

微机操作系统系列丛书(二)

X Window 实用手册

X Window 實用手冊 (含XFree86 for Linux)



劉順吉/曾守民 著

学苑出版社

希望

微机操作系统系列丛书(二)

X Window 实用手册

刘顺吉 曾守民 原著

李 东 木 子 改编

甘登岱 东 岳 陈金凤 审校

学 苑 出 版 社

(京)新登字 151 号

内容提要

本书对 X Windows 系统的管理与使用作了系统介绍,内容包括 X 窗口系统的安装与设置、X 窗口系统的管理与使用、基于 X 窗口的各种应用软件及游戏软件的介绍。

本书内容新颖、叙述清晰,在保证可理解性的前提下,希望能在有限的篇幅内覆盖丰富的内容。本书可适合于 X Windows 的使用者及系统管理员使用。

欲购本书的读者可直接与北京海淀 8721 信箱书刊部联系。邮政编码:100080,电话:2562329。

版 权 声 明

本书繁体字中文版名为《X Window 实用手册》,由第三波文化事业股份有限公司出版。版权归第三波文化事业股份有限公司所有。本书简体字中文版由第三波文化事业股份有限公司依出版授权合同约定,授权出版。未经出版者书面许可,本书的任何部分均不得以任何形式或任何手段复制或传播。

■ 操作系统系列丛书(二)

■ X Window 实用手册

原 著:刘颖吉 曾守民
改 订:李 东 木 子
审 定:甘登岱 东 岳 陈金凤
责任编辑:魏国宪

出版发行:学苑出版社 邮政编码:100036

社址:北京市海淀区万寿路西街 11 号

印刷:北京昌平沙河建华印刷厂

开 本:787×1092 1/16

印 张:12 字数:274 千字

印 数:1~5000 册

版 次:1994 年 10 月第 1 版第 1 次

ISBN 7-5077-0885-3/TP·27

本册定价:17.00 元

学苑版图书印、装错误可随时退换

前 言

由于计算机硬件价格的不断下降和性能的不断提商,使得大型主机已不多见,而个人计算机、RISC 工作站及 GUI 已成为计算机发展的主流。苹果公司推出 Macintosh、微软公司推出 Windows 3.1 所产生的影响也说明了图形用户界面势不可挡的趋势。而以前仅能在工作站上执行的 X 窗口系统,曾因高不可攀的硬件需求及复杂的操作方式而阻碍了其发展,而今天在 486 微机上配备 SVGA 卡、调制解调器及一套免费的 LINUX 操作系统,也能使用户享受工作站的操作环境。

本书是作者根据长期管理和使用 X Window 的实践经验而编写的,其内容侧重于如何管理和有效地使用 X Window,可适于 X Window 的使用者及系统管理员使用。

由于提到的软件非常多,部份软件甚至都能单独出书,而且大部分窗口软件,只要试用一两次,使用者即可熟悉,因此本书未提供大部分软件的使用细节,这些使用细节主要靠读者去挖掘。

本书附录有作者管理工作站时所用的设置文件,以及常见的问题和解答,系统管理员可参考并应用到自己的环境中。

本书是根据刘顺吉等编著的《X Window 实用手册》一书改编而成。在改编过程中力求忠实原著的基础,对原著中不当之处以及不符合大陆情况的部分做了适当的修改。

本书由李东和木子改编,甘登岱、东岳和陈金凤审校,参与改编工作的还有刘彬、郑文化、章林、周东明、高峰、连志强、吴斌和章文东等。

限于改编者的水平,书中不妥之处在所难免,敬请读者批评指正。

改编者

目 录

第一章 X 窗口系统介绍	1
1.1 X 窗口系统简介	1
1.2 X 窗口的特性	2
1.3 使用 X 窗口时须具备的一些基本概念	2
1.4 结束语	5
第二章 X Window 的安装及设置	6
2.1 Liunx 系统下的安装	6
2.2 其它工作站的安装.....	14
第三章 窗口管理系统的介绍及设置	19
3.1 基本概念及操作方法.....	19
3.2 twm 的设置	20
3.3 Olwm 的设置	25
3.4 mwm 的设置	30
3.5 结束语.....	36
第四章 资源管理	38
4.1 何谓资源.....	38
4.2 如何指定资源.....	39
4.3 资源的设置.....	42
4.4 资源载入的管道.....	44
4.5 xrdp 的使用及其他工具程序	46
第五章 基本 X 窗口的应用程序	50
5.1 X 共同命令列参数.....	51
5.2 再论字型.....	54
5.3 基本 X11 图形处理软件	59
5.4 X 窗口系统的桌面工具.....	63
第六章 X Window 的进阶应用软件介绍	71
6.1 终端模拟程序.....	71
6.2 编辑程序.....	77
6.3 绘图处理工具.....	79
6.4 图像处理工具.....	92
6.5 其余工具.....	98
第七章 游戏软件	100
7.1 棋类游戏	100
7.2 牌类游戏	106
7.3 俄罗斯方块	110

