OpenAFS Road Map Discussion

Derrick Brashear

Jeffrey Altman

OpenAFS Governance

- Gatekeepers
  - Two commercial and two academic
- Elders
  - Three commercial and three academic
- Gatekeepers and Elders set long term direction, promote and fund raise
- Gatekeepers are responsible for code quality, documentation, and release management

OpenAFS Legal

- OpenAFS does not legally exist
  - Can’t sign contracts
  - Can’t directly accept checks
    - USENIX (501c3) accepts contributions
    - CMU holds our limited funds
  - Can’t hold a copyright on our software
  - Can’t own domain names

Development Projects Currently in Progress

- Provides Removing Reliance on DES with new security classes:
  - rxk5 (krb5) – developed at UMich; almost ready for testing
  - rxgk (GSS-API) – implementation meeting at KTH this month
  - Estimate: TBD
- Improving Performance of Namei file servers
  - Necessary for supporting AFS storage on JFS
  - Being fixed by safe removal of unnecessary fsync calls
  - Patch submitted to Gatekeepers – Testing in progress
- MacOS X Finder issues
  - To be fixed with Apple in a future release of MacOS X

More Development Projects Currently in Progress

- Demand Attach File Server
  - Estimate: 2 developer weeks
- Instrumentation
  - Estimate: 2 developer months

Development Projects in Search of Support

- Improving performance on 10GBit networks and Implementing Support for IPv6 support
  - RX/TCP (NRL funded initial development)
  - Estimate: 20 to 25 developer months
- Native Windows File System
  - XP SP2, 2003 SP1, XP 64, 2003 64, and Vista
  - Replace SMB server w/ File System Redirector, Network Provider, and Kernel TDI module
  - Estimate: 10 to 11 developer months
- Disconnected Operations
  - Read-only Estimate: 2 developer weeks
  - Read-write Estimate: 6 to 7 developer weeks

More Development Projects Seeking Support

- Removing Directory Limitations (poor search times, < 64,000 entries, UNICODE unaware, single stream per file)
  - New directory format and RPCs
  - Backward compatibility functionality for old clients
  - Estimate: 3 to 4 developer months
- Mandatory Locking and Byte Range Locks
  - Estimate: 2 developer months
- Windows User Interface Improvements
  - New Explorer Shell Extensions, Control Panel and MMC snap-in
  - Estimate: 4 developer months

secureendpoints

More Development Projects Seeking Support

- HostAFSD – Local File System Exporter
  - Peer to Peer file sharing
  - Estimate: 2 developer months
- PTServer
  - Multiple typed names per AFS ID
  - Estimate: 2.5 to 3.5 developer months
- Object Storage
  - Estimate: 2 developer weeks (integration / testing only)
- BSD ports
  - Estimate: 3 to 4 developer months (for all)

secureendpoints

More Development Projects Seeking Support

- Windows Servers
  - Estimate: 4 to 6 developer weeks
- Admin Tool Suite
  - MacOS X Estimate: 2 developer months
  - Windows Estimate: 1 to 2 developer weeks
- Multiple Volume Versions
  - NameI architectural changes
  - Estimate: 1 developer month

More Development Projects Seeking Support

- Long Volume Names
  - Estimate: 1 developer week
- Partition UUIDs
  - Estimate: 1 to 2 developer weeks
- NDMP
  - Estimate: TBD

