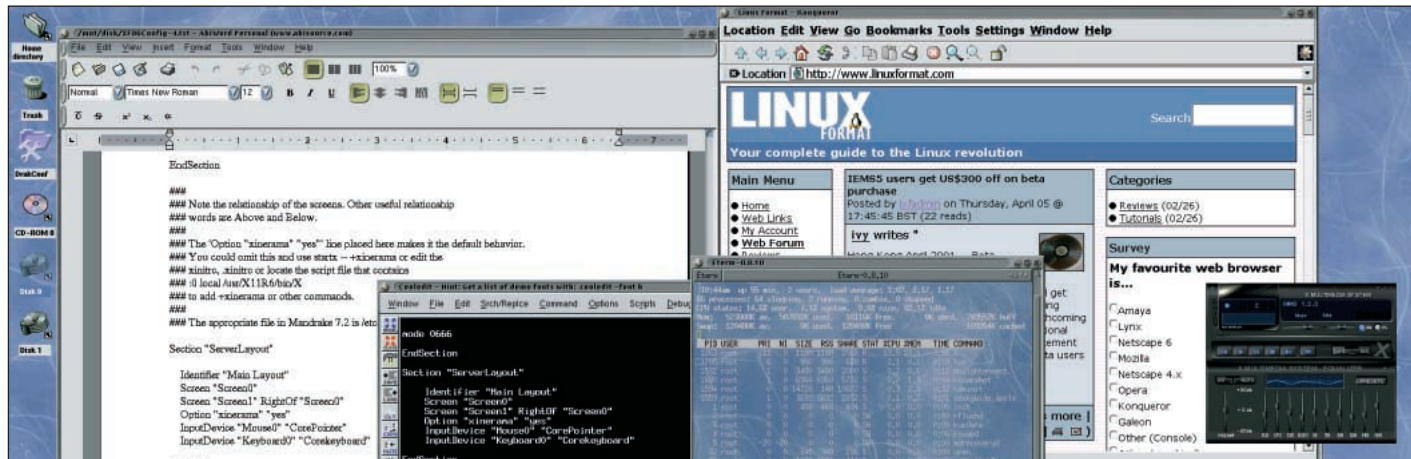


Tutorial Desktop MultiHead



GET MORE X

Multi-Head Displays

Hoyt Duff discovers a use for that old video card and monitor left in the closet after your last upgrade.



Useful info

There are sample XF86Config files on your system located in `/usr/X11R6/lib/X11/`.

If you need to know which driver is being used by your video card, look in `/usr/X11R6/lib/X11/Cards+`

Ever since last summer's release of XFree86 version 4, there has been much talk about one of its most intriguing features, but little written about it. In previous issues of *Linux Format*, we have titillated you with screenshots of a panoramic desktop. Now, we set the record straight and show you how to configure your hardware and edit your config files allowing you to use multiple monitors at one time, a situation called multi-head or, in its most common incarnation, dual-head.

Multi-head is nothing new: Macintosh had it in the late 1980's and Windows 98 provided it to MS users. Of course, prior to version 4, XFree86 was capable of running multiple servers and X has always been network aware, allowing the remote display of an X-client on any display on a network. Tools like *xmx*, *vnc*, *x2vnc* and *xmest* have made virtual multiple displays possible.

Actually, multi-head for Unix has been around awhile, but only available in commercial (and expensive) X-servers from Metro-X and XiGraphics. Until now, multi-head was out of the reach of the loyal, but poor, Linux user. Regrettably, this is one of those areas where our Mac and Windows brethren can justifiably gloat as multiple monitor support is much more mature in those operating systems, hence much easier to configure with a wider range of supported hardware. For the Linux faithful (as well as any other OS that can run XFree86 such as BSD), support for multiple monitors is now available; we just have to work a little harder at it – you should be used to that by now.

Why Bother?

You are likely to be asking yourself what justification there is to use multiple displays. Here are a few reasons:

1 One-Upmanship

Be the first one in your LUG to demonstrate a dual-head configuration and you will achieve instant high status – it's simply that cool an achievement. Just ignore the smarmy Mac users in the group for now.

