

# XFree86 für Hurd

MARCUS BRINKMANN

*Auf der letzten freeX-CD befand sich die Binary-Distribution von Debian GNU/Hurd. Sie enthält aber kein X-Window-System.*

*Dieses Manko wird nun mit dieser Heft-CD behoben.*



XFree86 wurde erstmals von Uchiyama Yasushi auf Hurd portiert, damals in der Version 3.3.3.1 [1]. Es gab jedoch einige Ecken und Kanten, die erst mit der Erstellung der Debian-Pakete abgeschliffen wurden. Inzwischen ist sowohl die Version 3.3.6 als auch 4.0.1 [2] nach Hurd portiert und lauffähig, beide Portierungen sind aber noch nicht Bestandteil der offiziellen Releases. Die Einarbeitung der Patches wird im Laufe der nächsten Zeit erfolgen.

## Installation

Zuerst müssen die Pakete in folgender Reihenfolge mit `dpkg -install` oder mit `apt-get` installiert werden. Wenn Sie die aktuellen Entwicklungen des Debian-FTP-Archivs mitverfolgen, kann es sein, daß Sie bereits über ein neueres Hurd-Paket verfügen. In diesem Fall haben Sie auch die `glibc`-Pakete aktualisiert und sollten nicht zur Hurd-Version 20000921 zurückkehren. Neben diesen Paketen werden auch noch einige andere Pakete aus der Standard-Distribution benötigt (zum Beispiel »`cpp`«, »`util-linux`« und »`zlib1g`«). Haben Sie die Anweisungen aus dem letzten Heft befolgt, sind diese anderen Pakete bereits installiert:

```
gnumach 1.2-7
hurd 20000921
xFree86-common 3.3.6-11
xlib6g 3.3.6-10
xbase-clients 3.3.6-10
xfonts-base 3.3.6-2
xfonts-75dpi 3.3.6-2
xfonts-scalable 3.3.6-2
xserver-common 3.3.6-10
```

Als nächstes muß der Server installiert werden. Installieren Sie das entsprechende »`xserver-*`«-Paket für die von Ihnen verwendete Grafikkarte. Im Zweifelsfall sollte das `xserver-vga16`-Paket für die ersten Versuche ausreichen. Beantworten Sie bei der Installation die Frage, ob Sie diesen Server als Default-Server verwenden wollen, mit »`yes`«.

Es wird auch wärmstens empfohlen, einen Window-

Manager und einen Terminal-Emulator zu installieren. Diese befinden sich in den Paketen »`fvwm`« und »`fvwm-common`« sowie »`rxvt`«.

Der Standard-Terminal-Emulator »`xterm`« funktioniert zwar, ist aber fehlerhaft. Laufende Prozesse, die aus diesem gestartet wurden, können nicht mit `[Ctrl]+[C]` oder `[Ctrl]+[Z]` unterbrochen werden, und generell gibt es Probleme mit den Zugriffsrechten auf das Pseudo-terminal. Ebenso funktioniert auch der einfache Window-Manager »`twm`«, aber er muß mit der Option `-force-depends` installiert werden, da das notwendige Paket »`menu`« noch nicht für Hurd verfügbar ist.

Die neue Hurd-Version ist notwendig, um die Maus und Tastatur in X verwenden zu können. Legen Sie zunächst die Gerätedatei für die Tastatur an:

```
cd /dev
./MAKEDEV kbd
```

Dann legen Sie die Gerätedatei für die Maus an. Die dafür notwendigen Optionen hängen vom verwendeten Maustyp ab. Für eine serielle Maus ist folgende Einstellung korrekt:

```
settrans -c /dev/mouse /hurd/mouse -protocol=microsoft \
-devi ce=com0
```

Für eine PS/2-Maus verwenden Sie:

```
settrans -c /dev/mouse /hurd/mouse -protocol=ps/2
```

Die Konfigurationsdatei für den X-Server muß sich in »`/etc/X11/XF86Config`« befinden. Wie X genau konfiguriert wird, entnehmen Sie bitte der Dokumentation von X. Im Regelfall ist es ausreichend, wenn Sie eine vorhandene Konfiguration von einer BSD- oder Linux-Installation verwenden. Nur der Abschnitt mit der Mauskonfiguration muß angepaßt werden:

```
Section "Pointer"
    Protocol      "osmouse"
    Device        "/dev/mouse"
EndSection
```

Eine andere Möglichkeit ist, direkt den Maustreiber im X-Server zu verwenden. Dazu braucht »/dev/mouse« nicht aufgesetzt zu werden. Unterstützt werden zumindest die Protokolle *Microsoft* und *IntelliMouse* mit Device »/dev/com0« (die Gerätedatei »/dev/com0« erzeugen Sie wie »kbd« mit *MAKEDEV*). Die Maus ruckelt bei dieser Variante allerdings sehr stark. Verwendet man als Protokoll »osmouse«, sind nur wenige weitere Optionen unterstützt, unter anderem »Emulate3Buttons«. Bei den anderen Protokollen sollten mehr Optionen zur Verfügung stehen.

