

# How the FreeBSD Project Works

Daniel Geržo

`danger@FreeBSD.org`,

slide materials provided by:

Robert Watson – `rwatson@FreeBSD.org`

Philip Paeps – `philip@FreeBSD.org`



Eötvös Loránd University,

Budapest, Hungary

November 20, 2010

# Introduction to FreeBSD



- ▶ Open source BSD UNIX-derived operating system
- ▶ ISP network server platform
  - ▶ Yahoo!, Verio, NY Internet, ISC, ...
- ▶ Appliance/product/embedded OS foundation
  - ▶ NetApp, Avid, Isilon, Nokia, Panasas, Timing Solutions, Juniper, VXWorks, Mac OS X, FreeNAS, ...
- ▶ One of the most successful open source projects
- ▶ Focus on storage, networking, security

# FreeBSD: The Kernel

- ▶ Multi-processing, multi-threaded kernel
- ▶ Support for many popular hardware architectures
  - ▶ Intel/AMD 32/64-bit, sun4v/SPARC64, ARM, PowerPC, MIPS
- ▶ UNIX, POSIX, BSD programming interfaces
- ▶ Multi-protocol network stack
  - ▶ IPv4, IPv6, IPX/SPX, AppleTalk, IPSec, ATM, Bluetooth, 802.11, SCTP, ...
  - ▶ Reference implementation for various protocols
- ▶ Unified, coherent build-system across components
- ▶ Extensive documentation
- ▶ 2.7M lines of C code (FreeBSD 8.1)



# FreeBSD: The Userland

- ▶ Complete, integrated UNIX system
  - ▶ Expected tools are there by default, no extra packages needed
  - ▶ Build-time options to disable tools (for embedded platforms)
- ▶ Kernel and userland maintained by the same people
  - ▶ Userland is always in sync with changes in the kernel
  - ▶ All new kernel features are immediately available in userland
- ▶ Strong focus on consistency and “POLA” (Principle of Least Astonishment)
- ▶ Extensive documentation
- ▶ 4.2M lines of C code (FreeBSD 8.1)



# FreeBSD: Third-Party Applications

- ▶ Maintained collection of ported software – aka. the “Ports Collection” or “Ports Tree”
  - ▶ Flexible, source-based package management framework
  - ▶ Over 22,000 applications available
  - ▶ Not “tagged” to specific releases
  - ▶ Easy to maintain local additional packages
- ▶ Linux binary compatibility layer
  - ▶ Not an emulator!
  - ▶ It is compatible with Linux 2.6
- ▶ Most UNIX/Linux applications will Just Work™



# FreeBSD: Documentation

- ▶ Manual pages considered important
- ▶ Release notes, papers, examples, . . .
- ▶ The FreeBSD Handbook
  - ▶ Full-text “book” available in many formats
  - ▶ Over 1,000 pages
  - ▶ Translated to many languages
- ▶ Other “Handbooks”
  - ▶ FreeBSD Developer’s Handbook
  - ▶ FreeBSD Porter’s Handbook
  - ▶ FreeBSD Architecture Handbook
  - ▶ FreeBSD Documentation Project Primer for New Contributors



# The FreeBSD Project

- ▶ Global online community of developers and users
  - ▶ Central source code repository and revision control
  - ▶ 372 developers, thousands of contributors
  - ▶ Extensive online user community (user groups, advocacy, training, etc.)
- ▶ Liberal Berkeley open source license
  - ▶ Designed to maximize commercial re-use
  - ▶ No requirement that derived works be open source
  - ▶ Extensive use in commercial, research systems

<http://www.FreeBSD.org/>



# The FreeBSD Foundation



- ▶ Non-profit organization based in Boulder, Colorado
  - ▶ Intentionally independent from the FreeBSD Project
  - ▶ Sponsored development
  - ▶ Intellectual property, contracts, licensing, legal
  - ▶ Developer travel grants
  - ▶ Event sponsorship
  - ▶ Hardware purchase
  - ▶ Collaborative research & development agreements
- ▶ Anybody can donate –  
<http://www.FreeBSDFoundation.org/>





# Project Output



- ▶ Kernel, libraries, toolchains, utilities
- ▶ Security advisories and fixes, release engineering
- ▶ Set of ported third-party applications, with pre-compiled binary packages
- ▶ FreeBSD releases
- ▶ Manuals, handbook, web pages, marketing
- ▶ Technical support, debugging, consultancy
- ▶ A variety of user/community events



# Project Input

- ▶ Beer, soda, chocolate, and other vices
- ▶ Donated and sponsored hardware
  - ▶ Especially in racks, with hands
- ▶ Bandwidth in vast and untold quantities
- ▶ Travel grants, salaries, contracts, grants
- ▶ Thanks, user testimonials, good press
- ▶ Yet more bandwidth
- ▶ And beer :-)



