

# Kylix 程序设计

杨宗志 编著

清华大学出版社

(京)新登字 158 号

## 内 容 简 介

Kylix 是 Borland 公司新近推出的 Linux 系统上的可视化开发工具,它比 Linux 系统上的其他开发工具都更具有亲和力,而且能快速、轻松地建立 Linux 应用程序。本书从 Kylix 的安装、主界面开始介绍,由浅入深、循序渐进地介绍了如何用 Kylix 开发应用程序。最后,综合所介绍的基础知识,用 Kylix 创建了“我的记事本”和“我的小画家”两个应用程序,将理论与实践相结合。

本书适合所有 Kylix 程序设计的入门者及希望在 Linux 系统上快速开发应用程序的读者阅读。

本书繁体字版名为《Kylix 学习教本》,由文魁资讯股份有限公司出版,版权属杨宗志所有。本书简体字中文版由文魁资讯股份有限公司授权清华大学出版社独家出版。未经本书原版出版者和本书出版者书面许可,任何单位和个人不得以任何形式或任何手段复制或传播本书的部分或全部。

北京市版权局著作权合同登记号:图字 号

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

书 名:Kylix 程序设计

作 者:杨宗志

责任编辑:张秋香

出 版 者:清华大学出版社(北京清华大学学研大厦,邮编 100084)

<http://www.tup.tsinghua.edu.cn>

印 刷 者:

发 行 者:新华书店总店北京发行所

开 本:787×960 1/16 印张:21.75 字数:497 千字

版 次:2001 年 7 月第 1 版 2001 年 7 月第 1 次印刷

书 号:ISBN 7-302-04847-9/TP·2874

印 数:0001~

定 价:00.00 元

# 序 言

当作者得知 Borland 公司将推出 Linux 系统上的 Delphi——Kylix 时，非常兴奋。经实际操作使用后，发现 Kylix 同样拥有可视化的设计环境，比其他 Linux 系统上的任何开发工具都具有亲和力，而且能快速、简单地建立 Linux 应用程序。

毕竟 Kylix 还是个新软件(目前才正式发行第 1 版)，一定会有一些功能不够完善，再加上 Linux 操作系统的版本很多，使得 Kylix 的安装过程中发生很多问题。作者曾在 RedHat 6.2+CLE0.9pl、RedHat7.0+CLE1.0 等中文系统中安装过，都有很多问题。但尽管如此，Kylix 的开发环境、交互式除错和强大的编译功能，仍使得 Kylix 成为 Linux 系统中最受欢迎的开发工具。

作者站在非程序员的角度来编写本书，即使是程序设计的入门者，也可以轻易地通过阅读本书学会如何使用 Kylix 来开发 Linux 系统上的应用程序。

现将本书主要内容分列如下：

- 第 1 章 介绍如何安装 Kylix 及安装过程中容易出现的问题，并介绍了 Kylix 的开发环境。
- 第 2 章 介绍程序的界面设计，包括窗体的设计、组件的布置等。
- 第 3 章 介绍程序设计的规则，包括程序单元的基本内容、变量及 Kylix 的工作模式等。
- 第 4 章 介绍触发事件及组件属性设置语句。
- 第 5 章 介绍利用 If、Case Of、For 语句编写具有判断能力的程序，并介绍如何跟踪程序的流程。
- 第 6 章 介绍 Kylix 中的各种事件程序。
- 第 7 章 介绍基本组件的应用，包括多样化按钮、选择类组件等。
- 第 8 章 介绍列表框、状态栏、计时器等基本组件的应用。
- 第 9 章 介绍菜单及工具栏的制作，包括下拉菜单、快捷菜单、工具栏组件等。
- 第 10 章 介绍如何利用各种组件来完成一个记事本程序。
- 第 11 章 综合本书各章的主要内容，制作一个简易的画图软件。