2 Maximum Screen Real Estate

All fun aside, this is a more practical reason. Look at it in terms of square inches and cost. You want a bigger desktop: there are two options: (1) purchase a 21" monitor or (2) purchase two 17" monitors and a second video card. The maths is left as an exercise for the student (meaning we didn't want to do it), but the display area of a 17" monitor is slightly more than half that of a 21" monitor while the cost is significantly less than half (We know your thoughts here: "Hey – can I buy THREE monitors?") and a decent new video card is about the price of a 17" monitor. It's possible (although not as desirable as we'll see later) to re-use that older video card and monitor you have in the closet from your last upgrade. Essentially, for the same cash outlay (or less) you can have a bigger display and some bragging rights.

The low-budget approach can be surprisingly affordable as you re-use that old PCI video card and 14" monitor.

3 Able to View More Windows at Once

This is the reason you use if you absolutely must justify the expense to your boss (or spouse or parents): you can be more productive. Multi-head allows you to have more large windows open at the same time without having to page through them, saving valuable time:

- Web developers can easily compare HTML pages side-by-side in full size browsers; create pages in an editor in one head, view them in the second head at the same time.
- Writers can have multiple documents or revisions open and simultaneously viewable.
- Programmers can view their development environment on one head and view the results of their efforts on the other.
- Sysadmins can display additional windows of monitoring programs, all visible at once.
- CAD/CAM, graphics designers, video editors and others can also benefit from multiple screens available for viewing at the same time.

We already hear the protests from hard-core Linux users: "That's what virtual desktops are for!" And they are correct, but with multi-head, all windows are viewable at the same time, on

the same desktop. Additionally, you can still use virtual desktops to have multiple virtual displays.

Don't knock it...

Like many things in life, multi-head displays are something you should experience before you dismiss the idea. As with anything Linux, there are choices to be made as multi-head works in two basic ways. With the release of XFree86 version 4.x, Linux users can now enable multiple monitors in two ways: multiple displays (a unique desktop for each monitor) or a single display spread over multiple monitors (known as *xinerama*). In either case, you are required to have a separate video card and monitor for each display, or 'head', that you desire.

Multiple independent displays

When XF86Config is properly configured for multiple video cards, this is the default behaviour. You can move the mouse across all monitors. If one were to simply run `startx`, the selected desktop would appear in the primary display and X would likely appear bare in the other monitors. Why? Because many window managers are not multi-head aware, so they don't know to start a wm for those displays. Multi-head aware (to some extent) window managers include *Enlightenment*, *WindowMaker* and *Sawfish*. The multi-head capabilities of wms are not well documented.

To use `.xinitrc` to start non-multi-head aware window managers in separate heads, edit it as follows:

```
.xinitrc
twm :0.0 &
fvwm :0.1 &
exec fvm2 :0.2
```

substituting your preferred wm as appropriate.

You may need to specify the display when launching programs, for example:

```
xterm -display :0.1 &
```

launches an `xterm` in the second head.

If you are able, tell your window manager to ignore program supplied position hints. A multi-head aware wm will attempt to intelligently place the windows, otherwise, they tend to display in the 'centre' and straddle the heads.

Xinerama

The *xinerama* extension to XFree86 version 4.x allows a single, large desktop to be displayed across multiple heads. It is video

driver independent, meaning that if your video card is supported, it will most likely work with *xinerama* (no guarantees, though). Assuming that you would normally use a 1024x768 screen size, *xinerama* in a dual head display would allow a screen of 2048x768 and display it across two monitors sitting side by side. Your mouse cursor will travel between displays and windows can be dragged across displays.

Ideally, that part of the desktop under the physical monitor edges would be resistant to windows expanding beyond them and dialogues would pop up centred in the physical monitor. We say 'ideal' because the limiting factor to all this is the window manager. The placement of windows is controlled by the window manager, not X, so the wm needs to be *xinerama*-aware.

