

The Evolution of FreeBSD Governance

Marshall Kirk McKusick
mckusick@mckusick.com
<http://www.mckusick.com>

Based on a paper written with

Benno Rice

The FreeBSD Journal, 4, 5.
September 2017, pp 26–31

EuroBSD Conference
Bucharest, Romania

22–23 September 2018

Copyright 2018 by Marshall Kirk McKusick

Background and Introduction

The Berkeley Software Distribution (BSD) started in 1977 as a project of Bill Joy at the University of California at Berkeley

Became a full distribution with the release of 3BSD for the VAX in 1979

Use of source-code control (SCCS) started in 1980

A decade of releases were managed by the Computer Systems Research Group (CSRG), a four-person development team.

Nearly-full open-source release of Net/2 in 1991 followed by 4.4BSD-Lite in 1994

The Formation of the FreeBSD Project

FreeBSD was named on June 19, 1993 and was derived from Bill Jolitz version of 4.4BSD-Lite for the Intel 386

Managed under the CVS source-code control system

Core team (with lifetime terms) created to decide who should be allowed to commit

Initially distributed by Walnut Creek CDROM

Separated into base system and ports to keep base system size manageable

GNATS was brought up manage bug reports

The FreeBSD Project Moves into Companies

Yahoo ran entirely on FreeBSD and agreed to host CVS and distribution

Justin Gibbs starts the FreeBSD Foundation aiming to provide the FreeBSD infrastructure

Deadwood and apathy in the 20-member core team lead to creating bylaws that set up a 9-member elected core team

First elected core team in 2000 with few carryovers from old core team

Core-Managed Teams

Core appoints members to other teams that manage different aspects of FreeBSD

- Port manager team oversees 217 ports committers
- Documentation team oversees 126 document committers
- Security officer handles security issues, alerts, and updates
- System administration team oversees FreeBSD infrastructure
- Release engineering team manages the FreeBSD releases
- Quality assurance team run continuous integration builds and expand regression tests

The FreeBSD Project Today

Initially communication was on a single mailing list

Broken up into many area-specific lists: architecture, network, filesystems, etc

Cross-area collaboration became necessary

- Initially done through bug tracking (initially GNATS, later Bugzilla)
- Moved to Phabricator in 2014 to expand discussion to non-committers

FreeBSD Source-Code Control

Project started on CVS

After much debate moved to Subversion in 2000

On-going debate on switching to Git

- Many programmers know and use Git
- For several years all changes in Subversion pushed to GitHub
- Upstreaming from GitHub requires going through Phabricator

FreeBSD Workflow

Committers are in up to three groups

- Documentation: All FreeBSD documentation
- Ports: all of the ports available on FreeBSD
- Source: the base system

Changes that may affect a kernel interface or operational semantics or that may have other noticable or impactful effects are put up in Phabricator for review

To reduce instances of an uncompileable or broken system, all non-trivial changes require a review

FreeBSD Commit Tags

Each commit includes tags that include

- which other project member had reviewed the changes
- the bug report number from which it was identified
- the Phabricator thread on which it was discussed
- when to send a reminder to merge the change to older stable/release branches
- if relevant, the sponsoring organization
- if appropriate, an acknowledgement that the commit fixes an earlier mistake made by the committer (the “Pointy Hat” tag)

Guidelines on How to Work and Play Together

Project started out with behavior guidelines based around commit misbehaviour with rules for core to suspend commit privileges

Worked well until gamer-gate blew up

A Code of Conduct was drawn up by well-meaning but inexperienced folks (feedback: insufficient)

It was replaced by a Code of Conduct drawn up by folks with more experience (feedback: over-bearing)

A third revision of the Code of Conduct is in the works

FreeBSD Committer Turnover

Open-source projects must eliminate deadwood

- drop commit privileges after one year of inactivity
- allow auto-reinstatement for up to two years
- accelerated process for reinstatement after more than two years

FreeBSD Recruitment

To maintain viability, must bring in new developers

- contact via university course
- working for a company using FreeBSD
- involvement in FreeBSD Google Summer of Code project
- discovering the project at a conference or through social media

Project must be welcoming and easily entered

- Provide an easy path to connect with the project
- Provide mentors to help get new committers up to speed

