Information for Solaris

David Holland, modified by Marc Aurele La France

2001 October 01

1. What is XFree86

XFree86 is a port of X11R6.5.1 that supports several versions of Unix. It is derived from X386 1.2 which was the X server distributed with X11R5. This release consists of many new features and performance improvements as well as many bug fixes. The release is available as a source code distribution, as well as binary distributions for many architectures.

The sources for XFree86 are available by anonymous ftp from:

ftp://ftp.XFree86.org/pub/XFree86/current

Solaris binaries for XFree86 are available for anonymous ftp at the same address. Currently, two binary distributions are available: one for Solaris 8 x86, the other for previous Solaris x86 releases. They are *not* interchangeable. No binary distribution is available for Solaris/SPARC as it is still under development.

2. The VT-switching sub-system in Solaris x86

The virtual terminal sub-system is a undocumented, and unsupported feature of Solaris x86. Therefore if you use virtual terminals, you do so at **YOUR OWN RISK**.

Virtual terminals are not available in Solaris SPARC, and their availability has been removed in Solaris8 x86.

When available, the virtual terminals of Solaris work basically the same way as most other SVR4 VT sub-systems. However, there are a number of limitations documented below.

First, if you are running a Solaris 2.4 x86 system, and you want VT's, you will have to create the necessary devices first, so become root.

Verify the chanmux device driver's major number is 100:

```
# grep -i chanmux /etc/name_to_major
chanmux 100
#
```

If the number after 'chanmux' is anything but 100, I would suggest you immediately abort your attempt to create virtual terminals, and learn to live without them.

However, if it is 100, then as root type the following commands to create the maximum allowable number of virtual terminals.

```
# cd /dev
# mknod vt01 c 100 1
# mknod vt02 c 100 2
# mknod vt03 c 100 3
# mknod vt04 c 100 4
# mknod vt05 c 100 5
# mknod vt06 c 100 6
# mknod vt07 c 100 7
```

There is no need for a reconfiguration boot.

Secondly, for both 2.1, and 2.4 x86 systems, add a few lines to the `inittab` to enable logins on them.

(**Note**, do **NOT** make a mistake here, you could lock yourself out of the system)

```
---------------------->Snip Snip<-----------------------------------------------
v1:234:respawn:/usr/lib/saf/ttymon -g -h -p "`uname -n` VT01 login: " -T AT386 -d /dev/vt01 -l console
v2:234:respawn:/usr/lib/saf/ttymon -g -h -p "`uname -n` VT02 login: " -T AT386 -d /dev/vt02 -l console
v3:234:respawn:/usr/lib/saf/ttymon -g -h -p "`uname -n` VT03 login: " -T AT386 -d /dev/vt03 -l console
v4:234:respawn:/usr/lib/saf/ttymon -g -h -p "`uname -n` VT04 login: " -T AT386 -d /dev/vt04 -l console
---------------------->End Here<-------------------------------------------------
```

These four lines enable four VT's on Alt-SysReq-F1 through Alt-SysReq-F4.

Then execute the command '`init q`' to immediately enable the virtual terminals.

The keys used for VT switching are as follows:

`Alt-SysReq-F1` through `Alt-SysReq-F7` enable VT screens 1-7 respectively (if the VT is active).

`Alt-SysReq-n` enables the next active VT screen.

`Alt-SysReq-p` enables the previous active VT screen.

`Alt-SysReq-h` returns to the console.

If you are using virtual terminals, you must leave at least one free for use by the Xserver.

Limitations of the virtual terminal sub-system under Solaris x86:

There are only a total of 8 available VT's (7 normal VT's + 1 console) not the usual 15. If you have all 8 allocated, and you attempt to allocate a additional VT you will panic the system. (This bug is worked around in the Solaris XFree86 Xserver.)

From a programming stand point, they work pretty much as documented in the AT&T Unix System V/386 Release 4 Integrated Software Development Guide, however a number of `ioctl()` calls are broken.

3. Notes for building XFree86 on Solaris

1. Both GCC, and ProWorks are supported by XFree86. The minimum recommended GCC release is 2.7.2. Some earlier GCC's are known to not work and should be avoided.

   You should also make certain your version of GCC predefines 'sun'. If needed edit `/usr/local/lib/gcc-lib/*/*/specs`, and modify the `*predefines:` line.

   On SPARCs, regardless of the compiler you use, ensure it generates 32-bit binaries. At this time, 64-bit binaries will probably not work.

