

Kepler Seminar, 12.1.2001:

Open Source: Software (R)evolution

Die Notwendigkeit freier Software
Entstehung - Prinzipien - Kommerz

Jan Theofel

ETES GbR (www.etes.de) und
Kepler Seminar (www.stiftung-sbb.de)

Vortragsfolien unter:

<http://www.etes.de/vortraege/2001-01-12/>

Übersicht

1. Vorwort / Begriffe
2. Was ist freie Software?
3. Die Geschichte freier Software
4. Vorteile freier Software
5. Nachteile freier Software
6. Die GPL-Lizenz
7. Vergleich zu anderen Lizenzen
8. Paradebeispiel: Linux
9. Linux: Vor- und Nachteile
10. Linux-Distributionen
11. Andere OpenSource-Betriebssysteme
12. Andere OpenSource-Software
13. Freie Software und Kommerz
14. Gefährdung der freien Software
15. Join us!

Vorwort / Begriffe

Hacker

≠ böser Cracker

= Programmierer

Sourcecode – Binary

Sourcecode = Programmbefehle für Menschen lesbar

Binary = Maschinencode

Die *Geeks* unter euch mögen mir verzeihen,
daß ich oftmals stark vereinfache.

Was ist freie Software?

1. Unterschied:

- Freibier
- Freie Meinungsäußerung
- „Think of free speech, not free beer.“ (RMS)

2. Vier Freiheiten (Vergleich: „bill of rights“):

- Ein Programm zu jedem beliebigen Zweck einsetzen.
- Ein Programm zu analysieren und (teilweise) selbst zu übernehmen.
- Ein Programm zur Hilfe anderer weiterverbreiten.
- Ein Programm verbessern und diese Verbesserungen veröffentlichen.

3. Veröffentlichung als Source!

4. „Synonym“: Open Source Software (OSS)

Die Geschichte freier Software

1969 bis 1970:

Aus Multics wird UNIX entwickelt

(KEN THOMPSON und DENNIS RITCHIE, Bell Labs)

1971:

Hardwareunabhängigkeit von UNIX 4 durch

Programmierung in C (extra entwickelt!)

1975:

AT&T darf UNIX nicht kommerziell vermarkten, Source wird veröffentlicht, schnelle Entwicklung von UNIX

1977:

Beginn der Implementierung von T_EX

1983:

„Free UNIX Posting“ → GNU Projekt (RMS)

1984:

AT&T darf UNIX wieder vermarkten, es entsteht UNIX V,

Gründung der FSF (RMS)

1991:

LINUS TORVALDS beginnt mit der Linux-Implementierung

ab 1993:

Linux wird auf andere Rechnerarchitekturen portiert
(Zuerst wird das 68k Linux für Amiga/Atari 1994/95 fertig)

1995:

Kernel 1.2 erscheint mit beeindruckender
Netzwerkfunktionalität

1998:

„Open Source“ (Palo Alto, Bruce Perens, ESR)
Netscape Quellen werden freigegeben
Gimp 1.0
212% Wachstum für Linux

1999:

Linux wird von IBM, HP, . . . unterstützt
Entgeltliche Etablierung im kommerziellen Umfeld

2000 / Heute:

SO / IRIX-FS / . . . Quellen frei
Über 10 Millionen Linux-Anwender

Vorteile freier Software

1. Alle dürfen teilnehmen
2. Spezialistenwissen fließt mit ein
3. Anwender = Programmierer
4. Direkter Autorenkontakt
5. Ständig neue „developer versions“
6. Ständige Weiterentwicklung
7. Source zum Lernen verfügbar
8. Keine Bindung an Firmen
9. Geringer Preis / Verfügbarkeit im Netz
10. Geringe TCO (Total Cost of Ownership)
11. Hilfe und Doku aus der „Community“
12. Fun durch die „Community“
13. Lösungsansatz für gemeinsame Probleme
14. Schnelle Bug- und Security-Fixes
15. Extrem hohe Sicherheit
16. Garantiert die Freiheit (keine Filter!)

