Munich/MIT Survey:

Development of Embedded Linux

May 10, 2004

Joachim Henkel, Mark Tins

Institute for Innovation Research, Technology Management and Entrepreneurship

University of Munich

Contact: joachimh@mit.edu, mark.tins@campus.lmu.de

Address: Kaulbachstr. 45, D - 81929 Munich, Germany.

www: http://www.inno-tec.bwl.uni-muenchen.de/mitarbeiter/henkel/index_e.html

1. Introduction

The use of Linux in embedded devices has increased enormously in recent years. Software vendors offer Linux distributions for embedded systems, device manufacturers employ Linux in their devices, and component manufacturers increasingly make drivers for Linux available.

Most of the publicly available code for embedded Linux is developed by commercial firms, not by hobbyists. While it is true that also for standard Linux many contributions come from IBM and other large firms, the situations differ. IBM pursues the strategic goal of establishing Linux as a widely used operating system. In contrast, embedded Linux is an integral part of their products for hardware manufacturers, and the core market offering for specialized software firms.

This raises the question if and how the development process of embedded Linux differs from that of other open source software. After all, companies working with embedded Linux have a legitimate interest in protecting their competitive advantage when the code helps to differentiate their market offering. How can this interest be reconciled with the open source culture and the requirements of the GPL? And if firms voluntarily make code public – what are the benefits they derive from doing this, and how important are these?

The present survey aims at understanding these issues. Among others, it addresses the following questions: Who contributes to publicly available code for embedded Linux? To what extent are different types of companies involved (software firms, device manufacturers, component manufacturers)? How important are contributions from hobbyists? Do companies make some of their developments related to embedded Linux public? How much? What types of developments? What are the reasons to make code public or to keep it secret? What means are used to keep important code secret? Is there a company policy in place regarding the publication of code? How often do developers interact with one another in public open source projects? In what way? What impact has a developer's personal attitude towards open source on his behavior?

We are aware of the fact that no single well-defined "embedded Linux" exists. When we speak, somewhat loosely, of the "development of embedded Linux", we mean the development of parts or versions of Linux that are primarily relevant for embedded applications. Examples are real-time extensions such as RTAI and RTLinux, BusyBox, uclibc, ports to processors such as ARM and PPC, and customized versions of Linux for embedded devices.

A comment concerning the selection of participants is in place. We posted a link to the questionnaire on several websites and mailing lists. Participants are thus self-selected, which is hard to avoid in a survey such as this one. However, it means that some results have to be interpreted with care. For example, it is plausible that hobbyists working on embedded Linux feel less time pressure than employed programmers. Hence, they are probably more inclined to participate in the survey and will likely be over-represented in the sample. Similar considerations apply to other fields.

The survey was online from November 18, 2003, to March 8, 2004. We received a total of 268 valid responses. The present paper is a collection of descriptive results. A thorough evaluation containing interpretation and multivariate analysis is in preparation and will be available, in the next months, on http://opensource.mit.edu and on the website of one of the authors (see first page).

This survey was initiated while one of the authors was a visiting scholar at MIT's Sloan School of Management. It benefited greatly from discussions with Eric von Hippel and Karim Lakhani. We would also like to thank those developers who commented on earlier versions of the questionnaire, and all participants for taking the time to fill it out. Finally, many thanks to those who posted a link to the survey on their websites or mailing lists, in particular to Rick Lehrbaum and Henry Kingman at LinuxDevices.com.

2. Demographics of participants

Gender

Male: 259; Female: 5; Missing data: 4

Age

16-25 years:	15.6 %
26-35 years:	44.1 %
36-45 years:	24.0 %
46-55 years:	14.8 %
56- older:	1.5 %

Regions

North America	111	42.4 %
Europe	104	39.5 %
Asia	26	9.9 %
Australia	12	4.6 %
South America	9	3.4 %
Africa	1	0.4 %
Missing data	5	1.9 %

Countries

Participants work in 39 different countries:

USA	96	Czech Republic	4
Germany	28	Danmark	4
Canada	15	New Zealand	4
United Kingdom	12	South Korea	4
India	9	Netherlands	3
Australia	8	Singapore	3
Italy	8	Spain	3
France	7	Taiwan	3
Brasil	6	Norway	2
China	6	Switzerland	2
Sweden	6	Other	15
Austria	5	Missing data	5
Belgium	5		
Finland	5		

Current occupation

Professional programmer	166	61.9%
IT-manager	20	7.5%
Student	21	7.8%
Engineer	21	7.8%
Full-time faculty	15	5.6%
Systems administrator	5	1.9%
Consultant	4	1.5%
Other	11	4.1%
Missing data	5	1.9%

Websites where participants found out about the survey

LinuxDevices.com	138	51.5 %
PPC embedded mailing list	28	10.4 %
RTAI mailing list	26	9.7 %
Handhelds.org	25	9.3 %
Busybox mailing list	11	4.1 %
Link from colleague/supervisor	6	2.2 %
Other mailing lists	6	2.2 %
Other	12	4.5 %
Missing data	16	6.0 %

Experience as software developer

Years developing…	any kind of Software	Open Source Software	software for embedded systems	embedded Linux
Mean	14.2	4.9	7.1	2.5
Median	14.0	4.0	5.0	2.0
Standard dev.	8.3	3.7	6.5	1.7
Minimum	1.0	0.2	0.1	0.1
Maximum	35.0	20.0	30.0	10.0
Missing data	13	49	16	35
N	255	219	252	233

Time spent on development of embedded Linux

hours per week	during working time	during leisure time
Mean	25.3	8.2
Median	20.0	5.0
Standard dev.	18.2	7.4
Minimum	1.0	1.0
Maximum	100.0	45.0
Missing data	50	93
N	218	175

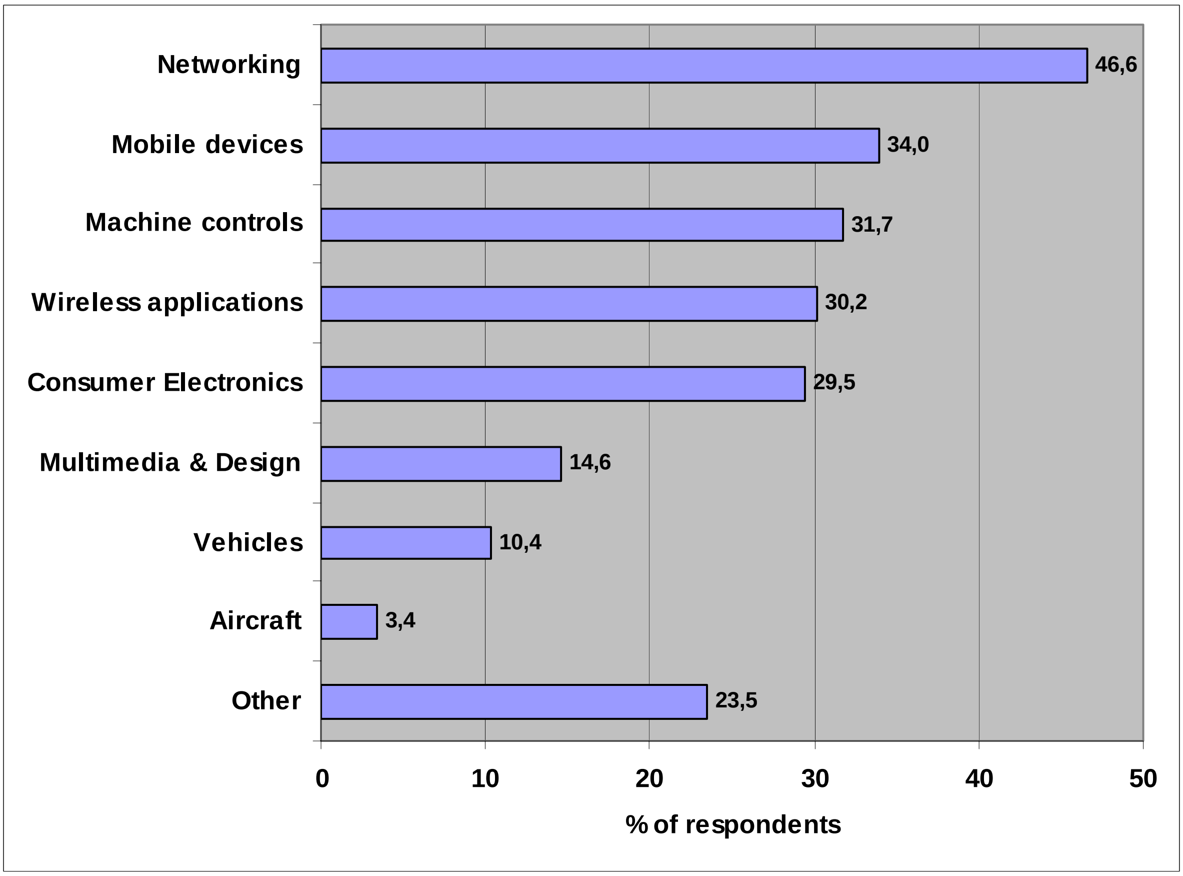
30 percent of respondents work 38 hours or more per week in their working time on embedded Linux.

3. Description of companies that participants work for

Type of organization participants work for

Device manufacturer	42.5%
Software company specializing on embedded Linux	22.4%
Working as a hobbyist	15.3%
University or other non-profit research organization	11.2%
Manufacturer of components like chips and boards	8.6%

Fields of application for which participants develop embedded Linux



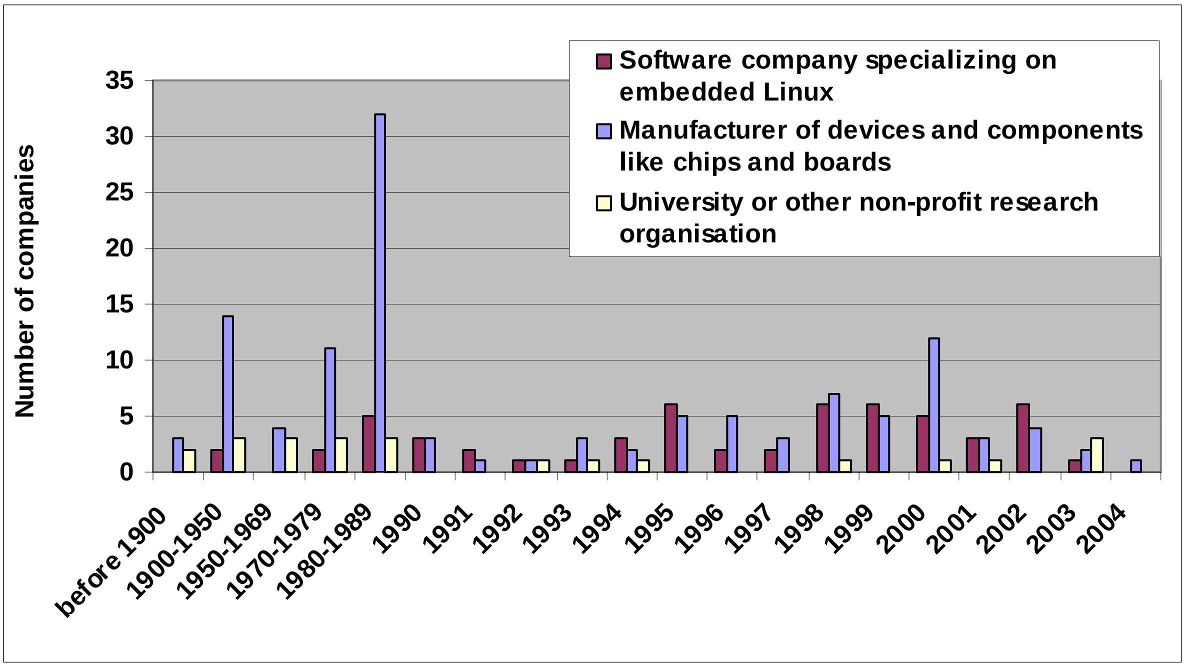
Other fields of application mentioned:

