

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, 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	3
1.1	Anmerkungen zur Übersetzung	3
1.2	Geschichte	4
1.3	Geschichte der deutschen Übersetzung	4
1.4	Copyright	4
2	Wie druckt man?	4
3	Kernel-Drucker-Devices	4
3.1	Das lp-Device	5
3.2	Das parport-Device (Kernelversionen $\geq 2.1.33$)	5
3.3	Serielle Devices	6
4	Unterstützte Drucker	6
4.1	Drucker-spezifische Bemerkungen	7
5	Welche Spool-Software?	7
6	Grundlagen	8
7	Grundeinstellungen	8
7.1	Traditionelle lpd-Konfiguration	8
7.2	File Permissions	10
8	Woher man Drucker-Software bekommt	10
9	Distributionsabhängige Lösungen	10
9.1	Red Hat	10
9.2	Sonstige Distributionen	10
10	Ghostscript	11
10.1	Aufruf von Ghostscript	11

10.2 Feineinstellung der Ghostscript-Ausgabe	11
10.2.1 Position und Größe der Ausgabe	11
10.2.2 Gamma, Punktgröße usw.	12
11 Wie man auf einen Drucker über ein Netzwerk druckt	12
11.1 Auf einen UNIX/lpd Host	12
11.1.1 Mit lpd	12
11.1.2 Mit rlpd	13
11.2 Auf einen Win95, WinNT, LanManager oder Samba-Drucker	13
11.3 Auf einen Netware-Drucker	13
11.4 Auf einen EtherTalk- (Apple-) Drucker	14
11.5 Auf einen HP oder anderem Ethernet-Drucker	14
11.5.1 Auf alten HPs	15
11.6 Ein <i>if</i> für Netzwerkdrucker	16
11.7 Von Windows aus.	17
11.8 Von einem Apple aus.	17
11.9 Von Netware aus.	17
12 Windows-only printers	17
12.1 Der Ghostscript Windows Druckertreiber	17
12.2 Das pbm2ppa Programm	18
13 Wie man auf einem Faxgerät druckt	18
13.1 Mit einem Faxmodem	18
13.2 Mit dem Remote Printing Service	18
14 Wie man etwas Druckbares herstellt.	18
14.1 Formatierungssprachen	19
14.2 WYSIWYG	19
15 Voransicht von druckbaren Sachen auf dem Bildschirm	20
15.1 PostScript	20
15.2 TeX dvi	20
15.3 Adobe PDF	20
16 Serielle Drucker unter lpd	21
16.1 Einstellungen in der printcap	21
16.2 Ältere serielle Drucker, die Zeichen verschlucken	22
17 Credits	22

1 Einleitung

Da dies eine Übersetzung ist, werden die Versionsnummern beibehalten, d.h. die deutsche Übersetzung beginnt mit Version 3.9.

Die Drucker HOWTO sollte alles beinhalten, was Sie wissen müssen, um Ihnen dabei zu helfen, Druckdienste auf Ihrer/n Linux-Maschine(n) einzurichten. Wie im Leben ist es etwas komplizierter als in der Klick-Herum Welt von Microsoft und Apple, aber es ist dafür ein wenig flexibler und in großen Netzen auf jeden Fall leichter zu handhaben.

Dieses Dokument ist strukturiert, so 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 Kapiteln 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 verloren gegangen. Das ist Absicht, da die früheren HOWTOs 60 Seiten und mehr umfaßten, und den Erzählfluß einer toten Schildkröte hatten. Wer hier keine passende Antwort finden, sollte a) die vorhergehende Version auf der Printing HOWTO Home Page (<http://www.picante.com/~gtaylor/pht/>) durchsuchen und b) dem Autor eine Nachricht hinterlassen, was in der 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, sie ist auch auf `sunsite.unc.edu` im LDP-Bereich erhältlich.

Die deutsche Version ist auf

<http://www.tu-harburg.de/~semb2204/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 gelassen worden:

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`.

Man-page

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

Permissions

Zugriffsrechte von Dateien (Files). Bestimmt, wem die Datei gehört, wer wie darauf zugreifen darf u.a.

1.2 Geschichte

Dies ist die dritte Generation, d.h. die dritte komplette Neufassung der *Printing HOWTO*. Die Geschichte der PHT ist diese:

1. Ich schrieb die *Printing HOWTO* als Antwort auf zu viele Fragen zum Drucken in `comp.os.linux`, und postete sie. Dieses Posting war einige Monate vor dem HOWTO-Projekt und war damit die erste FAQ, die eine 'HOWTO' genannt wurde. Diese Version war in reinem ASCII.
2. Nachdem Ich dem HOWTO-Projekt beitrug, wurde die *Printing HOWTO* mit einem Lpd FAQ von Brian McCauley (`B.A.McCauley@bham.ac.uk`) verschmolzen, und wir arbeiteten als Co-Autoren an der PHT etwa 2 Jahre lang weiter zusammen. In dieser Zeit wurde auch die Arbeit von Karl Auer (`Karl.Auer@anu.edu.au`) eingebaut. Diese Generation der PHT war in TeXinfo, und erhältlich in PS, HTML, Ascii und Info.
3. Nachdem die PHT für über ein Jahr vor sich hin gerotet war, und ein Versuch, jemand anders für die Weiterführung zu gewinnen, scheiterte, entstand diese Neufassung. Diese Generation der PHT ist in Linuxdoc-SGML (auch bekannt als die sgml-Tools).

