

# Software development in embedded Linux - informal collaboration of competing firms

**Joachim Henkel**

Ludwig-Maximilians-Universität München

Institut für Innovationsforschung und  
Technologiemanagement

**TIM 2003**

München, 24.10.2003



## 1. Einführung

2. Untersuchung: embedded Linux
3. Freigabe von Entwicklungen
4. Informelle Entwicklungskooperation
5. Motive
6. Interpretation

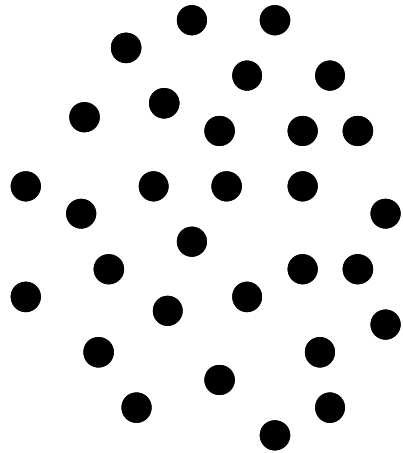
# Einführung – Open-Source-Software

---

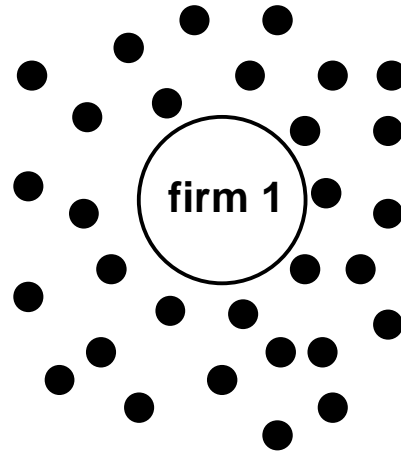
## Open-Source-Software (OSS)

- Beispiele: Linux, Apache, Sendmail, Jabber, ...
- Eigenschaften:
  - Quellcode muss Nutzern verfügbar gemacht werden
  - Lizenzgebühren nicht möglich
  - Jeder darf Veränderungen vornehmen
  - Jeder darf Software weiterverbreiten
- Bedeutung der Beiträge von Unternehmen stark zunehmend
- Beispiele: IBM, Sun, Hewlett Packard, ...

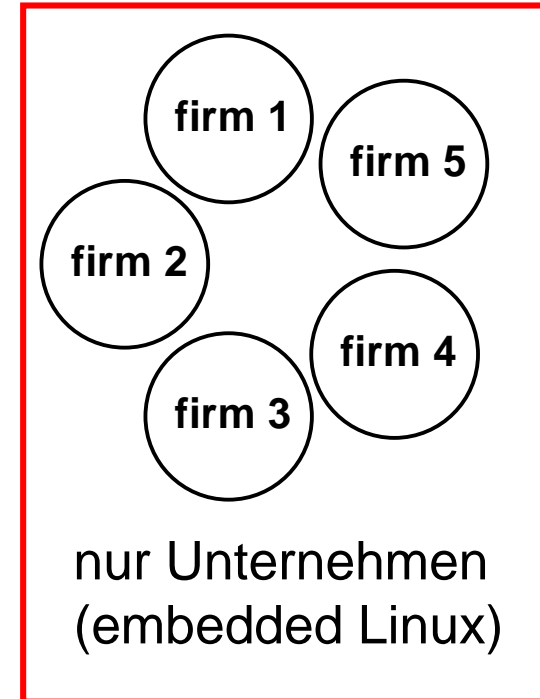
# Einführung – Typen von OSS-Projekten



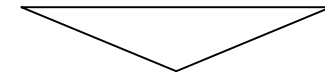
nur Community  
(Linux anfangs)



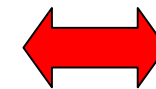
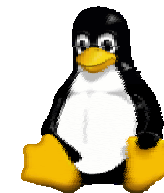
Unternehmen u. Community  
(Linux heute)



nur Unternehmen  
(embedded Linux)



**Widerspruch?**



# Einführung – Motivation

---

## Open-Source-Software:

- Schwache Schutzmöglichkeiten
  - Quellcode offen
  - Veränderung erlaubt
- Eingeschränkte Aneignungsmöglichkeiten
  - Lizenzgebühren ausgeschlossen
  - Weitergabe erlaubt