- Measurement
- Security/Safety
- Robotics
- Military Devices
- Medical Devices
- Education
- Data Collection
- Consulting
- Telemechanics
- Space
- Navigation
- Diagnostic
- Simulation
- Scientific Research
- Radio Frequency Identification
- Transponders
- Power Generation and Distribution
- Large LED Signs
- Heating, Ventilating, Air-Conditioning

Size of company

How many people work for your company?	
	Frequency
just me	25
2-10	54
10-50	60
50-200	39
over 200	81
Missing data	9
N	259

Year of foundation



Year when company started developing embedded Linux

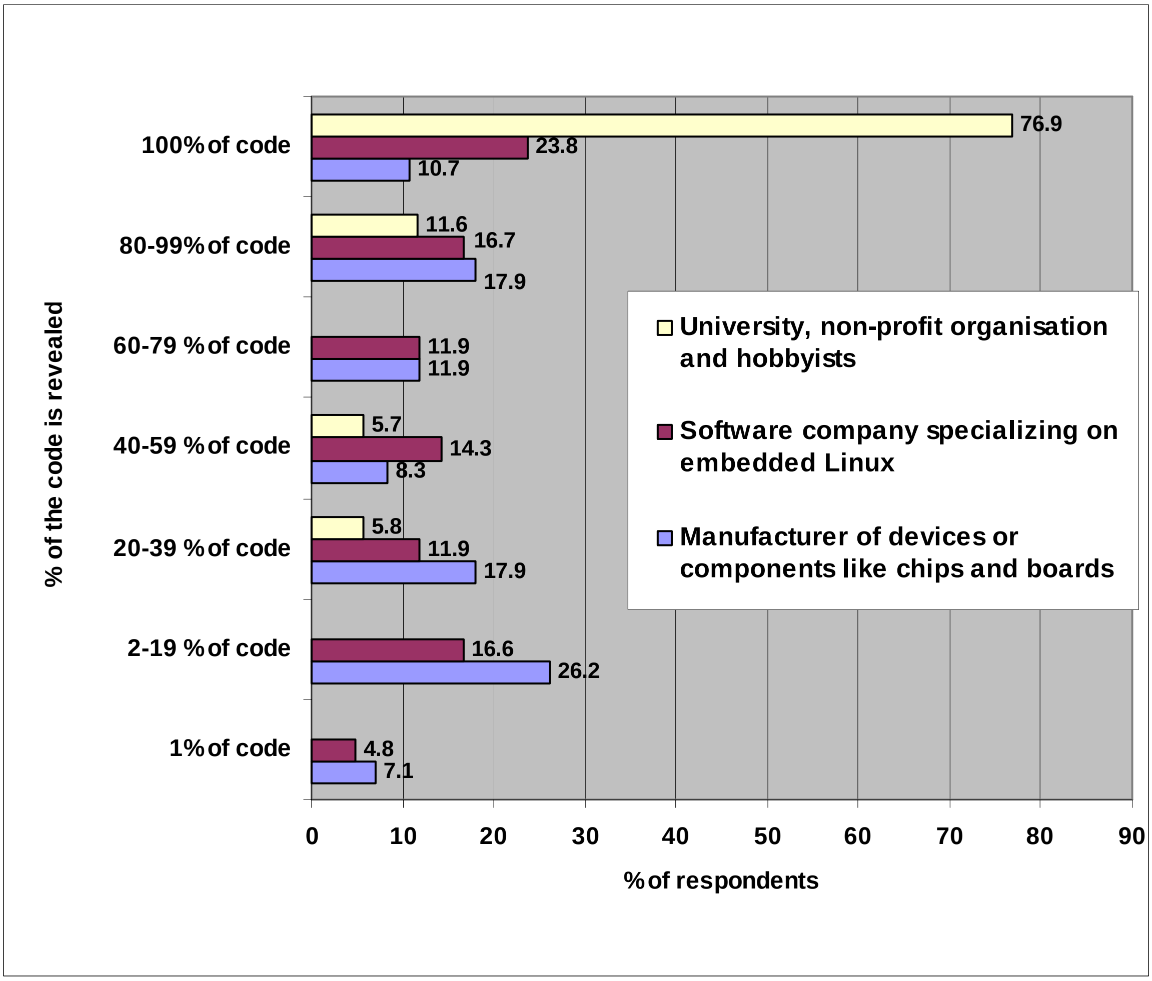
Mean	2000.4
Median	2001
Standard dev.	2.3
Minimum	1985
Maximum	2004
Missing data	42
N	226

4. Revealing Code

Share of revealed code

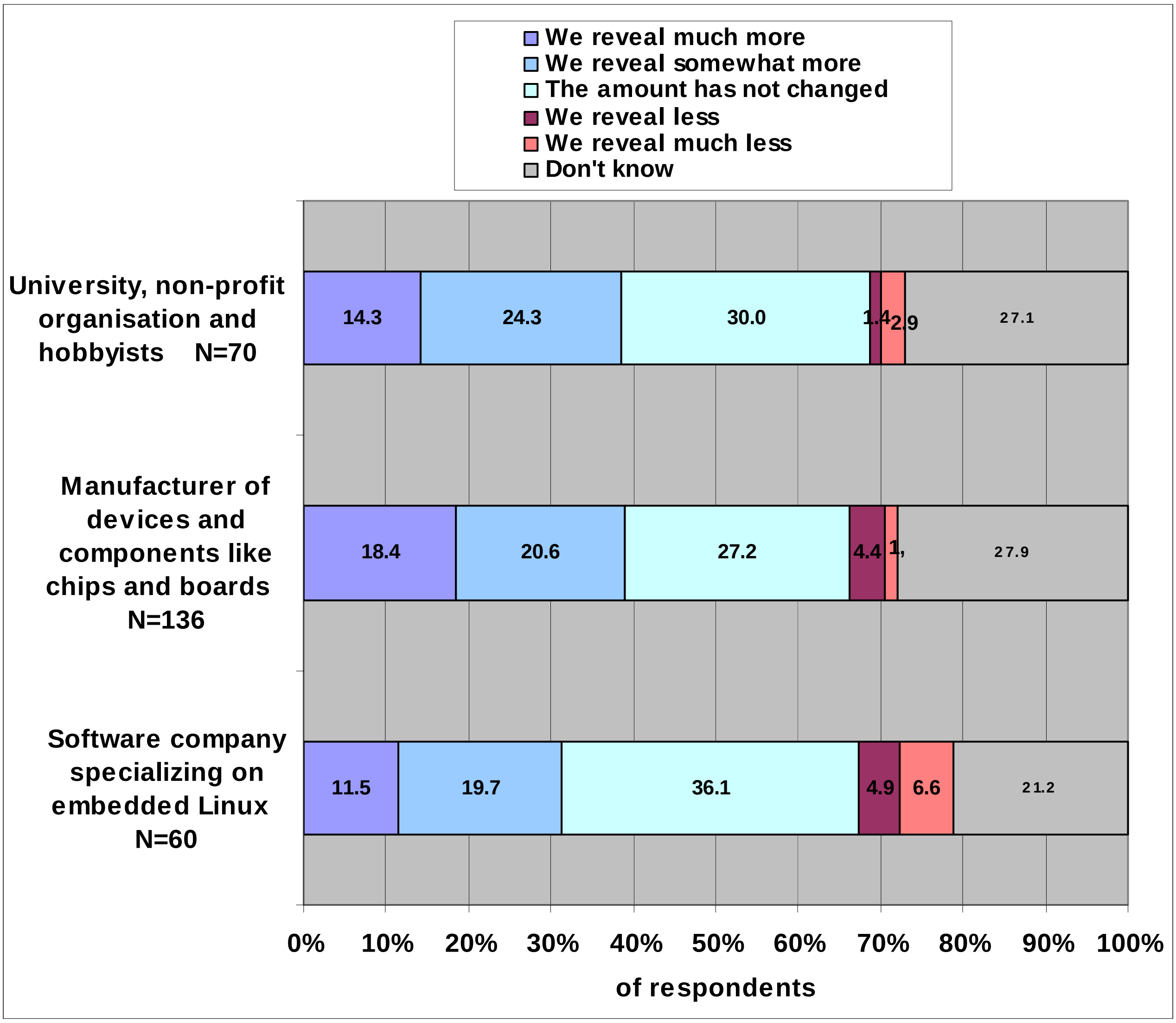
Question: “Please consider those *embedded Linux developments by your firm* that are potentially *useful for others*. That is, they are *not too specific to your firm*, and others would benefit (use them, develop them further) if they were made public. What share of this code is freely revealed?”

Approx % of code	Software companies	Hardware manufacturers	Universities, hobbyists
Mean	57.5	45.5	92.0
Median	60.0	31.5	100.0
Standard dev.	35.9	37.5	20.1
Minimum	1	1	25
Maximum	100	100	100
Missing data	21	51	20
N	39	86	51



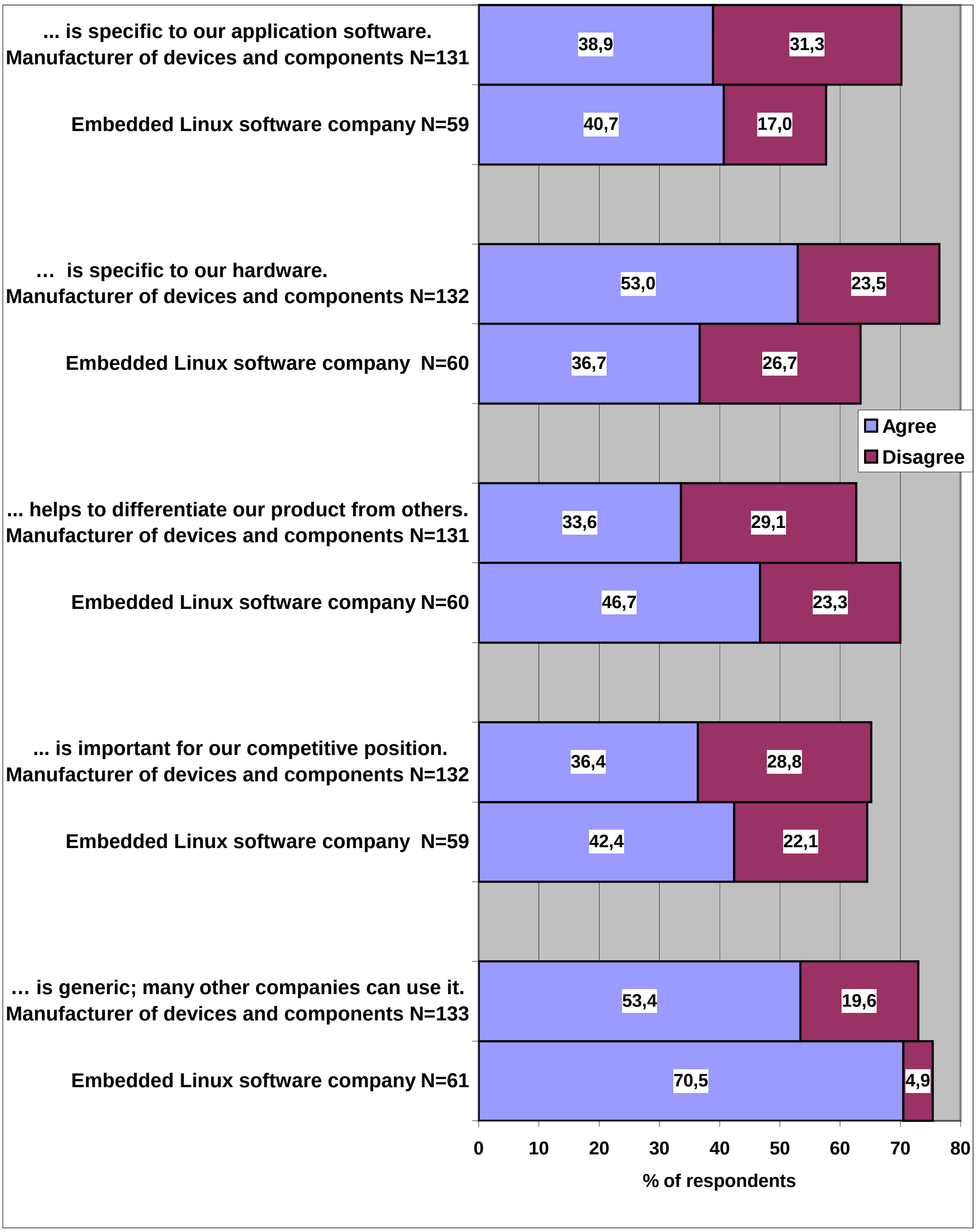
Change of revealing over time

Question: “Has the total amount of code for embedded Linux that your company reveals (per year) changed since 2000?”



