Einblicke in die Linux Kernel Entwicklung

Dipl.-Math. techn. Detlev Zundel (dzu@member.fsf.org)

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.

22. Oktober 2016

1 Geschichtsabriss
- Ken Thompson
- Richard Stallman
- Andrew Tanenbaum
- Linus Torvalds

2 Zurück in die Zukunft
- Linux heute
- Entwicklung
- Versionsverwaltung

Ken Thompson

Unix™(1)

- Unix begann als ein Freizeitprojekt von *Ken Thompson*, damals bei den **Bell Labs** beschäftigt.
- Erfahrungen der Entwicklung von **MULTICS**, welches sich als zu komplex herausstellte wurden eingearbeitet
- Erstes in einer eigens dafür entwickelten Hochsprache (`C`) implementiertes Betriebssystem
- Wurde dann in Betriebssystemkursen an Universitäten eingesetzt

Unix™(2)

- Mit wachsendem Bekanntheitsgrad wollte **AT&T** den Erfolg versilbern. Aufgrund eines Gesetzurteils durfte das Betriebssystem nicht verkauft werden, also wurde der Quellcode lizensiert
- Da der Quellcode vorhanden war, entwickelte und diversifizierte sich Unix sehr schnell. Es entstanden u.a. **AT&T** Unix, **BSD** (Berkeley University), **Ultrix** (DEC), **AIX** IBM, **Solaris** (Sun), **HP-UX** (HP), **Xenix** (Microsoft)

Richard M. Stallman

GNU (1)

- Im März 1985 veröffentlicht *Richard Stallman* das **GNU Manifest** (**G**nu is **N**ot **U**nix) als Grundlage für ein *komplett Freies Betriebssystem* (Kern + Utilities)
- Rechtliche Basis ist die **GPL** Lizenz (auch *Copyleft* genannt) mit den 3 Freiheiten:
- Um den Umstieg zu erleichtern, benutzte er das Beste vorhande OS als Grundlage: Unix
- Damit war auch die Koordination von Freiwilligen einfach

GNU (2)

- Stallman selbst entwickelte **Emacs**, **GCC** und **GDB** als Grundlage
- Viele Freiwillige begannen (überlegene) **GPL** Versionen der bekannten Unix Tools beizutragen (`ls`, `make`, `sed`, ..)
- Die **GNU Tools** etablierten sich auch auf den proprietären Unices
- Bis 1992 war der Löwenanteil geschafft - einziger fehlender Teil war ein **Kernel**
- Die **FSF** begann die Arbeit an dem Micro-Kernel "**The Hurd**"

GPL

Die **GPL** garantiert die folgenden Freiheiten (*"Free as in Free Speech not as in Free Beer"*):

- Das Programm darf in jeder gewünschten Form genutzt werden
- Der Quellcode darf eingesehen und geändert werden um es nach eigenem Wunsch zu nutzen
- Kopien dürfen erstellt und weitergegeben werden
- Auch veränderte Versionen dürfen weitergegeben werden

Veränderte Versionen unterliegen selbst wieder der **GPL**

Andrew Tanenbaum

Minix

- Für das Buch "*Operating Systems: Desgin and Implementation*" Re-Implementierte *Andrew Tanenbaum* einen Unix artigen Betriebssystem Kernel 1987 und nannte ihn **MINIX**. Dieser stabd unter dem gewöhnlichen Copyright (war also nicht *frei*).
- Quellcode des Kerns war vorhanden, der C-Compiler aber z.B. nicht

Linus Torvalds (1991)

Linux (1)

- In einer Usenet Newsgroup gab *Linus Torvalds* 1991 bekannt, dass er an einem Unix artigen Kernel für die Intel 80286 CPU arbeitet (**IBM AT**)
- *"It is NOT portable and it probably never will support anything other than AT-hard disks, as that's all I have :-("*
- Für den **Linux** Kernel hatte Torvalds die **GPL** als Lizenz gewählt, ohne sich mit den ethischen Zielen der Freien Software zu identifizieren

Linux (2)

