

# Customizing einer Linux-Distribution

## Anpassen von Debian GNU/Linux an Spezialgebiete

Andreas Tille

4. Linux-Workshop der Max-Planck-Institute

Leipzig, 25. Februar 2005

# Gliederung

- 1 Debian für jeden Zweck
  - Was ist Debian?
  - Custom Debian Distributions
- 2 Techniken
  - Meta-Pakete
  - Nutzermenüs
  - Kommunikation
- 3 Ausblick

# Debian für Anfänger

- Linux = *Kern* des Betriebssystems
- Benötigt werden: *nutzerspezifische Anwendungen*
- Menge der Anwendungen um Linux-Kern = *Distribution*
- Firma, die Distribution zusammenstellt = *Distributor*
- Geschäftsmodell: Verkauf der Distribution, Nutzerunterstützung und -schulung
- Beispiele: Mandrake, RedHat, SuSE, ...
- Debian ist einer dieser Distributoren.

# **Verzeihung**

**Das war nicht korrekt.**

# Was ist Debian? (nächster Versuch)

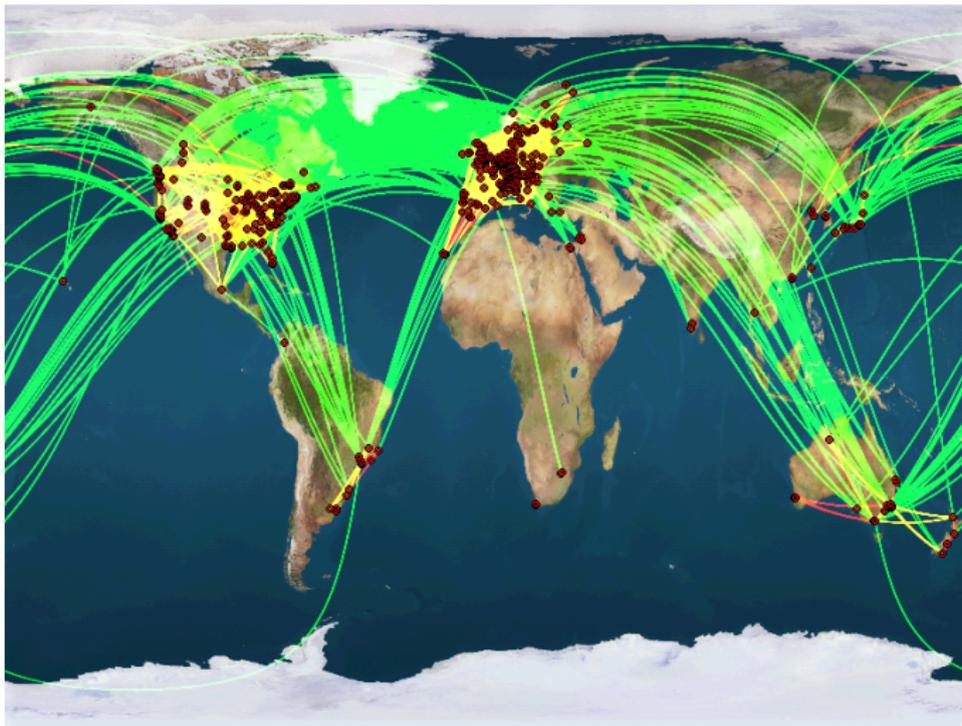
Das Debian-Projekt ist eine Gemeinschaft von Individuen die in Gemeinschaftsarbeit ein freies Betriebssystem entwickeln. Dieses Betriebssystem, das wir entwickelt haben, wird

*Debian GNU/Linux,*

genannt, oder einfach nur Debian.

Weiterhin wird daran gearbeitet, Debian mit anderen Kernen anzubieten, hauptsächlich für Hurd. Andere mögliche Kerne sind BSD und auch Portierungen für MS Windows<sup>TM</sup> sind denkbar.

