

481

Borland/Inprise核心技术丛书

最新JBuilder开发 人员指南

田勇 孙新 等编著

北京宝兰英博思公司推荐用书

本书附盘可从本馆主页 <http://lib.szu.edu.cn/>
上由“馆藏检索”该书详细信息后下载，
也可到视听部复制



A0988300



机械工业出版社
China Machine Press

JBuilder是美国Borland公司最新推出的可视化Java开发工具，是目前计算机开发者首选的Java开发环境。JBuilder 4支持最新的J2EE标准，它的可视化工具和开发向导使Java应用开发变得非常方便快捷。本书结合具体的开发实例详细讲解了JBuilder 4的开发过程。内容包括：Java语言的相关知识，数据库应用系统的开发过程，JBuilder 4的DataExpress体系结构，以及如何用JBuilder 4开发Web应用、Java RMI应用、CORBA应用和EJB应用。相信本书无论对JBuilder的初学者，还是Java语言的技术专家都会有一定的帮助。

版权所有，侵权必究。

图书在版编目（CIP）数据

最新JBuilder开发人员指南 /田勇，孙新等编著. -北京：机械工业出版社，2001.5
（Borland/Inprise核心技术丛书）

ISBN 7-111-09041-1

I. J… II. ①田…②孙… III. JAVA语言-程序设计 IV. TP312

中国版本图书馆CIP数据核字（2001）第039508号

机械工业出版社（北京市西城区百万庄大街22号 邮政编码 100037）

责任编辑：谢君英 张鸿斌

北京忠信诚胶印厂印刷·新华书店北京发行所发行

2001年7月第1版第1次印刷

787mm × 1092mm 1/16 · 24.25印张

印数：0 001-5 000册

定价：68.00元（附光盘）

凡购本书，如有倒页、脱页、缺页，由本社发行部调换

前 言

Java语言是Sun公司推出的计算机应用程序开发语言，由于具有与平台无关的特点及完备的面向对象特性，因此越来越受到开发人员的喜爱和认可。美国Borland公司是世界著名的计算机软件厂商，其推出的集成开发工具享誉全球。JBuilder 4是Borland公司最近推出的遵循Sun公司J2EE标准的集成开发工具，是目前使用最广泛的Java开发工具之一。

JBuilder 4提供了可视化的集成开发环境，支持开发人员构建标准的Java应用系统。其开放的环境、基于组件的开发方式以及丰富的Java开发向导极大地简化了开发人员的工作。无论是构建标准的Java应用程序、Java Applet、Java Bean、数据库应用，还是JSP、Servlet等Web应用，或者RMI、CORBA、EJB等分布式应用系统，JBuilder 4都提供了大量的组件、开发向导以及应用模板，使开发更为快捷和简便。

在本书中，我们将结合具体的开发实例详细讲解使用JBuilder 4进行应用开发的过程。内容共包括三个部分：

在第一部分“基础篇”中，我们介绍了Java语言的一些相关知识、JBuilder 4的集成开发环境、以及使用JBuilder 4开发Java应用的过程。

在第二部分“数据库应用篇”中，我们详细介绍了JBuilder 4的DataExpress体系结构，以及如何使用JBuilder 4的数据库组件和可视化组件开发强大的数据库应用系统。

在第三部分“分布式应用篇”中，我们着重介绍如何开发Java Web应用、Java RMI应用、CORBA应用以及Enterprise JavaBean (EJB) 应用，还介绍了如何进行分布式调试以及团队开发的一些内容。

本书对Java开发语言以及JBuilder 4开发工具都有详细的阐述，非常适用于计算机领域的开发人员及高等院校相关专业的师生阅读。相信无论对JBuilder的初学者，还是Java的专家都会有一定的帮助。

本书由北京宝兰英博思信息技术有限公司提供素材，主要由田勇、孙新编著完成，杨少雄、陈秀娟、李萍等人也参与了其中部分工作。书中如有错误与疏漏之处，请读者指正。

田 勇

2001年5月于北京

Borland
ENTERPRISE

第一部分 基础篇

第1章 JBuilder 4概述

JBuilder 4是一个基于Java编程语言的快速可视化开发工具。可以用它开发各种Java应用程序和系统,包括Java Applet、数据库应用程序和分布式多层结构的应用。JBuilder 4提供了百分之百的纯Java开发环境。所有用Java语言编写的程序都能在JBuilder 4中运行和调试。与其他Java开发环境相比,JBuilder更好地发挥了Java语言的编程能力。开发Java应用需要一个高效率的工具简化编程工作。JBuilder 4为开发应用程序提供了必要的工具以及语言支持。JBuilder 4支持的Java技术包括:JavaBeans、Java 2、JDK1.x()、JFC/Swing、Servlet、RMI(Remote Method Invocation)\JDBC(Java Database Connectivity)、ODBC(Open Database Connectivity)、SQL(Structured Query Language)、EJB(Enterprise JavaBeans)、XML(Extensible Markup Language)、JSP(JavaServer Pages)、CORBA(Common Object Request Broker Architecture)。在后面的章节中,我们将分别介绍相关的内容。

JBuilder 4提供了一个开放的体系结构,开发人员可以方便地在JBuilder 4 IDE中加入新版本的JDK、第三方工具、插件以及JavaBean组件。

本章将简要介绍JBuilder 4的集成开发环境,帮助读者快速掌握其使用方法。

1.1 JBuilder 4集成开发环境

1.1.1 主界面介绍

JBuilder 4集成开发环境(IDE)提供了一个单一窗口界面,我们称之为应用浏览器(AppBrowser)。其中预设了大量的开发功能,我们可以在此窗口中创建、编辑、管理文件与工程,并进行可视化设计、编译、调试与运行应用程序。JBuilder 4集成开发环境如图1-1所示。

