

第 5 章 X Window 系统

(翻译中出现的任何问题或错误，请广大读者及时反馈给我：freebsdhandbook@163.com)

5.1 概要

FreeBSD 使用 XFree86 来为用户提供一个功能强大的图形用户接口。XFree86 是一个开放源代码的 X 视窗系统。下面的这章将介绍在 FreeBSD 上安装和配置 Xfree86。有关你的显卡是否被 X11 所支持，请查看 Xfree86 的网站。

读完这章，你将了解到：

1. 如何安装和配置 Xfree86.
2. X 视窗系统的不同组件，它们是如何协同工作的。
3. 如何安装和使用不同的视窗管理系统。
4. 如何在 XFree86 中使用 TrueType 字体。
5. 如何为你的系统安装图形登陆口 (XDM)。

在阅读这章之前，你应当了解：

1. 知道如何安装额外的第三方应用程序 (第 4 章)。

5.2 安装 XFree86

XFree86 被制作成一个 port 和一个 package, 使得它很容易被安装。你可以直接从 XFree86 组织的网站直接下载，并根据 XFree86 组织的指示手动安装它。

你只需要决定安装哪个版本。XFree86 3.X 是 XFree86 开发的一个分支。现在，已经不对它更新了。现在 XFree86 4.X 是重新设计的一个分支。它有很多新特性，但它支持的显卡相对要少一些。

如果你的显卡能在 XFree86 4.X 下运行，就不要使用 XFree86 3.X 了。这章的其它部分将介绍如何配置 XFree86 和你可以使用的几种 X 程序。

5.3 XFree86 的配置

5.3.1 介绍

本章将介绍在 FreeBSD 上安装和配置 XFree86 X 视窗系统的步骤。一旦 X 服务器被安装和正确配置。用户就可以看看第 5.7 节来设置桌面环境。

5.3.2 开始之前

在用户开始配置 XFree86-4 之前，你必须知道你机器的下面一些信息：

1. 显示器的规格
2. 显卡的芯片
3. 显存

显示器规格可以被 XFree86 用来决定显示器的刷新率。这些规格通常可以从显示器所带的文档中找到，也可以从生产商的网站找到。需要知道两个数字范围：垂直刷新率和水平刷新率。

显卡的芯片组定义了 XFree86 要使用什么驱动模块来驱动硬件。对于绝大多数的芯片，它都能自动选择，但有时也可能检测错误。显卡的显存大小决定了目标系统的刷新率和颜色深度。这非常重要，所以用户需要知道目标系统的详细情况。

5.3.3 安装 XFree86 4.X 的软件

XFree86 4 能够使用 FreeBSD 的 ports 进行安装或使用 `pkg_add` 命令。如果用户直接从源代码来构建 XFree86-4，而且在 `/etc/make.conf` 中已有这个项目。如果想得到 XDM-AUTHORIZATION-1 的支持，用户可能需要加上 `Wraphe1 p.c`。这个文件可以在建构 ports 以前被放置在 `file/` 的子目录中。

5.3.4 配置 XFree86 4.X

配置 XFree86 4.X 有几步。第一步是建一个初始化的配置文件。作为超级用户，可以这样做：

```
# XFree86 -configure
```

这会在一个叫做 XF86Config.new 的当前工作目录中产生一个 XFree86 配置文件。XFree86 将试图探测到系统中的显示设备。在你的机器上，将写入一个配置文件以便探测硬件时能够加载正确的驱动程序。

下一步将测试当前的配置以检验 XFree86 是否能在当前系统的显示设备上正常工作。要完成这个任务，你只需要这样做：

```
# XFree86 -xf86config XF86Config.new
```

如果用户看到一个黑灰的格子和一个 X 形的鼠标指针，配置就是成功的。要退出测试，只要键入 ctrl , alt 和 backspace。接下来，用户需要调整 XF86Config.new 配置文件以适应个人的口味。用 emacs 或 ee 这样的编辑器打开文件。用户要做的第一件事是为当前系统的显示器添加频率。这些包括垂直和水平的刷新率。这些值只要在 XF86Config.new 文件的 "Monitor" 地方把这些值添加进去，例如：

