

Master Defense
Moritz Neeb
9.02.17

Tracking Innovation in Open Source Software Projects

Freie Universität



Berlin



Innovations in Open Source Software

- OSS is under a license that allows redistribution and modification
- OSS is developed in an open, “asynchronous” way by developers around the world
- Innovation Introduction in OSS is an orchestrated (i.e. active) change to the development process

Research Goal

Motivation

- The field of OSS is not explored enough compared to its relevance
- Understanding OSS developers can be extrapolated
- Knowledge about Innovation helps new OSS projects

Research Questions

- How are innovation episodes of 2007 continued during 2008-2016?
- What constraints are given for the search for these episodes?
- How can the new data be integrated into existing theory?

Do continuations of 2007 episodes exist?

- Around 130 “innovation episodes” were identified for the year 2007



Roadmap

- 1) Background Information
- 2) Search Process
- 3) Results on Concept Level
- 4) Conclusion
- 5) Discussion

Background

Definitions

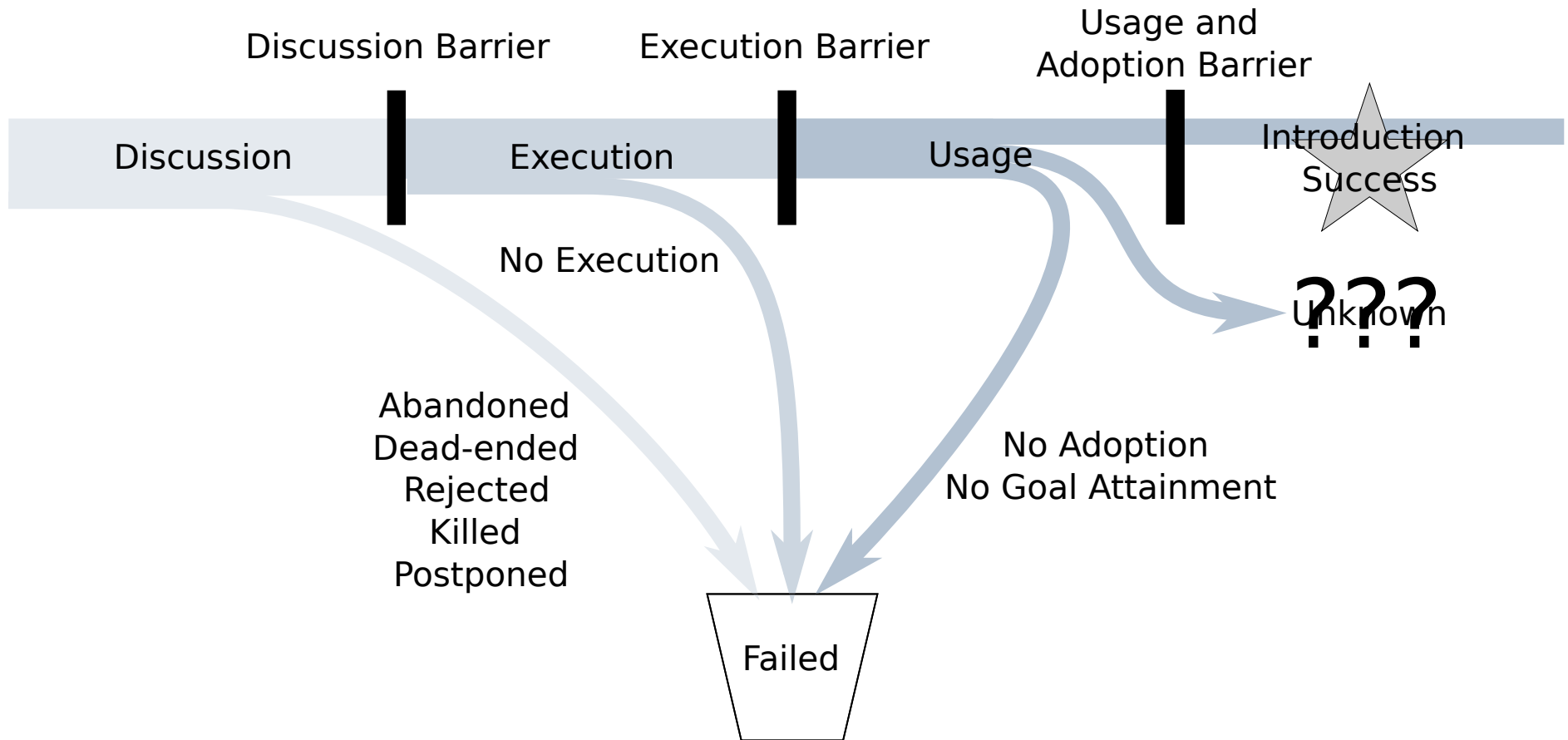
- Innovation: Either the process/system/tool under discussion or the one in place.
- Episode: An attempt by one or more participants to change an innovation

