说明和使用	3.3.3 混合窗口数组	3.4 实例窗口小控件的引用
3.3.3 混合窗口数组	4.1 事务对象的一些概念	4.2 事务对象的使用
事务对象与数据库操作	4.2.1 事务的基本概念	4.2.2 事务对象的赋值
4.2.1 事务的基本概念	4.2.3 同时操作多个数据库	4.3 使用事务对象调用存储过程
4.2.2 事务对象的赋值	4.2.3 同时操作多个数据库	5.1 概述
4.2.3 同时操作多个数据库	4.3 使用事务对象调用存储过程	5.2 准备工作
4.3 使用事务对象调用存储过程	5.1 概述	5.2.1 建立Data Pipeline对象
5.1 概述	5.2 准备工作	5.2.2 建立Window对象
5.2 准备工作	5.2.1 建立Data Pipeline对象	5.2.3 建立Window对象
5.2.1 建立Data Pipeline对象	5.2.2 建立Window对象	5.3 数据管道操作及处理程序
5.2.2 建立Window对象	5.2.3 建立Window对象	5.3.1 准备工作
5.2.3 建立Window对象	5.3 数据管道操作及处理程序	5.3.2 数据管道操作
5.3 数据管道操作及处理程序	5.3.1 准备工作	5.3.3 出错处理
5.3.1 准备工作	5.3.2 数据管道操作	5.3.3 出错处理
5.3.2 数据管道操作	5.3.3 出错处理	第6章 灵活运用图形表现数据
5.3.3 出错处理	第6章 灵活运用图形表现数据	6.1 Graph控件简介
第6章 灵活运用图形表现数据	6.1 Graph控件简介	6.2 生成单一产品的销售走势图
6.1 Graph控件简介	6.2 生成单一产品的销售走势图	6.2.1 界面设计
6.2 生成单一产品的销售走势图	6.2.1 界面设计	6.2.2 建立DataWindow对象
6.2.1 界面设计	6.2.2 建立DataWindow对象	6.2.3 编写程序代码
6.2.2 建立DataWindow对象	6.2.3 编写程序代码	6.3 生成多个产品销售走势对比图
6.2.3 编写程序代码	6.3 生成多个产品销售走势对比图	6.4 动态改变图形的类型
6.3 生成多个产品销售走势对比图	6.4 动态改变图形的类型	第7章 窗口中几个控件的使用
6.4 动态改变图形的类型	第7章 窗口中几个控件的使用	7.1 Tab标签控件
第7章 窗口中几个控件的使用	7.1 Tab标签控件	7.1.1 术语及实例说明
7.1 Tab标签控件	7.1.1 术语及实例说明	7.1.2 准备工作
7.1.1 术语及实例说明	7.1.2 准备工作	7.1.3 建立Tab控件和选项卡的程序
7.1.2 准备工作	7.1.3 建立Tab控件和选项卡的程序	7.1.4 管理Tab控件及其选项卡
7.1.3 建立Tab控件和选项卡的程序	7.1.4 管理Tab控件及其选项卡	7.1.5 写Tab控件的程序
7.1.4 管理Tab控件及其选项卡	7.1.5 写Tab控件的程序	7.2 ListView控件
7.1.5 写Tab控件的程序	7.2 ListView控件	7.2.1 利用Listview控件进行查询
7.2 ListView控件	7.2.1 利用Listview控件进行查询	7.2.2 ListView控件的详细资料方式
7.2.1 利用Listview控件进行查询	7.2.2 ListView控件的详细资料方式	7.3 TreeView控件
7.2.2 ListView控件的详细资料方式	7.3 TreeView控件	7.3.1 用TreeView控件实现带层次的查询
7.3 TreeView控件	7.3.1 用TreeView控件实现带层次的查询	7.3.2 TreeView控件应用技术
7.3.1 用TreeView控件实现带层次的查询	7.3.2 TreeView控件应用技术	7.4 轨迹条控件
7.3.2 TreeView控件应用技术	7.4 轨迹条控件	7.5 进度条控件
7.4 轨迹条控件	7.5 进度条控件	第8章 Rich TeXt应用
7.5 进度条控件	第8章 Rich TeXt应用	8.1 RichTextEdit控件的使用
第8章 Rich TeXt应用	8.1 RichTextEdit控件的使用	8.1.1 RichTextEdit控件的属性设置
8.1 RichTextEdit控件的使用	8.1.1 RichTextEdit控件的属性设置	8.1.2 RichTextEdit控件的文本
8.1.1 RichTextEdit控件的属性设置	8.1.2 RichTextEdit控件的文本	8.1.3 打开和保存文件的实例
8.1.2 RichTextEdit控件的文本	8.1.3 打开和保存文件的实例	8.2 RichTextEdit控件中的输入域
