多个X服务器（我们称它们为工作站）的连接，即使它们可能与X程序在同一台机器上。在消息丢失与重复由网络软件来处理的意义上，X认为这一连接是可靠的，并且它不用担心通信错误。通常在服务器与客户之间使用的是TCP/IP。

四种类型的消息通过连接传递：

1) 从程序到工作站的绘图命令。

2) 工作站对程序请求的应答。

3) 键盘、鼠标以及其他事件的通告。

4) 错误消息。

从程序到工作站的大多数绘图命令是作为单向消息发送的，不期望应答。这样设计的原因是当客户与服务器进程在不同的机器上时，命令到达服务器并且执行要花费相当长的时间周期。在这一时间内阻塞应用程序将不必要地降低其执行速度。另一方面，当程序需要来自工作站的信息时，它只好等待直到应答返回。

与Windows类似，X是高度事件驱动的。事件从工作站流向程序，通常是为响应人的某些行动，例如键盘敲击、鼠标移动或者一个窗口被显现。每个事件消息32个字节，第一个字节给出事件类型，下面的31个字节提供附加的信息。存在许多种类的事件，但是发送给一个程序的只有那些它宣称愿意处理的事件。例如，如果一个程序不想得知键释放的消息，那么键释放的任何事件都不会发送给它。与在Windows中一样，事件是排成队列的，程序从队列中读取事件。然而，与Windows不同的是，操作系统绝对不会主动调用在应用程序之内的过程，它甚至不知道哪个过程处理哪个事件。

X中的一个关键概念是资源（resource）。资源是一个保存一定信息的数据结构。应用程序在工作站上创建资源。在工作站上，资源可以在多个进程之间共享。资源的存活期往往很短，并且当工作站重新启动后资源不会继续存在。典型的资源包括窗口、字体、颜色映射（调色板）、像素映射（位图）、光标以及图形上下文。图形上下文用于将属性与窗口关联起来，在概念上与Windows的设备上下文相类似。

X程序的一个粗略的、不完全的框架如图5-38所示。它以包含某些必需的头文件开始，之后声明某些变量。然后，它与X服务器连接，X服务器是作为XOpenDisplay的参数设定的。接着，它分配一个窗

```
#include <X11/Xlib.h>
#include <X11/Xutil.h>

main(int argc, char *argv[])
{
    Display disp;                          /* 服务器标识符 */
    Window win;                            /* 窗口标识符 */
    GC gc;                                 /* 图形上下文标识符 */
    XEvent event;                          /* 用于存储一个事件 */
    int running = 1;

    disp = XOpenDisplay("display_name");   /* 连接到X服务器 */
    win = XCreateSimpleWindow(disp, ...);  /* 为新窗口分配内存 */
    XSetStandardProperties(disp, ...);     /* 向窗口管理器宣布窗口 */
    gc = XCreateGC(disp, win, 0, 0);       /* 创建图形上下文 */
    XSelectInput(disp, win, ButtonPressMask | KeyPressMask | ExposureMask);
    XMapRaised(disp, win);                 /* 显示窗口；发送Expose事件 */

    while (running) {
        XNextEvent(disp, &event);          /* 获得下一个事件 */
        switch (event.type) {
            case Expose:      ...; break;  /* 重绘窗口 */
            case ButtonPress: ...; break;  /* 处理鼠标点击 */
            case Keypress:    ...; break;  /* 处理键盘输入 */
        }
    }

    XFreeGC(disp, gc);                     /* 释放图形上下文 */
    XDestroyWindow(disp, win);             /* 回收窗口的内存空间 */
    XCloseDisplay(disp);                   /* 拆卸网络连接 */
}
```

图5-38 X窗口应用程序的框架