Hardware required

Linux has always been finicky about hardware and X is no different; multi-head even more so, and there are a couple of issues. A big issue for some people is that 3D support is not well-supported in multi-head mode or even available in most instances. This is a problem with the drivers, not the hardware, and until it becomes a priority for the development team (or you, and then you contribute your work to the community), greatly improved support is not likely to appear; a workaround is easy to implement (see below). Some video cards are not supported, and some are not suitable for use with multi-head; the reason having to do with how the video BIOS is initialised.

Common wisdom has it that Matrox cards are better for multi-head displays than others (although many others work) for the reason that some have been designed with multiple displays in mind, especially their G450 and G400 multi-head cards, and driver support is good. We found that it was necessary to update the video BIOS of an older Matrox Mystique card we purchased second-hand in order for it to behave properly. Additionally, this particular card had a place for a switch on the board to enable BIOS support for multi-head, only the switch was missing; an examination of the circuit and the use of a soldering iron solved our problem, but details on that kind of hardware hacking is beyond the scope of this article. Just be aware that your inability to get a particular video card working in multi-head may be the card itself. This is a common experience as we have observed that many of those attempting to enjoy the multi-head experience do so with video cards and monitors found in the junkbox, those having been discarded during previous

Glossary

Head – a monitor.

Display – an environmental variable representing the name of the X server. Used synonymously, but incorrectly, with monitor. The displays are named using this convention:

localhost:0.0 – first display
localhost:0.1 – second display
and so on. The word 'localhost' is replaced with an IP address for remote displays.

Multi-head – using more than one monitor at the same time.

Dual head – the most common example of multi-head using two monitors.

xinerama – a special case of multi-head display where a single virtual desktop is created that spans all the displays.

Which video card?

There are several options for identifying the video cards in your computer. All work pretty much the same; our preference is XFree86 -scanpci. The goal is to discover the Bus ID so that this information can be used in the XF86Config-4 file.

```
Terminal - Terminal
File Sessions Options Help
lspci
00:00.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:01.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:02.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:03.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:04.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:05.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:06.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:07.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:08.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:09.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:0a.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:0b.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:0c.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:0d.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:0e.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:0f.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:10.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:11.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:12.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:13.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:14.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:15.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:16.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:17.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:18.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:19.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1a.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1b.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1c.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1d.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1e.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.1 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.2 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.3 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.4 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.5 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.6 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.7 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.8 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.9 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.a Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.b Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.c Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.d Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.e Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.f Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.1 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.2 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.3 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.4 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.5 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.6 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.7 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.8 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.9 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.a Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.b Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.c Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.d Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.e Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.f Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.1 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.2 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.3 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.4 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.5 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.6 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.7 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.8 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.9 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.a Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.b Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.c Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.d Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.e Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.f Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.1 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.2 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.3 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.4 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.5 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.6 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.7 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.8 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.9 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.a Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.b Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.c Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.d Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.e Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.f Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.1 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.2 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.3 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.4 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.5 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.6 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.7 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.8 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.9 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.a Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.b Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.c Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.d Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.e Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.f Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.1 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.2 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.3 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.4 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.5 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.6 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.7 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.8 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.9 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.a Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.b Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.c Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.d Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.e Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.f Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.1 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.2 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.3 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.4 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.5 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.6 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.7 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.8 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.9 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.a Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.b Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.c Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.d Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.e Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.f Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.1 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.2 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.3 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.4 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.5 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.6 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.7 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.8 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.9 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.a Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.b Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.c Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.d Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.e Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.f Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.1 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.2 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.3 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.4 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.5 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.6 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.7 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.8 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.9 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.a Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.b Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.c Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.d Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.e Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.f Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.1 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.2 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.3 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.4 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.5 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.6 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.7 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.8 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.9 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.a Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.b Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.c Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.d Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.e Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.f Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.1 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.2 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.3 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.4 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.5 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.6 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.7 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.8 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.9 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.a Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.b Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.c Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.d Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.e Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.f Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.1 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.2 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.3 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.4 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.5 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.6 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.7 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.8 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.9 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.a Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.b Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.c Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.d Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.e Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.f Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.1 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.2 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.3 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.4 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.5 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.6 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.7 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.8 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.9 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.a Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.b Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.c Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.d Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.e Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.f Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.1 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.2 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.3 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.4 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.5 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.6 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.7 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.8 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.9 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.a Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.b Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.c Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.d Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.e Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.f Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.1 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.2 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.3 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.4 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.5 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.6 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.7 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.8 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.9 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.a Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.b Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.c Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.d Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.e Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.f Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.1 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.2 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.3 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.4 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.5 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.6 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.7 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.8 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.9 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.a Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.b Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.c Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.d Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.e Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.f Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.1 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.2 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.3 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.4 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.5 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.6 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.7 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.8 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.9 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.a Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.b Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.c Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.d Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.e Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.f Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.0 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.1 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.2 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.3 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.4 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.5 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.6 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.7 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.8 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.9 Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.a Host bridge: Intel 82443AGX (Pentium III) PCI-to-AGP bridge
00:1f.b
```

