Linux Drucker HOWTO

Grant Taylor (gtaylor@picante.com), Jens Vonderheide (Jens_Vonderheide@wharfrat.fido.de) und Robert Fendt (fendt@student.physik.uni-dortmund.de) v3.17-3, 14. September 1998

Dieser Text ist eine Zusammenfassung von Informationen, wie man unter Linux (und anderen UNIX-Derivaten) Dokumente generiert, ansieht, druckt oder faxt.

Inhaltsverzeichnis

1 Einleitung

Die *Drucker HOWTO* sollte alles beinhalten, was sie wissen müssen, um Druckdienste auf Linux-Maschinen einzurichten. Die Installation von Druckern unter Linux ist zwar komplizierter als bei anderen Betriebssystemen, dafür ist die Lösung von Linux bzw. Unix erheblich flexibler und auch in großen Netzverbunden ohne Probleme einsetzbar.

Dieses Dokument ist so strukturiert, daß die meisten Leute nur etwa die erste Hälfte lesen müssen. Die meisten der schwierigeren und situationsabhängigen Informationen finden sich in der zweiten Hälfte und können im Inhaltsverzeichnis leicht gefunden werden, wobei ein Großteil der Informationen in den Abschnitten 9 bzw. 10 wahrscheinlich von den meisten Leuten gebraucht wird.

Diese HOWTO ist mit der Version 3 komplett neu geschrieben worden, deshalb sind viele Informationen aus früheren Versionen verlorengegangen. Das ist Absicht, da die alten Versionen 60 Seiten und mehr umfaßten und den Erzählfluß einer toten Schildkröte hatten. Wer hier keine passende Antwort findet, sollte zum einen die vorhergehende Version auf der Printing HOWTO Home Page

```
http://www.picante.com/~gtaylor/pht/
```

durchsuchen und zum anderen dem Autor einen Nachricht hinterlassen, was in dieser HOWTO noch stehen sollte.

Die alten Versionen stehen nur in englischer Sprache zur Verfügung.

Die Printing HOWTO-Home Page ist eine gute Stelle, um die aktuelle (englische) Version dieser HOWTO zu bekommen.

Die deutsche Version ist auf

```
http://www.tu-harburg.de/dlhp/
```

zu bekommen. Hier sind auch weitere deutsche Übersetzungen zu haben. Für die Koordination des deutschen HOWTO-Projekts ist Marco Budde (`Budde@tu-harburg.de`) verantwortlich.

1.1 Anmerkungen zur Übersetzung

Die Übersetzer haben sich bemüht, möglichst wörtlich zu übersetzen. Daher wurden in dieser HOWTO-Übersetzung auch die Sätze des Autors, die in der ersten Person geschrieben sind, ohne Änderung übernommen. Im folgenden bezieht sich daher »Ich« auf den Autor des englischen Original HOWTO-Textes, Grant Taylor.

Um die Verständlichkeit zu erhalten, sind einige Fachbegriffe, die im Englischen selbst-erklärend sind, unübersetzt geblieben:

Spool

Warteschlange. Hier »warten« Druckaufträge, bis der Drucker bereit ist.

Device

Wörtlich: Gerät. Kann sich auf ein physikalisches Gerät wie den Drucker beziehen, bezeichnet aber meistens ein logisches Gerät für Linux, z.B. `lp1`.

Manual Page

Handbuchseite. Beschreibt einen Befehl o.ä. Wird durch `man` <`Befehl`> angezeigt.

1.2 Copyright

2 Wie druckt man?

Wenn `lpd` bereits so eingerichtet ist, daß man drucken kann, dann muß nur noch der Gebrauch des `lpr`-Kommandos erlernt werden. Dieses ist jedoch nicht Gegenstand dieser HOWTO. Zu diesem Thema gibt es eine eigene HOWTO, die *Printing Usage HOWTO*.

Wenn Sie allerdings ein neues System oder einen neuen Drucker haben, dann werden Sie die Druckdienste auf die eine oder andere Weise einrichten müssen, bevor Sie drucken können.

3 Kernel-Drucker-Devices

3.1 lp-Device

Der Linux-Kernel (<=2.1.32), falls das lp-Device einkompiliert oder geladen wurde (die Ausgabe von `cat /proc/devices` sollte das Device lp einschließen), stellt ein oder mehr Devices `/dev/lp0`, `/dev/lp1` und `/dev/lp2` zur Verfügung. Diese werden *nicht* dynamisch zugeordnet, sondern korrespondieren vielmehr mit einer bestimmten I/O-Adresse der Hardware. Das heißt, daß der erste Drucker entweder `lp0` oder `lp1` sein kann, abhängig von der verwendeten Hardware. Testen Sie einfach beides ;-).

Einige Leute haben berichtet, daß ihre bidirektionalen lp-Ports nicht erkannt wurden, wenn sie ein altes, unidirektionales Kabel verwendeten. Überprüfen Sie also, ob Sie über ein vernünftiges Kabel verfügen.

Man kann nicht die plip- und lp-Treiber gleichzeitig auf einem Port laufen lassen. Man kann aber einen der beiden Treiber entweder manuell oder mit dem kerneld (mit Kernelversion 2.x oder späten 1.3.x) laden. Wenn man die Interrupts u.ä. sinnvoll setzt, kann man lp auf dem einen und plip auf dem anderen Port laufen lassen. Jemand hat das gemacht, indem er die Treiber geändert hat; es bleibt abzuwarten, ob es auch eine einfache Kommandozeilen-Methode gibt.

Es gibt ein kleines Utility namens tunelp, mit dem man als root den verwendeten Interrupt, die Übertragungsgeschwindigkeit und anderes eines lp-Device einstellen kann.

Wenn der Treiber in den Kernel eingebunden wurde, können mit der `lp=`-Option während des Bootvorganges die Portadressen und Interrupts eingestellt werden. Die Option hat folgenden Syntax:

```
lp=port0[,irq0[,port1[,irq1[,port2[,irq2]]]]]
```

Um die Portadresse 0x378 und keinen Interrupt (Polling) zu verwenden, würde dann folgendes benutzt werden:

```
lp=0x378,0
```

Mit nachfolgender Option würde die Adresse 0x278 mit dem Interrupt 5 und die Adresse 0x387 mit dem Interrupt 7 verwendet werden:

```
lp=0x278,5,0x378,7
```

Die Einstellungen des letzten Beispieles spiegeln die Standardwerte für die beiden parallelen Schnittstellen wieder.

Wenn diese Funktion benutzt wird, müssen *alle* Ports angegeben werden, die berücksichtigt werden sollen, es gibt keine Voreinstellungen. Der eingebaute Treiber kann mit `lp=0` deaktiviert werden.

Wenn der Treiber als Modul geladen wird, können die Adressen und Interrupts in der üblichen Syntax dem insmod-Kommando übergeben werden. Falls der kerneld Verwendung findet, können die Einstellungen in der Datei `/etc/conf.modules` vorgenommen werden. Die Parameter sind `io=port0,port1,port2` und `irq=irq0,irq1,irq2`. Weitere Informationen liefert die Manual Page zu `insmod`.

Der Quelltext für den Linux Parallelporttreiber befindet sich in der Datei `/usr/src/linux/drivers/char/lp.c`.

3.2 parport-Device

Beginnend mit Kernel 2.1.33 (und verfügbar als Patch für Kernel 2.0.30) ist das lp-Device eigentlich nur noch ein Client des neuen parport-Devices. Die Erweiterung um das parport-Device beseitigt einige der Probleme, die den alten lp-Treiber quälten: ein Port kann mit anderen Treibern »geteilt« werden, den verfügbaren Parallelports werden dynamisch Device-Nummern zugeordnet, anstatt eine feste Zuordnung zu erzwingen, und einiges mehr.

Die Verfügbarkeit des parport-Devices hat eine ganze Reihe neuer Treiber für den Parallelport ermöglicht, so z.B. für parallele ZIP-Laufwerke, externe CD-ROMs und Festplatten. Einige davon sind auch für Kernel 2.0.x verfügbar; weitere Informationen hierzu finden sich im WWW.

Eine Dokumentation zum parport-Treiber ist in den Kernelquellen in der Datei `Documentation/parport.txt` zu finden. Auch ein Blick auf die *parport-Webseite* unter folgender Adresse könnte sinnvoll sein:

```
http://www.cyberelk.demon.co.uk/parport.html
```

3.3 Serielle Devices

Die Devices serieller Schnittstellen heißen unter Linux `/dev/ttySx`, wobei `x` die spezielle Schnittstelle bezeichnet. Die erste serielle Schnittstelle wäre also `/dev/ttyS0`.