8.1.3 打开和保存文件的实例	8.2 RichTextEdit控件中的输入域	8.2.1 在文档中插入域
8.2 RichTextEdit控件中的输入域	8.2.1 在文档中插入域	8.2.2 为输入域指定值
8.2.1 在文档中插入域	8.2.2 为输入域指定值	8.2.3 为日期和页码指定输入域
8.2.2 为输入域指定值	8.2.3 为日期和页码指定输入域	8.3 预览和打印
8.2.3 为日期和页码指定输入域	8.3 预览和打印	8.3.1 预览
8.3 预览和打印	8.3.1 预览	8.3.2 打印
8.3.1 预览	8.3.2 打印	8.4 使用数据库中的数据
8.3.2 打印	8.4 使用数据库中的数据	8.5 RichText
8.4 使用数据库中的数据	8.5 RichText	数据窗口
8.5 RichText	数据窗口	第9章 OLE技术及应用
数据窗口	第9章 OLE技术及应用	9.1 OLE的基本概念
第9章 OLE技术及应用	9.1 OLE的基本概念	9.2 OLE控件和可插入的对象
9.1 OLE的基本概念	9.2 OLE控件和可插入的对象	9.2.1 建立和设置OLE控件
9.2 OLE控件和可插入的对象	9.2.1 建立和设置OLE控件	9.2.2 链接和嵌入
9.2.1 建立和设置OLE控件	9.2.2 链接和嵌入	9.2.3 Offsite或In-place激活
9.2.2 链接和嵌入	9.2.3 Offsite或In-place激活	9.3 OLE定制控件
9.2.3 Offsite或In-place激活	9.3 OLE定制控件	9.4 可编程的OLE对象
9.3 OLE定制控件	9.4 可编程的OLE对象	9.4.1 OLEObject对象数据类型
9.4 可编程的OLE对象	9.4.1 OLEObject对象数据类型	9.4.2 OLEObject的事件
9.4.1 OLEObject对象数据类型	9.4.2 OLEObject的事件	9.4.3 OLEObject对象之间的赋值
9.4.2 OLEObject的事件	9.4.3 OLEObject对象之间的赋值	9.5 OLE程序设计
9.4.3 OLEObject对象之间的赋值	9.5 OLE程序设计	9.5.1 自动操作接口
9.5 OLE程序设计	9.5.1 自动操作接口	9.6 Browser画板中的OLE信息
9.5.1 自动操作接口	9.6 Browser画板中的OLE信息	9.7 操作OLE对象的高级方法
9.6 Browser画板中的OLE信息	9.7 操作OLE对象的高级方法	9.7.1 OLE存储的结构
9.7 操作OLE对象的高级方法	9.7.1 OLE存储的结构	9.7.2 存储和流的对象类型
9.7.1 OLE存储的结构	9.7.2 存储和流的对象类型	9.7.3 打开和保存存储
9.7.2 存储和流的对象类型	9.7.3 打开和保存存储	9.7.4 建立和使用存储的例子
9.7.3 打开和保存存储	9.7.4 建立和使用存储的例子	9.7.5 OLE流的概念和应用
9.7.4 建立和使用存储的例子	9.7.5 OLE流的概念和应用	9.7.6 使用存储的策略
9.7.5 OLE流的概念和应用	9.7.6 使用存储的策略	第10章 PowerBuilder自动服务器的基本概念
9.7.6 使用存储的策略	第10章 PowerBuilder自动服务器的基本概念	10.1 Automation Server
第10章 PowerBuilder自动服务器的基本概念	10.1 Automation Server	10.2 用户对象作为自动服务器对象
10.1 Automation Server	10.2 用户对象作为自动服务器对象	10.2.1 建立作为服务器的类用户对象
10.2 用户对象作为自动服务器对象	10.2.1 建立作为服务器的类用户对象	10.2.2 建立对象的运行时库
10.2.1 建立作为服务器的类用户对象	10.2.2 建立对象的运行时库	10.2.3 注册用户对象
10.2.2 建立对象的运行时库	10.2.3 注册用户对象	10.2.4 编写访
10.2.3 注册用户对象	10.2.4 编写访	

<<PowerBuilder 7.0应用技>>

问用户对象的客户端程序	10.3 使用PowerBuilder作为自动服务器	10.3.1 建立要访问的用户对象
客户端程序	10.3.2 生成运行时库	10.3.3 编写访问PowerBuilder和用户对象的客户端程序
动态数据交换的概念	10.4 命名服务器的建立和使用	第11章 动态数据交换
11.2.1 DDE客户端程序设计	11.2 动态数据交换的编程	11.1 11.2.2 DDE服务器程序编程
11.2.3 DDE的事件和函数	11.2.3 DDE的事件和函数	11.3 动态数据交换应用实例
11.3.1 DDE服务器程序设计实例	11.3.1 DDE服务器程序设计实例	11.3.2 DDE客户程序设计实例
