

# Kapitel 1

## Linux und freie Software

### Inhalt

- Vorstellungsrunde Linux, GNU, Debian und Ubuntu
- Was freie Software ausmacht
- Sicherheit und Freiheit kann gemeinsam gehen
- Ideologien und was sie nutzen

Linux hat sich im Bereich der Server weitgehend durchgesetzt. Das erscheint auf den ersten Blick ungewöhnlich, weil Linux nicht durch eine kommerzielle Firma vermarktet wird.

### 1.1 Linux ist nicht Windows

Falls Sie auf Ihrem Arbeitsplatz-Computer Windows verwenden, werden Sie vermutlich automatisch Vergleiche zwischen Windows und Linux ziehen und vielleicht auch hin und wieder über die Unterschiede fluchen: Warum verhält sich Linux anders als Windows? Können die Programmierer das nicht besser?

Das könnten sie sicher. Aber das wollen sie nicht. Die Programmierer, die die Entwicklung von Linux anstießen und fortsetzten, taten dies zu einem großen Teil in ihrer Freizeit und wollten sehr bewusst ein anderes, ein besseres Betriebssystem programmieren als das, was sie auch für wenig Geld im Laden kaufen könnten.

Das Vorbild für Linux war UNIX, das durch klare Konzepte glänzte. Aber die Entwicklung blieb nicht stehen und so wurden die Konzepte immer wieder hinterfragt und durch modernere, noch klarere und performantere Ansätze ersetzt. Um solche Änderungen streiten sich die Entwickler oft sehr engagiert. Das wirkt auf Außenstehende manchmal etwas irritierend, führt aber langfristig zu einem leistungsfähigen Betriebssystem, das auf der Höhe der Zeit bleibt.

Dagegen schleppt Windows immer noch die Altlasten längst vergangener Tage mit sich herum. Ein typisches Beispiel sind die Laufwerksbuchstaben. Warum wird die erste Festplatte immer mit dem Buchstaben C: bezeichnet? Weil A: und B: immer noch für zwei Diskettenlaufwerke reserviert sind, obwohl man heutzutage solche Laufwerke nicht einmal mehr kaufen kann. Und die Bindung an diese Laufwerksbuchstaben führt noch heute dazu, dass der Anwender ein massives Problem hat, wenn sein Laufwerk C: voll ist. Linux verzichtet dagegen ganz auf Laufwerkskennungen. Alle Dateien befinden sich in genau einem Verzeichnisbaum, der sich über mehrere Laufwerke verteilt. Und das Umlagern von Programmen oder Dateien auf ein anderes Laufwerk ist für einen Linux-Administrator eine Kleinigkeit.

Darum sollten Sie sich nicht ärgern. Linux ist einfach anders. Es will anders sein und es ist gut, dass es anders ist. Lassen Sie sich lieber auf diese Andersartigkeit ein. Sie werden sehen, dass es meist sehr logisch strukturiert ist und auf lange Sicht Ihre Probleme einfacher löst, als dies mit Windows möglich wäre.

### 1.2 Linux? Debian? Ubuntu?

Sie wollen einen Linux-Server betreiben, sonst hätten Sie dieses Buch nicht in der Hand. Aber was ist eigentlich Linux genau? Sie finden nirgends einen Laden, in dem Sie neben den Schachteln von Windows eine Schachtel mit dem Aufdruck »Linux« finden.

Denn obwohl Linux als Bezeichnung für das Betriebssystem verwendet wird, ist es streng genommen nur der Name des Betriebssystemkerns, um den das Betriebssystem gestrickt ist. Zu einem Linux-System bekommen Sie neben dem Kern aber noch reichlich Software dazu. Und da Linux Open Source, also freie Software ist und freie Software auch noch kostenlos ist, ist das Paket großzügig bemessen. Bis hin zur Textverarbeitung, Datenbank und diversen Serverprogrammen ist alles inklusive!

Dieses Gesamtpaket aus Systemkern und Programmen wird als *Distribution* bezeichnet. Eine Distribution umfasst das Installationsmedium und die Software-Pakete. Sie sorgt dafür, dass das System installiert werden kann, aktualisiert wird und alle Bestandteile zusammenpassen. In diesem Buch werden die Distributionen Debian und Ubuntu Server verwendet, um einen Linux-Server zu betreiben.

### Warum Debian und Ubuntu?

Debian ist eine der ältesten Distributionen und im Gegensatz zu manch anderer Distribution nicht kommerziell. Sie gilt als außerordentlich stabil und eignet sich darum ideal als Basis für einen Server.

