Free/Libre/Open Source Software for developing countries: skills, employment and costs

USUARIA 2005

2nd National Congress on Software Libre

Buenos Aires, Argentina, June 7, 2005

Rishab Aiyer Ghosh
            rishab@dxm.org

MERIT, University of Maastricht, The Netherlands
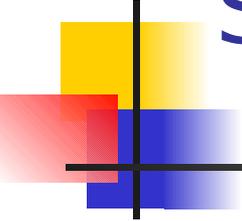
Software in society

"Access [to ICTs] is not enough, it is the ability to create, to add value, that is important"

Felipe Gonzalez
former Spanish Prime Minister,
Speaking at Open Source conference in Málaga, Spain, 18/2/04
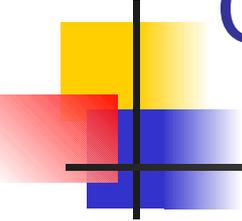
Costs, skills and economic growth

Costs

Windows Office, US$560, is 2.3 months of average income in Brazil, equivalent to US$ 6777 in the US. Cost does matter.

Skills development: "the ability to create"

FLOSS is a training environment that increases the earning capacity of community participants without any explicit investment in training: a novel form of technology transfer

Economic growth: "ability to add value"

FLOSS allows local entrepreneurs to provide a greater share of total value added, thus retaining a greater share of profits within the local economy

Skills and economic growth

Skills development: "the ability to create"
FLOSS is a training environment that increases the earning capacity of community participants without any explicit investment in training: a novel form of technology transfer

Economic growth: "ability to add value"
FLOSS allows local entrepreneurs to provide a greater share of total value added, thus retaining a greater share of profits within the local economy
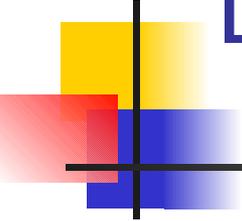
Local value addition: proprietary

Building over a platform
This applies equally to any platform, which is simply used as a (non-modifiable) base on which new services or software are built: 100% of the added value is local

Sales commissions
Something which is rarely possible with free software, but also represents little value. Only the commission is retained locally, which is a small part of the total value.

Support, integration, customisation...
Local value addition limited, as "deep" (high-value) services require "deep" access – only the proprietor has it.
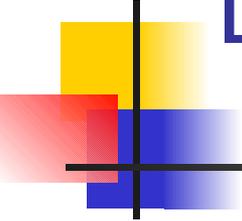
Local value addition: FLOSS

Building over a platform

As with proprietary software, free software platforms can be used as a (*modifiable!*) base on which new services or software are built: 100% of the added value is local

Sales commissions

Rarely possible with free software, but also represents little value. However, the entire "sale price" can be retained locally, as no proprietor has to be paid a royalty or licence.

Support, integration, customisation…

Local value addition extensive, as "deep" access is available. 100% of such services can be provided locally, retaining 100% of the value locally.
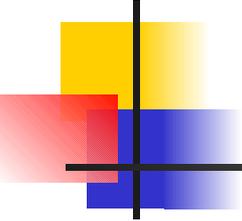
"Deep" support, more local value

Local companies are limited in the integration and support services they can provide for proprietary software

Deep support: fixing software bugs, customising it to user requirements, or integrating extensively with other software requires deep access.
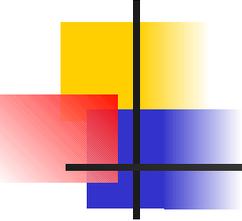
"Deep" support, more local value

Deep access to proprietary software is controlled by the proprietor (limits access or requires royalties, diminishing value retained locally)

Deep access to free software is available to anyone – limited only by their skills. This allows every provider to potentially provide deep support services, and retain 100% of the value.
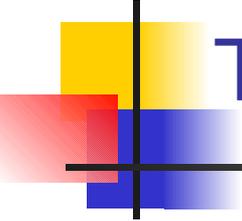
The importance of customisation…

Custom or in-house software represents about 67% of total software produced (in the US; more elsewhere)

If based on free software, custom solutions greatly benefit the solutions provider who captures 100% of the total value, not just the value added locally – no royalties/licences paid
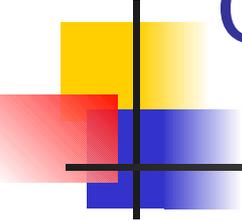
Code re-use, higher service levels

Free software allows providers to reuse code rather than build from scratch, and to reuse a huge base of code written by others

Re-using (and modifying) allows the creation of much better end-user solutions for the same effort than writing from scratch

Put together, this provides better value for money for customers and better profit margins for local service providers
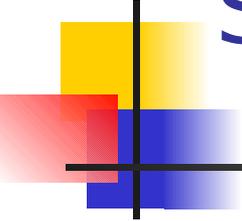
Skills and economic growth

Skills development: "the ability to create"

FLOSS is a training environment that increases the earning capacity of community participants without any explicit investment in training: a novel form of technology transfer

Economic growth: "ability to add value"

FLOSS allows local entrepreneurs to provide a greater share of total value added, thus retaining a greater share of profits within the local economy
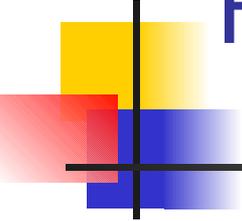
FLOSS develops local skills

FLOSS encourages not only passive "use" but active participation in the creative process

FLOSS provides a very low barrier to entry for creativity – you don't *have* to be creative but if you want to, you easily *can*
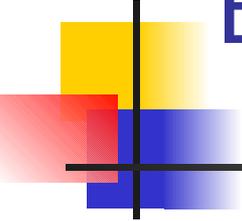
But do we all want to program?

How will we know, unless we can try?

HTML is a programming language – the web only took off because it was open, so people could learn to write their own sites just by "viewing source", copying and modifying other websites

"Programming" covers a very broad range of skills from HTML to C; FLOSS allows entry at any degree with little investment in time or effort

Why develop FLOSS?