Das Programm `stty` erlaubt es, interaktiv die Einstellungen für die seriellen Schnittstellen anzusehen oder zu verändern; mit `setserial` können einige erweiterte Attribute eingestellt werden und IRQs bzw. I/O-Adressen für nicht-standardmäßige Schnittstellen konfiguriert werden. Weitere Informationen über serielle Schnittstellen können in der *Serial HOWTO* nachgelesen werden.

Wenn man einen langsamen seriellen Drucker mit serieller Flußkontrolle benutzt, kann es sein, daß einige Druckaufträge nicht vollständig ausgedruckt werden. Das kann mit der seriellen Schnittstelle zusammenhängen. Damit die von einem Programm gelieferten Daten nicht verloren gehen, wenn der Drucker per Flußkontroller dem Computer mitteilt, daß er zur Zeit keine weiteren Daten empfangen kann, weil sein Speicher voll ist, speichert der Treiber der seriellen Schnittstelle die Daten in einen Puffer.

Diese Puffer kann 4096 Zeichen aufnehmen. 30 Sekunden nachdem das Programm das Schnittstellendevice geschlossen hat, löscht der Treiber den Puffer. Falls der Drucker also nicht schnell genug ist, in den 30 Sekunden die im Puffer stehenden Daten zu empfangen, gehen diese verloren.

Wenn der Befehl

```
cat file > /dev/ttySx
```

kurze Dateien komplett ausdruckt, bei längeren aber das Ende abschneidet, könnte genau dieser Fall vorliegen.

Die 30-Sekunden-Wartezeit kann durch die `closing_wait`-Kommandozeilenoption von `setserial` (Version 2.12 oder später) eingestellt werden. Die seriellen Schnittstellen eines Rechners werden üblicherweise durch einen Aufruf von `setserial` in einer der Bootdateien initialisiert. Der Aufruf für den Port, an dem der Drucker hängt, kann so modifiziert werden, daß zusammen mit den anderen Parametern die `closing_wait`-Option eingestellt wird.

4 Unterstützte Drucker

Der Linux-Kernel unterstützt praktisch alle Drucker, die man an eine serielle oder parallele Schnittstelle anschließen kann, aber es gibt einige Dinge, auf die man achten sollte, und auch einige Drucker, die Sie nicht werden benutzen können, obwohl sie elektrisch gesehen mit Linux kommunizieren können.

Zu diesen gehören vor allem jene inkompatiblen Drucker, die auf das *Windows Printing System* angewiesen sind. In Prospekten oder Handbüchern werden diese Drucker gerne als »Drucker für Windows« oder als GDI-Drucker bezeichnet. Diese Drucker funktionieren *nicht* mit Linux. Dieses liegt darin begründet, daß diese Drucker im Prinzip nur das eigentliche Druckwerk und eine Schnittstelle zum PC enthalten. Das führt dazu, daß der PC viele Aufgaben beim Drucken übernehmen muß, die der Drucker normalerweise selbst erledigt. Die gesamte Intelligenz des Druckers steckt deshalb in den Druckertreibern, die auf dem PC installiert sind. Bisher gibt es solche Treiber nur für Windows und selbst dort gibt es teilweise Probleme, passende Treiber zu bekommen. Wenn Sie Linux einsetzen möchten, sollten Sie also sehr aufpassen, nicht einen GDI-Drucker zu kaufen. Auch sonst lohnt sich die Anschaffung von GDI-Druckern nicht, da vernünftige Geräte kaum teurer sind und erheblich weniger Probleme verursachen. Wenn Sie bereits einen solchen Drucker besitzen, gibt es Möglichkeiten, um Linux zur Zusammenarbeit zu bewegen, aber diese sind unzureichend. Weitere Informationen zu diesem Typ von Druckern finden Sie in Abschnitt 12.

Die beste Wahl für Linux sind Drucker, die einen Interpreter für PostScript enthalten. Fast jedes Unix-Programm produziert PostScript-Daten, wenn gedruckt werden soll, so daß es offensichtlich keine schlechte Idee ist, einen Drucker zu verwenden, der diese Seitenbeschreibungssprache direkt versteht. Im professionellen Bereich wird fast ausschließlich auf PostScript als Druckersprache gesetzt, da PostScript viele Vorteile bietet. Gegen Drucker mit PostScript-Unterstützung sprechen vor allem die Kosten. PostScript findet man standardmäßig eigentlich nur bei teurer Laserdruckern. Ganz selten gibt es auch Tintenstrahldrucker mit PostScript. Bei den meisten Druckern muß man für die PostScript-Unterstützung mindestens Zusatzkosten von 300,- DM einkalkulieren.

Wenn Sie die zusätzlichen Kosten eines PostScript-Druckers scheuen, so können Sie einen Drucker anschaffen der von Ghostscript unterstützt wird. Ghostscript ist ein kostenloser PostScript-Interpreter für Linux. Mittels dieses Programmes können Sie PostScript-Daten in die Druckersprache Ihres Druckers übersetzen. Ihr Drucker wird also quasi zu einem PostScript-Drucker. Auf der *Ghostscript Homepage*, die Sie unter der Adresse

```
http://www.cs.wisc.edu/~ghost/
```

erreichen, finden Sie eine Liste der unterstützten Drucker und Informationen über den Status von neuen und experimentellen Treibern.

Adobe hat eine neue Druckersprache namens *PrintGear* entwickelt. Es handelt sich dabei wohl um eine stark vereinfachte Sprache im Binärformat, die einiges von PostScript geerbt hat, aber nicht dazu kompatibel ist. Anscheinend gibt es bisher keine Unterstützung durch Ghostscript für diese neue Druckersprache.

4.1 Drucker-spezifische Bemerkungen

Dieser Abschnitt ist unvollständig und wird es immer sein. Aber die enthaltenen Informationen sollten korrekt sein.

Canon BJ-10ex

Ein Bericht schlägt vor, daß der Canon sehr viel besser im Epson LQ Modus arbeitet, der per DIP-Schalter eingeschaltet werden kann. Anscheinend war die Ausgabe im Canon-Modus unterbrochen oder falsch. Andererseits gibt es mittlerweile bessere Unterstützung für Canon-Drucker in Ghostscript, so daß sich dieses Problem erledigt haben dürfte.

Lexmark

Lexmark stellt viele verschiedene Drucker her; viele sind allerdings GDI-Drucker:

- Color Jetprinter 1000
- Color Jetprinter 1020 (non Business Edition)
- Color Jetprinter 2030
- Color Jetprinter 2050
- Color Jetprinter 2070
- Color Jetprinter 7000
- Color Jetprinter 7200
- Winwriter 100
- Winwriter 150c
- Winwriter 200

5 Welche Spool-Software?

Bis vor kurzem war die Wahl für Linux-Anwender einfach; jeder verwendete denselben alten `lpd`, der fast komplett aus der Net-2-Distribution von BSD übernommen wurde. Auch heute liefern die meisten Distributoren noch diese Software aus. Aber das beginnt sich zu verändern. SVR4-ähnliche Systeme wie Sun Solaris kommen mit einem komplett anderen Spool-Paket, das auf lpsched beruht. Es gibt ferner Anzeichen, daß einige Linux-Distributoren auf LPRng umsteigen werden, eine wesentlich neuere Implementierung, die frei erhältlich ist. LPRng ist für große Installationen wesentlich einfacher zu verwalten und besitzt einen nicht so erschreckend zusammengewürfelten Code wie der ursprüngliche `lpd`. Es kann sogar wirklich von sich behaupten, sicher zu sein; es gibt keine SUID-Teile, und es unterstützt Authentisierung via PGP oder Kerberos.

Im Moment ist der `lpd` wohl für die meisten Linuxanwender trotz der neuen Möglichkeiten der anderen Programme die beste Lösung. Denn obwohl es sich nicht um das schönste System handelt, funktioniert es sehr gut, wenn es erst einmal eingerichtet ist. Sollten Probleme auftauchen, hat man den Vorteil, daß sich viele Leute mit dem `lpd` auskennen und einem so helfen können. Außerdem ist der `lpd` in vielen Unix-Büchern gut dokumentiert.

Weitere Informationen über LPRng sind auf der LPRng-Seite zu finden:

```
http://www.astart.com/lprng/LPRng.html
```

6 Grundlagen

Damit das Drucken gut funktioniert, ist es wichtig, die Funktionsweise des lpd-Systems zu verstehen.