7.4 其他游戏	113
第八章 X 窗口系统的设置	122
8.1 基本 X 窗口设置程序	122
8.2 X 窗口显示管理程序(x display manager)—xdm	129
8.3 结束语	133
第九章 集成 X 窗口的应用软件	134
9.1 RLaB 矩阵运算软件	134
9.2 结束语	139
附录 A startx 设置范例	140
附录 B .xinitrc', csession 设置范例 ..	142
附录 C .Xresources 设置范例	144
附录 D xdm 设置范例	146
附录 E twm 设置范例	148
附录 F olvwm 设置范例	158
附录 G MWM 设置范例	165
附录 H X 常见的问题及解答[FAQ]	174

第一章 X 窗口系统介绍

1.1 X 窗口系统简介

X 窗口(X Window)系统是一个以网络为基础(network-based)的图形窗口系统。它是在 1984 年由麻省理工学院(MIT)在 Project Athena 的基础上所开发出来的用户图形界面系统,当前的 X 窗口操作系统已广泛地为工业界所采用并成为工业标准软件系统。X 窗口的一个重要特征是它提供一种与设备无关的基本结构,使几乎所有形式的图形用户界面(GUI)皆可由此基本骨架构建出来。任何硬件只要提供 X 协议(X protocol),便可以执行应用程序显示一组包含图文的窗口,而无需重新编译和连结。这种与设备无关的特性,使得只要是根据 X 窗口标准所开发的应用程序,均可在不同的环境下如大型计算机、工作站、个人计算机上执行,因而奠定了当前 X 窗口成为工业标准的地位。

X 最早是由 MIT 在 DEC 的支持下所开发的。其 X 名称及其最初的设计思想是来自于由 Stanford 大学的 Brain Reid 与 Paul Asente 所设计的名为 W 的窗口系统(因为在英文字母的顺序中,W 的下一个就是 X),而 X 窗口是在 MIT 的计算机科学实验室内,为了 Athena 计划所完成的分散式、与硬件无关的用户接口。早期版本主要用于 MIT 及 DEC 内,但当改进至第 10 版时,许多制造商极感兴趣于使 X 窗口成为商业产品,因而纷纷投入开发行列。虽然有其他许多人对 X 窗口的第 11 版有贡献,但前 10 版 X 窗口主要是由 MIT 的 Robert Scheifler 与 Ron Newman 以及 DEC 的 JIM Gettys 所撰写。第 11 版的 X 窗口是由硬件及软件供应协会所支持,该会会员承诺通过其生产线,使 X 窗口成为基本的用户标准界面。X 窗口可以在大多数的 UNIX 系统的计算机上执行,如 Sun,DEC,IBM,HP 等,其它如 DEC 的 VAX/VMS,MS-DOS,及一些非 UNIX 的系统上也可以执行。

另外有一些厂商如 ATT, Adobe ,CDC,Data General,Fujitsu,Prime ,Siemens ,Silicon Graphics,Sony ,Texas Instruments,Wang,Xerox 均表示支持 X 窗口,而许多公司还特别生产为支持 X 窗口通讯协议所设计的硬件,如 X terminal。

现今市面上有许多的商用窗口软件,如 Sun 的 Sunview 及 NeWS,PC 上的 MS-Windows,Apple 公司 Macintosh 上的窗口环境等等各有其特色。而 X 窗口与其他窗口产品最重要的差别在于 X 窗口不定义任何独特的用户接口模式(style)。X 窗口提供一个机制(mechanism)支持许多接口模式而不是强行制定任何政策。其它的窗口系统,大多仅支持一个特定的用户接口。相应的 X 窗口提供弹性的、基本的窗口操作功能,从而使用户可以自行设计所要显示的方式,虽然或许有人一直无法理解为何同样都是 X 窗口管理程序而给人的观感(look and feel)却完全不一样,但其实它们都是用基本的 X 窗口通讯协议相联系的。X 窗口基本系统不提供用户接口部分,像按钮框(button boxes)、选单(menus)与对话框(dialog boxes)等。大多数的应用系统依赖于基本 X 窗口通讯协议上的高层程序库来提供这部分。

1.2 X 窗口的特性

以下三大特征说明了 X 窗口大部分的功能及其受用户欢迎的原因:

- 一、X 具有网络透明性(Network Transparent):通过网络传输 X 通讯协议,应用程序可以在网络上任何一台有 X 窗口服务器的机器上显示执行的结果及接受输入。这种通讯结构和网络上另一端其它机器的机种,甚至操作系统都没有关系,完全达到与设备无关的境界。总而言之,程序可以在另一种不同的显示器类型下执行,而无需重新编译(re-compilation)和重新连结(re-linking)。
- 二、可支持许多自由风格(Policy Free)的用户接口:基本上,管理窗口的功能,例如窗口的位置、大小及显示顺序等等,并不包含于 X 窗口服务器系统中,而是由应用程序来控制,因此程序设计员及用户可轻易选择自己所喜欢的风格。不同风格的接口与不同的应用程序有关,就像 Open Look 和 Motif 对选单及滚动甚至窗口的管理方式都完全不一样,但是,只要是用标准 X 窗口程序库开发的软件,保证可以在两种环境下执行。
- 三、X 不是计算机操作系统的一部分:大部份的窗口系统都是以系统核心为基础的(kernel based),也就是说他们与操作系统息息相关,没有该操作系统就无法生存。如 DOS+Windows, Macintosh 窗口, Sun 的 SunVieW 等等。然而对操作系统而言,它只是一个应用程序而已,因此, X 窗口很容易在不同的操作系统上安装,如最常见的在 UNIX 下(不管是 BSD, SYSV, OSF/1 等等),皆有 X Window, 其它如 DEC VMS、Microsoft Windows(DOS 下也有)、IBM OS2 等等,都有 X 窗口的服务器或客户程序。可以看到的是,当前能在绝大多数工作平台(Platform)上执行的、就属 X 窗口操作系统了。

1.3 使用 X 窗口时须具备的一些基本概念

以下我们来看看在使用 X 窗口系统时,需要了解的一些基本概念:

一、客户服务器模式(Client-Server Model)

X 窗口系统的基础结构是客户服务器模式。在整个工作状态下,窗口系统可分为:

- (1)显示服务器:它完全控制输入输出设备,能在屏幕上产生文字及图形并处理窗口,同时监控键盘及鼠标器(Mouse)。
- (2)客户程序:在 X 窗口系统下执行特定工作的应用程序。

服务器充当介于应用程序(application)与显示硬件及输入设备间的可移植层(portable layer)。当输入数据或移动鼠标器时,服务器会将这些信息传至相关的 Client 程序,而 Client 程序也会要求 Server 处理它所发出的信息,例如在屏幕上移动一个窗口或一条线等等。尽管某些厂商开发出将 X Server 写入硬件或固件(firmware)的产品,但 X Server 基本上是要在具有图形显示功能的工作站或 PC 上执行。

Client 主要是通过使用一个非同步字节流(Asyncheonousbyte-stream)通讯协议,也就是 X 通讯协议(X protocol)与 X Server 联系。最常见的 X 支持的网路通讯协议是 TCP/IP,当然,由于 DEC 的大量投资,它也支持 DECnet。

一个 Server 可以同时接受多个 Client 程序一起执行,而一个 Client 也可以连接到许多不

同的 Server。只要 Client 与 Server 两者都遵守 X 通讯协议,那么任何 Client 都能与任何 Server 通讯。

通常我们从远程机器移一个 Client 程序到自己机器上的理由有很多种,一般说来,大多是利用远程机器所拥有的强大运算能力的工作站(或超级计算机),利用它的运算速度来加快我们的作业,或是像并行处理机或特殊结构的机器,或是像只有在该机器上才能执行的软件(一套要花很多钱的那一种),或者是把它当文件服务器用(可能有一些大文件要处理而你不原意从网络传到你的机器上来处理),理由繁多,不过目的只有一个,那就是方便好用。

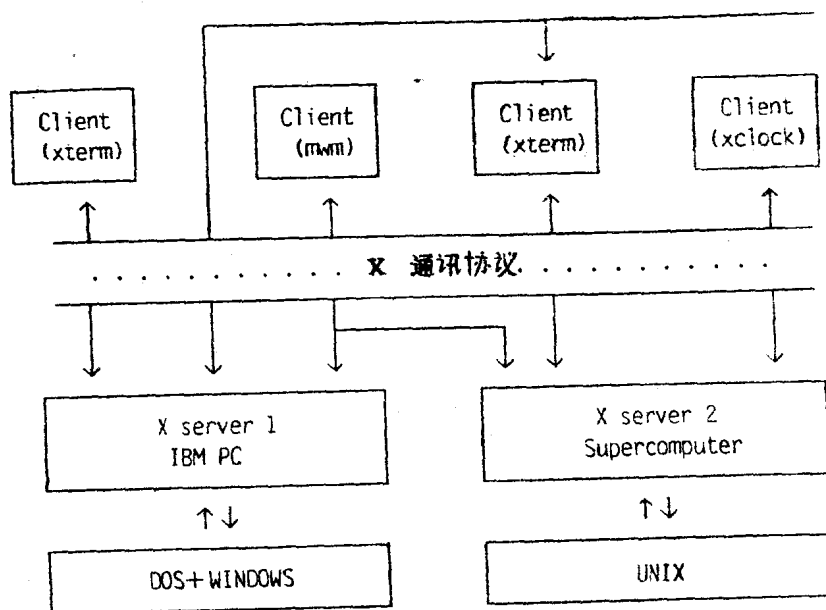


图 1.1 X server 与 client 的关系图

二、显示与屏幕(Displays and Screen)

