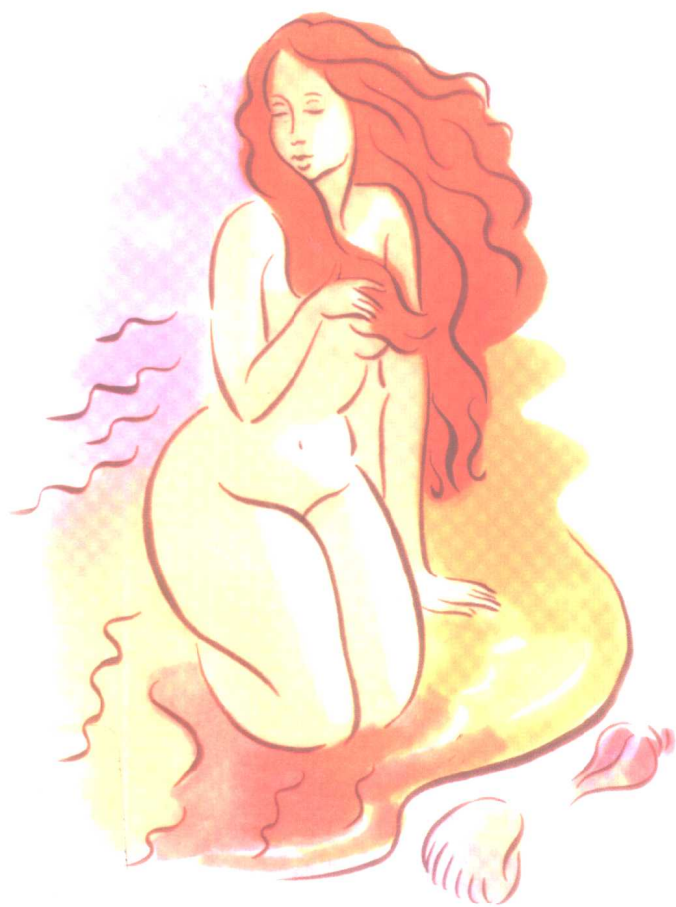


Gtk+ Programming in C

Gtk+

程序设计

(C语言版)



- ◆ 编写Linux和UNIX图形应用程序
- ◆ 充分发挥 Gtk 1.2, GLIB和GDK的强大功能
- ◆ 包括丰富的 Gtk+ 构件: 解释, 例程和参考
- ◆ 还包括Linux/UNIX下的C编程入门/复习的知识

Syd Logan 著
战晓苏 王宁 等译



清华大学出版社
<http://www.tup.com.cn>



TP316.89
55

Gtk+程序设计

(C 语言版)

Syd Logan 著

战晓苏 王宁 等译

清华大学出版社

(京) 新登字 158 号

北京市版权局著作权合同登记号: 01-2002-3004

内 容 简 介

本书全面详细地介绍了 Gtk+ 1.2 版本, 内容包括利用 Gtk+ 1.2 编写应用程序的简单介绍、Gtk+ 1.2 构件集的详细讲述, 并且为熟悉 Gtk+ 的程序员提供了快速参考。

本书内容包括 Gtk+ 开发环境, Gtk+ 基础知识, 信号、事件、对象和类型, 以及各种构件、容器和子类。对于每一种对象、类型等等, 本书都进行了详尽的讲述, 介绍了用 C 语言使用这些对象、类型编程的知识。本书图文并茂, 易于读者理解。

本书还包括 Gtk+ 1.2 构件体系, 是 Linux 管理员和程序开发人员的参考手册。也是一本内容详尽并具有参考价值的专业书籍, 特别适合于爱好 Linux 的用户学习使用, 同时也适合于广大科研人员、大专院校相关专业师生学习及教学时使用。

Simplified Chinese edition copyright © 2002 by Pearson Education NORTH ASIA LIMITED and Tsinghua University Press.

GTK+ Programming in c first publication by Syd Logan, Copyright © 2002.

All Rights Reserved.

Published by arrangement with Pearson Education, Inc., publishing as PH PTR.

This edition is authorized for sale only in the People's Republic of China (excluding the Special Administrative Region of Hong Kong and Macau).

