Information for SCO Users

David McCullough 14 December 1994

Contents

1 Binary Distribution

The following files are provided in the binary distribution, where XXX is the new version and NNN is the previous version:

README.SCO

This file.

gunzip.Z

The GNU `uncompress` utility.

***XFXXX-doc.tz**

The XFree86 specific documentation.

XFNNN-XXX.tz

Files to upgrade from all NNN versions to XXX. This file may not be present if the upgrade is too substantial (ie., X11R5 -¿ X11R6).

XFXXX-Mono.tz

The Mono server

XFXXX-VGA16.tz

The 16 colour VGA server

XFXXX-SVGA.tz

The Super VGA server

XFXXX-S3.tz

The S3 server

There will most likely be more servers than listed here. You should select the server you need. Check the directory listing for any other servers that may be there.

***XFXXX-cfg.tz**

The local configuration files for `xdm/fs/xinit`.

***XFXXX-bin.tz**

The `bin` directory, contains most executables.

***XFXXX-lib.tz**

The shared and unshared libraries.

***XFXXX-fnt1.tz**

`75dpi` and `misc` fonts.

XFXXX-fnt2.tz

`100dpi` and `Speedo` fonts.

***XFXXX-inc.tz**

The X11 include files.

XFXXX-man.tz

The formatted man pages.

XFXXX-lkit.tz

The server link kit (all drivers + PEX).

XFXXX-util.tz

Lots of PD utilities provided as is.

XFXXX-pex.tz
All files relating to PEX including libraries and header files. The LinkKit is required to obtain servers capable of running PEX.

To obtain a minimum XFree86 installation you will require the archives marked with a '*' above, the server binary best suited to your machine and optionally "`gunzip.Z`". All the files are compressed with "`gzip`" except of course "`gunzip.Z`" which is compressed using the conventional `compress` program.

To install the XFree86 binaries just follow these steps.

1. Obtain the files you require. as root,
2. create a directory `/usr/X11R6`, permissions `755` should do nicely.
3. cd `/usr/X11R6`
4. extract the archives, for example:

   ```
   gunzip < XFXXX-bin.tz | tar xvf -
   ```

5. if you have installed man pages see the later section on setting up man pages.
6. Look through `/usr/X11R6/lib/X11/doc/INSTALL`, especially section 2 on configuring and using XFree86. This should allow you to get a server up and running. Before starting the server check in the later section 3, in this document, to see if there are any system requirements you have to make for the server to operate correctly.

If you intend using the programs from the utilities archive then you should be aware that they are provided as is. They are not part of the XFree86 distribution nor are they part of the standard MIT distribution. Every attempt is made to ensure that they work to a usable degree but they are not perfect. All programs in the utilities archive come with a manual entry (which is also in the utilities archive). Information on the author(s) can usually be located in the man page or by running the executable.

2 Source Distribution

The SCO port comes as part of the standard XFree86 distribution. Consult the XFree86 `README` for more information on the location of sources.

3 Before Running XFree86

Before you can run XFree86 with shared libraries enabled you must up the shared library count in the kernel. This is only required for 3.2.2 systems. To accomplish this log in as root and run:

```
cd /etc/conf/cf.d
./configure SHLBMAX=6
cd ../bin
./idbuild
```

The SCO `xterm` terminfo description is not compatible with the `xterm` in the R5 distribution.

To use a Bus/Keyboard or PS2 mouse you should configure the mouse drivers under SCO as above using '`mkdev mouse`'. You may then use the `OsMouse` option in your XF86Config to specify that XFree86 should use the SCO mouse drivers. To do this, set the `Protocol` to "`OsMouse`" in the Pointer section of your XF86Config file. You can also use "`OsMouse`" for your serial mouse, especially if you are having trouble getting your mouse to work using the XFree86 mouse drivers.

If you are using an Accelerated Localbus/EISA card you may wish to install the "`dmmap`" driver. This driver allows the server to use a linear address mapping in high memory when accessing the card. The dmmap driver comes in two forms, a binary archive which is supplied with the full binary distribution and a source archive which is located in `/usr/X11R6/lib/X11/etc`. Both distributions provide instructions for installation/removal within the `README file`.

4 Switching Consoles

XFree86 uses similar console switching keys as the SCO R4 and R5 servers. That is, `Ctrl-PrntScr` takes you to the next console along from the one X is running on. If this is the last console it will take you to console 1. Those interested in modifying the console switching should look in `xc/programs/Xserver/hw/xfree86/common/xf86Events.c`.