Nachteile (?) freier Software

Gängige Vorurteile:

1. Keine Produkthaftung
2. Keine Bedienungsstandards
3. Schwierig zu Bedienen / Installieren
4. Teurer als kommerzielle Software (TCO)
5. Potentiell unsicher
6. „was nix kostet taugt auch nix“
7. „Softwarekommunismus“

Vorteile proprietärer Software:

1. Scheinbare Deckung der Entwicklungskosten
2. Möglichkeit der Werbung
3. Gekaufte Vergleiche
4. Kundenbindung durch Knebellizenzen

Die GPL-Lizenz

- GPL = General Public License
- von der FSF entwickelt (1989 – 1991)
- aktuell: Version 2 vom 2.6.1991
- Sichert den Benutzern die vier „Freiheiten“
- Schützt freie Software vor Kommerzialisierung
- Schützt die Autoren durch Haftungsausschluß
- Anhang beschreibt die Anwendung auf Programme
- Spezielle Anpaßung für libraries:
GNU LGPL Lesser General Public License
(früher Library General Public License)
- Verbietet das Verwenden der Software bei
Mißachtung der GPL!
- „GPL-Virus“ infiziert alle freie Software

Vergleich mit anderen Lizenzen

LGPL

wie GPL, aber Schnittstellen zu proprietärer Software werden erlaubt

BSD

Ursprüngliche Autoren müssen genannt werden

Haftungsausschluß

Recht auf Kommerzialisierung

Ursprünglich: BSD Varianten

BSD-artige Lizenzen

ähnlich der BSD Lizenz, aber kein Bezug auf Berkeley

XFree, Apache, ...

Artist License

ursprünglich Perl

Verkauf nur im Bündel

umgehbar, daher oft Doppellizenzierung

NPL/MPL/...

Spezielle Lizenzen von Firmen, die Source freigeben

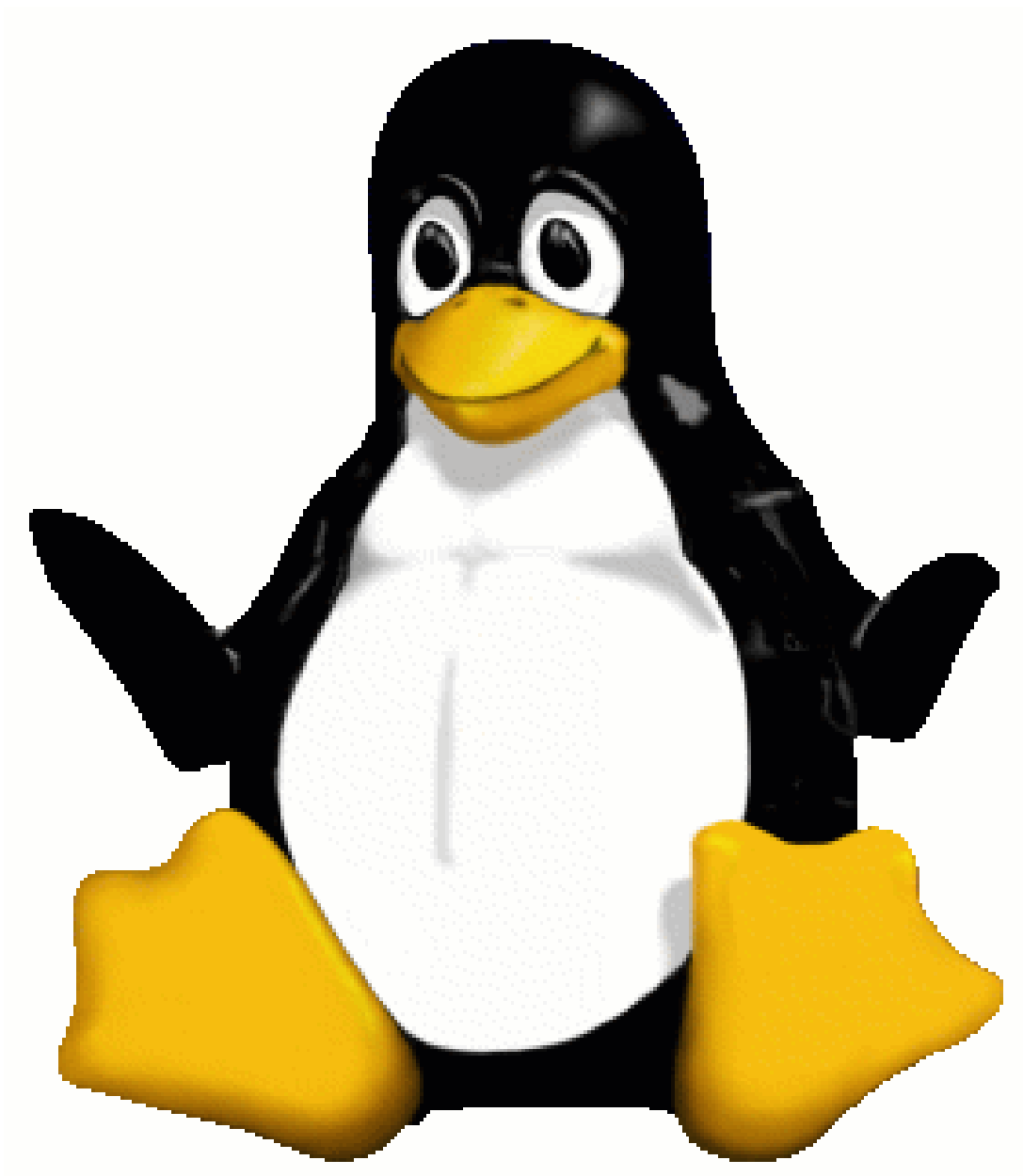
Es werden besondere Rechte vorbehalten

Parade-Beispiel: GNU/Linux

Was ist Linux?