Success

An innovation is successfully introduced, when:

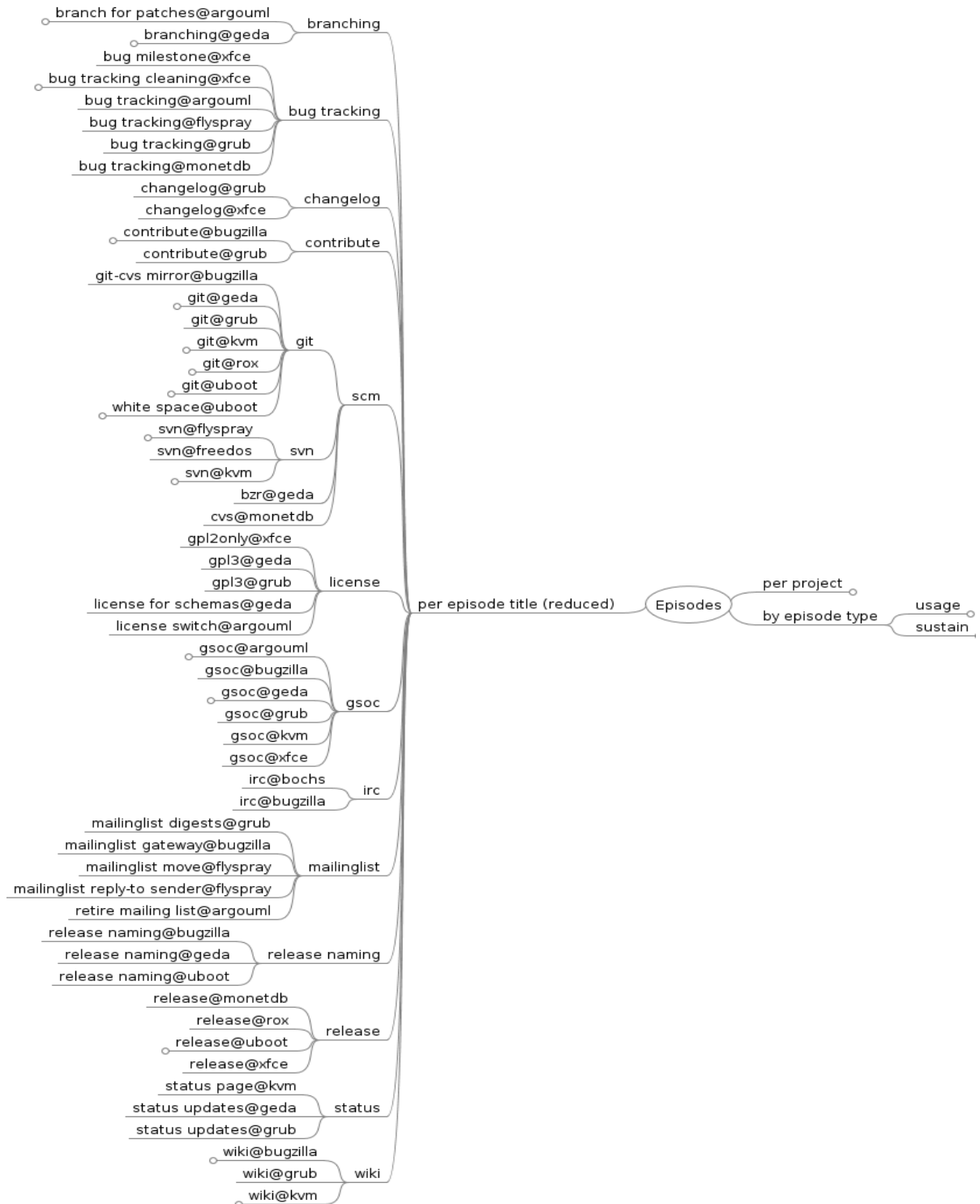
- 1) it is used on a routine basis and it has solved the problem it was designed to solve, or
- 2) it attained the goal it was designed to attain.