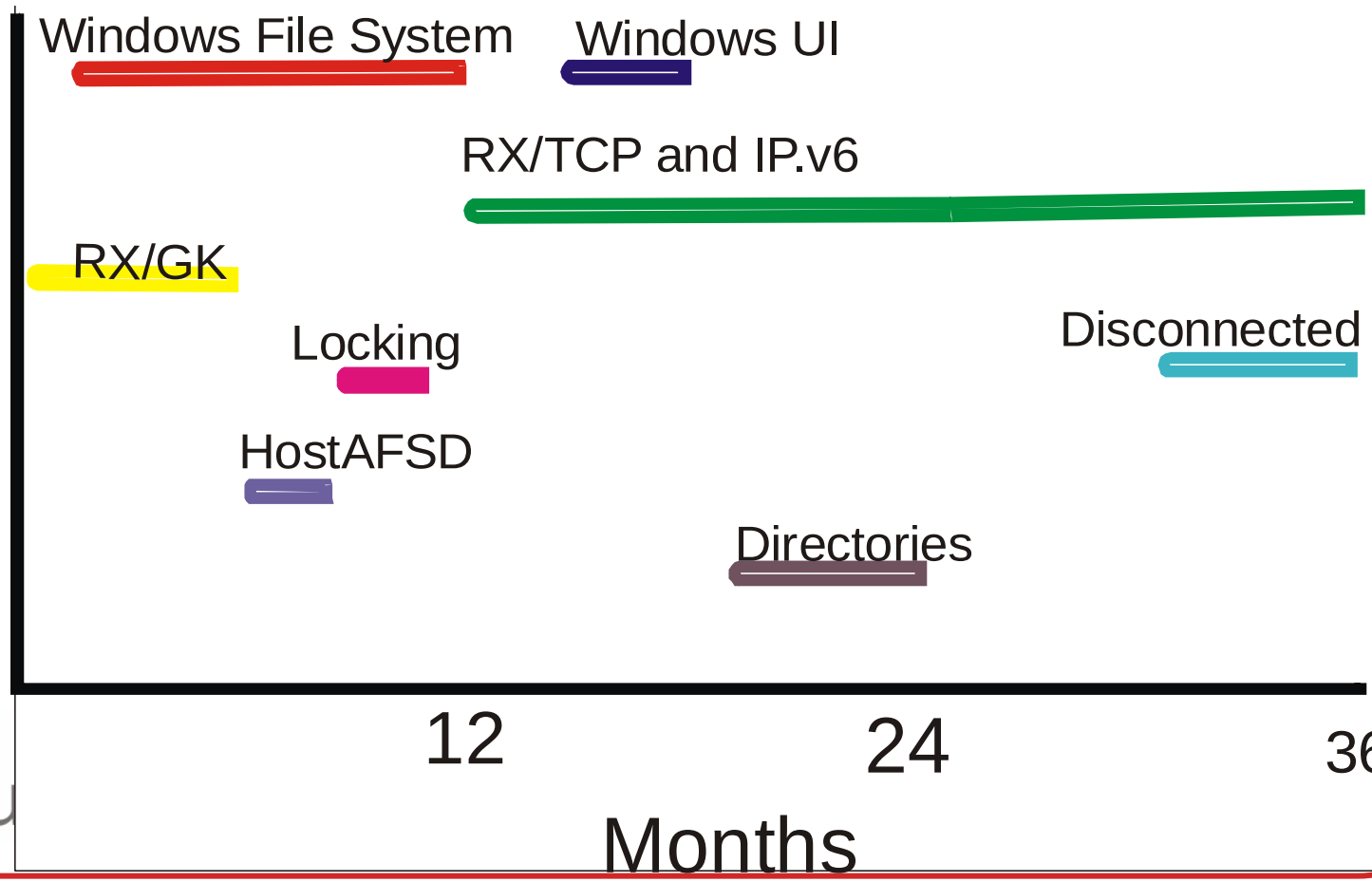
Non-development projects seeking support

- Documentation
- Web Site Design
- Test Infrastructure
- Repository Migration
- RT Improvements
- Restoration of openafs.org cell
- Development / Maintenance of OpenAFS virtual machine applicance

OpenAFS Developer Resources

- “Full Time”
  - Derrick Brashear
  - Jeffrey Altman
  - Tom Keiser
  - Asanka Herath
- Part Time
  - Chaskiel Grundman
  - Jeff Hutzelman
  - Chas Williams
  - Dale Ghent
  - Ken Hornstein
  - Hartmut Reuter
  - Matt Benjamin
  - Marcus Watts
  - Love Å
- Who did we forget?

Three Year Road Map
$350K / year (block grant)



Why plan for three or more years?

- Most of the OpenAFS projects that were low hanging fruit have already been accomplished.
- What remains are the larger and more difficult projects which will take months to years to complete.
- A multi-year commitment to support a limited number of significant OpenAFS development projects would make it much easier for OpenAFS to maximize developer time as well as improve additional fund raising.

Related Projects Independent of OpenAFS

- Backup Systems
  - (but no two sites want the same thing)
- Authenticated read/write export via WebDAV, CIFS and NFSv4
- Kerberos 5 Extensions
- PTS LDAP back-end

Why Should Your Organization Help Fund OpenAFS?

- Many University and Government organizations deploy AFS today in some capacity. Many researchers use AFS from partner institutions as part of their collaboration.
- Some organizations are committed to AFS for at least 3 to 7 years
  - One round of major operating system and hardware revisions
- If the rest decides to migrate:
  - It will take years. OpenAFS may be able to address its weaknesses before the migration is complete.
  - Migrating to any new storage solution is hard because it takes a long time to phase out existing clients
  - Storage is complex. If the migration fails, being able to fall back on an improved OpenAFS will be desirable

What if Your Organization Does Not Fund OpenAFS Development?

- OpenAFS will go on
  - Other organizations that are committed will provide what they can
  - The OpenAFS Elders will continue to solicit funding from other sources
- OpenAFS will take longer to achieve its potential
- Multi-year projects that require long term broad-based community support may never happen.
  - A native Windows file system is less pressing since there is already a client that works acceptably for most use cases
  - How many sites today have 10Gbit networks and wish to replicate or move tera byte volumes? What about five years from now?

Q & A

Further questions can be addressed to openafs-elders@openafs.org