Section "Monitor"

```
Identifier "Monitor0"

VendorName "Monitor Vendor"

ModelName "Monitor Model"

Hori zsync 30-107

VertRefresh 48-120
```

EndSection

Hori zsync 和 VertRefresh 关键字通常不存在用户的配置文件中。如果它们不存在，就需要添加它们。正确的水平同步刷新率被添加在 Hori zsync 关键字后面，垂直的同步刷新率被添加在 VertRefresh 关键字后面。在上面的例子中已经添加了。

当 XF86Config.new 配置文件被编辑器打开时，用户需要选择默认分辨率和颜色深度。如下：

Section "Screen"

```
Identifier "Screen0"

Device      "Card0"

Monitor     "Monitor0"

DefaultColorDepth 24

SubSection "Display"

    Depth    24

    Modes    "1024x768"

EndSubSection
```

EndSection

DefaultColorDepth 关键字描述了用户希望运行的默认的颜色深度。Modes 关键字描述了屏幕的分辨率。在上面的例子中，默认的颜色深度是 24 位色，分辨率是 1024x768。最后，用户可以写入配置文件，然后使用这个配置文件测试它。如果所有都正确，那配置文件需要被安装在 xfree86 的默认位置，如下：

```
/etc/X11/XF86Config 或 /usr/X11R6/etc/X11/XF86Config.
```

```
# cp XF86Config.new /etc/X11/XF86Config
```

一旦配置文件被放置在一个普通的位置，**XFree86** 就能通过 xdm 来使用。为了使用 startx，用户将必须安装 X11/wrapper port。

5.3.5 高级配置主题

5.3.5.1 配置 Intel i810 显示芯片组

配置 Intel i810 芯片组的显卡需要有针对 XFree86 的能够驱动显卡的 AGP 编程接口。要启用这个编程接口，agp.ko 的内核引导模块需要用 kldload 来加载到内核。这可以由 loader 来自动加载。只要在启动时，在 */boot/loader.conf* 中加入下面这一行，就可以由 loader 来加载 agp.ko 了。

```
agp_load="YES"
```

接下来，需要为编程接口创建一个设备。要创建 AGP 设备节点，可以在 /dev 目录下运行 `makedev`：

```
# cd /dev  
  
# sh MAKEDEV agpgart
```

5.4 理解 X

如果你非常熟悉使用窗口类型的操作系统，象 `windows`, `Macos`，那你会发现你对 X 的初次印象会很不同。今天，象 `KDE` 和 `GNOME` 这样的 `unix` 桌面环境已经变得非常流行。

5.4.1 Why X?

X 并不是针对 `UNIX` 的第一个视窗系统，但它是最流行的。X 的最初开发团队在开发 X 之前，就已经在为另一个视窗系统工作了。那个系统的名字是“W”（很明显，就是“window”）。X 只是在罗马字母中 W 后面的一个。

X 可能涉及很多的东西，象“X Window System”，“X11”和其它的一些。“X Windows”这个概念会尽可能地避免使用；可以看看 X 的联机手册了解更多信息。

5.4.2 X 客户端/服务器端模型

在 X 模型中，“X server”运行在有键盘，显示器，鼠标的电脑上。Server 用来管理显示信息，和处理来自键盘和鼠标的输入等。每个 X 应用（如 `XTerm` 或 `Netscape`）就是一个“client”。一个 client 发送信息给服务器（如：`Please draw a window at these coordinates`），然后服务器就返回处理信息（如：`The user just clicked on your OK button`）。如果你在只有一台电脑的家里或小型办公环境中使用 `FreeBSD`，你就需要在同一台电脑上运行 X server 和 X client。然而，如果你有很多运行 `FreeBSD` 的机器，你可以在你的桌面电脑上运行 X server，而在比较高档的服务器上运行 X 应用程序。在这样的环境中，X server 和 X client 之间的通讯就可以通过网络来进行。