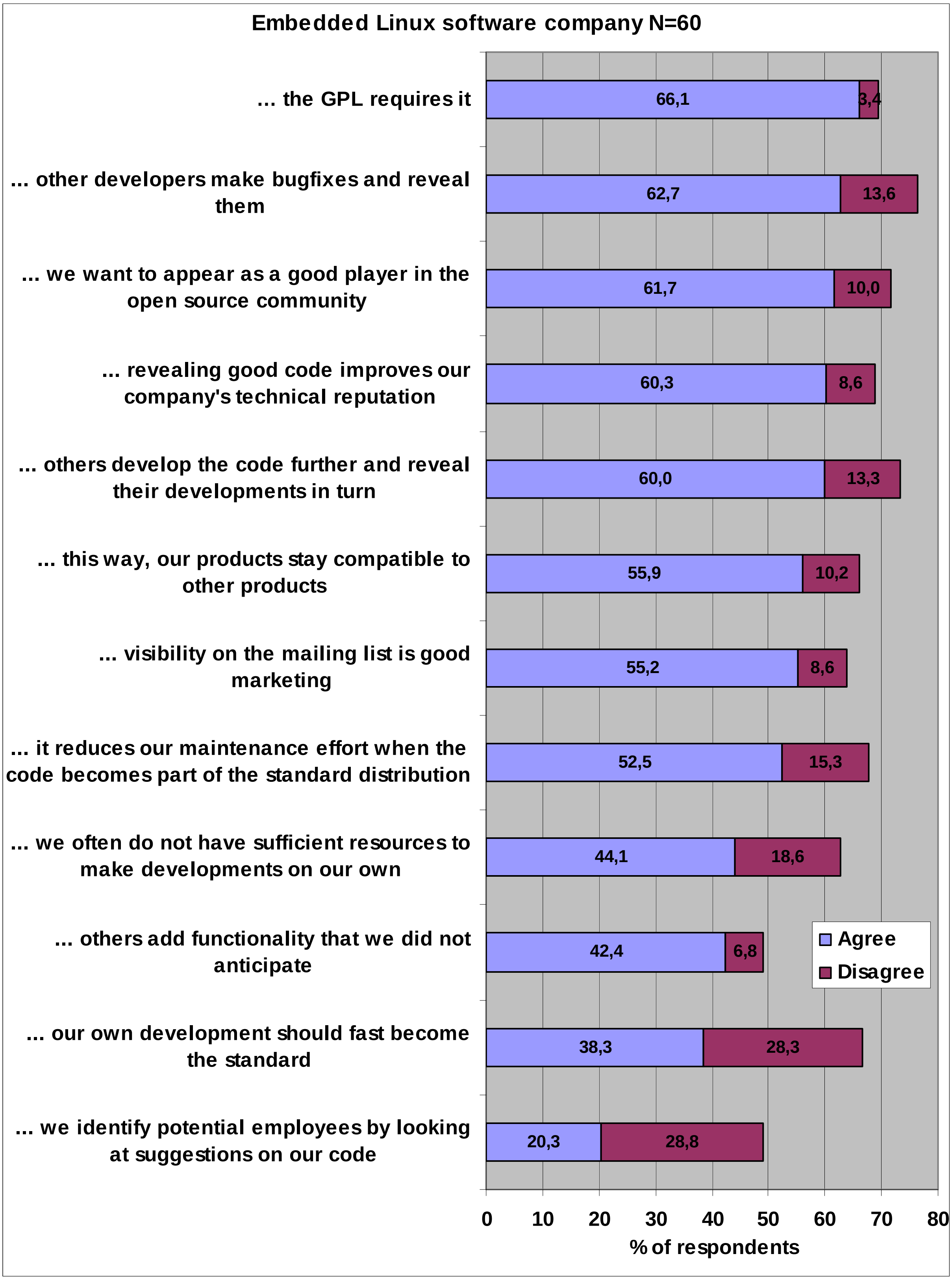
Description of revealed code

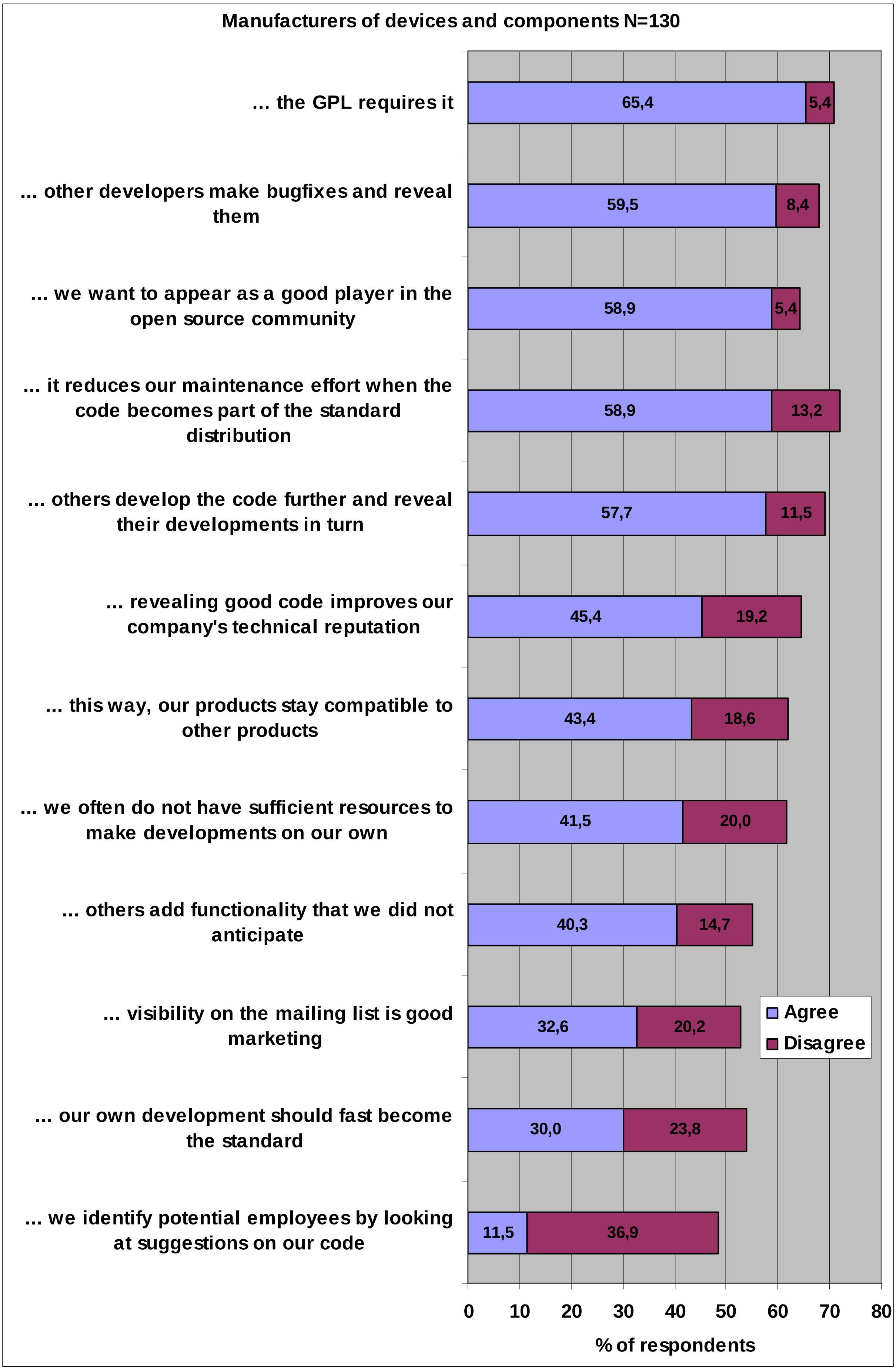
Question: “How would you describe the embedded Linux code your company typically reveals? The embedded Linux code that my company typically reveals…”



Reasons for companies to reveal code

Question: “What are the reasons for your company to reveal code? Please indicate your agreement to the following statements. My company reveals code because…”



