

# **Hurd Installieren**

---

**Neal H. Walfield**

---

Copyright © 2000, 2001, 2002, 2003 Neal H. Walfield <neal@cs.uml.edu>

Permission is granted to make and distribute verbatim copies of this manual provided the copyright notice and this permission notice are preserved on all copies.

Permission is granted to copy and distribute modified versions of this manual under the conditions for verbatim copying, provided also that the sections entitled “Copying” and “GNU General Public License” are included exactly as in the original, and provided that the entire resulting derived work is distributed under the terms of a permission notice identical to this one.

Permission is granted to copy and distribute translations of this manual into another language, under the above conditions for modified versions, except that this permission notice may be stated in a translation approved by the Free Software Foundation.

# Inhaltsverzeichnis

<b>1</b>	<b>Übersicht</b> .....	<b>1</b>
<b>2</b>	<b>Partitionierung</b> .....	<b>1</b>
<b>3</b>	<b>Der Boot Loader</b> .....	<b>2</b>
<b>4</b>	<b>Cross Installation</b> .....	<b>2</b>
<b>5</b>	<b>Hurd booten</b> .....	<b>3</b>
<b>6</b>	<b>Vorkonfiguration</b> .....	<b>4</b>
<b>7</b>	<b>Konfiguration</b> .....	<b>5</b>
	7.1 Netzwerk .....	5
	7.2 Dateisysteme .....	5
	7.3 Reboot .....	6
<b>8</b>	<b>Schlussworte</b> .....	<b>6</b>
	8.1 Dokumentation .....	6
	8.2 Das Grub Menu .....	6
	8.3 Gerätedateien hinzufügen .....	6
	8.4 Pakete installieren .....	7
	8.5 XFree86 .....	7
	8.6 Virtuelle Terminals .....	8
	8.7 Mailing Lists .....	8
	8.8 Weitere Quellen .....	8
<b>9</b>	<b>Zitierte Werke</b> .....	<b>8</b>

# 1 Übersicht

Die Debian GNU/Hurd Distribution hat, anders als Distributionen von anderen Betriebssystemen, keine sehr komfortable Installationsprogramm. Eines Tages wird es eines geben, vielleicht sogar mit Ihrer Hilfe. Bis dahin benötigt GNU/Hurd zur Installation noch ein anderes, Unix-artiges Betriebssystem.<sup>1</sup>

Einige Benutzer konnten dies mit Hilfe verschiedener Linux-Varianten und den BSD-Systemen bewerkstelligen. Was das System lediglich können muss ist: ein ext2 Dateisystem erstellen; ein tar-Archiv auf dieses extrahieren; GNU Grub installieren.

GNU Hurd ist ähnlich zu Unix-Systemen: Nachdem der Benutzer sich angemeldet hat, präsentiert sich eine Shell und das geläufige Unix-VFS (virtual filesystem). Obwohl GNU versucht, POSIX-konform zu sein, ist es kein Unix (GNU is Not Unix). Hurd baut auf vielen Konzepten von Unix auf und erweitert diese entweder um neue Funktionalitäten oder beseitigt deren Fehler im Design. Der grösste Unterschied besteht wohl in den so genannten Translatoren, die es ermöglichen, dass Programme direkt mit dem VFS interagieren. Diese Dateisysteme müssen weder im Kernel sein, noch müssen sie mit root-Rechten laufen. Worauf sie Zugriff haben müssen, ist zum einen das Speichermedium, das durch den Translator mit dem Mount-Punkt verbunden wird, zum anderen der Mount-Punkt selbst.

Ein weiterer Unterschied besteht darin, dass Prozesse statt einer einzelnen bei ihrer Erzeugung festgelegten User ID über Identifizierungs-Token verfügen, die unabhängig vom Prozess sind. D.h. die Tokens können mit der entsprechenden Berechtigung von einer Autorisierungsstelle hinzugefügt oder entzogen werden.

Mit Unix-Systemen vertraut zu sein, ist hier sehr von Vorteil. Sich mit den verschiedenen Debian-Tools auszukennen, bringt ebenfalls Zusatzpunkte in der Konfiguration, der Pflege und Wartung eines GNU Hurd Rechners.

