Open Source

How not beeing evil pays out.

Kai Dietrich

3. Juni 2009
Vorlesung Information Rules 1, TU-Berlin

created with LATEX-Beamer

Open Source
schon mal von gehört?

These:

These:
Es ist *ökonomisch effizienter*
freie Software zu produzieren
als proprietäre Software.

Was davor war

proprietäre Softwareentwicklung

Software als:

- Werk (Urheberrechte/Copyright)
- Erfindung (Patentrechte)
- Produkt (Markenrechte)

Was davor war

proprietäre Softwareentwicklung

Software als:

- Werk (Urheberrechte/Copyright)
- Erfindung (Patentrechte)
- Produkt (Markenrechte)

Was davor war

proprietäre Softwareentwicklung

Software als:

- Werk (Urheberrechte/Copyright)
- Erfindung (Patentrechte)
- Produkt (Markenrechte)

$\Rightarrow$ 3x Schutz, 3x Exkludierbarkeit

Bis eines Tages …

Bis eines Tages …

GNU GPL

- GNU = GNU is Not Unix (Operating System)
- GNU General Public License
- v1.0 – 1989, v2.0 – 1991
- Essential Freedoms:
  - Use
  - Modify
  - Distribute
  - Distribute Modifications
- zusätzliche Bedingungen

Lizenzen

Lizenzen

- Lizenzvertrag = Vertrag sui generis (nicht BGB vertypt)
- bei Kauf: Teil der AGBs
- Transaktion:
  - einfache Nutzungsrechte
  - exklusive Nutzungsrechte
- Deutschland: nicht Urherberrecht, nur Nutzungs-/Verwertungsrechte übertragbar
- Gebrauch für
  - Marken
  - Patente
  - Verwertungsrechte

copyleft

Copyleft

- Ziel: Freiheiten erhalten
- Weitergabe nur erlaubt wenn:
  - *vollständiger* Source Code beiliegt
  - abgeleites Werk wieder unter GPL steht
- starkes Copyleft:
  - Linking = ableiten eines Werkes
  - GPL Library → GPL Programm
- schwaches Copyleft (LGPL):
  - LGPL Library → whatever Programm

Die Schaffung eines freien Raums

	Exkludierbarkeit	
Rivalität	hoch	niedrig
hoch	Privatgut	Allmendegut
niedig	Clubgut	öffentliches Gut

Die Schaffung eines freien Raums

Rivalität	Exkludierbarkeit	
	hoch	niedrig
hoch	Privatgut	Allmendegut
niedig	Clubgut	öffentliches Gut

Urheberrecht

Die Schaffung eines freien Raums

Rivalität	Exkludierbarkeit	
	hoch	niedrig
hoch	Privatgut	Allmendegut
niedig	Clubgut	öffentliches Gut

GPL

Die Schaffung eines freien Raums

| | Exkludierbarkeit | |
Rivalität	hoch	niedrig
hoch	Privatgut	Allmendegut
niedig	Clubgut	öffentliches Gut

GPL

Software ist mit Urheberrecht freier als ohne

Der Linus-Torvalds Fall

Der Linus-Torvalds Fall

Hello everybody out there using minix –
I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones. [...]
I've currently ported bash(1.08) and gcc(1.40), and things seem to work. This implies that I'll get something practical within a few months, and I'd like to know what features most people would want. Any suggestions are welcome, but I won't promise I'll implement them :-)
Linus (torvalds@kruuna.helsinki.fi)

25. August 1991, comp.os.minix

Scratching an Itch

Every good work of software starts by scratching a developer's personal itch.

Eric S. Raymond – The Cathedral and the Bazaar

Vergleich

	Entwicklungsmodell	
	proprietär	offen
Entwickler-Allokation	zentral	self-selection
Motivation	extrinsisch	extrinsisch + intrinsisch

LKML.ORG

Latest kernels

stable	2.6.29.4	patch	log
stable prepatch	2.6.30-rc7	patch	log
stable snapshot	2.6.30-rc7-git4	patch	
2.4 release	2.4.37.1	patch	log
2.2 release	2.2.26	patch	log
2.2 prepatch	2.2.27-rc2	patch	log
stable mm patch	2.6.28-rc2-mm1	patch	