**Innovationsanreize für Unternehmen?**

**Entwicklungsgeschwindigkeit insgesamt?**

**Individuelles Verhalten: maximale Geheimhaltung?**

# Übersicht

---

1. Einführung

## **2. Untersuchung: Embedded Linux**

3. Freigabe von Entwicklungen

4. Informelle Entwicklungskooperation

5. Motive

6. Interpretation

# Embedded Linux – Schlagzeile

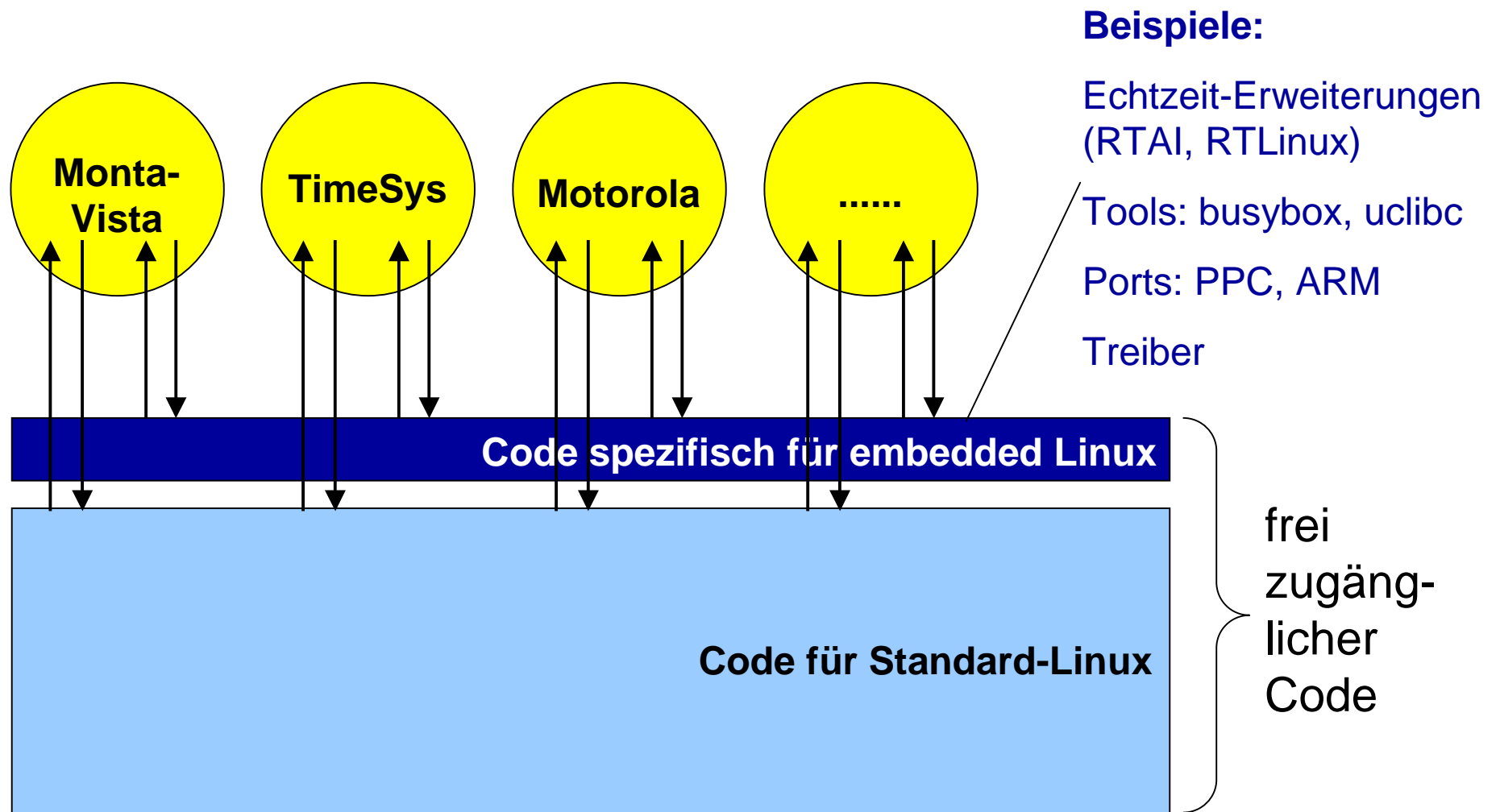
---

## **„Electronics Giants Love Linux“**

Gründung des „CELF“ (Consumer Electronics Linux Forum) durch Matsushita, Sony, Hitachi, IBM, NEC, Philips, Samsung, Sharp und Toshiba.

