



北京希望电脑公司 X Window System 系列丛书 之七

X Window 编程指南

(11.4版)

(共九册)

海洋出版社

X Window 编程指南

(11.4 版)

叶欣
叶连恩 编
陈建伟

海洋出版社

一九九一年五月·北京

内 容 提 要

X Window System(X 窗口系统)是 80 年代末推出的可移植标准,是软件开发和运行的必备支持系统。

本书是《X Window System 系列丛书》的第七册,它与第九册配套使用。全书包括十五章和四个附录,主要介绍如何利用 X Window 软件包中提供的各种工具来设计用户的应用程序,并提供一个绘图应用程序的实际例子。

欲购本书的用户可直接与北京 8721 信箱联系,电话 2562329 邮政编码 100080。

X Window System 系列丛书之七

X Window 编程指南

(11.4 版)

叶欣 叶连恩 陈建伟 编

审校:刘莉蕾

责任编辑:阎世尊

海洋出版社出版发行(北京市复兴门外大街1号)

双青印刷厂印刷

开本:787×1092 1/16 印张:22.9 字数 55 6686

1991 年 5 月第一版 1991 年 5 月第一次印刷

印数:1—3000 册

ISBN 7-5027-1909-1/TP·25

定价:16.00

序

近几年来，国际计算机界出现了一股“窗口”热，从微型机到小型机到大型机，从工作站到网络，窗口系统已成为软件开发和运行的必备支持系统。

由美国 Massachusetts 大学、IBM 公司、DEC 公司、AT&T 公司、Microsoft 公司等十几家团体联合开发的 X 窗口系统(X Window System)更成为窗口中热点，自从它在八十年代末推出以来，得到了国际软件界的高度评价和肯定。

X Window System 运行于工作站上，以其极其方便的用户接口、强大的编程功能和高质量的代码，改变了整个工作站的世界；X Window System 运行于 386、486 等微机上，更能充分发挥这些微机的特点，利用它们开发更好更完美的软件。

X Window System 推出以后，经过不断的更新和完善，已达到 11.4 版，本版系统在原来的基础上 性能方面又有了很大的提高。

X Window System 主要有核心系统(协议)和建立在其上的应用程序环境工具(Xt、XView、Xlib 等)组成，为了使国内广大计算机用户了解并掌握 X Window System 的基础知识和高级程序设计技术，我们在近几年使用它们的基础上，给合国外最新软件和资料，特编辑了 X Window System 系列丛书，包括：

- 第一册：《X 协议参考手册》
- 第二册：《Xlib 编程手册》
- 第三册：《Xlib 参考手册》
- 第四册：《X 教程》
- 第五册：《X Toolkit Intrinsics 编程手册》
- 第六册：《X Toolkit Intrinsics 参考手册》
- 第七册：《X Window 编程指南》
- 第八册：《X View 编程手册》
- 第九册：《X Window 高级编程指南》

《X Window 编程指南》和《X Window 高级编程指南》配套使用，主要介绍如何利用 X Window 软件包中提供的各种工具来设计用户的应用程序，其中第七册详细介绍了一个绘图应用程序的实际例子，第九册介绍高级程序设计的方法和技巧。

本丛书编辑过程中，得到了国内许多专家、学者的帮助和支持，特别是石清教授，在百忙之中审阅了全书，提出了许多修改意见，刘京同志、薛梅同志、王丽同志、张阳同志为本丛书的完成作出了许多努力，编者在此向他们表示感谢。