Ich bin bemüht, mit dieser Hilfe euch so schmerzlos wie nur möglich durch die Installation zu begleiten. Sollten dennoch Fehler auftreten (für die in vielen Fällen der Autor verantwortlich ist), meldet sie mir. Auch Anregungen oder Kritiken sind willkommen.

## 2 Partitionierung

Wenn keine freie(n) Partition(en) oder Festplatte(n) zur Verfügung stehen, wird dies der härteste Schritt. In diesem Fall benötigt man Software, um die Festplatte zu repartitionieren. Eine von vielen Lösungen wäre hier GNU's Partitions-Editor parted. Er unterstützt nicht nur einfaches Partitionieren. Auch Vergrössern, Verkleinern und Verschieben von Partitionen sind sein Element. Zu finden ist er unter <http://www.gnu.org/software/parted>. Es ist dringlichst auf die gelungene Dokumentation hinzuweisen die auch verschiedene Tutorials bietet.

Hurd kann nur mit Partitionen bis etwa 2Gb umgehen. Alles über diesem Wert wird nicht funktionieren. Das ist auf eine Entscheidung von vor mehreren Jahren zurückzuführen, durch die der Dateisystem-Server das ganze Dateisystem im Speicher abbildet. Es ist geplant, dies zu ändern.

---

<sup>1</sup> Philip Charles hat einen Satz CDs erstellt (verfügbar unter <http://www.debian.org/ports/hurd/hurd-cd>), die ein direkt bootbares Debian GNU/Linux System enthalten und diese Anforderung daher in gewisser Weise eliminieren, allerdings halten wir diese Lösung für äquivalent.

Sagen wir mal so, 2Gb ist mehr als genug für ein laufendes System. Es besteht ja weiterhin noch die Möglichkeit, mehrere Partitionen zu erstellen. Einmal 2Gb für '/' und (beispielsweise) nochmals 2Gb für '/home'.

Hurd unterstützt einige Erweiterungen des ext2 Dateisystem-Formates. Insbesondere sind passive Translator und ein viertes Rechte Tripel für unbekannte Nutzer (Nutzer ohne Identität – nicht "others") zu nennen.

Um diese zu nutzen, muss der Besitzer der Partition `hurd` sein. `mke2fs` wird, wenn nichts anderes angegeben wurde, den Besitzer auf das Betriebssystem setzen, auf dem es gerade läuft. Da Hurd diese Einstellung respektiert, muss man genau darauf achten, da es sonst zu nicht-offensichtlichen Problemen kommen wird. Zu beachten ist, dass, wenn ein Dateisystem einem bestimmten Betriebssystem gehört, andere es dennoch verwenden können; sie werden lediglich nicht in der Lage sein, diverse Erweiterungen zu verwenden.

Um aber nun ein Dateisystem zu erstellen, das den Besitzer `hurd` trägt, führen wir `mke2fs` mit der Option '-o hurd' aus.

```
# mke2fs -o hurd /dev/hda2
```

### 3 Der Boot Loader

Im Gegensatz zu zu GNU/Linux und den BSDs hat Hurd keinen eigenen Boot Loader. Jeder Boot Loader, der den Multiboot-Standard beherrscht kann Hurd starten. Momentan ist das aber nur einer. Grub, der "GRand Unified Boot loader".

Grub ist im Gegensatz zu den traditionellen Boot Loadern auf x86 (LILO sei als Beispiel genannt) sehr mächtig. Er besitzt eine Befehlszeile, beherrscht bootp, hat ein dummy-Terminal und eine Menge an weiteren Funktionen. Weiterhin kann er so gut wie jedes System booten. Wenn man jemals eine Sparc oder Alpha gebootet hat, wird man bereits wissen, zu was Grub im Stande ist. Deshalb, keine Angst: Grub ist besser. Du wirst ihn mögen. Du wirst keinen anderen mehr verwenden wollen.

Grub ist unter <http://www.gnu.org/software/grub/> zu finden. Dort befindet sich ein Quellen-Archiv. Wenn man sich die Quellen herunter geladen hat, kommt der klassische Dreisatz `configure`, `make` und `make install` zum Einsatz. Es ist nun Zeit, sich die Dokumentation von Grub durchzulesen. Alternativ kann auch ein Disketten-Image heruntergeladen werden, von dem dann gebootet wird. Dafür muss das Image auf eine Diskette geschrieben werden.

```
# dd if=grub-boot-image of=/dev/fd0
```

Später kann Grub auch auf der Festplatte installiert werden.