Das Ubuntu-Projekt ist angetreten, um vor allem im Desktop-Bereich die Linux-Welt einem breiten Publikum zugänglich zu machen. Hier hat Ubuntu auch viel erreicht. Heute ist es für jemanden, der beide Systeme kennt, erheblich leichter, auf einem Laptop Ubuntu zu installieren, als eine neue Windows-Version.

Ubuntu ist aber nicht etwa eine vollkommen eigenständige Distribution, sondern setzt auf dem Paketsystem und teils auch auf der Infrastruktur von Debian auf. Darum ist das Verhalten von Ubuntu und Debian an vielen Stellen sehr ähnlich, sodass beide Versionen zusammen betrachtet werden können. Auf gelegentliche Unterschiede werde ich Sie natürlich hinweisen.

Falls Sie nun eine andere Distribution verwenden, aber zufällig dieses Buch geschenkt bekommen haben oder nur noch dieses in der Bibliothek vorrätig war, werden Sie vielleicht überrascht feststellen, dass immer noch erstaunlich viele Beschreibungen auch auf Ihre Distribution passen.

### Was ist mit GNU, freier und offener Software?

Die Bezeichnung für Debian lautet vollständig »Debian GNU/Linux«. Das zeigt an, dass Debian und damit auch Ubuntu etwas mit GNU und mit Linux zu tun hat. Und sowohl GNU als auch Linux greifen auf die Ideen von UNIX zurück. Was GNU genau ist und wie es dazu kam, werden die folgenden Abschnitte ausführlicher erläutern.

Das wichtige Thema von GNU ist die Freiheit der Software. Dieses Thema Freiheit hat etwas mit der Geschichte von UNIX zu tun, das plötzlich nicht mehr frei war. Tatsächlich ist der Freiheitsgedanke das treibende Moment im Debian-Projekt. Hieraus erwachsen eine demokratische Organisationsstruktur, der Gesellschaftsvertrag und die oft leidenschaftliche geführte Debatte, welche Software aufgenommen werden soll und welche nicht.

## 1.3 Software soll frei sein

Wie kaum eine andere Distribution legt Debian besonderen Wert auf die Freiheit von Software. So meldet Debian am 15.12.2010 stolz, dass die Version Squeeze mit einem

Kernel ausgeliefert wird, der vollständig von unfreier Firmware befreit ist.<sup>1</sup> Debian steht damit in der Tradition von GNU, dessen Namen untrennbar mit der Distribution verbunden ist. Lassen Sie sich ein wenig in die Hintergründe dieser Thematik entführen, die in manchen Fällen eben auch Auswirkungen auf die Praxis hat.

Die Abkürzung GNU steht für »GNU is not UNIX«. Diese rekursive Auflösung zieht die Aufmerksamkeit immer sehr auf das »G«, das ja nicht aufgelöst wird. Dabei ist die Bedeutung von »not UNIX« viel interessanter. Warum distanziert sich die GNU-Bewegung so von UNIX, obwohl die von ihr hergestellte Software UNIX fast vollständig imitiert?

Tatsächlich implementiert die GNU-Software funktional die UNIX-Werkzeuge und übertrifft sie an vielen Stellen erheblich. Die Gegensätzlichkeit liegt nicht in der Funktion, sondern in den Lizenzen der Quelltexte. Keine Software, die GNU erstellt, hat mit den lizenzpflichtigen Quelltexten von UNIX etwas zu tun. Es ist alles neu geschrieben worden, ohne eine Zeile von UNIX zu übernehmen. Genau dies meint die Aussage »GNU is not UNIX«. Der Hintergrund findet sich in der Geschichte von UNIX.

### 1.3.1 Wie UNIX unfrei wurde

Alles fing mit dem Scheitern der Entwicklung von MULTICS an. Es sollte das Superbetriebssystem werden und MIT, General Electric sowie die Bell Labs von AT&T wollten es entwickeln. Es stellte sich aber heraus, dass die Erwartungen nicht mit den technischen Möglichkeiten seiner Zeit in Einklang zu bringen waren. Es war schlicht nicht implementierbar. Und so stiegen die Bell Labs aus. Einige der Entwickler für MULTICS, unter ihnen vor allem Dennis Ritchie und Ken Thompson, entwickelten aus diesen Ideen einen kleinen Abguss, den sie UNICS, später UNIX nannten. Damit das System leicht portierbar wurde, entwickelten sie extra dafür die Programmiersprache C. Da die Firma AT&T davon ausging, dass man damit kein Geld verdienen könnte, stellten sie einigen Universitäten neben UNIX auch dessen Quelltext für Forschung und Ausbildung zur Verfügung.