# 目 录

第 1 章 初识 Kylix.....	1	2.3.4 组件选择器和选中的组件.....	39
1.1 安装 Kylix .....	2	2.4 组件的外观设置.....	40
1.2 集成开发环境 .....	6	2.4.1 属性值设置方式 .....	41
1.2.1 主窗口的菜单 .....	7	2.4.2 设置组件属性值 .....	44
1.2.2 主窗口的工具栏 .....	9	2.5 保存程序和工程管理.....	46
1.2.3 窗体窗口 .....	15	2.5.1 工程与工程管理器 .....	47
1.2.4 组件查看器.....	15	2.5.2 打开旧工程 .....	49
1.2.5 程序代码查看与程序代码 编写窗口 .....	16	第 3 章 程序设计规则.....	51
1.3 方便易用的工具.....	18	3.1 无师自通的程序.....	52
1.3.1 对齐工具.....	18	3.2 工程的结构.....	53
1.3.2 工程管理器.....	19	3.3 程序单元的基础内容.....	56
1.3.3 窗口列表.....	20	3.3.1 interface 段 .....	58
1.3.4 组件列表.....	20	3.3.2 implementation 段 .....	59
1.3.5 工作提示.....	21	3.3.3 窗体与程序编写单元的 交互 .....	59
第 2 章 程序界面设计 .....	25	3.3.4 过程与程序编写单元.....	61
2.1 窗体设计所需的工具.....	26	3.4 初识 Object Pascal.....	63
2.2 组件的摆放 .....	27	3.4.1 代码的结束符号 .....	63
2.2.1 加入所需的组件 .....	27	3.4.2 赋值号 .....	63
2.2.2 改变组件的位置 .....	28	3.4.3 注释符号 .....	64
2.2.3 调整组件大小 .....	28	3.5 变 量.....	64
2.2.4 快速布置法 .....	29	3.5.1 变量的命名规则 .....	65
2.2.5 对齐组件 .....	31	3.5.2 变量的数据类型 .....	65
2.2.6 锁定组件 .....	33	3.6 Kylix 的运行模式.....	68
2.2.7 组件的比例缩放 .....	34	3.6.1 运行、暂停、跟踪、单步执行工 具按钮 .....	68
2.3 Object Inspector 窗口的操作 .....	35	3.6.2 设计、运行、暂停操作模式..	69
2.3.1 查看属性栏 .....	36	第 4 章 组件属性及对话框设置.....	73
2.3.2 属性值的设置 .....	37		
2.3.3 组件选择器.....	38		

4.1	触发事件 .....	74	6.1.2	事件程序的名称 .....	145
4.1.1	更改标题 .....	74	6.1.3	标题和名称 .....	146
4.1.2	与用户对话 .....	77	6.2	键盘的事件程序 .....	148
4.2	组件属性设置语句 .....	81	6.2.1	KeyPress 事件 .....	148
4.2.1	静态组件属性设置 .....	81	6.2.2	KeyDown 事件 .....	150
4.2.2	组件属性设置语句 .....	84	6.2.3	KeyUp 事件 .....	154
4.2.3	设置属性值的方式 .....	85	6.3	鼠标的事件程序 .....	157
4.3	信息对话框 .....	91	6.3.1	MouseDown 事件 .....	157
4.3.1	ShowMessage 过程 .....	91	6.3.2	MouseMove 事件 .....	161
4.3.2	MessageDlg()函数 .....	93	6.3.3	MouseUp 与 Click 事件 .....	163
4.3.3	MessageBox()函数 .....	101	6.4	拖 动 .....	166
4.4	输入对话框 .....	103	6.4.1	拖动属性设置 .....	166
<b>第 5 章</b>	<b>程序流程控制 .....</b>	<b>107</b>	6.4.2	DragOver 事件程序 .....	167
5.1	条件判断语句 If .....	108	<b>第 7 章</b>	<b>基本组件应用(上) .....</b>	<b>175</b>
5.1.1	If-Then 语句 .....	108	7.1	多样化的按钮 .....	176
5.1.2	If-Then 语句段 .....	110	7.1.1	BitBtn 的默认按钮图标 .....	176
5.1.3	If-Then-Else 语句 .....	113	7.1.2	BitBtn 的图标和文字位置 .....	179
5.1.4	If 语句嵌套 .....	116	7.1.3	SpeedButton 的特效属性 .....	180
5.1.5	Else If 逐条过滤条件语句 .....	117	7.2	文本类组件 .....	183
5.2	Case-Of 语句 .....	119	7.2.1	Label 与 StaticText .....	183
5.3	重复执行的循环语句 .....	125	7.2.2	Edit 与 MaskEdit .....	184
5.3.1	计数执行的 For-Do 语句 .....	125	7.2.3	Memo 组件 .....	187
5.3.2	While-Do 循环语句 .....	127	7.3	选择类组件 .....	193
5.3.3	Repeat-Until 循环语句 .....	130	7.3.1	GroupBox 集合组件 .....	194
5.4	程序流程跟踪 .....	130	7.3.2	单选按钮 .....	195
5.4.1	插入断点 .....	130	7.3.3	RadioGroup 集合组件 .....	198
5.4.2	监视变量值 .....	135	7.3.4	复选框 .....	204
5.4.3	表达式求值 .....	136	7.3.5	CheckListBox 集合组件 .....	209
5.4.4	监视并更改变量值 .....	136	<b>第 8 章</b>	<b>基本组件应用(下) .....</b>	<b>215</b>
5.4.5	跟踪 .....	137	8.1	列 表 框 .....	216
5.4.6	单步执行 .....	137	8.1.1	加入列表项目 .....	216
<b>第 6 章</b>	<b>事件程序 .....</b>	<b>139</b>	8.1.2	利用程序语句加入项目 .....	217
6.1	事件程序 .....	140	8.1.3	改变待选项目位置 .....	218
6.1.1	内置的事件程序 .....	140	8.1.4	删除及清除待选项目 .....	219