- Damit stand zum ersten Mal ein *vollständig freies Betriebssystem* zur Verfügung
- Dummerweise erlangte das System nur als **Linux** Bekanntheit, obwohl es ohne das GNU Projekt nie möglich gewesen wäre.
- Versuche vn Stallman den Kernel in **Lignux** umzubenennen, scheiterten
- Um dem GNU Anteil gerecht zu werden sprechen wir vom **GNU/Linux** System

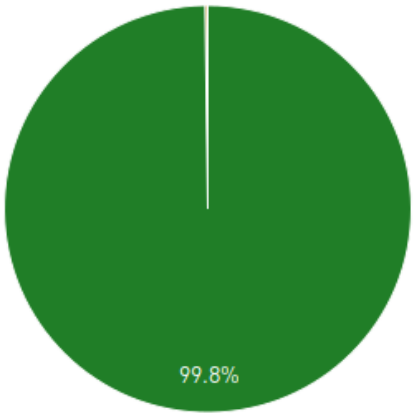
List Statistics | TOP500 Supercomputer Sites – Mozilla Firefox

List Statistics | TOP5...

https://www.top500.org/statistics/list/

Search

Operating system Family Performance Share



Click a row in the table below to drill down.

Operating system Family	Count	System Share (%)	Rmax (GFlops)	Rpeak (GFlops)	Cores
Linux	497	99.4	566,151,904	844,080,421	40,965,609
Unix	3	0.6	1,201,135	1,405,914	49,504

Top 500 Juni 20016

Linux Entwicklung (1)

- Versionen werden Kalender-basiert, nicht Feature-basiert freigegeben
- Zwischen zwei Versionen liegen ca. 66 Tage
- Mit Veröffentlichung einer stabilen Version öffnet sich das sogenannte *"Merge Window"* 2 Wochen lang um neue Änderungen aufzunehmen
- Das Ergebnis ist die **-rc1** Version
- Nur noch Bugfixes werden in die **-rc2**, **-rc3**, etc. Versionen aufgenommen bis diese Version als stabil erklärt wird.

Linux Entwicklung (2)

- Also z.B. **v4.5** wird veröffentlicht, Ergebnis des *Merge Windows* ist dann **v4.6-rc1**
- Nach **v4.6-rc2**, **v4.6-rc3**, etc. wird nach ca. 66 Tagen **v4.6** freigegeben
- Bugfixes werden in den stabilen Kernel mit einer weiteren Versionsnummer eingepflegt, also z.B. **v4.8.1**, **v4.8.2**, etc.

Linux Versionsverwaltung

- Kernelentwicklung erfolgt komplett dezentral
- Daher immer dezentrales Versionskontrollsystem, selbst als es noch kein freies solches gab (**BitKeeper**).
- Lizenzänderung von BitKeeper stoppte Kernel-Entwicklung
- Als Konsequenz konzipierte Linus seine eigene Versionsverwaltung: **git**

Git

- Völlig dezentrale Versionsverwaltung
- Jedes *"Repository"* ist gleichberechtigt
- Änderungen zwischen Repositories können extrem leicht übernommen werden
- Von Linux-Entwicklern für Linux entwickelt
- Heute schnellstes dezentrales System und de-facto Standard in der Freien Software

Kernel.org git repositories – Mozilla Firefox

Kernel.org git reposit...