(Source: "FLOSS Final report", Ghosh et al)
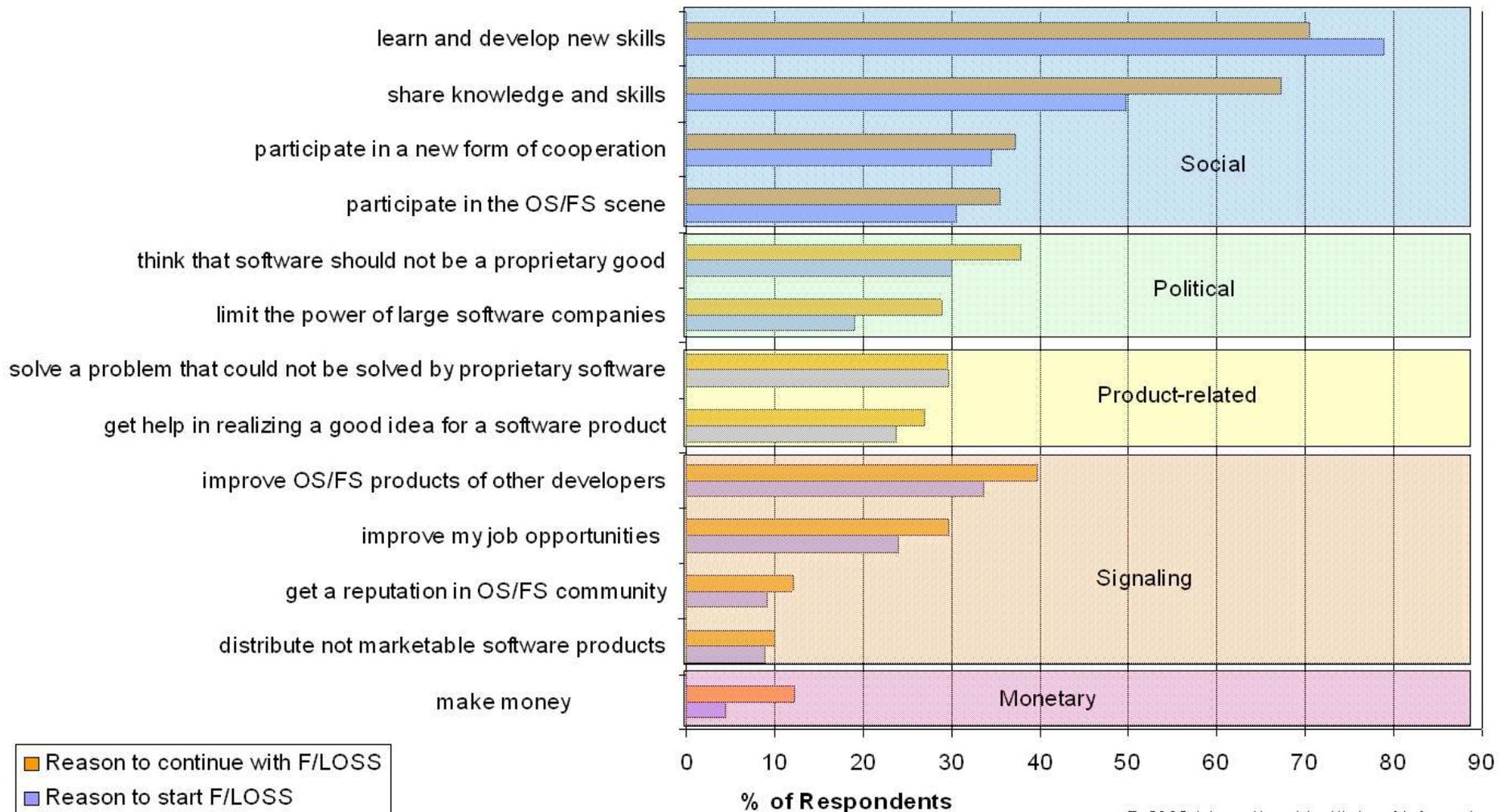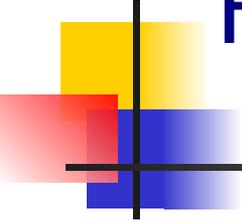
FLOSSPOLS Skills Survey

How is learning organised in the FLOSS community?

What mechanisms and patterns can be observed?

For which purposes do community members learn?

What is the impact of skills learnt on employment potential?
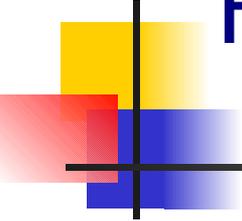
FLOSSPOLS Skills Survey

Aim: to study both skills learnt and impact on employability.

Separate questionnaires sent to:

developers (worldwide)

employers (EU)
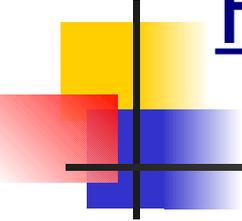
Hypotheses

Technical skills

    New participants should learn various skills

    Experienced participants should learn too

Management skills

    New and experienced participants should learn teamwork, coordination and management skills
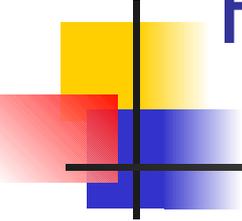
Hypotheses

Legal skills

- Participants should learn legal skills, more than in formal (non-legal) courses

General skills

- Non-English speakers improve their English
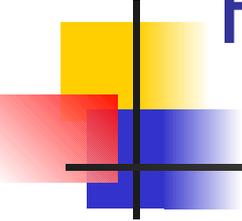
Hypotheses: formal learning

In comparison with formal ICT courses:

- FLOSS provides a better, practical learning environment for many technical skills:
  - Writing re-usable code & debugging
  - Working with code written by others
- FLOSS provides a better learning environment for most legal and teamwork skills, which are rarely taught in formal ICT courses
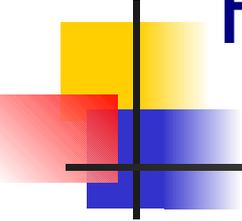
Hypotheses: learning methods

Formalised knowledge transfer less common than self-organised knowledge absorption, *even where* the knowledge is codified rather than tacit

Methods of "learning by doing" would be seen as more useful for technical skills development than formal training

(There is no uniform "learning strategy" for skills improvement within the community, even for those with the same level of prior knowledge)
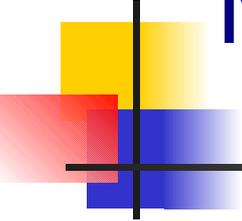
Methodology

Personalised survey: Based on FLOSS Developer Survey 2002, addressing 1449 of the more than 2800 respondents

Response: 361 respondents from 1151 valid addresses (31%)

Additional 1091 respondents from an open online survey (not personalised, on slashdot etc, results not used today)

Employer survey on value of FLOSS-skills for firms in different industries (software, business services, finance)

UK, France, Germany, Spain

147 responses so far (of which 128 are used here)
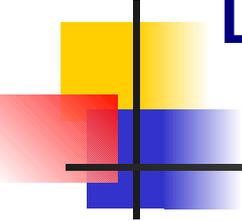
Developers: demographics

8.7 years of FLOSS participation

31 years old

All earn direct or indirect FLOSS income

68% employed, 13% students, 15% self-empl.

75% hold a university degree

Findings: types of participation

"How would you characterise your participation in the FLOSS community?

Legend: ■ very often ■ often ■ sometimes ■ seldom ■ not at all

n = 361
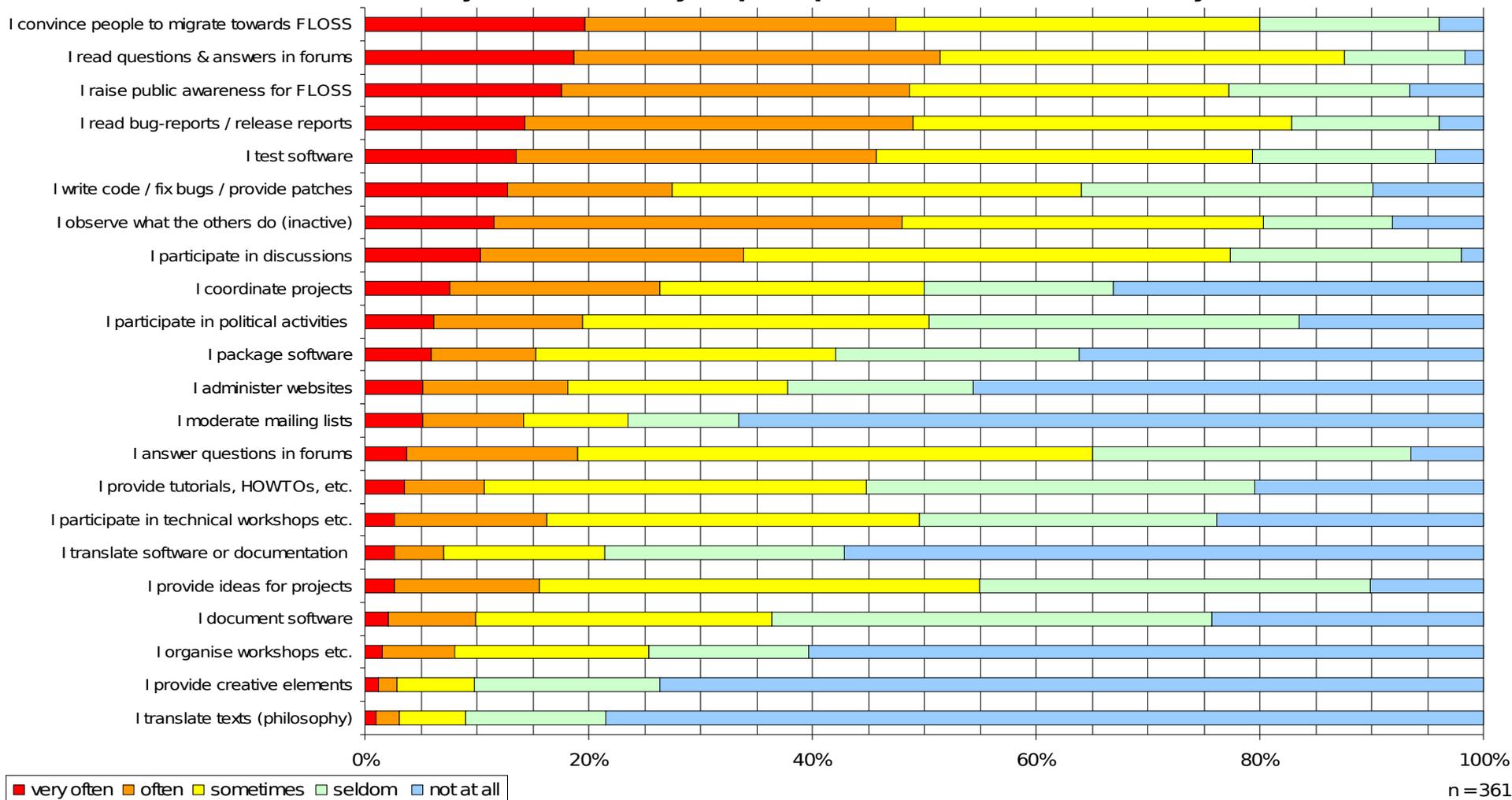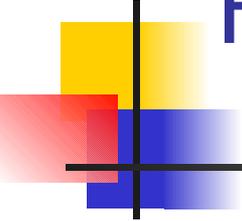
Findings: types of participation

Principal component analysis shows 6 activities:

- Organisational (workshops, websites, projects)
- Bugfixing / coding / testing activities
- Political activities (convincing others)
- Translation
- Communication (discussion, online forums)
- Support (graphics, documentation, tutorials)

Prior knowledge

Knowledge about FLOSS before joining the community

Question	I knew really a lot	I knew quite a lot	I knew little	I knew nothing
How much did you already know about the difference between FLOSS licenses and proprietary licenses?	7,0	25,5	43,7	23,8
How much did you already know about the philosophy of free software?	5,3	24,9	45,4	24,4
How much did you already know about the organisation of work and cooperation?	1,4	10,9	49,0	38,7
How much did you already know about technical aspects of FLOSS?	6,2	29,1	52,9	11,8

■ I knew really a lot ■ I knew quite a lot □ I knew little □ I knew nothing

Prior knowledge and skills learnt

Those with a lot of prior technical knowledge:

Less likely to learn English "a lot" (35%) than those without prior knowledge (47%)

Less likely to learn Basic programming "a lot" (34%) than those without (54%)

More likely to learn familiarity with progr. langs. "a lot" (49%) than those without (40%)
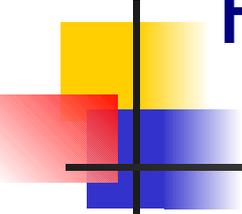
Prior knowledge and skills learnt

Those with a lot of prior knowledge on how "work and cooperation in FLOSS is organised":

- *More* likely to learn how to keep a community going (35%) than those without (13%)
- *More* likely to learn to motivate others (27%) than those without (10%)
- *More* likely to learn planning and scheduling (13%) than those without (5%)
- *More* likely to learn to achieve/define targets (24%) than those without (12%)

Skills learnt: technical

Improvement of technical skills through participation in the FLOSS community

Skill	a lot	some	little	nothing
To create new algorithms	11.7	28.2	38.1	21.9
To document code	18.2	42.8	29.3	9.7
To design modular code	32.7	35.6	21.6	10.2
To write code in a way that it can be re-used	42.4	36.0	12.7	8.9
To become familiar with different programming languages	43.0	33.2	20.3	3.4
To look for and fix bugs	43.6	33.4	18.6	4.4
Basic / introductory programming skills	46.6	24.9	14.7	13.8
To run and maintain complex software systems	49.3	29.5	14.0	7.2
To re-use code written by others	50.0	34.5	12.6	2.9

■ a lot ■ some ▢ little □ nothing

n = 361

Skills learnt: management

Improvement of managerial skills through participation in the FLOSS community

	a lot	some	little	nothing
To clearly articulate an argument	27.9	38.1	24.0	10.0
To express personal opinions	24.9	39.9	25.5	9.7
To accept and to respond to criticism from others	24.9	47.3	20.7	7.2
To coordinate own work with the work of others	22.8	43.7	20.1	13.5
To evaluate the work of others	17.4	46.0	25.4	11.2
To lead a project or a group of people	16.9	27.1	28.9	27.1
To keep a community going	15.9	30.0	33.0	21.1
To clearly define and achieve targets	12.8	34.6	29.9	22.7
To settle conflicts within a group	12.6	36.3	29.8	21.2
To motivate people	11.9	27.7	37.5	22.9
To plan work and stick to a work schedule	5.4	24.7	33.4	36.4

■ a lot ■ some ▨ little ▢ nothing

n = 361

Skills learnt: legal

Improvement of legal skills through participation in the FLOSS community

To understand the differences between copyrights, patents, and licences
- 57.7 | 27.4 | 12.9 | 2.0

To understand licences
- 52.0 | 37.1 | 8.3 | 2.6

To understand copyright law issues
- 49.6 | 33.6 | 14.0 | 2.8

To understand patent law issues
- 42.2 | 35.0 | 18.8 | 4.0

To improve my understanding of liability issues
- 24.6 | 39.2 | 28.9 | 7.3

■ a lot ■ some □ little □ nothing

n = 361
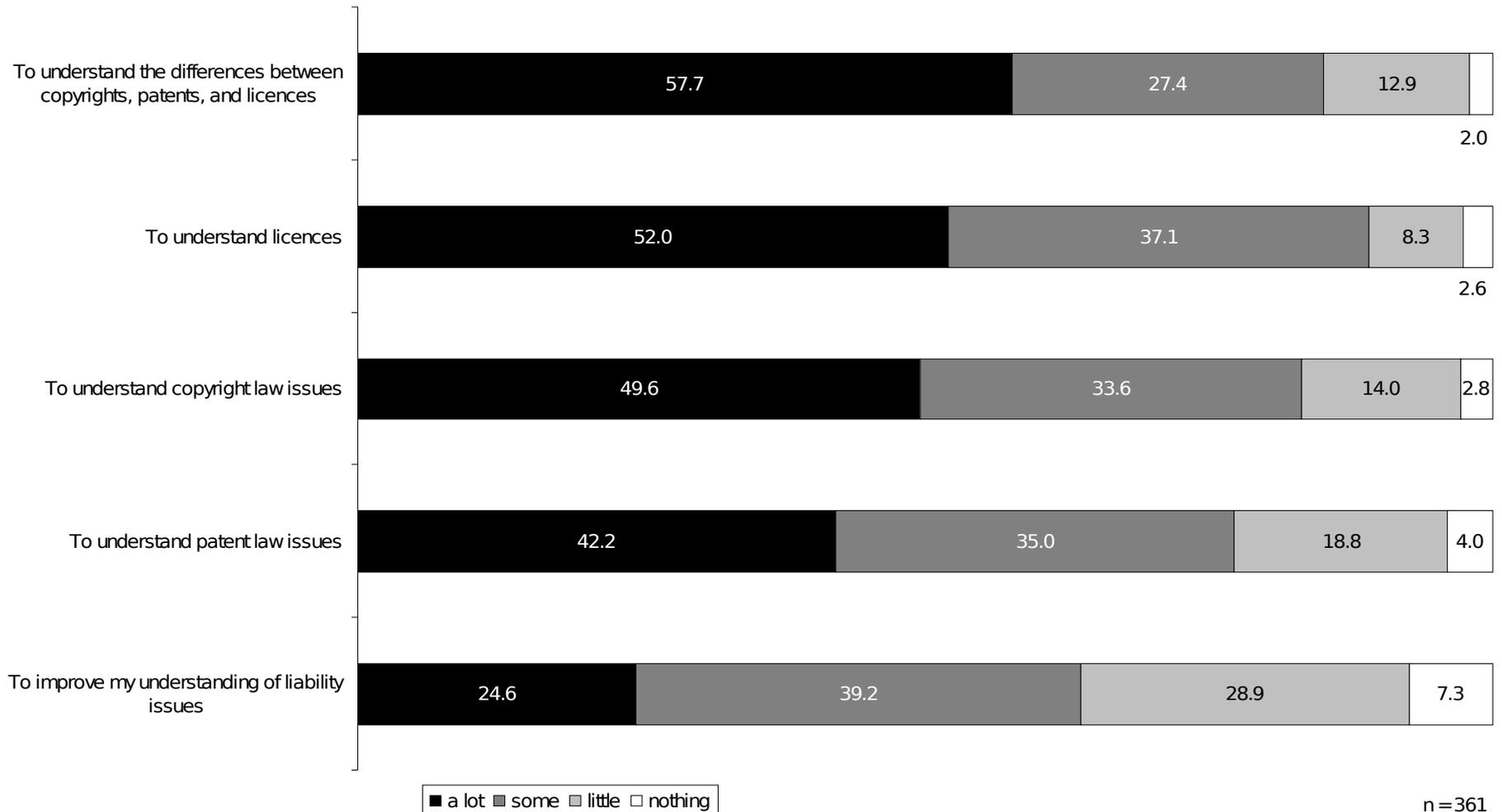
Skills learnt: general

Improvement of general skills through participation in the FLOSS community

	a lot	some	little	nothing
To better understand English, especially technical discussion	42.3	28.9	12.2	16.7
To get an overview of developments in software technology	40.4	41.0	17.8	0.9
To get an overview of the skills you need in the software professions	28.9	42.4	22.8	5.8
To understand and work with people from different cultures	23.0	43.1	24.8	9.1
To interact with other people	20.2	45.7	26.1	7.9

■ a lot ■ some □ little □ nothing

n = 361

Importance of skills learnt

Relationship between skills learnt
from FLOSS and professional career

10.00%

26.00%

64.00%

- Core skills for professional career
- Supplementary skills for career
- Skills not rel- evant to ca-
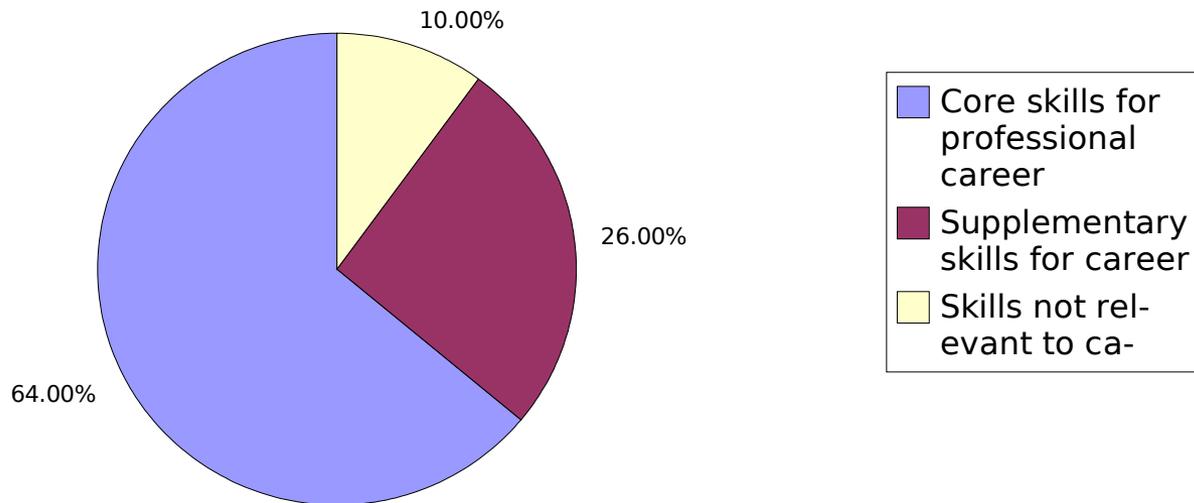
Learning compared to formal course

"Which of the following skills can be better learnt within the FLOSS community as compared to a formal computer science course?" - Developers' and employers' view


Skill	employers (n =128)	developers (n =361)
To write code in a way that it can be re-used	57.3	80.3
To accept and to respond to criticism from others	53.0	75.8
To coordinate own work with the work of others	61.5	72.8
To develop an awareness of legal issues relating to software	59.0	72.2
To run and maintain complex software systems	38.5	65.7
To evaluate the work of others	41.9	64.9
To express personal opinions	43.6	56.2
To document code	31.6	51.4
To lead a project or a group of developers	34.2	49.7
Basic / introductory programming skills	24.8	28.4
To clearly define and achieve targets	26.5	25.0
To plan work and stick to a work schedule	21.4	13.8

(multiple responses)

Learning compared to formal course

"Which of the following skills can be better learnt within the FLOSS community as compared to a formal computer science course?" - Employers' response by role of FLOSS in the organisation

To develop an awareness of legal issues relating to software: 26.0 / 72.0

To coordinate own work with the work of others: 45.0 / 70.0

To accept and to respond to criticism from others: 36.0 / 61.0

To express personal opinions: 29.0 / 51.0

To evaluate the work of others: 26.0 / 48.0

(multiple responses)

To run and maintain complex software systems: 19.0 / 47.0

To lead a project or a group of developers: 13.0 / 42.0

☐ FLOSS is not important to the employer
■ FLOSS is important to the employer

To clearly define and achieve targets: 16.0 / 31.0

Compensation for no formal degree?

Do you think that proven participation in the FLOSS community can compensate for the lack of formal degrees, like certificates or university degrees?

14.00%

16.00%

70.00%

- Yes
- No
- Don't know
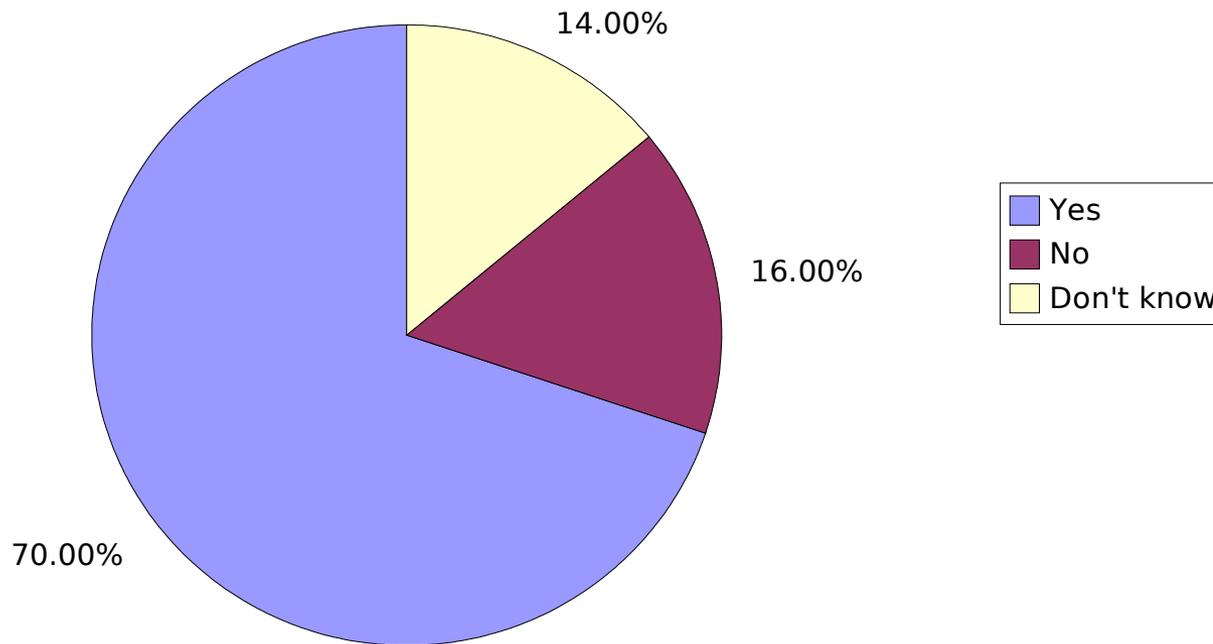
Compensation for no formal degree?

The employers' perspective: (n=114)

Crosstab

% within How do you assess the role of FLOSS within your organisation?

|		How do you assess the role of FLOSS within your organisation?		Total
		important	hardly important	
Do you think that FLOSS experience adds value to a formal comuter science qualification?	Yes	92.8%	67.7%	86.0%
	No	4.8%	9.7%	6.1%
	I don't know	2.4%	22.6%	7.9%
Total		100.0%	100.0%	100.0%

Compensation for no formal degree?

Employers' perspective: Do you offer prospective employees with FLOSS experience different pay than those with a formal degree?:

- No, they're paid the same
- Yes, those with formal degrees get paid more
- Yes, those with FLOSS skills get paid more
- Don't know

24.50%

55.10%

13.30%

7.10%
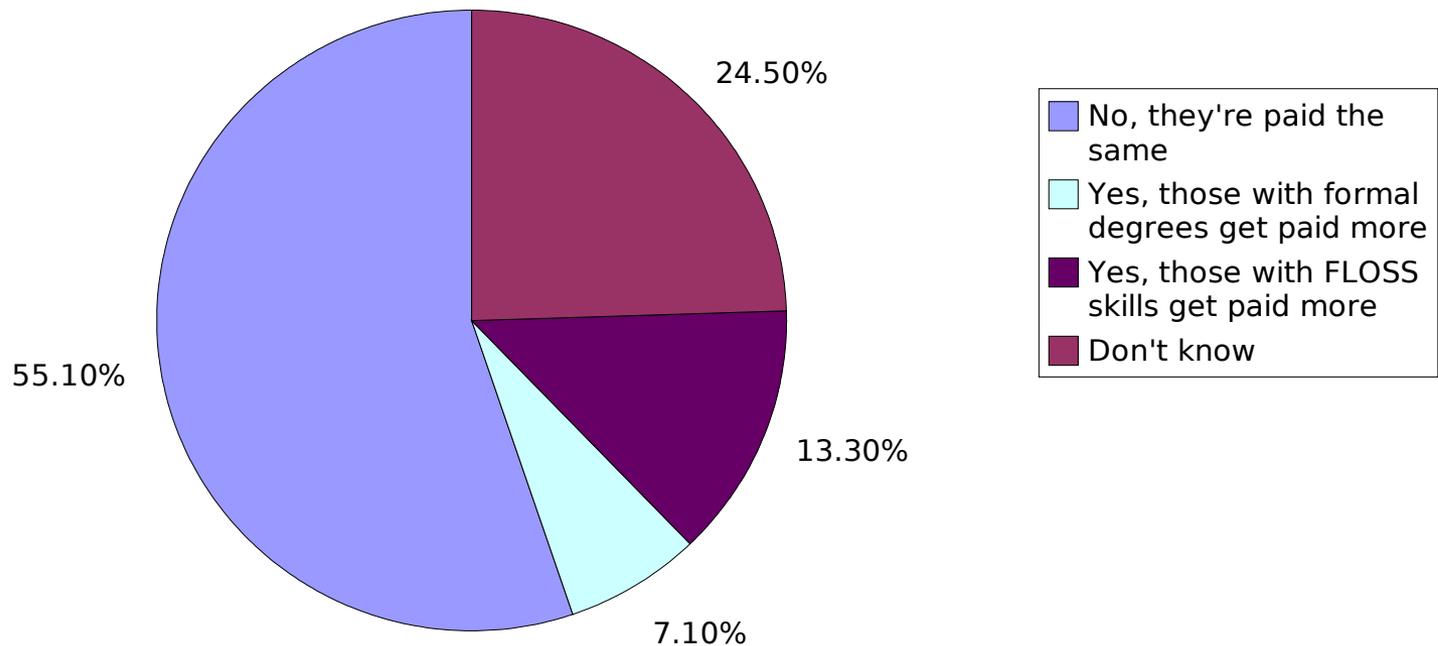
FLOSS skills valued on the job market

(Dis)advantage on the job market for someone whose past experience is as a proven developer of FLOSS software component vs someone whose past experience is as a proven developer of proprietary software component:

10.00%

11.00%

35.00%

44.00%

Legend:
- Advantage proprietary
- Advantage FLOSS
- Both equally likely to get job
- Don't know
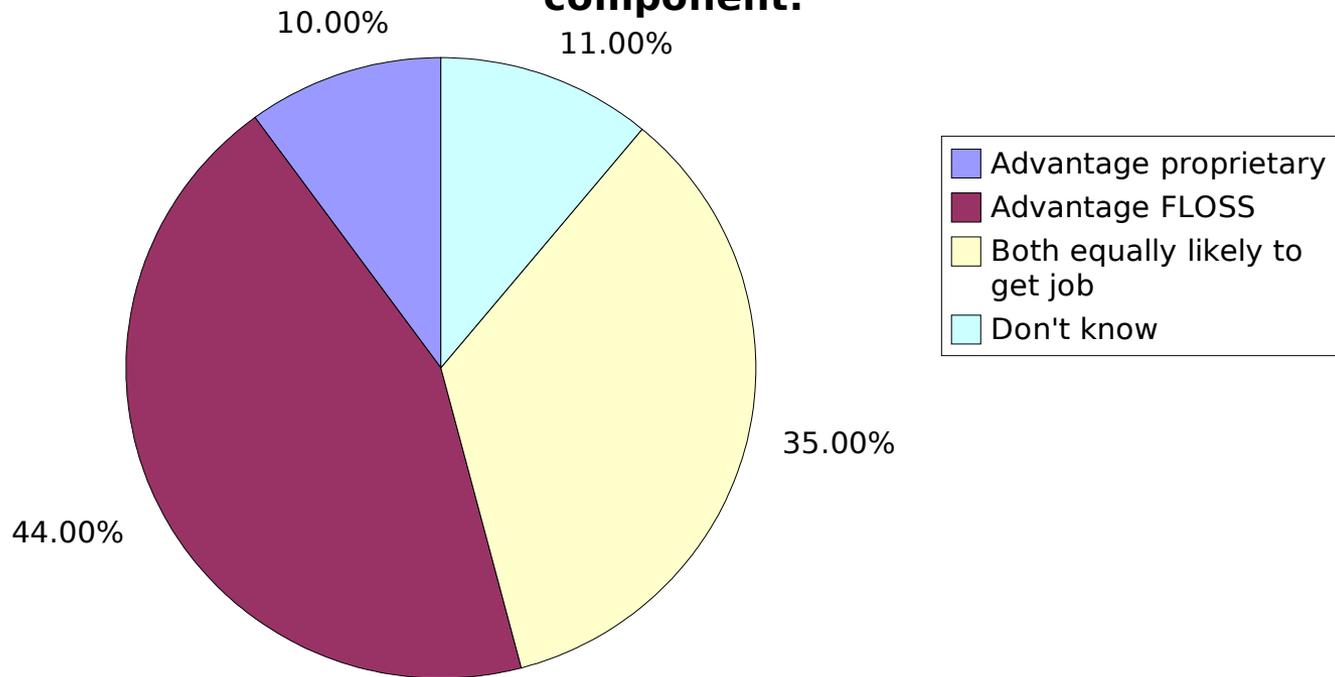
FLOSS skills valued on the job market

The employers' perspective: (n=114)

Crosstab

% within How do you assess the role of FLOSS within your organisation?

		How do you assess the role of FLOSS within your organisation?		
		important	hardly important	Total
Preference for hiring person with proprietary software skills (A) or with FLOSS skills (B)	Person A (proprietary)	3.6%	22.6%	8.8%
	Person B (FLOSS)	60.2%	12.9%	47.4%
	These differences would not influence my preference	32.5%	51.6%	37.7%
	I don't know	3.6%	12.9%	6.1%
Total		100.0%	100.0%	100.0%

FLOSS skills valued on the job market

Earning differences on the job market for past developer of FLOSS software component vs past developer of proprietary software component (regardless of whether job is with proprietary software or FLOSS):

28.00%

33.00%

9.00%

30.00%

- Proprietary experience pays more
- FLOSS experience pays more
- Both equally paid
- Don't know
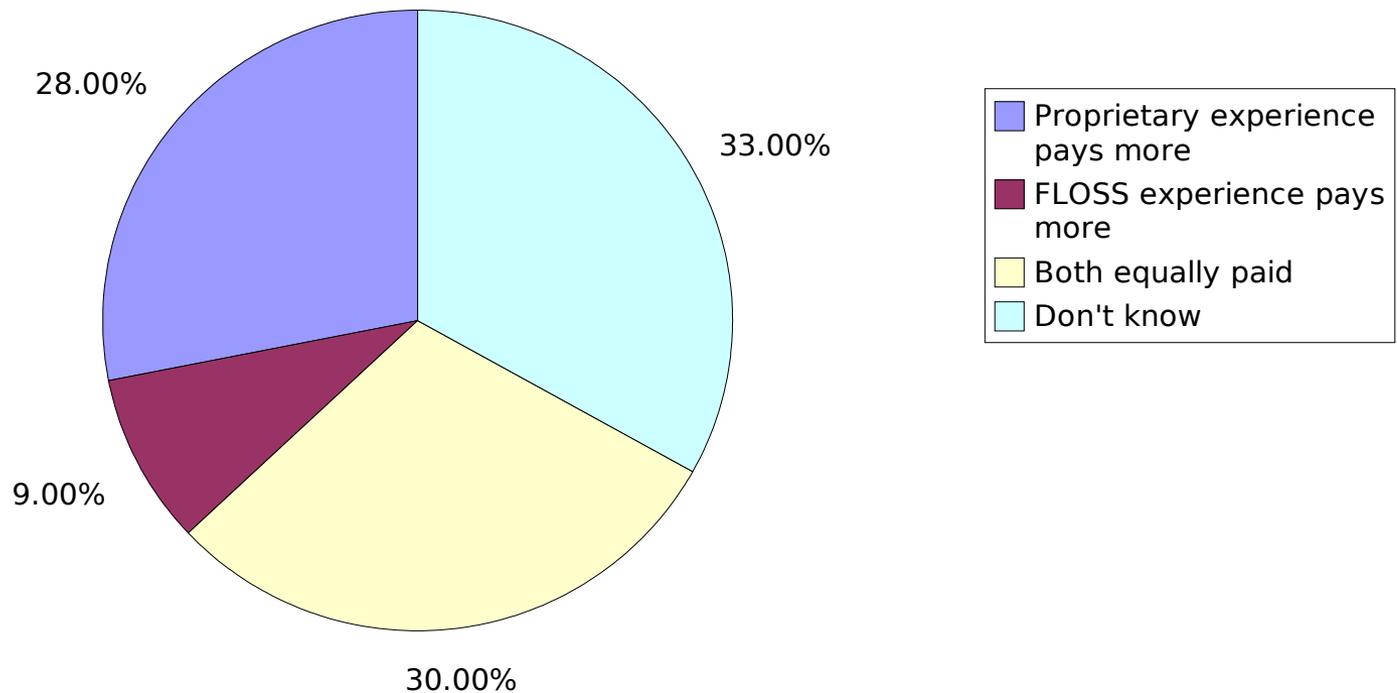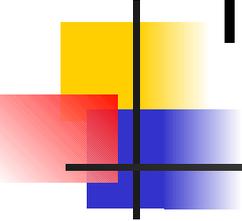
In conclusion

Skills are learnt in FLOSS community

Learnt skills not just technical, indeed legal skills are learnt "a lot"

Those with prior skills learn new ones, especially relating to bug-fixing, writing reusable code

Wide variety of learning strategies

Even those taking formal courses rate it relatively poorly as a learning environment
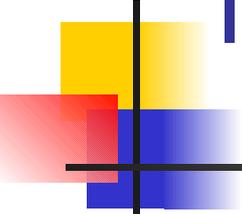
In conclusion

Several skills are learnt better than in formal courses (learn-by-doing skills – reusable code, bug-fixing, teamwork and coordination)

Proven FLOSS experience can compensate for lack of formal degrees in order to get a job

But developers feel those with proprietary experience often get paid more

Employers seem to agree, though awareness of FLOSS among employers increases their perceived value of skills learnt from FLOSS.
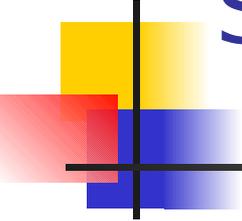
Skills and economic growth

Skills development: "the ability to create"
FLOSS is a training environment that increases the earning capacity of community participants without any explicit investment in training: a novel form of technology transfer

Economic growth: "ability to add value"
FLOSS allows local entrepreneurs to provide a greater share of total value added, thus retaining a greater share of profits within the local economy
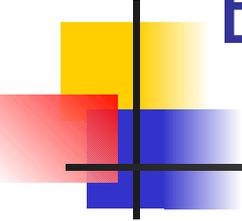
Building *local* ICT competencies

Be passive users of "black-box" software or active participants in global ICT?

Being active requires being able to create, locally – and choose with the least barriers the level of creativity

Skills development requires access to the ability to create – you don't have to be a programmer, but you *should* have the choice.

Relative local value addition is much higher with free software, as compared to proprietary (where the vendor controls and provides the most value)
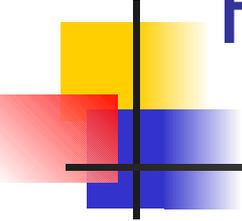
FLOSSWorld: global research

EU-funded project to conduct similar research, on skills, employment, education, government use of FLOSS in non-EU countries

Led by MERIT, Netherlands

Funded partners in Bulgaria, China, Croatia, India, Malaysia, South Africa and Argentina (USUARIA + Fundacion Via Libre)

May 2005 to April 2007
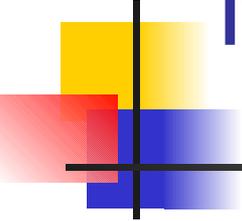
More information

FLOSSPOLS: http://flosspols.org

FLOSS Project report (2002)
http://flossproject.org/report/

FLOSSWorld: http://flossworld.org