8.1.5 读取待选项目数据 .....	220	10.4 我的记事本.....	283
8.1.6 选择多个列表框项目 .....	223	10.4.1 程序外观设计 .....	284
8.1.7 项目选中判断 .....	224	10.4.2 “文件”菜单程序代码 的编写 .....	288
8.1.8 项目排序 .....	227	10.4.3 “编辑”和“窗口”菜单 程序代码的编写 .....	293
8.2 下拉列表框 .....	229	10.4.4 最后的修饰 .....	296
8.2.1 下拉列表框的特点 .....	230	<b>第 11 章 程序实例——我的小画家</b> .....	<b>303</b>
8.2.2 下拉列表框的使用方法 .....	231	11.1 绘图语句.....	304
8.2.3 下拉列表框的样式 .....	233	11.1.1 画点与直线的 DrawPoint 方法 .....	304
8.3 状态栏 .....	236	11.1.2 画线专用的 LineTo 方法 .....	305
8.3.1 配置状态栏 .....	237	11.1.3 Rectangle 方法 .....	310
8.3.2 利用程序语句控制状态栏 .....	239	11.1.4 RoundRect 方法 .....	312
8.3.3 面板的斜角与对齐 .....	240	11.1.5 Ellipse 方法 .....	314
8.4 分隔条 .....	241	11.1.6 Arc 与 Chord 方法 .....	315
8.5 计时器 .....	245	11.1.7 Polyline 与 Polygon 方法 .....	316
<b>第 9 章 菜单与工具栏</b> .....	<b>251</b>	11.2 我的小画家第 1 版.....	318
9.1 主菜单 .....	252	11.3 绘图专用组件.....	324
9.1.1 菜单设计组件 .....	252	11.4 我的小画家第 2 版.....	326
9.1.2 建立菜单项目 .....	254	11.4.1 ColorDialog 组件 .....	326
9.1.3 快捷键、分隔线、灰色 菜单项 .....	257	11.4.2 为我的小画家加上颜色选 择功能 .....	328
9.2 快捷菜单 .....	259	<b>附录 A 常用函数</b> .....	<b>337</b>
9.3 为菜单项目加上图标.....	261	<b>附录 B Color 属性颜色说明</b> .....	<b>339</b>
9.3.1 ImageList 组件 .....	262	<b>附录 C 运算符</b> .....	<b>341</b>
9.3.2 产生图标 .....	264		
9.4 工具栏组件 .....	266		
9.4.1 工具栏组件 .....	266		
9.4.2 复选工具按钮 .....	268		
9.4.3 Grouped 工具按钮 .....	271		
<b>第 10 章 程序实例——我的记事本</b> .....	<b>273</b>		
10.1 创建 MDI 窗体程序.....	274		
10.2 子窗体的建立设置.....	276		
10.3 子窗体的管理.....	279		
10.3.1 为子窗体命名 .....	281		
10.3.2 排列子窗口 .....	282		