这可能会使一些人搞糊涂，因为 X 术语与他们所想象的有所不同。他们以为 X server 是运行在大型机器上的，而 X client 是运行在桌面机器上的。

正如你所记得的，X server 是用来处理显示器和键盘输入信息的，而 X client 是用来显示视窗程序的，这一点你不应当搞糊涂。

这样的设计有一个很有意思的副作用。并不需要 client 和 server 都运行在同一种操作系统上，或甚至运行在同一种类型的电脑上。把它运行在 Microsoft Windows 或 Apple's MacOS 上也是可以的，在它上面也有很多免费的和商业化的应用程序。

在 FreeBSD 上使用的 X server 叫做 **XFree86**，在大家很熟悉的许可协议下可以免费使用。针对 FreeBSD 的商业化 X server 也可以使用，你可以选择一个。

5.4.3 视窗管理器

X 的设计哲学有点象 unix 的设计哲学，“tools, not policy”。这些意味着不需要命令最终用户他们应当如何来完成他们的工作。而是，只给用户提供一些完成工作的工具，至于如何使用这些工具那是用户自己的事情。

这种哲学大大地扩展了 X，它不一定要规定这些视窗在屏幕上看起来象什么，你要如何移动鼠标，需要使用什么键来切换窗体，每个窗体的工具条应当看起来象什么，是否应该放关闭按钮等等。

实际上，X 只代表了叫做“Window Manager”的应用程序。有很多这样的应用程序：AfterStep, Enlightenment, ctwm, ftwm, sawfish, twm, WindowMaker 等。这些管理器中的每一个都提供了一个不同的界面；其中一些支持“虚拟桌面”，有一些允许你可以定制一些键来管理你的桌面；一些用作“start”按钮；一些用作“themeable”，允许你启用一个新桌面来改变你的桌面显示。这些视窗管理器都可以在 Ports Collection 的 x11-wm 中找到。另外，KDE 和 GNOME 桌面环境都有它们自己的视窗管理器与桌面集成。每一个视窗管理器也有一个不同的配置机制；一些需要你手工写入一个配置文件；其他特性的 GUI 工具能够完成绝大多数的配置任务；至少有一个配置文件需要用 Lisp 语言来改写，它很强大。

正如你可以想象的，这种灵活性来自价格。因为可用的选择是如此广泛，以至于很难有这种类型的文档来指导你选择和配置一个视窗管理器。

着重点：视窗管理器的另一个特性是鼠标的焦点。每个视窗管理器如 X, Microsoft Windows 或其它一些都有一个抽象概念“焦点”，在任何时候只有一个窗口有焦点。有焦点的那个窗口是你接收所有键盘输入和鼠标点击的窗口。

你通常比较熟悉一个叫做“click-to-focus”的焦点方案。这意味着你必须点击你想要拥有焦点的窗口。窗口可能会处在屏幕的上方，即使你把鼠标移出窗口，你键盘的输入仍然会出现在这个窗口中。你要把焦点移向另一个窗口，你必须点击它。

X 不支持一些特殊的焦点规则。而是，窗口管理器控制着在什么时候哪个窗口拥有焦点。不同的窗口管理器支持不同的焦点方案。它们都支持点击到焦点，它们中的大多数都支持好几个。

还有其它两个通常的焦点规则：

focus-follows-mouse

这种方式是，在鼠标指示器下面的视窗是拥有焦点的视窗。这可能不是其他视窗顶部的视窗。你可以通过点击另一个视窗类改变焦点。习惯了这种方式后，再回到使用click-to-focus方式的视窗时就会感到不适应。

sloppy-focus

这种方式是对focus-follows-mouse的一个小小的扩展。对于focus-follows-mouse，如果你在桌面上移动鼠标，那不会有视窗会获得焦点。

如果你使用sloppy-focus然后移动鼠标指示器，那拥有焦点的视窗将保留。你的视窗管理器也将支持其他的方案，所以确信你已经阅读了它提供的介绍文档。

Microsoft Windows 和 Apple MacOS 都有一个快速的窗口方案。应用开发人员被建议确保他们的应用程序共享一个普通的所见即所得的用户界面。在这些视窗系统之前，X 已经被设计了很长时间了。当 GUI 系统的研究还处于早期时，还没有感觉到要一个特殊的图形接口。如果你检查一下过去十年的 Windows 和 MacOS GUI，你将看到在人机交互的设计上发生了多大的变化，融入了多少新的思想。