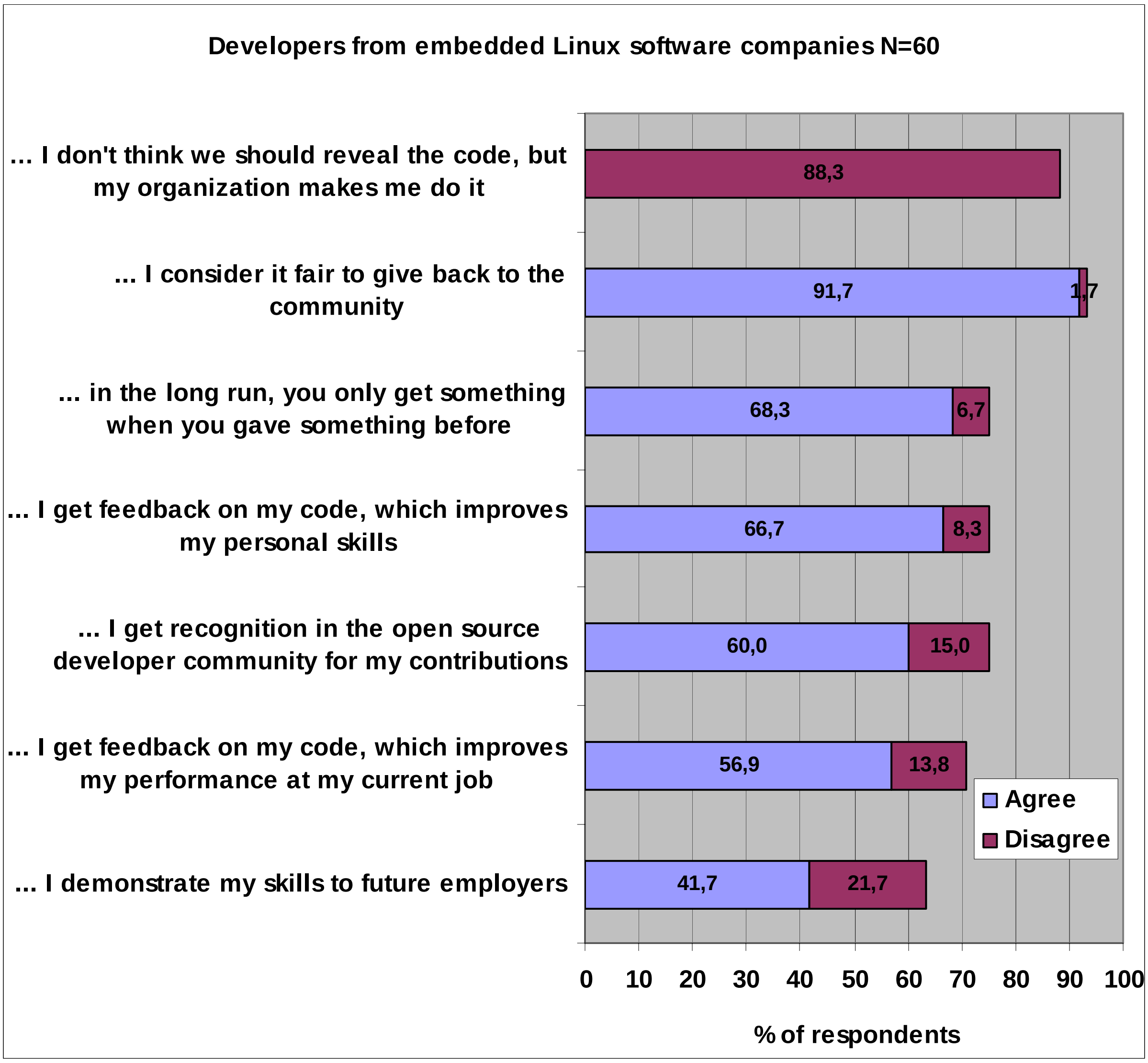

Manufacturers of devices and components N=130
... the GPL requires it
65,4
5,4
... other developers make bugfixes and reveal them
59,5
8,4
... we want to appear as a good player in the open source community
58,9
5,4
... it reduces our maintenance effort when the code becomes part of the standard distribution
58,9
13,2
... others develop the code further and reveal their developments in turn
57,7
11,5
... revealing good code improves our company's technical reputation
45,4
19,2
... this way, our products stay compatible to other products
43,4
18,6
... we often do not have sufficient resources to make developments on our own
41,5
20,0
... others add functionality that we did not anticipate
40,3
14,7
... visibility on the mailing list is good marketing
32,6
20,2
... our own development should fast become the standard
30,0
23,8
... we identify potential employees by looking at suggestions on our code
11,5
36,9
Agree
Disagree
0
10
20
30
40
50
60
70
80
% of respondents

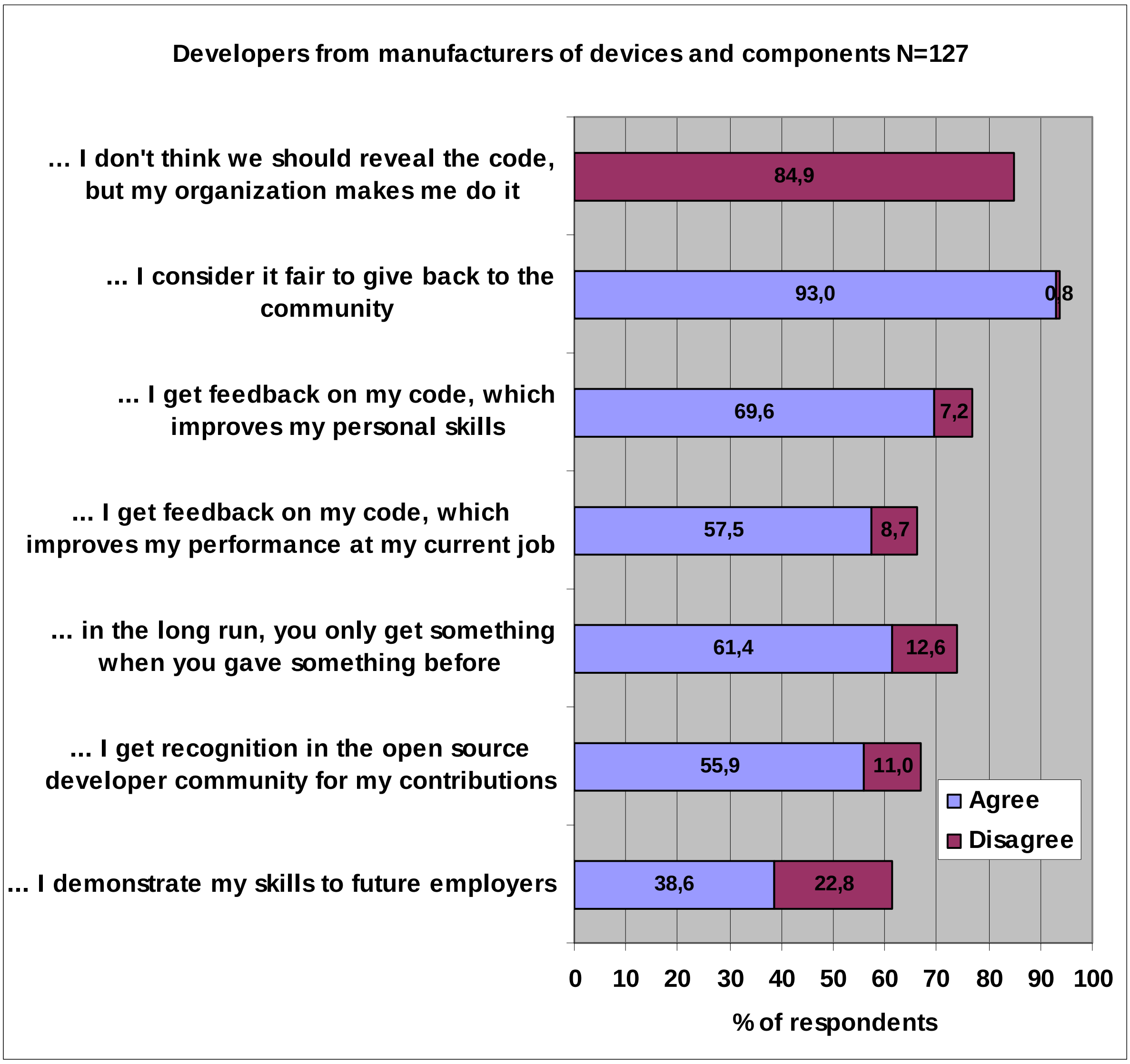
Other reasons:

- ... to find others with similar interest in a particular platform/port of embedded Linux
- This is how it all works; we stand on each other's shoulders.
- Revealing code can help prevent patent suits (by showing prior art)

Personal reasons of the developers to reveal code

Question: "What are the reasons for you personally to reveal code? Please indicate your agreement to the following statements. I reveal code because…"



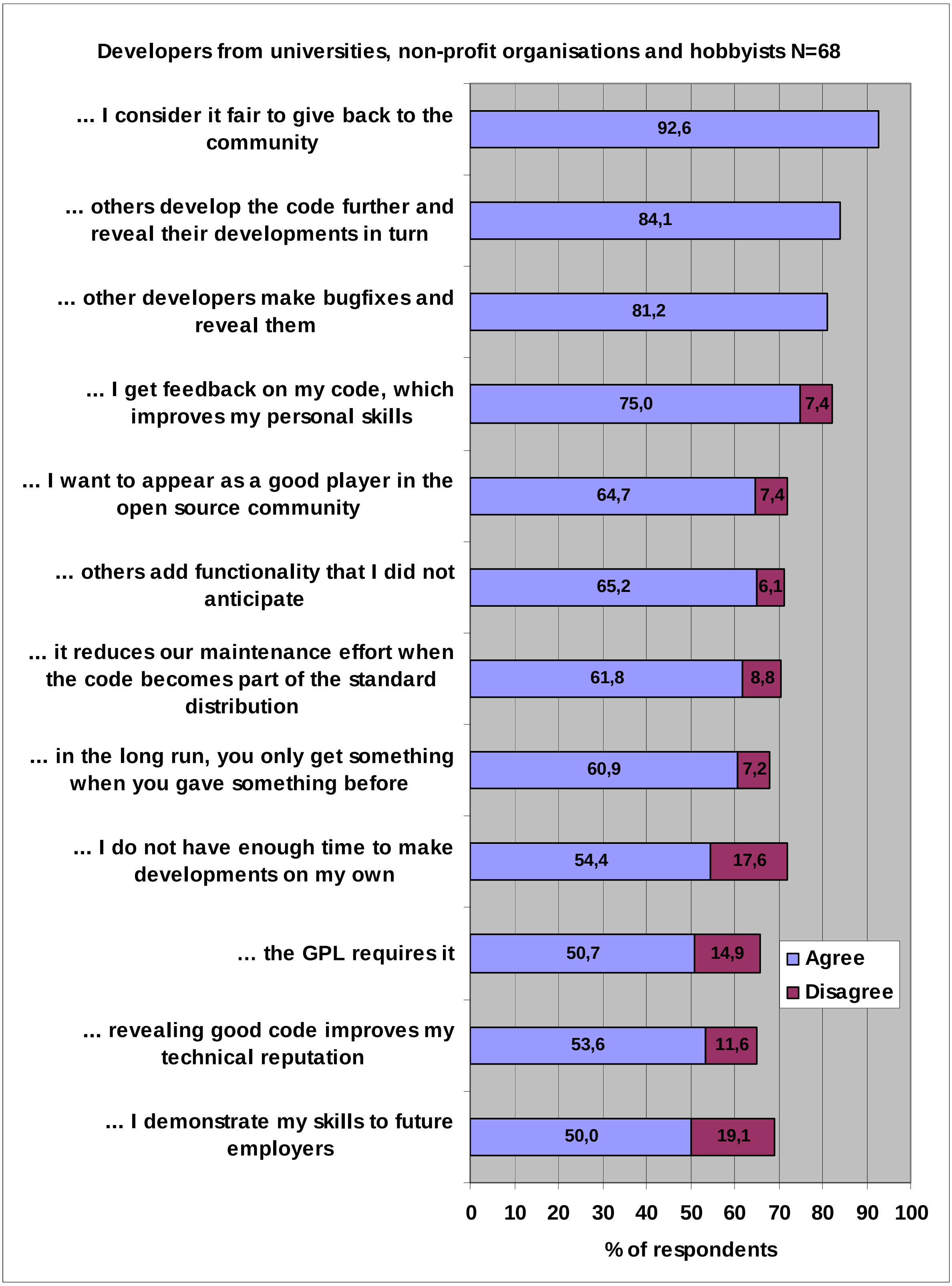


Other reasons:

- If my employer should go bust then at least my work won't be completely wasted

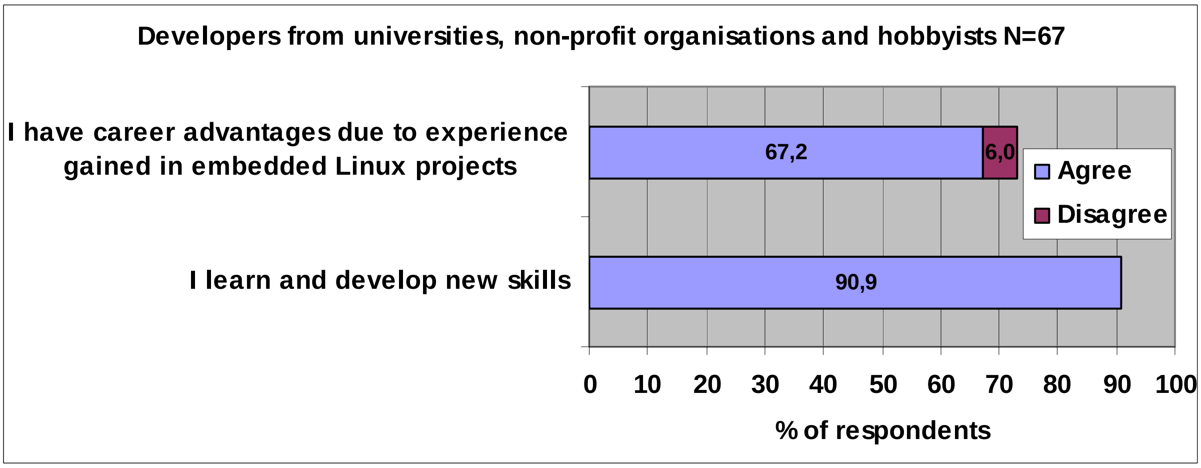
Reasons for university programmers and hobbyists to reveal code

Question: “What are the reasons for you to reveal code? Please indicate your agreement to the following statements. I reveal code because…”



Reasons for university programmers and hobbyists to develop code for embedded Linux

Question: “What are the reasons for you to develop embedded Linux code?”

