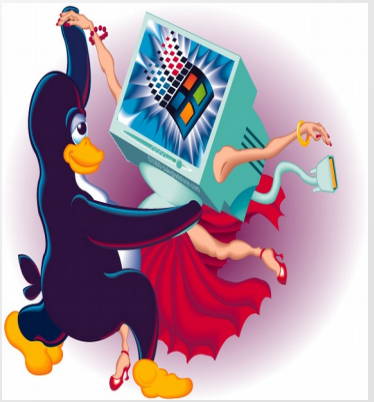
Linux SMB3 and pNFS: Shaping the Future of Network File Systems

Steve French
Principal Systems Engineer – Primary Data



Legal Statement

- This work represents the views of the author(s) and does not necessarily reflect the views of Primary Data Corporation
- Linux is a registered trademark of Linus Torvalds.
- Other company, product, and service names may be trademarks or service marks of others.

Who am I?

- Steve French smfrench@gmail.com
- Author and maintainer of Linux cifs vfs (for accessing Samba, Windows and various SMB3/CIFS based NAS appliances)
- Member of the Samba team, coauthor of SNIA CIFS Technical Reference and former SNIA CIFS Working Group chair
- Principal Systems Engineer, Protocols: Primary Data

Outline

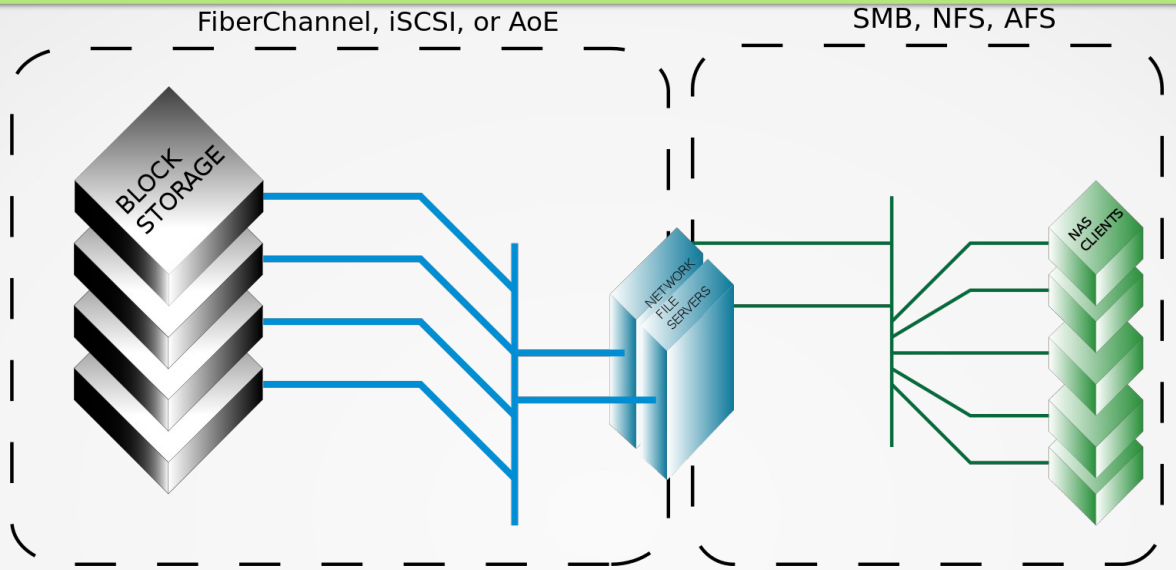
- Why pNFS and SMB3? Why Linux?
- Which key Features of SMB3 are implemented in Linux client?
- And for NFSv4.2? Which implemented in Linux client? And Linux nfsd kernel server?
- What is Flexfiles?
- Differences between NFSv4.2 and SMB3.1
- SMB3 kernel client status
- Work in progress

- To answer … why pNFS and SMB3 on Linux?
  - We can first ask … Why File Systems? Then …
    - Why NAS? Then ...
      - Why focus on NFS and SMB (NFSv4.2 and SMB3.11)?
        - And … Why on Linux?