这样的结果是你不能期望 X 应用程序只拥有一个普通的界面。有很多的流行的窗口设置，包括 Athena(MIT 开发)，Motif(模仿微软的)，openlook 和其他一些。今天，绝大多数比较新的 X 应用程序将使用一个现代风格的窗口设计，象 Qt，用来设计 KDE，或 GTK，用来设计 GNOME。

5.5 在 XFree86 中使用字体

5.5.1 Type1 字体

XFree86 使用的默认字体不是很理想。大型的字体显得参差不齐，看起来很不专业，在 netscape 中，小字体也显得莫名其妙。然而，有几个自由的，高质量的字体可以使用在 XFree86 中，version 3.X 或 version 4.X 都可以。例如，URW 字体的集合就包括了标准的 type1 字体(Times Roman, Helvetica, Palatino and others)。在 freefont 的集合中也包括了許多字体，但他们中的绝大部分使用在图形软件中，如 Gimp, 在屏幕字体中使用并不完美。

要安装上面的 Type1 字体，你只需要运行下面的命令。

```
# cd /usr/ports/x11-fonts/urwfonts
```

```
# make install clean
```

Freefont 或其他的集合也是一样的。要告诉 X 服务器这些字体的存在，你需要在 XF86Config 文件中添加一行（XFree86 version 3 在/etc/中，version 4 在/etc/X11/中）：

```
FontPath "/usr/X11R6/lib/X11/fonts/URW/"
```

在命令行，你可以写入：

```
% xset fp+ /usr/X11R6/lib/X11/fonts/URW
```

```
% xset fp rehash
```

5.5.2 TrueType Fonts

与 XFree86 捆绑在一起的默认字体比实际应用所需的字体要少的多。而且字体显示得比较难看。幸运的是 XFree86 允许配置 TrueType。

XFree86 4.0 已经内建支持了 TrueType 字体。有两种不同的模块能够启用这个功能。在这个例子中使用 " freetype " 这个模块，因为它与其他的字体是兼容的。要启用 freetype 模块，只需要在你的/etc/X11/XF86Config 文件中加入下面这行。

```
Load " freetype "
```

你需要运行额外的 TrueType 字体服务器。Xfstt 通常被用作这个目的。要在你的 FreeBSD 系统上安装 xfstt，可以从 `/usr/ports/x11-servers/Xfstt` 中安装相应的 port。你应当为你的 TrueType 字体创建一个目录，然后拷贝所有的 TrueType 字体到这个目录。记住你不能直接从 Macintosh 电脑上提取字体；他们必须是 Unix/DOS/Windows 格式的。一旦你已经拷贝了许多文件到这个目录，你需要使用 `ttmkfdir` 命令来创建一个 `fonts.dir` 的字体文件，以便 X 字体引擎知道你已经安装了这些新文件。在 `/usr/ports/x11-fonts/ttmkfdi r` 中，有一个 FreeBSD port。

```
# cd /usr/X11R6/lib/X11/fonts/TrueType
# ttmkfdi r > fonts.dir
```

现在你需要把你的 TrueType 目录添加到你的字体路径中。这样做最简单的方法是增加下面一行到你的 `~/.xinitrc` 文件。

```
% xset fp+ /usr/X11R6/lib/X11/fonts/TrueType
% xset fp rehash
```

就是这样。现在 Netscape, Gimp, StarOffice 和其他的 X 应用程序已经记住了你安装的 TrueType 字体。一些很小的字体和一些很大的字体现在看起来已经很好了。有一点需要记住：XFree86 现在不支持 anti-aliased 字体。在高分辨率下可能问题不大，但当与 MacOS 或 Microsoft Windows 相比较时仍然相差较大。

5.5.3 Anti-Aliasing 你的字体

启动 version 4.0.2 版，XFree86 支持 anti-aliased 字体。当前，绝大多数软件还没有升级到支持这个新的功能。然而，Qt 已经能够支持了；所以如果你运行 XFree86 4.0.2（或更高版本），Qt2.3（或更高版本）和 KDE，所有你的 KDE/Qt 应用程序都能使用 anti-aliased 字体。

