
Self-Organization in Open Source Development— Discussion

Carliss Baldwin

Von Krogh et. al.

- ◆ Phase 1 — self-determination
 - Intrinsic and extrinsic
 - No magic profile
- ◆ Phase 2 — contextual factors “institutions”
 - Contingency theory and crowding out

Alasdair MacIntyre

- ◆ Not self determination
 - Motives \implies Actions

- ◆ But Aristotelian virtue
 - Actions \implies Learning \implies Motives
 - in a about reinforce
 - Social Virtue Right
 - Context Action

Besten-Dalle-Galia

- ◆ Stigmergic agenda
 - “The code signals what it needs.”

- ◆ A “soft” is not a firm

Empirical test

- ◆ Max # of maintainers in a given file in a given month
- ◆ Correlated with:
 - Lines of code
 - “Youth”
 - 2 measures of complexity
 - “Modularity” (# functions in the file)

Issues

- ◆ What theory of collaboration points to this dependent variable?
- ◆ Max functions are erratic measures
 - Robustness checks
- ◆ Definition of modularity not consistent with literature (Suh, Ulrich)
- ◆ Measures of complexity are “inside the file”
 - File dependency with other files is another source of complexity

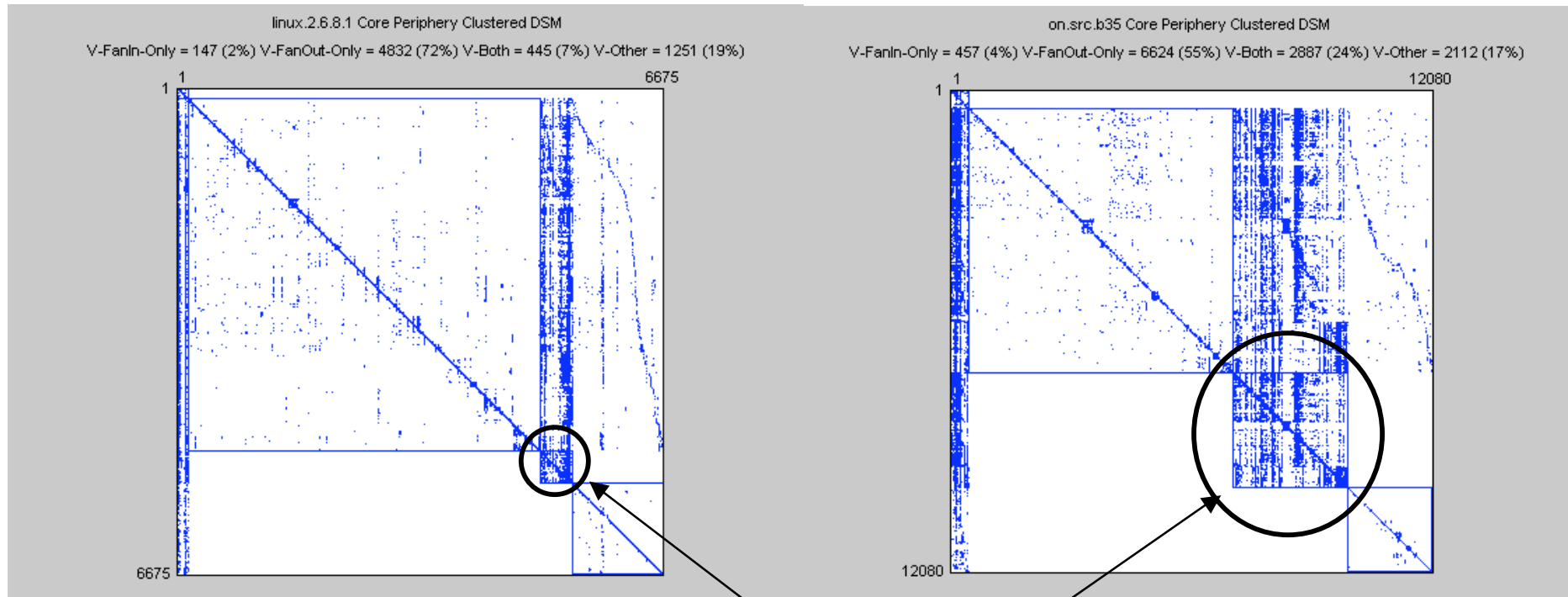
Ghosh-David

- ◆ Linux
- ◆ Authorship network
 - Files linked by common authors
- ◆ Function call network
 - Files linked when one calls another
- ◆ Are these networks correlated
- ◆ “Mirroring hypothesis”

Power of Visualization

Linux

Open Solaris



The "Core"

Problem of Aggregation

- ◆ Highly aggregated data
 - Functions => Files => Modules
- ◆ Number of observations goes down dramatically as you aggregate
- ◆ Files => Modules
 - Unnecessary loss of power in the test

Technological determinism is the common theme

- ◆ Properties of the technology affect social/organizational structure
- ◆ Standards of code excellence => Motives
 - von Krogh et. al.
- ◆ Complexity => Number of maintainers per file
 - Besten-Dalle-Galia
- ◆ Code dependencies => Authorship linkages
 - Ghosh-David