# Netzwerk des Vertrauens



# Unterschiede

Kommerzieller Distributor

Firma

Mitarbeiter

CDs, Service

laut Marketingplan

zertifiziert

Einsteigern

rpm

marktorientiert

Struktur

Personen

Verkauft

Release

Oracle, SAP, etc.

bevorzugt von

Pakete

Anpassungen

Debian

Organisation

Freiwillige

nichts

wenn 0 RC-Bugs

läuft prinzipiell

Administratoren

deb

Do-O-Cracy

# Besonderheiten

- Gemeinsames Ziel: Bestmögliches Betriebssystem
- Maintainer: „Missing link“ Entwickler  $\longleftrightarrow$  Anwender
- Größte Sammlung sofort installierbarer Freier Software
- Prinzip Freier Software auf Distributions-Ebene:  
Entkoppeln von Wirtschaftsinteressen des Herstellers
- Zwei Möglichkeiten Debian GNU/Linux zu beziehen:
  - Vertrieb durch andere Distributoren auf CD
  - Üblicherweise Installation aus Internet
- Netzinstallation sehr komfortabel

# Debian anpassen

- Debian > 10000 Pakete
- Nutzer nur an *Untermenge* dieser Pakete interessiert
- Spezielle Nutzergruppen mit unterschiedlichen Fähigkeiten und Interessen
- Leichte Installation und Konfiguration durch Zusammenfassung spezifischer Pakete
- **Keine Abspaltung („fork“)** von Debian

*Grundidee: Es wird keine separate Distribution erstellt, sondern Debian für einen speziellen Zweck angepaßt.*

# Motivation

- 1 Profil der ins Auge gefaßten Nutzer:
  - Geringe technische Kompetenz
  - Zu aufwendig, Programme aus Quellen zu installieren
  - Kein Interesse an der Administration
  - *Spezifische Untermenge* verfügbarer Freier Software
  - Bedarf für leichte Bedienbarkeit
  - Exakt beschreibbare Sicherheitsanforderungen
  - Sprachbarriere
- 2 Profil der ins Auge gefaßten Administratoren
  - Begrenzte Zeitressourcen
  - Oft wiederholte Tätigkeiten automatisieren
  - Keine Spezialkenntnisse im Fachgebiet

# Existierende Custom Debian Distributions

- Debian-Junior - Debian für Kinder von 1 bis 99
- Debian-Med - Debian in der Medizin
- Debian-Edu (SkoleLinux) - Debian in der Schule
- DeMuDi - Debian Multimedia Distribution
- Debian-Desktop - Debian für jedermann
- Debian-Lex - Debian für Juristen
- Debian-NP - Nichtprofitorientierte Organisationen
- Debian-Enterprise - unternehmensrelevante Anwendungen
- Debian-Accessibility - Blinde und Sehgeschädigte
- Debian-eGov - E-Government

# Debian - adaptierbar für jeden Zweck?

- Etwa 1000 freiwillige Entwickler
- Flexibel, da nicht an kommerzielle Interessen gebunden
- Policy (=Regelwerk) sorgt für harmonisches Zusammenspiel aller Teile
- Gemeinsames Ziel aller Entwickler:  
Das beste Betriebssystem für sich selbst zu erhalten.
- Entwickler haben im „realen Leben“ Kinder oder arbeiten im medizinischen Umfeld etc.
- Entwickler haben Freiheit, Visionen zu verwirklichen, ohne Einschränkung durch Firmeninteressen
- Jeder Entwickler kann Einfluß auf die Entwicklung von Debian nehmen - er muß es einfach nur *in Angriff nehmen*

*Do-O-Cracy*

# Was sind Programmpakete

- Hauptbestandteil der Arbeit des Distributors
- Enthält
  - Ausführbare Programme
  - Konfiguration
  - `{pre/post}install + {pre/post}remove` Skripte
  - Beziehungen zu anderen Paketen

→ Mehr als nur ein Archiv von Dateien
- Ermöglicht einfache Handhabung
  - Sichert Vorhandensein aller Abhängigkeiten
  - Saubere Upgrades
  - Einfache Handhabung von Sicherheitsupdates

→ Basis moderner Linux Distributionen

*Problem: Wachsende Anzahl der Pakete*

# Kurzeinführung Metapakete bauen

## Control Datei

```
$ cp -a /usr/share/doc/cdd-dev/examples/tasks .  
$ cat tasks/README  
$ edit tasks/task1  
Task: Taskname  
Description: Kurzbeschreibung  
Langbeschreibung wie in debian/control Datei  
Depends: dependency1, dependency2, ...  
Suggests: suggests1, suggests2, ... (optional)  
$ cp tasks/task1 tasks/<Meta-Paket-Name>
```