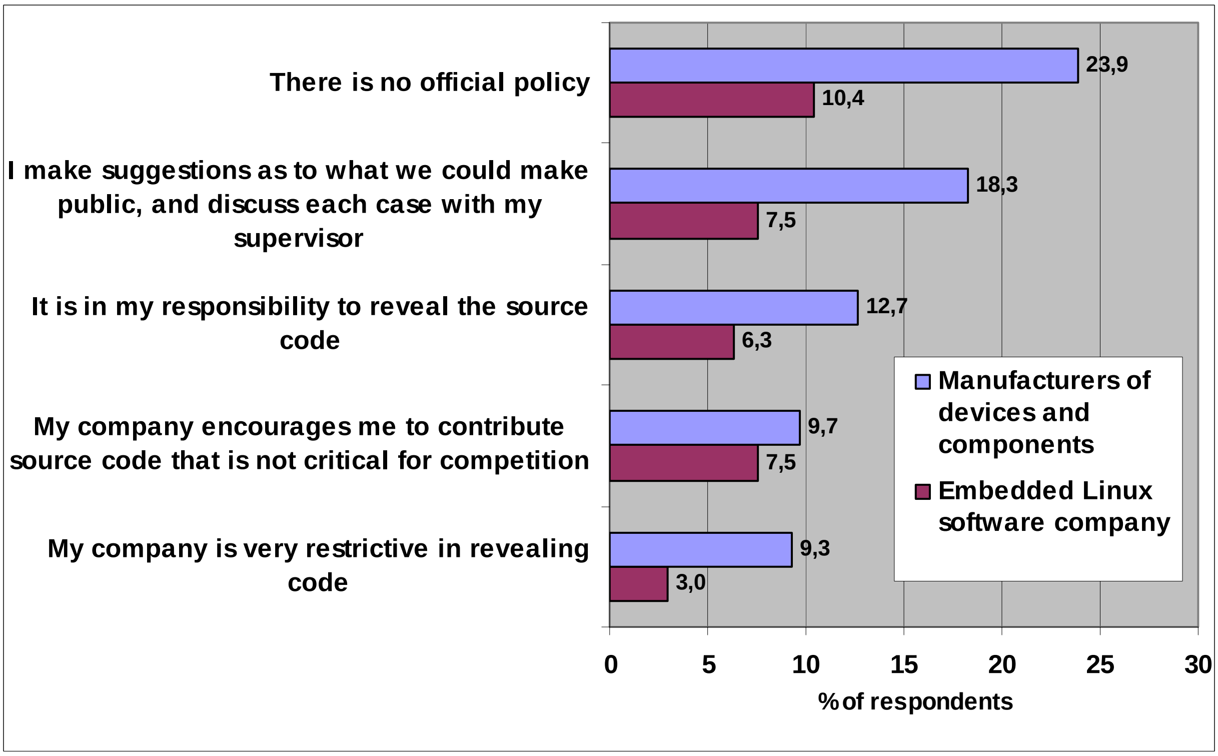


Other reasons:

- I enjoy it
- To help reduce the price of consumer electronic devices in general
- With own contribution people seem to more willing to share experience

Company policy towards revealing developments

Question: “What is your company’s official policy towards revealing developments?” (several choices were possible)



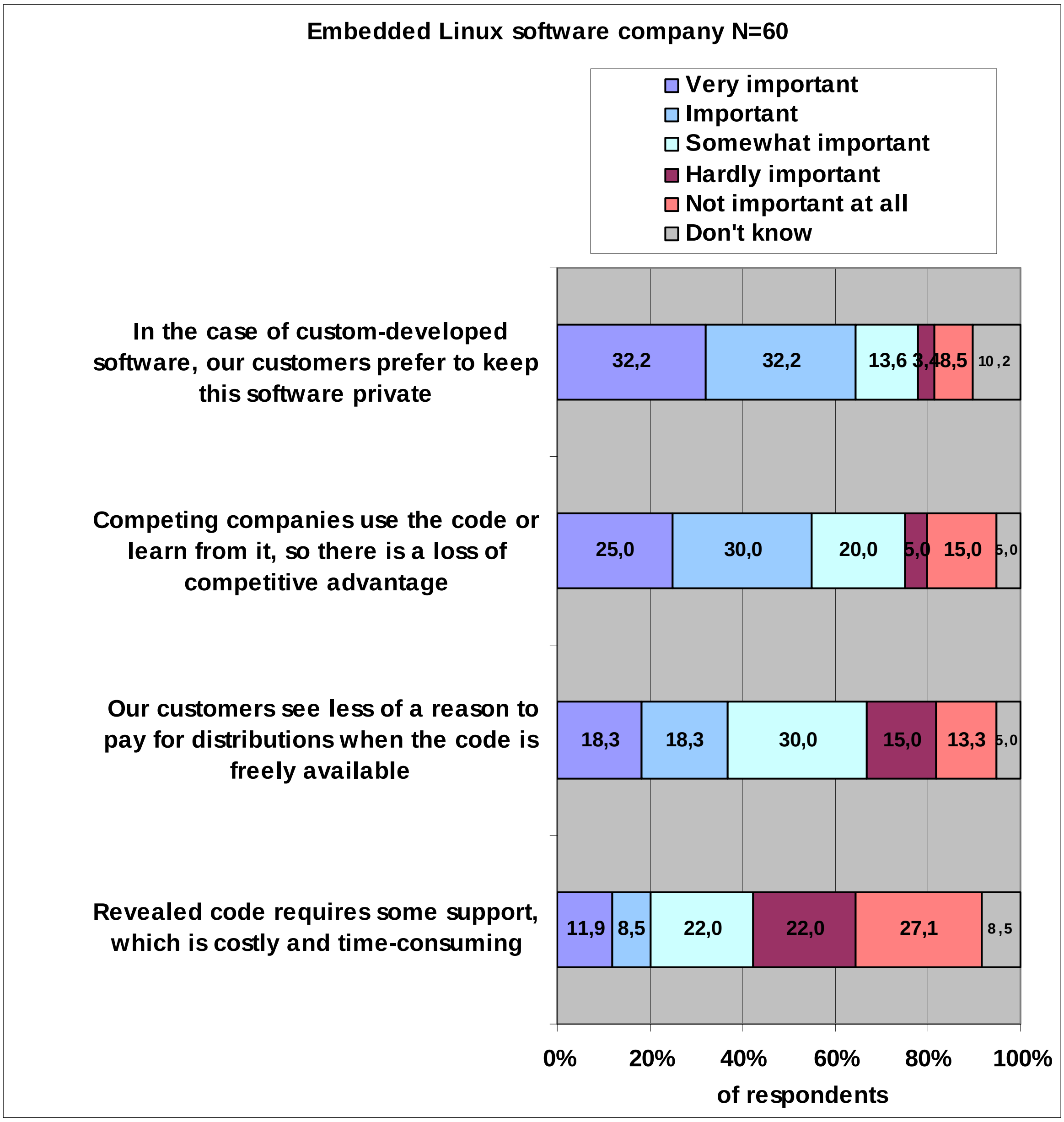
Other answers:

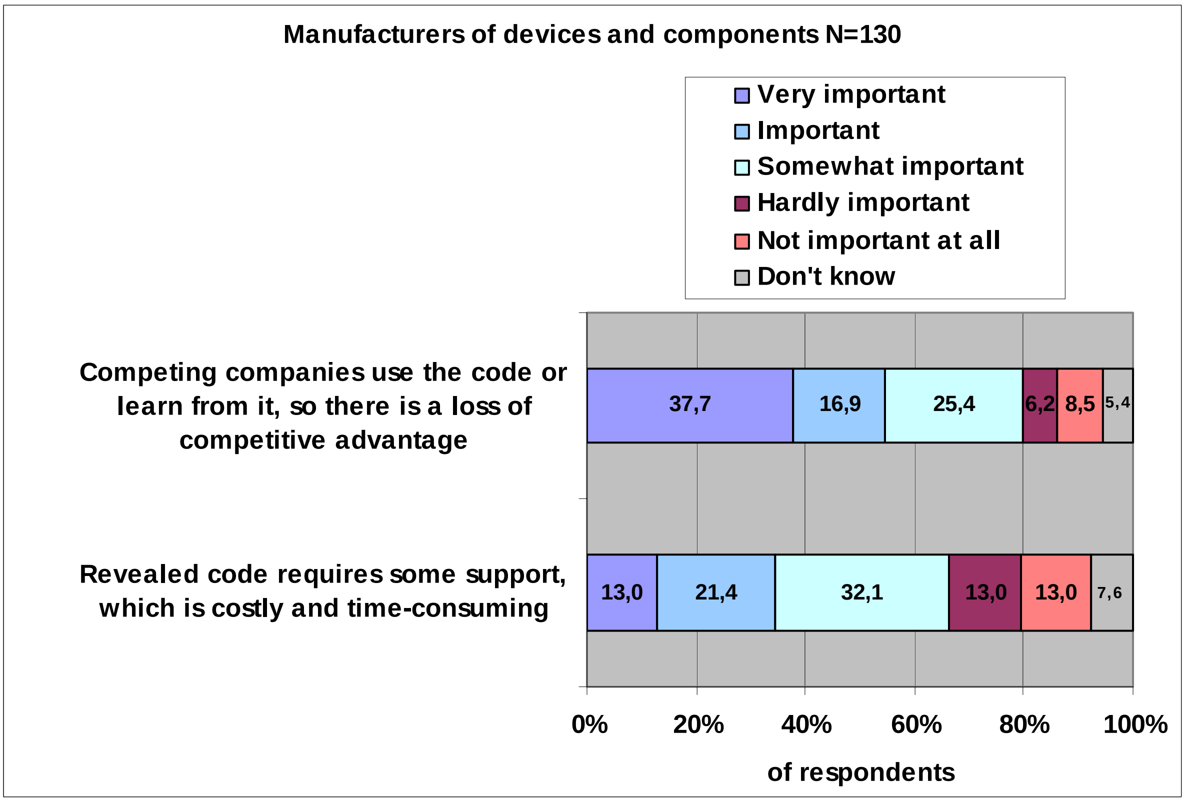
- Policy is to comply with GPL.
- It is new territory for this company. The spirit is willing, but the legal staff is weak.

5. Keeping code secret

Reasons not to make code public

Question: “What are the most important reasons for your company not to make code public?”