1.3 Geschichte der deutschen Übersetzung

Die deutsch-sprachigen Versionen sind reine Übersetzungen, die Geschichte stimmt daher weitestgehend mit der oben genannten Geschichte überein. Die erste übersetzte HOWTO war die Version 3.9, der Übersetzer bemüht sich, neue englisch-sprachige Versionen möglichst schnell einzubauen.

1.4 Copyright

Dieses Dokument ist urheberrechtlich geschützt. Das Copyright für die englische *Printing HOWTO*, auf der dieses Dokument basiert, liegt bei Grant Taylor. Das Copyright für die deutsche Übersetzung liegt bei Jens Vonderheide und Robert Fendt.

Das Dokument darf gemäß der GNU *GPL* kostenlos verbreitet werden. Das bedeutet, daß der Text sowohl über elektronische wie auch physikalische Medien ohne die Zahlung von Lizenzgebühren verbreitet werden darf, solange dieser Copyright Hinweis nicht entfernt wird. Eine kommerzielle Verbreitung ist erlaubt und sogar erwünscht. Bei einer Verbreitung in Papierform ist das deutsche HOWTO-Projekt hierüber zu informieren.

2 Wie druckt man?

Wenn lpd bereits so eingerichtet ist, daß es druckt, oder der System-Betreuer es bereits eingerichtet hat, dann muß nur der Gebrauch des `lpr`-Kommandos erlernt werden. Die *Printing Usage HOWTO* bezieht sich darauf, und auf einige weitere Kommandos, die man kennen sollte.

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. Also: weiterlesen!

3 Kernel-Drucker-Devices

3.1 Das lp-Device

Das Linux-Kernel (<=2.1.32), falls das lp-Device eincompiliert 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, das 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 `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` (<http://www.picante.com/~gtaylor/pht/man/tunelp.8.html>), mit dem man als root Interrupts, Geschwindigkeit und anderes am lp-Device einstellen kann.

Wenn die Option in Kernel 2.x oder einigen 1.3.x-Kernels compiliert wurde, kann eine `lp=` Option (über LI-LO/LOADLIN) gewählt werden, um die Portadressen und Interrupts einzustellen:

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

Beispiel: `lp=0x378,0` oder `lp=0x278,5,0x378,7 **`

** Für alle diejenigen, die sich nie an die Standard-Porteinstellungen erinnern können, wenn sie sie brauchen (so wie ich): sie entsprechen dem zweiten Beispiel oben. Der andere Port (lp0) liegt bei 0x3bc; keine Ahnung, was für einen Interrupt er benutzt.

Hinweis: Wenn diese Funktion benutzt wird, müssen **alle** Ports angegeben werden, die berücksichtigt werden sollen, es gibt keine Voreinstellungen. Man kann einen eingebauten Treiber mit `lp=0` deaktivieren.

Wenn das Device als Modul geladen wird (**nur** in 2.x und späten 1.3.x Kernels), können die Adressen und Interrupts in der üblichen Syntax im `insmod`-Kommando angegeben werden (oder in `/etc/conf.modules` für den `kerneld`). Die Parameter sind `io=port0,port1,port2` und `irq=irq0,irq1,irq2`. Auf der Man-page zu `insmod` gibt es weitere Informationen.

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

3.2 Das parport-Device (Kernelversionen >= 2.1.33)

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`-Device. Die Erweiterung um das `parport`-Device beseitigt einige der Probleme, die den alten lp Treiber quälen - es kann den Port mit anderen Treibern teilen, es ordnet verfügbaren Parallelports dynamisch Device-Nummern zu, anstatt eine feste Zuordnung zu erzwingen, und einiges mehr.

Die Verfügbarkeit des `parport`-Device hat eine ganze Reihe neuer Treiber für den Parallelport ermöglicht, so z.B. für ZIP-Laufwerke, externe CD-ROMs und Festplatten usw. Einige davon sind auch für Kernel 2.0.x verfügbar; schauen Sie sich im Web um.

Ich werde den `parport`-Treiber vollständiger beschreiben, sobald ich ihn selbst benutze; in der Zwischenzeit können Sie das File `Documentation/parport.txt` in den Kernelquellen oder auf <http://www.cyberelk.demon.co.uk/parport/parport.txt> lesen, oder einen Blick auf die *parport-Webseite* (<http://www.cyberelk.demon.co.uk/parport.html>) werfen.

3.3 Serielle Devices

Serielle Devices heißen üblicherweise unter Linux `/dev/ttyS1` oder ähnlich. Das Programm `stty` (<http://www.picante.com/~gtaylor/pht/man/stty.1.html>) erlaubt es, interaktiv die Einstellungen für die seriellen Schnittstellen anzusehen oder zu verändern; mit `setserial` (<http://www.picante.com/~gtaylor/pht/man/setserial.8.html>) 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* (<http://sunsite.unc.edu/LDP/HOWTO/Serial-HOWTO.html>) nachgelesen werden.