Lpd steht für *Line Printer Daemon* und bezieht sich je nach Zusammenhang entweder auf den Daemon selbst oder auf die gesamten Programme, die für das Druck-Spooling zuständig sind. Hierzu gehören:

`lpd`

Dieses ist der eigentlich Daemon, der für die Ausgabe der Daten auf den Drucker sorgt.

lpr

Mittels dieses Befehls kann der Anwender Druckaufträge in die Warteschlange des lp-Daemon abschicken.

lpq

Listet die Aufträge in einer Drucker-Warteschlange auf.

lpc

Der Lpd-Systemkontrollbefehl. Mit `lpc` können die Warteschlangen gestoppt, gestartet, umsortiert, etc. werden.

lprm

`lprm` entfernt einen Auftrag aus der Warteschlange.

Und wie paßt das alles zusammen? Nun, wenn der Computer bootet, wird der `lpd` gestartet. Der Daemon durchsucht die Datei `/etc/printcap`, um festzustellen, für welche Drucker Warteschlangen verwaltet werden. Jedesmal, wenn man `lpr` aufruft, nimmt `lpr` mit dem `lpd` durch den Named Socket `/dev/printer` Verbindung auf und übergibt dem `lpd` die zu druckende Datei und einige Informationen, wer druckt und wie zu drucken ist. `lpd` druckt dann die Datei auf dem passenden Drucker aus.

Das lp-System wurde ursprünglich entwickelt, als die meisten Drucker Zeilendrucker waren. Die Drucker kannten weder verschiedene Zeichensätze noch Grafikmodi. Es wurde vom lp-Daemon einfach der Text im ASCII-Format an den Drucker übertragen. Daß das lp-System auch heute noch verwendet werden kann, liegt darin begründet, daß auch heute das lp-System einzig die Aufgabe hat, die Druckdaten, z.B. in PCL oder PostScript, an den Drucker zu übertragen.

7 Grundeinstellungen

7.1 Traditionelle lpd-Konfiguration

Die Minimaleinstellungen für `lpd` liefern ein System, das Dateien in Warteschlangen verwalten und sie drucken kann. Es kümmert sich aber nicht darum, ob der Drucker die Dateien überhaupt versteht und wird vermutlich keine ansehnlichen Ausgaben produzieren. Trotzdem ist diese einfache Konfiguration der erste Schritt, um das System zu verstehen.

Um eine neue Warteschlange zu erzeugen, muß man einen Eintrag in `/etc/printcap` hinzufügen und ein neues Spool-Verzeichnis unter `/var/spool/lpd` erzeugen.

Ein Eintrag in `/etc/printcap` sieht etwa so aus:

```
# lokaler DeskJet 500
lp|dj|deskjet:\
        :sd=/var/spool/lpd/dj:\
        :mx#0:\
        :lp=/dev/lp0:\
        :sh:
```

Dies definiert eine Warteschlange mit den Namen `lp`, `dj` und `deskjet`, wobei `/var/spool/lpd/dj` als Spool-Verzeichnis benutzt wird. Die maximale Größe der Aufträge ist nicht begrenzt und am Anfang der Druckaufträge wird kein Deckblatt, z.B. mit dem Namen der Person, die den Druckauftrag abgeschickt hat, ausgegeben. Der Drucker würde in diesem Beispiel an `/dev/lp0` hängen.

Jetzt wäre der richtige Augenblick, um die *printcap Manual Page* zu lesen.

Das obige Beispiel sieht sehr einfach aus, hat aber ein Problem. Wenn man nicht Dateien an das Drucksystem übergibt, die ein DeskJet 500 verstehen kann, wird dieser Drucker seltsame Sachen ausgeben. Wenn man z.B. eine gewöhnlichen UNIX-Textdatei an den Deskjet schickt, wird man folgende Ausgabe erhalten:

```
Zeile eins
          Zeile zwei
                    Zeile drei
```

Verursacht wird dieser Fehler dadurch, daß Linux einen Zeilenumbruch in einer Textdatei anders kodiert als DOS und Windows, deren Kodierung die meisten Drucker erwarten. Würde eine PostScript-Datei ausgegeben, würde der Drucker einfach die PostScript-Befehle ausdrucken, statt diese zu interpretieren.

Offensichtlich wird mehr benötigt, und genau das ist die Aufgabe von Filtern. Dem aufmerksamen Leser werden bei der *printcap Manual Page* die Spool-Attribute `if` und `of` aufgefallen sein. `if`, der Inputfilter, ist genau das, was wir jetzt brauchen.

Um das Problem mit dem Zeilenumbruch beim Drucken von Textdateien zu lösen, könnte man ein Shellskript mit dem Namen `filter` schreiben, das die Kodierung des Zeilenumbruches anpaßt. Damit der `lpd` dieses Skript aufruft, muß dem `printcap`-Eintrag des Druckers eine `if`-Zeile hinzugefügt werden:

```
lp|dj|deskjet:\
        :sd=/var/spool/lpd/dj:\
        :mx#0:\
        :lp=/dev/lp0:\
        :if=/var/spool/lpd/dj/filter:\
        :sh:
```

Ein einfaches Filterskript könnte sein:

```
#!perl
# Die obige Zeile muß den kompletten Pfad zu perl
# enthalten. Dieses Skript muß ausführbar sein:
#    chmod 755 filter

while(<STDIN>){chop $_; print "$_\r\n";};

# Eventuell möchte man, daß am Ende des Druckauftrages
# ein Seitenvorschub ausgeführt wird. Dieses ist
# insbesondere bei Tintenstrahl- und  Laserdruckern
# sinnvoll:

#print "\f";
```

Würde man das System so konfigurieren, hätte man eine Warteschlange, die wunderbar für UNIX-Textdateien funktionieren würde. Natürlich gibt es vier Millionen bessere Möglichkeiten, diesen Filter zu schreiben, aber wenige sind so anschaulich. Der Leser möge dieses effizienter gestalten.

Das einzige verbleibende Problem besteht darin, daß man heute meistens keine Textdateien drucken möchte. Vielmehr sollen meistens PostScript- oder Grafikdateien ausgegeben werden. Auch dieses Problem läßt sich mit einem Inputfilter lösen. Dazu muß einfach der obige Zeilenumbruch-Filter erweitert werden. Wenn man einen Filter schreibt, der beliebige Dateitypen akzeptiert und diese in DeskJet-geeignete Ausgaben umwandelt, hat man wirklich einen cleveren Druck-Spooler.

So ein Filter wird *Magic*-Filter genannt. Man sollte sich nicht die Mühe machen und selber einen schreiben, solange man keine wirklich ungewöhnlichen Sachen drucken will. Es gibt bereits einige wirklich gute Filter. Der APS-Filter ist einer der besten Filter. Viele Linux-Distributionen werden auch mit Setup Tools für den Drucker ausgeliefert, die die Konfiguration von Druckern und passenden Filtern deutlich erleichtern.

7.2 Dateizugriffsrechte

Auf Grund von häufigen Nachfragen folgt hier eine Liste der Zugriffsrechte der wichtigen Dateien, wie sie auf meinem System gesetzt sind. Es gibt sicherlich bessere Möglichkeiten, aber so wurde das System installiert und es funktioniert.

```
-r-sr-sr-x   1 root     lp    /usr/bin/lpr*
-r-sr-sr-x   1 root     lp    /usr/bin/lprm*
-rwxr--r--   1 root     root  /usr/sbin/lpd*
-r-xr-sr-x   1 root     lp    /usr/sbin/lpc*
drwxrwxr-x   4 root     lp    /var/spool/lpd/
drwxr-xr-x   2 root     lp    /var/spool/lpd/lp/
```

`lpd` muß momentan als root aufgerufen werden, da nur root das Recht hat, den Netzwerk-Port für lp zu belegen. Im Prinzip könnte der Daemon nach der Initialisierung seine UID ändern, wie das eigentlich jeder gut programmierte Daemon machen sollte. Der `lpd` tut dieses zur Zeit aber nicht.

8 Woher man Drucker-Software bekommt

Viele fertige Filterpakete und weitere Software rund ums Drucken sind auf folgendem Server zu finden:

```
metalab.unc.edu:/pub/Linux/system/printing/
```

Hier können auch Utilities wie psutils, a2ps, mpage, dvitodvi, flpr usw. bezogen werden.