### 4 Cross Installation

Als nächstes wird das Basissystem heruntergeladen:

```
ftp://alpha.gnu.org/gnu/hurd/contrib/marcus/gnu-latest.tar.gz.
```

Das Tarball ist so konfiguriert, dass es alle benötigten Dateien in die vorgesehenen Verzeichnisse entpackt. Zunächst muss das Dateisystem gemountet werden. Davon ausgegangen, dass sich das einzuhängende Dateisystem auf '/dev/hda2' befindet, der Punkt, an dem

es eingehangen werden soll `/gnu` ist und sich das Archiv im Heim-Verzeichnis befindet, geben wir folgende Befehle ein:

```
# mount -t ext2 /dev/hda2 /gnu
# cd /gnu
# tar --same-owner -xvzpf ~/gnu-latest.tar.gz
```

## 5 Hurd booten

Nun sind wir bereit, Hurd zum ersten mal zu booten. Nachdem man sich versichert hat, dass sich die Bootdiskette mit Grub an Bord im Laufwerk befindet, starten wir den Rechner neu. Wenn alles gut geht, befindet man sich nun auf der Kommandozeile von Grub. Wenn widererwarten ein Menu angezeigt werden sollte, drücke `C` um auf besagte zu gelangen.

Als erstes muss GNU Mach geladen werden. Dazu muss man zunächst wissen, auf welcher Partition und in welchem Verzeichnis sich der Mach-Kernel befindet. Grub nutzt dazu eine andere Methode um die Festplatte und die darauf befindlichen Partitionen anzusprechen, als es bei Linux oder Hurd der Fall ist. IDE und SCSI Laufwerke werden demnach mit `(hdN,M)` angesprochen (zwischen IDE und SCSI wird nicht unterschieden), wobei `N` für die Nummer der Festplatte (bei Null beginnend) und `M` für die Partitionsnummer (ebenfalls bei Null beginnend) steht. Grub hilft einem aber auch ein wenig.

Um eine bestimmte Datei zu finden, von der man den genauen Ort der Speicherung nicht kennt, hat Grub ein Kommando `find`. Wenn man dieses zusammen mit einem Dateinamen aufruft, durchsucht Grub alle Dateisysteme und gibt aus, wo er sie gefunden hat. Um nach der Datei `/boot/gnumach.gz` zu suchen, gibt man folgendes ein:

```
grub> find /boot/gnumach.gz
(hd0,1)
```

Somit weiß man nun, dass sich `/boot/gnumach.gz` auf `(hd0,1)` befindet.

Bevor nun der Kernel geladen werden kann, muss noch eine weitere Option angegeben werden, die für Hurd, nicht für Grub wichtig ist. Und zwar muss man Hurd mitteilen, wo sich der Kernel befindet.

GNU Mach nummeriert die Festplatten bei 0 (Null) beginnend. IDE-Festplatten tragen das Präfix `hd`, wobei SCSI-Festplatten das Präfix `sd` tragen. Nummeriert wird in der Reihenfolge, in der sie an dem Controller angeschlossen sind. Die Festplatte am Master-Port des Primären Controller wird mit `hd0` und die Festplatte am Slave-Port mit `hd3` angesprochen. Die Partitionen werden ähnlich wie unter BSD mit `sm` angesprochen. Die Partitionen werden bei 1 (Eins) beginnend hochgezählt.

Da Hurd noch nicht konfiguriert wurde, muss man im Single-User-Modus starten. Das geschieht, indem man Hurd `-s` als Boot-Option mit auf dem Weg gibt.

Davon ausgehend, dass sich Hurd auf der zweiten Partition der Festplatte am Master-Port des Primären IDE-Controllers (`/dev/hda2`) befindet, muss folgendes eingegeben werden:

```
grub> kernel (hd0,0)/boot/gnumach.gz root=device:hd2s1 -s
[Multiboot-elf, ...]
```