Wenn man einen langsamen seriellen Drucker mit Flußkontrolle benutzt, kann es sein, daß einige Druckaufträge abgeschnitten werden. Das kann an den seriellen Schnittstellen liegen, die standardmäßig 30 Sekunden, nachdem das Schnittstellendevice geschlossen wurde, alle nicht gesendeten Zeichen löschen. Der Puffer hält bis zu 4096 Zeichen, wenn der Drucker Flußkontrolle verwendet und nicht schnell genug ist, um innerhalb von 30 Sekunden, nachdem die Software die Schnittstelle geschlossen hat, alle Daten aus dem Puffer zu empfangen, ist das Ende des Pufferinhalts verloren. Wenn der Befehl `cat file > /dev/ttyS2` 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` im `rc.serial` Bootfile 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 einen seriellen oder parallelen Port 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 (Oft werden sie schwammig mit "für Windows" bezeichnet, oder als "GDI-Drucker"). Diese Drucker *funktionieren nicht* mit Linux. Sie sind *btrohdoof* und auf den Hauptprozessor angewiesen, um den größten Teil der Arbeit zu machen, die normalerweise von der CPU des Druckers erledigt wird. Unglücklicherweise kann diese Aufgabe nur von den Treibern des Herstellers erledigt werden, welche nur für Windows verfügbar sind. Also: kaufen Sie nicht einen solchen Drucker, wenn Sie ihn mit Linux verwenden wollen. (Wenn Sie bereits einen solchen Drucker besitzen: es gibt Möglichkeiten, um Linux zur Zusammenarbeit zu bewegen, aber diese sind unzureichend und ich habe es nie selbst versucht. Beachten Sie Kapitel 12 dieses Dokuments für eine weitergehende Besprechung von Windows-only-Druckern.)

Bezüglich der Drucker, die mit Linux *funktionieren*, so ist die beste Wahl ein Drucker mit eingebautem PostScript-Support. Fast jede Unix-Software, die Druckfähigkeiten besitzt, produziert PostScript-Daten, also wäre es offensichtlich am besten, einen Drucker zu nehmen, der PostScript direkt unterstützt. Unglücklicherweise ist PostScript-Support selten bei nicht-Laserdruckern, und außerdem teilweise kostenpflichtiges Zubehör.

Wenn Sie die (Mehr-)Ausgabe scheuen, die nötig ist, um einen PostScript-Drucker zu kaufen, so können Sie jeden Drucker verwenden, der von Ghostscript unterstützt wird, einem freien PostScript-Interpreter, der bei fehlender echter PostScript-Unterstützung Verwendung findet. Auf der *Ghostscript Homepage* (<http://www.cs.wisc.edu/~ghost/>) finden Sie eine Liste der unterstützten Drucker und Informationen über den Status von neuen und experimentellen Treibern. Bitte helfen Sie dabei, die Ghostscript-Druckerseite zu verbessern, indem Sie der Bitte nachkommen, Ihre (Miß-)Erfolge mitzuteilen.

Adobe hat eine neue Druckersprache namens PrintGear entwickelt. Ich denke, es handelt sich dabei um eine stark vereinfachte Sprache im Binärformat, die einiges von PostScript geerbt hat, aber nicht dazu kompatibel ist. Ferner habe ich nichts davon gehört, daß es von Ghostscript unterstützt wird. Einige PrintGear-Drucker scheinen auch eine

zweite Sprache wie z.B. PCL zu unterstützen, und solche Drucker funktionieren mit Linux.

4.1 Drucker-spezifische Bemerkungen

Dieser Abschnitt ist unvollständig und wird es immer sein. Aber die Informationen, die ich hier schreibe, sollten korrekt sein.

Canon BJ-10ex

Ein Bericht schlägt vor, daß der Canon sehr viel besser im Epson LQ Modus arbeitet (per DIP-Schalter zu setzen). Anscheinend war die Ausgabe im Canon-Modus unterbrochen oder falsch. Andererseits gibt es mittlerweile besseren Canon-Support in Ghostscript, so daß sich dieses Problem erledigt haben dürfte.

Lexmark

Lexmark (früher Teil von IBM, glaube ich) stellt viele verschiedene Drucker her; viele sind allerdings Windows-only 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 Linuxanwender einfach - jeder verwendete dasselbe alte lpd, das fast wörtlich aus der Net-2-Distribution von BSD genommen wurde. Auch heute liefern die meisten Distributoren noch diese Software. 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 das ursprüngliche lpd. Es can sogar wirklich von sich behaupten, sicher zu sein - es gibt keine SUID-Teile, und es unterstützt Authentisierung via PGP oder Kerberos.

Im Moment ist lpd wohl für die meisten Linuxanwender die beste Lösung, auch mit den neuen Möglichkeiten verglichen. Obwohl es nicht das schönste System ist, funktioniert es gut, sobald es einmal eingerichtet ist, es kennen sich viele Leute damit aus und es ist in vielen Unix-Büchern gut dokumentiert.