JBuilder 4集成开发环境中各个部分的名称及其功能如下:

- 主菜单栏(Main menu): 提供菜单访问方式,如文件菜单、编辑菜单、查找菜单、运行菜单与向导菜单。
- 主工具栏(Main toolbar): 按功能划分成若干小工具栏,上面的按钮提供了对菜单命令的快捷访问。
- 工程工具栏(Project bar): 包含了一个当前所打开工程的列表及一些按钮,这些按钮用来增加、删除文件,关闭及更新工程面板中的工程文件。
- 工程窗口(Project pane): 显示所选工程的内容。用于浏览与操作工程树。
- 结构窗口(Structure pane): 结构窗口显示内容窗口中当前文件的结构。对一个Java文件而

言,该窗口以树的形式显示了所有方法、属性与事件。结构窗口具有向下展开的功能,双击一个类或者界面就可以查看它的祖先类。

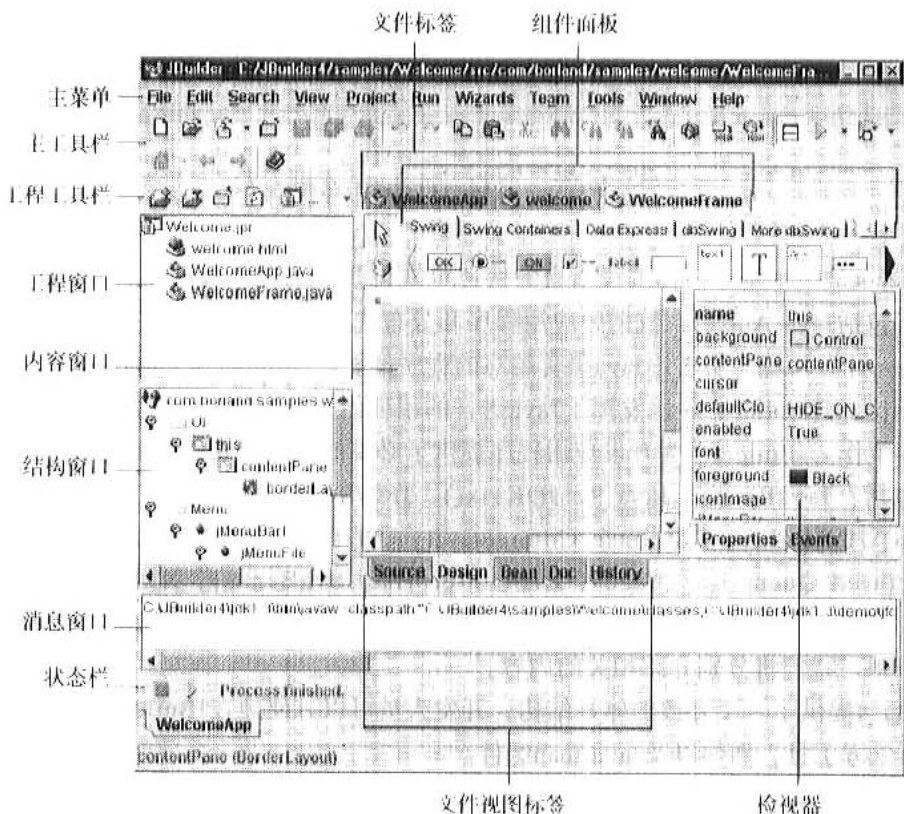


图1-1 JBuilder 4集成开发环境

- 内容窗口(Content pane): 在内容窗口中可以查看所打开的文件。每一个打开的文件都有一个显示其文件名的标签以及位于底部的各种视图标签。
- 文件视图标签: (File view tabs): 可以在内容窗口中查看一个文件的各种视图,如Source、Design、Bean、Doc或History视图。
- 文件标签(File tabs): 显示打开的文件名。JBuilder 4只显示活动工程的文件标签。要查看一个打开的文件,请选择它的文件标签。
- 消息窗口(Message pane): 是一个多标签页显示区域。它显示来自于子系统的消息,如设计器、搜索结果、编译器、调试器以及运行的进程。
- 状态栏(Status bar): 可获得有关进程及其结果的最新信息。

如果要在UI设计器中查看一个文件,请选择内容窗口中的Design标签页。Design页面显示的是文件的设计视图,组件板显示在Design页面的顶部,该面板只有在Design页中才能使用。

要创建一个用户界面,需要把组件面板上的组件拖放到内容窗口中,或拖放到结构窗口中的相关节点下面。JBuilder 4会自动地将所生成的代码插入到文件中。使用检视器(Inspector)可以调整组件的相关属性。

调试器在运行时显示在消息窗口中,多个调试会话显示为标签页,它们排列在AppBrowser的底部。要使用调试器,请选择Run | Debug Project菜单命令。JBuilder 4的调试界面如图1-2所示。