Forbes.com, 2. Juli 2003

# Embedded Linux / Standard Linux



# Embedded Linux – Forschungsfragen

---

Entwicklungen, die **spezifisch für embedded Linux** sind:  
fast **ausschließlich von Unternehmen**



- **Open-Source-Entwicklungsprozess möglich?**
- **Freigabe von Entwicklungen? Warum? Was? Was nicht?**
- **Entwicklungskooperation?**
- **Innovation trotz schwacher Schutzmöglichkeiten?**

# Embedded Linux – Methodik

---



**Interviews mit Experten**

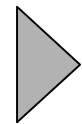


**Web-basierter Fragebogen an Experten**

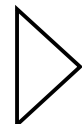
heute



**Spieltheoretische Modellanalyse**



**Analyse von Mailing-Listen**



**Großzahlige Befragung von Entwicklern**

# Embedded Linux – Interviewpartner

Embed. Linux SW-Unternehmen		Geräte-hersteller	andere emb.-SW-Unternehmen	Branchen-experten
USA	Europa			
CodePoet FSMLabs Independent developer Lineo LynuxWorks MontaVista TimeSys	Denx emlix Mind Pengutronix reLinux Sysgo	Convergence Innominate Maintainer, RTAI Siemens CT Siemens ICN SSV Embedded Systems	Microsoft QNX SleepyCat Wind River	Autor “Embedded Linux” CodeWeavers Intevation FSF LinuxDevices LinuxJournal Maintainer, Debian Linux

# Übersicht

---

1. Einführung
2. Untersuchung: embedded Linux
- 3. Freigabe von  
Entwicklungen**
4. Informelle Entwicklungskooperation
5. Motive
6. Interpretation

# Freigabe – verfügbarer Code

---

*Code für embedded Linux ist in großem Umfang öffentlich verfügbar.*

*Meine Erfahrung ist, dass sehr viel öffentlich zugänglich ist, dass es gerade im embedded Bereich [...] einige sehr wichtige und [...] gut funktionierende Projekte gibt.*

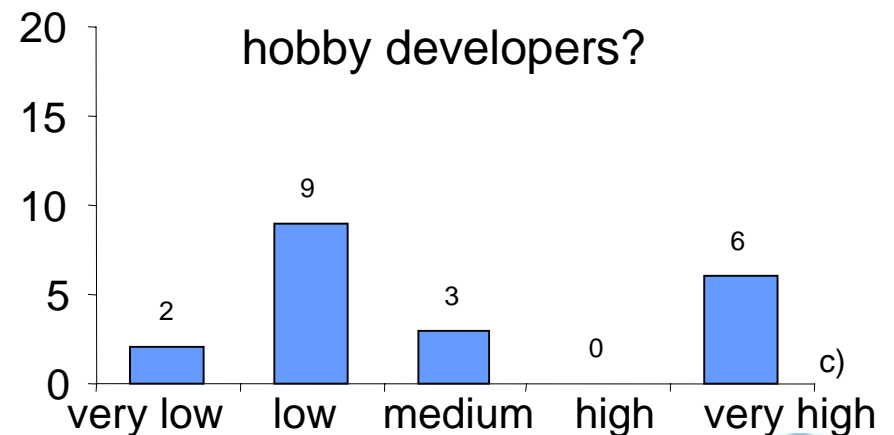
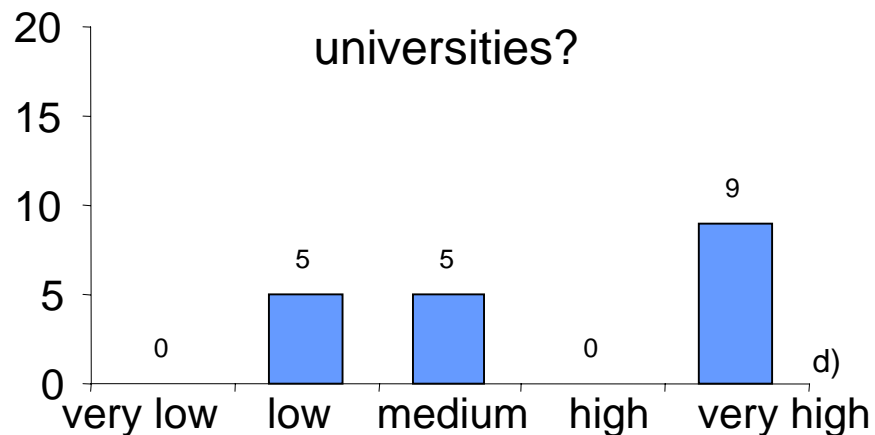
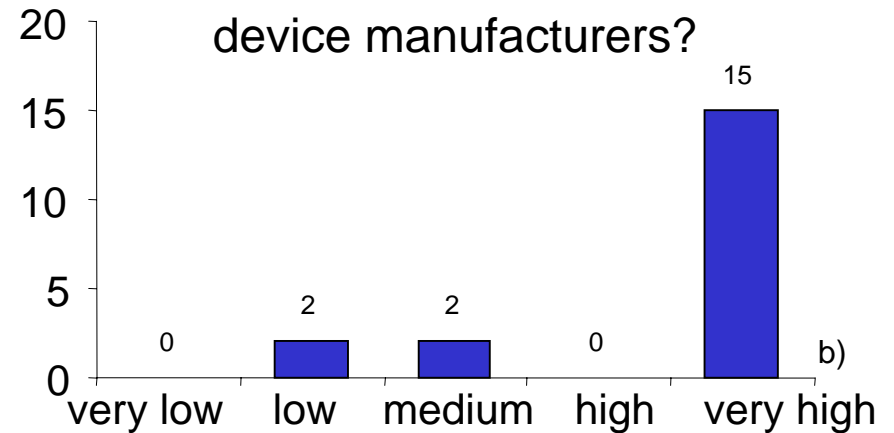
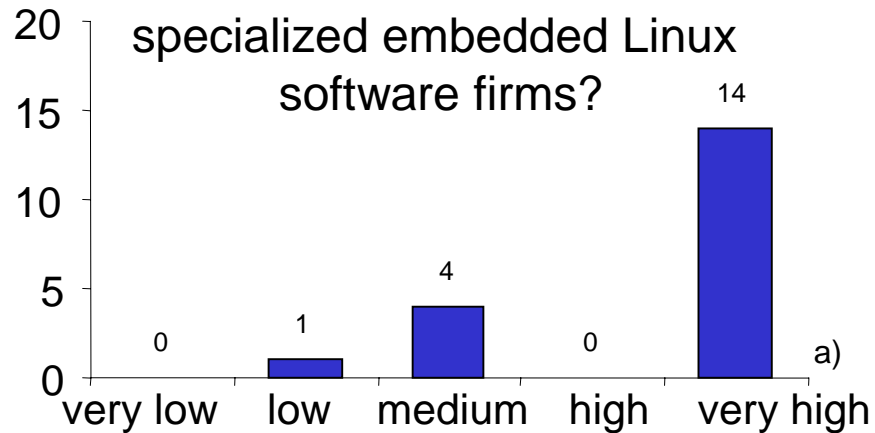
(Embedded-Linux-Unternehmen)