本书中文简体字版由美国培生教育集团授权清华大学出版社出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

版权所有, 翻印必究。

本书封面贴有 Pearson Education 出版集团激光防伪标签, 无标签者不得销售。

图书在版编目(CIP)数据

Gtk+程序设计(C语言版)/(美)洛根编著;战晓苏等译. — 北京:清华大学出版社, 2002

书名原文: GTK+ Programming in C

ISBN 7-302-06037-1

I. G... II. ①洛... ②战... III. ①Linux 操作系统程序设计 ②C 语言—程序设计 IV. TP316.81

中国版本图书馆 CIP 数据核字(2002)第 084549 号

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

责任编辑: 陈宗斌

印 刷 者: 北京牛山世兴印刷厂

发 行 者: 新华书店总店北京发行所

开 本: 787×1092 1/16 印 张: 47 字 数: 1202 千字

版 次: 2002 年 12 月第 1 版 2002 年 12 月第 1 次印刷

书 号: ISBN 7-302-06037-1/TP·3602

印 数: 0001~3000

定 价: 98.00 元

译者序

现今，个人计算机的操作系统几乎被微软的 Windows 系统所垄断。但是，当我们细心观察就会发现，以开放源代码为宗旨的 Linux 操作系统正在崛起，Linux 系统的用户数量正在以惊人的速度增长。Linux 所具有的友好并且高度可定制的桌面环境吸引了越来越多的用户。在 Linux 中，用户可以自行设置大多数的环境变量，编制个性化的 shell，甚至可以通过编程重新编译内核。

而 Linux 最令人激动的桌面环境大都出自 Gtk+、Qt 和 Motif 之手。最流行的桌面环境 KDE 是用 Qt 开发的。而 GNU 所认定的标准桌面环境 GNOME 就是用本书所介绍的 Gtk+ 开发的。Gtk+ 工具包的设计目的是在 X Window 系统下开发应用程序，它只是整个开发环境的一部分。要真正了解 Gtk+，读者需要具备 X Window 系统的基础知识。

本书介绍的核心内容是 Linux 平台下的 X Window 图形窗口编程工具 GTK+，文中详细地给出了用 GTK+ 编程的基本要素和步骤，介绍了 GIMP Toolkit(Gtk+) 的 1.2 版本。

Gtk+ 最初是由 California 大学 Berkeley 分校的两名学生 Spencer Kimball 和 Peter Mattis 开发而成的。Gtk+ 从不完善逐渐发展，到目前为止，成百上千的程序都已经使用 Gtk+ 来编写，这预示着 Gtk+ 在未来某个时候将得到广泛的运用。

Gtk+ 是一个图形用户编程的接口工具。可免费注册，所以用来开发自由软件或商业软件都不需要什么花费。现在很多 Linux 集成系统都已经将 Gtk+1.2 版本打包进去了。包括 RedHat Linux 6.0 以上版本，还有中文版的 Turbo Linux 等等。它也越来越被普遍地应用于 UNIX 系统编程。它为开发 GIMP(GNU Image Manipulation Program, GNU 图形操作程序)而研制的一套图形界面库，在功能上类似于微软的 MFC 库，就是为应用软件提供一套平台无关的、简单易用的图形界面接口。它几乎包括所有基本的图形界面元素，如窗口、按钮、列表、滚动条等等。

GDK 是 Gtk+ 的底层图形库，GDK 将所有与平台相关的函数和系统调用都封装起来，为 Gtk+ 提供了一套平台无关的开发接口。Gtk+ 的所有界面函数都基于 GDK。因此只需为不同系统平台编写相应的 GDK 层，Gtk+ 就能够具备跨平台的移植能力，现在 GDK、Gtk+ 已经支持 X Window、Windows 等多种系统平台。

还有一个组件叫 Glib，它包含了一些标准应用的新扩展用来提高 Gtk+ 的兼容性。用于 Linux 系统的某些函数可能不适合标准的 UNIX 系统，例如 `g_strerror()` 函数等等。某些函数也扩展了 GNUC 的一般功能，例如 `g_malloc` 函数就有自己加强的调试功能。

Gtk+ 可以与多种语言绑定，包括 C++、Guile、Perl、Python、Ton、Ada95、Objective C、Free Pascal 和 Eiffel。用标准 C 开发的程序，编译软件可用 GNU 并附上 Gtk+ 选项即可。想用除了标准 C 以外的其他语言来开发 X window 图形用户程序，则需要先参考一下有关绑定软件的内容(<http://www.gtk.org>)。

