<https://blog.csdn.net/lvbian/article/details/23031635>

一个与linux下使用Xlib进行窗口编程的小例子,作为一个刚开始了解Xlib的初学者,应该从一个小例子开始,从编译,运行,再到逐步深入...,根据X11协议原理,其实我们所编写的程序都是Xclient程序,Xserver负责接收硬件事件,Xclient负责计算,结果会反馈给Xserver,Xserver来负责与硬件沟通,例如显示Xclient绘制的图形到显示器.

具体编译:gcc x11.c -o output -I/usr/X11R6/include -L/usr/X11R6/lib -lX11

源码如下:

**[cpp]** [view plain](https://blog.csdn.net/lvbian/article/details/23031635) [copy](https://blog.csdn.net/lvbian/article/details/23031635)

1. /\*
2. \* 下边是编译命令
3. \* gcc x11.c -o output -I/usr/X11R6/include -L/usr/X11R6/lib -lX11
4. \*/
6. #include <X11/Xlib.h>
7. #include <stdio.h>
8. #include <stdlib.h>
9. #include <string.h>
11. **int** main(**void**)
12. {
13. Display \*display;
14. Window window;
15. XEvent event;
16. **char** \*msg = "你好 ,这是一个测试窗口!";
17. **int** s;
19. /\* 与Xserver建立连接 \*/
20. display = XOpenDisplay(NULL);
21. **if** (display == NULL)
22. {
23. fprintf(stderr, "Cannot open display\n");
24. exit(1);
25. }

28. s = DefaultScreen(display);
30. /\* 创建一个窗口 \*/
31. window = XCreateSimpleWindow(display, RootWindow(display, s), 10, 10, 200, 200, 1,
32. BlackPixel(display, s), WhitePixel(display, s));
34. /\* 选择一种感兴趣的事件进行监听 \*/
35. XSelectInput(display, window, ExposureMask | KeyPressMask);
37. /\* 显示窗口 \*/
38. XMapWindow(display, window);
40. /\* 事件遍历 \*/
41. **for** (;;)
42. {
43. XNextEvent(display, &event);
45. /\* 绘制窗口或者重新绘制 \*/
46. **if** (event.type == Expose)
47. {
48. XFillRectangle(display, window, DefaultGC(display, s), 20, 20, 10, 10);
49. XDrawString(display, window, DefaultGC(display, s), 50, 50, msg, strlen(msg));
50. }
51. /\* 当检测到键盘按键,退出消息循环 \*/
52. **if** (event.type == KeyPress)
53. **break**;
54. }
56. /\* 关闭与Xserver服务器的连接 \*/
57. XCloseDisplay(display);
59. **return** 0;
60. }