Weitere Informationen über LPRng sind auf der LPRng-Seite (<http://www.astart.com/lprng/LPRng.html>) zu finden. Zukünftige Versionen dieser HOWTO werden sowohl über lpd als auch über LPRng Informationen enthalten.

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 oder auf die gesamten Programme, die für das Druck-Spooling zuständig sind. Das sind:

lpd

Der Spooling-Daemon. Einer kontrolliert alles auf einer Maschine, UND einer pro Drucker läuft, solange der Drucker druckt.

lpr

Der Spooling-Befehl für den Anwender. Lpr ruft lpd auf und setzt einen neuen Druckauftrag (Job) in den Spool.

lpq

Listet die Jobs 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 lpd gestartet. Es durchsucht die Datei `/etc/printcap`, um festzustellen, für welche Drucker es Spools verwaltet. Jedes Mal, wenn man lpr aufruft, nimmt lpr mit lpd durch den named socket `/dev/printer` Verbindung auf und übergibt 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 - das heißt, die meisten Anwender druckten reines ASCII. Wie sich herausgestellt hat, ist nur wenig mehr Aufwand nötig, damit lpd mit heutigen Druckaufträgen klarkommt, die häufig in PostScript sind, oder Text oder dvi oder...

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 nicht darum, ob der Drucker die Dateien überhaupt versteht und wird vermutlich keine ansehnlichen Ausgaben produzieren. Trotzdem ist es der erste Schritt zum Verständnis, also weiterlesen!

Um eine Warteschlange zu lpd hinzuzufügen, 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 djet500
lp|dj|deskjet:\
    :sd=/var/spool/lpd/dj:\
    :mx#0:\
    :lp=/dev/lp0:\
    :sh:
```


Dies definiert einen Spool namens `lp`, `dj` oder `deskjet` mit dem Spool-Verzeichnis `/var/spool/lpd/dj`, ohne maximale Auftragsgröße, der auf das Device `/dev/lp0` ausgibt und am Anfang des Druckauftrags kein Deckblatt hinzufügt (mit dem Namen der Person, die gedruckt hat, etc.).

Jetzt wäre der richtige Augenblick, um die `printcap` Man-page zu lesen.

Das obige Beispiel sieht sehr einfach aus, hat aber ein Problem. Wenn man keine Files sendet, die ein DeskJet 500 verstehen kann, wird dieser Deskjet seltsame Sachen drucken. Wenn man z.B. eine gewöhnlichen UNIX-Textdatei an den Deskjet schickt, interpretiert er die Zeilenwechsel "wörtlich" und gibt aus:

```
Dies ist Zeile eins.
        Dies ist Zeile zwei.
                Dies ist Zeile drei.
```

und so weiter. Der Ausdruck einer PostScript-Datei würde eine schöne Liste der PostScript-Befehle geben, mit diesem Treppenstufeneffekt ausgedruckt, aber keine sinnvolle Ausgabe.

Offensichtlich wird mehr benötigt, und genau das ist die Aufgabe des Filterns. Dem aufmerksamen Leser werden bei der `printcap` man-page die Spool-Attribute `if` und `of` aufgefallen sein. `if`, der Inputfilter, ist genau das, was wir jetzt brauchen.

Wenn man ein kleines Shellscript namens `filter` schreibt, daß Zeilenumbrüche (CR) vor die Newlines setzt, wird der Treppenstufeneffekt verhindert. Also muß in den `printcap`-Eintrag eine `if`-Zeile eingefügt werden:

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

Ein einfaches Filterscript könnte sein:

```
#!/perl
# The above line should really have the whole path to perl
# This script must be executable: chmod 755 filter
while(<STDIN>){chop $_; print "$_\r\n";}
# You might also want to end with a form feed: print "\f";
```

Wenn man das machen würde, hätte man einen Spool, in den man gewöhnliche UNIX-Textdateien drucken kann und sinnvolle Ergebnisse bekäme. (Ja, es gibt vier Millionen bessere Möglichkeiten, diesen Filter zu schreiben, aber wenige so anschauliche. Der Leser möge dies effizienter gestalten.)

Das einzige verbleibende Problem besteht darin, daß gewöhnlichen Text zu drucken nichts Besonderes ist - es wäre sicherlich besser, wenn man PostScript und andere formatierte Texte oder Grafik-Ausgaben drucken könnte. Ja, das wäre es, und es ist einfach zu machen. 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 einige wirklich gute im Netz. Der APS-Filter ist einer der besten, oder Ihre Linux-Distribution hat vielleicht ein Drucker-Setup Tool, das all dies deutlich vereinfacht.

7.2 File Permissions

Auf Grund von häufigen Nachfragen folgt hier eine Liste der Permissions der wichtigen Dateien, wie sie auf meinem System gesetzt sind. Es gibt bessere Möglichkeiten, im Idealfall ohne SUID root, sondern mit SGID Binärdateien, aber so wurde das System installiert und es funktioniert. (Ehrlich gesagt, wenn ein Verkäufer noch nicht einmal ein funktionierendes lpd liefern kann...)