*Was sich bei uns dramatisch geändert hat, ist, dass es seit 2000 viele Lösungen speziell für den embedded-Markt gibt, Module.*

(Gerätehersteller)

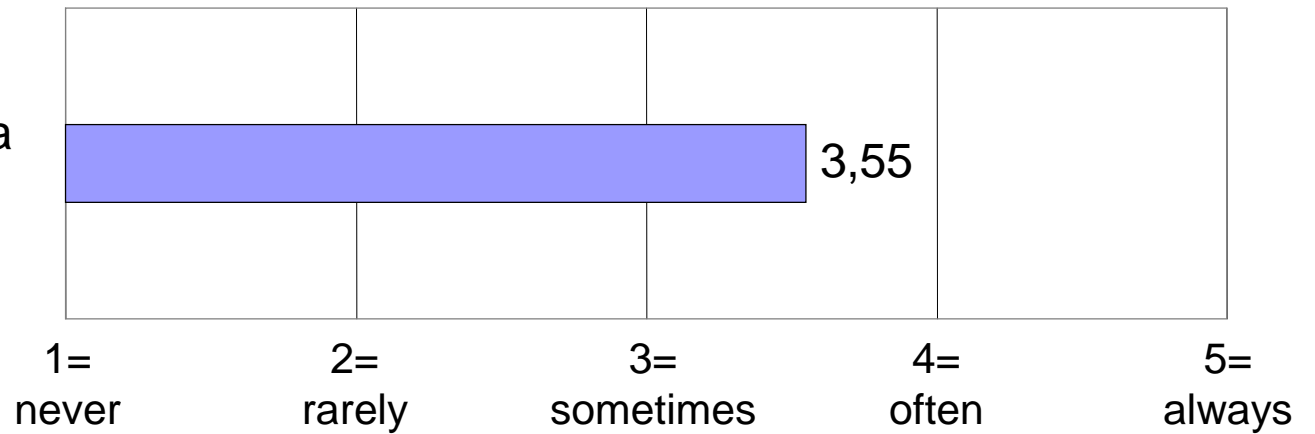
# Freigabe von Code – Quellen

„How important for the development of embedded Linux are contributions from ...“

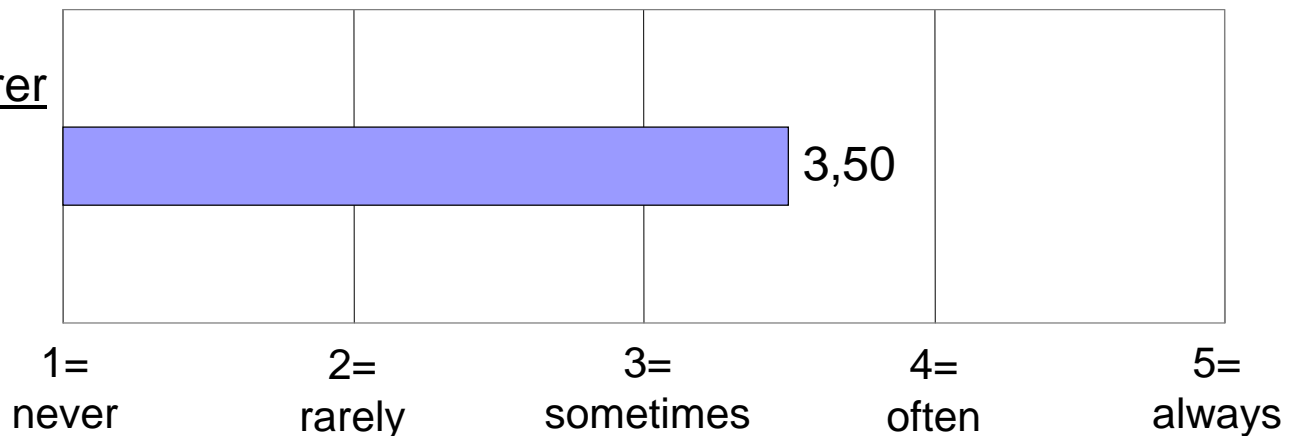


# Freigabe von Code – Bsp.: Treiber

**C2.** Think of an embedded Linux software firm that has developed a new driver. Would the firm make this driver public?