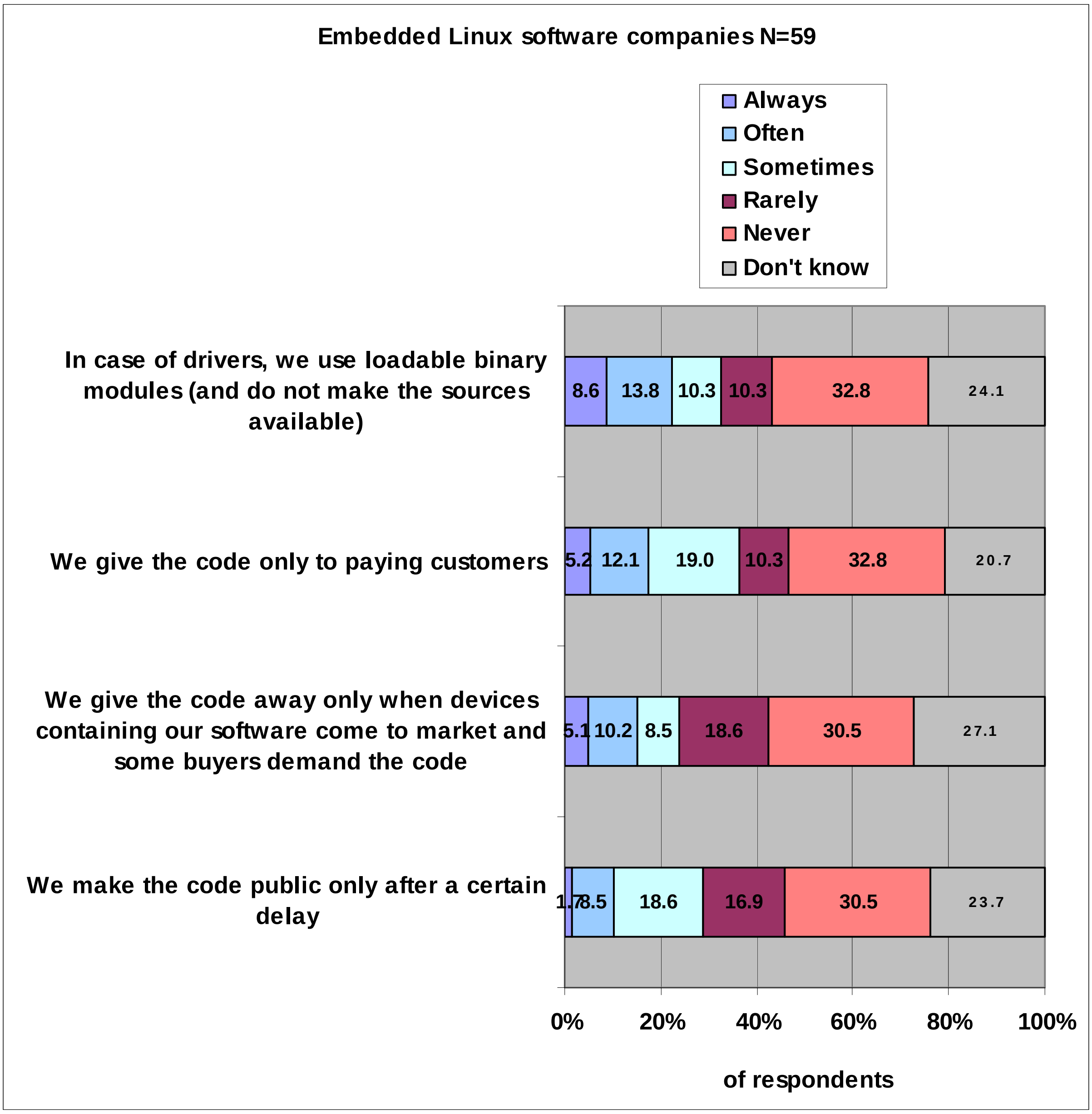


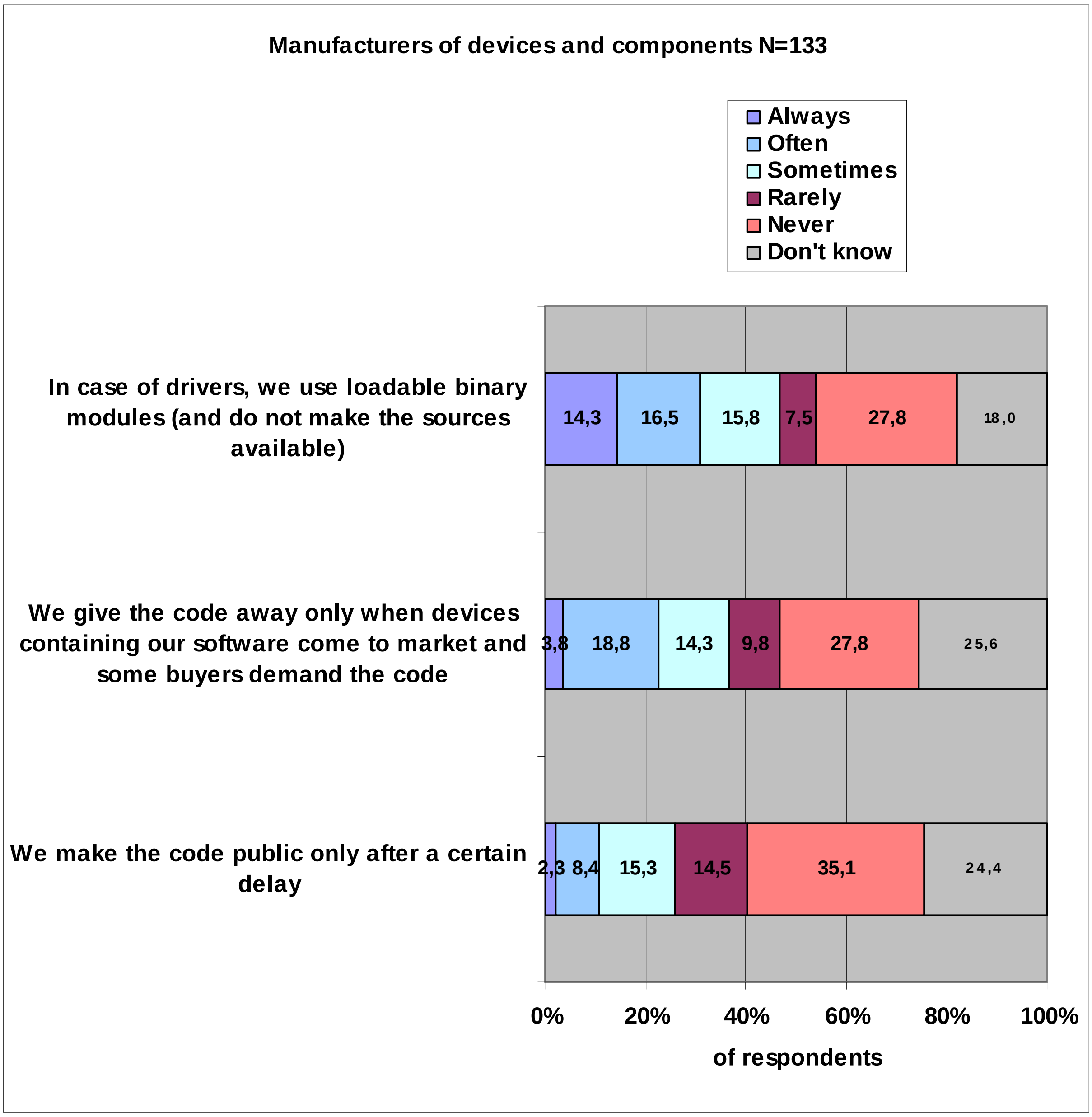
Other reasons:

- Non-disclosure agreements
- Open source development model new/unknown/absurd for managers
- The company has built a perception of a superior range of products. Revealing code would blow that image.
- Large competitors may use revealed code as a trigger to launch a patent infringement suit

How the code is kept secret

Question: “If some development is derivative work under the GPL but your company would rather keep it secret – what is done to keep those developments from becoming public? Please indicate how often the following means are used in your company.”





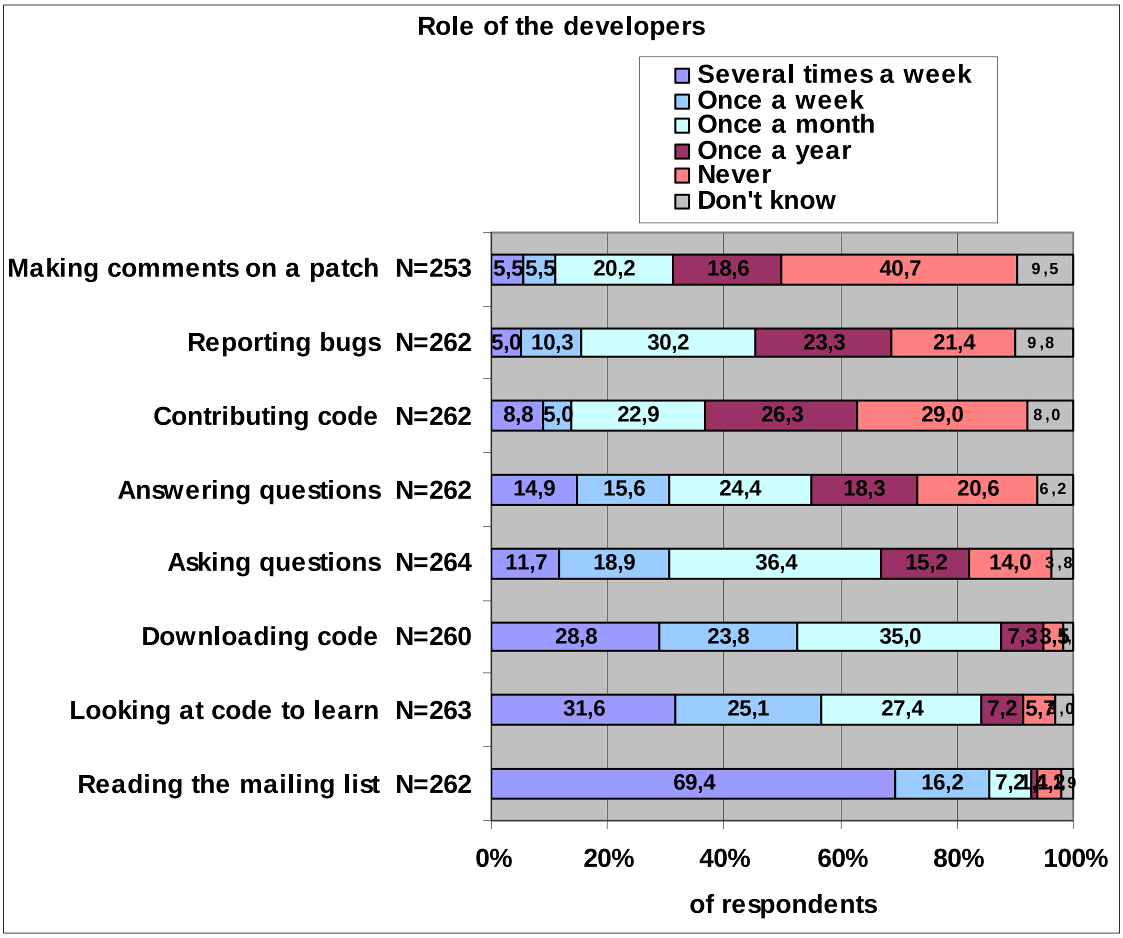
Other answers:

- Before GPL code is used in any products, a great deal of thought is given to whether or not it is a portion of the project that can be made public. If it can not be made public, the desired functionality is written from scratch.
- Delays are generally because of lack of time to devote to such "low priority tasks"
- We take measures to make the code none-derivative (i.e. introduce barriers from the GPL'd code)

6. Cooperation

Cooperation and the role of the developers

Question: “Please describe your role in public embedded Linux projects (e.g. RTAI, uclibc, PPC embedded etc.)”