5 Setting up Man Pages

After compiling the tree, or after installing the binary distribution you can get man to recognise the XFree86 man pages by adding `/usr/X11R6/man` to the `MANPATH` in `/etc/default/man`, the line should look similar to:

```
MANPATH=/usr/man:/usr/X11R6/man
```

This allows all users to view the X man pages. You may change your own `MANPATH` environment variable if you do not want everyone to access the man pages.

By default the man pages are compressed using "`pack`" under 3.2.2 and "`compress`" under 3.2.4 to conserve space. If you do not want to compress the man pages change `CompressManPages` to `NO` in your `xf86site.def`. Those using the binary distribution can use "`unpack`" or "`uncompress`" as appropriate to uncompress the man pages.

6 Using SCO binaries/servers.

XFree86 will accept connections from SCO binaries (R3 upwards) and the SCO R5 server will also accept connections from XFree86 binaries. This means you may mix and match the two if you have ODT. For example you may still use the Motif window manager (mwm) if you prefer.

7 Compiling XFree86 under 3.2.2

Currently XFree86 is configured to install under `/usr/X11R6`. The binary distribution will be configured to go into this directory. If you are compiling your own XFree86 there is no reason why this cannot be changed. Check in the X11 `config/README` on how to change your `ProjectRoot`.

The defaults are set such that you should only require minimal changes to the `site.def` or the `xf85site.def` file. The `BOOTSTRAPCFLAGS` do most of the work for you. The defaults will use gcc 2 or above and they also compile for shared libraries and setup formatted man pages using compression. The most likely Parameters you will be interested in changing are:

1. What Fonts etc get compiled.
2. What gets installed (`XdmConfig, FSConfig`, ...)

Use gcc, preferably 2.5.8 or later but an earlier version may still do the job. Configure it with "`./configure i386-*-sco`".

Make sure you do the fixincludes for gcc.

In order to compile the current XFree86 version for SCO you will have to create the following header files in the appropriate places.

A) create a `/usr/include/sys/uio.h` containing the following:

```
#ifndef _XFREE86_UIO_
#define _XFREE86_UIO_ 1

#ifndef __IOVEC__
#define __IOVEC__ 1
typedef struct iovec {
    caddr_t iov_base;
    int iov_len;
} iovec_t;
#endif /* __IOVEC__ */

#endif /* _XFREE86_UIO_ */
```

B) create a `/usr/include/net/errno.h` containing

```
#ifndef _XFREE86_ERRNO_
#define _XFREE86_ERRNO_ 1
#include <errno.h>
#endif /* _XFREE86_ERRNO_ */
```

C) create a `/usr/include/sys/bsdtypes.h` which is empty for now.D) create a < `sys/time.h` > under the gnu include directory containing the following (this protects the system header file from multiple inclusions) You do not have to change the system header file!

```
#ifndef _SYS_TIME_H_
#define _SYS_TIME_H_ 1
#include {$<$} /usr/include/sys/time.h {$>$}    #endif /* _SYS_TIME_H_ */
```

You also need to obtain a copy of `libcrypt` or the international crypt supplement from SCO. The supplement is available on the SCO ftp site, *sosco.sco.com* (The name may have changed) as an SLS, lng225. This is required for the correct compilation of "xdm".

To compile from here, follow the XFree86 `README`, basically after obtaining X11r5, applying the XFree86 patches and completing the above changes you should only need to run:

```
make BOOTSTRAPCFLAGS="-DSYSV -Di386 -DSCO" World
```

The install script `mit/util/scripts/bsdinst.sh` will not set 's' bit permissions when installing over the top of a previously installed XFree86. To fix this apply the following patch from the directory containing the '`mit`' directory. This is personal taste only, if you are installing through symbolic links then leave this one alone.

```
----------8<-----------------8<------------------------------------
*** mit/util/scripts/bsdinst.sh%        Tue Mar  2 14:27:16 1993
--- mit/util/scripts/bsdinst.sh Tue Mar  2 14:27:17 1993
***************
*** 150,154 ****
  # This rm is commented out because some people want to be able to
  # install through symbolic links.  Uncomment it if it offends you.
! # rm -f $dst/$srcbase
  (cd $srcdir ; $installcmd -f $dst $flags $src)

--- 150,154 ----
  # This rm is commented out because some people want to be able to
  # install through symbolic links.  Uncomment it if it offends you.
! rm -f $dst/$srcbase
  (cd $srcdir ; $installcmd -f $dst $flags $src)
----------8<-----------------8<------------------------------------
```

8 Compiling XFree86 under 3.2.4

Compiling gcc2.5.8 is accomplished after doing the following.

```
./configure i386-*-sco3.2v4.2
```

DO NOT USE "`./configure i386-*-sco`" it will not work.