要配置 anti-aliasing，你需要创建一个文件 `/usr/X11R6/lib/X11/XftConfig`。需要对这个文件作几个比较高深的配置；这节只描述最简单的情况。

第一，你需要告诉 Xserver 你要的 anti-aliased 字体。你只需要在每个字体目录中键入这一行：

```
dir "/usr/X11R6/lib/X11/Type1"
```

然后，同样的包含其他字体的目录（如 URW, truetype 等）。Anti-aliasing 可以理解为可升级的字体（基本上，Type1 和 TrueType），所以不包括 bitmap 字体目录。你要包含的目录可以在你的 XF86Config 文件中注释掉。

接下来，你不要使用 anti-alias 的普通大小的文本。（Anti aliasing 使得字体边缘有些模糊，这样增加了非常小的文本的可读性，然后从大文本字体中删除 "staircases"，但如果使用普通的文本，可能会引起眼疲劳。）要将 9 到 13 号之间的字体排除在外，包括这些行：

```
match

    any size > 8

    any size < 14

edit

    anti alias = false;
```

当你打开 anti-aliasing 时，也可能会发现一些字体之间的间距比较混乱。这似乎是 KDE 的一个问题。要修复这个问题需要确保每个字体之间的间距保持在 100：加入下面这些行：

```
match any family == "fixed"      edit family += "mono";

match any family == "console"    edit family += "mono";
```

（这里把其它普通的修复的字体作为“mono”），然后加入：

```
match any family == "mono" edit spacing = 100;
```

如果你想要使用字体，你可以将后面的行改成这样：

```
match any family == "mono"      edit family += "LuciduxMono";

match any family == "Lucidux Mono"    edit family += "LuciduxMono";

match any family == "LuciduxMono"    edit family += "Lucidux Mono";
```

最后，你需要允许用户通过他们每个人的 `.xftconfig` 文件把命令添加到这个文件。加入下面一行：

```
include f ~/.xftconfig"
```

就这么简单；`anti-aliasing` 在你下次启动系统时就能用了。然而，注意你的程序必须知道如何使用它。现在，`toolkit Qt` 已经可以做到，所以整个 KDE 环境能够使用 `anti-aliased` 字体了；对于 `gtk+` 需要加一些补丁，所以如果用加过补丁的 `gtk+` 来编译，那 GNOME 环境和 Mozilla 也能够使用 `anti-aliased` 字体。对于 FreeBSD 和 XFree86 来讲，`Anti-aliasing` 还是比较新的，相信不久会有越来越多的应用程序来支持它。

5.6 X 显示管理

5.6.1 概要

X 显示管理器 (XDM) 是一个 X 视窗系统用来登陆的管理器的可选项。这个在几种情况下是有用的，包括小“X Terminal”，桌面，大网络显示服务器。既然 X 视窗系统不受网络和协议的限制，那对于通过网络连接起来的运行 X 客户端和服务端的不同机器，就会有很多的可配置项。提供了一个选择连接哪个显示服务器的图形接口，只要键入登陆用户名和密码就可以进行验证信息。

你也可以把 XDM 想象成与 `getty` 工具提供了同样功能。它可以完成系统的登陆任务，然后为用户运行一个管理器。接下来 XDM 就等待这个程序退出，发出信号用户已经登陆完成，应当退出屏幕。这时，XDM 就可以为下一个登陆用户显示登陆和选择屏幕。

5.6.2 使用 XDM

XDM 后台程序在 `/usr/X11R6/bin/xdm` 中。你可以在任何时候用 `root` 来运行这个程序，在本地机器上，它将启动管理 X 的画面。如果你要 XDM 每次在启动时在后台运行，一个简单的办法是在 `/etc/ttys` 中加入一个入口。有关这个文件的更多的格式和使用方法，可以看看第 16.3.2.1 节。在每个运行 XDM 程序的虚拟终端上，在默认的文件中有这样一行：

```
ttyv8 "/usr/X11R6/bin/xdm -nodaemon" xterm off secure
```

默认情况下，这个入口是关闭的，要启用它，你需要把 `field 5` 从 `off` 改为 `on`，然后重新启动 `init`。第一个 `field`，这个程序将管理的终端的名称，是 `ttyv8`。这意味着 XDM 将

