

# Economics of FLOSS business models

---

Karsten Gerloff  
gerloff@merit.unu.edu

United Nations University/MERIT  
Maastricht, The Netherlands

# Outline

---

- Size of the FLOSS economy
- FLOSS business models
  - “deep” support: retaining value locally
  - service, customisation
- Skills and economic growth

# FLOSS economy: value

---

- Existing FLOSS code – if a company were to recreate it, this would cost:
  - Euro 12 billion (substitution cost, till 2005)
  - 163 thousand person-years
  - Euro 100 billion (till 2010)

Doubling in size every 18-24 months

- Actual investment by firms in code development: at least Euro 1.2 billion

# FLOSS economy: jobs

---

- Only 7% of programmers in the US work in packaged software companies
- 30% work in sectors producing mainly custom software / integration / support
- Almost 60% work in the “user sector” - finance, government, manufacturing, retail, etc

# FLOSS economy: spending

---

- 16% of software spending in the US is on packaged proprietary software (19% in EU)
- >50% is in-house software development (30% in EU)
- Rest is custom software

# FLOSS economy: summary

---

- *Today's* economics of software: most people (and firms) who write software do *not* make money by selling software, but by selling their time!
- This is the economics of FLOSS, not the economics of proprietary packaged software

# Business models: 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.

# Business models: 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 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.

# Skills: Informal apprenticeships

---

- FLOSS as informal training environment
  - a form of technology transfer
- skills development is no.1 reason for developers to participate in FLOSS communities
  - technical skills
  - management skills
- often better learned than in formal courses, often lead to jobs

# 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

- **Develop skills, not customers**

Proprietary software is a black box. Working with it doesn't develop skills or build businesses – it only develops customers

# Some business aspects

---

- networks
  - share your marketing costs
  - reference each other
- public procurement
  - specify functions, not products
  - demand your freedoms

# Summary

---

- most people make their money not by selling software, but by selling their time
- FLOSS businesses add local value
- FLOSS raises local skill levels

# FLOSSInclude

---

- how to use FLOSS as a tools for social and economic development?
- how to increase use, development, deployment of FLOSS in Africa, Asia and Latin America
- building networks: please get in touch!
  - UWC
  - AITI
  - UNU-MERIT

# Thank you!

---

Karsten Gerloff  
gerloff@merit.unu.edu