Make sure you run the "`make fix-includes`" after you have installed.

It is recommended that you use a gcc compiled without any changes to the value of `__STDC__`. Some people may have compiled gcc by adding the define "`-DSTDC_VALUE=0`" to their `CPP_PREDEFINES` flag. If you have done this and are able to recompile gcc without this change then you should. Otherwise refer to the end of this section for instructions on how to fix the XFree86 tree so that it will still compile.

Compiling XFree86,

1. Check that you have a "`sys/uio.h`" as there have been reports of systems which have it and systems which don't! If you already have one it need not be changed. Otherwise see the 3.2.2 section above on how to create one.

2. Check the paragraph on `site.def/xf86site.def` in the compiling for 3.2.2 section.

3. Add a `-DSCO324` to the `BOOTSTRAPCFLAGS` when you do the make World.

   ```
   make BOOTSTRAPCFLAGS="-DSYSV -Di386 -DSCO -DSCO324" World
   ```

   Under 3.2.4 several different options become activated.

   - Compress is used instead of pack for compressing the man pages.
   - The default compile line will contain `-D_NO_PROTOTYPE` instead of `-DNO_PROTOTYPE` as this is what 3.2.4 expects.

4. If you have problems with `mkshlib` reporting an error like

   ```
   mkshlib: : cannot exec /bin/sh
   ```

   when compiling `libX11`, then the following should fix it,

   ```
   rm -f lib/X/libX11.a
   env - make
   ```

   The error is related to a lack of environment space.

9 Compiling with __STDC__=0

DO NOT make these changes unless you have gcc configured with `STDC_VALUE=0`. You can test for this by compiling this small program. If it prints out `STDC_VALUE=1` do not make these changes. If it prints out `STDC_VALUE=0` then you will have to make the following changes.

```
main() { printf("STDC_VALUE=%d\n", __STDC__); }
```

So now you have a gcc compiled with `STDC_VALUE=0` which cannot be recompiled with a `STDC_VALUE=1`. To compile XFree86 with this compiler you will have to edit `mit/config/x386.cf`. Look for the pattern `GCC_STDC_VALUE` and replace the 1 by a 0. For example the line

```
#define GCC_STDC_VALUE 1
```

should become,

```
#define GCC_STDC_VALUE 0
```

It is recommended to use gcc compiled without the `STDC_VALUE=0` for building XFree86. It is, however, possible with the above changes. These changes may become obsolete or invalid in a later version of XFree86.

10 Relevant Documentation

Some relevant documentation for SCO Users and Developers can be found in the following files.

README

the standard XFree86 `README` (`/usr/X11R6/lib/X11/doc`)

README.SVR3

Although a lot of this readme is based on Interactive a substantial proportion is still relevant.

All of the VGA/Config documentation.

`/usr/X11R6/lib/X11/doc/VideoModes.doc` and the `README` files for particular video cards.

11 Known Problems

- After running the server you may see some strange characters in your input to the shell. This is due to some unprocessed scancodes and is of no concern. This will be fixed in a future release.
- Not all of the applications in `/usr/X11R6/bin` have been debugged.
- Shared libraries are not implemented in XFree86 3.1 yet.

12 Trouble Shooting

Problem:

Server aborts with the error message, "Failed to map video display memory".

Causes:

Your system does not have the correct `/etc/conf/pack.d/cn/class.h`, You can confirm this by editing the file and looking for the string "SVGA", if it is not there then you should re-install this file from the "Extended Utilities" diskettes provided with your OS. If this is not possible then installing the "`dmmap`" driver from the distribution may allow the server to operate correctly.

Problem:

`xf86install` does not work.

Causes:

You should not be running `xf86install` when using the XFree86 server under SCO. It is used for Interactive (ISC) installations.

Problem:

The server starts but the screen is not aligned correctly or is shaky and impossible to view.

Causes:

This is most likely due to an incorrect `XF86Config` setup. Look for the files `README.Config` `VideoModes.doc` (in `/usr/X11R6/lib/X11/doc` with the binary distribution). These files explains how to fix up your video modes.

Problem:

1. Can only run a limited number of xterms.
2. `xterm` does not work but other programs like `xclock` do work.

Causes:

Not enough or no pseudo ttys devices are present on your system. Depending on which version of SCO you are using you will have to do different things.