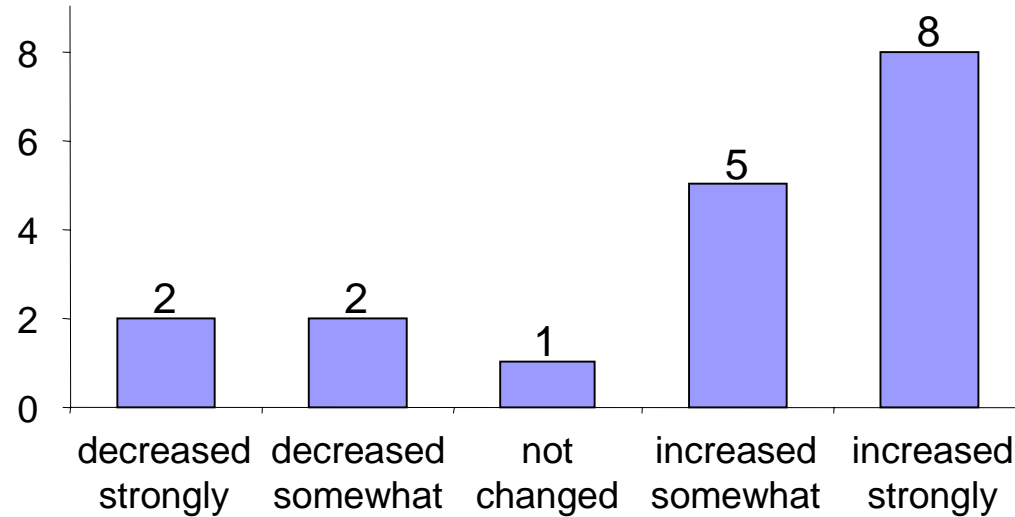


**D2.** Think of a device manufacturer that has developed a new driver. Would the firm make this driver public?

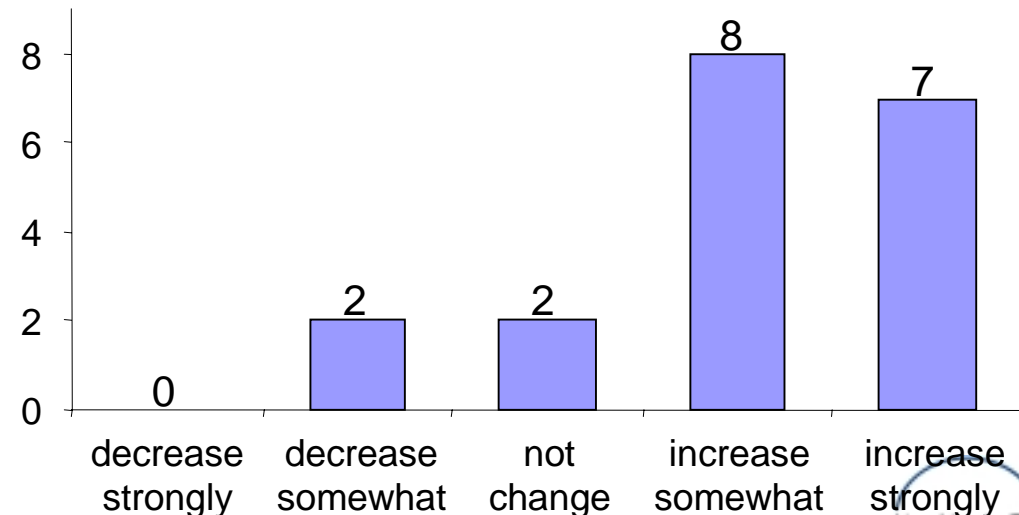


# Freigabe von Code – zunehmend

**Since 2000,**  
revealing of  
embedded Linux  
code has ...



**In the next three years,**  
revealing of embedded  
Linux code will ...



# Übersicht

---

1. Einführung

2. Freigabe von Entwicklungen

## **3. Informelle Entwicklungskooperation**

4. Motive

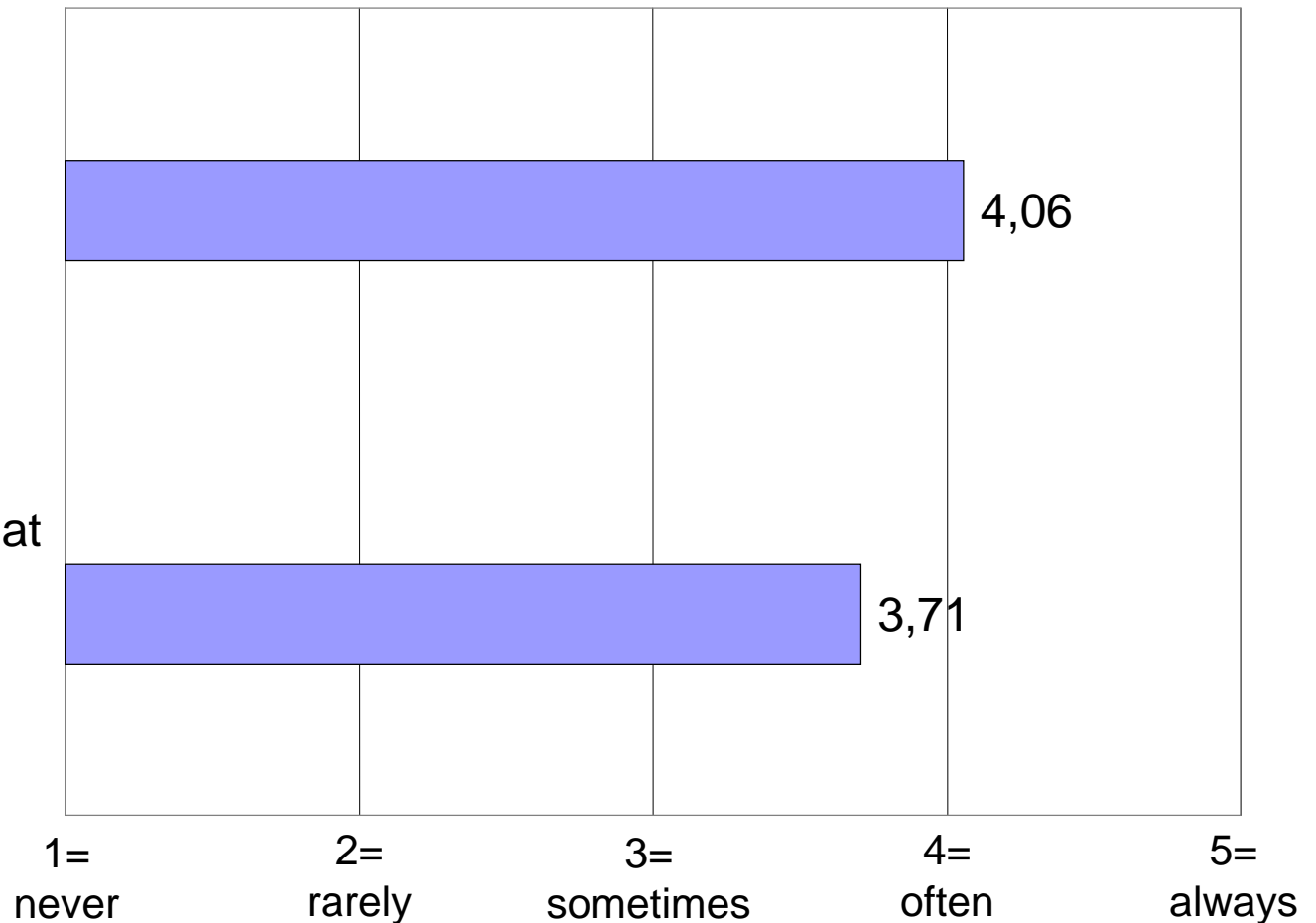