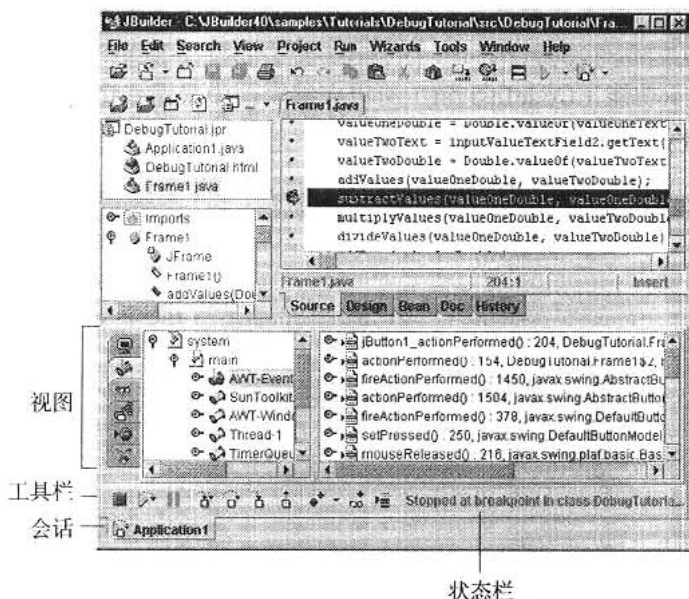


图1-2 JBuilder 4的调试界面

1.1.2 编辑器的使用

我们可以选择内容窗口的Source标签页访问编辑器,然后在编辑器中编写我们的Java应用程序。双向设计工具支持在可视化编辑器对代码的修改、同时可以反映到设计视图中。JBuilder 4编辑器支持文本搜索、语法高亮显示、代码模板、代码预示(CodeInsight)功能。JBuilder 4编辑器如图1-3所示。

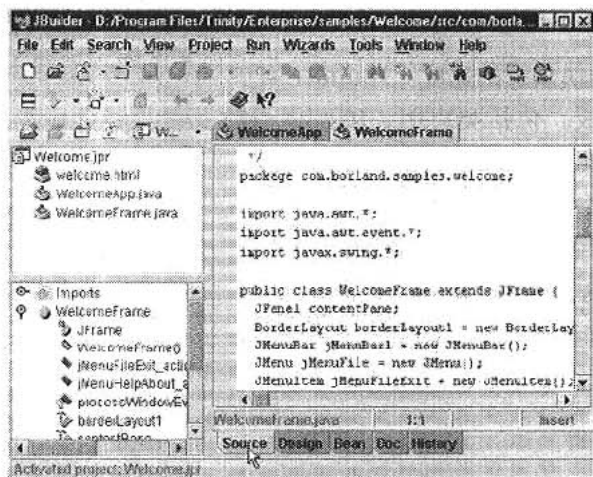


图1-3 JBuilder 4编辑器

程序开发的时间。我们可以使用这些向导快速创建和修改文件、设置应用程序和参数。JBuilder 4包括三种类型的向导：用于创建新文件的向导、用于创建应用组件的向导以及通用工具向导。

选择File | New菜单后，可以打开对象库(Object Gallery)对话框。其中包含以下两个页面：New和Enterprise。New页面包含使用Java 2标准版技术的向导。Enterprise页面包含使用Java 2企业版技术的向导。

在对象库中可以访问到用于创建新文件的向导包括：工程向导、Application向导、Applet向导、JavaBean向导；用于创建应用组件的向导包括：Dialog向导、Frame向导、Panel向导；通用工具向导包括：Archive Builder向导、Class向导、Interface向导、Data Module向导。

在File | New的Enterprise页面中还包含以下向导：EJB向导、CORBA向导、Sample IDL向导、JavaServer Page 向导等。JBuilder 4 的部分向导如图1-6所示。

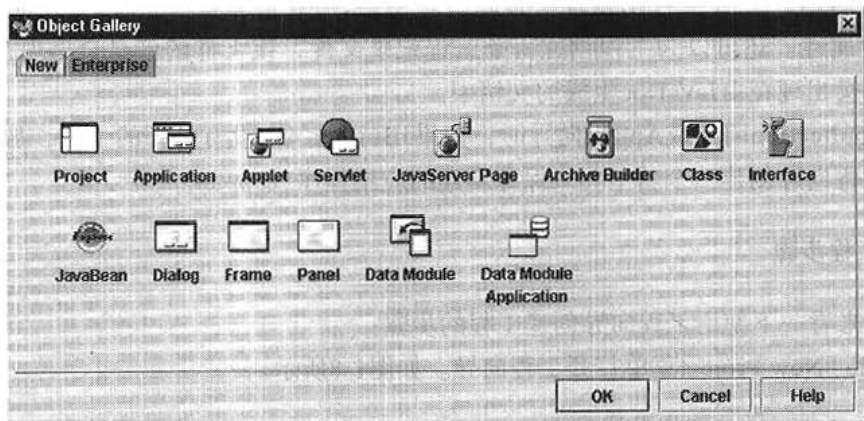


图1-6 JBuilder 4向导

还可以从Tools菜单及Wizards菜单中访问以下向导：New Library向导、New JDK向导、Implement Interface向导、Override Method向导、Archive Builder向导、Use DataModule向导、Interface向导和Test Client向导。

1.1.6 JBuilder 4提供的工具

在Tools菜单中访问JBuilder 4提供的部分工具，其中包括：BeanInsight用于检查JavaBean的有效性及有关属性；Package Migration用于调整旧版本JBuilder创建的文件，以使其与更新后的JDK以及JBuilder 4功能兼容；JDBC Monitor用于监控SQL应用；JDBC Explorer用来编辑数据，查看基于JDBC的元数据的信息、表、视图以及存储过程。在后面的章节中我们会涉及相关内容。