Eine Zeitlang gab es mehrere Pakete, die alle versuchten, die Druckereinrichtung zu vereinfachen. Sie existieren wahrscheinlich alle noch, aber eines der besten und aktuellsten ist das APS-Filter Paket von Andreas Klemm, das über eine menügeführte `printcap`-Einrichtung verfügt und praktisch mit allen denkbaren Datentypen klarkommt. Wenn Ihr Distributor keine gute Druckereinrichtung mitliefert, ist APS die richtige Wahl.

9 Distributionsabhängige Lösungen

Dieser Abschnitt ist per Definition unvollständig. Sie können gerne Details Ihrer eigenen Lieblingsdistribution schicken.

9.1 Red Hat

Red Hat besitzt ein grafisches Druckerverwaltungsprogramm (im Control Panel), mit dem man sowohl entfernte als auch lokale Drucker einrichten kann. Es läßt den Benutzer einen von Ghostscript unterstützten Drucker und das Unix Device, auf das gedruckt werden soll, auswählen und installiert sowohl eine Drucker-Queue in `/etc/printcap` als auch ein kurzes PostScript- und ASCII-Filterscript, das auf `gs` und auf `nenscript` basiert. Diese Lösung funktioniert ziemlich gut, und ist für normale Anforderungen einfach einzurichten.

9.2 Sonstige Distributionen

Der Autor bittet um Informationen, was andere Distributionen machen.

10 Ghostscript

Ghostscript, das von

`http://www.cs.wisc.edu/~ghost/`

bezogen werden kann, ist ein unglaublich wichtiges Programm, um unter Linux zu drucken. Die meisten Programme unter Unix generiert PostScript und nur wenige, meistens teurere Drucker unterstützen PostScript direkt. Ghostscript allerdings ist frei und übersetzt die PostScriptbefehle in die Drucksprache des verwendeten Druckers. Wenn Ghostscript in Verbindung mit einem `lpd`-Eingabefilter verwendet wird, erhält man einen virtuellen PostScript-Drucker und das erleichtert das Leben enorm.

Ghostscript gibt es in zwei Ausführungen. Die kommerzielle Version von Ghostscript, *Aladdin Ghostscript*, darf zwar frei verwendet, aber nicht mit kommerziellen Linux-Distributionen verbreitet werden. Die kommerzielle Version ist dem freien Ghostscript normalerweise um ungefähr ein Jahr voraus; im Moment unterstützt es beispielsweise bereits das Portable Document Format (PDF) des Adobe Acrobat, während die älteren Ghostscripts dies nicht tun.

Die freie Version von Ghostscript ist *GNU Ghostscript*. Die GNU-Version ist einfach eine ältere Version von Aladdin Ghostscript, die freundlicherweise GNU überlassen wurde. Ein großes Lob hierfür an Aladdin für diese Regelung; mehr Softwarehersteller sollten freie Software so unterstützen.

Was Sie auch immer mit `gs` anfangen, vergewissern Sie sich unbedingt, es mit der Option für ausgeschaltete Dateizugriffe (`-dSAFER`) zu starten. PostScript ist eine voll funktionale Sprache und erlaubt z.B. das Ausführen von externen Programmen, was zu echten Sicherheitsproblemen führen kann, da z.B. der `lpd` ja als root läuft.

Was PDF angeht, so ist Adobes Portable Document Format in Wirklichkeit wenig mehr als organisiertes PostScript in einer gepackten Datei. Ghostscript kommt mit PDF genausogut klar wie mit PostScript. Daher könnten Sie in Ihrer Gegend der erste mit einem PDF-fähigen Drucker sein.

10.1 Aufruf von Ghostscript

Normalerweise wird Ghostscript von dem Magic-Filter aufgerufen, den Sie verwenden; allerdings ist es zur Fehlersuche oft nützlich, Ghostscript direkt aufzurufen.

`gs -help` gibt eine kurze, informative Übersicht über die Optionen und verfügbaren Treiber aus. Hierbei sollte man beachten, daß nur die Treiber angezeigt sind, die in `gs` einkompiliert worden sind. Falls der gewünschte Drucker nicht angezeigt wird, könnte es sich also lohnen, `gs` selbst zu kompilieren.

Sie können gs für Testzwecke folgendermaßen aufrufen:

```
gs optionen -q -dSAFER -sOutputFile=/dev/lp1 test.ps
```

10.2 Feineinstellung der Ausgabe

Es gibt eine Reihe von Dingen, die man machen kann, wenn die Ausgabe von `gs` nicht zufriedenstellend ist. Tatsächlich können Sie alles machen, was Ihnen Spaß macht, da Ihnen der Quelltext zur Verfügung steht.

10.2.1 Position und Größe

Die Position, die Größe und das Seitenverhältnis des Bildes auf einer Seite wird vom spezifischen Druckertreiber in Ghostscript kontrolliert. Wenn Sie feststellen, daß Ihre Seiten gestaucht oder gestreckt ausgegeben werden oder um einen Faktor 2 zu groß sind, sollten Sie vielleicht in das Quelltext-Modul Ihres Treibers schauen und all die Parameter einstellen, die Ihnen auffallen. Unglücklicherweise ist jeder Treiber anders, so daß sich schlecht pauschal sagen läßt, was wie eingestellt werden muß. Aber die meisten Treiber sind ausreichend gut kommentiert.

10.2.2 Gamma, Punktgröße usw.

Die meisten Nicht-Laserdrucker leiden unter der Tatsache, daß ihre Punkte recht groß sind, so daß sich die einzelnen Punkte teilweise überdecken. Dies führt dazu, daß Bilder zu dunkel ausgegeben werden. Wenn Sie dieses Problem haben, sollten Sie Ihre eigene Korrekturfunktion verwenden. Erstellen Sie einfach die folgende Datei im Ghostscript `lib`-Verzeichnis und fügen Sie ihren Namen dem `gs`-Aufruf genau vor der eigentlichen zu druckenden Datei hinzu. Eventuell müssen die tatsächlichen Werte der Datei an den verwendeten Drucker angepaßt werden. Niedrigere Werte führen zu einem helleren Ausdruck. Besonders, wenn der benutzte Treiber den Floyd-Steinberg Algorithmus verwendet, um Farben zu rastern, sind kleine Werte (0.20-0.15) wahrscheinlich eine gute Wahl.

```
---8<---- gamma.ps ----8<---
%!
%transfer functions for cyan magenta yellow black
{0.3 exp} {0.3 exp} {0.3 exp} {0.3 exp} setcolortransfer
---8<------------------8<---
```

Es ist durch Veränderung dieser Werte auch möglich, Drucker zu korrigieren, die irgendeine Art von Farbfehler besitzen. Wenn Sie so etwas tun, ist es empfehlenswert, die Datei `colorcir.ps` zu benutzen, die Ghostscript im `examples`-Verzeichnis als Testseite beiliegt.

11 Drucker in einem Netzwerk

Eine der Fähigkeiten von `lpd` ist es, daß man über ein Netzwerk auf Druckern drucken kann, die physikalisch an einen anderen Rechner angeschlossen sind. Mit einer sorgfältigen Kombination von Filterskripten und anderen Utilities kann man mit `lpr` transparent auf allen möglichen Druckern über alle möglichen Netzwerke drucken.

11.1 UNIX/lpd Server

Um es anderen Rechnern zu ermöglichen, auf dem eigenen Drucker zu drucken, müssen diese Rechner in `/etc/hosts.equiv` oder `/etc/hosts.ldp` aufgelistet werden. Hierbei sollte man beachten, daß `hosts.equiv` viele weitere Effekte hat; man sollte sich sicher sein, was man tut, wenn man hier einen Rechner auflistet. Man kann auch nur bestimmten Benutzern des anderen Rechners erlauben, auf dem eigenen Drucker zu drucken, indem man das `rs`-Attribut benutzt; siehe dazu auch die *lpd Manual Page* für mehr Informationen.

11.1.1 Client

Um auf einem anderen Rechner zu drucken, erstelle man einen `/etc/printcap`-Eintrag wie diesen:

```
# DeskJet 500 als Netzwerkdrucker
lp|dj|deskjet:\
        :sd=/var/spool/lpd/dj:\
        :rm=geraet.da.draussen.de:\
        :rp=druckername:\
        :lp=/dev/null:\
        :sh:
```

Hierbei gibt `rm` die Internetadresse des Rechners, an dem der Drucker angeschlossen ist, und `rp` die Warteschlange an. Beachten Sie, daß es weiterhin ein Spool-Verzeichnis auf dem lokalen Rechner gibt, das von `lpd` verwaltet wird. Wenn der Netzwerkrechner beschäftigt oder nicht erreichbar ist, bleiben Druckaufträge des lokalen Rechners im Spoolverzeichnis, bis sie gesendet werden können.