Why we care about FS?

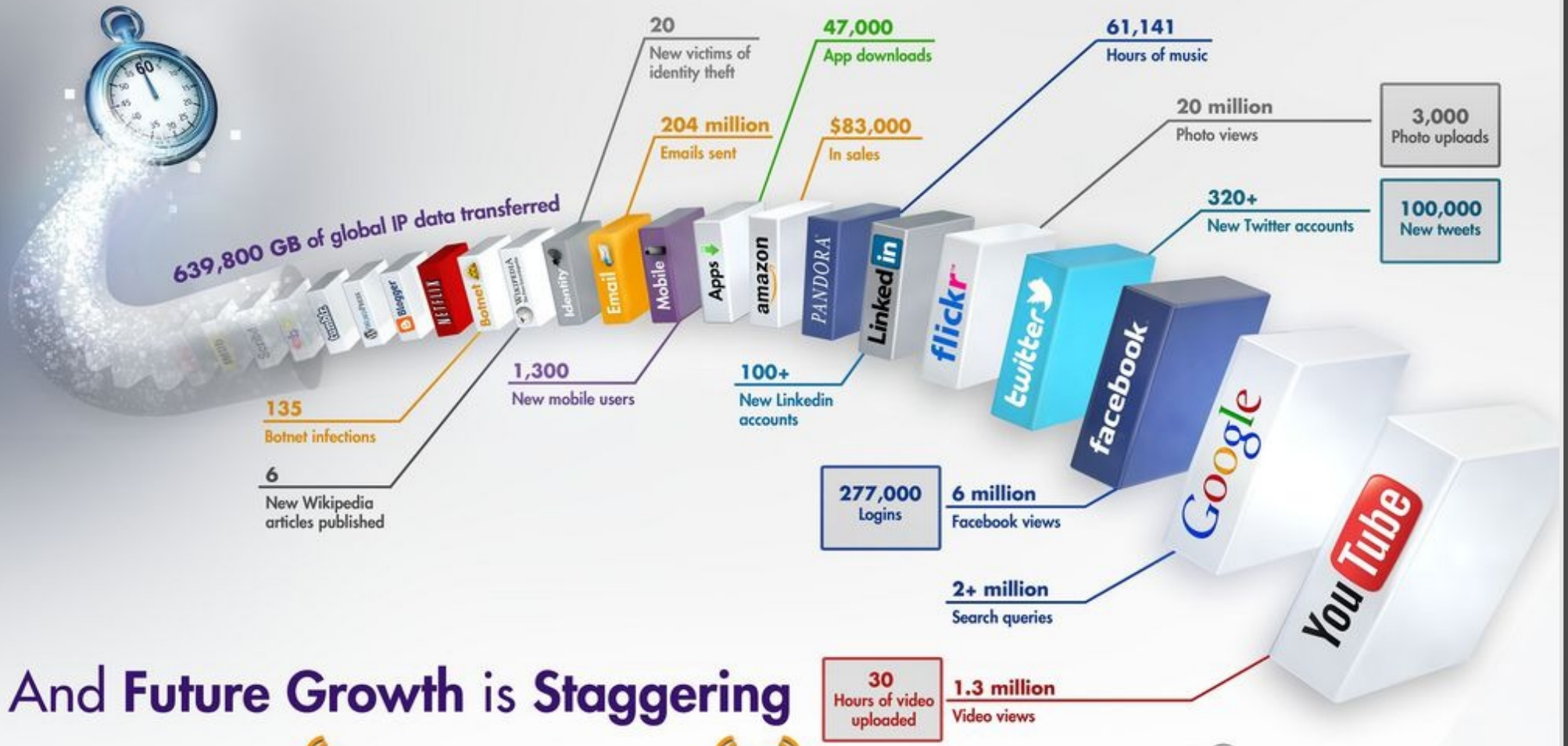
- 50 years ago first Hierarchical File System was built, http://www.multicians.org/fjcc4.html , yet more than ever we care how we store our data. Amount of data (largely unstructured) exceeded two Zettabytes by 2012 (IDC estimate), continues to double every two to three years.
- And it is transferred around A LOT
  - “Annual global IP traffic will surpass the zettabyte (1000 exabytes) threshold in 2016.” (CISCO estimate)
- Nearly all workloads depend heavily on file systems.