对计算机用来显示图形与文字的阴极射线管(CRT)而言,显示与屏幕经常是互相代换使用的。然而在 X 中我们使用显示表示一个 X Server,而一个屏幕表示一个单一的硬件输出设备(通常就是一个屏幕)。一个单一 X 显示可支持管理许多屏幕。不过通常正常状况下是一个主机对应一个显示。在一个客户能与 X Server 通讯之前,它必须先打开一个 connection(连接)到服务器上。只要一个客户与 Clipper 建立了一个连接,它就能使用任何一个 Server 控制的屏幕。

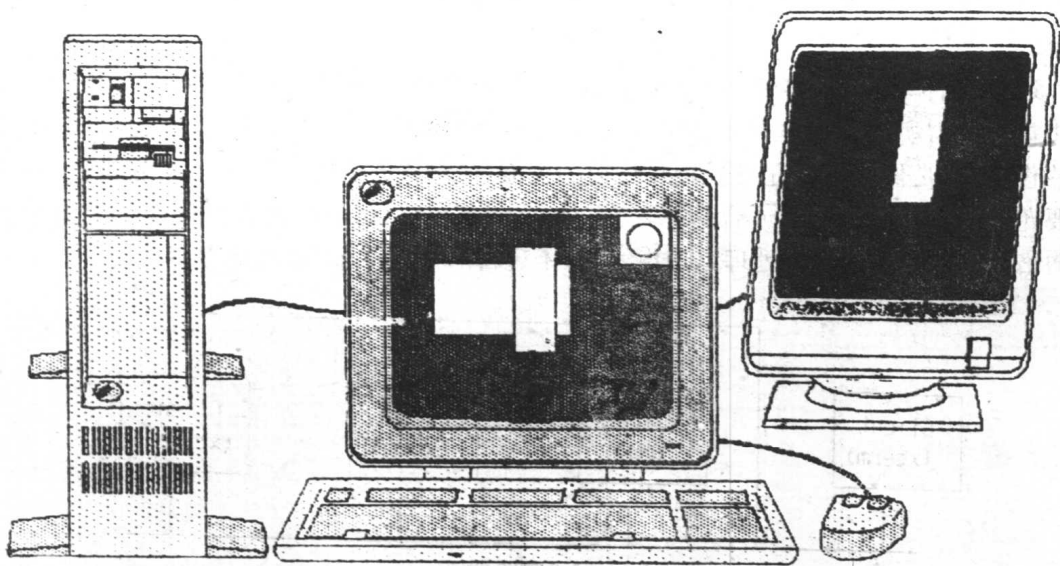


图 1.2 一个 Display Server 和两个 Screen

三、资源 (Resources)

在 X 整个窗口环境系统中,我们所称的“资源”包括窗口、位图(bitmaps)、字型(fonts)、颜色及其他一个应用程序所需用到的数据结构。X Server 在 Server 内独立地维护这些数据,使 Client 能正确地使用及共享(share)这些资源结构。Client 程序则可以通过资源识别码(Resource ID)来取得系统的资源或产生自己本身要用的资源。当 Client 执行结束时 Server 也会自动收回该程序所配置的资源。有关整个资源环境的设置我们在后面的章节中会详细介绍。

四、基本窗口概念

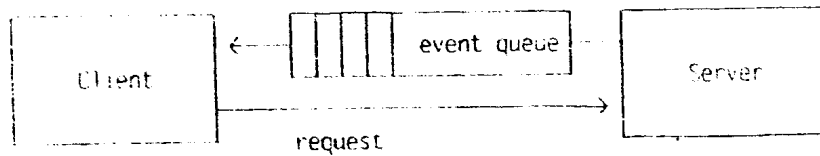
X 窗口最基本的资源是窗口,一个窗口代表一个屏幕上的一个长方形部分。一般讲,X 窗口不像其他窗口系统,在 X 窗口中的窗口是不含标题杆(title bar)、滚动轴(scroll bar)或其他装饰的。X 的窗口看起来像一个用背景色或图样所构成的长方形区域,每一个窗口也含有边界线(border)。或许已经看过 Motif 或 OpenLook 的操作环境,发现每个窗口外面皆有一定的使用风格,那是由窗口管理员(Window Manager)组合两个或多个窗口以产生滚动轴、标题杆及其他较适合使用的用户接口,也只有在这种环境下才会有这种外观。

五、事件模式与请求(Event Model and Request)

X Server 与 Client 的通讯都是通过传送事件(event)到 Client 应用程序来完成的。用户对任何设备操作的结果使 Server 产生事件(例如按下键盘的一个键、移动鼠标器(Mouse)或按下鼠标器的一个键等等)。当 Client 收到 Server 所传送过来的事件时,经过适当处理后,向 Server 发出请求(request),以使 Server 对所显示的画面产生适当的反应。譬如说,Client 在收到 Server 传来移动鼠标的事件时,如果鼠标正好在可以移动该窗口的区域(如标题杆区),则 Client 可以发出要求,请 Server 将该窗口的位置移动,以达到移动该窗口的目的。