本书介绍了 Gtk+ 1.2 版本，内容包括利用 Gtk+ 1.2 编写应用程序的简单介绍、Gtk+ 1.2 控件集的详细讲述，并且为熟悉 Gtk+ 的程序员提供了快速参考。这是一本内容详尽且极具参考价值的专业书籍，特别适合爱好 Linux 的用户学习使用。

目前已经开发出来 Gtk 的增强版 Gtk+。Gtk+ 是将 Gtk、GDK、Glib 集成在一起的开发包，



可以工作在许多类似于 UNIX 的系统上，没有 Gtk 的平台限制。

本书通俗易懂，内容详尽，是学习 Gtk 编程的良好教材。

全书由战晓苏、王宁主持翻译。参加翻译工作的有：贾振华、苗敬利、张乐新、李德勇等。参加翻译和审校工作的还有：李凯、蔡汉魁、王友军、范锦勇、付培国、申晓明、刘学军、屠长庆、王会松、张晓军、扬腊武、周端生、曾月宝、籛树天等。

在本书的翻译过程中，我们深深被 Gtk+ 的卓越性能所吸引，也被本书原作者生动活泼、深入浅出的写作风格所打动。译者对 Gtk+ 引入的新概念，都做了仔细的推敲与斟酌。但由于译者水平有限，内容多、时间紧，国内又很少有 Gtk+ 方面的资料，疏漏和不当之处在所难免，敬请广大读者批评指正。同时，也衷心希望本书能对读者真正有所帮助，这也将给予我们莫大的鼓舞和支持。

译 者

2002 年 5 月

前 言

Gtk+最初是由 California 大学 Berkeley 分校的两名学生 Spencer Kimball 和 Peter Mattis 开发的。他们已经开发了一个名为 GNU Image Manipulation Program(GIMP, www.gimp.org)的图像处理程序,它是一个学院项目的一部分。这个程序最初是用 Motif 编写的,但由于(在那时)Motif 在本质上是开放源代码的,结果在可免费使用的开放源代码的 UNIX 平台上就有一些缺憾,于是在 Motif 的启发下,他们决定开发一个开放源代码的工具包,这样就可以用这个工具包来开发 GIMP。目的倒不是为 X Windows System 开发一个通用的工具包,但事实上 Gtk+已经实现了这一目的。到目前为止,成百上千的程序都已经使用 Gtk+来编写,这预示着 Gtk+在未来某个时候将得到广泛的应用。要想知道更多的有关 Gtk+(和 GIMP)的信息,可以查阅 www.gimp.org/~sjburgess/gimp-history.html。

本书主要内容

本书介绍了 GIMP Toolkit(Gtk+)的 1.2 版本,它的编写出于以下目的:

- 概述使用 Gtk+ 1.2 编程的知识
- 详细描述 Gtk+ 1.2 构件集(widget set)
- 为那些已经熟悉 Gtk+的程序员提供一个对 Gtk+ 1.2 构件集的快速参考

要想知道关于 Gtk+编程的介绍,建议您先阅读本书前面 1~4 章,接着阅读第 10 章——容器类和基类。开始几章(1~3 章)描述了 Gtk+的总体结构,提供了编程所需的信息,并构建了一个简单的 Gtk+应用程序。大部分读者将很想浏览第 4 章——构件,它描述了 GtkWidget。GtkWidget 是个父类,Gtk+类层次结构中的其余构件继承了其大部分功能。容器构件(Container Widget)用来组织其他构件在窗口中(或其他容器中)的布局。第 10 章的前几节描述了容器构件的概念。GtkBox 构件是 Gtk+ 1.2 所能实现的最灵活多变的容器构件,因此也得到了最为普遍的应用。GtkBox、GtkHBox 和 GtkVBox 都在第 10 章进行了介绍。

其余章节详细描述了 Gtk+ 1.2 所能实现的大量构件类。作者使出浑身解数详细描述了本书介绍的 Gtk+构件集所具有的应用级编程界面。书中还包括了作者研究 Gtk+构件集时所编写的大部分代码。在某些情况下,源代码是由一个完整的应用程序组成的(尽管功能有限)。在其他情况下,书中只给出了简短的代码段,用来帮助说明文中的观点。可以在作者的 Web 站点 www.cts.com/crash/s/logan/gtkbook.html 上找到本书全部的源代码示例。