Why NAS?



- NAS is superset of block (SAN) and object … but easier to manage
- NAS (now) can get 90+ of the performance of SAN with lower administrative costs and more flexibility. Easier to setup
- Attributes at the right granularity (file/directory/volume)
- Ownership information, easier to understand security, easy backup, optimizable with useful info on application access patterns, intuitive archive/encryption/compression policy, quotas, quality of service

What Happens in an **Internet Minute?**

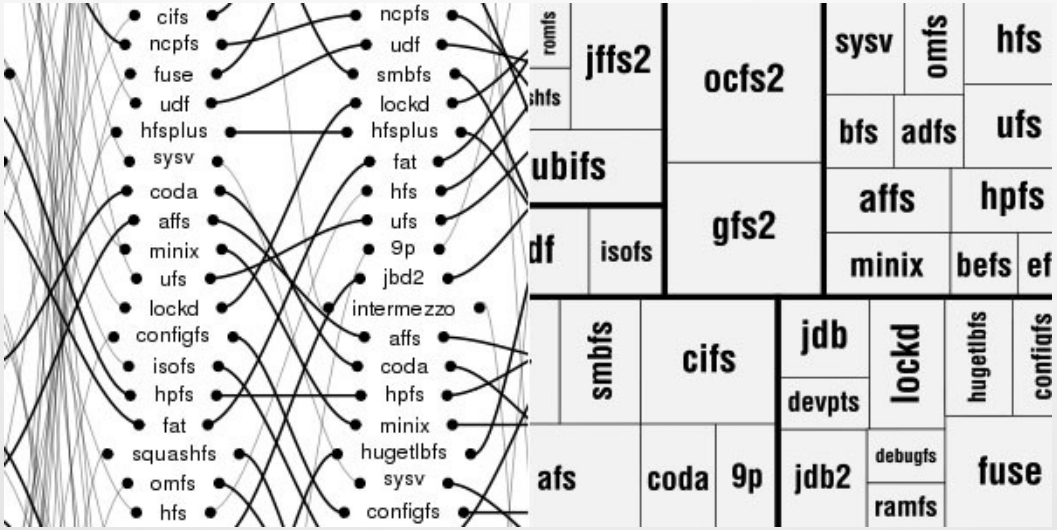


And **Future Growth** is **Staggering**

http://www.intelfreepress.com/news/networks-strain-to-keep-pace-with-data-explosion/191/

And why Linux?

- Large Talented Community. Incredible improvement rate. In past 13 months since 3.16:
  - More than 90,000 changesets developed, reviewed tested and merged to the kernel (almost 15,000 just in the last release, Linux 4.2). Development pace accelerating
  - More than 5100 in the file system alone (almost 1000 in the last release alone)
    - 822,000 lines of File System kernel code (often very terse, and highly optimized)
    - Changes from over 1200 developers are added to the kernel each release
  - Development never stops – constant incremental improvements and fixes
  - Great processes and pragmatic tools (e.g. “git” distributed source code control and xfstest)
- Broad selection of file systems. More than 50 to choose from not just cifs, nfs and ext4!



Linux FS Community is talented (Picture at 2015 FS Summit in Boston)

What are the most actively maintained Linux Filesystems?