# 第 1 章 初识 Kylix

以前，要开发一个 Linux 操作系统上的应用程序是非常困难的。首先，必须要懂得高级的 C 语言；其次，传统的 Linux 应用程序开发工具非常难以学习和使用，而且没有我们在 Windows 操作系统上习惯使用的图形界面。但是当您用 Kylix 在 Linux 操作系统上开发应用程序时，将会发现编写程序是那样的简单。Kylix 就像 Windows 操作系统上的 Delphi，同样提供许多操作简单而功能强大的组件，用这些组件就可编写出属于您自己的程序。

- 安装 Kylix
- 集成开发环境
- 方便易用的工具

在安装 Kylix 之前，必须先检查一下系统的硬件环境。以下是 Borland 公司对于安装 Kylix 的硬件配置建议：

- Intel Pentium 200 以上兼容微处理器(建议采用与 Intel Pentium II 400 以上兼容微处理器)
- 64 MB 内存(建议使用 128 MB 以上)
- 最小安装需要 80 MB 硬盘空间，完全安装需要 230 MB 硬盘空间
- Linux 兼容的 VGA 或更高级的显示卡
- Linux 兼容的鼠标
- Linux 兼容的光驱

对操作系统的要求，Borland 公司提出以下的建议：

- 支持英文的 Linux 版本
  - ◆ RedHat 6.2 以上版本
  - ◆ Mandrake 7.2 以上版本
  - ◆ SuSE(r)7.0 以上版本
- 支持中文的 Linux 版本
  - ◆ RedHat 6.2 附加 CLE(GNUChineseLinuxExtension)0.9p1



### 注 意

本书内容是按照 Borland 公司所建议的中文系统环境，将 Kylix 安装在 RedHat 6.2 附加 CLE(GNUChineseLinuxExtension)0.9p1(Yami)的运行环境中，也可以安装在 RedHat 7.0 附加 CLE1.0(Ataya)的运行环境中。本书所使用的桌面环境是 KDE 的 X Window 图形界面。

## 1.1 安装 Kylix

在 Windows 系统中安装软件，一般都是直接将安装光盘放入光驱中就会自动开始安装软件了。但是，在 Linux 操作系统中，要使用光驱则必须经过安装(mount)才能使用。如何

安装光驱不在本书的讨论范围内。但是不用担心，只要将光盘放入光驱中，Linux 系统就会自动将光驱安装进来，同时会出现文件管理器(File manager)窗口，把光盘根目录下的文件夹及文件显示出来，如图 1.1 所示。



图 1.1 Linux 系统自动安装光驱并显示出光盘中的文件

请注意，在地址(Location)栏中所见到的 file: /mnt/cdrom 就是光驱安装到 Linux 文件系统目录的起始位置。也就是说，file: /mnt/cdrom 是光盘内容的根目录。

在光盘目录中见到的 setup.sh 文件，就是 Kylix 的安装文件。在正常的情况下，只要单击这个文件就可以进行 Kylix 的安装。可是，作者按照 Borland 公司的系统环境要求使用 RedHat 6.2 附加 CLE0.9p1 的运行环境却无法安装 Kylix。这是因为 glibc 组件有问题，在运行 Kylix 安装程序之前必须先更新 glibc 的版本。这些.rpm 文件就放置在 Kylix 安装光盘的/patches/glibc\_redhat/6.2 目录下，只要直接运行 glibc-2-1.3-22b.i386 更新组件就可以了，如图 1.2 所示。