2. Also on SPARCs, you will need to set BuildXFree86OnSparcSunOS to YES in ~xc/config/cf/host.def. Otherwise, you can only build the old deprecated Xsun* servers.

3. A Threaded Xlib compiled with GCC has subtle problems. It'll work 98% of the time, however clients will occasionally exhibit strange hangs. Most notably image viewers such as xv-3.10 exhibit this problem.

   It is recommended that you set ThreadedX in `~xc/config/cf/host.def` to NO, if you are using GCC. ProWorks does not have this problem. Whether this behaviour still exists with newer GCC's has not been verified.

4. To build XFree86 with GCC you need gcc and (optionally) c++filt from GNU binutils. Don't install gas or ld from GNU binutils, use the one provided by Sun.

   You might need to setup a /opt/SUNWspro/bin directory containing symbolic links named `cc`, `CC`, and `c++filt` pointing respectively to the actual `gcc`, `g++` and `c++filt` commands.

5. If you are using ProWorks to compile the XFree86 distribution, you need to modify your PATH appropriately so the ProWorks tools are available. Normally, they should be in `/opt/SUNWspro/bin`

6. You **MUST** put `/usr/ccs/bin` at the front of your PATH. There are known problems with some GNU replacements for the utilities found there. So the `/usr/ccs/bin` versions of these programs must be found before any possible GNU versions. (Most notably GNU '`ar`' does not work during the build).

4. Notes for running XFree86 on Solaris

1. Depending on the release or architecture of Solaris you are running, you might need to install an OS driver for an aperture device.

   Under Solaris x86 2.5 and later, there's a system driver (`/dev/xsvc`) that provides this functionality. It will be detected automatically by the server, so you don't need to install the aperture driver.

   For older Solaris x86 and for any SPARC release, the source for this driver is included in `xc/programs/Xserver/hw/xfree86/etc/apSolaris.shar` of the source distribution. This file can usually also be found in the `/usr/X11R6/lib/X11/etc` directory when XFree86 is installed. Building, and installing the driver is relatively straight forward. Please read its accompanying README file.

2. If you have not made the Virtual Terminal devices, you will need to specify the terminal device to run the Xserver on. The correct device is vt00 so your `xinit` command would look like so:

   ```
   xinit -- vt00
   ```

   If you have made the virtual terminal devices you do not need to specify the VT to run the Xserver on.

3. For Solaris you will probably want to set your LD_LIBRARY_PATH to `/usr/X11R6/lib:/usr/openwin/lib:/usr/dt/lib`. Including `/usr/X11R6/lib` in your LD_LIBRARY_PATH is probably not necessary, however it doesn't hurt. :)

   Including `/usr/openwin/lib` in the LD_LIBRARY_PATH is recommended because some Sun supplied binaries were not compiled with LD_RUN_PATH set properly at compile time.

   Motif and CDE applications may require `/usr/dt/lib` in your LD_LIBRARY_PATH too.

4. Xqueue is **NOT** supported under Solaris. The includes necessary for Xqueue are available, however the driver does not seem to be in the kernel. (Go figure)
5. If you want to use xdm with Solaris, extract the files from the shar file in `/usr/X11R6/lib/X11/etc/XdmConf.svr4` into a temporary directory. The `README` file tells where the individual files need to be installed. Be sure to read through each file and make any site-specific changes that you need.

5. Known bugs, and work arounds with Solaris

1. The Solaris 2.1 for x86 OpenWindows filemgr does not work against a X11R5 Xserver, it probably will also not work against a X11R6 Xserver. Attempting to 'Drag and Drop' a file causes the filemgr to abort with an 'X error'

   Solaris x86 2.4 does not have this problem.

   There is no known work around.
2. The SPARC port is still quite new, so instability is to be expected (and reported!). It might even have broken some aspects of the x86 port.
3. On SPARC, xload should be updated to deal with both 32-bit and 64-bit SunOS kernels. It currently is compiled as a 32-bit binary and cannot handle a 64-bit kernel.

6. Bug Notification

Bug reports should be sent to <XFree86@XFree86.org>.

CONTENTS

1. What is XFree86 1
2. The VT-switching sub-system in Solaris x86 1
3. Notes for building XFree86 on Solaris 2
4. Notes for running XFree86 on Solaris 3
5. Known bugs, and work arounds with Solaris 4
6. Bug Notification 4

$XFree86: xc/programs/Xserver/hw/xfree86/doc/sgml/Solaris.sgml,v 1.6 2004/11/26 19:09:23 tsi Exp $