- 5095 kernel filesystem changes in last year (since 3.16 kernel)!
  - Linux kernel file system activity is continuing to be very strong
  - Lots of improvement in defacto standard Linux xfstest test suite as well
- Six file systems and the VFS drive the majority of activity
- NFS client (and server) and cifs.ko (cifs/smb3 client) get more changesets than all but a few fs
  - VFS (overall fs mapping layer and common functions) 803 changesets
  - BTRFS 730
  - NFS client 568
  - XFS 434
  - NFS server 405 (activity increased dramatically this year)
  - Ext4 307
  - CIFS/SMB2/SMB3 client 182
- NB: Samba (cifs/smb2/smb3 server) is more active than all those put together since it is broader in scope (by a lot) and also is in user space not in kernel

SMB3 Performance – Linux client

Kernel (including cifs and nfs clients) continue improving

- Now we have Linux 4.3-rc1 ie “Hurr Durr I'm a Sheep”

13 months ago 3.16 “Shuffling Zombie Juror”

SMB2/SMB3 Optional Feature Status

- Security
  - Complete: Downgrade attack protection, SMB2.1 signing
  - Almost complete: SMB3.11 negotiate contexts
  - Not yet: per-share encryption, krb5 mnt (is in cifs), CBAC (DAC ACLs
- Data Integrity:
  - Durable Handle Support (complete)
- Performance
  - Complete: multicredit, large I/O, fast server side copy offload (copy chunk and also “duplicate extents”)
  - Not yet: T10 copy offload, Multichannel, RDMA, directory leases, Branch Cache integration, use of compound ops on wire
- Clustering
  - Not yet: Witness protocol integration, Persistent Handles/Continuous Availability
- Other
  - Set/Get Compression, SetIntegrity and Sparse File support (complete)
  - Unix/POSIX extensions: prototyped (looks promising)

Key NFSv4.2 Feature Status

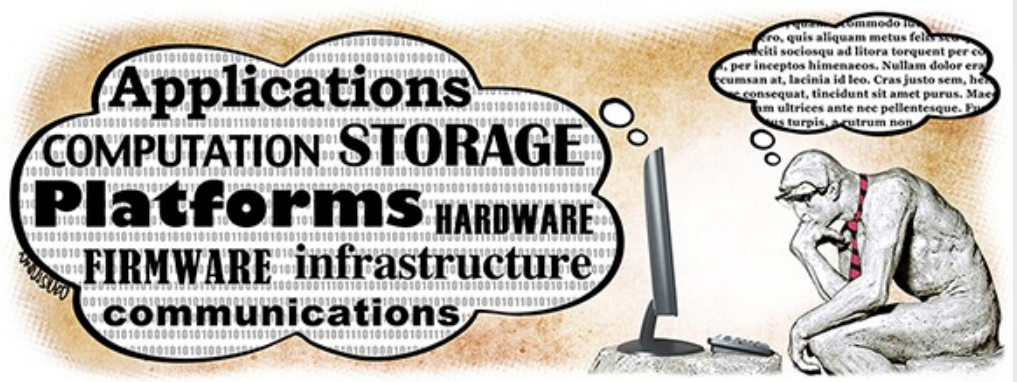
pNFS Variants – “layout types”

What is Flexfiles?

NFSv4.2 and SMB3.11 comparison

SMB3 Work In Progress

- Improved xfstest (automated verification test) compatibility (fix a few remaining bugs)
  - Fix fallocate/punch hole bug
- SMB3 (vs. CIFS) implementation gaps
  - CIFS ACLs, KRB5
- Better POSIX emulation/support for SMB3
- Improved ACL support
- Performance improvements
- Per-share encryption