作者已经把 Gtk+构件的参考资料直接放在正文中了(而不是放置在书后的附录中)。参考资料为本书描述的构件所具有的每个应用级函数提供了函数模型,并用一句话描述了每个函数的用途。在参考部分,作者还列举了所有的对象属性,可以在构件上设置和(或)检索这些属性(参阅第 3 章——信号,事件,对象和类型的“对象属性”那一节)。作者也列出了由构件产生的信号。作者为每个信号提供了当激活它们时所调用的应用级信号处理程序的函数模型(查看第 3 章的“信号”一节)。参考资料为初次编程的人员介绍了构件,也可以为已经熟悉部件的程序员提



供一个快速参考。有关参考部分结构的更多信息将在第 4 章得到详细说明(从参考部分中查看 GtkWidget 构件)。

本书重点介绍了 Gtk+构件集。本书没有涉及 Gtk+ Drawing Kit(GDK)或 G Library(Glib), 或有关构件的更详细内容(除了不可避免之处以外)。对于 GDK 和 Glib, 我可以向您推荐一、两本书: Eric Harlow 的 *Developing Linux Application with GTK+ and GDK* 和 Havoc Pennington 的 *GTK+/Gnome Application Development*。您也可以在 www.gtk.org 上找到关于这个主题的参考资料。作者计划在自己的 Web 站点中再增添一个有关 Gtk+构件开发的章节, 这在本书出版后不久就可实现。我希望这个资料能出现在本书的后续版本中。

源代码

作者在本书中加入了源代码, 为所讨论的概念提供了例证。有时, 把行号作为每行源代码的前缀, 例如:

```
014 static void
015 ClickedCallback(GtkWidget *widget, Gtkwidget *dialog_window)
016 {
017     Gtkwidget *window, *labelFrame, *labelTest, *vbox;
018
019     window = gtk_window_new( GTK_WINDOW_TOPLEVEL);
020     gtk_window_position (GTK_WINDOW (window), GTK_WIN_POS_MOUSE);
021     gtk_widget_set_usize( window, 320, - 1);
022     vbox = gtk_vbox_new (FALSE, 0);
023     gtk_container_add (GTK_CONTAINER (window), vbox);
```

对于那些容易阅读或者相对较短的代码(几行左右), 作者常常会忽略行号。在完整的(有时不完整但很长的)例子代码清单前面, 会加上清单号和标题, 如下所示:

清单 1-1 Gtk Label 属性示例代码

```
001 #include <gtk/gtk.h>
002 #include <stdio.h>
003
004 static GtkWidget *leftButton, *rightButton, *centerButton;
005 static GtkWidget *trueButton, *falseButton;
006
007 static void
008 QuitCallback(GtkWidget *widget, GtkWidget *dialog_window)
009 {
010     gtk_main_quit( );
011     exit( 0 );
```

作者计划可以在线使用本书的源代码。要作进一步地了解, 请访问 www.users.cts.com/crash/s/slogan/gtkbook.html。

继续

购买这本书，您一定是决定了要编写 Linux/UNIX 程序，并且还决定使用 Gtk+ 来开发应用程序。除了学习 Gtk+ 之外，您还必须花上一些时间来学习更多的有关桌面环境的知识，用户将在其中执行应用程序。如果您(也可以是多位读者)的目标是 GNOME，则必须使用书本中和 Internet 上的资源来开发 GNOME 环境。本书介绍了 Gtk+ 工具包(所有的 GNOME 应用程序都是用这些工具包开发的)，但在其中作者没有对 GNOME 应用程序开发作特别介绍。

致谢

作者要衷心感谢下面这些人在本书的编写过程中所给予的帮助。首先要感谢的是 Prentice Hall 的 Greg Doench，在此项目还只是雏形的时候，他花费了大量的时间和作者一起讨论，听取作者的提议并促成了该项目。同时还要感谢 Prentice Hall 的 Jim Markham，指导作者做了大量的编辑工作。Army Lepore 对原稿进行了审阅，使作者杂乱无章的稿子变得井然有序，从而引起软件开发人员和英语教师的兴趣。Caroline Roop 安排了整个审稿过程并帮助制作成您现在所看到的书的版式。对于审阅人 Ariel Rios(GNOME 项目)和 Pob Flynn(GAIM 维护者)，作者也同样深表感谢。本书中可能还有的错误全都应该归咎于作者。Joe Hewitt 为本书制作了图片，在此特表感谢。作者也感谢为本书出谋划策的人们，包括 Ramiro Estrugo(以前是 Eazel)，他指出了在正文中存在参考资料的重要性。最后作者要感谢使 Gtk+ 和 GNOME 得以问世的人们。在他们的辛勤努力下，对越来越多的用户来说，Linux 桌面正逐渐成为现实。