11.1.2 rlpr

Man kann auch `rlpr` benutzen, um einen Druckauftrag direkt in eine Warteschlange auf einem Netzwerkrechner zu schicken, ohne sich die Mühe zu machen, den lokalen `lpd` entsprechend einzurichten. Das ist besonders sinnvoll, wenn man nur selten auf verschiedenen Druckern druckt. Aus der Beschreibung von `rlpr`:

`rlpr` verwendet TCP/IP, um Druckaufträge an `lpd`-Server in einem Netzwerk zu schicken.

Anders als bei `lpr` müssen die Netzwerkdrucker dem lokalen Rechner nicht bekannt sein (z.B. durch `/etc/printcap`). Deshalb ist `rlpr` wesentlich flexibler und benötigt weniger Verwaltung.

`rlpr` kann überall da verwendet werden, wo ein traditionelles `lpr` verwendet werden kann und ist abwärtskompatibel zum traditionellen BSD `lpr`.

Der Hauptvorteil von `rlpr` ist die Möglichkeit, *von überall nach überall* zu drucken, ohne Rücksicht darauf, wie das System, von dem aus man drucken will, konfiguriert ist. `rlpr` kann genau wie das traditionelle lpr als Filter arbeiten, so daß Clients, die auf einem Netzwerkrechner laufen (z.B. Netscape, XEmacs, etc.) ohne größere Probleme auf Ihrem lokalen Rechner drucken können.

`rlpr` ist erhältlich von:

```
metalab.unc.edu:/pub/Linux/system/Printing/
```

11.2 Win95, WinNT, LanManager oder Samba Server

Es ist möglich, mit dem `smbclient`-Programm (Bestandteil des Samba-Pakets) eine `lpd`-Warteschlange für einen TCP/IP-basierenden SMB-Druckservice einzurichten. Samba beinhaltet hierfür ein Script namens `smbprint`. Kurz gesagt legt man eine Konfigurationsdatei für den gewünschten Drucker im Spoolverzeichnis an und installiert das `smbprint`-Script als `if`.

Der `/etc/printcap`-Eintrag sieht wie folgt aus:

```
lp|remote-smbprinter:\
    :lp=/dev/null:sh:\
    :sd=/var/spool/lpd/lp:\
    :if=/usr/local/sbin/smbprint:
```

Sie sollten die Dokumentation im `smbprint`-Skript lesen, um weitere Informationen zu erhalten.

Man kann auch `smbclient` verwenden, um eine Datei direkt an einen SMB-Druckservice zu schicken, ohne `lpd` zu benutzen. Wie dieses funktioniert, ist in der Manual Page nachzulesen.

11.3 Netware Server

Im ncpfs-Paket ist ein Programm namens `nprint` enthalten, das die gleichen Funktionen wie `smbprint` für NetWare bietet. ncpfs ist erhältlich bei:

```
metalab.unc.edu:/pub/Linux/system/filesystems/ncpfs/
```

Auszug dem LSM-Eintrag von Version 0.16:

> Mit ncpfs kann man Laufwerke auf dem Netzwerk-Server unter Linux mounten. Man kann auch über Netware-Warteschlangen drucken und Netware-Warteschlangen im Linux Drucksystem spoolen. Es wird ein Kernel 1.2.x oder 1.3.54 oder höher benötigt. ncpfs funktioniert *nicht* mit 1.3.x-Kernels unter 1.3.54.

Damit `nprint` über `lpd` arbeitet, schreibt man ein kleines Shellscript, das als `if` für die entsprechende `lpd`-Warteschlange aufgerufen wird. Dieses Script gibt die Druckdaten, die es vom dem `lpd` erhält, dann auf dem Netware-Drucker aus.

```
sub2|remote-NWprinter:\
        :lp=/dev/null:sh:\
        :sd=/var/spool/lpd/sub2:\
        :if=/var/spool/lpd/nprint-script:
```

Das `nprint-script` könnte ungefähr so aussehen:

```
#! /bin/sh
# Als erstes sollten Sie den guest Account mit keinem
# Paßwort ausprobieren!

/usr/local/bin/nprint -S net -U name -P passwd \
                      -q printq-name -
```

11.4 Apple Server

Im netatalk-Paket ist etwas Ähnliches wie `nprint` und `smbclient` enthalten. Andere Leute haben die Vorgehensweise beim Drucken über und von einem Apple-Netzwerk aus wesentlich besser beschrieben, als ich es jemals werde; bitte schauen Sie sich hierfür das *Linux Netatalk HOWTO* auf `http://thehamptons.com/anders/netatalk/` an.

11.5 Drucker mit Ethernetanschluß

Drucker für den professionellen Einsatz verfügen häufig über ein Ethernet-Interface. Die meisten dieser Geräte unterstützen direkt das lp-Protokoll, so daß man sie einfach per lpd und der `rp`-Option ansprechen. Man sollte den Anweisungen folgen, die mit dem Drucker oder dem Netzwerkadapter des Druckers geliefert wurden. Ein Drucker von HP könnte z.B. mit einem solchem `printcap`-Eintrag arbeiten:

```
lj-5|remote-hplj:\
        :lp=/dev/null:sh:\
        :sd=/var/spool/lpd/lj-5:\
        :rm=printer.name.com:rp=raw:
```

HP LaserJet Drucker mit JetDirect Interface stellen im allgemeinen zwei eingebaute Queues zur Verfügung:

raw

Akzeptiert Druckdaten in PCL und eventuell PostScript.

text

Kann zum Druck von reinen ASCII-Dateien verwendet werden, wobei der beschriebene Treppeneffekt automatisch korrigiert wird.

Wenn Sie eine JetDirect Plus3 3-Port Box besitzen, besitzen die Queues die Bezeichnungen »raw1«, »text2« usw.

In einem großen Netz, in dem einige Drucker kein PostScript unterstützen, könnte es sinnvoll sein, einen Print-Server einzurichten, auf dem alle Rechner drucken und auf dem alle Ghostscript-Jobs laufen.

Dies erlaubt Ihrem Linux-Rechner auch, als Spool-Server zu arbeiten, so daß die Netzwerkbenutzer ihre Druckaufträge schnell beenden und weiterarbeiten können, ohne darauf warten zu müssen, daß der Drucker erst fremde Aufträge fertigstellt.

Um dies zu erreichen, richten Sie eine Queue auf Ihrem Linux-Rechner ein, die auf den HP Laserjet mit Ethernetkarte verweist (siehe oben). Nun stellen Sie alle Clientrechner in Ihrem LAN so ein, daß sie die Linux-Queue benutzen (z.B. lj-5 im Beispiel oben).

Anscheinend beachten einige HP Netzwerkdrucker die Deckblatteinstellungen nicht, die von den Clients geschickt werden; Sie können das intern generierte Deckblatt abschalten, indem Sie sich per telnet auf den Drucker einloggen, zweimal Return drücken und danach `banner: 0` gefolgt von `quit` eingeben. Sie können auch andere Einstellungen auf diese Art verändern; tippen Sie `?` ein, um eine Liste angezeigt zu bekommen.

11.5.1 Alte HP-Drucker

Einige Drucker und externe Druckserver haben zwar eine Ethernetschnittstelle, unterstützen jedoch nicht das `lp`-Protokoll. Oftmals finden statt dessen spezielle Treiber Verwendung, die für Linux natürlich nicht existieren. Einige dieser Geräte speichern Druckaufträge auch nicht zwischen, so daß es einen Rechner geben muß, der Druckjobs sammelt und nacheinander an den Drucker schickt.