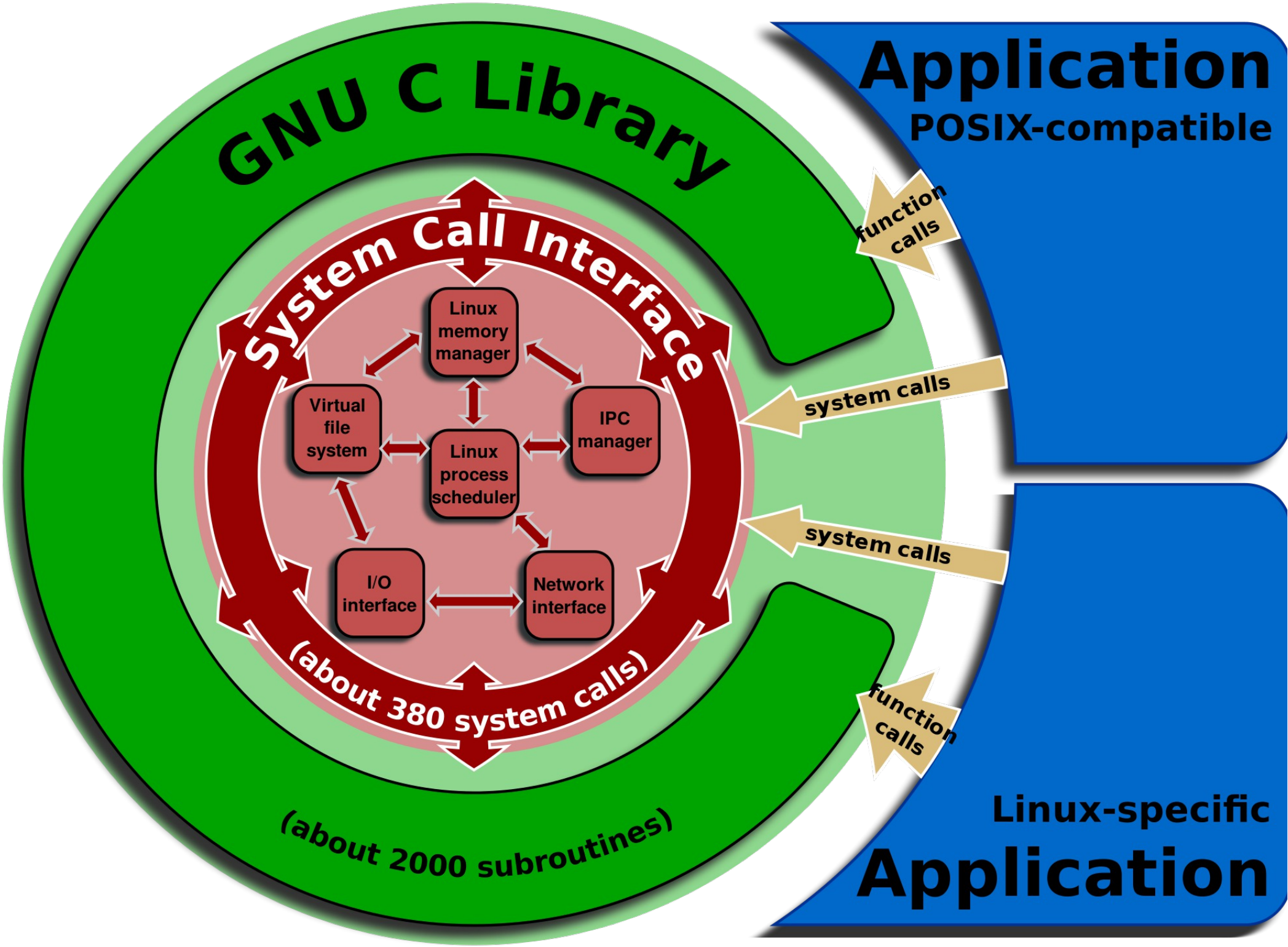
POSIX Compatibility

- *The problem:* **SMB/CIFS deprecation** (now that SMB3 is pervasive and more secure and faster and ...). See: http://blogs.technet.com/b/josebda/archive/2015/04/21/the-deprecation-of-smb1-you-should-be-planning-to-get-rid-of-this-old-smb-dialect.aspx
- Specialized POSIX Protocol Extensions that Samba implements are **CIFS only**
- *The Answer:* Move to SMB3 (and later) … BUT …
- *2nd problem:* Full "POSIX" compatibility (actually better to say we need "pragmatic Linux application interoperability") for SMB3 or at least as good CIFS ("good enough")
- *Requirement*:

  for (all key features)

  SMB3 >= CIFS

- Customers don't want SMB3 to be a step back or to break their apps
- Fortunately we are close to solving this and making Linux SMB3 support even better!