1.2 使用JBuilder 4工程

在JBuilder 4环境中，开发应用程序的第一步是创建一个工程。JBuilder 4用工程组织和维护应用程序。我们可以在工程窗口中查看、管理工程。在JBuilder 4工程中添加文件时，可以不改变文件在目录结构中的位置。这意味着多个工程可共享同一个文件。

工程文件保存了与工程有关的信息。其中包含工程所使用的文件与包以及工程属性。在装载、建立和分发工程时, JBuilder 4要用到这些信息。图1-7显示了一个工程的结构。

每个工程都保存在自己的目录中。每个工程目录包括一个工程文件(.jpr或.jpx)、一个保存工程记录的可选文件(.html)、一个保存类文件的classes目录、一个保存源文件的src目录以及一个保存备份的bak目录。缺省情况下, 工程目录保存在jbproject目录中。可以用工程向导与Project 1 Project Properties对话框改变这些路径。



图1-7 工程文件

JBuilder 4的工程向导简化了工程的创建。当使用工程向导创建新工程时, 向导会自动设置工程的目录框架, 创建并保存工程属性。还可以为工程创建一个工程记录文件, 该文件的信息会出现在工程源文件的注释中, 并会出现在由JavaDoc生成的文档中。

在使用Application向导或Applet向导时, 如果JBuilder 4中没有打开的工程, 那么JBuilder 4会首先启动工程向导创建一个新工程来保存新建的应用程序或applet。

1.2.1 创建工程步骤

下面我们来看一下使用工程向导创建工程的步骤:

1. 通过File | New Project菜单打开工程向导。如图1-8所示。

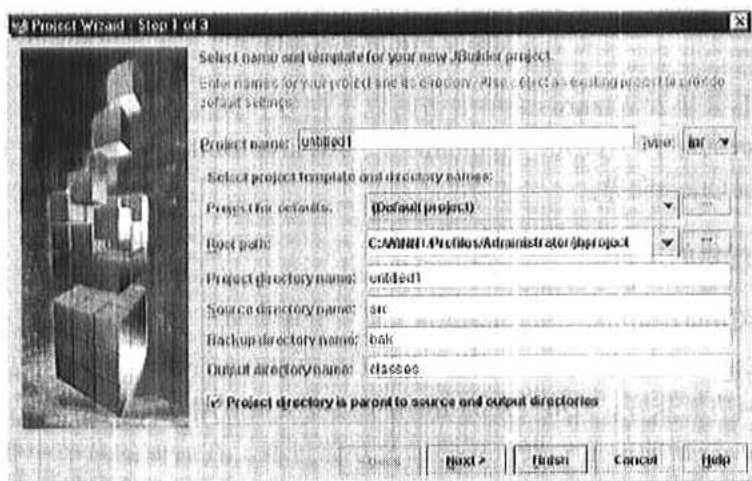


图1-8 工程向导(1)

在工程向导的第一个画面中, 我们需要输入工程名, 选择一个工程文件的类型: .jpr或.jpx。挑选一个工程模板。这可以从一个包含所有曾打开过的工程的列表中选择 一个工程作为模板, 也可以点击省略号(...)来选择一个所要用作模板的工程。还需要设置根路径和工程目录的名称。

点击Next按钮，进入工程向导的第2步。

2. 在接下来的页面中，可以设置工程的路径、编译所使用的JDK版本以及工程所需要的库。如图1-9所示。

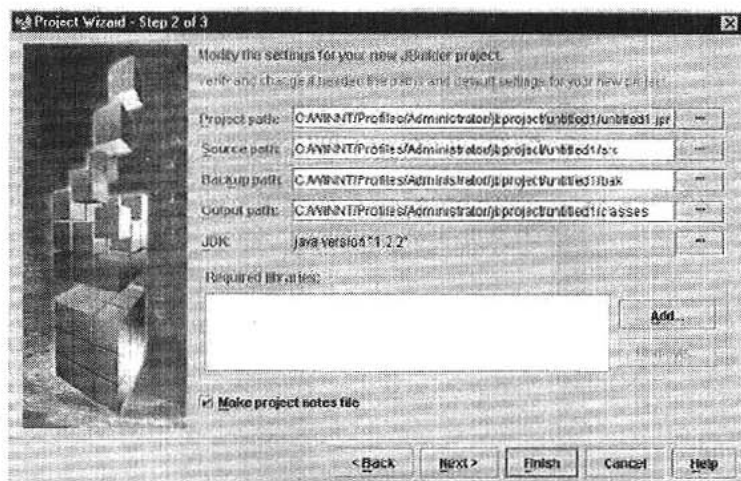


图1-9 工程向导(2)

3. 在步骤3的页面中，可以输入工程标题、作者与公司名称。在下面的框中填写该工程的描述。点Finish按钮，完成新建一个工程的所有工作。如图1-10所示。

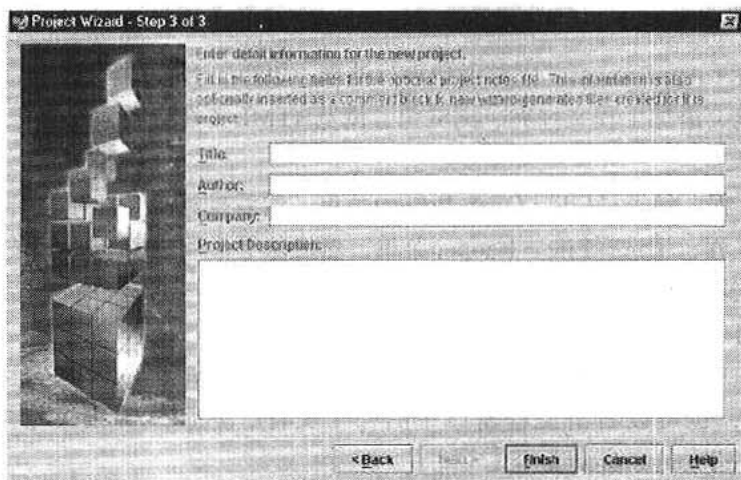


图1-10 工程向导(3)