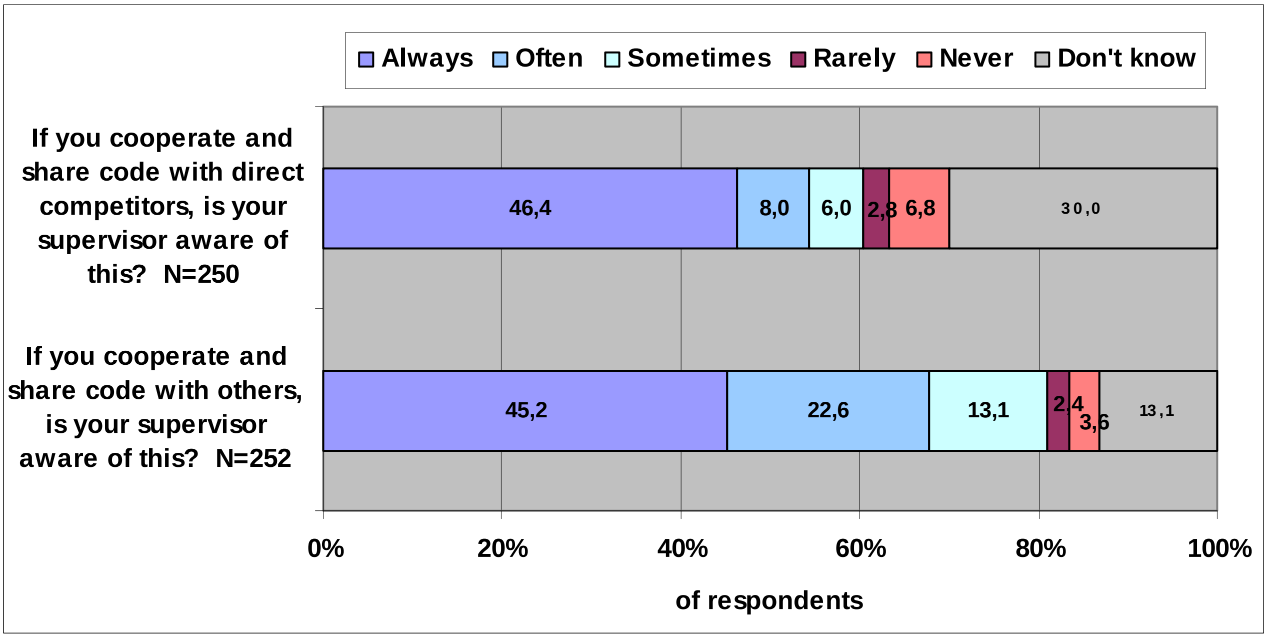


Cooperation during last month

Question: “Asking and answering questions: How often during the last month did you…”

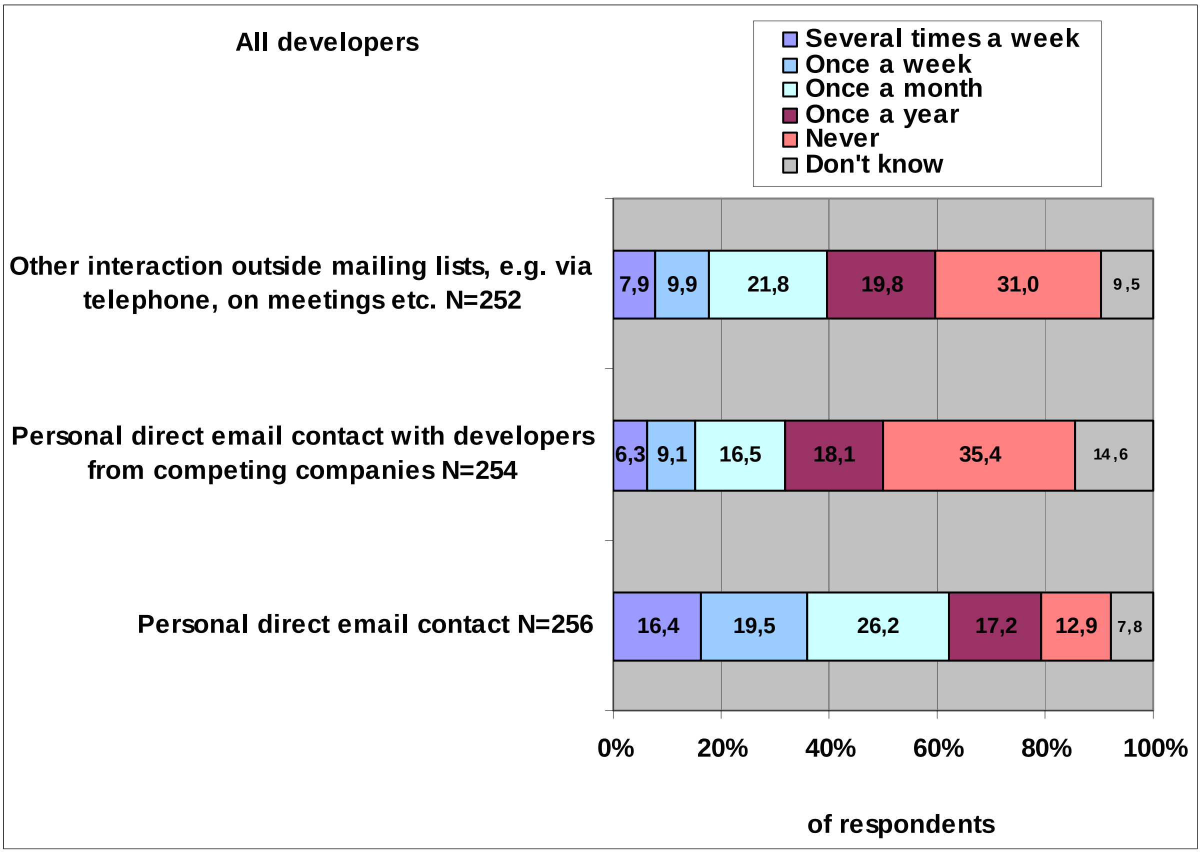
	… answer somebody’s question relating to embedded Linux on a public mailing list?	… answer a question which was relevant for competition?	… write a question relating to embedded Linux on a public mailing list?	… get an answer to your question on the mailing list?	… ask other developers for existing unpublished code?	… get these pieces of code after you had asked for them?
Mean	10,5	5,6	3,4	4,1	3,9	4,3
Median	3,0	2,0	2,0	2,0	1,0	1,0
Standard dev.	32,8	9,1	5,7	9,4	13,6	14,6
Minimum	1	1	1	1	1	1
Maximum	300	50	50	100	100	99
Missing data	128	221	114	119	214	222
N	140	47	154	149	54	46

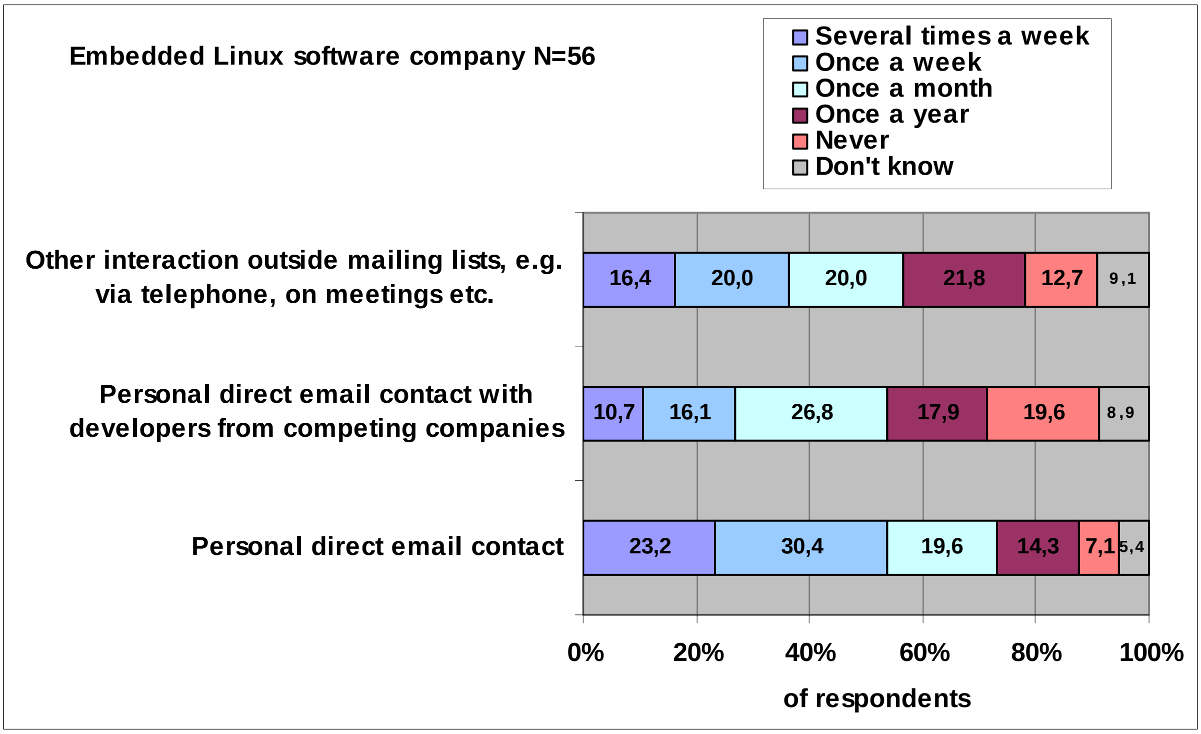
Is supervisor aware of cooperation

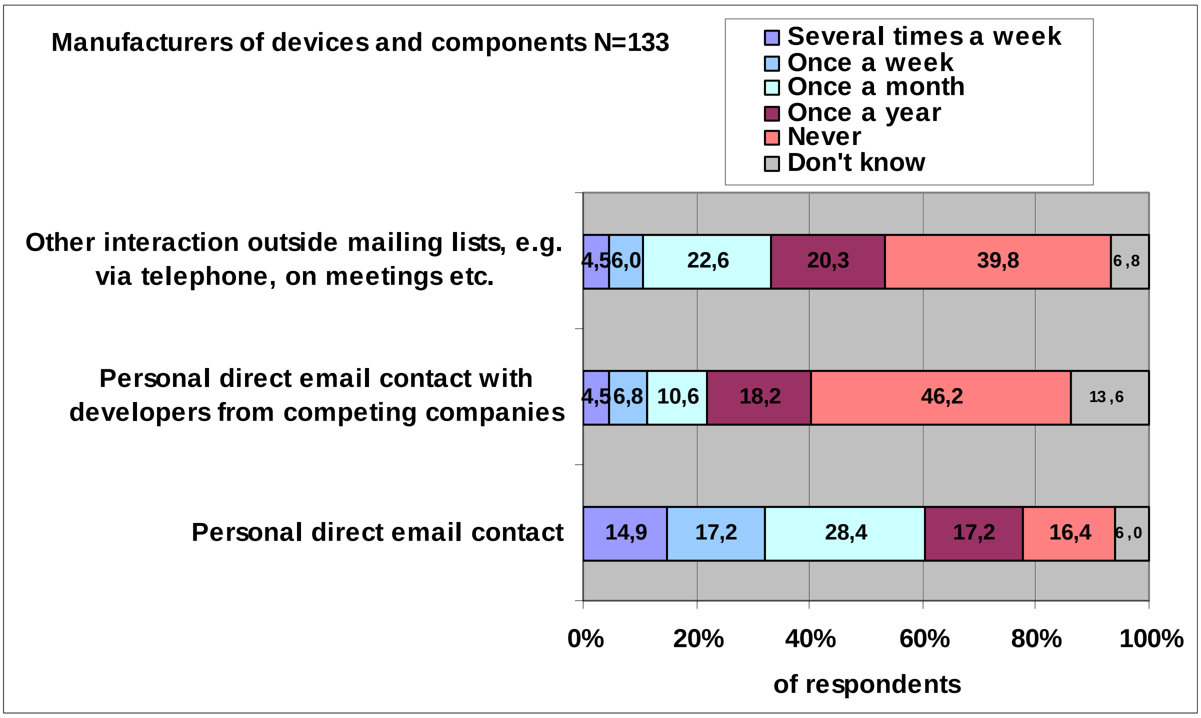


Direct contact between developers

Question: “How often do you have direct contact to other developers outsde your firm, related to discussions started on a mailing list?”