11.4 使用Excel打印PowerBuilder的报表	第12章 几种常用编程技术	
12.1 使用初始化文件和Windows注册表	12.1.1 使用初始化文件	12.1.2 使用Windows注册表
12.2 处理Blob数据	12.2.1 处理Blob数据的SQL语句	12.2.2 利用Blob数据完成对图片或大文本的处理
12.2.3 数据窗口中的Blob列	12.3 动态链接库与外部函数的调用	12.3.1 说明外部函数
12.3.2 外部函数调用举例	12.4 嵌入邮件应用	12.4.1 与邮件应用相关的技术
12.4.2 邮件会活编程	第13章 分布式应用概述	13.1 为什么要使用分布式应用技术
13.1.1 客户/服务器模式存在的问题	13.1.2 分布式计算机提供的解决方案	13.2 Jaguar CTS
13.2.1 什么是Jaguar CTS	13.2.2 Jaguar服务器结构	13.2.3 Jaguar客户结构
13.3 微软事务服务器(MTS)	13.4 分布式PowerBuilder	
13.4.1 服务器应用程序	13.4.2 客户应用程序	13.4.3 通信驱动程序
第14章 建立Jaguar组件	14.1 建立Jaguar组件方法概述	14.1.1 使用向导建立Jaguar组件
14.1.2 建立Jaguar组件的步骤	14.2 Jaguar管理程序变量空间的方式	14.3 定义组件接口
14.2.1 Jaguar与分布式PowerBuilder的不同之处	14.2.2 Jaguar共享组件	14.3.1 说明接口
14.3.2 接口包含的内容	14.3.3 方法的命名与方法的重载	14.3.4 数据类型
14.3.5 通过引用传送参数	14.3.6 传递只读数据	14.3.7 传递对象
14.3.8 对空值提供支持	14.3.9 Jaguar有效性	14.4 从Jaguar组件访问数据库
14.4.1 概述	14.4.2 使用连接缓存(connection caching)	14.4.3 对事务提供支持
14.4.4 实现检索操作	14.4.5 实现修改	14.4.6 传送结果集
14.5 实例池(Instance Pooling)	14.5.1 为什么使用实例池	14.5.2 在向导的选项中定义实例池
14.5.3 控制放入实例池中的实例状态	14.5.4 组件的生命周期	14.6 使用共享组件
14.6.1 概述	14.6.2 创建共享组件	14.6.3 创建服务组件
14.7 调用其他服务器组件的方法	14.8 厚取组件属性	14.9 将消息放入Jaguar日志
14.10 测试和调试组件	第15章 建立Jaguar客户端应用	15.1 建立Jaguar客户端应用程序
15.2 与Jaguar服务器连接	15.2.1 手工编写代码	15.2.2 使用向导创建Connection对象
15.3 生成Jaguar代理(Proxy)对象	15.4 调用组件方法	15.5 请求从服务器返回信息
15.4.1 调用Jaguar服务器组件的方法	15.4.2 清除代理对象实例	15.6.1 概述
15.6.2 编写Connection对象Error事件的程序	15.6.3 编写SystemError事件的程序	15.7 插接客户应用程序
第16章 建立COM或MTS组件	16.1 建立COM和MTS组件概述	16.1.1 使用向导
16.1.2 开发步骤	16.2 组件对象模型	16.2.1 PowerBuilder的COM服务器
16.2.2 自动服务器和PB COM服务器	16.3 定义组件接口	16.3.1 方法和数据类型
16.3.2 编码时的限制	16.4 从COM和MTS组件访问数据库	16.4.1 传送结果集
16.4.2 从客户端存取了MTS组件的结果集	16.4.3 在PowerBuilder中使用ADO Result Set	16.4.4 从MTS组件中返回结果集
16.5 提供对事务的支持	16.6 为控制对象生存期提供支持	16.7 调用另一个服务器组件的方法
16.7.1 使用OLEObject对象	16.7.2 使用TransactionServer对象	16.8 在日志文件中记录错误
16.9 安全问题	16.9.1 在Project画板或向导中设定权限	16.9.2 标明安全
16.10 在Project画板中建立COM和MTS组件	16.10.1 使用Project画板建立COM服务器	16.10.2 自动注册组件
16.10.3 插接组件到MTS		

<<PowerBuilder 7.0应用技>>

16.10.4 选择定制或双接口	16.11 运行PowerBuilder的COM对象	16.12
插接PB COM服务器	16.12.1 使用带有可用COM应用程序的PB COM服务器	16.