Dort wurde UNIX ständig erweitert und entwickelte sich so zu einem leistungsfähigen System. Das Internet-Protokoll TCP/IP wurde auf der Basis von UNIX entwickelt, und so wurde das System auch kommerziell interessant. Plötzlich legte AT&T den Rückwärtsgang ein und verkaufte UNIX zu erheblichen Lizenzgebühren. Darüber hinaus entzog man den Universitäten den Zugriff auf die Sourcen. Sogar die an den Universitäten entstandenen UNIX-Versionen wurden mit Lizenzgebühren belegt, und infolgedessen mussten die Universitäten unter dem Erfolg von UNIX leiden, den sie teilweise selbst herbeigeführt hatten.

<sup>1</sup> <http://www.debian.de/News/2010/20101215>

### 1.3.2 GPL: Lizenz für die Freiheit

Der Ärger darüber führte zur GNU-Bewegung. Es sollte wieder ein frei verfügbares Betriebssystem entstehen, das so leistungsfähig war wie UNIX. Aber diesmal sollte es frei bleiben. »GNU is not UNIX« sollte deutlich signalisieren, dass einerseits kein Quellcode von UNIX verwendet wurde und dass diese Software andererseits nicht den gleichen Weg in die Unfreiheit gehen sollte wie UNIX.

Damit die freie Software auch frei blieb, wurde eine Lizenz entwickelt, die eine solche Entwicklung wie die bei UNIX verhindern sollte. Sie wurde »Copyleft« genannt, um deutlich zu machen, dass sie genau die Umkehrung des Copyrights bewirken sollte: Freie Software musste immer frei bleiben. Zentrale Bestandteile der GPL (GNU General Public License) sind folgende Forderungen:

- Die Software kann frei genutzt werden, auch kommerziell.
- Kopien dürfen frei verteilt werden, auch gegen Bezahlung. Dem Empfänger müssen allerdings die Bedingungen der GPL genannt werden.
- Der Source-Code muss mit der Software ausgeliefert oder auf Nachfrage zum Selbstkostenpreis nachgeliefert werden.
- Die Sourcen dürfen beliebig verändert werden. Veränderte Programme unterliegen allerdings wiederum der GPL.

Warum die Lizenz so wichtig ist, erschließt sich vielleicht an einem einfachen Beispiel. Nehmen wir an, Sie entwickeln eine Textverarbeitung und wollen diese allen interessierten Anwendern zur Verfügung stellen. Sie stellen die Software inklusive Quelltexte ohne jede Einschränkung zur Verfügung. Nun kommt eine Firma, die schon immer eine Textverarbeitung auf den Markt bringen wollte, übernimmt Ihren Code, entfernt Ihren Namen und setzt das eigene Logo darauf. Die Firma beseitigt ein paar Fehler und fügt noch ein paar kleine Features hinzu. Anschließend verkauft sie diese Software für teures Geld und behält den gesamten Quelltext für sich. Daraufhin geht die Firma in Insolvenz. Die Software wird eingestellt. Sie könnten zwar theoretisch auf dieser Basis weiterentwickeln, aber da Sie keine Rechte an der Software haben, erhalten Sie nicht einmal Einblick in die Erweiterungen. Sie würden dieses Verhalten für unfair halten, könnten sich aber nicht dagegen wehren. Sie haben ja die Rechte an der Software freigegeben!

Genau dieses Szenario soll die GPL verhindern. Eine Software unter GPL muss immer seinen Quelltext zur Verfügung stellen. Jeder darf den Quelltext verbessern und verändern. Er darf aber das veränderte Programm nur dann weitergeben, wenn er auch den Quelltext veröffentlicht und den Empfänger über seine Rechte aus der GPL aufklärt. Durch die Sicherung des Quelltextes kann jederzeit ein anderer Programmierer

den Faden aufgreifen und eine Software weiter verbessern, auch wenn der bisherige Programmierer kein Interesse mehr an dem Projekt hat. Die investierte Arbeit geht nicht verloren.

### 1.3.3 Die Anwender profitieren

Der Freiheitsgedanke der Software scheint den normalen Linux-Benutzer nur am Rand zu tangieren. Er freut sich über die kostenlose Software, macht sich über die Offenheit der Quelltexte allerdings keine Gedanken. Im Gegenteil, es ärgert manchen Benutzer, wenn ein Treiber nicht in der freien Distribution eingesetzt wird, nur weil der einzig verfügbare Treiber vom Hersteller als »unfreie« Software zur Verfügung steht. Ihn interessieren keine Ideologien, sondern die Lauffähigkeit seines Systems.