Latest messages

Jiri Slaby	broken ide-tape
tom.leiming@gmail ...	[PATCH 8/8] kernel:lockdep:add statistics info for...
tom.leiming@gmail ...	[PATCH 7/8] kernel:lockdep: update memory usage in...
tom.leiming@gmail ...	[PATCH 6/8] kernel:lockdep: implement lockdep_coun...
tom.leiming@gmail ...	[PATCH 5/8] kernel:lockdep:introduce print_shortes...
tom.leiming@gmail ...	[PATCH 4/8] kernel:lockdep:implement find_usage_*w...
tom.leiming@gmail ...	[PATCH 3/8] kernel:lockdep:implement check_noncirc...
tom.leiming@gmail ...	[PATCH 2/8] kernel:lockdep: introduce match_functi...
tom.leiming@gmail ...	[PATCH 1/8] kernel:lockdep:improve implementation ...
tom.leiming@gmail ...	[PATCH 0/8] kernel:lockdep:replace DFS with BFS
Alan Stern	Re: [Bug #13372] Oops in usb-serial with keyspan a...
Rik van Riel	Re: [PATCH] Use kzfree in crypto API context initi...
Rik van Riel	Re: [PATCH] Use kzfree in crypto API context initi...
Arjan van de Ven	Re: [patch 0/5] Support for sanitization flag...

Hottest messages

Eric Paris	Introducing SELinux sandbox, confining untrusted b...
Jeremy Fitzhardinge	Xen is a feature
Linus Torvalds	Linux 2.6.30-rc7
Ingo Molnar	Re: [Xen-devel] Re: [GIT PULL] Xen APIC hooks (wit...
David Miller	Re: Xen is a feature
Greg KH	Re: Xen is a feature
David Miller	Re: Xen is a feature
Jaswinder Singh Rajput	Re: Xen is a feature
George Dunlap	Re: Xen is a feature
Jeremy Fitzhardinge	[GIT PULL] Xen APIC hooks (with io_apic_ops)
Jeremy Fitzhardinge	Re: [Xen-devel] Re: Xen is a feature
Pasi Kärkkäinen	Re: Xen is a feature
Alan Cox	Re: What Distro is best for Starting Kernel Dev?
Linus Torvalds	Re: [GIT PULL] scheduler fixes
Jonathan Corbet	Re: What Distro is best for Starting Kernel Dev?

Messages in this
thread

- *First message in
  thread*
- **Jiri Slaby**

Google Search ☐Web ☑lkml.org

Date	Sun, 31 May 2009 16:52:08 +0200
From	Jiri Slaby <>
Subject	broken ide-tape

Hi,

ide-tape is broken since commit 346331f82823682fb5c2df6e827db25d95cc2e92
(ide-tape: remove atomic test/set macros for packet commands)
from Apr 18 2008.

The commit introduced flags which are defined as (1 << nr) and uses them
in {clear,set,test}_bit. And hence it must oops on each do_request.
Is it worth fixing (something like ilog2 added to each bit operation) or
is a complete removal of the driver better "fix"?

Google Search ☐Web ☑lkml.org

From	Jiri Slaby <>
Subject	[PATCH 1/1] rapidio: fix kerneldoc for rio_mem_ops
Date	Sun, 31 May 2009 17:25:38 +0200

```
Add structure name to the kerneldoc comment, so that we won't get
a kdoc error:
Error(include/linux/rio.h:336): cannot understand prototype: 'struct rio_mem_ops '

Signed-off-by: Jiri Slaby <jirislaby@gmail.com>
---
 include/linux/rio.h |    2 +-
 1 files changed, 1 insertions(+), 1 deletions(-)
diff --git a/include/linux/rio.h b/include/linux/rio.h
index dd61538..9710506 100644
--- a/include/linux/rio.h
+++ b/include/linux/rio.h
@@ -327,7 +327,7 @@ extern struct resource rio_resource;
 #define release_rio_region(start, n) __release_region(&rio_resource, (start), (n))

 /**
- * Struct for RIO memory definition.
+ * struct rio_mem_ops - struct for RIO memory definition.
  * @map_inb: The function for mapping inbound memory window.
  * @map_outb: The function for mapping outbound memory window.
  * @unmap_inb: The function for unmapping inbound memory window.
--
1.6.3
```

Bugs + Diversität

Given a large enough beta-tester and co-developer base, almost every problem will be characterized quickly and the fix obvious to someone.
[. . .]

Bugs + Diversität

Given a large enough beta-tester and co-developer base, almost every problem will be characterized quickly and the fix obvious to someone.

[. . .]

Or less formally: „Given enough eyeballs, all bugs are shallow."

Eric S. Raymond

Vergleich

	Entwicklungsmodell	
	proprietär	offen
Entwickler-Allokation	zentral	self-selection
Motivation	extrinsisch	extrinsisch + intrinsisch
Qualitätssicherung	zentral, teuer	verteilt, billig
Wissens-Vielfalt	niedrig	hoch

Ressourcen

Ressourcen

Good programmers know what to write. Greate ones know what to rewrite (and reuse).