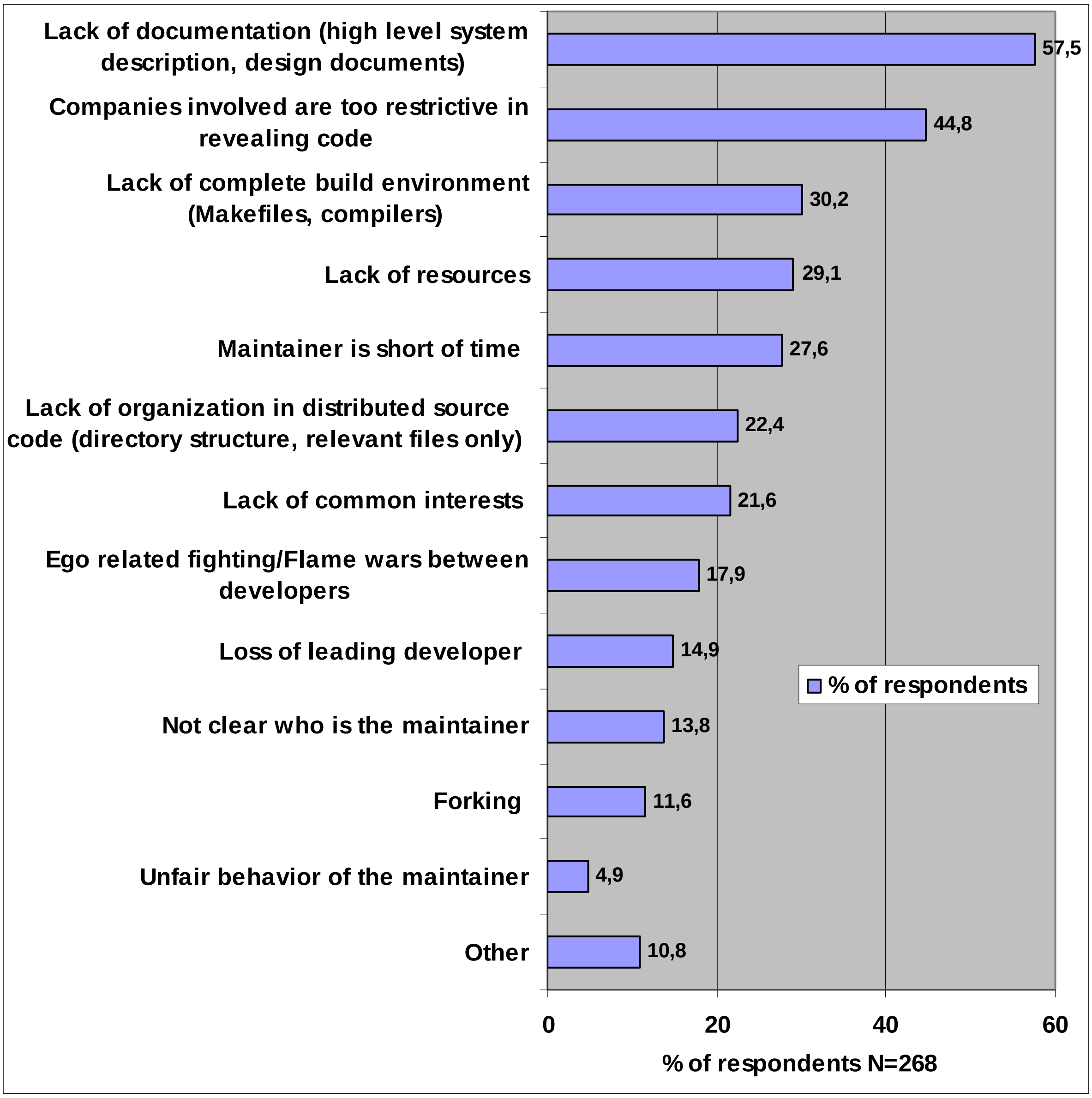





Manufacturers of devices and components N=133
Several times a week
Once a week
Once a month
Once a year
Never
Don't know
Other interaction outside mailing lists, e.g. via telephone, on meetings etc.
4,5
6,0
22,6
20,3
39,8
6,8
Personal direct email contact with developers from competing companies
4,5
6,8
10,6
18,2
46,2
13,6
Personal direct email contact
14,9
17,2
28,4
17,2
16,4
6,0
0%
20%
40%
60%
80%
100%
of respondents

Factors that hinder a public embedded Linux project

Question: “Which factors hinder a public embedded Linux project in its development (from your own experience)? Select all that apply.”



Other factors that hinder a public embedded Linux project:

- A lot of poorly written code; lots of unnecessary changes made to data structures which affect other code
- Commercial interests of some companies
- Companies not letting developers participate as much as they wish they could
- Companies that die without releasing code (e.g., RidgeRun)
- I have seen a certain mindset among fellow programmers that linux is difficult and is for only a certain "enlightened" group of people
- Lack of 'version discipline'. Typically, a developer writes a software using newest versions of all libraries disregarding current stable releases. Getting a simple thing to work usually involves painful compiling/cross compiling number of libraries.

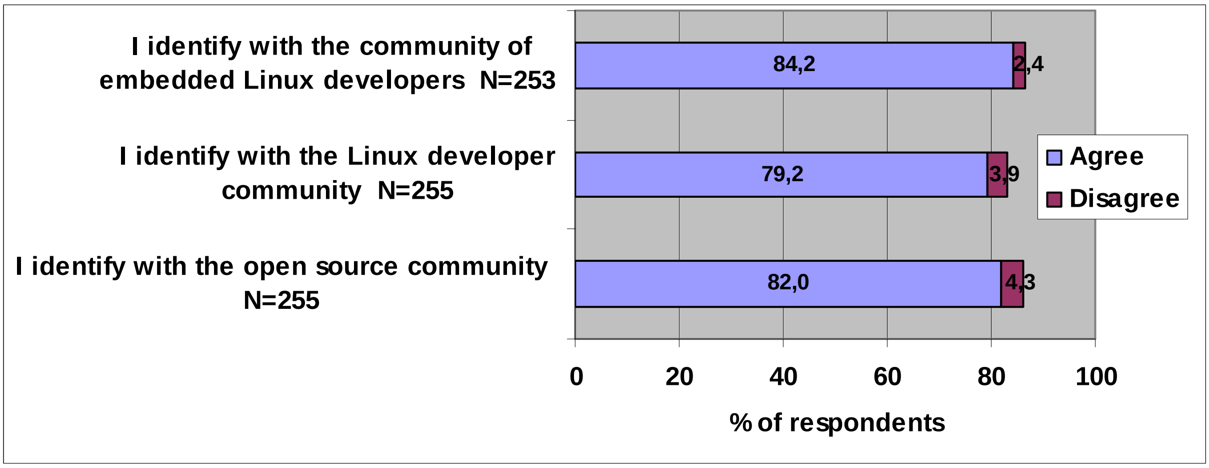
- Lack of advertising
- Lack of commonly available embedded hardware
- Lack of discipline
- Lack of funding
- Lack of Hardware Documentation not covered by NDA and unwillingness of companies to work with open source
- Lack of knowledgeable developers who can actually fix problems
- Lack of support
- Lack of time after a day at work
- Many ambitious projects only work for a narrow range of options
- Poor decision making that doesn't analyze the HW and SW Risks and look at them as an integrated whole.
- Poor leadership
- Religious fervor of Linux users who tend to scare away businesses away from Linux
- Structure of the GPL
- Too much competition beween distributors
- Toolchains come and go as companies pull them back into their private holdings.
- Unfair behavior of MontaVista
- Useful sources leached into private trees

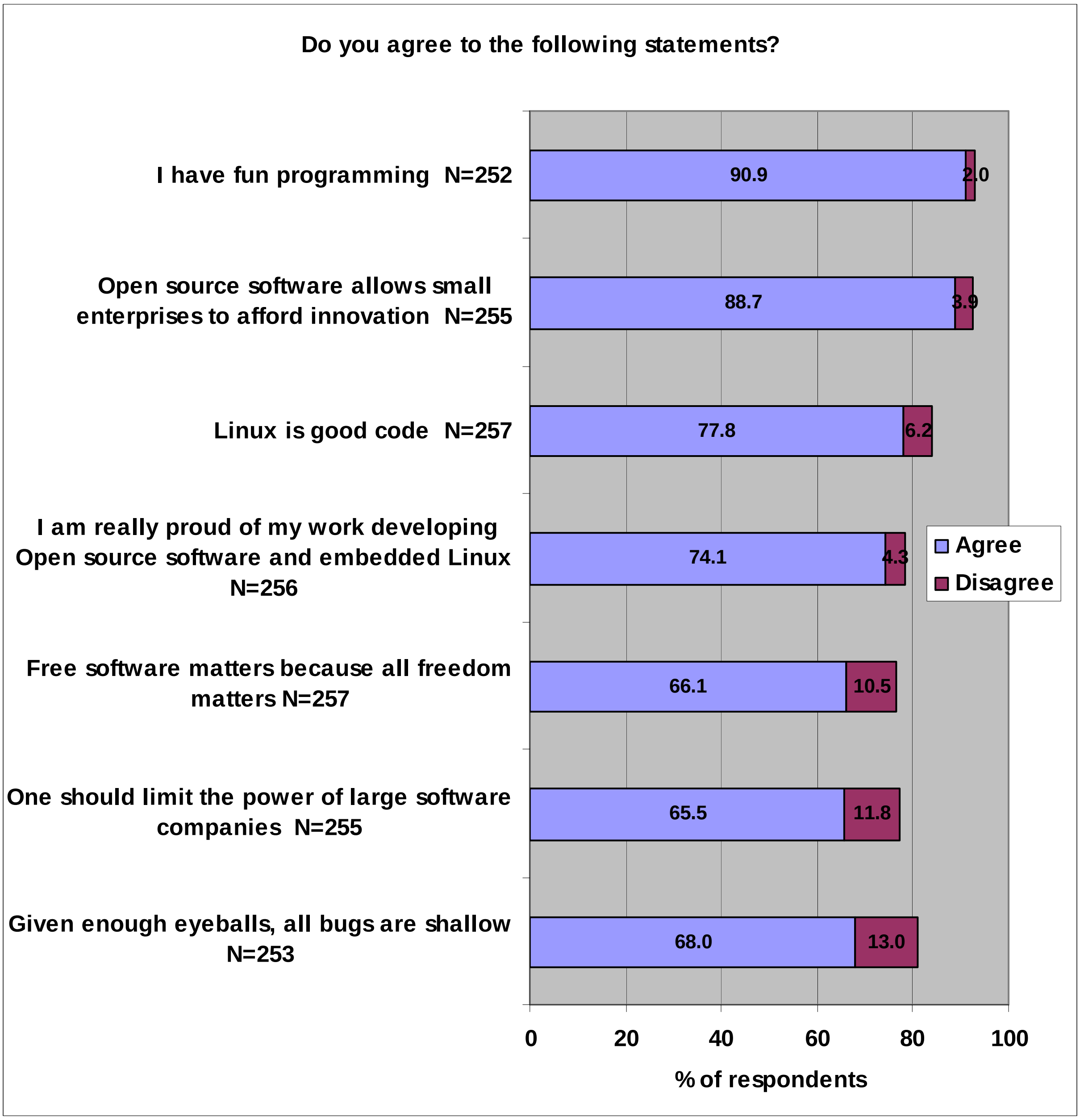
7. Identification of developers with open source ideas

Spending time on OSS

	In your spare time, how many hours per week do you work on Free/Open Source software?
Mean	8.3
Median	5
Standard dev.	9.1
Minimum	1
Maximum	60
Missing data	106
N	162

Identification with the community


I identify with the community of embedded Linux developers N=253
I identify with the Linux developer community N=255
I identify with the open source community N=255
Agree
Disagree
% of respondents


Do you agree to the following statements?
I have fun programming N=252
Open source software allows small enterprises to afford innovation N=255
Linux is good code N=257
I am really proud of my work developing Open source software and embedded Linux N=256
Free software matters because all freedom matters N=257
One should limit the power of large software companies N=255
Given enough eyeballs, all bugs are shallow N=253
Agree
Disagree
% of respondents