原稿的每一行每个字最初都是在 GNU/Linux 操作系统上使用 VIM 编辑器(一种开放源代码的 vi 兼容产品，默认情况下通常可以在 GNU/Linux 系统上使用并可为 Win32、Mac 等其他平台下载它；请查看 www.vim.org)编写的。可以使用 GIMP(www.gimp.org)抓取图像，它是一个可免费使用的图像编辑程序，事实上也是编写的第一个 Gtk+ 应用程序。示例程序也是用 VIM 创建的，使用 gcc 进行编译，并且有时用 gdb 进行调试。gdb 调试器还用来在执行中检查 Gtk+ 的内部情况，可以了解更多的特定构件的操作。

在审阅过程中，本书的超文本标记语言(HTML)版本是用 Gtk+ 的 Bluefish 编辑器(<http://bluefish.openoffice.nl/>)开发的。在上传给审阅者以前，可以使用 Netscape 6 的开发版本显示 Bluefish 的输出结果，这个版本源于开放源代码的 Mozilla 浏览器。所有的这一切当然都是在 GNU/Linux 系统中进行的。

目 录

第 1 章	Gtk+的开发环境	1
1.1	X Window 系统	1
1.2	X 体系结构	2
1.2.1	X 协议	2
1.2.2	X 服务器	3
1.2.3	客户端(Xlib)	3
1.3	工具包	3
1.4	窗口管理器	5
1.5	桌面环境	6
1.5.1	CDE	6
1.5.2	GNOME	7
1.5.3	KDE	7
1.6	小结	7
第 2 章	Hello Gtk+	9
2.1	简介	9
2.2	一个简单的示例: Hello Gtk+	10
2.3	为控制台程序添加交互功能	11
2.4	范型转移	11
2.5	了解 Gtk+版的 Hello World 示例	14
2.6	编译客户程序示例	19
2.7	调试客户程序示例	20
2.8	应用程序启动、终止和主循环函数	22
2.8.1	应用程序启动	22
2.8.2	应用程序终止	27
2.8.3	Gtk+主循环函数	28
2.8.4	超时和空闲处理	32
2.8.5	空闲函数	33
2.8.6	监听按键事件	36
2.9	小结	39
第 3 章	信号、事件、对象和类型	40
3.1	信号	40



3.1.1	示例: GtkButton 信号	40
3.1.2	处理信号	41
3.2	事件	44
3.2.1	事件回调函数原型	44
3.2.2	事件类型	49
3.2.3	信号和事件 API	62
3.3	信号查询	63
3.3.1	发出信号	64
3.3.2	控制信号	69
3.4	对象	73
3.4.1	作为对象的按钮构件	73
3.4.2	对象 API	76
3.4.3	对象属性	79
3.4.4	获取并设置对象属性	83
3.4.5	将客户端数据与对象或构件相关联	87
3.5	类型	91
3.6	小结	92
第 4 章	构件	93
4.1	使用构件的原因	93
4.1.1	简化用户界面开发	93
4.1.2	为用户做简化	95
4.2	GtkWidget	95
4.2.1	类名	95
4.2.2	父类名	95
4.2.3	宏	95
4.2.4	其他宏	96
4.2.5	信号函数原型	99
4.2.6	支持的参数	103
4.2.7	应用程序级 API 概要	103
4.2.8	类描述	110
4.3	小结	144
第 5 章	标签和按钮	145
5.1	控件和容器	145
5.2	GtkLabel	146
5.2.1	类名	146
5.2.2	父类名	146
5.2.3	宏	146