Tutorial Desktop MultiHead

Dead space

If you attempt to enable *xinerama* using a 21" monitor and that 14" SVGA monitor it originally replaced, you may encounter "dead space" when you configure different screen resolutions for each of your monitors.

Let us assume that you configure a resolution of 1024x768 on the 17" monitor and 800x600 on the smaller one (because that's as large as it allows) and the monitors are sitting side by side. The virtual desktop *xinerama* creates is now 2048x768, but the smaller monitor doesn't cover all of its 1024x768 area; the "L" shaped space is the "dead space". A window manager may unwittingly open a window or display a dialogue within that space, effectively out of your reach. Different sized monitors and *xinerama* don't mix well without some annoyance.

hardware upgrades. Monitors, to a lesser extent, may be culprits if they don't support the resolutions and scan rates you would like (or need) to use.

See 3D

As mentioned before, there is a workaround for using multi-head displays without letting all that costly 3D power go to waste. Fortunately, multiple XF86Config configuration files can be used with XFree86. By default, XFree86 will look for the default configuration files in /root, /etc/X11 and a few other places. Version 4 of XFree86 will look for a file named XF86Config-4 first and, if it can't find one, uses XF86Config. This can cause a problem if you have both versions of XFree86 installed on your system and aren't aware of this feature. Actually, XFree86 is so flexible, it allows you to use any config file located anywhere if you want to specify it with the **-xf86config** command followed by the full path and filename of the appropriate configuration file. This command can be passed to XFree86 through startx by using the double dash command, so an example would look like:

```
startx --xf86config /etc/X11/XF86Config.dual head
```

Since both version 3 and version 4 can co-exist on the same system, you can run X in single head mode using the accelerated 3D graphics driver under XFree86-3.3.x to play games and then restart X with XFree86-4.x and your 2D multi-head configuration to actually get some work done.

Carpe Deim

Configuring multiple monitors is not easy, but not overly complex either when you possess an understanding of your hardware and the intricacies of X. The main stumbling block for ease of use is better driver support and better software support. These will come in time, especially if demand for this feature increases. We have found older 4Mb PCI video cards at reasonable prices and even the comparatively newer 16Mb Voodoo3 PCI cards are affordable (and even the Voodoo2 cards can be used for the second head). Of course, the dual-head Matrox cards are more expensive, but enjoy Linux driver support. Concurrently, 17" monitors are becoming more affordable and even the prices of LCD displays are coming within the grasp of the average person; 15" monitors are very affordable.

As long as you have safely saved your original working XF86Config file, you have little to lose by trying. Dive right in and expand your window on the world.

Multi-Head 12-Step Program

1 Make certain that you will be booting into a runlevel where X is not automatically started or running. This makes life simpler for a while if you normally start that way.

Just edit the line in /etc/inittab to read: **id:3:initdefault:** or, you could use **linux 3** at the LILO prompt.

2 Backup and uniquely label any working XF86Config files you

```
File Sessions Options Help
root@wind /root]# cat /proc/pci
PCI devices found:
Bus 0, device 0, function 0:
Host bridge: Acer Labs M1541 Aladdin V (rev 4),
Slow device, Master Capable, Latency=64,
Non-prefetchable 32 bit memory at 0xe0000000 [0xe0000000],
Bus 0, device 1, function 0:
PCI bridge: Acer Labs M5043 AGP (rev 4),
Slow device, Master Capable, Latency=64, Min Gnt=8,
Bus 0, device 2, function 0:
USB Controller: Acer Labs M5237 USB (rev 3),
Medium device, Fast back-to-back capable, IRQ 9, Master Capable, Latency=64,
Max Lat=255,
Non-prefetchable 32 bit memory at 0xd8000000 [0xd8000000],
Bus 0, device 3, function 0:
Bridge: Acer Labs M7101 PMU (rev 0),
Medium device, Fast back-to-back capable,
Bus 0, device 7, function 0:
ISA bridge: Acer Labs M1533 Aladdin IV (rev 195),
Medium device, Master Capable, No bursts,
Bus 0, device 9, function 0:
Ethernet controller: Realtek 8029 (rev 0),
Medium device, IRQ 9,
I/O at 0xb000 [0xb000],
Bus 0, device 10, function 0:
Multimedia audio controller: Ensoniq Unknown device (rev 2),
Vendor id=1274, Device id=5880,
Slow device, IRQ 5, Master Capable, Latency=48, Min Gnt=12, Max Lat=128
I/O at 0xb400 [0xb400],
Bus 0, device 11, function 0:
VGA compatible controller: Matrox Mystique (rev 2),
Medium device, Fast back-to-back capable, IRQ 10, Master Capable, Latency=64,
Non-prefetchable 32 bit memory at 0xd8000000 [0xd8000000],
Prefetchable 32 bit memory at 0xe0000000 [0xe0000000],
Bus 0, device 15, function 0:
IEEE Interface: Acer Labs M5229 Dipeo (rev 193),
Medium device, Fast back-to-back capable, Master Capable, Latency=32,
Min Gnt=2, Max Lat=255,
I/O at 0xb000 [0xb000],
Bus 1, device 0, function 0:
VGA compatible controller: 3DFx Unknown device (rev 1),
Vendor id=21a, Device id=5,
Fast device, Fast back-to-back capable, IRQ 11,
Non-prefetchable 32 bit memory at 0xd8000000 [0xd8000000],
Prefetchable 32 bit memory at 0xe0000000 [0xe0000000],
I/O at 0xb000 [0xb000],
```

You can use 2D and 3D features by creating multiple XF86config files.

```
File Sessions Options Help
root@wind /root]# lspci
0000:00 Host bridge: Acer Laboratories Inc. [ALL] M1541 (rev 04)
0001:00 PCI bridge: Acer Laboratories Inc. [ALL] M5043 (rev 04)
0002:00 USB Controller: Acer Laboratories Inc. [ALL] M5237 USB (rev 03)
0003:00 Bridge: Acer Laboratories Inc. [ALL] M7101 PMU
0007:00 ISA bridge: Acer Laboratories Inc. [ALL] M1533 PCI to ISA Bridge [Aladdin IV] (rev c3)
0009:00 Ethernet controller: Realtek Semiconductor Co., Ltd. RTL-8029 (rev 02)
000a:00 Multimedia audio controller: Ensoniq 5880 AudioPCI (rev 02)
000b:00 VGA compatible controller: Matrox Graphics, Inc. MGA 1064SG [Mystique] (rev 02)
000f:00 IEEE interface: Acer Laboratories Inc. [ALL] M5229 IEEE (rev c1)
0100:00 VGA compatible controller: 3DFx Interactive, Inc. Voodoo 3 (rev 01)
```

You'll need to configure each of your monitors.

may already have, then delete all XF86Config files from the default location (usually /etc/X11) and shut down the computer.

3 Install all the cards you will be using in the AGP or PCI slots they will ultimately reside in. For those with AGP cards, the system BIOS usually has a selection to make the AGP card the primary video card (the one initialised first). Depending on the other cards you use, you may have to play with this setting.

4 Boot the system, observing which monitor/video card is initialised for the console display – that's your primary display. The other displays will remain uninitialised and blank. If everything is blank, you have a card conflict (they are fighting over which will be the primary display. Remove cards one at a time until it works.