5. Interpretation

# Entwicklungs Kooperation

*Informelle Entwicklungszusammenarbeit erfolgt oft.*

**F1.** Assume that a new patch is submitted to a mailing list. How often does it happen that the merits and drawbacks of such a development are **openly discussed** on the mailing list?

**F2.** How often does it happen that developers from different firms contribute to and **work on the same patch**?



# Entwicklungskooperation

---

*"...in general there are, on the public lists,  
different competing companies  
contributing to improve the public version.  
That's very obvious."*

(Embedded-Linux-Unternehmen, EU)

# Übersicht

---

1. Einführung
2. Untersuchung: embedded Linux
3. Freigabe von Entwicklungen
4. Informelle Entwicklungskooperation

## **5. Motive**

6. Interpretation

# Motive für Freigabe

## C3. Embedded Linux software firms reveal code because ...

others will improve the code and reveal their improvements in turn.

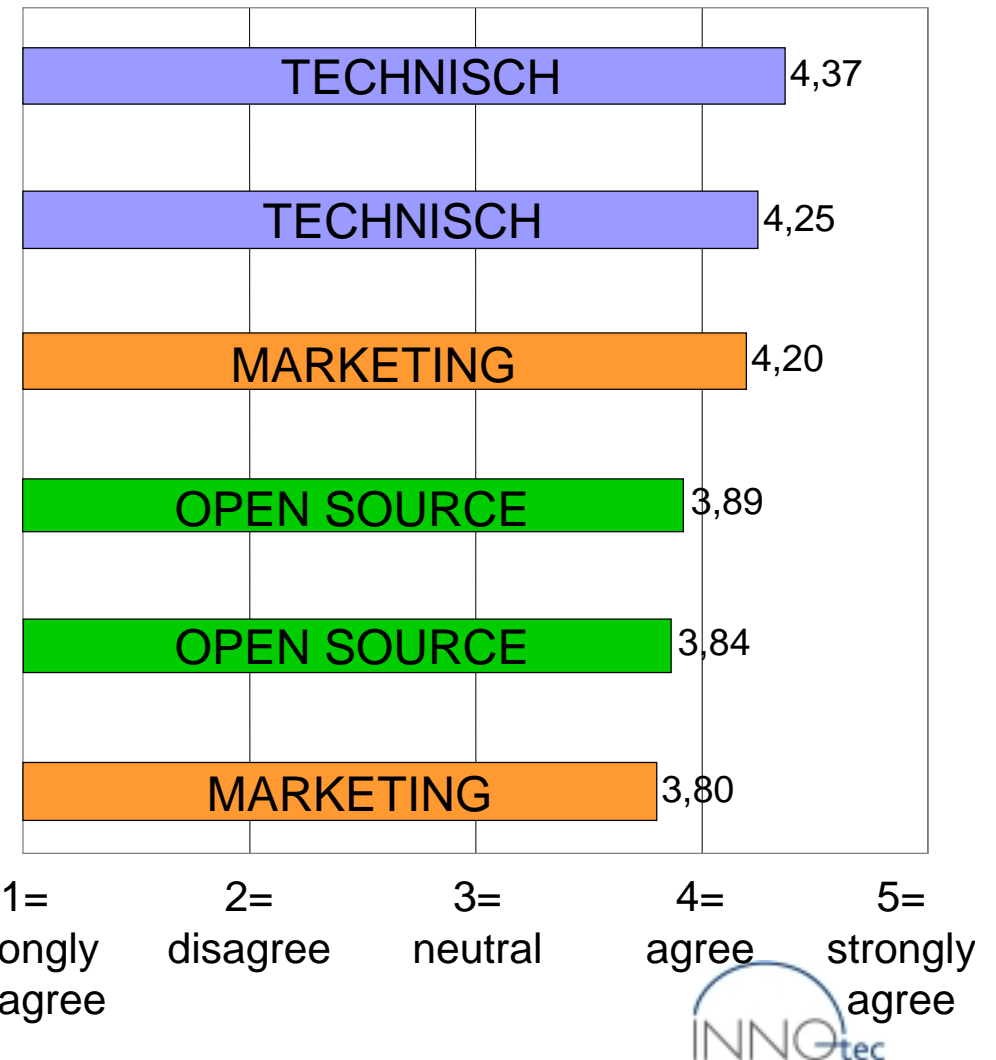
it reduces maintenance effort when the code becomes part of mainstream Linux.

