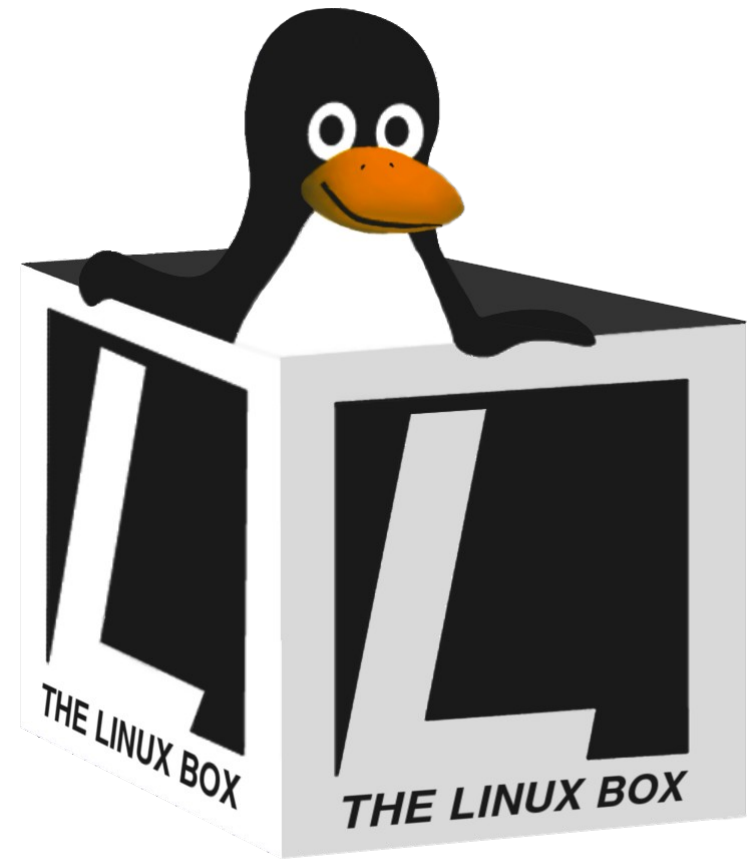


OpenAFS: BSD Clients 2009



Matt Benjamin
<matt@linuxbox.com>

OpenAFS: BSD Clients 2009

Who am I?

- OpenAFS developer interested in various new-code development topics
 - for the last while, “portmaster” for BSD clients except DARWIN/MacOS X
 - involves evolving the ports, interfacing with users and port maintainers in the BSD communities

OpenAFS: BSD Clients 2009

Former Maintainers

- Tom Maher, MIT
- Jim Rees, University of Michigan (former Gatekeeper and Elder)
- Garret Wollman, MIT

OpenAFS: BSD Clients 2009

Other Active

- Ben Kaduk (FreeBSD)
- Tony Jago (FreeBSD)
- Jamie Fournier (NetBSD)

OpenAFS: BSD Clients 2009

Historical Remarks

- AFS originated in a BSD 4.2 environment
 - extend UFS with coherence across a group of machines
 - Terminology in common with BSD, SunOS, etc, e.g., vnode
 - Followed SunOS and Ultrix to Solaris and Digital Unix in Transarc period
 - 386BSD released at 4.3 level in the Transarc period, some client development never publically released (or independent of Transarc)

OpenAFS: BSD Clients 2009

BSD Clients Today

- Descendents of 386BSD distribution and successors, not including DARWIN/MacOS X
 - DARWIN separately maintained, though of course there are similarities

OpenAFS: BSD Clients 2009

Today

- FreeBSD
- OpenBSD

Soon

- NetBSD
- OpenBSD

Not yet supported (as a client):

- Dragonfly BSD

OpenAFS: BSD Clients 2009

BSD Port History I

- First 386BSD port probably that of John Kohl (MIT), for NetBSD
- First to appear in OpenAFS is FreeBSD, by Tom Maher
- Next to appear in OpenAFS is OpenBSD, by Jim Rees
- Significant evolution on FreeBSD port (vm integration, Wollman and Rees)
- VFS integration fixes for FreeBSD 7.x and 8.x (mbenjamin, ongoing)

OpenAFS: BSD Clients 2009

BSD Port History II

- FreeBSD official client port (freebsd-afs, Boris Fomitchev, Maintainer)
- NetBSD re-based at OpenBSD (mbenjamin, in development)

OpenAFS: BSD Clients 2009

Status Report By Client Port

- OpenBSD
- FreeBSD
- NetBSD
- Dragonfly BSD

OpenAFS: BSD Clients 2009

OpenBSD

- Updated for OpenBSD 4.3 and 4.4
- amd64/x86_64 subtarget added
- param files reorganized, supports building all Intel subtargets, including amd64 (and some other subtargets) back to OpenBSD 3.6
- locking primitives reset to platform defaults (SMP support effectively added)

We are building the current stable OpenBSD release at x86 and amd64

OpenAFS: BSD Clients 2009

FreeBSD

- updated successfully for FreeBSD 7.0 and some 7.x
 - Most fully vetted at early 7.1 (then 8-CURRENT) with full WITNESS
- locking primitives switched from lockmgr (heavy weight) to mtx (light weight, MP-fast, no-sleep profile)
- exporting as MP-SAFE

We are building the current stable FreeBSD releases at x86 and amd64

OpenAFS: BSD Clients 2009

FreeBSD: Open Issues

- Some vnode locking issues introduced at 8-CURRENT (ongoing)
- Some vm issues may remain (ongoing)
- Ongoing subsystem reorganization on 8-CURRENT
 - Network stack virtualization
 - Privilege separation (removing suser(), thanks to Ben Kaduk, FreeBSD 8 contains named privilege levels for AFS)

AFS: Extended CallBack Information

New Development

- MAC support introduced in 8 included per-thread credentials support, FreeBSD client may integrate with this in future (open)

OpenAFS: BSD Clients 2009

NetBSD

- OpenAFS has had no NetBSD client port, although as noted, one existed for Transarc AFS
- In ongoing work, NetBSD port reset at OpenBSD port, which as stated, is close to the BSD ancestor port and, of course, closely related to OpenBSD
- Inherits OpenBSD locking fixes

If all goes well, I will have a testable release of rebased port later in the week

OpenAFS: BSD Clients 2009

NetBSD: Open Issues

- unlike OpenBSD, NetBSD has a unified buffer cache (UBC)
 - vm integration work to support it will be the next large task for the port
- NetBSD developer Elad Efrat is removing interfaces required for privilege checking in a style modeled on DARWIN
 - Implementing kauth plugin will be required
- NetBSD 5+ has 1:1 threading
- Vnode locking in NetBSD > 5 may change significantly

OpenAFS: BSD Clients 2009

Dragonfly BSD

Currently only userland and servers ported, not extensively tested. Feedback requested from potential users

Remarks

- Currently kernel resembles a traditional BSD in VFS and locking respects, at least on the surface
- Intended evolution to a single-system-image profile, tracking this in OpenAFS may require changing fundamental assumptions about data locality and cache organization (open)

AFS: Extended CallBack Information

Final Remarks

- Get in touch/say hi!
- Report open issues
- Offers of assistance gladly accepted!