```
-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, damit es die lp Service-Ports mit niedrigen Nummern ansprechen kann. Es sollte vielleicht UID lp.lp oder so etwas nach der Anbindung werden, aber das tut es wohl nicht. Dumm, das.

8 Woher man Drucker-Software bekommt

Viele fertige Filterpakete (und andere Druckersoftware) kann man unter

```
ftp://sunsite.unc.edu/pub/Linux/system/printing/
```

finden. Utilities wie psutils, a2ps, mpage, dvtodvi, flpr usw. finden sich alle hier.

Eine Zeitlang gab es mehrere Pakete da draußen, 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 Sie einen von Ghostscript unterstützten Drucker auswählen und das Unix Device, auf das gedruckt werden soll, und installiert sowohl eine Drucker-Queue in /etc/printcap als auch ein kurzes PostScript-und-ASCII Filterscript, das auf gs und auf nenscript (<http://www.picante.com/~gtaylor/pht/man/nenscript.1.html>) 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 (<http://www.cs.wisc.edu/~ghost/>) ist ein unglaublich wichtiges Programm, um unter Linux zu drucken. Die meiste Software unter Unix generiert PostScript, was häufig ein DM200-Zubehör (\$100, d. Übers.) für einen Drucker bedeutet. Ghostscript allerdings ist frei, und generiert aus den PostScriptdaten Daten in der Sprache Ihres Druckers. Wenn er in Verbindung mit Ihrem `lpd` Eingabefilter eingesetzt wird, sorgt er für einen virtuellen PostScript-Drucker und erleichtert das Leben enorm.

Ghostscript gibt es in zwei Ausführungen. Die kommerzielle Version von Ghostscript, „Aladdin Ghostscript“, darf zwar frei verwendet, aber nicht auf kommerziellen Linux-Distributionen verbreitet werden. Es ist dem freien Ghostscript normalerweise 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“, und ist einfach eine ältere Version des 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` (<http://www.picante.com/~gtaylor/pht/man/gs.1.html>) anfangen, vergewissern Sie sich unbedingt, es mit der Option für ausgeschaltete Dateizugriffe (`-dSAFER`) zu starten. PostScript ist eine voll funktionale Sprache, und ein schlechtes PostScript-Programm könnte Ihnen ziemliche Kopfschmerzen bereiten.

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, es direkt aufzurufen.

`gs -help` gibt eine kurze, informative Übersicht über die Optionen und verfügbaren Treiber aus (beachten Sie bitte, daß dies die Liste der einkompilierten Treiber ist, nicht die vollständige Liste aller verfügbaren Treiber).

Sie können `gs` für Testzwecke folgendermaßen aufrufen:

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

10.2 Feineinstellung der Ghostscript-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 der Ausgabe

Die Position, 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 ich Ihnen nicht sagen kann, was Sie einstellen müssen, aber die meisten sind ausreichend gut kommentiert.

10.2.2 Gamma, Punktgröße usw.

Die meisten nicht-Laserdrucker leiden unter der Tatsache, daß ihre Punkte recht groß sind. Dies führt dazu, daß Bilder zu dunkel ausgegeben werden. Wenn Sie dieses Problem haben, sollten Sie Ihre eigene Übertragungsfunktion verwenden. Erstellen Sie einfach die folgende Datei im Ghostscript lib-Verzeichnis und fügen Sie ihren Namen dem gs-Aufruf hinzu, genau vor der eigentlichen Druckdatei. Vielleicht müssen Sie die tatsächlichen Werte Ihrem eigenen Drucker anpassen. Niedrigere Werte führen zu einem helleren Ausdruck. Besonders, wenn Ihr 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, empfehle ich Ihnen die Benutzung der Datei `colorcir.ps`, die Ghostscript als Testseite beiliegt (im `examples/`-Unterverzeichnis).

11 Wie man auf einen Drucker über ein Netzwerk druckt

Eine der Fähigkeiten von lpd liegt darin, 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 Auf einen UNIX/lpd Host

Um es anderen Rechnern zu ermöglichen, auf dem eigenen Drucker zu drucken, müssen diese in `/etc/hosts.equiv` oder `/etc/hosts.lpd` aufgelistet werden. Achtung: `hosts.equiv` hat viele weitere Effekte; 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 Man-page für mehr Informationen.

11.1.1 Mit lpd

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

```
# NETZWERK djet500
lp|dj|deskjet:\
    :sd=/var/spool/lpd/dj:\
    :rm=geraet.da.draussen.com:\
    :rp=printername:\
    :lp=/dev/null:\
    :sh:
```

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 Mit rlpd

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

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

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

`rlpd` kann überall da verwendet werden, wo ein traditionelles `lpr` verwendet werden kann und ist abwärts-kompatibel zum traditionellen BSD `lpr`.

Der Hauptvorteil von `rlpd` ist die Möglichkeit, *von überall nach überall* zu drucken, ohne Rücksicht darauf, wie das System, von dem aus man drucken will, konfiguriert ist. Es 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.

`rlpd` ist erhältlich von:

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

11.2 Auf einen Win95, WinNT, LanManager oder Samba-Drucker

Es ist möglich, eine `lpd`-Warteschlange mit dem `smbclient`-Programm (ein Teil des Samba-Pakets) zu einem TCP/IP-basierenden SMB-Druckservice zu schicken. 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. Hierzu siehe die Man-page.