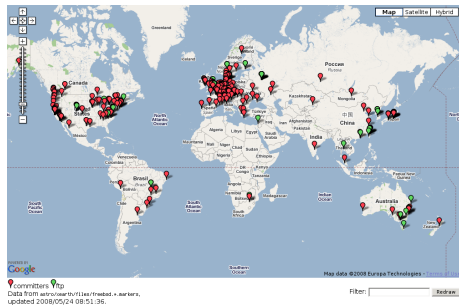
# FreeBSD Committers



- ▶ A committer is someone with SVN or CVS commit rights
- ▶ Selected based on key characteristics
  - ▶ Technical expertise
  - ▶ History of contribution to the FreeBSD Project
  - ▶ Ability to work well in the community
  - ▶ Having made these characteristics obvious!
- ▶ Key concept: mentor
  - ▶ Mentor proposes to the project leaders
  - ▶ Guide through first few months of work



# Who are Committers?



## ▶ Locations

- ▶ 48 countries
- ▶ 6 continents

## ▶ Ages

- ▶ Oldest (documented) committer born 1948
- ▶ Youngest (documented) committer born 1989
- ▶ Mean age 32.5, median age 31, stddev 7.3
- ▶ Professional software developers, hobbyists, consultants, university professors, students, . . .



# Groups and Sub-Projects (1)

## ▶ Developers

- ▶ Source Developers
- ▶ Core Team
- ▶ Core Team Secretary
- ▶ Release Engineering Team
- ▶ Release Engineering Build Teams
- ▶ Security Officer
- ▶ Security Team
- ▶ Ports Team
- ▶ Port Managers
- ▶ Port Manager Secretary
- ▶ Documentation Engineering Team
- ▶ Documentation Team
- ▶ Vendor Relations Team

## ▶ Administrative

- ▶ Foundation Board of Directors
- ▶ Foundation Operations Manager
- ▶ FreeBSD.org Administrators
- ▶ FreeBSD.org Webmaster
- ▶ Sentex Cluster Admins
- ▶ ISC Cluster Admins
- ▶ Mirrors Team
- ▶ Donations Team
- ▶ Marketing Team
- ▶ Perforce Admins
- ▶ CVS & SVN Administrators
- ▶ Postmaster
- ▶ CVSup Mirrors Team

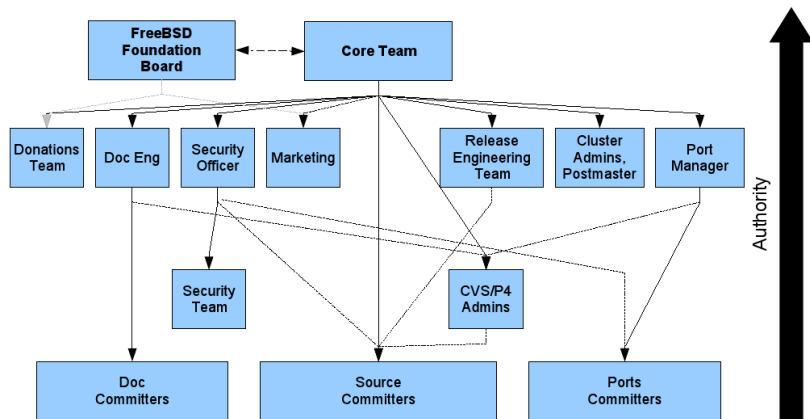


## Groups and Sub-Projects (2)

- ▶ Other Contributors
  - ▶ Perforce Contributors
  - ▶ Mailing List Subscribers
- ▶ Software Adaptation Projects
  - ▶ FreeBSD GNOME Project
  - ▶ FreeBSD KDE Project
  - ▶ X.Org on FreeBSD
  - ▶ Mono on FreeBSD
  - ▶ OpenOffice.Org on FreeBSD
  - ▶ Java on FreeBSD
  - ▶ VirtualBox, Gecko, Apache on FreeBSD
  - ▶ Python, Ruby, Perl, Haskell support
- ▶ Special Projects
  - ▶ Stress Testing
  - ▶ FreeBSD Tinderbox
  - ▶ FreeBSD Standards
  - ▶ Summer of Code Mentors
  - ▶ Monthly Status Reports
  - ▶ Coverity Team
- ▶ External Projects
  - ▶ KAME Project
  - ▶ TrustedBSD Project
  - ▶ PC-BSD
  - ▶ DesktopBSD