由于 X Window System 资料非常新颖，国内又没有可供借鉴的中文参考资料，所以本丛书在编辑过程中，难免会存在一些不足之处，希望各位同仁批评指正。

本丛书出版过程中，得到了中国科学院希望高级电脑技术公司资料部秦人华经理、杨淑欣老师的大力帮助和支持，编者在此表示衷心的感谢。

编者

一九九一年五月

目 录

绪论	1
0.1 X的发展历史	1
0.2 X为什么是这样?	2
0.2.1 柔性窗口系统	2
0.2.2 资源共享	3
0.2.3 产品系列一致性	3
0.3 X是什么?	3
0.3.1 客户(Client)/服务器(Serve)轴心	3
0.4 X的组成	5
0.4.1 Xlib	5
0.4.2 X工具箱	5
0.4.3 X网络协议	6
0.5 X运行环境	6
0.6 编排介绍	6
第一部分 X编程教程	7
第一章 建立第一个X程序	8
1.1 示例1: 一个最小的X程序	9
1.1.i 示例1源代码	10
1.2 示例2: 打开一个最小窗口	14
1.2.1 创建一个窗口	14
1.2.2 向窗口管理程序发送“提示(Hints)”	17
1.2.3 刷新显示器	18
1.2.4 释放窗口	19
1.2.5 示例2源代码	19
1.3 示例3: 对openWindow函数添加新的功能	23
1.3.1 图符(icon)	23
1.3.2 示例3源代码	25
1.3.3 图符位图(Icon Bitmap)	28
1.4 小结	29
第二章 用X绘图	30
2.1 画直线和矩形	30
2.2 图形关联	30
2.2.1 窗口创建图形关联	31
2.3 示例1: 画直线和矩形	31
2.3.1 示例1源代码	32

2.4	利用圆弧画椭圆形	41
2.5	填充椭圆形和矩形	42
2.6	示例 2: 画出并填充椭圆形和矩形	42
2.6.1	示例 2 源代码	42
2.7	画多条直线, 多个矩形和圆弧	46
2.8	小结	49
第三章 颜色的使用		50
3.1	颜色应用软件的可移植性和其用法	50
3.2	X 的彩色映象(Colormap)	51
3.3	示例 1. 用红色画一条直线	52
3.3.1	示例 1 源代码	53
3.4	示例 2: 使用更多的颜色	58
3.4.1	示例 2 源代码	59
3.5	小结	64
第四章 正文的使用		65
4.1	字体	65
4.1.1	加载字体	65
4.1.2	设置图形关联	66
4.1.3	画正文	66
4.1.4	所画正文的位置	68
4.1.5	指出字体大小	68
4.1.6	释放字体	69
4.1.7	使用字体的技巧	70
4.1.8	找到可用的字体	70
4.1.9	有限的 X 服务器资源	72
4.2	示例程序	72
4.2.1	示例程序源代码	72
4.3	小结	76
第五章 事件		77
5.1	事件驱动的编程	77
5.2	X 事件模型	77
5.3	请求事件	78
5.4	事件屏蔽	78
5.5	从 X 服务器接收事件	79
5.6	查询事件	80
5.7	事件类型	81
5.8	鼠标按钮事件	82
5.9	键盘事件	84
5.10	进入/离开事件	85
5.11	暴露事件(Exposure Event)	87

5.12	示例 1: 检查事件的程序	88
5.12.1	示例 1 源代码	88
5.13	示例 2: 基于鼠标的绘图程序	94
5.13.1	示例 2 源代码	95
5.14	用 XEV 程序进一步了解 X 事件模型	100
5.15	小结	100
第六章 键盘事件		101
6.1	元按键(meta key)	104
6.2	鼠标按钮	105
6.3	示例程序	106
6.3.1	示例程序源代码	106
6.4	键盘匹配的改变	112
6.5	小结	113
第七章 拖动画线		114
7.1	光栅操作函数	114
7.2	GC 图形函数总结	121
7.3	示例 1: 拖动画线	121
7.3.1	示例 1 源代码	123
7.4	示例 2: 拖动画椭圆形	129
7.4.1	示例 2 源代码	130
7.5	示例 3: 光栅操作函数的演示	134
7.5.1	示例 3 源代码	138
7.6	小结	145
第八章 X 的标准以及与窗口管理程序的交互作用		146
8.1	X 命令行参数	146
8.2	显示器名	146
8.3	X 几何	147
8.4	为窗口创建光标	148
8.4.1	光标为正文	149
8.4.2	创建自己的光标	152
8.4.3	光标资源的释放	153
8.5	示例 1: 接收命令行参数并设置光标	153
8.5.1	示例 1: 源代码	156
8.6	示例 2: 向窗口管理程序发送更多的信息	169
8.6.1	分类特征	170
8.6.2	窗口大小	171
8.6.3	图符名	172
8.6.4	示例 2 源代码	172
8.7	小结	177