Eric S. Raymond

In Zahlen

- sf.net: 230.000 Projekte[1]
- Distributionen:
  - Gentoo: 1059 Libraries, 12.347 Pakete
  - Debian: 95.000 Pakete [2]
- Lines of Code (Debian) [3]

[1] http://www.sourceforge.net/softwaremap/
[2] http://popcon.debian.org/
[3] http://www.ohloh.net/p/debian/analyses/latest

Vergleich

	Entwicklungsmodell	
	proprietär	offen
Entwickler-Allokation	zentral	self-selection
Motivation	extrinsisch	extrinsisch + intrinsisch
Qualitätssicherung	zentral, teuer	verteilt, billig
Wissens-Vielfalt	niedrig	hoch
Ressourcen	teuer	billig

Innovation, closed

Innovation, closed

Company   Community
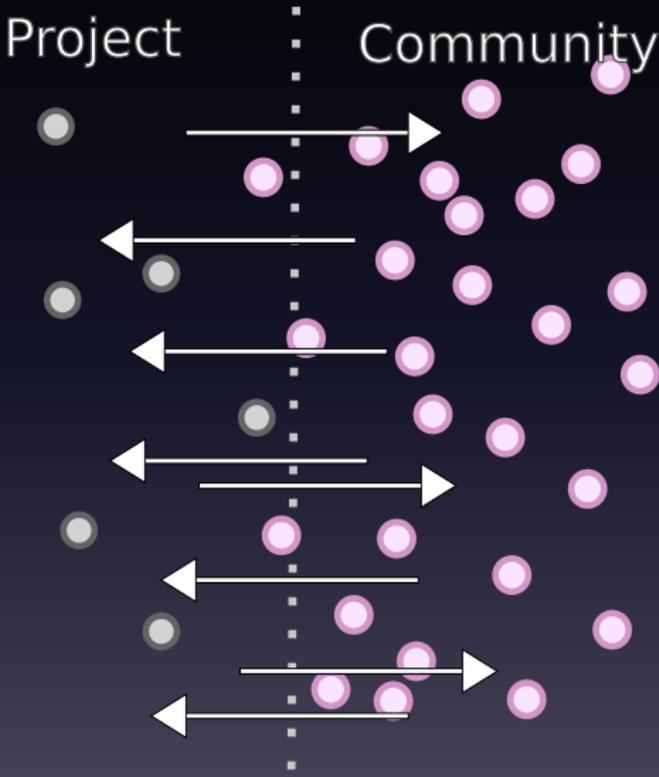
User Innovation Networks

Eric van Hippel, 2005:
„Anwender-Innovationsnetzwerke: Hersteller entbehrlich."
in: Open Source Jahrbuch 2005

- Entwicklung
- Verbreitung
- teilweise Produktion

Innovation, open

Innovation, open

Project   Community

Innovation

The next best thing to having good ideas is recognizing good ideas from your users.

Eric S. Raymond

Vergleich

	Entwicklungsmodell	
	proprietär	offen
Entwickler-Allokation	zentral	self-selection
Motivation	extrinsisch	extrinsisch + intrinsisch
Qualitätssicherung	zentral, teuer	verteilt, billig
Wissens-Vielfalt	niedrig	hoch
Ressourcen	teuer	billig
Innovation	von innen	von innen + außen

Vergleich

	Entwicklungsmodell	
	proprietär	offen
Entwickler-Allokation	zentral	self-selection
Motivation	extrinsisch	extrinsisch + intrinsisch
Qualitätssicherung	zentral, teuer	verteilt, billig
Wissens-Vielfalt	niedrig	hoch
Ressourcen	teuer	billig
Innovation	von innen	von innen + außen

LKML.ORG

Latest messages

Jiri Slaby	broken ide-tape
tom.leiming@gmail ...	[PATCH 8/8] kernel:lockdep:add statistics info for...
tom.leiming@gmail ...	[PATCH 7/8] kernel:lockdep: update memory usage in...
tom.leiming@gmail ...	[PATCH 6/8] kernel:lockdep: implement lockdep_coun...
tom.leiming@gmail ...	[PATCH 5/8] kernel:lockdep:introduce print_shortes...
tom.leiming@gmail ...	[PATCH 4/8] kernel:lockdep:implement find_usage_*w...
tom.leiming@gmail ...	[PATCH 3/8] kernel:lockdep:implement check_noncirc...
tom.leiming@gmail ...	[PATCH 2/8] kernel:lockdep: introduce match functi...
tom.leiming@gmail ...	[PATCH 1/8] kernel:lockdep:improve implementation ...
tom.leiming@gmail ...	[PATCH 0/8] kernel:lockdep:replace DFS with BFS
Alan Stern	Re: [Bug #13372] Oops in usb-serial with keyspan a...
Rik van Riel	Re: [PATCH] Use kzfree in crypto API context initi...
Rik van Riel	Re: [PATCH] Use kzfree in crypto API context initi...
Arjan van de Ven	Re: [patch 0/5] Support for sanitization flag i...