1.2.2 设置工程属性

在Project Properties对话框中设置工程的属性。比如可以通过设置“路径”来设置输出路径、源文件路径、备份路径、JDK版本与库路径等，运行、调试、程序风格及EJB与JSP的处理设置。

用以下两种方法设置工程属性：(1)右击工程窗口中的.jpr或.jpx工程文件，然后选择

Properties命令；(2)选择Project | Project Options菜单。设置工程属性的界面如图1-11所示。

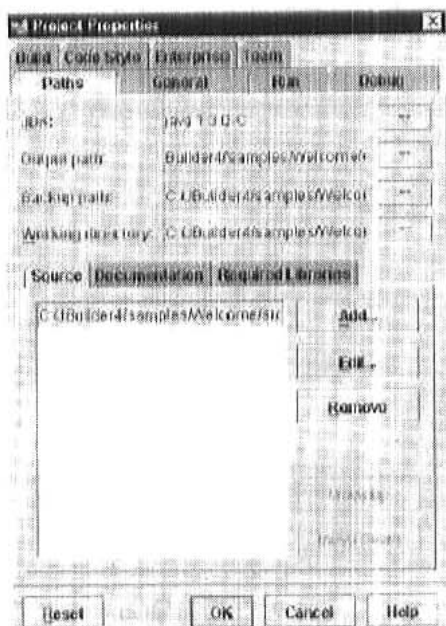


图1-11 设置工程属性

1.2.3 工程的管理

在应用浏览器（AppBrowser）中对工程进行处理，可以打开、编辑多个文件或工程。在工程中添加源文件与包、添加工程文件夹、浏览包、浏览HTML文件与Web图形等。

一、打开工程

要打开一个工程，请选择File | Open Project菜单，然后找到所要打开的工程。如果是打开一个曾经打开过的工程，请选择File | Reopen菜单，然后从下拉列表中选择工程文件。也可点击主工具栏上的Open与Reopen按钮来打开工程。

二、添加与删除文件

可以使用工程工具栏上的Add Files/Packages按钮或Remove From Project按钮在一个工程中添加或删除文件，也可以右击工程窗口中的一个文件，然后从弹出菜单中选择Add Files/Packages或Remove From Project命令。

三、保存以及关闭工程

可以用以下方法保存工程：选择File | Save菜单，选择File | Save Current Project菜单或点击主工具栏上的Save All按钮。

可以用以下方法关闭工程：选择File | Close Project、File | Close Files菜单或点击主工具栏上的Close Project按钮。

四、更改工程与文件的名称

如果需要更改工程的名称，可以首先在工程窗口中选择一个工程，然后选择Project |

Rename菜单。在Rename对话框的File Name栏中输入新的工程名。点OK按钮即可。

可以按以下步骤更改一个打开文件的文件名：选择File | Rename菜单或右击内容窗口上的文件标签，然后选择Rename命令。在Rename对话框中的File Name栏中输入新的文件名。点击Save按钮。

五、同时使用多个工程

在同时使用多个工程时，可以在一个应用浏览器实例中打开，也可以在不同的实例中打开它们。可以从任何一个打开的应用浏览器实例中使用所有已打开的工程。但是在每个应用浏览器的实例中，一次只能查看一个工程的文件。可以用以下方法在多个打开的工程或文件之间进行切换：从工程工具栏的下拉列表选择一个工程。如图1-12所示。

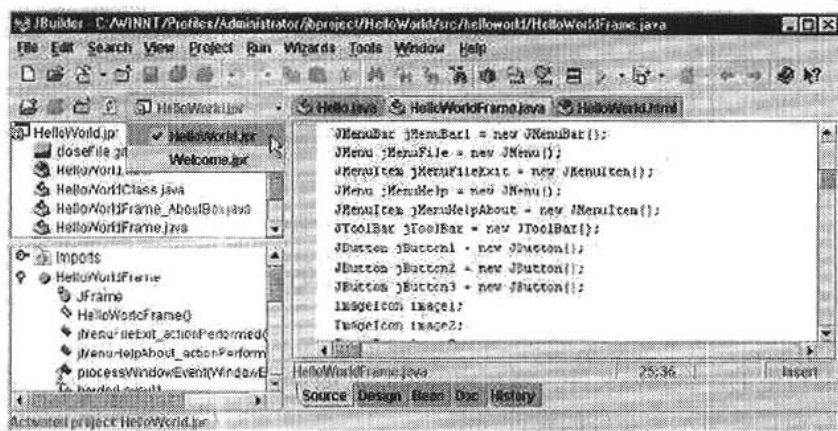


图1-12 使用多个工程

我们也可以从Window菜单中的打开文件列表来选择一个文件。还可以另打开一个应用浏览器实例或从Window菜单中切换打开的应用浏览器。

1.3 建用户界面

使用JBuilder 4的可视化设计工具可以快速创建Java应用程序的用户界面。在构建用户界面时，需要用到一个包含多种组件的面板，该面板上包含如按钮、文本框、列表、对话框以及菜单等组件。如图1-13所示。为了设置组件的属性和响应UI事件，还需要为组件的事件编写相应的事件处理程序。