第二部分 建立一个 X 窗口系统应用软件	178
第九章 Draw 应用软件简介	179
9.1 浮动选择调色板	179
9.1.1 点	180
9.1.2 正文	180
9.1.3 加载新文件	180
9.1.4 消除绘图操作	181
9.2 命令行参数	181
9.3 特权用户的元按键简缩	182
9.4 严格的错误处理	182
9.5 处理致命的 IO 错误	184
9.6 小结	186
第十章 多窗口应用程序	187
10.1 浮动调色板	187
10.1.1 加亮调色板选择项	194
10.1.2 刷新调色板窗口	195
10.1.3 调色板光标	198
10.2 画图窗口	202
10.3 清除 X 窗口和 GC 资源	203
10.4 Draw 应用程序中的文件格式	204
10.5 象图和窗口	206
10.5.1 XCopyPlane	206
10.5.2 用象图进行撤销工作	207
10.6 小结	208
第十一章 draw 应用程序中的事件处理	210
11.1 事件屏蔽和事件选择	211
11.2 画图事件	212
11.3 调色板选择事件	219
11.4 正文入口事件	222
11.5 键盘简缩输入事件	225
11.6 X 服务器产生的事件	228
11.7 小结	230
第十二章 弹出式窗口和对话框	232
12.1 在 X 中创建弹出式窗口	232
12.2 从调色板调出弹出式窗口	234
12.2.1 查找鼠标指示器位置	235
12.2.2 弹出式文件名对话框	237
12.2.3 对话框的实现	238
12.2.4 建立对话窗口	241
12.2.5 显示对话窗口的内容	242

12.2.6	对话事件	243
12.2.7	消除	246
12.3	关于瞬间窗口的一些提示	246
12.4	小结	247
第十三章	Draw 应用程序的源代码	249
第十四章	Draw 应用程序的改进	319
14.1	暴露事件的处理	319
14.2	放大	319
14.3	滚动	320
14.4	光标的编辑	320
14.5	区域的截取与粘接/移动与拷贝	320
14.6	获取	320
14.7	倒置、反置或旋转	321
14.8	小结	321
第三部分	X 的工具箱	322
第十五章	X 的工具箱简介	323
15.1	Xt 工具箱	324
15.2	Xt 示例程序	324
15.2.1	初始化 Xt 库	324
15.2.2	创建子目标基	325
15.2.3	目标基的管理	326
15.2.4	目标基的实现	326
15.2.5	目标基事件循环	327
15.2.6	目标基示例程序源代码	327
15.2.7	编译 Xt 程序	329
15.3	X Toolkit 的进一步说明	329
15.4	小结	329
附录 A	X 的颜色库	330
附录 B	X 事件类型和结构	332
附录 C	图形关联	344
附录 D	X 客户程序的几个示例	346

绪 论

这是一本关于如何学会 X 窗口系统的书。在我们需要学会怎样编 X 窗口系统的应用软件的时候,这种类型的书很难发现,但却是必须的资料,展示如何简易和愉快的开始 X 的编程。这本书和其它我们先前遇到的 X 窗口系统的运行环境资料不同,它是通向 X 窗口系统的桥梁。在大部分有关 X 窗口系统的书籍中,仅是提供了 X 程序库或称为 Xlib 的资料。那些书籍尽管对高级 X 编程很有使用价值,但它们没能提供 X 的恰当介绍,也没能提供 X 编程的实际例子。

在绪论中将概要说明 X 的一些基本概念和结构,以及 X 窗口系统的发展历史。如果在阅读了绪论后没能立即进入 X 编程也没关系,因为它只是使读者大体了解 X 的术语和环境。绪论中提到的任何内容将在以后各章节中作详细说明。

本书并不包括 X 的全部内容。事实上,它的目的在于帮助读者在 X 环境下编程应用软件。本书的特点是把 X 函数联合在一起调用。如:如何画拖动直线(在第七章讲述)。总之,想要通过简单地叙述而建立一个详细庞大的应用软件是比较困难的。

X 窗口系统是一个庞大而复杂的系统。它的复杂性主要在于试图通过虚拟来处理任何类型的计算机图形显示。它还试图提供一个完整的图形系统——用来创建窗口接口或处理精密图形页设计以及计算机辅助设计软件包。