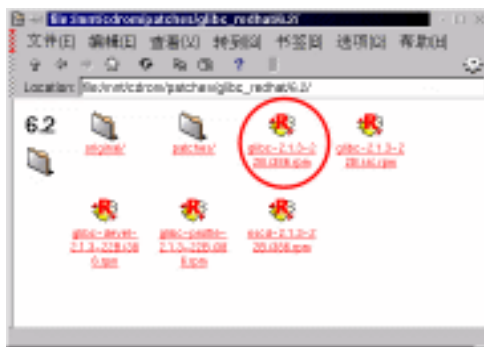


图 1.2 Kylix 光盘中提供的 glibc 更新组件

更新 glibc 后,将文件管理器的显示路径切换回光盘的根目录,也就是 file:/mnt/cdrom,在目录中单击 setup.sh 文件开始安装 Kylix。如果以终端机的字符模式进行安装,将会见到 Kylix 安装前的环境系统检查信息,不过必须先切换工作路径到光盘目录,然后运行 sh setup.sh 命令,如图 1.3 所示。

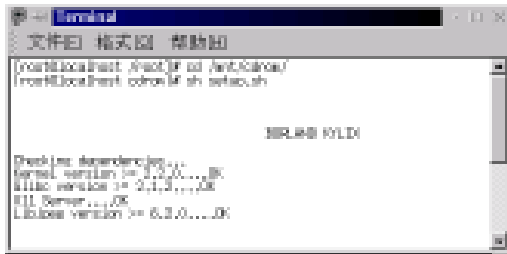


图 1.3 在终端机模式下运行安装程序

检查过安装环境没有问题之后,第 1 个出现的安装界面就是版权声明画面,如图 1.4 所示。

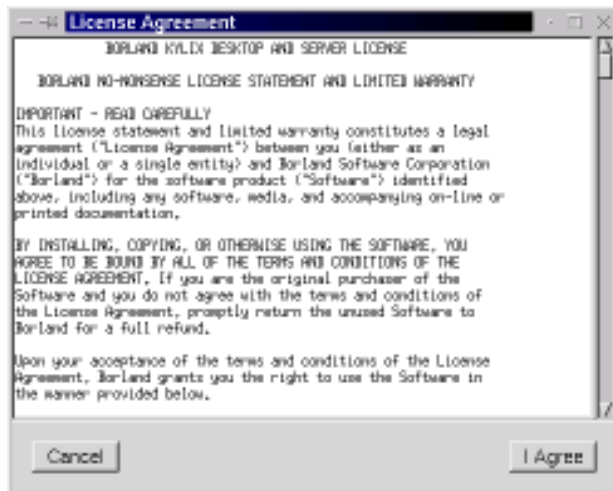


图 1.4 License Agreement 对话框

单击 I Agree 按钮后进入安装选项设置对话框。作者建议:如果您没有特殊的需求,就直接单击 Begin Install 按钮,按照默认的条件安装就行,如图 1.5 所示。