Den Nachteil nicht freier Software bemerken Sie, falls Ihr teures Notebook vielleicht schon etwas älter ist, aber dennoch mit der neuesten Windows-Version ausgestattet werden soll. Leistungstechnisch wäre dies kein Problem, aber leider gibt es keine Treiber vom Hersteller. Der interessiert sich nämlich nicht für das Altgerät, sondern möchte Ihnen lieber ein neues Notebook verkaufen.

Stünden die Treiber unter GPL, könnten Sie Glück haben, dass ein Programmierer ebenfalls dieses oder ein vergleichbares Notebook besitzt. Hätte er Zugriff auf die Quelltexte, würde er die Treiber anpassen. Mit etwas Glück würde er die neuen Treiber vielleicht zur Verfügung stellen. Aber der Hersteller behält seine Betriebsgeheimnisse lieber für sich.

Falls Sie meinen, das wäre ein Einzelfall, so werden Sie überrascht sein, wie viele Notebooks beim Umstieg auf eine neuere Windows-Version auf der Strecke bleiben. Das Problem betrifft auch nicht nur Notebooks. Bei jedem Windows-Wechsel wird bei eBay eine ganze Menge Peripherie erstaunlich billig angeboten. Die Besitzer verkaufen sie, weil die Geräte mit den neueren Windows-Versionen nicht mehr laufen. Und aus dem gleichen Grund will niemand diese Geräte haben. Mit etwas Glück finden sie als Käufer einen Linux-Anwender. Die Chance ist nämlich groß, dass auch ältere Peripherie auf Linux-Systemen tadellos funktioniert.

### 1.3.4 Sicherheitsfragen

Die offenen Quelltexte bieten aber auch einen Sicherheitsgewinn. Haben Sie eigentlich kein schlechtes Gefühl, wenn Sie einer Software Ihre Kontodaten überlassen? Dann wissen Sie vielleicht nicht, dass allein die Kenntnis Ihrer Bankverbindung genügt, um Geld von Ihrem Konto einzuziehen. Erst wenn Sie es rechtzeitig merken und die Lastschrift zurückrufen, behalten Sie Ihr Geld. Wer sagt Ihnen, dass die Software nicht heimlich

Ihre Daten abgreift und diese an Menschen verteilt, die auch gern mal ein wenig Geld einziehen möchten? Ach, Sie haben die Software von Ihrer Hausbank! Und Sie sind sicher, dass die dortigen Angestellten die Software auf Herz und Nieren geprüft haben. Aber was ist mit dem Sniper-Programm, dem Sie Ihr eBay-Passwort anvertrauen, oder mit dem Webdesign-Programm, das das Passwort Ihrer Website zum Hochladen der neuen Seiten verwendet? Jedes dieser Programme kann Ihre Daten missbrauchen und Sie würden es nicht einmal bemerken.

### Mehr-Augen-Prinzip

Wenn die entsprechende Software ihre Quellen offenlegt, kann ein Programmierer sehen, wenn das Programm Daten an eine fremde Adresse versendet. Und selbst dann, wenn Sie kein Programmierer sind und eine Manipulation selbst nicht erkennen könnten, würde ein solcher Eingriff bei quelloffener Software eher entdeckt, als wenn die Quelltexte nur dem Auftraggeber gegenüber offenliegen. Die Wahrscheinlichkeit, dass ein Bankangestellter die Hintertür in einer Software entdeckt, ist deutlich geringer, als wenn das gesamte Internet Zugriff auf die Quelltexte hat.

### Angriffe von außen

Die Sicherheit von Daten und damit die Rechte der Freiheit müssen inzwischen nicht mehr nur gegen Kriminelle verteidigt werden, sondern auch gegen die staatlichen Geheimdienste. Nach der derzeitigen Rechtslage in den USA kann man nicht ausschließen, dass amerikanische Unternehmen sich dem Druck der Terroristenverfolger nicht entziehen können und so Hintertüren einbauen müssen.<sup>2</sup> Die Aktivitäten gehen bis hin zur Bestechung von Mitgliedern von sicherheitsrelevanten Normungsgremien.<sup>3</sup> So muss man wohl davon ausgehen, dass alle Systeme, deren Source nicht öffentlich einsehbar sind, von vornherein untauglich für sicherheitskritische Anwendungen sind.

Sie sehen also, der Gedanke der freien Software hat durchaus praktische Bedeutung. Letztlich sind die Existenz, die Stabilität und die Sicherheit der Linux-Systeme den Leuten zu verdanken, die Wert darauf legen, dass Software frei ist und frei bleibt.