Server 都是将事件放入先进先出(FIFO)队列(Queue)中让 Client 可以去读取。每个事件都是由包含事件类型、事件占据的窗口及其他描述该事件的数据所组成的数据结构。大多数的

X 窗口应用系统完全是事件驱动,且设计成一直等待事件发生、响应事件、然后再等待下一个事件。事件驱动的处理方法,提供了一个交互式应用的自然模式。



六、窗口管理员(Window Manager)

在许多窗口系统中窗口管理员与窗口的其他部分是不可分离的,如 MS-Windows, OS/2, Macintosh 的窗口系统。然而在 X 窗口,一个窗口管理员只是一个普通的 Client 应用程序,阅读以下内容时,会发现居然有 twm, mwm, olwm, olvwm... 等等很多的窗口管理程序,以使每个人选用自己所喜欢的窗口管理员。还由于 X 窗口具有网络透明性,因而不不管在网络上的哪个机器,也不管该机器是否有需要窗口管理程序,都可以调用任何一部具有该窗口管理程序的机器,将该窗口管理员的 display 选项设为现在所使用的机器。譬如较习惯 HP 工作站 OSF/Motif 的 mwm,可是当换到另一部 Sun 的工作站时发现该部 Sun 只有 Open Look,如果还是坚持用 mwm 的话,这时可以在 Sun 的机器上把 X server 唤醒,然后注册到该部 HP 工作站,输入以下的命令(假设该部 Sun 作站的 Node name 为 cc_sun1):

```
% mwm -display cc_sun1:0
```

窗口管理是一项复杂的课题,基本上,它不只影响用户如何与一个系统通讯,也影响应用程序之间以及应用程序与 Server 之间的通讯。尽管当前的窗口管理程序有好几种,不过当前以 OSF/Motif 和 Open Look 较受欢迎,同时支持的厂商也较多。未来整个 UNIX 环境下的窗口管理程序是否会统一我们不得而知,尤其是在一个以打败 UNIX 为目标的 Windows/NT 出来以后!

1.4 结束语

在本章我们介绍了 X 窗口系统的基本结构及基本名词。X 窗口允许程序设计员开发可移植于任何支持 X 窗口系统的复杂用户接口。X 窗口所具有的网络透明性及客户服务器模式使 X Server 产生及处理窗口,以响应来自客户的请求,且发送用户输入或改变一个窗口状态的事件来通知 Client。Client 可在网络的任何地方执行,而且不管任何机型与操作系统。X 窗口系统成为分散式系统用户界面的最佳选择。X 窗口不支持任何特殊风格,而且极力提倡自由政策,虽然应用程序可以自由使用 X 的原始特性以设置它自己的用户接口模式,但只要依循一些基本原则,就很容易协调一个已存在的环境。

绝大部分的窗口系统将整个系统视为一个“窗口管理员”。然而在 X 窗口中,一个窗口管理人员则是一个 client,和其他 client 没有不一样,它允许使用者移动及处理窗口。用户可以选择不同的窗口管理人员,事实上,甚至没有窗口管理人员,X 窗口的应用程序也可以执行。

第二章 X Window 的安装及设置

本章将介绍 X Window 的安装及设置,包括 PC UNIX 以及一般的工作站。虽然 X11R6 已于 1994 年 5 月正式发行,但其安装程序与 R5 相差不多,且当前 R6 尚不稳定,所以笔者介绍将以 R5 为主。

2.1 Linux 系统下的安装

当前在 Linux 系统之下的 X server,也就是 XFree86,是由与 MIT X11R5 同步发行的 X386 1.2 版改进而来。当前测试过的可以执行 XFree86 的系统有以下所列的操作系统:

```
SVR4.0:
  Esix; 4.0.3A.4.0.4,4.0.4.1
  Microport;2.2,3.1,4.1,4.2
  Dell;2.2.2
  UHC;2.0,3.6
  Consensys; 1.2
  MST; 4.3.0(load 2.07 and Load 3.02)
  ISC; 4.0.3
  AT&T;2.1,4.0
  NCR;MP-RAS
  SunSoft;Solaris x86 2.1
SVR4.2:
  Consensys
  Univel UnixWare
SVR3:
  Interactive;2.0.2,2.2,3.0,4.0
  SCO;3.2,2.3,2.4
  AT&T;3.2.2
Others:
  386SD 0.1 NethSD 0.9, FreeBSD 1.1 (beta)
  BSD/386 1.0
  Mach 386
  OSF/1
  Linux 0.99pl5h
  Amoeba
  Minix-386
```