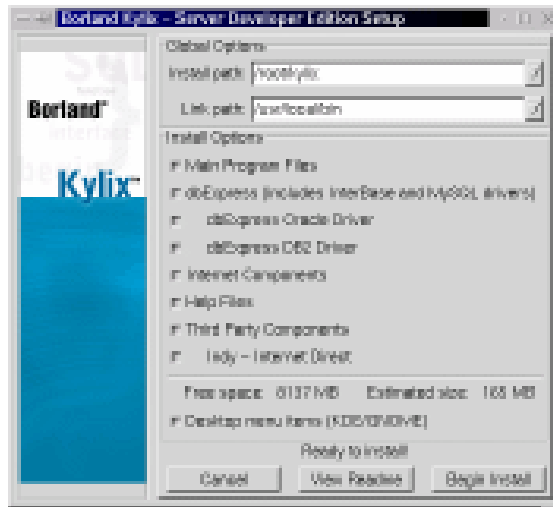


图 1.5 安装选项设置对话框

单击 Begin Install 按钮后，接着出现安装进度对话框，如图 1.6 所示。

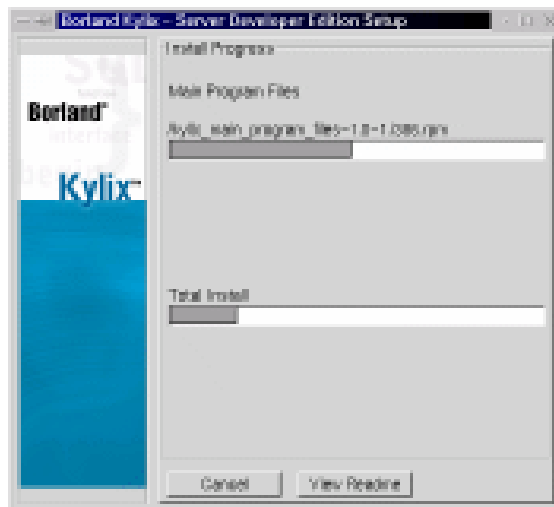


图 1.6 安装进度对话框

所有的程序组件安装结束后，出现安装结果的提示对话框，告诉用户 Kylix 的安装路径在哪里，要如何启动 Kylix 等，如图 1.7 所示。最重要的是我们所使用的桌面环境是 KDE 的 X Window 图形界面，所以安装完毕之后要重新启动 X Window 图形界面。

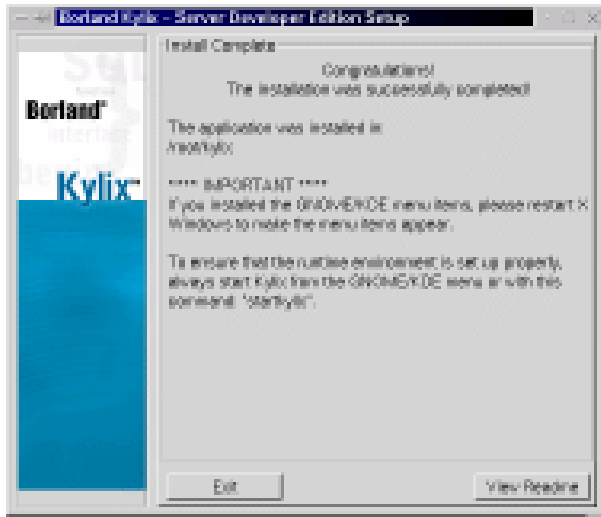


图 1.7 安装结果提示对话框

## 1.2 集成开发环境

安装完 Kylix 后选择 K 菜单中的 Borland Kylix | Kylix 命令，就可以直接启动 Kylix 了，如图 1.8 所示。



图 1.8 启动 Kylix

进入 Kylix 的开发环境后，您将发现 Kylix 的图形化开发环境很友好。如果您使用过 Windows 系统上的 Delphi，看到这样的开发环境一定感觉很亲切，两者的界面几乎一模一样。Kylix 开发环境的整体外观如图 1.9 所示。

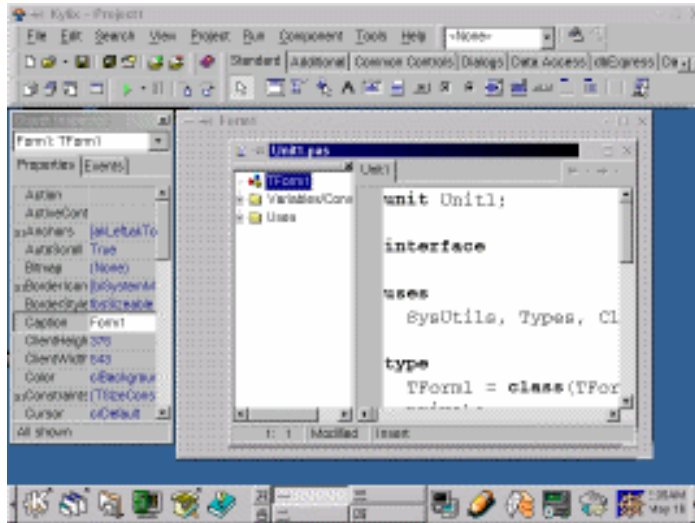


图 1.9 Kylix 的开发环境

进入 Kylix 后，共有 4 个工作窗口，分别是主窗口(标题栏上显示 Kylix - Project1 的窗口)、对象查看器(标题栏上显示 Object Inspector 的窗口)、窗体窗口(标题栏上显示 Form1 的窗口)、程序代码单元窗口(标题栏上显示 Unit1.pas 的窗口)。请注意：单击主窗口上的关闭按钮将关闭整个 Kylix 程序，而其他的窗口单元都只是 Kylix 的子窗口，用户可以根据自己的需要决定子窗口是否显示。