本书集中显示如何使一些工作得以实现并切实加以实现。对于 X 这样一个庞大而又复杂的系统来说,只用一本书就想完全地把它介绍清楚是不可能的。本书不是一本纯介绍性的书,而是一本描述 X 的主要特性以至教会编写 X 窗口系统应用软件的手册,在有关章节里介绍的主要特性将在每章的结尾处给以总结。这样在需要应用时,就可以反过来阅读。

X 窗口系统为解决一些看上去难以对付的问题提供了极大的保证,例如如何为在具有不同显示屏的多种操作系统下运行的各种计算机提供一个公共接口。X 可提供的窗口系统可以在从 IBM PC 到大型主机或巨型机上运行。Cray 巨型机可以在 DEC 或 Apollo 工作站的显示屏上显示其输出。从苹果机到 Xerox,从 Hewlett-Packard 到 Sun Microsystems,几乎所有的主要计算机厂家都接受 X 窗口系统。

本书还可扩展到窗业和科学技术领域的 X 初学者,在具有 C 语言编程知识的基础上可以试着进行下列工作:

- 确定接受 X 窗口系统的可行性。
- 移植实用软件在 X 下运行。
- 为各厂家提供应用软件的相同外观和风格。
- 掌握最新的网络图形窗口系统。
- 在 X 下进行应用软件开发。

0.1 X 的发展历史

要想概念性的理解 X 窗口系统,了解一下 X 的发展历史和 X 窗口系统设计的初始目的是很有帮助的。

1984年，麻省理工学院(MIT)在商业和科技领域的科研人员在计算机应用方面面临一个共同的问题:他们有许多不兼容的工作站。解决的办法是建立可用于辅助教学的图形工作的网络。面对着各种操作系统和不同厂家的硬件，MIT官员决定组织 Athena 工程，成立同 DEC 和 IBM 合作的 MIT 开发组。

Athena 工程的方案是设计一个网络。在调用远程资源的同时，可以运行局部的应用软件。这一网络以斯坦福大学的 W 软件环境为基础，通过图形网络环境把 IBM、DEC 及其它孤立的工作站联在一起，设计者创建了第一个独立于硬件和生产厂家的操作环境，即 X 窗口系统。

在设计 X 时，开发组确立了以下思想：

- 除非出于实现实际应用软件的需要，否则不增加新的函数。
- 确定系统不包括什么同包括什么同样重要。不必使系统无所不能，但需要有扩充性，以便以后需要时可附加进去。
- 从各种事例中总结。
- 在不能完全理解问题以前，最先不提任何解决方案。
- 使用最简单的方案。
- 尽量地分解问题的复杂性。
- 提供机制而不是模式。尤其是，让客户自己决定用户接口模式。

注意，与本书直接相关的是以上中的最后一条，这将在以后进行讨论。

X 窗口系统是成功的。到 1986 年，X 的消息已传遍世界，人们纷纷要求提供 X 窗口系统。因此，X10.4 发行了，它与早期实现的 UNIX 几乎同样流行。由于 X11 版本与 X10 具有明显的不兼容性，MIT 停止了 X10 版本研究，而在 1987 年 9 月非完全的第一次发行了 X11 版，并在 1988 年 3 月，正式第二次发行 X11。现在，X 窗口系统的进一步开发是由 X 财团(于 1988 年 1 月成立)监督下进行。

由于 X 财团的成员很快推出了一些在 X 基础上开发的产品，X 窗口系统在计算机新闻中已显露头角。此外，开放软件基金会(OSF)在 1989 年 1 月接受 DEC、HP 和 Microsoft 的建议把 X 作为它的图形用户接口——为使 UNIX 更简易地使用而设计的图形接口——的主要成员，X 窗口系统得到又一次发展。

0.2 X 为什么是这样？

从很大程度上来说，X 窗口是一个应时而生的窗口系统。过去，小型机及其以下水平的硬件不能在其可接受的价格内提供如何强功能的图形网络系统。随着高性能的工作站和微型的升起，X 窗口系统因此成为最可信的产品。此外，还有其它一些原因使 X 成为理想的开发工具。