11.3 Auf einen Netware-Drucker

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

```
sunsite.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. Man benötigt Kernel 1.2.x oder 1.3.54 oder höher. `ncpfs` funktioniert NICHT mit 1.3.x-Kernels unter 1.3.54.)

Damit nprint über lpd arbeitet, schreibt man ein kleines Shellsript, um stdin auf dem Netware-Drucker zu drucken und installiert es als if für eine lpd Warteschlange. Man erhält so etwas wie:

```
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
# You should try the guest account with no password first!
/usr/local/bin/nprint -S net -U name -P passwd -q printq-name -
```

11.4 Auf einen EtherTalk- (Apple-) Drucker

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 Auf einen HP oder anderem Ethernet-Drucker

HPs und einige andere Drucker besitzen ein Ethernet-Interface, auf dem man direkt mit lpd drucken kann. Man sollte den Anweisungen folgen, die mit dem Drucker oder Netzwerkadapter geliefert wurden, aber generell lassen solche Drucker lpd laufen und stellen eine oder mehrere Warteschlangen zur Verfügung. Ein HP zum Beispiel könnte 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", die PCL (und eventuell PostScript) akzeptiert, und text", die reines ASCII erwartet (und automatisch den Treppeneffekt korrigiert). Wenn Sie eine JetDirect Plus3 drei-Port Box besitzen, besitzen die Queues die Bezeichnungen raw1", text2 usw.

In einem großen Netz, besonders 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 einen Telnet auf den Drucker ausführen, zweimal Return drücken und danach "banner: 0" gefolgt von "quit". Sie können auch andere Einstellungen auf diese Art verändern; tippen Sie "?" ein, um eine Liste angezeigt zu bekommen.

11.5.1 Auf alten HPs

Einige Drucker (und Drucker-Netzwerk "Black Boxes") unterstützen nur ein kleines Nicht-Protokoll mit reinen TCP-Verbindungen. 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 drei-Port Boxen) öffnen und den Druckauftrag hierdurch schicken. Das kann u.a. in Perl implementiert werden:

```
#!/usr/bin/perl
# Thanks to Dan McLaughlin for writing the original version of this
# script (And to Jim W. Jones for sitting next to Dan when writing me
# for help ;)

$fileName = @ARGV[0];

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

$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 fuer 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 Ein *if* für Netzwerkdrucker

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

```

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:

```

mit diesem passenden *filter-lj-5-Script*:

```

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

```


Die -U-Option bei 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 Usernamen zu ermitteln, denn manchmal ist es nicht Argument 5. Siehe auch die Man-page zu `printcap`.

11.7 Von Windows aus.

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 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 Von einem Apple aus.

Netatalk unterstützt das Drucken von Apple-Clients über EtherTalk. Siehe hierfür auch die Netatalk HOWTO Page (<http://thehamptons.com/anders/netatalk/>).

11.9 Von Netware aus.

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 "Binderybasiertes 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 werfen (<ftp://ftp.gwdg.de/pub/linux/misc/ncpfs/>).

12 Windows-only printers

Wie ich schon vorher erklärt habe, 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 geleitet wird. Manchmal verstehen solche Drucker auch etwas anderes wie normales PCL, aber meistens nicht. In einigen (wirklich billigen) Fällen besitzen die Drucker nicht einmal eine normale Parallelverbindung, sondern benötigen den Herstellertreiber, um zu emulieren, was eigentlich die Hardware machen sollte (hauptsächlich Handshaking).

In jedem Fall gibt es einige mögliche Auswege, wenn Sie eine solche SZitrone erwischen haben.

12.1 Der Ghostscript Windows Druckertreiber

Es gibt mittlerweile einen Windows Druckertreiber (namens mswinpr2), der einen Druckauftrag durch Ghostscript leitet, bevor er ihn druckt. (Ähnlich einem if-Filter im Unix lpd.) Es gibt auch einen neuen Ghostscripttreiber, der über Windows- (GDI-) Aufrufe druckt. Zusammengefasst sollte(tm) dies einer Windowsmaschine erlauben, PostScript auf einen Windows-only Drucker über den Herstellertreiber auszugeben.

Wenn Sie dies zum Laufen bringen, können Sie den Anweisungen oben folgen, um von Linux aus über das Netzwerk auf einen Windowsdrucker zu drucken, um es so Unixrechnern (oder einem anderen Windows, Mac usw.) zu ermöglichen, auf Ihrer SZitrone zu arbeiten.

Es sei klargestellt, daß Ich dies nie ausprobiert habe, weil ich keinen Windowsdrucker besitze, daher mag Ihr Erfolg unterschiedlich ausfallen. Sämtliche nötige Software sollte auf der Ghostscript-Homepage (siehe Links im Ghostscript-Abschnitt) verfügbar sein.

12.2 Das pbm2ppa Programm

Einige HP Drucker benutzen die Printing Performance Architecture (marktgerecht für "wir waren zu billig, um PCL zu benutzen"). 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 den Drucker ausgegeben werden kann. Mittlerweile könnte es dieses Programm auch als Ghostscript Druckertreiber geben.