## 1.4 Inhaltsstoffe ohne Nebenwirkungen

Was ist das nun, das unter dem Namen Debian oder Ubuntu auf dem Computer installiert wird? Wie schon erwähnt, ist das die GNU-Software. Zum anderen gehört

<sup>2</sup> <http://www.heise.de/1972173.html>

<sup>3</sup> <http://www.heise.de/1953716.html>

## 1.4 | Linux und freie Software

der Linux-Kernel dazu, das TCP/IP, grafische Oberflächen, einige Serversysteme und andere Software, die zusammengenommen landläufig als Linux-Distribution angesprochen werden.

Das Allererste, was Sie allerdings zu sehen bekommen, wenn Sie die CD oder DVD starten, ist die Installationssoftware, die ebenfalls frei ist.

Die Installation basiert auf Softwarepaketen. Debian hat eine eigene Paketstruktur mit der dazugehörigen Software entwickelt, die es ermöglicht, die Kompatibilität und Abhängigkeit verschiedener Programme zu berücksichtigen. Auf dieser Basis setzt auch Ubuntu auf.

Die Softwarepakete werden auf Servern bereitgehalten und bei Änderungen aktualisiert. Auf diese Weise ist es möglich, den Systemen ständig den aktuellen Stand zur Verfügung zu stellen und Sicherheitslücken zentral zu bekämpfen.

### 1.4.1 Was wirklich drin ist

Debian und Ubuntu verwenden zu einem großen Teil die gleiche Software wie die Distributionen openSUSE oder Red Hat. In diesem Abschnitt erfahren Sie, welche Software zu einem Linux-System gehört und woher sie kommt.

#### **GNU-Software**

Eine Linux-Distribution enthält zunächst einmal die GNU-Software. Dazu gehören die vielen Kommandozeilenprogramme, die aus dem UNIX-Umfeld bekannt sind und die in Kapitel 4 dargestellt werden. Von besonderer Bedeutung ist der C/C++-Compiler, der von herausragender Qualität ist und mit dessen Hilfe die meiste Software unter Linux entstanden ist. Da der GNU-Compiler auf verschiedene Computersysteme portiert wurde, können Sie damit Software auf beinahe allen Plattformen zum Laufen bringen. Die GNU-Software wurde lange vor dem Erscheinen des Linux-Kernels bereits auf den verschiedenen Plattformen verbreitet. Unter den großen UNIX-Systemen wie Solaris, HP-UX oder AIX sorgte der GNU-Compiler für portable Programme und aktuelle Sprachenstandards. Aber die GNU-Programme finden Sie nicht nur bei UNIX-Derivaten. Auch unter OS/2, Windows oder dem Atari ST wurden sie eingesetzt, auch wenn sie dort oft nicht die gleiche Leistungsfähigkeit entfalten konnten.

Ziel der GNU-Bewegung war die Erstellung eines kompletten Betriebssystems. Es sollte ein Kernel entstehen, über dessen Konzept zum damaligen Zeitpunkt allerdings noch nicht vollständige Klarheit und Einigkeit herrschte.



### Linux-Kernel

Linux entstand in den frühen 90er-Jahren, als die GNU-Bewegung bereits seit vielen Jahren ziemlich aktiv war und mittlerweile viele Programme zur Verfügung standen. Als »Linux« wurde ursprünglich nur der Kern des Betriebssystems bezeichnet, der sogenannte Kernel. Die GNU-Bewegung war noch in der Phase des Entwurfs für einen eigenen Kernel, als Linus Torvalds seinen ersten Versuch ins Internet stellte. Er fand sehr viele interessierte Mitentwickler, und der Kernel, der weniger durch elegantes Design als durch effiziente Nutzung der vorhandenen Ressourcen bestach, wurde sehr populär. Der Kernel sorgt für die Verwaltung der Programme, der Ressourcen und den Kontakt zur Hardware mit seinen Treibern.

Durch die bereits existierenden GNU-Werkzeuge, die ja frei zur Verfügung standen, konnte aus dem nackten Kernel schnell ein System erstellt werden, das eigenständig auf einem Computer lief, ohne dass weitere Software benötigt wurde.

### Netzwerk von Berkeley

Die Netzwerkkomponente TCP/IP, die die Basis des Internets ist und heute weitgehend konkurrenzlos für Netzwerke steht, war von der Berkeley Universität entwickelt worden und damit freie Software. Sie ergänzte GNU und Linux um die Fähigkeiten eines Netzwerksystems. Damit war Linux nicht nur netzwerktauglich. Es hatte sogar die Originalquellen angezapft, die TCP/IP erfunden hatten.