在JBuilder 4的可视化设计工具中包含以下各部分：UI设计器(Designer)为放置和编辑窗口与其他UI组件提供了一个界面，可以选择内容窗口底部的Design页面，打开UI设计器。组件面板(Component palette)包含了可视与非可视Java组件。组件树(Component Tree)显示了源文件中的所有组件的结构化视图及其相互关系。组件树显示在应用浏览器左下部分中的结构窗口中。检视器(Inspector)用来查看与设置组件的属性，以及在方法与组件事件中建立连接。能够在设计中可视化地反映在监视器中所做的修改。菜单设计器用于在设计平面上设计菜单。双击组件树上的JMenuBar或JPopupMenu组件，或者在选中其中一个组件时按回车键将会启动菜单设计器。我们可以按照下面的方法操作UI设计器。

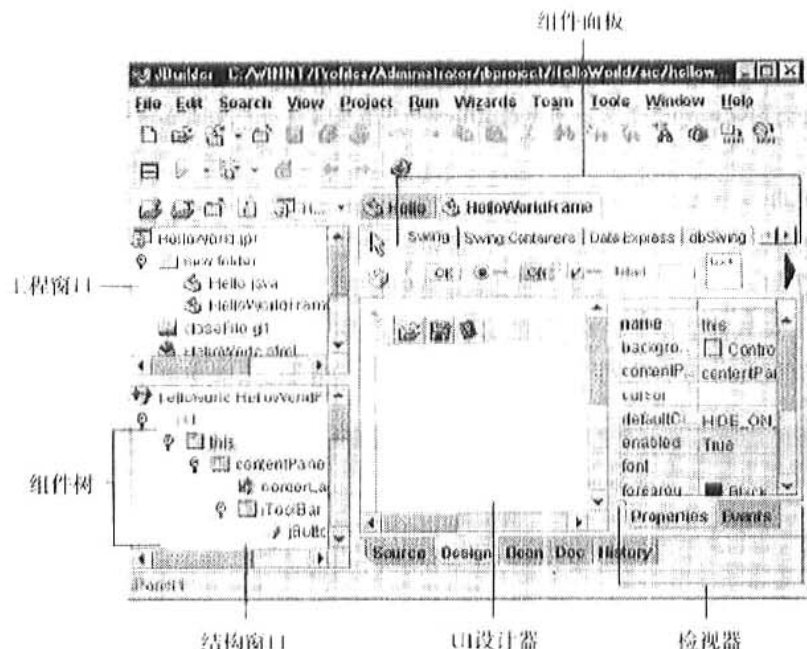


图1-13 应用浏览器与UI设计器

一、查看文件

双击工程窗口中的一个Java文件，将在内容窗口的源文件编辑器中打开该文件。选择内容窗口底部的设计页面，将显示该文件的设计视图。

二、添加与操作组件

选中组件面板的一个组件。点击设计器或在内容窗口中点击此组件的父组件，这将把选中的组件放置在设计器中。用结构窗口中的组件树追踪UI组件之间的关系。选中一个要应用布局管理器的容器，然后在检视器中选择并应用一个布局管理器。双击检视器中右边的栏位来查看可用的值或者激活文本域。在JBuilder 4中，设计器与Java源代码保持同步。当在UI设计器中改变了设计，JBuilder 4会自动修改源代码；当改变了源代码的内容，将会反映到UI设计器中。

1.4 使用菜单设计器

使用JBuilder 4的菜单设计器轻松地创建下拉菜单和弹出菜单。以下是打开菜单设计器的步骤：首先在工程窗口中双击要打开的文件。选择内容窗口底部的Design页面转换到设计器中。在设计器中添加一个菜单组件：点击组件面板上的一个菜单组件，然后点击设计器。双击组件树中新加入的菜单组件，就可以激活菜单设计器了。如图1-14所示。

1.5 使用组件检视器

可以使用组件检视器编辑组件属性，以及将代码与组件事件连接起来。当在检视器中对组件的属性和事件进行改动时，JBuilder 4会自动地在源程序中修改相应的代码。

使用检视器可以为容器、布局管理器以及容器中的组件设置初始属性值，创建、命名与删

除容器中的用来接收事件的事件处理代码等。在检视器中所做的任何改动都会立即反映到UI设计器和源代码中。可以通过下面的方式打开检视器：选中工程窗口中的一个文件，按Enter键在内容窗口中打开该文件。选择内容窗口底部的Design标签以访问设计器。这时监视器将显示在内容窗口的右边。如图1-15所示。我们可以拖动检视器的左边缘来调整检视器的大小。