Um X starten zu können, muß der Linker die Bibliotheken finden. Unter Linux wird dies durch Erweiterung der Datei »/etc/ld.so.conf« erreicht. Diese wird von Hurd jedoch nicht unterstützt. Bis das ELF-Format um das notwendige *DT\_RUNPATH*-Feld erweitert wird, müssen Sie die Umgebungsvariable *LD\_LIBRARY\_PATH* so setzen:

```
export LD_LIBRARY_PATH=/X11R6/lib
```

Dies kann zum Beispiel in der »~/bash\_profile« geschehen. Setzen Sie diese Variable nicht systemweit in »/etc/profile«, wenn die von Ihnen verwendete *glibc*-Version kleiner als 2.1.95 ist. Erst ab dieser Version trägt sich die Variable *LD\_LIBRARY\_PATH* mit *suid*-Programmen wie »login«.

Jetzt sollte es möglich sein, X mit

```
startx
```

zu starten. Dies kann als root geschehen. Wenn Sie X als normaler Benutzer starten wollen, müssen Sie in der Datei »/etc/X11/Xserver« in der zweiten Zeile das Wort »Console« durch »Anybody« ersetzen. Dies ist die Umgehung eines Fehlers, der verhindert, daß X den Besitzer der Konsole erkennt. Der X-Server sollte nun ohne Probleme laufen.

Wird bei erneutem Start von X die folgende Fehlermeldung angezeigt, so sind sie auf einen noch nicht identifizierten Fehler gestoßen.

```
_XSERVTransSocketINETCreateListener: ...SocketCreateListener
() failed
_XSERVTransMakeAllCOTSServerListeners: server already running
```

```
Fatal server error:
Cannot establish any listening sockets - Make sure an X
server isn't already running
```

When reporting a problem related to a server crash, please send the full server output, not just the last messages

```
_X11TransSocketUNIXConnect: Can't connect: errno = 1073741826
_X11TransSocketUNIXConnect: Can't connect: errno = 1073741826
_X11TransSocketUNIXConnect: Can't connect: errno = 1073741826
xinit: No such file or directory (errno 1073741826):
unexpected signal 2
```

Als Abhilfe können Sie mit *ps aux* die Prozeß-ID des Prozesses »/hurd/pfinet« anzeigen und diesen Prozeß dann mit

```
kill -9 PID
```

abschießen, wobei Sie für »PID« die Prozeß-ID einsetzen. Danach sollte ein Neustart des Servers möglich sein.

Wer gerne »xdm« unter Hurd verwenden möchte, wird mit der Schwierigkeit konfrontiert, daß »xdm« den Pfad zu den X-Bibliotheken aus den Umgebungsvariablen entfernt. Um dieses Verhalten zu unterbinden, fügen Sie die folgende Zeile der Datei »/etc/X11/xdm-config« hinzu, nachdem Sie das Paket »xdm« installiert haben:

```
DisplayManager.exportList: LD_LIBRARY_PATH
```

Generell findet die systemweite Konfiguration von X bei Debian im Verzeichnis »/etc/X11« statt. Es lohnt sich, die Dateien in diesem Verzeichnis einmal genauer durchzusehen. Zu »xdm« noch eine spezielle Warnung: Bei der Installation des Pakets »xdm« wird automatisch ein Eintrag zu den Bootskripten hinzugefügt, um »xdm« bei jedem Neustart des Rechners zu starten. Dies ist bei nur einer Konsole eventuell unerwünscht. Entfernen Sie in diesem Fall den Link »/etc/rc2.d/S99xdm«. Sie können xdm wie jeden Debian-Dämon durch Eingabe von

```
/etc/init.d/xdm start
```

manuell nachstarten.

Neben den oben erwähnten Fehlern ist eine weitere Reihe von Problemen bekannt, die bei der Verwendung von X auftreten können. So ist zum Beispiel eine große Zahl untätiger Threads bei dem Server »/hurd/pflocal« beobachtet worden, und auch der »/hurd/mouse«-Server gerät mitunter aus dem Takt. Das Abschießen der jeweiligen Prozesse vor einem Neustart des X-Servers reicht üblicherweise aus, um vorübergehende Erleichterung zu schaffen. Die Behebung solcher Probleme hat eine hohe Priorität bei den Hurd-Entwicklern. Trotz dieser Mängel ist mit der Verfügbarkeit von leicht installierbaren Paketen des X-Window-Systems ein wichtiger Meilenstein für die Debian-GNU/Hurd-Distribution erreicht worden.



**Links:**

- [1] Uchiyama Yasushis ältere X-Seite: »<http://f77.nop.or.jp/doc/XFree86-3.3.3.html>«
- [2] Für ganz Wagemutige: Leicht veraltete XFree86-4.0.1-Pakete im experimentellen Status: »<http://master.debian.org/~brinkmd/xfree86-4.0.1/>«