### Grafik vom MIT

Das MIT (Massachusetts Institute of Technology) entwickelte mit dem X Window System die grundlegende Basistechnologie, auf dem bereits die Grafiksysteme der kommerziellen UNIX-Systeme aufsetzten. Das X Window System stand ebenfalls unter freier Lizenz zur Verfügung. Allerdings war die mitgelieferte Oberfläche nicht besonders attraktiv.

Die großen UNIX-Hersteller lieferten darum unterschiedliche kommerzielle Desktops wie beispielsweise CDE/Motif mit dem X Window System aus. Diese Desktops konnte das neu entstandene, freie Betriebssystem natürlich nicht verwenden. Aber dann stellte der UNIX-Hersteller Sun seine Oberfläche OpenView zur freien Verfügung, die unter Linux als erster Desktop diente.

Später wurden für Linux eigene Desktops entwickelt. Die Geschwindigkeit dieser Entwicklungen führte dazu, dass die Oberflächen GNOME und KDE sogar das kommerzielle CDE an Benutzerfreundlichkeit schnell in den Schatten stellten.

## 1.4 | Linux und freie Software

Heute gibt es für Linux sehr viele unterschiedliche Desktops. Das hat zwar den Nachteil einer nicht einheitlichen Benutzeroberfläche, aber auch den Vorteil, dass man sich nicht den Vorgaben des Herstellers beugen muss. Wenn einem Windows-Nutzer Vista nicht gefiel, musste er auf Windows 7 oder Windows 8 hoffen. Ein Linux-Besitzer, dem GNOME nicht gefällt, wechselt einfach zu KDE, XFCE oder Mate.

### **Sammelbecken**

Die Begeisterung über ein freies System führte zu einem neuen Boom der freien Software. Das GNU/Linux-System wurde zu einem Sammelbecken und gleichzeitig zu einer Inspiration für wichtige Softwaresysteme wie den Webserver Apache (siehe Kapitel 20), den Dateiserver SAMBA (siehe Abschnitt 18.1), Datenbanken wie MySQL (siehe Abschnitt 19.3) oder PostgreSQL (siehe Abschnitt 19.4) und viele andere freie Softwaresysteme.

Wenn man vom Betriebssystem »Linux« spricht, ist dies also nicht ganz korrekt, weil Linux ja eigentlich nur ein Teil des Betriebssystems ist. Die von der Free Software Foundation bevorzugte Bezeichnung GNU/Linux trägt der Tatsache Rechnung, dass erst aus der Kombination von GNU und Linux erstmals ein eigenständig lauffähiges System entstand. Diese Namensgebung betont, dass Linux selbst unter der GNU-Lizenz steht. Im allgemeinen Sprachgebrauch hat sich allerdings »Linux« als Bezeichnung für das ganze System durchgesetzt, das aus Linux, GNU, TCP/IP, X Window System und anderen Bestandteilen besteht. Das Weglassen des GNU im Namen anderer Distributionen heißt aber nicht, dass ihnen nicht klar wäre, dass es Linux ohne GNU in der heutigen Form nie gegeben hätte. Und in diesem Sinne bitte ich es auch zu verstehen, wenn in diesem Buch einfach von Linux die Rede ist.

### **1.4.2 Debian-Pakete**

Debian war eine der ersten Distributionen und legte von vornherein Wert darauf, dass auch das Installationsprogramm freie Software war. Um Software zu installieren, wird es in Pakete zusammengepackt, die oft voneinander abhängen. Ein solches Paket muss es ermöglichen, die Dateien an die richtigen Orte zu schaffen, aber auch zu prüfen, ob die Abhängigkeiten zu anderen Paketen erfüllt sind. Darüber hinaus muss die Software direkt nach der Installation konfiguriert werden.

Zu Anfang wurde eine Linux-Distribution in komprimierten Tar-Archiven (siehe Abschnitt 7.5) geliefert. Diese hatten zwei Nachteile: Es können keine Programme aus dem Paket gestartet werden, um eine Konfiguration aktiv zu gestalten. Vor allem aber konnte in den Tar-Archiven nicht festgelegt werden, welche Abhängigkeiten oder Unverträglichkeiten zu anderen Paketen bestanden.

Dieses Problem lösen die Debian-Pakete, die an ihrer Dateierweiterung *.deb* erkennbar sind. Auch Ubuntu verwendet diese Pakete.

Auch bei Red Hat wurde dieses Problem erkannt und durch ein eigenes Paket-Management namens RPM gelöst. So gibt es heute zwei Arten von Softwarepaketen: RPM von Red Hat und die Debian-Pakete. Zu den Paketen gehören natürlich auch die passenden Programme, die dafür sorgen, dass die Pakete installiert und die Abhängigkeiten überprüft und gegebenenfalls erfüllt werden.

Bei Sicherheitsproblemen muss es möglich sein, die Pakete zu aktualisieren. Debian und Ubuntu stellen Server zur Verfügung, die die jeweils aktuellen Pakete bereithalten. Nicht nur Updates funktionieren auf dieser Basis, sondern sogar Upgrades im laufenden Betrieb. Lediglich bei der Installation eines neuen Kernels kommt man nicht umhin, den Rechner neu zu starten.

## 1.5 Debian

In diesem Abschnitt erfahren Sie mehr über die Besonderheiten von Debian als Distribution und auch als Organisation.

### 1.5.1 Die Debian-Distribution

Debian veröffentlicht neue Releases im Schnitt alle zwei Jahre. Der Release-Zeitpunkt wird immer dann verschoben, wenn das Release noch nicht stabil ist. Der Inhalt eines Release wird recht ausgiebig getestet. Einige Monate vor seiner Veröffentlichung wird der Entwicklungsstand eingefroren. Es werden also keine neuen Versionen von Software-Paketen mehr hinzugenommen, sondern es wird nur noch auf Stabilität der nun festgelegten Komponenten getestet.

Damit es dennoch möglich ist, frei zwischen Stabilität und Aktualität zu wählen, gibt es die verschiedenen Release-Stände: *unstable*, *testing*, *stable* und *oldstable*. Bei erheblichen Umbauten des Systems hat es sogar schon eine Version *experimental* gegeben.

- **unstable:** Stabilität ist ein wichtiges Markenzeichen des Debian-Systems. Soll ein neues Debian-Release entstehen, wird es zunächst als *unstable* bezeichnet. Es werden die neuesten Versionen der Programme gesammelt. Ob die zusammengestellten Programme überhaupt miteinander arbeiten, ist jedoch nicht gewährleistet.

- **testing:** Die nächste Variante wird als *testing* bezeichnet. Hier werden wieder neue Programmupdates eingefügt. Dieses Release ist durchaus stabil und aktualisiert sich ungefähr in der Häufigkeit anderer Linux-Distributionen.
- **stable:** Die Variante *stable* ist ein *testing*-Release, das für etwa zwei Jahre eingefroren wird und nur durch Sicherheits-Updates verändert wird. Diese Variante ist ideal für Serversysteme, weil langfristig ein definierter Stand zur Verfügung steht.
- **oldstable:** Mit dem Erscheinen des neuen *stable*-Release wird das bisherige zum *oldstable*. Es wird noch etwa ein weiteres Jahr mit Sicherheits-Updates unterstützt.

Für einen Server werden Sie vermutlich ein *stable*-Release verwenden. Im Moment ist das Debian 7 namens *Wheezy*.

### 1.5.2 Wer hinter Debian steht

Für die Entwicklung der Installations- und Update-Software, die Pflege der Server und Zusammenstellung und Auswahl der Komponenten werden Menschen benötigt. Die Mitarbeiter von Debian sind in erster Linie von der Idee der freien Software begeistert. Die Stärke der Organisation ergibt sich aus der Identifizierung seiner Mitglieder mit den Zielen und der Art, wie man sie erreicht. Dazu sind demokratische Strukturen geschaffen worden. Es gibt eine eigene Verfassung und einen Vertrag, der die Zielrichtung der Distribution vorgibt: den Gesellschaftsvertrag.

#### Wählen und Mitmachen

Der Name »Debian« leitet sich von Ian Murdock her, der das Projekt am 16.8.1993 ins Leben rief. Der Vorname seiner Ex-Frau Debra und sein eigener Vorname ergaben zusammengesetzt den Namen des Projekts. Das Debian-Projekt wird von der Free Software Foundation unterstützt.

Im Gegensatz zu den anderen Distributionen steht hinter Debian keine Firma. Der Leiter des Projekts wird jedes Jahr demokratisch gewählt. Es mag auf den ersten Blick nach Vereinsmeierei aussehen, aber die Frage nach der internen Organisation ist wichtig. Das Projekt lebt vom freiwilligen Einsatz aller Beteiligten. Und da ist es unabdingbar, dass Entscheidungswege transparent sind und dass diejenigen, die die Arbeit machen, auch entscheiden können. Aus diesem Grund gibt es sogar eine Verfassung, die Sie unter der URL <http://www.debian.org/devell/constitution> finden. Wenn Sie selbst mitmachen wollen, finden Sie Informationen dazu unter der URL <http://www.debian.org/devell/join>.

### Der Gesellschaftsvertrag

Auch das Ziel des Debian-Projekts wurde demokratisch in einem »Debian-Gesellschaftsvertrag« festgelegt.

- Der Vertrag garantiert, dass Debian zu 100% frei bleiben wird. Das bedeutet insbesondere, dass Debian nicht von unfreien Komponenten abhängig werden darf.
- Ein weiterer Aspekt ist, dass Probleme nicht verborgen werden. Jeder Fehler, der gemeldet wird, wird sofort öffentlich angezeigt. In der Fehlerdatenbank von Debian werden die Fehler öffentlich diskutiert.
- Die Prioritäten werden bewusst auf Anwender und freie Software gelegt. Debian ist also kein Studienobjekt, sondern soll den Anwendern Nutzen bringen. Die Anwenderorientierung geht so weit, dass die Benutzer auch dann unterstützt werden, wenn sie unfreie Software unter Debian verwenden wollen. Debian garantiert sogar, dass das Debian-Projekt keine Gebühren erhebt, auch wenn Debian als Basis für unfreie Software verwendet werden sollte. Unfreie Komponenten werden nicht nur geduldet, sondern es wird sogar die Infrastruktur zur Verfügung gestellt. Allerdings gibt es eine klare Abgrenzung zwischen freien und unfreien Paketen, sodass der Benutzer auch ein Debian einsetzen kann, das ausschließlich auf freier Software basiert.

Der Vertrag erlaubt, dass andere Distributionen Debian enthalten. Dieser Punkt trifft beispielsweise auf Ubuntu zu, das nicht nur das Format der Debian-Pakete verwendet, sondern auch die Debian-Strukturen für die eigene Distribution nutzt. Sie finden den Vertrag unter der folgenden URL im Internet:

*[http://www.debian.org/social\\_contract](http://www.debian.org/social_contract)*

## 1.6 Ubuntu

Ubuntu ist eine Distribution, die vor allem vom Desktop her bekannt ist. Für das Aufsetzen eines Servers interessiert uns die Ubuntu-Server-Distribution.

### 1.6.1 Die Ubuntu-Distribution

Die Distribution basiert auf Debian, geht aber etwas lockerer mit der Frage der freien Software um. Gibt es keinen freien Treiber, wird eben ein weniger freier verwendet. Was für viele Benutzer eine Erleichterung ist, vermindert natürlich den Druck auf die Hersteller, Informationen zur Verfügung zu stellen, um freie Treiber herstellen zu können. Darum wird diese Haltung von einigen Vertretern der Open-Source-Bewegung kritisiert.

Während Debian ein neues Release herausgibt, wenn die Entwickler meinen, dass es nun fertig sei, gibt es bei Ubuntu zwei Releases pro Jahr, eins im April und eins im Oktober. Die Nummerierung der Release bezieht sich zunächst auf das Erscheinungsjahr gefolgt vom Erscheinungsmonat. Damit sind die Releases nur etwa ein Jahr aktuell, was etwas knapp für einen Server ist.

Die Aprilausgabe jeden geraden Jahres ist ein LTS-Release (Long Term Support). Es wird für fünf Jahre unterstützt. In dieser Zeit werden Sicherheits-Updates zur Verfügung gestellt und die Server stellen Software-Pakete zur Nachinstallation zur Verfügung. Das zur Drucklegung dieses Buchs aktuelle Ubuntu-Release 14.04 namens *Trusty Tahr* (»Zuverlässiger Tahr<sup>4</sup>«) wird also bis ins Jahr 2019 unterstützt.

### 1.6.2 Wer hinter Ubuntu steht

Der Unternehmer Mark Shuttleworth gründete 2004 die Firma Canonical Ltd., die hinter Ubuntu steht. Ziel der Distribution Ubuntu war es, Linux populär und einfach zu machen. Auf diesem Weg hat sich Ubuntu definitiv verdient gemacht.

Nachdem die Firma Canonical lange Zeit Ubuntu gesponsert hat, wurde in der letzten Zeit vermehrt der Fokus darauf gesetzt, wie die Firma profitabel arbeiten kann. Dies soll im Server-Bereich vor allem durch kostenpflichtigen Support erreicht werden.

Die Zielrichtung von Ubuntu betont also eher die einfache Bedienung und den Profit, und weniger die Freiheit der Software.

<sup>4</sup> Ein Tahr ist ein ziegenartiges Tier aus Asien.