0.2.1 柔性窗口系统

当用户越来越习惯窗口式界面时，X 窗口系统提供了灵巧和柔性的用户界面。由于 Apple Macintosh 和 Microsoft windows 在微型机方面的出色成就，以及 Sun Microsystem 的 News 和 DEC 公司的 DECWindows 的成功都证明了窗口界面大受欢迎。过去几年的研究表明窗口界面比文字基础的界面更易于学习和使用。但 X 不仅仅是一个窗口系统，它比先前提到的系统有更好的柔性和更大的潜力。

在很大程度上, 这些系统仅是一个单一的硬件环境和单一的产品。在把各种类型的计算机联在一起时, 其它的系统使软件设计与开发人员不得不考虑不同厂家的计算机对他们的妨碍。而 X 窗口则不是这样。X 是一个与操作系统无关的且网络明晰的系统。通过把窗口管理系统和窗口服务器分开, 这样就可以在不需昂贵的仿真卡和外来的网络系统情况下, 把完全不同类型的硬件联在一起。由于用户界面仅用来产生 X 调用, 所以也就于操作系统无关了。

另外, 由于系统设计者的最根本信条是提供机制而不是模式, 软件设计者——一定程度上说是用户——才可以对界面的具体风格拥有最后的发言权。通过编程, 应用软件本身定义其窗口界面和应用软件的“风格”。而且, 还可以建立自己想要的界面类型: 可以象 Macintosh 的图标和窗口; 也可以象 News。至于 X 本身的界面相对来说并不很重要。X 提供机制, 可以在其上建立各种风格的用户界面。

这在讨论 X 窗口系统和 UNIX 操作系统时显得特别重要。过去, UNIX 曾受到谴责——由于其不友好的但又不可缺少的用户界面。把 X 窗口系统作为界面, UNIX 的应用软件变得容易使用。事实上, 这一观点可以移植到其它也被认为是不友好的操作系统上, 例如 MS-DOS 和 VMS。

由于 X 与操作系统无关, 因此有利于软件移植。标准的 X C 语言库, 称为 Xlib, 对于每个运行 X 的机器都相同, 这意味着可以把界面代码直接从一个机器移植到另一个机器上, 因为所有 X 调用都是相同的。而用户界面一般占代码的 30% 到 60%, 因此应用软件具有很大的可移植性。

0.2.2 资源共享

X 窗口系统允许同一时间的多个程序共享设备诸如鼠标、键盘和图形显示器等。事实上只要连上 X, 整个工作站就是一个显示器——由键盘、指示设备(通常为鼠标), 一个或几个监视屏幕。通过键盘和鼠标的连接, 各个屏幕能同时一起工作。

0.2.3 产品系列一致性

正如先前提到的, X 窗口系统也被几个不同的操作系统所使用, 包括 UNIX、MS-DOS、A/UX 和 VMS。X 窗口系统是一个标准的应用软件运行环境。为一台机器所写的应用软件不需经过重写就可在另一台机器上使用。

X 窗口系统作为一图形窗口环境已被几家大型计算机公司认可, 从而形成了 X 财团。X 财团包括 Apollo、Apple、AT&T、Bull、DEC、Fujitsu、HP、IBM、NCR、Sony、Sun Microsystems、王安和 Xerox; 其联系会员有 Adole Systems, 卡耐基——梅隆大学和软件产业集团。此外, 主要硬件制造厂已把 X 列为可达窗口系统。

0.3 X 是什么?

现在已经知道了 X 窗口系统是一个令人难以至信的操作环境, 下面我们将给出 X 窗口系统的概念核心和思路。

0.3.1 客户(Client)/服务器(Serve)轴心

X 窗口系统结构是以简单的客户/服务器关系为基础的, 其中客户是执行任务的应用

软件程序，而显示服务器是控制和显示到显示监视器上的所有输出、跟踪客户输入和更新窗口的程序。由于 X 是一个网络环境，客户和服务器的没有必要控制在同一系统中(虽然在某些情况下它们允许也能够这样)，相反，X 窗口系统允许“分布式处理”。例如，在 Apollo 工作站可以运行 X 服务器并调用网络内 Cray 巨型机上的处理功能，而在 Apollo 监视器上显示 Cray 的计算结果。

你会首次发现 X 的术语与传统计算机科学术语有些细微的不同，在微机和小型机网络中，服务器是一种硬件设备，它在网络中心运行，分配数据和处理网络内的工作站和终端。而 X 中却不是完全这样。从我们的目的出发，X 的服务器是控制显示的局部软件程序。因为网络内其它系统可以使用显示器，X 服务器与局部网络(LAN)的文件服务器不同。

在 X 中，显示设备是键盘、指示设备(通常是鼠标)和一个或多个屏幕，它通常与工作站有关，如图 0.1 所示，显示服务器跟踪多个输入，允许用户运行几个不同的客户(如数据库管理软件，字处理器和图形应用软件)。一个显示器可运行通过键盘和鼠标连结的多个显示屏幕。但只要单一用户限定在单一工作站，多显示屏幕就授权到单一显示器。

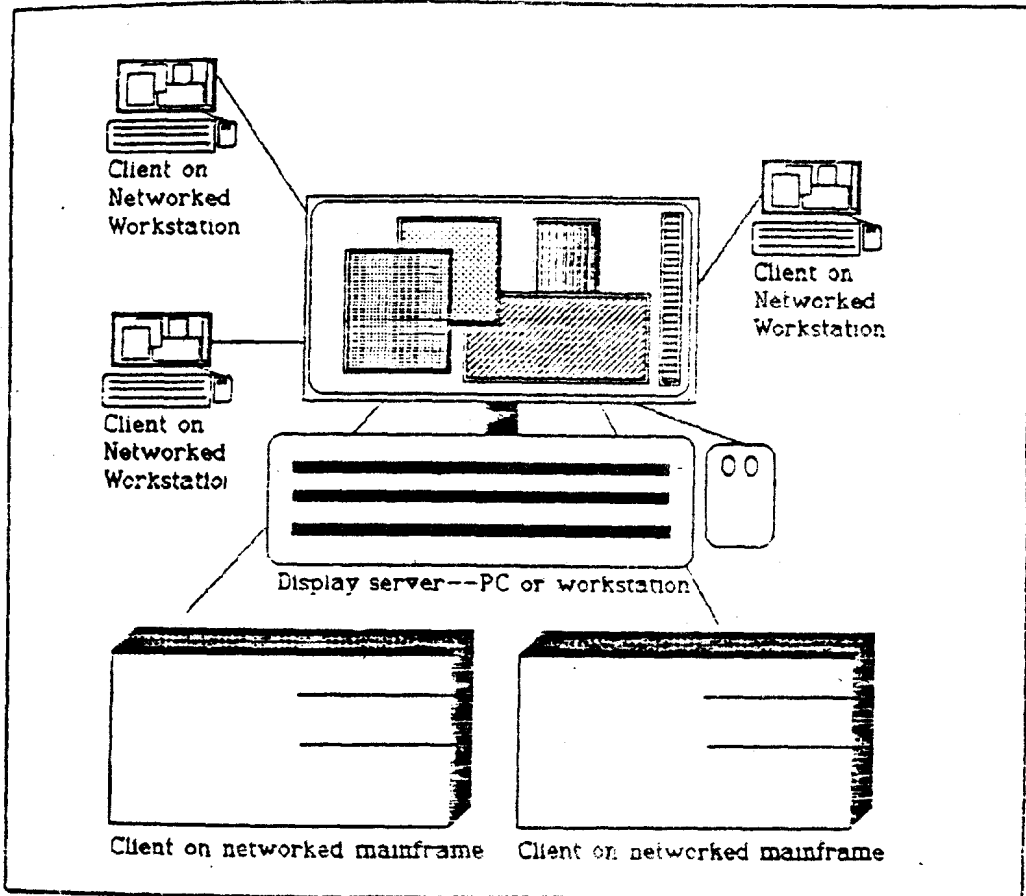


图 0.1 网络环境中的显示服务器

服务器就象是远程系统和局部系统上运行的程序之间(称为客户或应用程序)的交通警察。服务器的工作如下：

- 允许其它客户使用显示设备。
- 传送网络信息。
- 截取从其它客户来的网络信息。
- 执行二维图绘制、把客户从繁重图形处理的工作中解脱出来。
- 跟踪客户间共享的资源(如窗口、光标、字体和图形环境)。
- 允许上面提到过的分布式处理。
- 在 X 应用于多任务操作系统时, 允许多任务处理(例如在应用于 UNIX 时, X 允许调用 UNIX 的多任务处理功能)。

然而, 服务器最重要的功能是跟踪显示器输入并通知客户。在 X 中, 这种输入称作事件(event)。当按下下一个键时, 这就是一个事件; 而当松开该键时, 这就是另一个事件。类似地, 当用鼠标操纵光标时, 这又是另一个事件。这些事件通过事件队列传递到应用软件上。事件将在第五和第六章作更详细的讨论。

上面已经提到过, 客户是可以同时运行的应用程序。术语“应用软件”和“客户”在本书中是通用的。一个客户可同时使用几个服务器, 这取决于网络的结构。客户有几种类型, 本绪论中只介绍两种, 其它的可参见附录 D。

与其它许多系统如 Macintosh OS 不同, 窗口管理程序是控制显示器上窗口大小与位置的客户。所有 X 窗口系统的安装都要有 UWM, 它代表通用窗口管理程序; UWM 允许改变窗口大小, 安排窗口命令, 创建更多窗口和执行其它功能。在第八章中, 将对窗口管理程序作更深一步的解释。

由于 X 窗口系统生成位映象图形, 因此在不添加工作站或个人计算机的情况下, 可以在多用户系统上安装终端仿真器。Xterm 仿真 DEC VT102 或 Tektronix4041 终端, 使用“哑”终端可以运行多个应用软件, 这要用为 ASCII 终端所写的软件的巨型数组提供兼容性。

0.4 X 的组成

在很多方面, X 是其各部分的综合。首先把 X 分成几部分, 随处可看到这些部分是怎样一起工作的。

MIT 提供的通用 X 窗口系统包括 Xlib 图形子程序库, X 网络协议、X 工具箱和一些窗口管理程序。应用软件编程人员通过图形和窗口函数库 Xlib 把客户程序连接起来。

0.4.1 Xlib

Xlib 包含约 300 个与 X 协议请求匹配或提供使用的函数。Xlib 事实上是把 C 语言函数调用转换成实现指定功能的 X 协议请求, 如 XDrawline 是画一条直线。这些函数包括窗口的创建、删除、移动和控制大小; 画成和多边形(将在第二章介绍); 设置背景方式; 跟踪鼠标等。Xlib 还允许用不同方式使用窗口, 包括覆盖窗口和同时向多个窗口输出。它支持多种字体、图形方式, 画线以及彩色和单色应用软件。

0.4.2 X 工具箱

X 工具箱是为编程方便而提供的程序子例程库。各生产厂家诸如 HP、AT&T/Sun、IBM/Athena、DEC 和 Sony, 所提供的工具箱版本固定, 极其相似, 但稍有不同。多种工

工具箱包括滚动窗口、按键、弹出式菜单、窗口边界和对话框等。在第十五章将作详细叙述。

0.4.3 X 网络协议

X 网络协议为在客户和服务器间传送请求定义数据结构。从技术上讲，X 网络协议是一种基于流的异步通信而不是基于过程调用或核心调用接口。应用软件不做此工作，协议是 Xlib 的函数。这种结构加快了信息交流。

MIT X 财团给出了协议说明，它定义在 X 窗口系统协议 1.1 版中。如果已经在 UNIX 系统下联机安装上该文档，则可以在 `doc/Protocol/spec` 目录下找到。

现在，X 网络协议只在 decNET 和 TCP/IP 中实现。

0.5 X 运行环境

事实上任何级别的计算机——微机、小型机、大型机——都可运行 X。市场上很少有机器还没配有 X。

独立于厂家是该系统一个极大的优点。如果不能在 Sun 工作站或 DEC VAX 上使用 X，还可以找到在 MS-DOS 下运行的 X 窗口系统版本。一般在微机上，至少要在配有图形卡的 AT 机上才能很好运用 X。另外，Pine Systems 已出版了 Macintosh 操作系统下实现的 X。

0.6 编排介绍

本书共分十五章来介绍 X 窗口系统程序的主要概念。由于学习怎样编程的最好方法是进行实际编写，因此每章都包括实际代码例子。

在第一部分中，每一章都描述了包括该章概念内容的独立程序。第二部分中，每一章都建立一个 X 窗口系统的大型应用软件。第三部分介绍了 X 工具箱系列。由于工具箱是固定版本的，因此，第三部分只介绍基本的工具箱概念和主要应用，其它留给你自己。另外还给出色彩、事件类型、图形系统、X 客户例子与附录。