Introduction Lifecycle



Source: Dissertation C. Özbek

Episode Topics

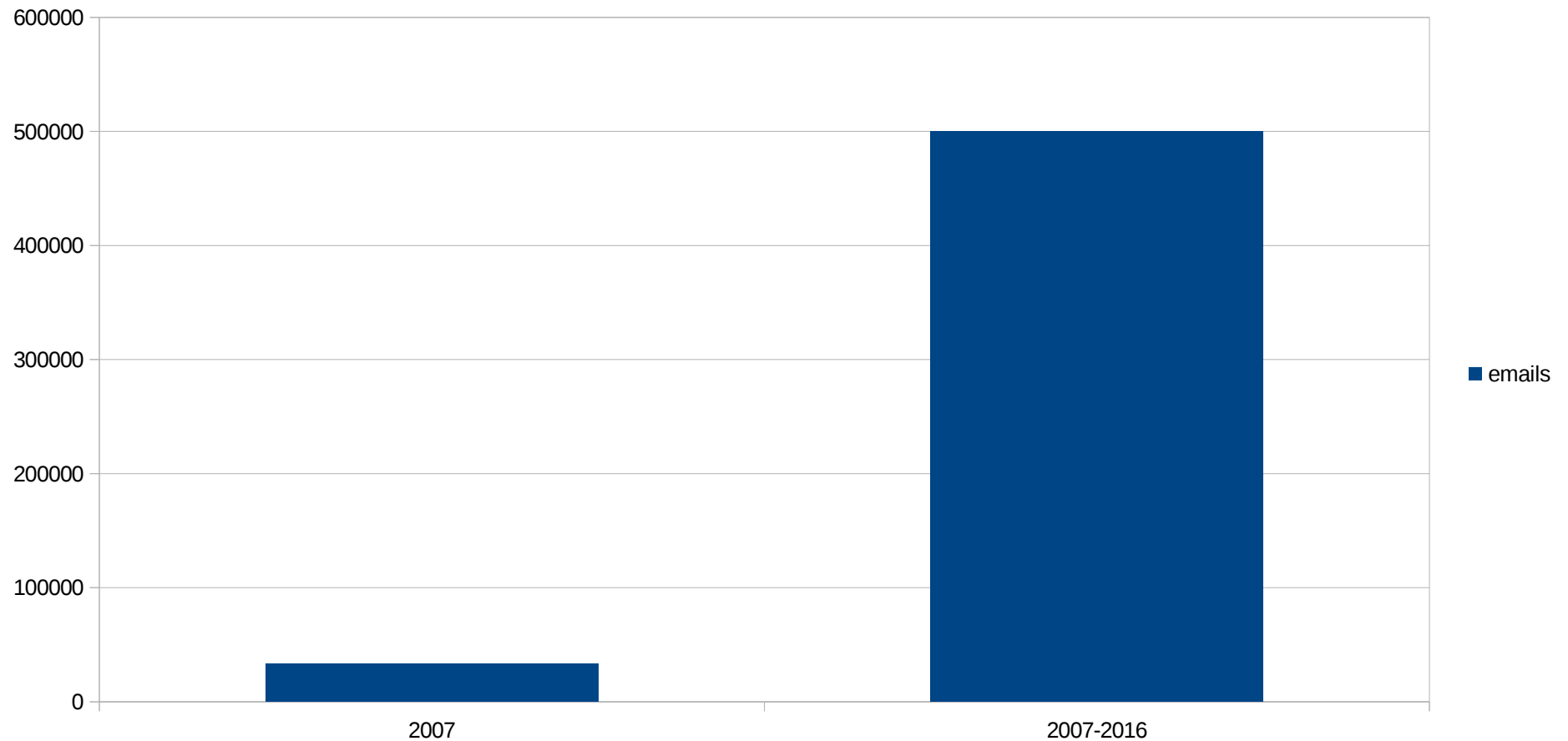


Data that was used