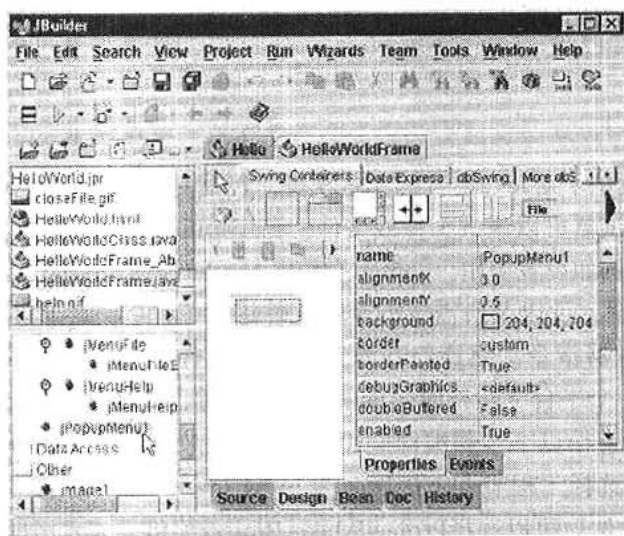


图1-14 激活菜单设计器

name	jPopupMenu1
alignmentX	0.0
alignmentY	0.5
background	204, 204, 204
border	custom
borderPainted	True
debugGraphicsOptions	<default>
doubleBuffered	False
enabled	True
font	"Dialog", 0, 12
foreground	Black
invoker	
lightWeightPopupEnabled	True
opaque	True
requestFocusEnabled	True
selectionModel	
toolTipText	

图1-15 组件检视器

1.6 使用布局管理器

用Java编写的应用程序可以分发到多种平台上。传统的UI设计技术必须指定组件的绝对位置与尺寸，这使UI在各种平台上的外观不一致。为了解决这个问题，Java提供了一个可移植的布局管理器系统。布局管理器可以正确地确定组件的位置，无论使用什么样的字体、屏幕分辨率

及平台。在运行的时候,当容器的大小改变时,它可以智能地确定组件的位置。

轻松转换不同长度的字符串。当字符串变长时,组件能够合适地对齐。在Java AWT以及Swing中,JBUILDER 4提供了以下布局管理器: BorderLayout、FlowLayout、GridLayout、CardLayout、GridBagLayout、Null、XYLayout(该布局保持容器中组件的原始尺寸与位置)、PanelLayout(该布局把一个容器分割为多个窗格)、VerticalFlowLayout(该布局垂直地排列组件)、BoxLayout2、OverLayout2。

1.7 编译和运行

JBUILDER 4编译器提供了对Java语言的完全支持,包括对内部类与Java Archive(JAR)文件的支持。用Java集成调试器可以调试工程或文件。在JBUILDER 4中,还可以对各种文件与程序进行相关的运行时配置,例如设置应用、applet、JSP、servlet和EJB的运行配置。

Java编译器对Java源文件以及所需要的其他文件进行编译,并生成由字节码组成的.class文件。字节码是JVM的机器指令。编译Java源文件会为文件中的每个类声明与接口声明生成一个.class文件。当你在一个平台上运行所生成的Java程序时,该平台的Java解释器运行.class文件中的字节码。我们可以遵循以下步骤编译应用源文件:首先打开一个工程,在工程窗口中,如果要编译一个应用,则选择工程节点(扩展名为.jpr或.jpx),如果是编译一个applet,选择调用该applet的HTML文件,然后可以选择Project/Make Project菜单,或者用鼠标右键点击一个节点,然后从弹出菜单中选Make命令实现编译,如图1-16所示。

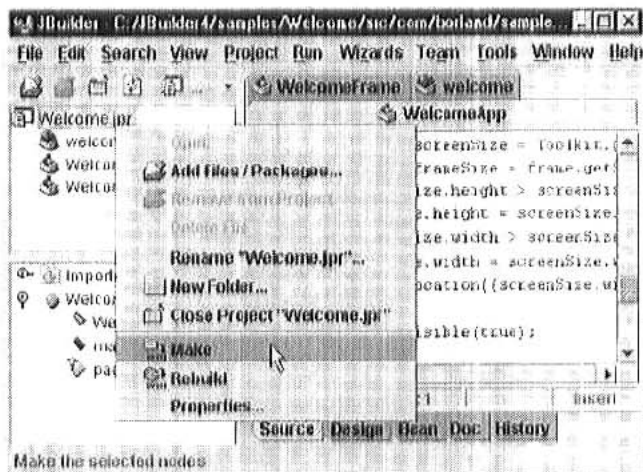


图1-16 编译工程

编译器的错误信息显示在应用浏览器的内容窗口下的消息窗口中。选择一个错误信息将显示与它相关的源代码。选中消息窗口中的一条错误信息,然后按F1键,会显示该错误信息的帮助内容。

运行一个工程就是运行该工程文件的主类。如果没有选定工程的主类,在运行此工程时,将弹出一个对话框要求为它选择一个主类。如果是利用Application向导或Applet向导创建文件的话,主类将自动被选定。要选择或改变主类,可以选择Project/Project Properties菜单的Run标签

页。如果要运行.java文件,如JSP文件,那么该文件必须包含main方法。如果需要运行一个applet,必须选择一个包含<applet>标签的HTML文件。该HTML文件会调用<applet>标签的CODE属性中的类。Applet类必须包含一个init()方法。

在JBuilder 4中运行Java程序的步骤通常是:首先保存所要运行的应用程序或文件。然后从工程窗口中选择要运行的应用程序或文件。如果要运行的是一个Java应用程序,那么先编译它(用Project | Make Project 菜单)。选择Project | Run Project,或右点工程窗口,然后从弹出菜单中选择Run命令。如果应用程序已经编译过,可以直接运行它。

运行错误信息也将显示在消息窗口中,选择一条错误信息将显示与该信息相关的源代码。

1.8 调试程序

调试是发现和修改程序错误的过程。可以在JBuilder 4环境中使用JBuilder 4集成调试器调试应用程序、applet、servlet以及JSP。许多调试功能可以从Run菜单访问,其他一些调试功能可从Search、View和Tools菜单中访问。