revealing good code improves the respective firm's technical reputation.

developers consider it fair to give back to the community, since the firm benefits from publicly available code.

the GPL requires it.

visibility on the mailing lists is good marketing.



# Motive für Freigabe – Entwicklungsunterstützung

---

*... wir es gerne zurückgeben, um es ... extern noch mal validieren zu lassen. Was andere Leute sagen, das ist definitiv relevant, denn heute durchblickt keiner mehr den Linux-Kernel komplett von vorn nach hinten.  
(Embedded-Linux-Unternehmen)*

*I know that our version of Linux is going to be extensively tested with both MySQL and with Oracle. That would be s.th. far beyond our ability ...;  
it would be an enormous financial burden for a company like QNX.  
But I get it for nothing, or for my participation in Linux.  
(Embedded-Linux-Unternehmen)*

# Motive, nicht freizugeben

---

*Unreflektierte Grundhaltung: „Wir geben nichts frei.“*

*Die Kunden sagen, wenn sie für eine Software-Lösung bezahlt haben, dann wollen die das auch für sich alleine haben.*

*(Europäisches Embedded-Linux-Unternehmen)*

*... proprietäre Erweiterungen, ... da möchte der Kunde sich nicht reinblicken lassen. Das ist in vielen Bereichen aus unserer Sicht ...*

*Paranoia ... (Europäisches Embedded-Linux-Unternehmen)*

# Übersicht

---

1. Einführung
2. Untersuchung: embedded Linux
3. Freigabe von Entwicklungen
4. Informelle Entwicklungskooperation
5. Motive

## **6. Interpretation**

# Interpretation – Generische Entwicklungen

---

*They will allow us to release the source code for generic parts and they will ask us not to publish the source code for more specific parts. (Embedded-Linux-Unternehmen, EU)*

*Wir konkurrieren ... mit unseren Produkten, nicht mit unserem Betriebssystem. Das ist nicht Gegenstand der Konkurrenz. (Gerätehersteller)*

*Nobody is going to have a competitive advantage in a TCP/IP stack. (Embedded-Linux-Unternehmen, US)*

# Interpretation – Systematik

		SOFTWARE	
		Grad der Differenzierung <i>hoch</i>	<i>niedrig</i>
HARD- WARE	<i>geschützt</i>	<p>„Idiosynkratisch“ System sehr spezifisch, Software uninteressant für andere</p> <p>→ <b>keine Freigabe</b></p> <p>1</p>	<p>„Schutz durch HW“ Software allgemein verwendbar, Schutz durch Hardware</p> <p>→ <b>Freigabe</b></p> <p>2</p>
	<i>nicht geschützt</i>	<p>„IP in Software“ Verlust von Wettbewerbsvorteilen zu erwarten</p> <p>→ <b>keine Freigabe</b></p> <p>3</p>	<p>„Generisch“ Schutz von Wettbewerbsvorteilen aus anderen Quellen</p> <p>→ <b>Freigabe</b></p> <p>4</p>

# Interpretation – Hypothesen

---

## Freigabe wird begünstigt, wenn ...

H1. ... Entwicklungen allgemein verwendbar („generisch“) sind

H2. ... Entwicklungen nur wenig zur Differenzierung beitragen

H3. ... Schutz durch komplementäre Assets besteht

H4. ... Inkremente klein sind

H5. ... heterogener Bedarf nach komplementären Technologien besteht

H6. ... Unternehmen relativ klein sind

H7. ... Entwickler positive Einstellung zu OSS haben

H8. ... Intensität des Wettbewerbs gering ist

# Zusammenfassung, Ausblick

---

- Informelle Entwicklungskooperation nachzuweisen
- Balance zwischen Freigabe und Geheimhaltung
- „Jukebox style of innovation“
  
- Fortschritt in embedded Linux:  
trotz / *wegen* geringer Schutzmöglichkeiten sehr schnell
- Freie Weitergabe als Aneignungsmechanismus
- Grundhaltung „Wir geben nichts frei“ vernachlässigt Gewinnpotentiale

Vielen Dank!