5.2.4 支持的参数	146
5.2.5 应用程序级 API 概要	146
5.2.6 类描述	147
5.3 按钮	152
5.4 GtkButton	154
5.4.1 类名	154
5.4.2 父类名	154
5.4.3 宏	154
5.4.4 信号函数原型	154
5.4.5 支持的参数	155
5.4.6 应用程序级 API 概要	155
5.4.7 类描述	156
5.5 GtkToggleButton	159
5.5.1 类名	159
5.5.2 父类名	159
5.5.3 宏	159
5.5.4 信号函数原型	159
5.5.5 支持的参数	159
5.5.6 应用程序级 API 概要	160
5.5.7 类描述	161
5.6 GtkCheckButton	164
5.6.1 类名	164
5.6.2 父类名	164
5.6.3 宏	164
5.6.4 应用程序级 API 概要	165
5.6.5 类描述	165
5.7 GtkRadioButton	166
5.7.1 类名	166
5.7.2 父类名	166
5.7.3 宏	166
5.7.4 支持的参数	166
5.7.5 应用程序级 API 概要	166
5.7.6 类描述	167
5.8 小结	170
第 6 章 列表	171
6.1 GtkList	172
6.1.1 类名	172



6.1.2	父类名	172
6.1.3	宏	172
6.1.4	信号函数原型	172
6.1.5	应用程序级 API 概要	172
6.1.6	类描述	174
6.2	GtkCList	182
6.2.1	类名	182
6.2.2	父类名	182
6.2.3	宏	182
6.2.4	信号函数原型	182
6.2.5	支持的参数	184
6.2.6	应用程序级 API 概要	185
6.2.7	类描述	192
6.3	小结	230
第 7 章	WINDOWS 和 DIALOGS	231
7.1	GtkWindow	231
7.1.1	类名	231
7.1.2	父类名	232
7.1.3	宏	232
7.1.4	支持的信号	232
7.1.5	信号函数原型	232
7.1.6	支持的参数	232
7.1.7	应用程序级 API 概要	232
7.1.8	类描述	234
7.2	GtkDialog	247
7.2.1	类名	247
7.2.2	父类名	247
7.2.3	宏指令	247
7.2.4	应用程序级 API 概要	247
7.2.5	类描述	247
7.3	GtkFileSelection	251
7.3.1	类名	251
7.3.2	父类名	251
7.3.3	宏指令	251
7.3.4	应用程序级 API 概要	251
7.3.5	类描述	252
7.4	GtkFontSelection	257

7.4.1	类名	257
7.4.2	父类名	257
7.4.3	宏指令	257
7.4.4	应用程序级 API 概要	258
7.4.5	类描述	258
7.5	GtkFontSelectionDialog	266
7.5.1	类名	266
7.5.2	父类名	266
7.5.3	宏指令	266
7.5.4	应用程序级 API 概要	266
7.5.5	类描述	267
7.6	GtkColorSelectionDialog	269
7.6.1	类名	269
7.6.2	父类名	269
7.6.3	宏指令	269
7.6.4	应用程序级 API 概要	269
7.6.5	类描述	269
7.7	GtkColorSelection	273
7.7.1	类名	273
7.7.2	父类名	273
7.7.3	宏	273
7.7.4	支持的信号	273
7.7.5	信号函数原型	274
7.7.6	支持的参数	274
7.7.7	应用程序级 API 概要	274
7.7.8	类描述	275
7.8	小结	277
第 8 章	Separator、Arrow、Image、Pixmap 和 Entry 构件	278
8.1	分隔线	278
8.2	GtkSeparator	279
8.2.1	类名	279
8.2.2	父类名	279
8.2.3	宏	279
8.2.4	应用程序级 API 概要	279
8.2.5	类描述	279
8.3	GtkHSeparator	280
8.3.1	类名	280



8.3.2	父类名	280
8.3.3	宏	280
8.3.4	应用程序级 API 概要	280
8.3.5	类描述	280
8.4	GtkVSeparator	283
8.4.1	类名	283
8.4.2	父类名	283
8.4.3	宏	283
8.4.4	应用程序级 API 概要	283
8.4.5	类描述	284
8.5	GtkArrow	286
8.5.1	类名	286
8.5.2	父类名	286
8.5.3	宏	286
8.5.4	支持的参数	286
8.5.5	应用程序级 API 概要	286
8.5.6	类描述	286
8.6	图像和 Pixmaps	288
8.7	GtkPixmap	289
8.7.1	类名	289
8.7.2	父类名	289
8.7.3	宏	289
8.7.4	应用程序级 API 概要	289
8.7.5	类描述	290
8.8	GtkImage	294
8.8.1	类名	294
8.8.2	父类名	294
8.8.3	宏	294
8.8.4	应用程序级 API 概要	294
8.8.5	类描述	295
8.8.6	从窗口中提取图像数据	300
8.9	GtkEntry	305
8.9.1	类名	305
8.9.2	父类名	305
8.9.3	宏	305
8.9.4	支持的参数	305
8.9.5	应用程序级 API 概要	305
8.9.6	类描述	306