运行在第 9 个虚拟终端上。

5.6.3 配置 XDM

XDM 的配置目录是在 `/usr/X11R6/lib/X11/xdm` 中。在这个目录中，你会看到几个用来改变 XDM 行为和外观的文件。你会找到这些文件：

文件	描述
Xaccess	Client authorization ruleset.
Xresources	Default X resource values.
Xservers	List of remote and local displays to manage.
Xsession	Default session script for logins.
Xsetup_*	
xdm-config	Global configuration for all displays running on this machine.
xdm-errors	Errors generated by the server program.
xdm-pid	The process ID of the currently running XDM.

当 XDM 运行时，在这个目录中有几个脚本和程序可以用来设置桌面。在下面的几节中，我将简要地描述这些文件的用途。这些文件的更精确的语法和用法在 xdm 的联机手册中将有详细描述。

默认的配置是一个长方形的登陆窗口，上面有机器的名称，“Login:”和“Password:”的命令登陆口。如果你想设计你自己个性化的 XDM 屏幕，这是一个很好的起点。

5.6.3.1 Xaccess

XDM 连接的用来控制显示的协议叫做 X Display Manager Connection Protocol (XDMCP)。这个文件基本上是用来从远程机器上控制 XDMCP 的连接。默认的，它允许任何客户端连接，但你将看到这没有用，因为默认的文件不会侦听到远程连接。

5.6.3.2 Xresources

这是一个默认的用来显示选项和登陆屏幕的应用程序文件。这就是你能够定制登陆程序的外观的地方。这个格式与 XFree86 文档中描述的应用程序文件是一样的。

5.6.3.3 Xservers

这是一个选择者应当提供选项的远程屏幕的列表。

5.6.3.4 Xsession

这是一个用户登陆后运行的针对 XDM 的默认脚本。通常，每个用户在 `~/.xsessionrc` 中将有一个定制脚本。

5.6.3.5 Xsetup_*

这些文件包含了在显示选项或登陆接口之前自动运行的脚本。这是一个每个显示都要用到的脚本，`Xsetup` 后面会跟一个本地显示的数字。典型的，这些脚本将在后台运行一个或两个程序如 `xconsole`。

5.6.3.6 xdm-config

这个文件包含了在可用于每个显示的默认应用程序的形式中的设置。

5.6.3.7 xdm-errors

这个文件包含了运行的 X servers 的输出。如果由于某种原因，XDM 正运行的一个显示被挂起，这是一个查看错误信息的好地方。这些信息也会被写到用户的 `~/.xsession-errors` 文件中。

5.6.4 运行一个网络显示服务器

对于其它客户端来说，为了连接你的显示服务器，你将必须编辑访问控制规则，启用连接侦听。默认的这些被设置成比较保守的值，这样比较安全一点。要让 XDM 来侦听连接，先要在 `xdm-config` 文件中注上一行：

```
! SECURITY: do not listen for XDMCP or Chooser requests
```

```
! Comment out this line if you want to manage X terminals with xdm
```

```
DisplayManager.requestPort: 0
```

然后重新启动 XDM。记住默认应用程序的文件以“!”字母开始，不是“#”。在这后面，你需要设置严格的访问控制。看看在 `Xaccess` 文件中的例子，可以参考 `xdm` 的联机手册。

5.6.5 替换 XDM

有几个替换默认 XDM 程序的方案。它们是，KDM(上一节已经描述过了)。它能让用户很容易地选择它们的窗口管理器来登陆。

5.7 桌面环境

这节将描述 FreeBSD 的 X windows 环境下使用的不同桌面。一个“desktop environment”可能仅仅是一个简单的窗口管理器，也可能是一个象 KDE 或 GNOME 这样的完整的桌面应用程序。

5.7.1 GNOME

5.7.1.1 有关 GNOME

GNOME 是一个用户界面友好的桌面环境，能够使用户很容易地使用和配置他们的电脑。GNOME 包括一个面板（用来启动应用程序和显示状态），一个桌面（存放数据和应用程序的地方），一套标准的桌面工具和应用程序，和一套与其他人相互协同工作的规则。其他操作系统的用户在使用 GNOME 提供的图形驱动的环境时会觉得很好。