Hottest messages

Eric Paris	Introducing SELinux sandbox, confining untrusted b...
Jeremy Fitzhardinge	Xen is a feature
Linus Torvalds	Linux 2.6.30-rc7
Ingo Molnar	Re: [Xen-devel] Re: [GIT PULL] Xen APIC hooks (wit...
David Miller	Re: Xen is a feature
Greg KH	Re: Xen is a feature
David Miller	Re: Xen is a feature
Jaswinder Singh Rajput	Re: Xen is a feature
George Dunlap	Re: Xen is a feature
Jeremy Fitzhardinge	[GIT PULL] Xen APIC hooks (with io_apic_ops)
Jeremy Fitzhardinge	Re: [Xen-devel] Re: Xen is a feature
Pasi Kärkkäinen	Re: Xen is a feature
Alan Cox	Re: What Distro is best for Starting Kernel Dev?
Linus Torvalds	Re: [GIT PULL] scheduler fixes
Jonathan Corbet	Re: What Distro is best for Starting Kernel Dev?

Motivationen

Informationsproduktion

Problem + Mensch + Ressourcen = Lösung

Informationsproduktion

Problem + Mensch + Ressourcen = Lösung
Problem = Informationsproduktion

Informationsproduktion

Problem + Mensch + Ressourcen = Lösung
Problem = Informationsproduktion

Information + Mensch = neue Information

Coase

Coase

Markt vs. Firma[4]

[4] Ronald H. Coase: „The Nature of the Firm"

Coase

Markt vs. Firma[4]

Marktkosten vs. Hierarchiekosten

---

[4] Ronald H. Coase: „The Nature of the Firm"

Pinguin-Modell

Pinguin-Modell

1. zerteile Problem in kleine Einheiten

Pinguin-Modell

1. zerteile Problem in kleine Einheiten

2. lasse das Problem von verschiedenen Teilnehmern lösen

Pinguin-Modell

1. zerteile Problem in kleine Einheiten

2. lasse das Problem von verschiedenen Teilnehmern lösen

Problem: Marktkosten
(Transaktionskosten = Verträge, Suchkosten, ...)

Pinguin-Modell

1. zerteile Problem in kleine Einheiten

2. lasse das Problem von verschiedenen Teilnehmern lösen

Problem: Marktkosten
(Transaktionskosten = Verträge, Suchkosten, . . .)

Funktioniert nicht!

Coase's Penguin

Coase's Penguin

Internet

Coase's Penguin

Internet
(niedrige Suchkosten)

Coase's Penguin

Internet
(niedrige Suchkosten)

Eigentum → Commons

Coase's Penguin

Internet
(niedrige Suchkosten)

Eigentum → Commons
(Überfluss an Ressourcen)

Coase's Penguin

Internet
(niedrige Suchkosten)

Eigentum $\rightarrow$ Commons
(Überfluss an Ressourcen)

Verträge $\rightarrow$ „Social Transactions"

Coase's Penguin

Internet
(niedrige Suchkosten)

Eigentum → Commons
(Überfluss an Ressourcen)

Verträge → „Social Transactions"
(niedrige Transaktionskosten)

Coase's Penguin fliegt

Coase's Penguin fliegt

„commons-based peer production"[5]

[5] Benkler, „Coases Penguin, or, Linux and the Nature of the Firm"

Coase's Penguin fliegt

„commons-based peer production"[5]

Ein dritter Produktionsmodus?

---

[5] Benkler, „Coases Penguin, or, Linux and the Nature of the Firm"

It may well turn out that one of the most important effects of open source's success will be to teach us that play is the most economically efficient mode of creative work.

Eric S. Raymond

Motivationen

Motivationen

Motivationen

Motivationen

- Altruismus
- Reputation
- Komplementärgüter
- Geschäftsmodelle
  - Dual Licensing
  - Support
  - Integration
  - . . .
- Property Preempting Investment[6]

---

[6] R. Merges: „A New Dynamism in the Public Domain"

Done?

Gefahren für freie Software

GPL v3.0

Unsaved spreadsheet

File   Edit   View   Format   Insert   Tools   Form   Help
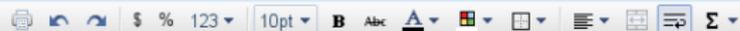
	A	B	C	D	E	F	G	H
1								
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								
16								
17								
18								
19								
20								
21								
22								
23								
24								
25								
26								
27								
28								
29								
30								
31								
32								
33								

Add Sheet   Sheet1 ▼

GNU Affero GPL

Bildquellen

Vielen Dank an: