

NMR NOTES #9

Configuring the X-WIN32 Xserver Software

In order to use the X-WIN32 X-server emulator software with VnmrX on the Sun workstations, it must be configured properly. Generally, what we will do is use the rexec (remote execution) command to run a script that will start the OpenLook Window Manager on the PC. A standard script should reside in your home directory named .xlogin. Running this script will start the OpenLook Window Manager (olwm) on PC and make your standard menu available by clicking the right hand mouse button. Before starting Xwin32, it will be necessary to run X-Util first. This is the utility program that will allow you to define login hosts and sessions. When you first start X-Win32, you get a full screen unix window. Click the box in the upper left corner to pull down the operations menu.

To configure X-Win32, run the X-util program, either from the program group or from the X-Win pull down menu.

The Options menu includes eight different items, and these should need to be defined or set only once. The three items we are interested in are Window mode, xhosts, and mouse. The Window mode permits definition of the Window to be used for running X-applications. The X-WIN default is to use the Windows 3.1 window manager instead of the OpenLook window manager, but this seems to be less convenient. Under Window mode, select 'Single Window Mode' in order to enable a separate window (called virtual root) that will be controlled by the olwm. Using a video card with 1024 by 768 resolution graphics, choose a screen size of 1016 by 732 to avoid having the borders scrolling off of the screen. The width and height values may also be left blank to give the maximum screen size possible without panning for the current display driver. A few pixels around the edges are used to define the Windows 3.1 window that contains the X-server virtual window. Reducing the size of the X-window by this amount will prevent the screen jumping when you move the cursor to the edges. It is possible to define a logical screen that is larger than the physical screen. In this case, moving the mouse cursor to the edge of the physical screen will cause the logical screen to scroll, permitting you to see all the contents. You may also chose a screen size that is smaller than the screen resolution, and in that case the virtual root window will simply be smaller than the full screen. In either case, do not select the option to enable screen 0.1, since this is incompatible with OpenLook 3.0.

If you prefer to use the Windows 3.1 window manager instead of the OpenLook Window Manager, do not select the single window mode. In this case, you will not have a choice about screen size, panning, or enabling screen 0.1. A choice to enable panning is possible, permitting a window that is off the screen to be panned on-screen. When using the Windows 3.1 window manager, you cannot start the openlook window manager, so it is necessary to run a different script to start the unix system. There is no script currently provided to do this, but the supplied .xlogin script can be modified readily. All that is necessary is to comment out (# in first column) the lines with the olwm and the xsetroot commands. Add a new line at the end of the file to start the xterm program:

```
exec $OPENWINHOME/bin/xterm &
```

When this script is run as part of a session login, it should open an xterm window which is connected to the unix workstation. You may then run any valid system commands, and the x-server output will be directed to the Windows 3.1 window manager. Use vn & to start the VnmrX program. You will then be able to use the xterm window for normal unix commands and have VnmrX available as well. It is also possible to run vnmr directly from the .xlogin script by adding the following:

```
exec vn &
```

This will run the VnmrX program without starting an xterm window first. Both of these commands can be included at the end of your .xlogin file, in either order.

The xhosts option permits defining hosts permitted to access X-WIN. If this list is empty, any host will be permitted to connect to the X-win server. If any hosts are defined, then only those hosts in the list will be permitted to connect. A host must be in the list and have a + sign in front of it in order to run the screen lock program xlock.

The mouse option permits defining the characteristics of your mouse. Set this appropriately.

The Edit Session menu will allow you to create new login sessions or redefine old login sessions. These are named sessions that may be recalled later. When defining a login session, you can specify the login type as rsh, rexec, or XDMCP. The rsh and rexec work the same way, except that rexec will ask for a password before executing the script. The rsh command does not require the password, but does require that your PC be listed in a .hosts file in your home directory. The rsh version is slightly less secure, and rexec should be the option normally used. The XDMCP is a completely different protocol that is not currently running on the Sun systems, so this can not be selected and used. There is also provision when editing a session to define that session to be an autostart session that will start automatically when you launch X-Win. The final part of defining a session is to define the host machine, the user, and the command to run. The host will be one of the Sun workstations, nmrsun1 or nmrsun2. The user name will be your login name on that system. The command to execute should be ' csh .xlogin \$MYIP:0.0 '. The \$MYIP will be replaced by X-WIN with the network address of your PC. This is passed as an argument to the .xlogin command so that it knows where to send the X-server graphics.

This is all that is necessary to configure and run the X-WIN x-server emulator. The other configuration that may be necessary is to set up an ftp server running in background. In order to print spectra to a local laserjet printer, the laserjet graphic file is sent to the PC in background with an ftp connection. The X-WIN server does not include a built in facility for accepting ftp connections, so it will be necessary to start a separate ftp daemon as a server running in a separate window to accept these file transfers. I have a qvt package available that includes an ftp daemon for the PC. When a plot is sent to the local laserjet, it is sent by ftp straight to LPT1. Depending upon how your printer is buffered, it may cause the ftp to time out and be incomplete. If this is the case, I can install a print buffer program to cache the information to disk or memory and spool it out to the printer. This would also permit remapping LPT1 to what ever port actually contains the printer, if different. This print buffer will not operate under Windows 95.