由于几乎所有版本的 XFree86 都已成形,而且可以从以下的 ftp site 获取,因此我们在此并不介绍如何重新产生一个新的服务器,仅就 Xconfig 文件的设置作一介绍,下面将 Linux 系统的设置作为范例,其它系统则大同小异。

```
ftp. physics. su. oz. ou      -SVR4 binaries
    under /XFree86/SVR4
ftp. win. tue. nl             -SVR4 binaries
    under/pub/XFree86/SVR4
ftp. tvp. com                 -SVR4 binaries
    under/pub/XFree86/SVR4
stasi. brsdley. edu           -SVR4 binaries
    under/pub/Xfree86/SVR4
stasi. bradley. edu           -Solaris x86 2.1 binaries
    under/pub/XFree86/solaris
```

stasi. bradley. edu	-SVR3 (SCO) binaries under/pub/XFree86/sco
ftp. physics. su. oz. au	-SVR3 (SCO) binaries under/XFree86/SCO
blancmange. ma. utesas. edu	-SVR3 (ISC) binaries under/pub/ISC
ftp. prz. tu-berlin. de	-SVR3 (ISC) binaries under/pub/pc/isc/XFree86
tsx-11. mit. edu	-Linux binaries under/pub/linux/packages/X11
ftp. unipi. it	-Linux binaries under/pub/Tinix/XFree86-2.1
XFree86. cdrom. edu	-FreeBSD binaries under/pub/XFree86/FreeBSD/XFree86-2.1
gil. physik. rwth-aachen. de	-FreeBSD binaries under/pub/XFree86
agate. berkeley. edu	-NetBSD 0.9 binaries under/pub/NetBSD/ports
sun-lamp. cs. berkeley. edu	-NetBSD 0.9 binaries under/pub/NetBSD/ports
ftp. cs. mcgill. ca	-NetBSD 0.9 binaries under/pub/NetBSD/XFree86

一般说来,在 Linux 下执行 X Window 最好有配置 1MB(含)以上的 SVGA 显示卡,当前 Linux 下最新的 XFree86 2.1 版所支持的显示卡如下表所列。而除了 ATI 和 Cirrus 卡仅支持 256 色模式外,其余的皆支持 256 色及 monochrome 模式。另外,ET4000/W32 也只有像 ET4000 的功能,并未支持加速的功能。

当前 VGA16 服务器还没有完全通过除错测试,但基本上,它应该可以顺利地工作(除了慢一些以外),而在 ET4000 和 Trident 卡上,该服务器支持 memory banking 的功能使虚拟分辨率可达 1600×1200(要在 1MB 存储器下),对于其它的卡则只有 800×600 而已。

单色服务器也支持一般的 VGA 卡(使用 64K 图像存储器)、Hercules 卡和 Hyundai HGC-1280 卡。但是这些驱动程序不在 XF86-Mono 服务器之中。所以如果要用它们的话,就必须用 Linkkit 重新构置 XF86-Mono 服务器。

其他的硬件就都没有支持了,如 Weitek P9000、TIGA、HITAGX、Microfield、MGA 等等...都没有支持。但在未来的版本中,会支持这些硬件的。不过像 TIGA 与 Microfield 可都不会支持,因为存在着版权与文件的问题。有关各种卡的说明可在 /usr/X386/lib/X11/etc 目录下的 README 文件中找到,里面含有各种选项与设置的说明。

表 2.1 XFree86 所支持显示卡

加速卡	一般 SVGA 卡
8514A/	Tseng Lab/ET3000,ET4000AX ET4000/W32
ATI/Mach 8	Tridnet/TVGA8800CS,TVGA8900B
/Mach 32	TVGA8900C,TVGA8900CL
Cirrus/CLGD5426	TVGA9000,TVGA9000i
S3/86C911	TVGA9100B,TVGA9200CX
86C801	TVGA9230,TVGA9400CX
86C805	TVGA9420
86C05i	WD/WD90C00,WD90C10,WD90C11
86C924	WD90C24,WD90C30,WD90C31
86C928	WD/PARADISE PVGA1
WD/WD90C31	Genoa/GVGA ATI/28800-4,28800-5,28800-a NCR/77C22,77C22E,77C22E+ Cirrus/CLGD5420,CLGD5422 CLGD5424,CLGD6205 CLGD6215,CLGD6225 CLGD6235 Compaq/AVGA OAK/OTI067,OTI077

当前可执行的 XFree86-2.1 for Linux 可由 ftp 从以下地方取得：

tsx-11.mit.edu:/pub/linux/packages/x11/XFree86-2.1

或

sunsite.unc.edu:/pub/Linux/XFree86-2.1

或

mctucca.edu.tw:/UNIX/linux/XFree86-2.1

这些软件都是用 gzip 压缩的 tar files。

XF86-2.1-8514.tar.gz	IBM8514 卡的服务器
XF86-2.1-Mach32.tar.gz	ATI Mach32 卡的服务器
XF86-2.1-Mach8.tar.gz	ATI Mach8 卡的服务器
XF86-2.1-Mono.tar.gz	单色卡的服务器
XF86-2.1-S3.tar.gz	S3 卡的服务器
XF86-2.1-SVGA.tar.gz	SVGA 卡的服务器
XF86-2.1-VGA16.tar.gz	16 vga 卡的服务器
XF86-2.1-bin.tar.gz	X11R5 应用程序
XF86-2.1-cfg.tar.gz	XDM 设置文件和 chooser
XF86-2.1-doc.tar.gz	XFree86 2.1 说明文件
XF86-2.1-fnt.tar.gz	字型
XF86-2.1-fnt100.tar.gz	100dpi 字型