From the command line, using your choice of **cat /proc/pci**, **lspci** or **XFree86 -scanpci**. The last one is recommended

you should be able to determine the Bus ID of each video card. Write down the Bus ID's of the cards.

5 Shut the computer down, remove all but the primary video card and restart the computer.

Using whatever tool you desire (we suggest 'XFree86 -

Web Resources

XFree86 – <http://www.xfree86.org>

Xi Graphics – <http://www.xig.com>

Metro-X – <http://www.metrolink.com>

x2x – <ftp://ftp.digital.com/pub/Digital/SRC/x2x/>

VNC – <http://www.uk.research.att.com/vnc/>

x2vnc – <http://www.hubbe.net/~hubbe/x2vnc.html>

xmx – <http://www.cs.brown.edu/software/xmx/>

xmove – <http://www.ensta.fr/internet/unix/misc/xmove.html>

The Xinerama HOWTO –

<http://www.linuxdoc.org/HOWTO/Xinerama-HOWTO.html>

XFree86 Documentation – <http://www.xfree86.org/support.html>

The XFree86 HOWTO – <http://www.linuxdoc.org/HOWTO/XFree86-HOWTO/index.html>

Remote X Applications HOWTO –

<http://www.linuxdoc.org/HOWTO/mini/Remote-X-Apps.html>

Securing X Windows –

<http://ciac.lnl.gov/ciac/documents/ciac2316.html>

Multi-monitor methods

Xnest

This tool allows you to run an x-server inside a window in an already running x-server. While mainly used for testing, this can be pretty interesting since you can launch xnest and start a complete desktop in it. Since we already have a :0 server running, let's start Xnest as the second server with: `Xnest :1` and then open an xterm in it with: `xterm -display :1` From that xterm, you may launch any app you desire or even open a new window manager and desktop. Essentially, this is what vnc is doing.

X2X

While not technically multi-head, x2x allows a computer running X to control, with its mouse and keyboard, another computer running X. To be useful, the monitor for the second computer should be on the desk next to your monitor. The source code is available from <ftp://ftp.digital.com/pub/Digital/SRC/x2x/>.

Assuming 'wind' is the current machine and 'marvin' is the target machine whose physical screen is to the left, the command line would be:

```
x2x -to marvin:0 -from wind:0 -west
```

X2VNC

If you happen to have both a Windows box and a separate Linux box, you can keep your desktop less cluttered and have control of both machines from the keyboard and mouse attached to your Linux box using x2vnc, a variant of VNC. The vncserver must be running

on the Windows machine. You can obtain x2vnc as source only. Follow its README instructions to compile it and to launch it. Be aware that running most games on the Windows box will break the network connection to x2vnc. Use smbmount to make the Windows box's partitions available to your Linux box. The homepage for vnc is <http://www.uk.research.att.com/vnc/> and the homepage for x2vnc is <http://www.hubbe.net/~hubbe/x2vnc.html>.

The vnc server must be running on the Windows machine, and assuming 'wind' is the Linux machine and 'marvin' is the Windows machine whose physical screen is to the left, the command line would be:

```
x2vnc -west marvin:0
```

Xmove

Are you running multi-head, but not *xinerama*, and want to move a running program to another display? That can be done with *xmove*, a pseudo-server. The way it works is that it starts a server on :1 (your first server is on :0). Then, any client sent to :1 will display on :0, but can be moved to another networked computer. Not quite as easy as a **Send Window To** dropdown menu, but useful if you have the need. The homepage is at <http://www.ensta.fr/internet/unix/misc/xmove.html>; and the source is available from <ftp://ftp.cs.columbia.edu/pub/xmove>. After gunzipping and un-tarring the source, you need to use *xmkmf* and make to compile the binary. Take a careful look at the README files before beginning. After compiling, you

will have two binaries, *xmove* and *xmovectrl*. The first starts the server, the latter moves the window to a different machine.