### 1.2.1 主窗口的菜单

Kylix 主窗口的菜单如图 1.10 所示。

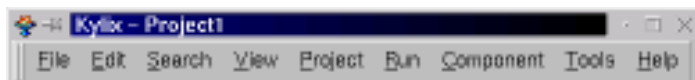


图 1.10 Kylix 主窗口的菜单

## 1. File(文件)

File 菜单命令包括打开、保存、打印、关闭新的和旧的工程或文件。也就是说，所有与文件相关的操作都在这个菜单内，如图 1.11 所示。

## 2. Edit(编辑)

我们常用的剪切、复制、粘贴的操作命令都在编辑菜单里。这个菜单内还提供在程序设计阶段时，可以用来编辑程序及组件属性的命令，如图 1.12 所示。

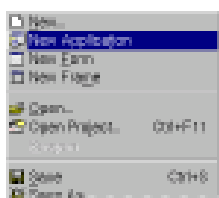


图 1.11 File 菜单



图 1.12 Edit 菜单



图 1.13 Search 菜单

## 3. Search(搜索)

在我们编写程序代码时将会经常用到搜索菜单中的命令，其中包括查找文字、错误、对象、变量、符号、程序单元，以及快速将光标移到指定的程序代码行等命令，如图 1.13 所示。

## 4. View(查看)

查看菜单用来显示或隐藏 Kylix 的各种工具栏及相关的工具窗口，例如工程管理器 (Project Manager)窗口等，如图 1.14 所示。

## 5. Project(工程)

必须先打开一个工程才能使用这个菜单内的各命令，工程菜单内的命令主要用来编译工程和管理工程，如图 1.15 所示。

## 6. Run(运行)

运行菜单包括各种不同的运行程序的方法，尤其是 Program Reset 命令，当程序在运行

中发生重大错误时，可利用这个命令强制中断程序的运行，如图 1.16 所示。



图 1.14 View 菜单

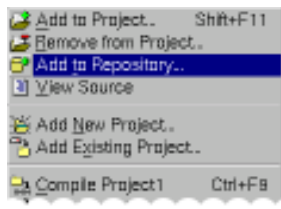


图 1.15 Project 菜单

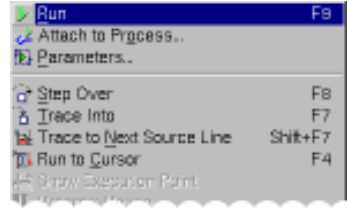


图 1.16 Run 菜单

## 7. Component (组件)

组件管理菜单包含组件制作、单独组件安装和组件包安装等命令，如图 1.17 所示。

## 8. Tools(工具)

工具菜单主要包括一些用来管理工具栏及设置各个工具栏的命令，并可利用其中的 Configure Tools 命令来添加常用的程序，如图 1.18 所示。

## 9. Help(帮助)

联机帮助菜单如图 1.19 所示。

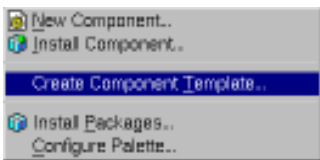


图 1.17 Component 菜单

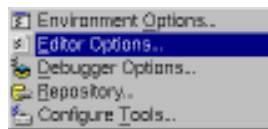


图 1.18 Tools 菜单

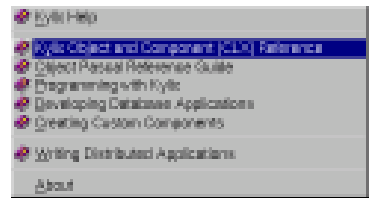


图 1.19 Help 菜单

### 1.2.2 主窗口的工具栏

在 Kylix 的主窗口中，共有 6 个工具栏，如图 1.20 所示。分别是 View(查看)工具栏、Standard(标准)工具栏、Debug(调试)工具栏、Custom(定制)工具栏、Component Palette(组件选项板)工具栏。每个工具栏都可脱离主窗口而成为单独的浮动窗口，并可按照个人的爱好