5.7.1.2 安装 GNOME

要从网络安装 GNOME，只要键入：

```
# pkg_add -r gnome
```

如果你想从 GNOME 的源代码亲自编译构建，可以使用 ports 树：

```
# cd /usr/ports/x11/gnome
```

```
# make install clean
```

GNOME 一旦被安装好，我们必须使用 X Server 来启动 GNOME 以代替默认的窗口管理器。如果你已经定制好了文件 .xini trc，你必须将启动当前窗口管理器的那行替换为 /usr/X11R6/bin/gnome-session。如果你不想添加任何特殊的配置文件，只需简单地键入：

```
# echo "/usr/X11R6/bin/gnome-session" > ~/.xini trc
```

就是这样简单。键入 startx，你就进入了 GNOME 桌面环境。

注意：如果你运行了一个象 XDM 这样的管理器，就不能这样做。而是，你用同样的命令创建一个可执行文件 `.xsession`。先编辑你的文件（如果它存在的话），然后用 `/usr/X11R6/bin/gnome-session` 替换已存在的窗口管理命令；

```
# echo "#!/bin/sh" > ~/.xsession

# echo "/usr/X11R6/bin/gnome-session" >> ~/.xsession

# chmod +x ~/.xsession
```

另一个选项是在登陆时配置你的显示管理器允许你选择窗口管理器的；KDE2 的那节会解释如何使用 `kdm` 来做，可以看 **KDE 显示管理器** 那节。

5.7.2 KDE2

5.7.2.1 有关 KDE2

KDE 是一个容易使用的现代的桌面环境。KDE 有很多很好的特性：

1. 一个美丽的现代的桌面
2. 一个集合了完美网络环境的桌面
3. 一个集成的帮助系统允许你方便，高效地帮助你使用 KDE 桌面和它的应用程序。
4. 所有的应用程序都是所见即所得。
5. 标准的菜单和工具栏，键盘布局，颜色配置等。
6. 国际化：KDE 可以使用超过 40 种语言。
7. 集中的一致化的桌面驱动配置。
8. 许多有用的 KDE 应用程序。

KDE 是一个以 KDE 的 KParts 为基础的办公软件套件，它包括电子表格，幻灯片，日程管理和新闻等软件。KDE 也包含一个叫做 **Konqueror** 的浏览器。如果需要了解更多信息，请访问 KDE 的网站。

5.7.2.2 安装 KDE2

在写这篇文章的时候，kde2 的软件包还没有发布。不过，没有关系！在 ports 树中已经包含了它的源代码。要安装 KDE2，只要这样做：

```
# cd /usr/ports/x11/kde2
```

```
# make install clean
```

这个命令可以取得所有必须的文件，配置和编译 KDE2，安装应用程序，然后自己进行整理。现在，你将必须告诉 X server 启动 KDE2 以代替默认的窗口管理器。只要这样做：

```
# echo "/usr/X11R6/bin/startkde" > ~/.xini trc
```

现在，无论你什么时候进入 X-windows，**KDE2** 就是你的桌面了。

（注释：如果你通过一个象 xdm 这样的显示管理器登陆，它就会不工作。这样的话，你有两个选择：创建一个文件 .xsession，用/usr/X11R6/bin/startkde 命令代替 **gnome-session** 命令；或者配置你的显示管理器允许你在登陆时选择桌面。下面，将解释如何在 kdm 中这样做）

5.7.3 有关 KDE2 的更多细节

现在 **KDE2** 已安装在你的系统中了，你将发现你能从它的帮助系统中学习到很多东西。Windows 或 Mac 用户会觉得很自然。有关 **KDE** 的最好的参考资料是它的在线文档。**KDE** 拥有它自己的 web 浏览器 **Konqueror**，还有很多应用程序和丰富的文档。

5.7.3.1 KDE 显示管理器

如果你是一个多用户系统的系统管理员，你可能想要给用户一个图形登陆界面。你可以使用 xdm。然而，也包含一个叫 **KDM** 的登陆界面，它看起来很吸引人，还包含很多登陆选项（**KDE2**，**GNOME** 或其它）。