https://git.kernel.org/cgit/

kernel/git/tiwai/hda-emu.git	HD-audio emulator program
kernel/git/tiwai/linux-firmware.git	Unnamed repository; edit this file 'description' to name the repository.
kernel/git/tiwai/salsa-lib.git	SALSA-library development repository
kernel/git/tiwai/sound-unstable.git	Unnamed repository; edit this file 'description' to name the repository.
kernel/git/tiwai/sound.git	Sound sub-system tree
kernel/git/tj/cgroup.git	cgroup export tree
kernel/git/tj/libata.git	libata export tree
kernel/git/tj/misc.git	miscellaneous kernel branches
kernel/git/tj/percpu.git	percpu export tree
kernel/git/tj/wq.git	workqueue export tree
kernel/git/tmlind/linux-omap.git	OMAP development tree
kernel/git/tomba/linux.git	Tomi Valkeinen's kernel tree
kernel/git/torvalds/linux.git	Linux kernel source tree
kernel/git/torvalds/pesconvert.git	PES-file thumbnailer, Brother embroidery format
kernel/git/torvalds/uemacs.git	Micro-emacs
kernel/git/trenn/cpupower.git	Latest cpupower work
kernel/git/tyhicks/ecryptfs.git	eCryptfs development tree
kernel/git/tytso/ext4.git	Ext4 filesystem tree
kernel/git/tytso/random.git	/dev/random tree
kernel/git/ulfh/mmc.git	Unnamed repository; edit this file 'description' to name the repository.
kernel/git/vapier/blackfin.git	Blackfin arch tree
kernel/git/vapier/headers.git	Exported sanitized userspace headers
kernel/git/vapier/lemote.git	MIPS Lemote kernel tree mirror of http://dev.lemote.com/cgit/
kernel/git/vfs/vfs.git	VFS group's fork of linux.git
kernel/git/vgupta/arc.git	ARC (Synopsys) arch tree
kernel/git/vireshk/linux.git	My development tree

Ausschnitt der Linux Repos auf kernel.org

- Per Konvention ist das Repo von Linus *der kanonische* Linux Kernel (*"Mainline"*)
- Änderungen fließen durch mehrere Repos bis sie dort ankommen
- Alle Änderungen werden öffentlicht auf *Mailing Listen* diskutiert
- Aufnahme erfolgt erst, wenn *alle* Beteiligten zufrieden sind
- Entwicklung organisiert sich völlig selbstständig

Zwei Wochen, was kann da schon passieren?

```
[dzu@harry linux (master)]$ git diff --stat v4.7..v4.8-rc1
 .cocciconfig                                       |     3 +
 .gitignore                                         |     2 +
 .mailmap                                           |    18 +-
 Documentation/.gitignore                           |     2 +
 Documentation/00-INDEX                             |     4 +-
 Documentation/ABI/testing/configfs-acpi            |    36 +
 Documentation/ABI/testing/configfs-iio             |    13 +
 Documentation/ABI/testing/sysfs-bus-iio            |    14 +
 .../ABI/testing/sysfs-bus-iio-health-afe440x       |    63 +-
 .../ABI/testing/sysfs-class-net-batman-adv         |    20 +-
 Documentation/ABI/testing/sysfs-class-pwm          |     9 +
 Documentation/ABI/testing/sysfs-devices-system-cpu |    10 +
 Documentation/CodingStyle                          |     2 +-
 Documentation/DMA-API.txt                          |    33 +-
....
 virt/kvm/arm/vgic/vgic.h                           |    38 +-
 virt/kvm/irqchip.c                                 |    35 +-
 virt/kvm/kvm_main.c                                |   155 +-
 10910 files changed, 647518 insertions(+), 307408 deletions(-)
[dzu@harry linux (master)]$
```

Wer sind die Entwickler von v4.8?

Lines changed

Samsung	120693	14.4%
Intel	104291	12.4%
(None)	102848	12.3%
Red Hat	48563	5.8%
IBM	42298	5.0%
Mellanox	29226	3.5%
(Unknown)	27671	3.3%
Linaro	22960	2.7%
Broadcom	18040	2.2%
Cisco	17868	2.1%
MediaTek	16292	1.9%
QLogic	15986	1.9%
ARM	14397	1.7%
Renesas	14283	1.7%

Changesets

Intel	1960	14.8%
Red Hat	1143	8.6%
(Unknown)	806	6.1%
(None)	746	5.6%
Linaro	662	5.0%
IBM	654	4.9%
Samsung	637	4.8%
SUSE	338	2.6%
Google	294	2.2%
AMD	281	2.1%
Oracle	259	2.0%
Texas Instruments	258	1.9%
Mellanox	243	1.8%
Renesas Electronics	223	1.7%

That's all folks!

Fragen?