Die ppa-Software kann von der ppa Homepage (<http://www.rpi.edu/~normat/technical/ppa/>) heruntergeladen werden; 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 Wie man auf einem Faxgerät druckt

13.1 Mit einem Faxmodem

Es gibt einige Faxprogramme, mit denen man faxen und Dokumente empfangen kann. Eines der komplexesten ist Sam Lefflers *HylaFax*, erhältlich von <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 nicht-Java Fax-Clients für die meisten Plattformen; Linux kann mit ziemlicher Sicherheit Ihren Netzwerk Fax Anforderungen genügen.

Ebenfalls verfügbar, und die bessere Wahl für die meisten Linuxboxen: efax (<http://www.picante.com/~gtaylor/pht/man/efax.1.html>). Das ist ein einfaches Programm, das Faxe sendet. Das getty-Programm mgetty kann Faxe empfangen (und sogar Sprache mit einigen Modems!).

13.2 Mit dem Remote Printing Service

Es gibt einen experimentellen Dienst, an den man eine eMail senden kann, die dann irgendwo anders auf einem Faxgerät erscheint. Es werden Formate wie PostScript unterstützt, so daß es ein sehr sinnvoller Dienst ist, obwohl die globale Unterstützung lückenhaft ist. Für weitere Informationen siehe die Remote Printing WWW Site (<http://www.tpc.int/>).

14 Wie man etwas Druckbares herstellt.

Hier kommt man in ein wahres Rattennest an Software. Grundsätzlich kann man viele verschiedene Typen von Binärdateien mit unterschiedlichem Erfolg laufen lassen: Linux/x86, Linux/Alpha, Linux/Sparc, Linux/foo, iBCS, Win16/Win32s (mit dosemu und irgendwann mit Wine) und Mac/68k (mit Executor). Ich werde nur reine Linux und UNIX-Programme behandeln.

Für Linux selbst ist man hauptsächlich auf die für UNIX generell erhältlichen Programme beschränkt:

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. Ein schönes Zeichen in einer Formatierungssprache könnte schmerzhaft sein...

nroff

Diese war eine der ersten Formatierungssprachen für UNIX. Man-pages sind das häufigste Beispiel für Texte, die mit `*roff`-Makros formatiert wurden; einige Leute schwören darauf, aber `nroff` hat für mich eine kompliziertere Syntax als nötig und ist deshalb wohl eine schlechte Wahl für neue Texte. Man sollte aber trotzdem wissen, daß man eine Man-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. Technische Werke werden häufig in LaTeX geschrieben, denn es vereinfacht die Layoutfragen stark und ist *immer noch* eines der wenigen Textverarbeitungssysteme, daß mathematische Formeln komplett und gut unterstützt. Das Ausgabeformat von TeX ist `dvi` und kann mit `dvips` oder `dvilj` nach PostScript bzw. Hewlett Packards PCL konvertiert werden.

SGML

Es gibt mindestens einen freien `sgml`-Parser für Unix und Linux; es bildet die Basis des selbst entwickelten Dokumentensystems von `Linuxdoc-SGML`. Es kann auch andere DTDs unterstützen.

HTML

Jemand hat vorgeschlagen, einfache Projekte in HTML zu schreiben und mit Netscape auszudrucken. Ich bin anderer Meinung, aber Ihre Erfahrungen mögen anders sein.

14.2 WYSIWYG

Mittlerweile gibt es keinen Mangel an WYSIWYG Textverarbeitungssoftware mehr. Es gibt mehrere vollständige Office-Pakete, und sogar eins, das für private Nutzung frei verfügbar ist (`StarOffice`).

StarOffice

Eine deutsche Fimat vertreibt `StarOffice` für Linux frei über das Netz. Dieses vollwertige Office-Paket verfügt über alle Fähigkeiten, die man erwartet, und der Preis ist unschlagbar. Es gibt ein Mini-HOWTO, das beschreibt, wie man es bekommt und installiert. Es produziert PostScript oder PCL, so daß es mit fast jedem Drucker arbeiten sollte, der auch sonst unter Linux funktioniert.

LyX

LyriX ist ein Frontend für LaTeX, das vielversprechend aussieht. Auf der LyX Homepage (<http://www-pu.informatik.uni-tuebingen.de/users/ettrich/>) gibt es weitere Informationen. 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. Ich glaube aber nicht, daß es bereits benutzbar ist.

GNOME

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

Kommerzielle Angebote

Zumindest Caldera und Red Hat beeinhaltet Pakete mit den üblichen Office-Anwendungen wie einer WYSIWYG-Textverarbeitung und einer Tabellenkalkulation. Ich nehme an, daß sie gut funktionieren, habe sie aber noch nie benutzt. Ich glaube, daß Caldera auch Sun's WABI vertreibt, so daß man vielleicht etwas wie MS Office laufen lassen kann, wenn man mit den Dateien anderer Leute arbeiten muß.