Nur (!) der Kernel, Rest ist GNU

„Where do you want to go tomorrow?“



Linux Maskottchen „TUX“

Linux-Distributionen

SuSE: (7.0)

höchster Anteil in Deutschland, benutzerfreundlich

RedHat: (7.0)

Spezielle Version komplett unter GPL, besonders
Amerika

Halloween: (5)

deutsche Anpassung von RedHat (LOB)

Debian: (2.2)

nicht kommerziell

Slackware: (?)

sehr viel Handarbeit, inzwischen sehr selten

LinuxPPC: (Release 5)

für PowerPC

Weitere (kleinere) Distributionen:

Caldera OpenLinux, PTS Linux, Corel Linux, EasyLinux,
Mandrake Linux (7.0), RockLinux, ...

Disketten-Linux:

muLinux, tombsr, ...

Sommer 2000:

ungefähr 180 verschiedene Distributionen

Linux: Vorteile

- Viel Software (bis 12 CD/Distribution / 1 DVD)
- Fast kostenlos
- Geringe laufende Kosten
- Sehr schnell
- Kommandozeile und GUI
- Server nicht GUI-lastig
- Flexibel & individuell anpaßbar
- Effiziente Tools
- Optimale Anpassung durch eigenes Compilieren
- Echtes Multitasking & Multiuser
- Vollwertiger Server (Datei-, Drucker-, Datenbank- und Kommunikationsserver)
- Hohe Sicherheit (lokal & Netzwerk)
- Stabil (kaum „allgemeine Schutzverletzungen“)
- Hochskalierbar (Palm, PC, S/390, Cluster)

Linux: Nachteile (?)

Größtenteils Vorurteile:

- Schwierige Installation
- Kaum Spiele
- Z.T. Mangelhafte Hardwareunterstützung
- Nachholbedarf bei (MS-)Standardsoftware
- Keine Produkthaftung
- Maximal 4GB Ram
- Schwierige Bedienung
- Keine einheitliche Shell / GUI
- Keine Standardisierungen

Andere OSS-Betriebssysteme

1. BSD (Berkly Standard Distribution)
 - 386BSD (früher)
 - NetBSD
 - FreeBSD
 - OpenBSD
2. FreeVMS (*nicht* OpenVMS)
3. OpenDOS

Andere OSS-Software

- BIND / DNS (Domain Name System)
- Apache (Webserver, Proxy, SSL)
- sendmail (MTA - Mail Transfer Agent)
- Samba (Windows-Server)
- XFree86 (Grafische Oberfläche)
- T_EX bzw. L^AT_EX (Textsatzsystem)
- Gimp (Grafikbearbeitung)
- ExtremeLinux, Beowulf (Clustering)
- mySQL (SQL-Datenbank-Server)
- KOffice, OpenOffice (Office-Paket)
- emacs, vi (Editoren)
- Perl, PHP3, Tcl/Tk (Programmiersprachen)
- gcc (Compiler)
- FreeCIV, Pingus, ... (Spiele)
- POV-Ray (Raytracer)
- Mozilla

Kommerz

Gründe für den kommerziellen Erfolg von Linux:

- Hohe Qualität der Software
- Geringer Preis
- Firmen können selbst mitentwickeln
- Entwickler als Kompetenzgarantie
- Einen Namen machen
- Geld durch Distribution / Support
- Einstieg der Branchenriesen (IBM, HP, Intel, ...)
- Hat proprietäre Software Zukunft?
- GPL als Knebellizenz

Gefährdung der freien Software

- Softwarepatente
- Know-How kann nicht gekauft werden
- Halb-freie Software v.a. Libraries
- Notwendig: Freie Dokumentation

Join us!

Nutzervergleich:

Nur 10 Millionen Linux User?

Wieviele Milliarden haben sich noch nicht entschieden?

Was kann jeder tun?

- Anwenden
- Programmieren
- Dokumentieren
- Anderen helfen
- Verbreiten