Committer Statistics

Active Source Committers = 223

Active Ports Committers = 219

Active Document Committers = 113

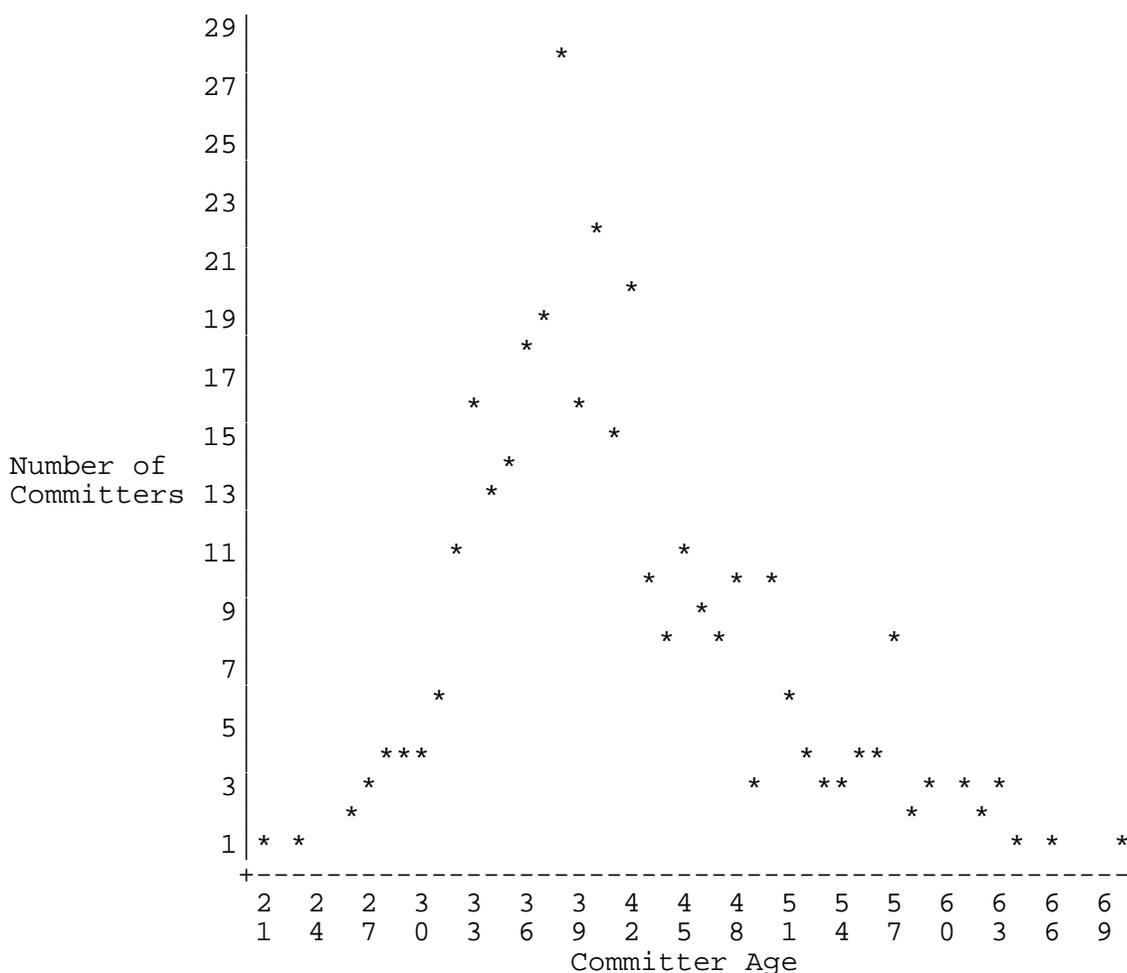
Total Active Committers = 379

Active Committers in calendar.freebsd = 333

Missing Committers in calendar.freebsd = 46

Retired Committers in calendar.freebsd = 185

Ages median 39, average 40



FreeBSD Development Model

Small changes are easy to identify, reach consensus on, and implement

Large or wide-ranging changes are hard

Switch from CVS to Subversion took 9 years

- Endless discussions dying out without resolution
- Finally accomplished by Peter Wemm by just doing it

Change from single-threaded to SMP kernel

- Debate between Solaris-style locking or Amiga-style lockless
- Ultimately FreeBSD used locking and Matthew Dillon left the project to start Dragonfly using lockless

Improving the Big Project Development Model

Recently created the “FreeBSD Community Process” modelled on the “Python Enhancement Process”

- Major changes are described in a document
- reviewed by FreeBSD Community Process editor
- published for discussion
- document is edited to reflect discussion conclusions
- when core team feels discussion has gone on long enough they vote on accepting the proposal
- once accepted the proposal gets implemented with usual review in Phabricator

FreeBSD Core Team Interaction with the FreeBSD Committers

Historically all core communications were private with only a monthly report made public

Monthly report discussed only actions taken

Recently a monthly agenda of discussion topics is released before each meeting allowing committers to provide input

Contemplating allowing committers to join the video conference (except for personnel agenda topics)

FreeBSD Security Team

Initially the project had a single security officer

In 2002 expanded to a security team

Moving to have a full-time person to be security officer

Security officer will oversee a pool of folks doing work necessary to address security issues

Summary

Evolution of governance model for leadership

- started with Benevolent Dictator For Life
- moved to core team for life
- now is elected core team

Four major changes in leadership

- each allowed the project to move forward and tackle new problems
- helped the project avoid aging out (median age remains in the mid to high 30's)
- retained enough wisdom to avoid rookie mistakes and rathole dives

FreeBSD Foundation has provided needed resources and stability

Conclusions

Governance is mundane

- Too much is stifling
- Too little and the project goes off the rails or collapses from lack of infrastructure
- Governance requires constant tuning to keep the right balance

Questions

FreeBSD home

<http://www.freebsd.org>

FreeBSD Foundation

<http://www.freebsd.foundation.org/>

Marshall Kirk McKusick

mckusick@mckusick.com

The End



May the Source Be With You!