THE UNIVERSITY of York

What have you designed today?

Software Quality in Open Source Projects

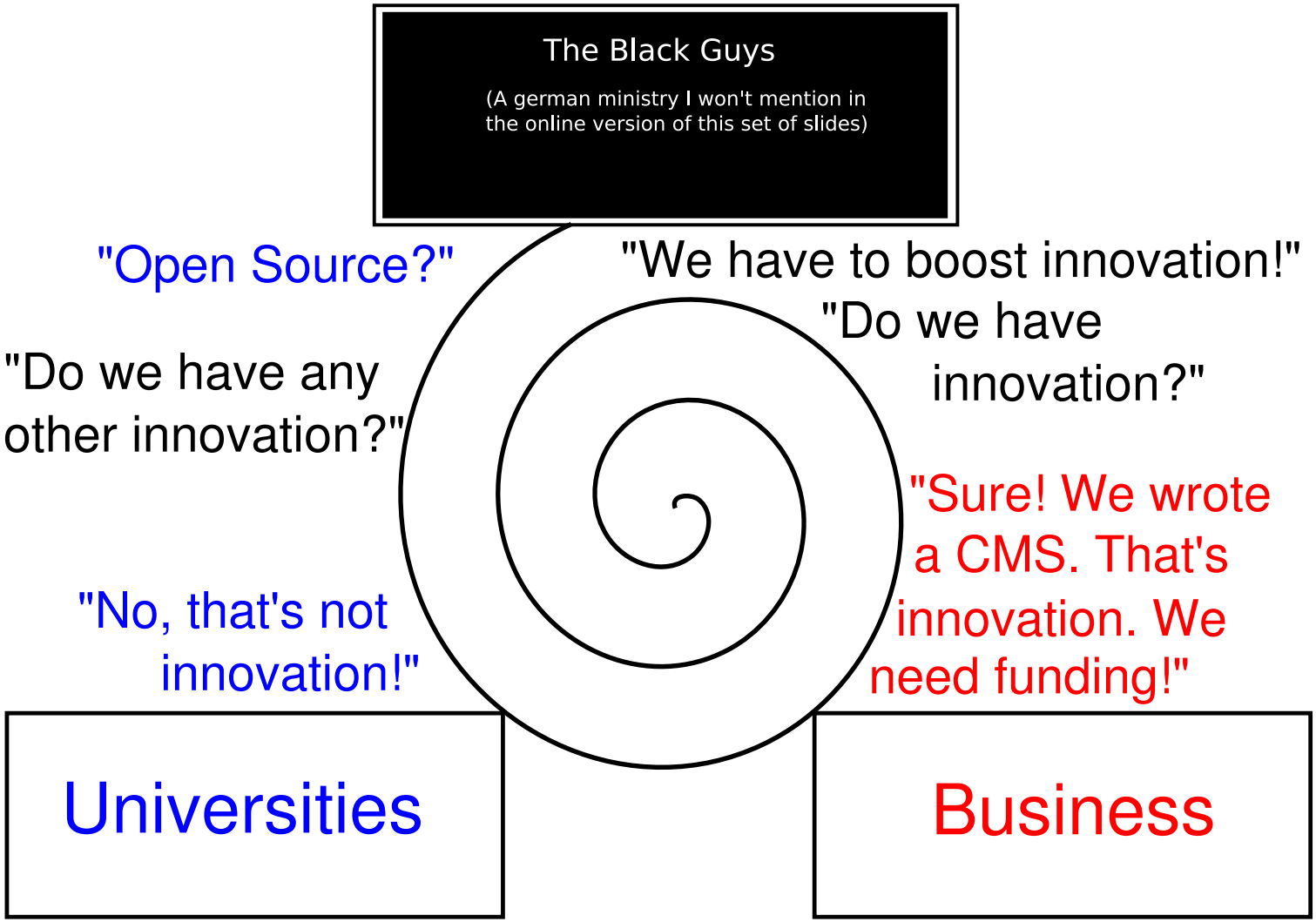
Jan Tobias Mühlberg
<muehlber@cs.york.ac.uk>

York, November 30, 2005

Overview

1. Why and for whom is it interesting?
2. What did we do?
3. OSS – Myth and Reality
4. Software Quality in OSS
5. The Problem: Code Maintainability!
6. Perspectives

Why and for whom. . .



Why and for whom. . .

M: "Is OSS good or bad for innovation?"

U: "What is innovation?"

M: "We don't care. Tell us whether it is good!"

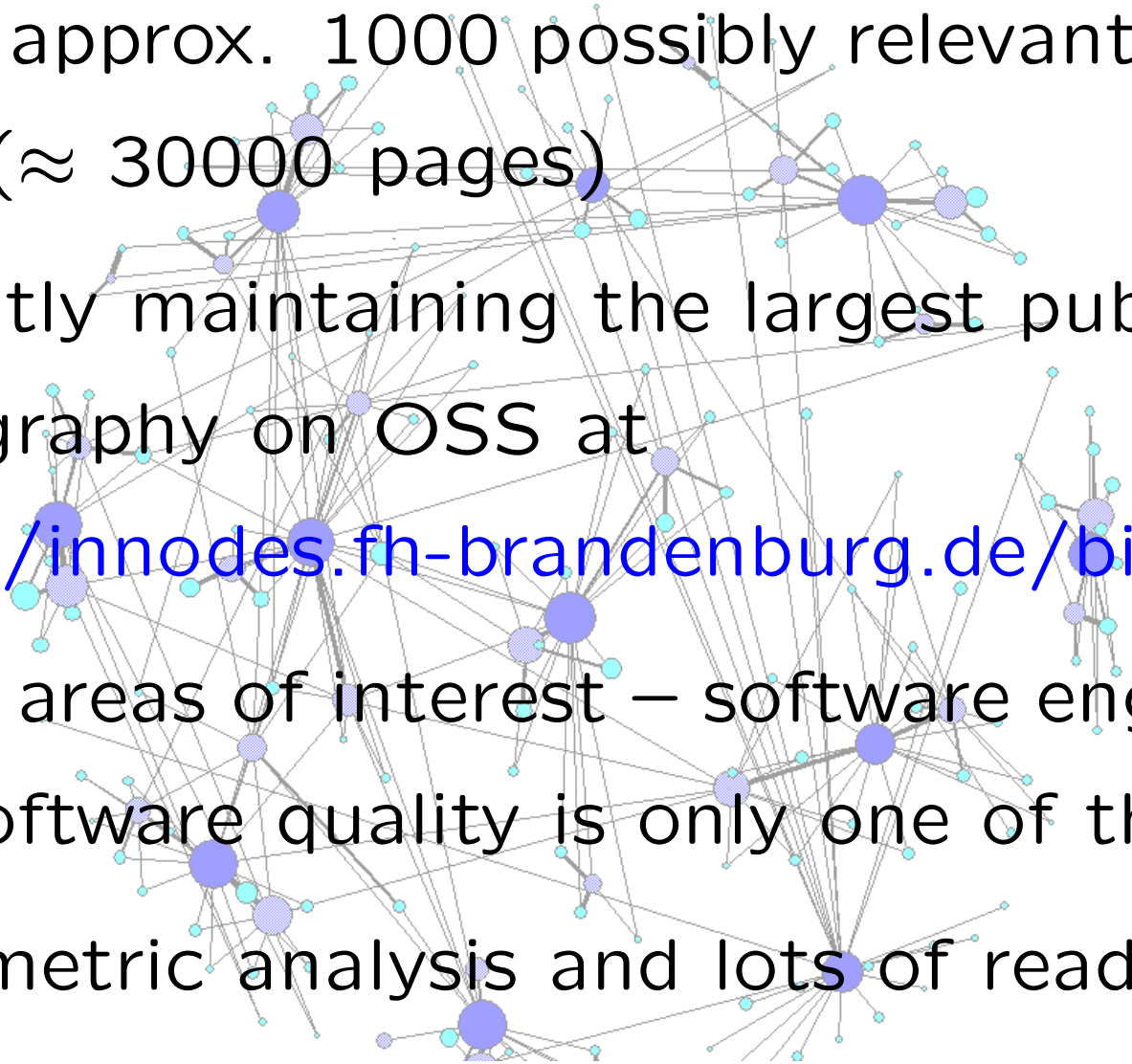
U: "There are plenty of studies available. . ."

M: "We don't care. Just tell us. . ."

What did we do?

- start a meta-study on "open source software and innovation"
- gather a few articles...
- write a nice summary...
- be done...?

What did we do?



- found approx. 1000 possibly relevant publications ($\approx$ 30000 pages)
- currently maintaining the largest public bibliography on OSS at http://innodes.fh-brandenburg.de/bibdb/
- define areas of interest – software engineering and software quality is only one of them
- bibliometric analysis and lots of reading

OSS – Myth and Reality

- common beliefs in OSS software development:
  - completely chaotic process
  - thousands of developers take a close look at every single line of code → no errors
  - software engineering techniques aren't used and design questions aren't asked
  - therefore, fast development and reliable results

OSS – Myth and Reality

- reality is different:
  - development is only partly chaotic, more like the bazaar around a cathedral
  - in general agile (and distributed) software development
  - "given enough eyeballs. . ." – sometimes
  - CSS is a mechanism of commerce; OSS tends to be a mechanism of science and philosophy

Software Quality in OSS

- What's "quality"?
  - reliability
  - maintainability
- How is QA done in OSS projects?
  - strong rules enforced by project leaders
  - external QA by some company
- Does it work?
  - In general? No.
  - For common projects? Yes.

Software Quality in OSS

- the main problems:
  - It's vital for an OSS project to have users. There's no other way of attracting developers.
  - Applying agile software development doesn't mean to do test-driven development.
  - A software architecture that was nice for 10 developers may not scale well for 1000.

Code Maintainability?!

- it's nice because it can be measured:
  1. size of the code (LOC)
  2. complexity (cyclomatic complexity)
  3. self-descriptiveness (ratio of comments)
  4. common (global) coupling

- Most project analysed perform great for 1, 2 and 3, but fail for 4. ([Schach et al., 2002], [Samoladas et al., 2004])

Code Maintainability?!

- Linux as an example. . .
- linear dependency between LOC and version
  - no surprise: successive versions provide additional functionality
  - indicates a good basic design: only small code additions are necessary for new interfaces and functionality

Code Maintainability?!

- Linux as an example. . .
- exponential growth of common coupling
  - basic design is badly followed or does not scale with number of developers and size of the project
  - refactoring isn't easy, even for OSS
  - design questions are rarely asked in later states of the development process

Code Maintainability?!

"Great programmers can work effectively without explicit design or coordination, but when average programmers try to emulate that improvisation, the results are rarely pretty."

[Wilson, 1999]

Code Maintainability?!

- sociological reasons:
  - fixing subtle bugs in a weird piece of code at 2 AM is a story worth telling...
  - design questions in general aren't
  - why not asking more often:

    "What have you designed today?"

Perspectives

- OSS has to deal with basically the same problems as CSS
- common OSS projects have a high level of quality – depending on what properties are analysed, they perform equal or sometimes better than CSS
- OSS should strive for greater code maintainability

Perspectives

- the meta-study will be released in March 2006
- there will be a couple of detailed publications in English and German
- the public bibliography will be hopefully maintained – any help is welcome
- many thanks to http://innodes.fh-brandenburg.de/

Thank you!