*Für jedes Meta-Paket wird dieses Gerüst einer  
debian/control Datei benötigt*

siehe */usr/share/doc/cdd-dev/examples/tasks*

## Kurzeinführung Metapakete bauen (2)

### Verzeichnis debian

```
$ cp -a /usr/share/doc/cdd-dev/examples/debian .  
$ cat debian/README  
$ edit debian/control.stub  
Ändern der Variablen des Beispiels  
$ edit debian/rules  
Ändern der Variablen _CDD_  
$ make -f debian/rules dist
```

Quell - tar.gz ist fertig

### Paket bauen

```
$ debuild
```

*Einfachste Variante erledigt*

# Common Meta-Paket

## Optional

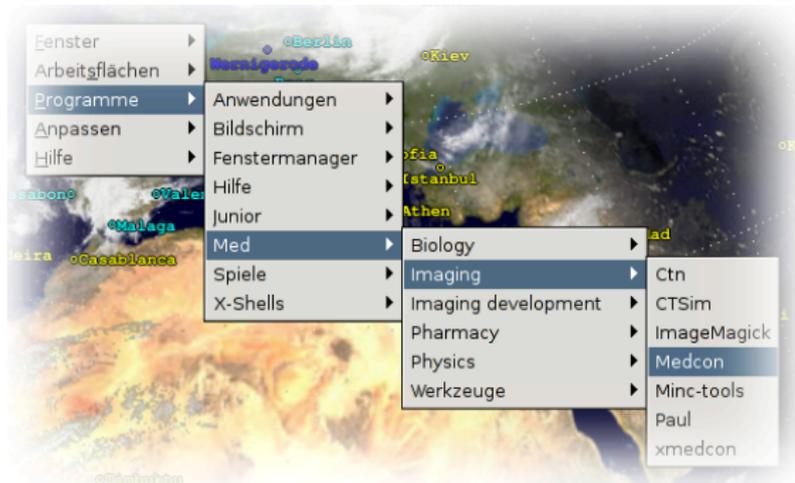
```
$ cp -a /usr/share/doc/cdd-dev/examples/common .  
$ cat common/README  
$ edit common/conf common/control common/common.1  
Ändern der Variablen _CDD_
```

## *Initialisierung der Rollen-basierten Nutzer-Menüs*

- Meta-Pakete hängen automatisch von `common` ab
- Installiert *auto-apt* Hilfsdatei
- Dringend empfohlen

# Rollen-basierte Nutzermenüs

- Nutzer mit Rolle `<cdä>` erhält zusätzliches Menü



- Andere Nutzer werden nicht damit belästigt
- Implementiert in aktuellen Debian-Med Paketen
- Rolensystem mittels flexibler Plugins

# Ändern von Nutzermenüs

## Optional

```
$ cp -a /usr/share/doc/cdd-dev/examples/menu .  
$ cat menu/README  
$ cat menu/task1/README  
$ edit menu/task1/dep1.menu  
$ edit menu/task1/dep2.txt  
$ edit menu/task1/dep3.html
```

Editieren zu gültigen Menü-Einträgen und sinnvollen Beschreibungen

## Task-Namen vergeben

```
$ cp menu/task1 menu/<meta package name>
```

*Nutzermenü-Einträge für jede Abhängigkeit - zumindest mit Dokumentation*

# Entwickler Plattformen

- Projekt auf Alioth
- Subversion Repository
- Gemeinsame Entwickler-Mailingliste  
*Von dieser Liste intensiv Gebrauch machen!*
- Nutzer Mailinglisten für die meisten CDDs  
`http://lists.debian.org/debian-<cdd>`
- Web Seiten der meisten CDDs  
`http://www.debian.org/devel/debian-<cdd>`

# Aktueller Stand

- Wahrscheinlich populärste CDD: Debian-Edu (SkoleLinux)
- Rückintegration: DeMuDi
- Anleitung zum Start einer neuen CDD
- Grundproblem: Rad wird oft neu erfunden
- Debian-Fork möglich → aber CDD sinnvoller
- Fork: Ständig Anpassungen nötig
- CDD: Arbeit fällt genau *einmal* an
- Debian = Geben und Nehmen

# Zukunft

- Mehr Custom Debian Distributions
- Weiterentwicklung der CDD-Werkzeuge
- Idee unter Entwicklern und Nutzern publik machen
- Angepaßte Live-CDs von CDDs
- Neue Methode zur Verbreitung von Debian
- Spezielle *Package* Dateien für jede CDD

Der Vortrag ist verfügbar unter  
<http://people.debian.org/~tille/talks/>  
Andreas Tille <tille@debian.org>