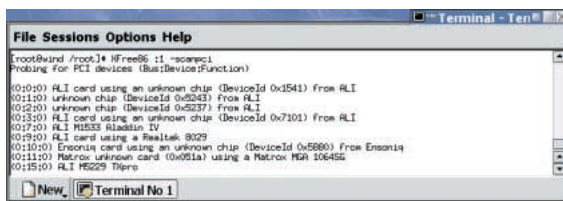
Start *xmove* from a console with: *xmove* and you now have a display :1 on your computer. Launch an xterm to it with: `xterm -display :1` and an xterm will appear on your desktop.

XMx

Developed at Brown University, *mxm* is 'an X protocol multiple-server'. In essence, it shares an X display among a group, for example students in a classroom. The latest snapshot is from two years ago, but you could still find it useful in a classroom setting. The homepage is at <http://www.cs.brown.edu/software/mxm>. Binaries for Solaris, AIX and Linux are available as well as source. The package includes a selection of practical examples in the README files.

Xhost +

The easiest way to permit a computer to receive X clients over a network is to use *xhost +* to allow it. However, the connection is very insecure. A detailed description of more secure approaches can be found in the Remote X Apps mini-HOWTO at <http://www.linuxdoc.org/HOWTO/mini/Remote-X-Apps.html> as well as Securing X Windows at <http://ciac.llnl.gov/ciac/documents/ciac2316.html>. This site also includes a nice overview of X as well.



-scanpci is the easiest way of working out which cards are correctly installed and which need more work.

configure' for version 4 configuration), create a working XF86Config file (for both versions 3 and 4, if you have a need for it) for a single head display.

Since you are working as root, the test config file will be found in /root, your home directory, upon creation. It is a good idea to test the new config file with **XFree86 -xf86config /root/filename**. This starts X bare, without a window manager, so we can determine if X is configured properly without having to worry if the window manager is working properly. You'll see a grey screen with a large X cursor. After you get a working configuration, you can see how it works with the window manager by using: `startx --xf86config /root/XF86Config`.

Once it's working to your satisfaction, save this configuration file with a unique filename, then delete any XF86Config files remaining to start clean next time.

6 Repeat for each video card, one at a time (remember that AGP BIOS setting, if necessary!). The goal here is to confirm that each video card and monitor works and initialises at the resolution and colour depth you desire for the multi-head system (we suggest 1024x768 at 16bpp as a start – you can experiment later, after you get it working). Save each config file with a unique filename as before.

7 Now reinstall all the video cards where you had them before, reboot and run **XFree86 -configure** one last time. Move the config file to /etc/X11 and name it XF86Config-4. For now, make certain it's the only XF86Config file there.

8 Edit the file and ascertain that the Bus IDs are correct for each video card, that all cards and monitors have a unique section. Verify that the section describing the mouse is correct; we had difficulty getting the config program to recognise our USB mouse and had to cut and paste a working description from a working config file. Examine the section that describes the physical arrangement of the monitors (how they sit in front of you).

9 Run X naked with the command **XFree86**. Hopefully, X will start in the primary display and you will then see the secondary displays initialise and display the classic grey X screen. Your mouse pointer should move across all the heads.


If there is a problem, some help can be found by examining the error messages written to /root/.xsession-errors. If there's nothing there, check /var/log.

Compare the information in the config file with that in a working config file and adjust as necessary.

10 At this point, you should be able to start X with the default window manager using the `startx` command (the specifics as to configuring the startx process for a specific wm and a specific distro are beyond the scope of this article, thank heaven). You should now see multiple independent desktops, but may not see a desktop in each monitor if the wm is not multi-head aware.

11 If you want *xinerama*, test it with: `startx - +xinerama`

Remember to have all the colour depths the same. Screen resolutions can be different to accommodate different size monitors, just be aware of dead space.

12 You can *xinerama* the default by adding: **Option "xinerama" "yes"** to the ServerLayout section of XF86Config-4. 

Which X?

XFree86 comes in two versions: version 3 and version 4. Some modern distributions install both, some only one. There are advantages to running version 3 if you need accelerated 3D support in some instances or have an older video card not supported in version 4. Version 4 is required to support a multi-head setup. Which do you have installed? Running **XFree86 -version** from a command line will disclose the version of X installed on your machine.