Erwähnenswert in dieser Kategorie sind frühe JetDirect und einige JetDirectEx Karten. Grundsätzlich muß man zum Drucken auf diesen Drucker eine TCP-Verbindung zu dem Drucker auf einem bestimmten Port (typischerweise 9100, oder 9100, 9101 und 9102 für 3-Port Boxen) öffnen und den Druckauftrag hier durch schicken. Das kann u.a. in Perl implementiert werden:

```
#!/usr/bin/perl
# Dank gebührt Dan McLaughlin, der die Originalversion
# dieses Skriptes geschrieben hat.

$fileName = @ARGV[0];

open(IN,"$fileName") ||
  die "Kann Datei $fileName nicht öffnen";

$dpi300 = "\x1B*t300R";
$dosCr = "\x1B&k3G";
$ends = "\x0A";

$port =  9100 unless $port;
$them = "bach.sr.hp.com" unless $them;

$AF_INET = 2;
$SOCK_STREAM = 1;
$SIG{'INT'} = 'dokill';
$sockaddr = 'S n a4 x8';

chop($hostname = `hostname`);
($name,$aliases,$proto) = getprotobyname('tcp');
($name,$aliases,$port) = getservbyname($port,'tcp')
    unless $port = /^\d+$/;;
($name,$aliases,$type,$len,$thisaddr) =
    gethostbyname($hostname);
($name,$aliases,$type,$len,$thataddr) =
    gethostbyname($them);
$this = pack($sockaddr, $AF_INET, 0, $thisaddr);
```

```
$that = pack($sockaddr, $AF_INET, $port, $thataddr);

if (socket(S, $AF_INET, $SOCK_STREAM, $proto)) {
#    print "socket ok\n";
}
else {
    die $!;
}
# Gibt dem Socket eine Adresse
if (bind(S, $this)) {
#    print "bind ok\n";
}
else {
    die $!;
}

# Ruft den Server auf.

if (connect(S,$that)) {
#    print "connect ok\n";
}
else {
    die $!;
}

# Setzt Befehlspuffer für den Socket.

select(S); $| = 1; select(STDOUT);

#   print S "@PJL ECHO Hi $hostname! $ends";
#   print S "@PJL OPMSG DISPLAY=\"Job $whoami\" $ends";
#   print S $dpi300;

# Durch Aufteilung Deadlock verhindern.

if($child = fork) {
    print S $dosCr;
    print S $TimesNewR;

    while (<IN>) {
        print S;
    }
    sleep 3;
    do dokill();
} else {
    while(<S>) {
        print;
    }
}

sub dokill {
    kill 9,$child if $child;
}
```

11.6 Filter für Netzwerkdrucker

Eine Eigenart von `lpd` ist, daß ein `if` für Netzwerkdrucker nicht aufgerufen wird. Wenn das nötig sein sollte, kann man eine doppelte Warteschlange einrichten und den Job an die zweite Warteschlange weitergeben. Ein Beispiel für eine passende `printcap` könnte so aussehen:

```
lj-5:remote-hplj:\
        :lp=/dev/null:sh:\
        :sd=/var/spool/lpd/lj-5:\
        :if=/usr/lib/lpd/filter-lj-5:
lj-5-remote:lp=/dev/null:sh:rm=printer.name.com:\
        :rp=raw:sd=/var/spool/lpd/lj-5-raw:
```

Das passenden `filter-lj-5`-Script wäre dabei:

```
#!/bin/sh
gs <options> -q -dSAFER -sOutputFile=- - | \
        lpr -Plj-5-remote -U$5
```

Die Option `-U` von `lpr` funktioniert nur, wenn `lpr` als ein Daemon gestartet wurde und setzt den Namen des Auftraggebers in der weitergegebenen Warteschlange. Man sollte eventuell eine stabilere Methode verwenden, um den Benutzernamen zu ermitteln, denn manchmal ist es nicht Argument 5. Weitere Informationen finden sich in der Manual Page zur `printcap`.

11.7 Windows Client

Drucken von einem Windows (oder wohl auch OS/2) Client auf eine Linuxmaschine wird über SMB direkt unterstützt, wenn man das Samba-Paket benutzt, das auch die Benutzung Ihres Linux-Filesystems von Windows aus ermöglicht.

Samba beinhaltet eine ziemlich vollständige Dokumentation, und es gibt die recht gute *Samba-FAQ*, welche sich ebenfalls damit beschäftigt. Sie können entweder einen Magic-Filter auf dem Linuxrechner einrichten und PostScript darauf ausgeben, oder druckerspezifische Treiber auf allen Windows-Maschinen installieren und eine Queue ganz ohne Filter benutzen. Sich auf die Windows-Treiber zu stützen sorgt manchmal für ein besseres Druckbild, ist aber etwas mehr administrativer Aufwand bei vielen Windows-Rechnern. Sie sollten es daher zuerst mit PostScript versuchen.

11.8 Apple Client

Netatalk unterstützt das Drucken von Apple-Clients über EtherTalk. Siehe hierfür auch die *Netatalk HOWTO* Page unter:

```
http://thehamptons.com/anders/netatalk/
```

11.9 Netware Client

Das ncpfs-Paket beinhaltet einen Daemon namens pserver, der benutzt werden kann, um Dienste für eine NetWare Druck-Queue zur Verfügung zu stellen. Soweit ich es verstanden habe, benötigt dieses System ein »Bindery«-basiertes NetWare, z.B. 2.x, 3.x oder 4.x, mit eingeschaltetem Bindery-Zugriff.

Für mehr Informationen über ncpfs und sein pserver-Programm sollten Sie einen Blick auf die ncpfs FTP-Site unter folgender Adresse werfen:

```
ftp.gwdg.de:/pub/linux/misc/ncpfs/
```

12 GDI-Drucker

Wie schon vorher erklärt wurde, werden einige Drucker unter Linux nicht unterstützt, weil sie keine normale Druckersprache benutzen, sondern den Hauptprozessor benutzen, um das Bild auszurechnen, das dann mit fester Geschwindigkeit an den Drucker übertragen wird. Manchmal verstehen solche Drucker auch etwas anderes als normales PCL. Meistens ist dieses jedoch nicht der Fall, statt dessen liegen den Druckern manchmal Treiber für Windows bei, die PCL per Software emulieren, was für andere Betriebssysteme natürlich nicht sehr hilfreich ist. In einigen extremen Fällen besitzen die Drucker nicht einmal eine normale Parallelverbindung, sondern benötigen auch noch spezielle Treiber für die Ansteuerung der Schnittstelle.

In jedem Fall gibt es einige mögliche Auswege, wenn Sie eine solche »Zitrone« erwischt haben. Der beste Ausweg dürfte sein, den Drucker an einen Windowsbenutzer weiterzuverkaufen.

12.1 Der Ghostscript Windows Druckertreiber

Es gibt mittlerweile einen Windows Druckertreiber (namens mswinpr2), der einen Druckauftrag durch Ghostscript leitet, bevor er ihn druckt. Das Prinzip ist ähnlich einem `if`-Filter bei einem `lpd`. Es gibt auch einen neuen Ghostscripttreiber, der nicht eine spezielle Druckersprache verwendet, sondern über die Windows-Druckschnittstelle druckt. Zusammengenommen sollte dies einem Windows-Rechner erlauben, PostScript auf einem Windows-only-Drucker über den Herstellertreiber auszugeben.

Wenn Sie dieses zum Laufen bringen, können Sie den Anweisungen oben folgen, um von Linux aus über das Netzwerk auf einen Windows-Drucker zu drucken und es so Rechnern zu ermöglichen, auf Ihrer »Zitrone« zu arbeiten.

Sämtliche nötige Software sollte auf der Ghostscript-Homepage, siehe Abschnitt 10, verfügbar sein.

12.2 pbm2ppa

Einige HP Drucker benutzen die *Printing Performance Architecture*. Diese wird über einen Umweg durch den `pbm2ppa` Umsetzer von Tim Norman unterstützt. Im Grunde benutzt man Ghostscript, um PostScript in eine Bitmapgrafik im pbm-Format umzuwandeln, und dann `pbm2ppa`, um aus dieser das druckerspezifische ppa-Format zu generieren, das dann auf dem Drucker ausgegeben werden kann. Mittlerweile könnte es dieses Programm auch als Ghostscript Druckertreiber geben.

Die ppa-Software kann von der ppa Homepage heruntergeladen werden, die unter folgender Adresse zu finden ist:

```
http://www.rpi.edu/~normat/technical/ppa/
```

`pbm2ppa` unterstützt einige Modelle der Serien HP 720, 820 und 1000; bitte lesen Sie die Anleitung zum Paket für mehr Details zur ppa-Drucker Unterstützung.

13 Faxen

13.1 Faxen mit einem Modem

Es gibt einige Faxprogramme, mit denen man faxen und Dokumente empfangen kann. Eines der komplexesten ist Sam Lefflers *HylaFax*, erhältlich bei SGI:

```
ftp.sgi.com
```

Es unterstützt alle möglichen Funktionen, von mehreren Modems bis zum Broadcasting.