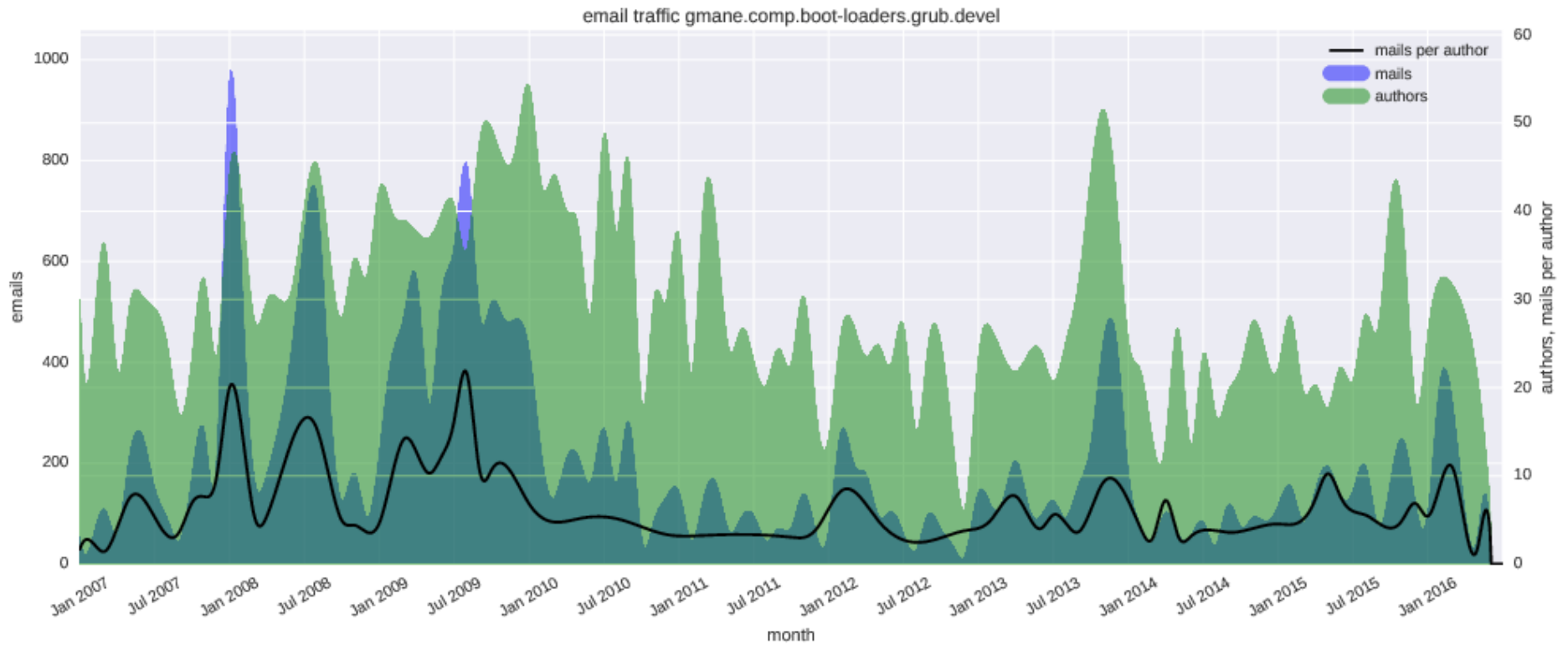
- Mailinglist data
- Annotated mails (coding)
- Second channel information
 - code repositories
 - projects website
 - wikis

The Search Process

Motivation for a search method



emails @ GRUB



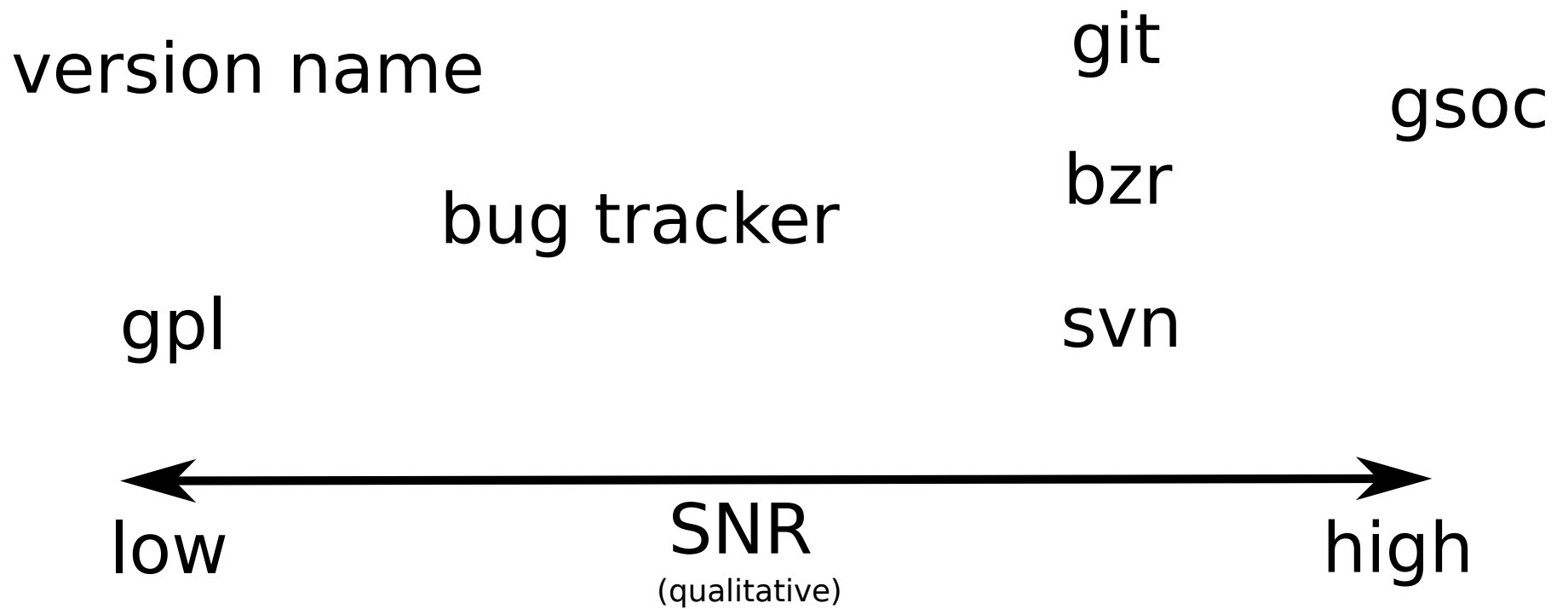
Keyword Search

- Signal-to-noise ratio
- Difficulty to find innovations
- Evolution of SCM innovation

Identified continuations per innovation

- SCM – all (8)
- GSoC – all (6)
- Bug Tracking – 4/6
- License – 2 / 5
- Version Naming – 1 / 3

Signal-to-noise ratio of different innovations



Finding continuations

- Easy to check success, e.g. via
 - “git still in use?”
 - “license still the same?”
- Hard to find unprecedented problems, e.g.
 - “problems with license/git?”
- But with evidence, hit-rate is very high, e.g.
 - “there was a change of version number scheme since 2007 – when?”

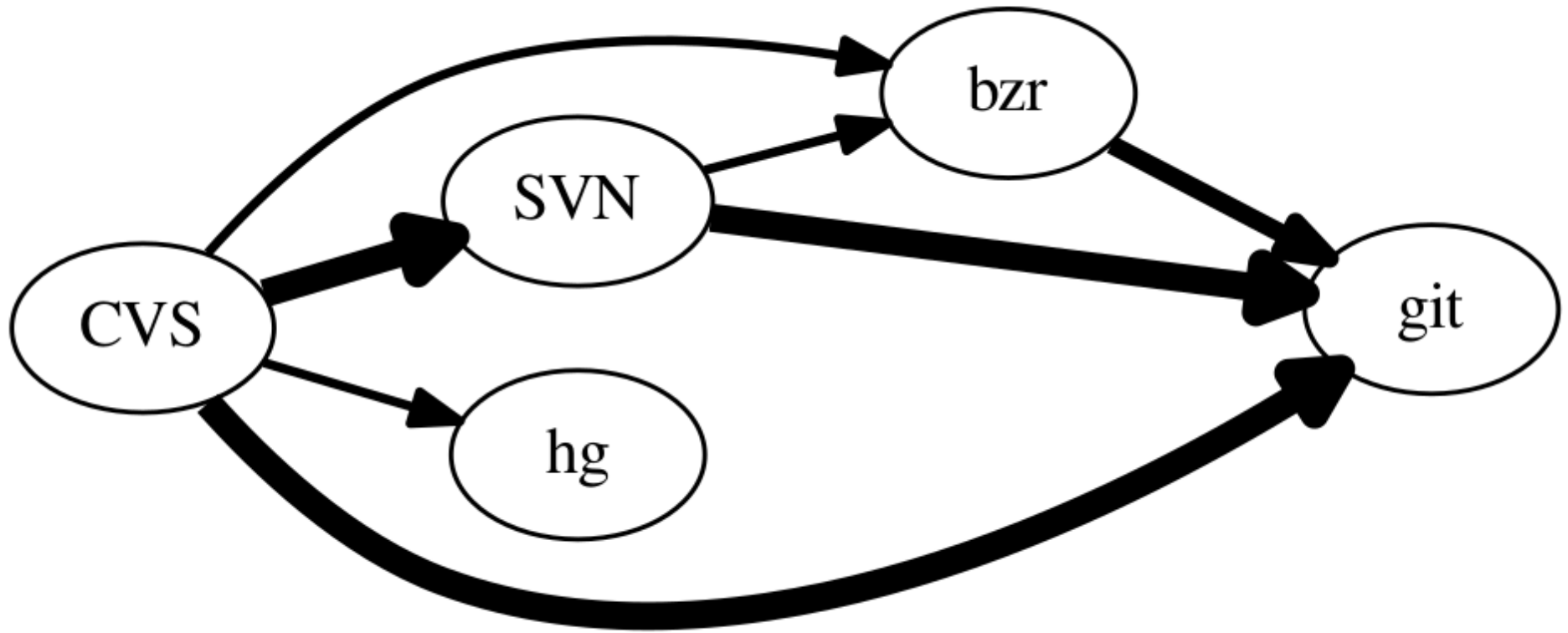
More evidence: developer activity



SCM switches

	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016
ArgoUML					SVN					
Bochs		CVS					SVN			
Bugzilla		CVS				bzr			git	
Flyspray			SVN						git	
FreeDOS	CVS					SVN				
gEDA	CVS					git				
GRUB	CVS		SVN			bzr			git	
KVM	SVN					git				
MonetDB		CVS						Mercurial		
ROX	CVS					git				
RequestTracker		SVN						git		
U-Boot						git				
Xfce		SVN						git		

git is becoming popular



The search process

Summary

- Keyword search works fine for most of the relevant cases
- Evidence of change helps a lot
 - via emails: e.g. author activity
 - second channel: e.g. project's infrastructure
- Pressure from “outside” helps a lot for innovations

Results on the conceptual level

Continuation “capability”

- There were two groups of innovation continuations:
 - core episodes
 - peripheral episodes
- These groups have very different “barrier-heights”
- The visibility on the mailinglist is also different