特别提醒，使用本书，必须具备 C 语言编程的知识。

X 编程教程

第一部分 X 编程教程

本部分从解释 X 窗口系统开始，然后介绍建立 X 窗口应用软件所需要的基础知识，我们将学习：

- 如何建立同服务器的联系，随后是一个介绍 X 理论和过程的实例。
- 如何用 X 绘图。X 是一个图形接口，最重要的图形应用软件包括一些绘图类型。
- 如何为 X 应用软件配制颜色。包括 X 颜色数据库的说明。
- 如何写出正文。还将讨论如何通过 X 调用不同字体。
- 怎样处理事件。在显示器发生某一事件时，应用软件如何获知。不彻底理解 X 事件就无法在 X 下编程。
- 如何编制拖动画线，这是基于鼠标绘图应用软件的基础。
- 如何与 X 窗口管理程序交互作用。

第一章 建立第一个 X 程序

在这一章，将讨论如何编制第一个 X 程序，这一程序中打开了与 X 服务器的连接并报告有关服务器的一些信息。第一个 X 程序的源代码由三个 C 语言程序文件构成：`initX.C`、`quitX.C` 和 `test1.C`。随后又通过第一个程序建立另外一些实例程序。其中第二个实例程序打开一个最小的窗口，第三个实例程序利用 `openWindow` 函数创建窗口并增加一个图符(icon)位映像。

本书各章采用以下相同的结构。程序的不同部分都分隔成模块，再在以后运用这些模块。各章中，只有在前面章节出现过的文件有所改变时，才把它们列出。有些章节由于新概念的介绍而加入更多的源代码到这些文件中，但力求加入这些新代码后具有再用性。

X 是一个面向网络的窗口系统，它分成两部分：服务器(Server)和客户(Client)。服务器用来控制显示屏幕。而客户则是诸如正文编辑器，时钟显示和表格等应用软件程序。客户请求服务器的一些服务，象打开窗口和在窗口中画线或写正文等。然而，只有服务器才拥有一个或多个屏幕。客户通过连接与服务器通信，这种连接通常是网络型连接。因此任何 X 程序的第一步必须是与服务器建立连接。

`Xopendisplay` 调用就是和服务器建立网络连接。根据所使用的运行 X 服务器的机器，这种连接可以是 TCP/IP 插座，或者是 DECnet 连接，或者可能是与局部机器连接的公共内存。当然，不用担心，`XOpenDisplay` 会建立恰当的连接。`XOpenDisplay` 只要一个参数——而想要通话的显示设备(服务器)的名字，它是一个字符串。典型的缺省显示设备是局部机器和屏幕。如果使用的是一个监视器的 Sun386i 工作站，则缺省显示设备是带有 Sun 的键盘和鼠标的外设屏幕。下面是打开与缺省设备服务器连接的示例代码。

```
#include <stdio.h>
#include <X11/Xlib.h>
.
.
Display *theDisplay; /* -- A pointer to a display structure */
theDisplay = XOpenDisplay( NULL );
```

每个调用 X 库(或 Xlib)子例程的 C 源程序代码应包括 `Xlib.h` 文件。为方便起见，这个文件放在 X11 子目录下(在运行 UNIX 操作系统的机器上通常放在 `/usr/include/X11` 或 `/usr/local/include/X11` 中)。X11 子目录下的一些其它头文件将在以后需要时再陆续介绍。

几乎所有 Xlib 子例程都以大写 X 开始。`XopenDisplay` 函数的参数为显示设备名，并返回一个指向 `Display` 结构的指针。以后几乎每个 X 函数调用都要用到 `Display` 指针，它说明 X 的输出应到哪一个显示设备。

`XOpenDisplay` 的参数为 `NULL` 时说明用缺省显示设备。事实上，它告诉 `XOpenDisplay` 使用 `DISPALY` 环境变量，而 `DISPLAY` 环境变量必须是缺省显示设备名。由于 X 是一个面向网络的窗口系统，因此可以指定在另外机器的显示屏幕上显示窗口。例如，可以在 Cray-2 机器上运行一个数字计算的客户程序，而在 Apollo 工作站上显示器上显示其输出。