当需要调试一个文件时,通常需要遵循下面的步骤:首先打开一个工程,选择Project | Project Properties菜单命令。选择Run标签页,确定是否要在调试工程之前先编译它。可以设置源代码的断点。选择Run | Debug Project菜单命令或点击工具栏上的Debug图标。

文件的编译和调试与工程的编译和调试相同。如果JBuilder 4被设置为先编译后调试,应用浏览器底部的消息窗口的编译页会显示编译错误信息。通过点击一条错误信息,可以确定与该错误相关的代码行的位置。如果禁用了编译器,或者没有编译错误,调试器会显示在消息窗口中。可以用左边页面的工具提示看一下调试器提供了哪些信息。如图1-17所示。

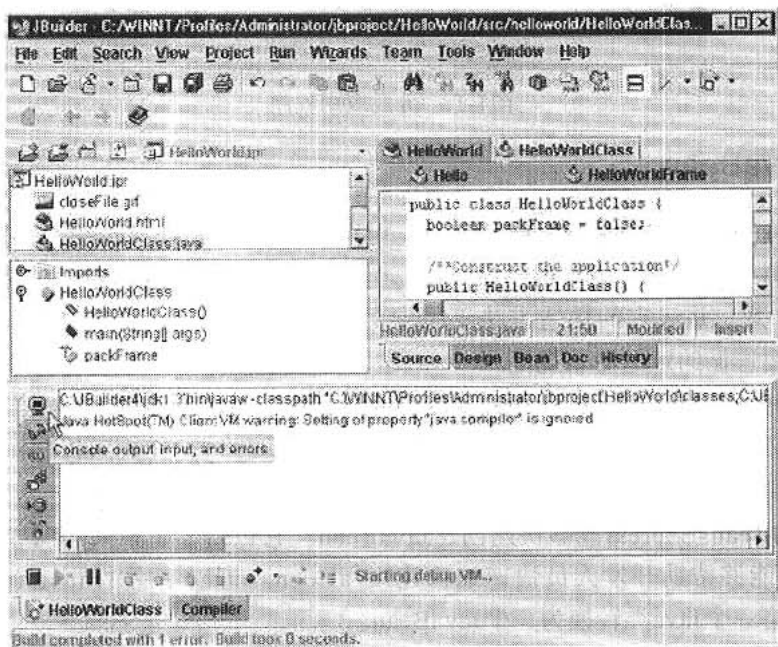


图1-17 JBuilder 4调试器

1.9 分发Java应用程序

应用程序的分发包括打包各种Java类文件、图像文件和工程所需要的其他文件，以及把它们拷贝到服务器或客户计算机上。所分发的应用程序可以是压缩的，也可以是没有压缩的存档文件。JBuilder 4的Archive Builder将程序所需的类及文件收集在一起，然后将这些文件保存到一个既可以压缩也可以不压缩的档案文件中，通常是一个JAR文件。而且还可以创建该档案的列表文件。Archive Builder会在工程中创建一个存档节点，从而简化了存档文件的访问。在开发过程中可以随时编译存档文件，或重新设置它的属性，还可以查看存档以及列表文件的内容。

通常分发应用程序的步骤如下：首先在JBuilder 4中创建与编译程序，运行Archive Builder来创建存档文件。如图1-18所示。创建安装程序，分发以下文件：JAR文件、所有必需的可再分发的JAR文件和安装文件。

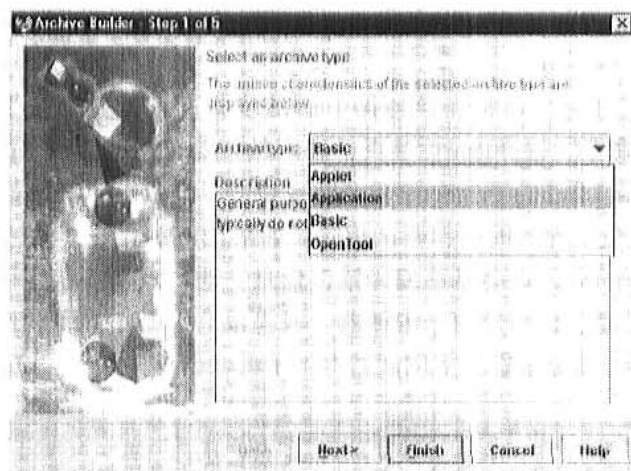


图1-18 分发应用程序

1.10 创建JavaBean

JavaBean是由一个或多个类所组成的独立、可重用的组件集合。JavaBean可以是一个用于构建用户界面的组件，也可以是像数据模块或计算引擎这样的非用户界面组件。最简单的JavaBean就是一个公共的类，并且它的构造方法不带参数。JavaBean通常具有属性、方法以及事件。JavaBean的事件要遵循约定的命名规范。

与其他组件相比，JavaBean具有一些独特的优点，如：JavaBean是纯Java、跨平台的组件。

可以把它们安装到JBuilder 4的组件板中，并用它们构建与设计应用程序；也可以在其他Java应用构建工具中使用它们。可以用JAR文件的形式分发它们。

JBuilder 4的BeansExpress能够快速创建JavaBean。它包括了一系列的向导、可视化设计器以及代码示例，能够帮助开发人员快速地构建JavaBean。可以使用BeansExpress修改已有的JavaBean，或者将一个Java类转换为一个JavaBean。

使用JBuilder 4的JavaBean向导新建JavaBean的步骤如下：首先选择File | New Project菜单打开工程向导新建一个工程。选择File | New菜单打开对象库。在对象库的New页面上双击