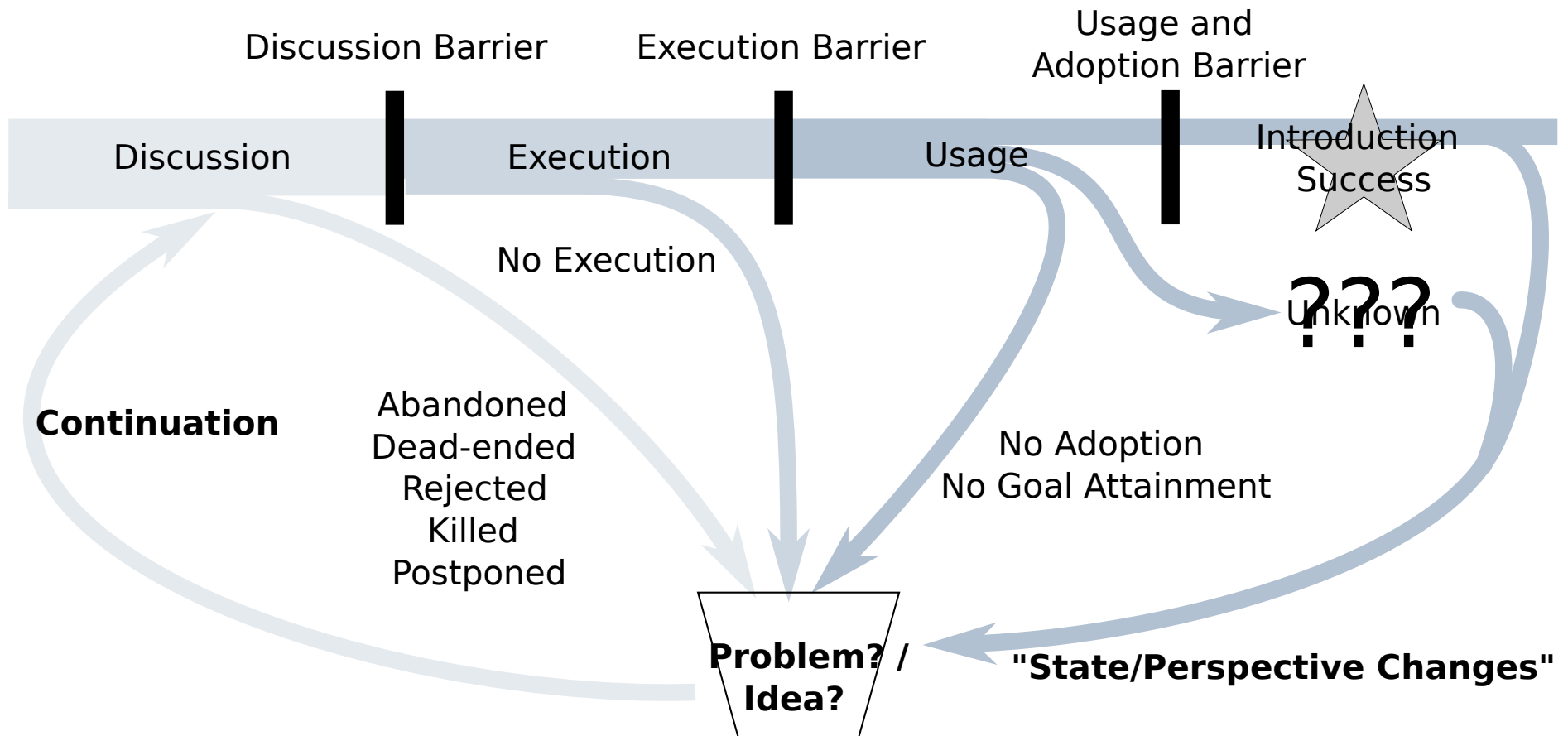
Core Episodes

- Innovation is used on a daily basis
- If there is a problem, it will be discussed (urgency high)
- high numbers of people affected
 - continuation is likely

Peripheral Episodes

- “One-shots” or minor changes
- Can still be important for progression
- Continuation is unlikely (did not find any)
- Probably hard to find on their own without knowledge of their existence