SuSE liefert einen Java HylaFax-Client, der angeblich auf jeder Java Plattform (inklusive Windows und Linux) läuft. Es gibt auch Fax-Clients, die nicht in Java geschrieben worden sind, für die meisten Plattformen; Linux kann mit ziemlicher Sicherheit Ihren Netzwerk-Fax-Anforderungen genügen.

Ebenfalls verfügbar, und die bessere Wahl für die meisten Linuxrechner ist *efax*. Dieses ist ein einfaches Programm, das Faxe sendet. Das getty-Programm *mgetty* kann Faxe empfangen und mit manchen Modems auch einen Anrufbeantworter emulieren.

13.2 Remote Printing Service

Einige Firmen und Organisationen bitten im Internet das Versenden und teilweise auch das Empfangen von Faxen per EMail oder WWW an. Teilweise werden diese Leistungen sogar kostenlos angeboten, wobei dann neben dem eigentlichen Dokument oft auch Werbung übermittelt wird.

Es werden hierbei Übertragungsformate wie ASCII oder PostScript unterstützt. Weitere Informationen finden Sie z.B. auf der *Remote Priting* WWW-Seite unter:

```
http://www.tpc.int/
```

14 Textverarbeitung

Hier kommt man in ein wahres Rattennest an Software. Grundsätzlich kann man nicht nur auf die speziell für Linux erhältlichen Programme zurückgreifen, sondern z.B. auch ältere Programme für Windows oder DOS über einen Emulator benutzen. Dieses ist allerdings aus verschiedenen Gründen nicht ratsam.

14.1 Formatierungssprachen

Die meisten Formatierungssprachen sind am besten für große oder sich wiederholende Projekte geeignet, in denen man dem Computer die Kontrolle über das Layout überlassen möchte, damit alles einheitlich aussieht. Folgende Sprachen haben eine große Verbreitung unter Linux Benutzern:

`nroff`

Diese war eine der ersten Formatierungssprachen für UNIX. Manual Pages sind das häufigste Beispiel für Texte, die mit *roff-Makros formatiert wurden; einige Leute schwören darauf, aber nroff hat eine recht komplizierte Syntax und findet deshalb heute kaum noch Verwendung. Man sollte aber trotzdem wissen, daß man eine Manual Page mit groff direkt nach PostScript konvertieren kann. Die meisten man-Befehle machen das mit

```
man -t foo | lpr
```

TeX

TeX und das Makropaket LaTeX gehören zu den am häufigsten verwendeten Formatierungssprachen auf UNIX-Systemen. LaTeX wird vor allem für technische Werke eingesetzt, da es eigentlich das einzige Textverarbeitungssystem ist, mit dem man mathematische Formel vernünftig setzen kann und das ohne Probleme mit sehr großen Dokumenten umgehen kann. Ein weiterer Vorteil ist, daß LaTeX eigentlich für fast jeden Rechnertyp diese Erde verfügbar ist und selbst auf sehr langsamen Rechnern problemlos läuft.

TeX wird, vor allem von Leuten, die es noch nie benutzt haben, nachgesagt, es wäre schwierig zu bedienen und im Zeitalter der Officeprodukte schlicht veraltet. Solchen Aussagen sollte man nicht viel Bedeutung schenken. Die Grundfunktionen erlernt man in wenigen Stunden. Der Hauptvorteil von LaTeX ist, daß der Benutzer im

Gegensatz zu vielen anderen Programmen keine Layout-Kenntnisse haben muß, sondern LaTeX im Prinzip nur mitteilen muß, um was für einen Text es sich handelt, also z.B. um eine Überschrift. Die Formatierung übernimmt LaTeX.

Das Ausgabeformat von TeX ist `dvi` und kann mit `dvips` oder `dvilj` nach PostScript bzw. Hewlett Packards PCL konvertiert werden.

SGML

Bei SGML werden in einem noch viel stärkeren Maß wie bei LaTeX der Inhalt und die Formatierung getrennt. Der Benutzer beschreibt im Idealfall nur den Inhalt der einzelnen Zeilen. Nach Fertigstellung des Textes wird dieser dann mit speziellen Konvertern in die eigentlichen Ausgabeformate umgesetzt.

Durch die Trennung von Inhalt und Formatierung ist es zum einen sehr leicht, die Dateien in viele verschiedene Ausgabeformate zu konvertieren. Außerdem ist es sehr leicht, die eigentliche Formatierung von einigen 1000 Dateien nachträglich mit wenigen Befehlen zu ändern.

Unter Linux findet zur Zeit vor allem das freie sgml-tools Paket Verwendung, mit dem z.B. alle HOWTOs geschrieben werden:

```
http://www.sgmltools.org/
```

Dieses Paket ist jedoch nicht vollständig fehlerfrei und die Trennung von Inhalt und Formatierung ist bisher nur sehr begrenzt realisiert. Zur Zeit wird daran gearbeitet, daß Paket auf die bekannte DocBook-DTD umzustellen.

14.2 WYSIWYG

Mittlerweile gibt es keinen Mangel an WYSIWYG Textverarbeitungssoftware mehr. Es gibt mehrere vollständige Office-Pakete:

StarOffice

Die deutsche Firma StarDivision bietet ihr StarOffice für Linux zum kostenlosen Download im Internet an. Allerdings darf diese Version nur privat genutzt werden und leider auch nicht mit den Linux Distributionen zusammen verbreitet werden. Eine kommerzielle Version wird von Caldera verkauft.

StarOffice ist vor allem für Benutzer interessant, die mit diesem Programm bereits unter anderen Betriebssystemen wie Windows gearbeitet haben. Gegen die Nutzung sprechen zum einen die enormen Anforderungen an die eigene Hardware, eine schnelle 586er CPU mit 64 MB Speicher sind das absolute Minimum, und zum anderen die Mißachtung fast aller UNIX-Standards. So bringt das Programm einen eigenen Window-Manager und Desktop mit, die beide, wie die Oberfläche selbst auch, Win95 nachempfunden sind.

LyX

LyX ist ein Frontend für LaTeX, das vielversprechend aussieht. Auf der LyX Homepage gibt es weitere Informationen:

```
http://www-pu.informatik.uni-tuebingen.de/users/ettrich/
```

Es gibt auch eine KDE-gerechte Version von LyX namens Klyx; der Autor von LyX und der Initiator des KDE-Projekts sind ein und dieselbe Person.

Das Andrew User Interface System

AUIS beinhaltet `ez`, einen WYSIWYG-artigen Editor mit grundlegenden Textverarbeitungsfähigkeiten, HTML und voller MIME-eMail- und Newsgroup-Unterstützung. Leider wird AUIS nicht länger gepflegt.

Koffice

Das KDE-Projekt arbeitet an einem kompletten Office-Paket. Dieses dürfte zur Zeit aber noch nicht benutzbar sein.

GNOME

Auch das GNOME-Projekt arbeitet an mehreren Office-Tools mit GNU-Lizenz. Allerdings sind noch keine verfügbar.

WordPerfect

Es existiert eine auf Linux portierte Version von Corels WordPerfect. Informationen hierzu finden sich unter:

`http://www.corel.com`

Corel hat außerdem bekanntgegeben, daß man plant, zukünftig weitere Produkte nach Linux zu portieren.

Applixware

Eine recht weite Verbreitung hat das kommerzielle Officepaket Applixware unter Linux-Benutzern gefunden. Das Programmpaket kann recht preiswert von allen größeren Linux-Vertreibern bezogen werden. Applix ist für viele Plattformen verfügbar und beinhaltet eine Tabellenkalkulation, eine Textverarbeitung, ein Präsentationspaket usw. Es gibt eine weitere Version von Applix in Java namens ApplixAnyWhere. Weitere Informationen zu diesen Paketen sind unter

`http://www.applixware.com`

erhältlich.

15 Voransicht von druckbaren Sachen auf dem Bildschirm

Unter Linux können viele Datenformate, die für den Druck fertig formatierte Dokumente erhalten, nicht nur gedruckt, sondern auch auf dem Bildschirm angezeigt werden. Dies kann zum einen bei der Erstellung eines Dokumentes nützlich sein, da dieses dann nicht immer gleich zur Korrektur ausgedruckt werden muß, zum anderen können so auch die Dokumente anderer Autoren auf dem Bildschirm betrachtet werden.