Jeff Phillips (jeff@I_RATUS.org) verwendet Caldera's WordPerfect für Linux (auf Slackware) und sagt, daß es gut funktioniert. Es besitzt anscheinend eingebaute Druckerunterstützung, wie zu erwarten war. Caldera sollte mehr Informationen unter <http://www.caldera.com/> haben. Sie können ferner eine neuere Version von WordPerfect direkt von der Firma beziehen, die Corel's Unix-Ports herstellt.

RedHat und SuSE vertreiben ein Paket namens *Applixware*; ihre Website ist unter <http://www.redhat.com> zu erreichen. 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; ich würde vermuten, daß es ziemlich dasselbe ist, aber langsamer und mit weniger Fähigkeiten.

Sonstige Firmen können mich gerne über ihre Produkte unterrichten.

15 Voransicht von druckbaren Sachen auf dem Bildschirm

Fast alles, das man drucken kann, kann auch dem Bildschirm angezeigt werden.

15.1 PostScript

Ghostscript besitzt einen X11-Treiber, der am besten unter der Verwaltung des PostScript-Betrachters gv verwendet wird. Die neuesten Versionen dieser Programme sollten auch PDF-Dateien betrachten können. Beachten Sie, daß gv das ältere Preview-Programm "Ghostview" ersetzt hat; die neue Benutzerschnittstelle ist wesentlich ansehnlicher und funktionaler als ghostview's altes reines Athena-Interface.

15.2 TeX dvi

TeX DeVice-Independent-files (geräteunabhängige Dateien) können unter X11 mit xdvi betrachtet werden. Moderne Versionen von xdvi rufen Ghostscript auf, um PostScript- β specials zu verarbeiten.

Es gibt ebenfalls einen VT100-Treiber. Er wird dgv_t genannt. Tmview arbeitet mit Linux und sygalib, falls das alles ist, was verfügbar ist.

15.3 Adobe PDF

Adobe's Acrobat Reader gibt es auch für Linux; Sie können es einfach von ihrer Web-Site herunterladen (<http://www.adobe.com/>).

Sie können auch xpdf benutzen, daß Freeware ist und mit Quellen verfügbar ist, und ich glaube, daß Ghostview mit gs mittlerweile das Betrachten von PDF-Dateien unter X11 unterstützt.

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 des seriellen Ports zu kontrollieren, an den der Drucker angeschlossen ist. Man lese dazu die `printcap` Man-page und beachte die Bedeutungen von `br#`, `fc#`, `xc#`, `fs#` und `xs#`. Die letzten vier dieser Attribute sind Bitmaps, die für die Einstellungen dieses Ports 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 Man-page von `stty`.

Benutzen Sie `stty`, um den Druckerport so einzustellen, so daß man ein File an ihn mit `cat` schicken kann, und es korrekt ausgegeben wird. So sieht `'stty -a'` für meinen Druckerport aus:

```
dina:/usr/users/andy/work/lpd/lpd# 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`. Ihr Port kann durchaus anders sein, abhängig von der Flußkontrolle Ihres Druckers.

Wirklich verwendet wird `stty` auf eine etwas ungewöhnliche Weise. Weil `stty` auf dem Terminal arbeitet, das mit seinem Standardinput (ßtdin", d. Übers) verbunden ist, kann man damit einen seriellen Port manipulieren, indem man das `'<'`-Zeichen wie oben verwendet.

Sobald die `stty`-Einstellungen richtig sind, wenn also `'cat file > /dev/ttyS2'` (in meinem Fall) die Datei an den Drucker schickt, sollte man die Datei `/usr/src/linux/include/asm-i386/termbits.h` betrachten. Diese Datei enthält viele `#defines` und einige structs (man sollte diese Datei auf den Drucker ausgeben (das funktioniert ja jetzt, oder?) und als Vorlage verwenden). Man betrachte den Abschnitt, der mit

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

anfängt. Dieses 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 (ich mache es so).

Jetzt macht man das gleiche für die Einstellungen, die kein `-` vor sich haben. In dem Beispiel sind die wichtigen CS8 (0000060), HUPCL (0002000), und CREAD (0000200). Beachten Sie auch die Flags für die Baudrate (meine ist 0000015). Zählen Sie alle zusammen und Sie bekommen in diesem Beispiel 0002275 heraus. Das kommt in die `fs#`-Einstellung (`'fs#02275'` funktioniert in meinem Beispiel gut).

Machen Sie das gleiche mit `betünd clear` für den nächsten Abschnitt der Includedatei, `c_lflag bits`. Ich für meinen Teil mußte nichts einstellen und verwende deshalb `'xc#0157777'` und `'xs#0'`.

16.2 Ältere serielle Drucker, die Zeichen verschlucken

Jon Luckey hat festgestellt, daß einige ältere serielle Drucker mit 10-cent seriellen Interfaces und kleinen Buffern *wirklich* Stopp meinen, wenn sie das mit ihrer Flußkontrolle sagen. Er fand heraus, daß man das aufhörte, wenn er für den "16550Serialbaustein mit `setserial` den FIFO ausschaltete (man gibt offensichtlich nur den UART-Typ als "8250" an).

17 Credits

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

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

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

Der Teil über `gammas` und `sowetter` für `gs` wurde eingeschickt 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.