Information for Solaris for x86 Users

David Holland 13 December 1994

Contents

1 What is XFree86 3.1.1 1

2 Solaris for x86, versions on which XFree86 3.1.1 has been tested 1

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

4 Notes for building XFree86 on Solaris x86 3

5 Notes for running XFree86 on Solaris x86 4

6 Known bugs, and work arounds with Solaris x86 4

7 Bug Notification 5

1 What is XFree86 3.1.1

XFree86 3.1.1 is a port of X11R6 that supports several versions of Intel-based 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 source patches against the X Consortium code, as well as binary distributions for many architectures.

The sources for XFree86 are available by anonymous ftp from:

ftp.XFree86.org:/pub/XFree86/current `(ftp://ftp.XFree86.org/pub/XFree86/current)`

Solaris binaries for XFree86 are available for anonymous ftp from:

ftp.XFree86.org:/pub/XFree86/current/binaries/Solaris
`(ftp://ftp.XFree86.org/pub/XFree86/current/binaries/Solaris)`*abode.ccd.bnl.gov:/pub/XFree86*
`(ftp://abode.ccd.bnl.gov/pub/XFree86)`

Many thanks to **Jerry Whelan** for making the space available for these binaries.

2 Solaris for x86, versions on which XFree86 3.1.1 has been tested

XFree86 3.1.1 has been actively tested on:

- Solaris 2.1 for x86
- Solaris 2.4 EA1 for x86
- Solaris 2.4 for x86 FCS

And is expected to run under:

- Solaris 2.4 EA2 for x86

3 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**.

The virtual terminals of Solaris work basically the same way as most other Intel based 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.

First 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.

4 Notes for building XFree86 on Solaris x86

1. The majority of all modifications you will need to make are now in `~xc/config/cf/xf86site.def`, except one. You will need to set OSName, and OSMinorVersion correctly in `~xc/config/cf/sun.cf`.

   Under Solaris 2.1 for x86, `sun.cf` needs to read (near the top of the file)

   ```
   #ifdef SVR4Architecture
   #ifdef i386Architecture
   #define OSName                  SunOS 5.1 x86
   XCOMM operating system:  OSName
   #define OSMajorVersion          5
   #define OSMinorVersion          1
   ```

Under Solaris 2.4 for x86, `sun.cf` needs to read (near the top of the file)

```
#ifdef SVR4Architecture
#ifdef i386Architecture
#define OSName                  SunOS 5.4 x86
XCOMM operating system:  OSName
#define OSMajorVersion          5
#define OSMinorVersion          4
```

2. Both Gcc, and ProWorks are supported by XFree86. Gcc-2.5.8 is suggested, Gcc-2.6.0 is known not to work. You also need to set HasGcc2 correctly in `~xc/config/cf/xf86site.def`, and use CC=gcc on your `'make World'` command.

   You will also need to either create a link from gcc to cc, or comment out the CC=cc line in `~xc/config/imake/Makefile.ini`.

3. If your are using ProWorks to compile the XFree86 distribution, you need to modify your PATH appropriately so the ProWorks tools are available.

4. 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 other possible GNU versions. (Most notably GNU '`ar`' does not work during the build).

5. If you wish to use the "memory aperture" feature of the S3, and Mach32 servers, you need to compile, and install the Solaris x86 aperture driver for memory mapped I/O support. This driver is **REQUIRED** for the P9000 server.

   The source for this driver is included in `~xc/programs/Xserver/hw/xfree86/etc/apSolx86.shar`. Building, and installing the driver is relatively straight forward. Please read its accompanying README file.

   You also need to set HasSolx86apertureDrv to YES in `~xc/config/cf/xf86site.def`.

6. The BOOTSTRAPCFLAGS for Solaris x86 is:

```
"-Dsun -Di386 -DSVR4"
```

5 Notes for running XFree86 on Solaris x86

1. 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.

2. For Solaris you will probably want to set your LD_LIBRARY_PATH to `/usr/X11R6/lib:/usr/openwin/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.

3. 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)

4. 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.

6 Known bugs, and work arounds with Solaris x86

1. Wabi tickles a obscure bug in the Mach32 hardware cursor code. The hardware cursor shows up as a multicolored square while inside of Wabi windows.

   The work around, add:

   ```
   Option "sw_cursor"
   ```

   to you XF86Config.

2. 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 a 'X error'

   Solaris 2.4 does not have this problem.

   There is no known work around.

7 Bug Notification

Bug reports need to be sent to **XFree86@Xfree86.org**, or posted to the comp.windows.x.i386unix newsgroup. Questions or comments about the Solaris support, or the Solaris distribution need to be made to *davidh@use.com*, or *danson@lgc.com.*

```
$XConsortium: SOLX86.sgml,v 1.2 95/01/16 13:17:48 kaleb Exp $
$XFree86: xc/programs/Xserver/hw/xfree86/doc/sgml/SOLX86.sgml,v 3.3 1995/01/28 16:02:26 dawes Exp $
```