Innovation Lifecycle is a circle indeed



modified version of
source: Dissertation C. Özbek

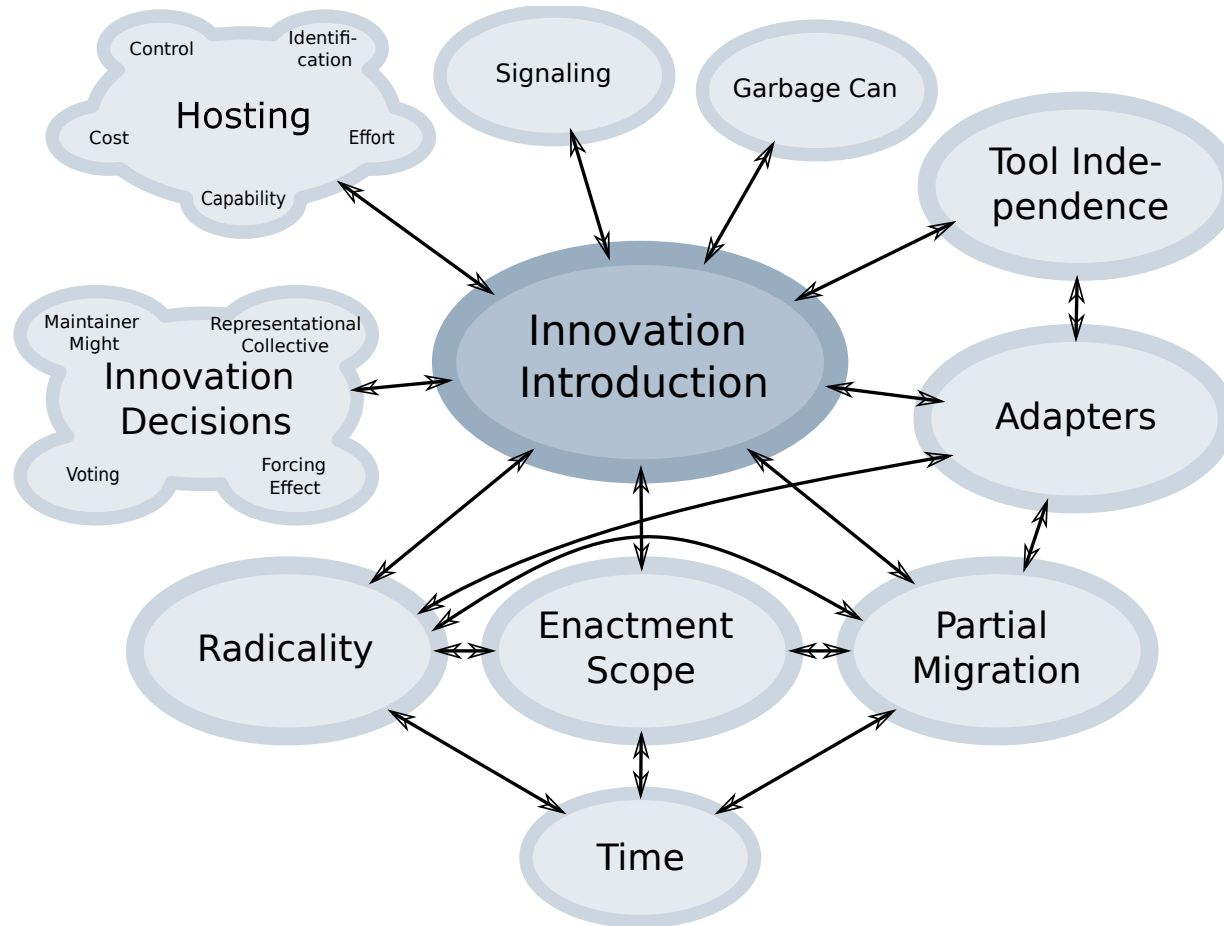
Continuation: From problem to solution

- to let this happen, an innovator has to step up again
- evidence of repeated innovation attempts
 - autotools @ Xfce
 - SCM @ GRUB

State change / perspective change: From a solution to a problem

- A successful episode can turn into a problem or a new idea to improve the situation
- evidence:
 - bug tracker @ GRUB (process is “rotten”)
 - SCM @ GRUB (data loss motivates innovators)

Innovation Concepts



Source: Dissertation C. Özbek

Results summary

- Keyword search on mailinglists reveals enough material to evaluate the situation
- SCM management systems progression
- Lifecycle revision
- “Concept” validation

Conclusion

- The high-level “innovation processes” have not changed much
- Mailinglist communication reveals the controversial topics, but “success” is difficult to see (i.e.: silence means a good thing)
- The “concepts” of 2007 are still valid, but might need a restructuring
- The “outside progression” keeps projects evolving

Thanks for your attention.

Discussion

- Firefox WebExtensions as outside pressure?
- Are peripheral episodes important?
- Relationship to Q&U research:
the projects' “develop-ability”?
- Your questions?