进行工具栏的排列组合。

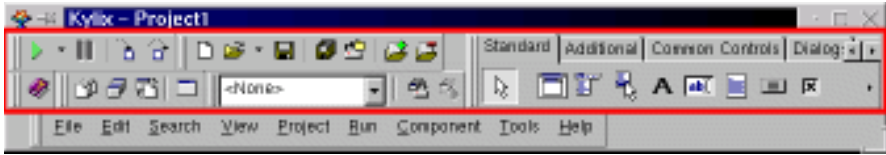


图 1.20 Kylix 主窗口中的工具栏

### 1. Standard 工具栏

Standard 工具栏中的工具按钮就是由 File 菜单中的 New、Open、Reopen、Save、Save All、Open Project 等 6 个命令和 Project 菜单中的 Add to Project、Remove from Project 两个命令所组合而成，用来管理与工程文件相关的文件，如图 1.21 所示。

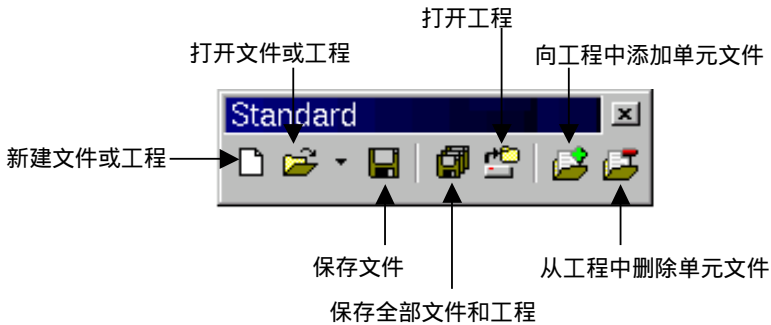


图 1.21 Standard 工具栏

单击新建文件或工程按钮将会弹出如图 1.22 所示的 New Items 对话框。

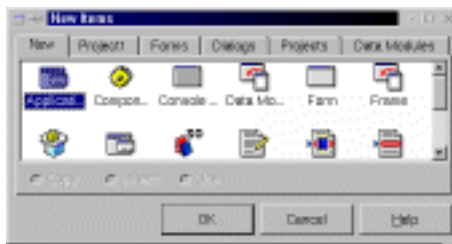


图 1.22 New Items 对话框

## 2. View 工具栏

View 工具栏中的工具按钮是由 File 菜单中的 New Form 命令和 View 菜单中的 Toggle Form/Unit、Units、Forms 等 4 个命令所组合而成，如图 1.23 所示。

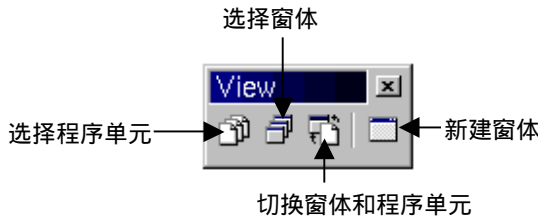


图 1.23 View 工具栏

在互相对应的窗体与程序单元间可使用 Toggle Form/Unit 按钮来做快速切换，特别是操作环境中有很多不同单元窗体时。如果要在工程中添加新的窗体则可单击 New Form 按钮。

如果我们设计的程序(工程)中有很多的窗体及程序单元，那么使用 Units 与 Forms 按钮是很好的选择。单击 Units 或 Forms 工具按钮将弹出 View Unit 或 View Form 对话框，如图 1.24 所示，我们可从列表中直接选择所需的窗体或单元。

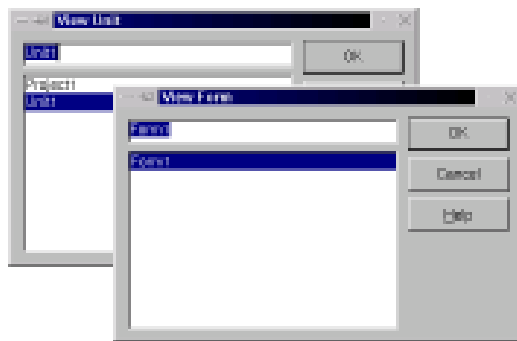


图 1.24 View Form 和 View Unit 对话框

## 3. Component Palette 工具栏

Component Palette 工具栏是最重要的一个工具栏，其中放着各式各样的组件。组件的分类比较明确，各种不同性质的组件摆放在不同的选项卡中。要选择不同的选项卡只要使