Als nächstes sorgt man dafür, dass der `root` Dateisystem-Server und der `exec`-Server geladen wird. Das wird mit Grub's Möglichkeit, Module zu laden, erreicht. Die Werte für `{var}` werden von GNU Mach gefüllt. Da die Zeile für das `ext2fs`-Kommando doch etwas

lang ist, hat man die Möglichkeit, die Zeile mit einem Backslash zu trennen. Es ist dabei darauf zu achten, dass sich am Ende einer Zeile keine Leerzeichen hinter dem Backslash befinden. Weiterhin ist zwischen `{var}` und `(var)` zu unterscheiden (geschweifte und runde Klammern).

```
grub> module (hd0,0)/hurd/ext2fs.static \
--multiboot-command-line=${kernel-command-line} \
--host-priv-port=${host-port} \
--device-master-port=${device-port} \
--exec-server-task=${exec-task} -T typed ${root} \
$(task-create) $(task-resume)
[Multiboot-module 0x1c4000, 0x2cfe6a bytes]
grub> module (hd0,0)/lib/ld.so.1 /hurd/exec $(exec-task=task-create)
[Multiboot-module 0x494000, 0x27afe bytes]
```

Keine Angst, wenn Hurd fertig eingerichtet ist, kann man Grub automatisieren und die Kommandos in der Konfigurationsdatei `/boot/grub/menu.lst` ablegen.

GNU Hurd kann nun gestartet werden:

```
grub> boot
```

Nun lehn Dich zurück und beobachte die Meldungen auf dem Bildschirm. Noch ein wichtiger Hinweis: GNU Mach hat einen Bug, bei dem der Kernel beim Drücken einer Taste beim Bootvorgang hängen bleibt (Kernel-Panic).

Wenn Hurd nicht bootet, kann es am IRQ-Sharing liegen. GNU Mach beherrscht dieses nicht. Auch werden keine nachladbaren Kernel-Module unterstützt, woraufhin Treiber für die gängigsten Geräte in den Kernel kompiliert sind. Weiterhin können ältere Geräte Probleme bereiten, wenn auf Anfragen von GNU Mach an nicht vorhandene Geräte fehlerhafte Antworten kommen. Um dies zu lösen, muss ein eigener Mach-Kernel kompiliert werden, der nur die benötigten Treiber enthält. Wenn Du Debian GNU/Linux dein Eigen nennst, installiere das Paket `'gcc-i386-gnu'`.

Wenn weiterhin Probleme bestehen, schau bei den Links am Ende des Dokumentes vorbei. Wenn alle Stricke reißen, wird eine Anfrage an die zuständige Mailingliste zur Lösung beitragen.

## 6 Vorkonfiguration

Sobald man sich auf der Shell befindet, und jedesmal, wenn man im Single-User-Modus ist, muss der Terminal-Typ angegeben werden.

```
# export TERM=mach
```

Weiterhin wird `Strg-C` und dergleichen im Single-User-Modus nicht unterstützt.

Nun startet man das `native-install` Script, das die Pakete und verschiedene wichtige Translator konfiguriert:

```
# ./native-install
```

Nachdem der erste Durchlauf dieses Scriptes erfolgt ist, wird uns selbiges darauf hinweisen, dass es einen zweiten, finalen, Durchlauf benötigt. Also startet man den Rechner mittels `reboot` neu und befolgt nochmals die Anweisungen aus Section '5. Hurd booten', um abermals das `native-install` Script auszuführen.

## 7 Konfiguration

### 7.1 Netzwerk

Um die Netzwerkkarte zu konfigurieren, muss der pfinet-übersetzer konfiguriert werden. Dazu benutzt man `settrans`, um einen Übersetzer an einem Knoten auf dem Dateisystem anzuschließen. Wenn nun ein Programm auf diesen Knoten zugreift, um beispielsweise ein ICMP/TCP/UDP-Paket zu senden, wird Hurd automatisch einen Server starten, um die Anfrage auszuführen.

```
# settrans -fgap /servers/socket/2 /hurd/pfinet -i eth0 \
-a a.b.c.d -g e.f.g.h -m i.j.k.l
```

Erklärung der Argumente: `'fg'` beendet vorher gestartete Translator `'ap'` macht den Translator aktiv und passiv `'-i'` das Interface an dem `pfinet` lauschen soll `'-a'` die IP-Adresse `'-g'` der Gateway `'-m'` die Netzwerkmaske

Als aktiv bezeichnet man einen Translator, wenn er sofort gestartet wird und er nach Beendigung nicht mehr vorhanden ist, z.B. wenn der Rechner neu gestartet wird. Ein passiver Translator hingegen speichert die Einstellungen im Dateisystem. Dadurch bleiben diese auch erhalten, wenn der Rechner neu gestartet wird. Er wird erst wieder gestartet, wenn ein Prozess auf den zugewiesenen Knoten zugreift.

Nun werden unter `'/etc/resolv.conf'` die DNS-Server eingetragen:

```
nameserver 145.253.2.11
nameserver 145.253.2.74
```

Um nun zu testen, ob die Netzwerkkarte richtig eingerichtet ist, wird `ping -c2 gateway` eingegeben. `'-c2'` veranlasst `ping` dazu, nur 2 ICMP-Pakete abzusetzen, da `Strg-C` nicht greifen würde und man keine Möglichkeit hat, `ping` zu beenden.

DHCP wird unter Hurd bisweilen nicht unterstützt. Das ist auf `pfinet` zurückzuführen, das auf den TCP/IP-Code von Linux zurückgreift, und nicht in der Lage ist, an 0.0.0.0 zu lauschen.

Um Hilfe für `settrans` zu erhalten, wird dem Programm `'--help'` als Argument übergeben. Auch die einzelnen Translatoren werden daraufhin eine kleine Übersicht geben.

```
# /hurd/pfinet --help
```

Die Ausgaben sollte man der Länge wegen durch eine less-Pipe schicken.

```
# /hurd/pfinet --help |less
```

### 7.2 Dateisysteme

Nun widmet man sich der Datei `'/etc/fstab'`. Dort sind alle Partitionen einzutragen, die man unter Hurd nutzen will. Es ist sehr wichtig, dass eine Swap-Partition eingebunden wird, da Hurd dadurch wesentlich stabiler läuft. Hurd ist auch in der Lage mit Linux eine Swap-Partition zu teilen. Beachte die 2Gb-Grenze! Als Standard-Editor ist `nano` installiert.

Hier ein Beispiel einer `'/etc/fstab'`:

```
# <file system> <mount point> <type> <options> <dump> <pass>
/dev/hd0s2      /          ext2  rw    0     1
/dev/hd0s3      /home     ext2  rw    0     2
/dev/hd0s4      none      swap  sw    0     0
```

Man sollte nicht vergessen, die nötigen Gerätedateien zu erstellen. Das geschieht mit MAKEDEV

```
# cd /dev
# ./MAKEDEV hd0s1 hd0s2 hd0s3 hdMsN
```

Um NFS-Dateisysteme einzuhängen, wird der `/hurd/nfs`-Translator genutzt. Wenn der Translator ohne root-Rechte gestartet wird, versucht er zu einem unprivilegierten Port auf dem NFS-Server zu verbinden, was in den meisten Fällen von diesem zurückgewiesen wird. Um dieses zu erlauben, muss auf dem NFS-Server in der Datei `/etc/exports` in den zuständigen Zeilen `'insecure'` eingetragen werden.

```
/home 192.168.0.10(rw,insecure)
```

Um dieses nun in Hurd einzubinden wird wieder `settrans` benötigt:

```
# settrans -cgap /mount/point /hurd/nfs 192.168.0.2:/home
```

## 7.3 Reboot

So, nun ist es endlich so weit sich in den Multi-User-Modus zu befördern. Dazu gibt man zunächst `reboot` auf der Kommandozeile ein. Siehe [Kapitel 5 \[Hurd booten\]](#), Seite 3. Diesmal wird das `'-s'` in der Kernel-Zeile weggelassen.

Happy Hacking!

# 8 Schlussworte

## 8.1 Dokumentation

Um Informationen über Hurd zu erhalten, solltest Du Debian's Hurd-Seite: <http://www.debian.org/ports/hurd/> und GNU's Seite: <http://hurd.gnu.org> besuchen.

## 8.2 Das Grub Menu

Um nicht für jeden Bootvorgang die Monsterzeilen für Grub einzugeben, ist es ratsam `'/boot/grub/menu.lst'` zu editieren.

## 8.3 Gerätedateien hinzufügen

Es befinden sich nur wenige Gerätedateien in `'/dev'`. Nutze das `'MAKEDEV'` Script, um alle benötigten zu erstellen.