8.10 小结	310
第 9 章 菜单	311
9.1 GtkItemFactory	312
9.1.1 类名	312
9.1.2 父类名	312
9.1.3 宏	312
9.1.4 应用程序级 API 概要	313
9.1.5 类描述	314
9.2 GtkMenuBar	332
9.2.1 类名	332
9.2.2 父类名	332
9.2.3 宏	332
9.2.4 支持的参数	332
9.2.5 应用程序级 API 概要	332
9.2.6 类描述	333
9.3 GtkMenuItem	334
9.3.1 类名	334
9.3.2 父类名	334
9.3.3 宏	334
9.3.4 支持的信号	334
9.3.5 信号函数原型	335
9.3.6 应用程序级 API 概要	335
9.3.7 类描述	336
9.4 复选菜单项和单选菜单项	342
9.5 GtkCheckMenuItem	342
9.5.1 类名	342
9.5.2 父类名	342
9.5.3 宏	342
9.5.4 支持的信号	342
9.5.5 信号函数原型	343
9.5.6 应用程序级 API 概要	343
9.5.7 类描述	343
9.6 GtkRadioMenuItem	346
9.6.1 类名	346
9.6.2 父类名	346
9.6.3 宏	346
9.6.4 应用程序级 API 概要	347



9.6.5 类描述	347
9.7 GtkTearoffMenuItem	350
9.7.1 类名	350
9.7.2 父类名	350
9.7.3 宏	350
9.7.4 应用程序级 API 概要	350
9.7.5 类描述	350
9.8 GtkMenu	352
9.8.1 类名	352
9.8.2 父类名	352
9.8.3 宏	352
9.8.4 应用程序级 API 概要	352
9.8.5 类描述	353
9.9 GtkOptionMenu	359
9.9.1 类名	359
9.9.2 父类名	359
9.9.3 宏	360
9.9.4 应用程序级 API 概要	360
9.9.5 类描述	360
9.10 小结	365
第 10 章 容器类和基类	366
10.1 GtkContainer	367
10.1.1 类名	367
10.1.2 父类名	367
10.1.3 宏	367
10.1.4 支持的信号	367
10.1.5 信号函数原型	367
10.1.6 支持的参数	368
10.1.7 应用程序级 API 概要	368
10.1.8 类描述	369
10.2 GtkFixed	373
10.2.1 类名	373
10.2.2 父类名	373
10.2.3 宏	373
10.2.4 应用程序级 API 概要	373
10.2.5 类描述	373
10.3 GtkBox	377

10.3.1	类名	377
10.3.2	父类名	377
10.3.3	宏	377
10.3.4	支持的参数	377
10.3.5	应用程序级 API 概要	378
10.3.6	类描述	379
10.4	GtkVBox	386
10.4.1	类名	386
10.4.2	父类名	386
10.4.3	宏	387
10.4.4	应用程序级 API 概要	387
10.4.5	类描述	387
10.5	GtkHBox	387
10.5.1	类名	387
10.5.2	父类名	387
10.5.3	宏	388
10.5.4	应用程序级 API 概要	388
10.5.5	类描述	388
10.6	GtkButtonBox	388
10.6.1	类名	388
10.6.2	父类名	388
10.6.3	宏	389
10.6.4	应用程序级 API 概要	389
10.6.5	类描述	390
10.7	GtkVButtonBox	394
10.7.1	类名	394
10.7.2	父类名	395
10.7.3	宏	395
10.7.4	应用程序级 API 概要	395
10.7.5	类描述	395
10.8	GtkHButtonBox	396
10.8.1	类名	396
10.8.2	父类名	396
10.8.3	宏	397
10.8.4	应用程序级 API 概要	397
10.8.5	类描述	397
10.9	GtkNotebook	398
10.9.1	类名	398