12.2 插接一个PB COM对象到MTS	16.12.3 插接文件到MTS服务器计算机	16.12.
4 导入PB COM对象到MTS	16.12.5 创建客户插接文件	16.12.6 安装客户插接文件
16.13 从客户端访问PB COM服务器	16.13.1 VB作为客户	16.13.2
C++作为客户	16.13.3 使用PB COM服务器和DCOM对象	第17章 建立COM/MTS
客户应用	17.1 建立COM/MTS客户应用程序	17.2 连接到COM服务器
17.3 与COM组件交互	17.3.1 调用组件方法	17.3.2 传送结果集
3.3 处理执行错误	17.4 从客户端控制事务	第18章 建立分布式PowerBuilder
18.1 建立用户界面	18.2 连接到服务器	18.2.1 定义连接变量
2 建立连接	18.2.3 WinSock命名问题	18.2.4 修改主机和服务器文件
18.2.5 建立多重连接	18.3 调用远程对象函数	13.3.1 远程对象的使用
18.3.2 进行异步函数调用	18.3.3 从服务器要求信息返回	18.4 产生代理对象
18.4.1 代理对象	18.4.2 使用代理对象的优点	18.4.3 生成代理对象
18.5 处理通信错误	18.5.1 处理通信错误的概念	18.5.2 编写Connection对
象的Error事件程序	18.5.3 编写SystemError事件的程序	18.6 管理程序变量空间
18.6.1 范围规则	18.6.2 共享对象	18.6.3 共享数据的其他方法
18.7 使用ConectionInfo对象	18.7.1 检索连接信息	18.7.2 对服务器进行管理
活动	18.7.3 断开其他客户的连接	18.8 插接客户应用程序
应用程序打包	18.8.2 远程对象类定义	18.8.1 将
19.1 创建用户界面	19.2 管理客户连接	第19章 建立分布式PowerBuilder服务器
2.2 处理客户请求	19.3 建立远程对象	19.2.1 监听客户连接
2 建立远程对象	19.4 一个完整的分布式PowerBuilder应用	19.3.1 编写用户对象方法
19.4.2 客户端应用	19.4.3 运行	19.4.1 服务器应用
概念	19.5 访问数据库	19.5.1 基本
入事务池中	19.5.2 实现检索操作	19.5.3 实现修改
作	19.6 使用共享对象	19.5.4 将数据库事务放
19.7.1 同步处理和异步处理的不同	19.6.1 概述	19.6.2 共享对象如何工
先权	19.6.3 执行期间的情况	19.7 处理异步请求
19.8 传送消息到客户端	19.6.4 举例	19.7.1 同步请求采取优
19.8.3 客户端传送对象引用	19.7.2 请求进队列	19.7.2 请求进队列
19.9 防止死锁	19.8.1 服务器发送	19.8.2 请求进队列
19.10 插接服务器应用程序	19.8.4 服务器使用远程引用	19.8.5 举例
生成器	19.9.1 引起死锁的原因	19.9.2 如何防止死锁
20.2 Java数据类型	第20章 使用JavaBeans代理生成器	20.1 JavaBeans代理生
20.4.1 使用Java类的步骤	20.1.1 开发过程	20.1.2 生成Java类
20.4.4 在Constructor事件中设置连接属性	20.1.2 生成Java类	20.1.3 生成其他的类
在连接方法中设置连接属性	20.3 大Project画板中生成Java类	20.4 使用生成的Java类
在JavaBeans代理对象中调用方法	20.4.1 使用Java类的步骤	20.4.2 建立Java工程
引用卡集成Java类	20.4.2 建立Java工程	20.4.3 建立连接
21.1.1 准备调试组件	20.4.4 在Constructor事件中设置连接属性	20.4.5 单独设置连接属性
21.1.4 状态	20.4.5 单独设置连接属性	20.4.6
21.2.1 概述	20.4.7 利用JavaBeans的持续性设置连接属性	20.4.8
21.2.2 用Local驱动程序测试	20.4.9 使用Date和DateTime类	20.5 使用PowerJ
21.2.4 使用跟踪程序	第21章 多层应用程序排错	21.1 排除Jaguau组件的错误
	21.1.1 准备调试组件	21.1.2 开始调试
	21.1.2 开始调试	21.1.3 与本地调试的区别
	21.1.4 状态	21.1.5 多个实例
	21.2 排除分布式PowerBuilder应用程序中的错误	
	21.2.1 概述	21.2.2 用Local驱动程序测试
	21.2.2 用Local驱动程序测试	21.2.3 用通信驱劝程序在本
	21.2.3 用通信驱劝程序在本	地测试

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>