运行 **KDE2** 的控制面板，需要是 root 用户。

注意：通常情况下，以 root 用户来运行 X 环境是很不安全的。你应当以普通用户来使用窗口管理器，打开一个终端窗口 xterm 或 **KDE's** konsole，用 su 来成为 root 用户，然后键入 kcontrol。

点击左面标记“system”的图标，然后选择“Login manager”。在右面，你将看到几个配置选项，KDE 联机手册会详细介绍的。点击“sessions”。根据你要安装的窗口管理器或桌面环境，你可以在“New type”中键入他们的名字然后加入它们。你也可以用同样方法定制其他菜单。完成之后，键入“Apply”按钮，推出控制中心。

要确信 KDM 理解你上面的标签(KDE, GNOME 等)的意思，你需要编辑一些文件：xdm 也是一样。在你的终端窗口，作为 root 用户，要编辑/usr/X11R6/lib/X11/xdm/Xsession 文件。

```
case $# in
1)
    case $1 in
        failsafe)
            exec xterm -geometry 80x24-0-0
            ;;
        esac
    esac
```

你需要在这部分加一行。你给出的标签是 KDE2 和 GNOME，就象下面这样：

```
case $# in
1)
    case $1 in
        KDE2)
            exec /usr/X11R6/bin/startkde
            ;;
        GNOME)
```

```
exec /usr/X11R6/bin/gnome-session

;;

fail safe)

exec xterm -geometry 80x24-0-0

;;

esac

esac
```

你需要在 `/usr/X11R6/lib/X11/xdm/Xsetup_0` 中加入下面这行：

```
/usr/X11R6/bin/kdmdesktop
```

现在，你只需要确信在系统下次启动时，`kdm` 会被启动。要知道如何做，可以读读有关 `xdm` 的章节，只要将 `xdm` 的程序替换成 `kdm` 的程序就可以完成同样的功能。就这么简单。你的下一个登陆屏幕将有一个漂亮的外观和许多菜单。

5.7.3.2 Anti-aliased 字体

启动 version 4.0.2，**XFree86** 可以通过它的 "RENDER" 扩展来支持 anti-aliasing，Qt 支持这个扩展。在 5.5.3 节描述到了这些。所以如果你正运行的是更高版本的软件，anti-aliasing 也可以在你的 **KDE2** 桌面上使用。只需要到你的 **KDE2** 菜单，到 Preferences -> Look and Feel -> Style，然后点击 "Use Anti-Aliasing for Fonts and Icons"。就这样。如果你运行的 Qt 应用程序不是 **KDE** 的一部分，你可能需要在启动程序前设置 `QT_XFT` 的环境变量为 true。

5.7.4 XFCE

5.7.4.1 有关 XFCE

XFCE 是被 GNOME 使用的以 `gtk+ toolkit` 为基础的。它看起来很像 UNIX 商业系统开发的 CDE。它的很多特性包括：

1. 一个简单，容易使用的桌面。
2. 完全通过鼠标用拖放和按键来控制等。

3. 与 CDE 相似的主面板，菜单，applets 和 launchers。
4. 集成的窗口管理器，文件管理器，声音管理器，GNOME 应用模块，和其他一些。
5. 有很多主题界面。
6. 快速，高效。

更多有关 XFCE 的信息可以在 XFCE 网站中找到。

5.7.4.2 安装 XFCE

安装 xfce，执行下面的命令：

```
# pkg_add -r xfce
```

另外，你也可以直接从源代码构建。

```
# cd /usr/ports/x11-wm/xfce
```

```
# make install clean
```

所有必需的源代码包会被自动地找到，构建和安装，然后建构区域会被自动清理干净。

现在，你要告诉服务器在下次启动时执行 XFCE。执行下面的命令：

```
# echo "/usr/X11R6/bin/startxfce" > ~/.xinitrc
```

接下来就是执行 X，XFCE 将成为你的桌面。

(注意：如果你通过象 xdm 这样的显示管理器来登陆，你需要创建一个 .xsession 文件，使用 /usr/X11R6/bin/startxfce 命令；或者配置你的显示管理器允许在启动时选择一个桌面)。