## 8.4 Pakete installieren

Es gibt mehrere Möglichkeiten, Pakete zu installieren. Ein Paket herunterzuladen und es mit `dpkg -i` zu installieren, ist recht unbequem. Die einfachere Variante ist da wohl `apt-get`. Dazu sind folgende Einträge in der Datei `/etc/apt/sources.list` nötig:

```
deb ftp://alpha.gnu.org/gnu/hurd/debian unstable main
deb ftp://ftp.debian.org/debian unstable main
```

Sollte GNU Mach die Netzwerkkarte nicht unterstützen oder wird ein Modem genutzt, bleibt nur die Möglichkeit die Pakete herunterzuladen und sie auf Hurd zu übertragen. Unter `/usr/share/doc/apt/offline` gibt es weitere Informationen dazu.

## 8.5 XFree86

XFree86 wurde portiert und alle Grafikkarten, die keine Kernelmodule erfordern, werden unterstützt.

Zunächst konfiguriere den Keyboard-Translator:

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

Danach den Maus-Translator. Für eine serielle Maus steht folgende Zeile, wobei `--device=com0` mit den zutreffenden COM-Port ersetzt wird:

```
# settrans /dev/mouse /hurd/mouse --device=com0 --protocol=microsoft
```

Es ist sicherzustellen, das `/dev/com0` existiert. Wenn nicht, nutzt man MAKEDEV um dieses zu erstellen.

Eine PS/2 Maus benötigt keine Gerätedatei. Es reicht folgende Zeile um zum gewünschten Ergebnis zu gelangen:

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

Andere Mäuse werden auch unterstützt. Man startet `/hurd/mouse` mit der Option `--help` für mehr Details.

Es werden verschiedene Pakete benötigt. `x-window-system-core`, `rxvt` und `twm` oder `fvwm` sollten fürs erste reichen.

Debconf kann zum Konfigurieren von XFree86 genutzt werden, was aber eine Nachbearbeitung der `XF86Config` bzw. der `XF86Config-4` nach sich zieht. Die Section Pointer ist wie folgt zu ändern:

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

Optional kann `Emulate3Buttons` hinzugefügt werden. Andere Optionen werden nicht funktionieren.

GNU Hurd nutzt keine `ld.so.conf`. Da sich `/X11R6/lib` nicht im Standard Suchpfad für die Bibliotheken befindet, ist folgender Eintrag in `~/profile` vonnöten:

```
export LD_LIBRARY_PATH=/X11R6/lib:$LD_LIBRARY_PATH
```

Nun führe `startx` zum Starten des X-Servers aus.

Es sind noch einige Dinge zu beachten. `xterm` arbeitet nicht richtig, nutze dafür `rxvt`. `update-menu` wurde noch nicht portiert. Somit gibt es kein sauberes Debian-Menü. Obwohl eine `pthread`s Implementation existiert, wurden noch nicht sämtliche Pakete portiert. GNOME und KDE werden daher noch nicht lauffähig sein.

## 8.6 Virtuelle Terminals

Es wird an virtuellen Terminals unter Hurd gebastelt. Nutze bis dahin `screen`.

## 8.7 Mailing Lists

1. `debian-hurd@debian.org` Debian's Hurd-ML
  1. Archiv: <http://lists.debian.org/#debian-hurd>
2. `web-hurd@gnu.org` Entwicklung der Hurd-Seite <http://hurd.gnu.org>.
  1. Archiv: <http://mail.gnu.org/pipermail/web-hurd/>
3. `help-hurd@gnu.org` GNU's ML für Hilfe mit Hurd
  1. Archiv: <http://mail.gnu.org/pipermail/help-hurd/>
4. `bug-hurd@gnu.org` Hier gehen Bug's und Patches hin.
  1. Archiv: <http://mail.gnu.org/pipermail/bug-hurd/>

## 8.8 Weitere Quellen

1. The Hurd Wiki <http://hurd.gnufans.org/bin/view/Hurd/WebHome>
2. Deutsche FAQ <http://www.gnu.org/software/hurd/faq.de.html>
3. gut sortierte Linkliste <http://debian.fmi.uni-sofia.bg/~ogi/hurd/links/>

## 9 Zitierte Werke

“The Easy Guide to Installing Hurd on a Linux Box” Copyright © 1999 Matthew Vernon [matthew@debian.org](mailto:matthew@debian.org). <http://www.pick.ucam.org/~mcv21/hurd.html>