3.2.2
`cd /etc/conf/cf.d; ./configure` and increase the number of pseudo ttys. Rebuild the kernel. "`ls -al /dev/ttyp0 /dev/ptyp0`" to get the major numbers for the slave and master drivers. Use `mknod` to create the extra devices using the naming schemes ttyp$0-9a-f$ and ptyp$0-9a-f$.

3.2.4
Run "`mkdev ptty`" and increase the number of ptty's.

Problem:

When running curses/termcap applications in an `xterm` the output gets corrupted especially when scrolling.

Causes:

1. You are running an original 1.3 distribution of XFree86. Update to the latest version (2.0 or greater).

2. You have resized the window and not ran "`eval 'resize'`" before using your application. The SCO operating system does not support dynamic resizing or xterms fully so this command must be run after resizing an `xterm` in order for curses/termcap applications to operate correctly.

Problem:

1. When starting X it dies with an error "Cannot access a needed shared library".
2. When starting an X application is dies with the above error.

Causes:

1. You do not have the binaries installed in the correct directory. Check that they are in `/usr/X11R6`
2. You have upgraded to a new binary distribution which has a new version of the shared libraries which are not compatible with your old binaries. To fix this you will need to re-install the old shared libraries or recompile your application against the new libraries.

Problem:

My SCO 3.2.4 system fails to build `libXext.a` for shared libraries.

Causes:

This behaviour was observed when compiling with a `STDC_VALUE=0` for gcc. mkshlib was core dumping on building the shared library in this case. Using a new gcc without the `STDC_VALUE=0` or using a 3.2.2 mkshlib appears to fix this problem.

Problem:

When linking against the SCO motif library I get an unresolved external for "`XtDisplayStringConversionWarning`" when using gcc.

Causes:

The SCO library is compiled with limited length identifiers. To work around this add the following code to your application when compiling under XFree86 with gcc and SCO motif.

```
#ifdef SCO
void XtDisplayStringConversionWarnin(dpy, from, toType)
    Display*  dpy;
    String    from;
    String    toType;
{ XtDisplayStringConversionWarning(dpy, from, toType); }
#endif
```

Problem:

When compiling the XFree86 source under 3.2.4, libX11 fails to get built with an error like "`mkshlib :  :  cannot exec /bin/sh`"

Causes:

Read the section on compiling for 3.2.4. To fix this use "`env - make`".

Problem:

The server fails to run and prints out a line similar to:`XFree86: Cannot open /dev/spx for ???? listener: No such file or directory`

Causes:

All SCO unix installations appear to have the Streams pseudo tty driver installed its just not all the devices are present.

1. there should be a `/etc/conf/pack.d/sp` directory,
2. `/etc/conf/sdevice.d/sp` should have a 'Y' in it.
3. You need a file in `/etc/conf/node.d` which contains something like:

```
clone   spx     c   sp
sp      X0S     c   127
sp      X0R     c   126
sp      X1S     c   125
sp      X1R     c   124
sp      X2S     c   123
sp      X2R     c   122
sp      X3S     c   121
sp      X3R     c   120
sp      X4S     c   119
sp      X4R     c   118
sp      X5S     c   117
sp      X5R     c   116
sp      X6S     c   115
sp      X6R     c   114
sp      X7S     c   113
sp      X7R     c   112
```

if you don't have something like this (maybe called "Xsco") then create one and that should fix your problem. As far as I can tell the streams pseudo tty driver should be there.

The simplest way to get the devices if you had to create this file is to rebuild the kernel and the environment. If you don't want to do this then:

```
touch /etc/.new_unix
cd /etc/conf/bin
./idmkenv
```

and try it out.

13 Acknowledgements

Thanks to the Core team for their previous and continuing help with the SCO work. Many thanks to **Stacey Campbell** at SCO for all the advice and insights provided. Thanks to SCO in general for making information available for XFree86 development.

Thanks also to **Peter Eubert** (*peter.eubert@iwb.mw.tu-muenchen.dbp.de*) and **Kent Hamilton** (*kenth@stl.scscom.COM*) for input on compiling under 3.2.4 systems. **Larry Plona** (*faxi@world.std.com*) and **Didier Poirot** (*dp@chorus.fr*) for their input on xdm and 3.2.4 compilation under 3.1. And of course the beta list for its input on everything.

Special thanks to **Jerry Whelan** (*guru@stasi.bradley.edu*) for providing an ftp site for the binary distribution.

```
$XConsortium: SCO.sgml,v 1.4 95/01/27 16:14:31 kaleb Exp $
$XFree86: xc/programs/Xserver/hw/xfree86/doc/sgml/SCO.sgml,v 3.6 1995/01/28 16:02:21 dawes Exp $
```