XF86-2.1-fnt75.tar.gz	75dpi 字型
XF86-2.1-fntbig.tar.gz	
XF86-2.1-fntschl.tar.gz	
XF86-2.1-inc.tar.gz	程序库,dynamic stubs,设置文件和包含文件(include file)
XF86-2.1-lib.tar.gz	动态连结程序库,点阵图和基本字型
XF86-2.1-lkit.tar.gz	Linkkit 用以构建 X 服务器
XF86-2.1-man.tar.gz	Manual pages
XF86-2.1-pex.tar.gz	PEX 程序库和范例程序
XF86-2.1-slib.tar.gz	静态程序库
XF86-2.1-xdmsd.tar.gz	
XF86VidDoc.tar.gz	Video Document

需要 libc 4.4.1 或更新的版本才能执行 Free86-2.0 以上。用加速版本服务器的人则至少需要 Linux 0.99p113 或更新的版本。

其它 SVGA,VGA16T Mono 等服务器则需 Linux 0.99p112 或更新版本。而且必须安装有 David Engel's shared dynamic linker ld.so 1.3 或更新版本,这可以从以下的地方找到:

nctuccca.edu.tw 的/pub/linux/packages/Gcc 目录

存储器的需求是至少 8 MB 和虚拟存储 16 MB(永久置换文件)。其实若能小心管理存储器的使用,4 MB 也是可以执行的。我们的建议是至少有 8 MB 的存储器,因为用置换方式效率很差。若要执行占用大量存储器的程序,像 gcc;那么至少应该有 16 MB 用于存储和 16MB 用于置换文件。

不安装 Linkkit 至少要有 17 MB 的磁盘空间,安装的话,则需 21 MB 空间。如果去掉不需要的其它服务器应该也可以节省几 MB 的空间。

安装之前,请备份已变更过的文件,以备不时之需。

安装的步骤如下:(请以 root 身份在/目录下处理)

应将 umask 设置为 022

gzip -dc tarfilename|tar xvvo-

注意:

这个操作将会覆盖旧的 XFree86 版本。

在设置时需要如下的信息:

1. 屏幕规格(如水平和垂直的频率,频宽等等...),这是最重要的事情;
2. 卡的名称。有些公司会用"S3"做为"accelerated"的同义字;
3. 卡所提供的图像存储器有多少;
4. 卡可用的 dot-clocks 有那些,或直接是 programmable 的有那些。此部分是最难设置的。在 RDADME.config 中描述了如何得到这些数据;

5. mouse 使用的协定和连接的设备。在 Xconfig 的 manpage 中列出了可用的 mouse 协定。下面是一些 mouse 连接的 device 名称:

```
/dev/atibm ATI XL busmouse (注意:ATI GU busmouse 事实上就是 logitech busmouse)
/dev/logibm Logitech busmouse (注意:这个用的是 busmouse 的协定,不是 Logitech 的协定)
/dev/inportbm Microsoft busmouse
/dev/psaux ps/2 或 quickport mouse
```

注意:

上面的名称是新的名称,也许在旧的版本中还是使用旧的名称。

new	old	major	minor	device	number
atibm:	bmouseatixl	10	3		
logibm:	bmouselogitec	10	0		

```
inportbm: bmousems      10      2
psaux:    bmouseps2 or ps2aux 10      1
```

其它的 mouse 就属于 serial mouse, 所以也都连上一些 serial ports, 像/dev/ttyS? 或/dev/ttyS?? 或/dev/cua?。

busmouse 有时并不容易看出是连接哪一个 device, 要判别的话就要将所有的 drivers 放到 kernel 中, 并在关机时看看开机时的信息, 它会告诉你它所检测到的 busmouse 是哪一种。此时就会知道了, 它连的是哪一个 device。

现在看 README.Comfig 文件。若屏幕不在 modeDB.txt 文件中, 应该选一个一般的模式。并确定所选的模式规格是否在你的屏幕规格之中。若想调整或换一个模式, 则要先详读 VideoModes.doc, 并依其指示去设置。

警告:

别人的 Xconfig 文件不一定适合你用, 因为各人的硬件设备不尽相同。特别是屏幕模式的设置, 一定要确定是你的屏幕的规格所具有的, 以避免不必要的硬件损坏。

接下来是一个 xconfig 的设置范例, 我们将一一介绍其中的设置。

```
-----Xconfig 设置范例-----
# $XFree86;mit/server/ddx/x386/Xconfig.cpp.v 1.9 1993/03/27/ 09:03:01 dawes Exp $
# $xconsortium;Xconfig.v 1.2 91/08/26 14:34:55 gildea Exp $
#
# copyright 1990. 91 by Thomas Roell. Dinkelcherben/Germany.
# Permission to use, copy, modify, distribute, and sell this software and its
# documentation for any purpose ys hereby granted without fee, provided that
# the above copyright notice appear in all copies and that both that
# copyright notice and thia permission notoce appear in supporting
# documentation, and that the name of Thomas Roell not be used in
# advertising of publicity pertaining to distribution of the software without
# specific, written prior permission, Thomas Roell makes no representations
# about the suitability of this software for any purpose. It is provided
# "as is" without express ot implied warranty.
#
# THOMAS TODLL DOSCLAOMS ALL WARRANTOES WOTH REGARD TO THIS SOGOWARE.
# ONCLUDING ALL OMPLIED WARRANTOES OF MERCHANTAVILITY AND FITNESS. IN IO
# DVENT SHALL THOMAS ROELL BE LIABLE FOR ANY SPECIAL. IMDIRECT OR
# CONSEQUENTIAL DAMAGES OR ANY KAMAGES WHATSOEVER RESULTING FROM LOSS LF USE.
# DATA OR PROFOTS. WHETHER ON AN ACTION OF CONTRACT, MEGLIGENCE OR OTHER
# TORTOUS ACTION. ARISING OUT OF OR IN CONNECTION WITH THE USE OR
# PERFORMANCE OF THIS SOFTWARE.
#
# Author; Thomas Roell. roell@informatik.tu-muenchen.de
#
# Extensive modifications by the XFree86 Core Team
#
#
# * * Refer to the X386 man page for details about the format of this file * *
# * * The X386 man page is installed as/usr/X386/man/man1/XFree86.1X * *
#
#
# some nice paths, to avoid conflicts with other X-servers
#
RGBPath 是设置在 X 窗口中的 rgb.txt 的路径, 这样一来你就可以用颜色名称来
# 代替指定的 R、G、B 的值, 比如说, 以 "navy blue" 代表 #00080.
```

--- (Which are concatenated together),

```

# as well as specifying multiple comma-separated entries in one Fontpath
# command (or a combination of both methods)
#
Fontpath      "/usr/X386/lib/X11/fonts/Type1/"
Fontpath      "/usr/X386/lib/X11/fonts/Speedo/"
Fontpath      "/usr/X386/lib/X11/fonts/75dpi/"
Fontpath      "/usr/X386/lib/X11/fonts/chinese/"

# Fontpath 最主要是指定字型文件搜索的目录, 可以用 xset fp 来改变, 不过在
# Xconfig 中设置的字型文件通常都是系统的字型, 一般说来并不需要改变它, 但
# 是如果有自己的字型的话, 则可以用 xset fp+ 来加入已有的路径之中。

# Uncomment this to cause a core dump at the spot where a signal is
# received. This may leave the console in an unusable state, but may
# provide a better stack trace in the core dump to aid in debugging
#
# NoTrapSignals

#
# enable this to use the XQUEUE driver for keyboard and mouse handling
# under System V. This may go away in the future.
#
# Note-If you use XQUEUE, you must comment out the keyboard and
# mouse definitions.
#
# Xqueue

#
# Keyboard and various keyboard-related parameters
# 设置键盘重复的时间间隔(以 ms 为单位), 并设置数字键盘是否锁定。
Keyboard
    AutoRepeat 500 5
    ServerNumLock
# Xldes          1 2 3
# Dontzap
#
# To set the LeftAlt to Meta. RightAlt key to ModeShift.
# RightCtl key to Compose, and ScrollLock key to ModeLock:
# 下面的键盘对应是为配合使用习惯而设置一些修饰键的设置, 不过通常也不
# 用改它, 如果有需要的话只要将前面的井号(#)拿掉便可。
# LeftAlt      Meta
# RightAlt     ModeShift
# RightCtl     Compose
# ScrollLock   ModeLock
#
# Mouse definition and related parameters
# 下面设置鼠标器的设备(Device), 如果是 Serial Mouse 的话, 通常只要如
# 下设置便可(因为大部份的鼠标器皆与 Microsoft Mouse 兼容), 如果你的
# /dev/mouse 并不存在的话, 那么可以直接指定鼠标器所在的 Com port。
# 如果在 COM1, 则将"/dev/mouse"换成"/dev/cua0"
# 如果在 COM2, 则将"/dev/mouse"换成"/dev/cua1"
# 如果发现设置不对(起动 X server 时, 它会提示 mouse dtype 不对)
# 这时, 将第一行前面加上(#), 然后将第二行(MouseSystems)前的井号
# 拿掉, 再试试, 如果不行, 把第二行再加上井号把第三行的井号拿掉, 依序试
# 下去, 总会有一个成功的, 如果都不行, 那... 换鼠标吧!!
# 还有, 如果你的鼠标是 bus mouse, 通常选 Busmouse 那一行很有可能是对的

Microsoft     "/dev/mouse"
# MouseSystems "/dev/mouse"
# MMSeries     "/dev/mouse"
# MouseMan     "/dev/mouse"
# Logitech     "/dev/mouse"

```