15.1 PostScript

Ghostscript kann nicht nur zur Ausgabe von PostScriptdateien auf Druckern verwendet werden, sondern es ermöglicht mittels des X11-Treibers auch, solche Dateien auf dem Bildschirm anzuzeigen. Allerdings wird man in den wenigsten Fällen Ghostscript direkt aufrufen. Viel mehr benutzt man einen der beiden PostScript-Betrachter:

- Ghostview
- gv

Beide Programme benutzen intern Ghostscript. Sie erlauben es, bequem zwischen den einzelnen Seiten zu blättern, bestimmte Seiten auszuwählen, um sie zu speichern oder zu drucken und vieles mehr. Es empfiehlt sich, nur gv anstelle von Ghostview zu verwenden, da gv über einige zusätzliche Möglichkeiten verfügt und im Gegensatz zu Ghostview noch aktiv entwickelt wird.

Neben PostScript kann gv auch PDF-Dateien anzeigen.

15.2 TeX DVI

TeX DeVice-Independant-files (geräteunabhängige Dateien) können unter X11 mit `xdvi` betrachtet werden. Um einige Beschränkungen von TeX zu umgehen, binden einige Makropakete direkt PostScript-Befehle in die DVI-Dateien ein. Da `xdvi` diese Daten nicht selbst auswerten kann, wird auch hier auf Ghostscript zurückgegriffen, so daß es empfehlenswert ist, Ghostscript zu installieren, wenn man TeX benutzt.

Es gibt ebenfalls einen VT100-Treiber. Er wird `dgvt` genannt. `Tmview` benutzt die svgalib statt X11, um die Dateien anzuzeigen. Da die svgalib jedoch nicht mehr weiterentwickelt wird und auch eine Insellösung von Linux ist, sollte besser auf `xdvi` zurückgegriffen werden.

15.3 Adobe PDF

Adobes Acrobat Reader gibt es auch für Linux. Er kann von der Adobe-WWW-Seite bezogen werden:

```
http://www.adobe.com/
```

Neben dem Programm von Adobe gibt es noch das freie Programm `xpdf`, das jedoch etliche Einschränkungen im Vergleich zu dem Programm von Adobe hat und deshalb nicht zu empfehlen ist. Eine gute Alternative stellt allerdings gv mit Ghostscript da. So berechnet gv immer die komplette Seite, so daß man wesentlich besser scrollen kann. Nachteilig ist allerdings, daß es keine Suchfunktion gibt.

16 Serielle Drucker unter lpd

16.1 Einstellungen in der printcap

`lpd` besitzt 5 Attribute, die in der `/etc/printcap` eingestellt werden können, um die Einstellungen einer seriellen Schnittstelle zu kontrollieren, an die der Drucker angeschlossen ist. Die Beschreibung der Attribute ist in der *printcap Manual Page* zu finden, wobei man vor allem die Bedeutung von `br#`, `fc#`, `xc#`, `fs#` und `xs#` beachten sollte. Die letzten vier dieser Attribute sind Bitmaps, die für die Einstellungen dieser Schnittstelle stehen. Das `br#`-Attribut ist einfach die Baudrate, z.B. '`br#9600`':

Es ist sehr einfach, die `stty`-Einstellungen in `printcap` Flags zu übersetzen. Wenn nötig, studieren Sie nun bitte die Manual Page von `stty`.

Benutzen Sie `stty`, um die Schnittstelle zum Drucker so einzustellen, daß man eine Datei über sie mit cat an den Drucker schicken kann, und diese dann korrekt ausgegeben wird. So könnte die Ausgabe von '`stty -a`' für eine Schnittstelle z.B. aussehen.

```
# stty -a < /dev/ttyS2
speed 9600 baud; rows 0; columns 0; line = 0;
intr = ^C; quit = ^\; erase = ^?; kill = ^U; eof = ^D;
eol = <undef>; eol2 = <undef>; start = ^Q; stop = ^S;
susp = ^Z; rprnt = ^R; werase = ^W; lnext = ^V; min = 1;
time = 0;
-parenb -parodd cs8 hupcl -cstopb cread -clocal -crtscts
-ignbrk -brkint -ignpar -parmrk -inpck -istrip -inlcr
-igncr -icrnl ixon -ixoff -iuclc -ixany -imaxbel
-opost -olcuc -ocrnl -onlcr -onocr -onlret -ofill -ofdel
nl0  cr0 tab0 bs0 vt0 ff0
-isig -icanon -iexten -echo -echoe -echok -echonl -noflsh
-xcase -tostop -echoprt -echoctl -echoke
```

Die einzigen Unterschiede zwischen diesen Einstellungen und der Initialisierung beim Booten sind `-clocal`, `-crtscts` und `ixon`. Die Ausgabe bei ihrer Schnittstelle kann durchaus anders aussehen, da dieses von der Flußkontrolle Ihres Druckers abhängt.

Wirklich verwendet wird `stty` auf eine etwas ungewöhnliche Weise. Ursprünglich war `stty` dafür gedacht, die Einstellungen eines Terminals anzupassen. Deswegen gibt `stty` die Steuerzeichen auf der Standardausgabe aus. Um mit dem Programm die Einstellungen einer beliebigen seriellen Schnittstelle zu manipulieren, muß deshalb wie oben das <-Zeichen verwendet werden.

Sobald die `stty`-Einstellungen richtig sind, man also mit

```
cat datei > /dev/ttyS2
```

eine Datei an der Drucker, hier an `/dev/ttyS2`, schicken kann, sollte man die Datei `/usr/src/linux/include/asm-i386/termbits.h` betrachten. Diese Datei enthält viele `#define`- und einige `struct`-Zeilen. Man betrachte den Abschnitt, der mit

```
/* c_cflag bit meaning */
#define CBAUD   0000017
```

anfängt. Dieser Abschnitt listet die Bedeutungen der `fc#`- und `fs#`-Teile auf. Ihnen wird auffallen, daß die Namen (nach den Baudraten) mit einer der `stty`-Ausgabeleitungen übereinstimmen. Habe ich nicht gesagt, daß es leicht sein würde?

Achten Sie auf die Einstellungen, die mit einem »-« in der `stty`-Ausgabe anfangen. Wenn man alle diese Nummern aufsummiert (sie sind oktal), erhält man die Bits, die man löschen will. Also ist das die `fc#`-Einstellung. Erinnern Sie sich daran, daß Sie diese Bits direkt nach dem Löschen setzen werden, also können Sie einfach `fc#0177777` verwenden.

Jetzt macht man das gleiche für die Einstellungen, die kein »-« vor sich haben. In dem Beispiel sind die wichtigsten `CS8` (0000060), `HUPCL` (0002000) und `CREAD` (0000200). Beachten Sie auch die Flags für die Baudrate. Zählen Sie alle zusammen und Sie bekommen in diesem Beispiel 0002275 heraus. Dieses ist der Wert für die `fs#`-Option.

Machen Sie das gleiche mit `set` und `clear` für den nächsten Abschnitt der Includedatei, `c_lflag bits"`

.

16.2 Ältere serielle Drucker, die Zeichen verschlucken

Jon Luckey hat festgestellt, daß einige ältere serielle Drucker mit sehr schlechten seriellen Interfaces und kleinen Buffern *wirklich* Stop meinen, wenn sie das mit ihrer Flußkontrolle sagen. Er fand heraus, daß sich das Problem lösen läßt, wenn der FIFO-Buffer des 16550-UARTs mit `setserial` ausgeschaltet wird. Hierfür gibt man einfach den UART-Typ 8250 an.

17 Dank

Die Informationen über `smbprint` sind aus einem Artikel von Marcel Roelofs (`marcel@paragon.nl`).

Die Informationen über `nprint`, um NetWare-Drucker zu verwenden, wurden von Michael Smith (`mikes@bioch.ox.ac.uk`) zur Verfügung gestellt.

Der Abschnitt *Serielle Drucker und lpd* ist von Andrew Tefft (`teffta@engr.dnet.ge.com`).

Der Teil über die Gammakorrektur und weitere Tips rund um Ghostscript stammt von Andreas (`quasi@hub-fue.franken.de`).

Die zwei Abschnitte über die 30-Sekunden-`closing_wait` des seriellen Treibers sind von Chris Johnson (`cdj@netcom.com`).

Robert Hart hat ein paar hervorragende Abschnitte über das Einrichten eines Printservers für vernetzte HPs eingeschickt, die ich wörtlich verwendet habe.

Besonderer Dank geht an die vielen von euch, die auf Tippfehler, falsche URLs und Sachfehler im Dokument im Laufe der Jahre aufmerksam gemacht haben.