# FreeBSD Project Organization



# FreeBSD Core Team

- ▶ Historically “key” developers but now ...
- ▶ ... 9-member elected management body
  - ▶ Votes and candidates from the full set of active committers
  - ▶ Core team secretary
- ▶ Responsibilities
  - ▶ Administrative (commit bits, hats, team charters)
  - ▶ Strategic (project direction, coordination, cajoling)
  - ▶ Rules, conflict resolution, enforcement





# Conflict Resolution

- ▶ Developers generally characterized by:
  - ▶ Independence
  - ▶ Cooperation
  - ▶ Common sense
- ▶ Facilitated by intentional avoidance of overlap
- ▶ Strong technical disagreements, personality conflicts, etc., do occur sometimes
- ▶ When they get out of hand, generally mediated by a member of the Core Team



# Bikesheds

- ▶ A very special kind of conflict
- ▶ Not specific to FreeBSD, but one of the favorites
- ▶ Strong opinions easier to have on unimportant details



# Communication

- ▶ Vast majority of FreeBSD discussion and planning takes place on mailing lists
  - ▶ Both developer and user
  - ▶ Organized loosely by topic
- ▶ Over 40 active topics
  - ▶ announce, current, stable, cvs-all, ports, security, ...
  - ▶ chat, hackers, questions, ...
- ▶ Mostly public, a few private
  - ▶ core, portmgr, doceng, re, security-officer, ...
- ▶ FreeBSD Forums
- ▶ IRC channels on EFNet, FreeNode (#bsdports, #bsddocs, #bsddev, ...)

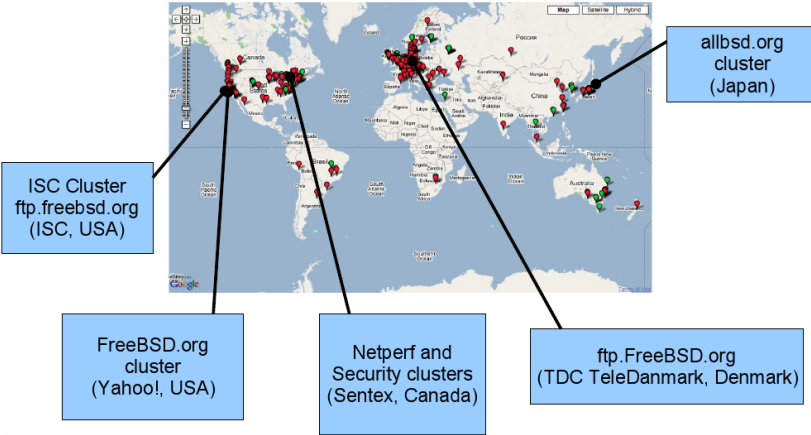


# Developer Summits

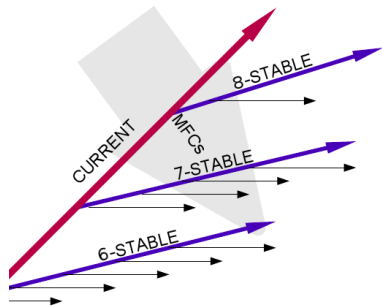
- ▶ Venues for working on on-going projects, not necessarily to make important decisions
- ▶ Strong community participation, with few sharp edges to foster collaboration
- ▶ This is where much of the beer is consumed :-)



# FreeBSD.org Clusters



# Development Model: Branches



- ▶ Concurrent development
- ▶ Divergence based on feature maturity
- ▶ MFCs (“merge from current”)
- ▶ Errata/security branches
- ▶ 18-24 month “dot zero” cycle



# Subversion & CVS

- ▶ Historically CVS is the primary revision control system
  - ▶ Authoritative project activity is in CVS
  - ▶ Actually there are four repositories
    - ▶ `ncvs` – `src` (from SVN) (228 developers)
    - ▶ `pcvs` – `ports` (169 developers)
    - ▶ `dcvs` – `doc/www` (55 developers)
  - ▶ More than 10 years of revision history
  - ▶ One commit in about every 12 minutes for the last years
- ▶ Move to Subversion
  - ▶ The `src` repository is moved to SVN in May 2008
  - ▶ A CVS `src` tree is kept in sync with SVN



# Perforce

- ▶ Secondary revision control system
  - ▶ Supports heavily branched development
  - ▶ It is mainly for FreeBSD developers
  - ▶ But it is also available for “guests”
- ▶ Dozens of active projects including
  - ▶ SMPng, TrustedBSD, Audit+MAC+SEBSD, Superpages, ...
  - ▶ UART, ARM, MIPS, Summer of Code, DTrace, Xen, ...
  - ▶ GEOM, GJournal, AFS, ZFS, highly parallel network stack, ...





Thank you for your attention!