GNU C Library
System Call Interface
Linux memory manager
Virtual file system
Linux process scheduler
IPC manager
I/O interface
Network interface
(about 380 system calls)
(about 2000 subroutines)
function calls
system calls
system calls
function calls
Application
POSIX-compatible
Linux-specific
Application

POSIX/Linux Compatibility: Details

- *Implemented:*
  - *Hardlinks*
- *Emulated: (current cifs.ko SMB3 code)*
  - *POSIX Path Names:* Approximately 7 reserved characters not allowed in SMB3/NTFS etc. *(e.g. ? * \ : !)*
  - *Symlinks* (ala “mfsymlinks” Minshall-French symlinks)
  - *Pseudo-Files:* FIFOs, Pipes, Character Devices (ala “sfu” aka “Microsoft services for unix”)
- *Partial:*
  - *Extended attribute flags* (lsattr/chattr) including compressed flag
  - *POSIX stat and statfs info*
  - *POSIX Byte Range Locks*
- *Not implemented, but emulatable with combination of SMB3 features and/or use of Apple AAPL create context*
  - *Xattrs* (Security/Trusted for SELinux, User xattrs for apps)
  - *POSIX Mode Bits*
  - *POSIX UID/GID ownership information*
  - *Case Sensitivity* in opening paths
- *Not solvable without additional extensions:*
  - *POSIX Delete (unlink) Behavior*

Demo

- Client:
  - Current kernel (4.3-rc) mainline (on an Ubuntu VM in this machine)
- Mounted
  - via SMB3.0 to Samba server version 4.3 Ubuntu
  - and Mac …
  - And Windows 10

Other Features under investigation

- SMB3 ACL support
- Better streams support (how to list streams, useful for backup e.g.)
- DCE/RPC over SMB3: Pipe reads/write over IPC$ pseudo-mount
- Recovery of pending byte range locks after server failure (we already recover successful locks)
- Investigation into additional copy offload (server side copy) methods
- Full Linux xattr support
  - Empty xattr (name but no value)
  - Case sensitive xattr values
  - Security (SELinux) namespace (and others)

Improvements by release (continued)

- 3.12 40 changes, cifs version 2.02: **SMB3 support much improved**
  - SMB3.02 dialect negotiation added
  - Authentication overhaul
  - SMB3 multiuser signing improvements, (thank you Shirish!) allows per-user signing keys on ses
  - SMB2/3 symlink support (can follow Windows symlinks)
  - Improved data integrity: Lease improvements (thank you Pavel!)
  - debugging improvements
- 3.13 34 changes
  - Add support for setting (and getting) per-file compression (e.g. “chattr +c /mnt/filename”)
  - Add SMB copy offload ioctl (CopyChunk) for very fast server side copy
  - Add secure negotiate support (protect SMB3 mounts against downgrade attacks)
  - Bugfixes (including for setfacl and reparse point/symlink fixes)
  - Allow for O_DIRECT opens on directio (cache=none) mounts. Helps apps that require directio such as newer specsfs benchmark and some databases
  - Server network adapter and disk/alignment/sector info now visible in /proc/fs/cifs/DebugData
- 3.14 27 changes
  - Security fix for make sure we don't send illegal length when passed invalid iovec or one with invalid lengths
  - Bug fixes (SMB3 large write and various stability fixes) and aio write and also fix DFS referrals when mounted with Unix extensions

Improvements by release (continued)

- 3.15 18 changes
  - Various minor bug fixes (include aio/write, append, xattr, and also in metadata caching)
- 3.16 25 changes
  - Allow multiple mounts to same server with different dialects
  - Authentication session establishment rewrite to improve gssapi support
  - Fix mapchars (to allow reserverd characters like : in paths) over smb3 mounts
- 3.17 65 changes (cifs version 2.04 – visible in modinfo)
  - Much faster SMB3 large read/write: including multicredit support (thank you Pavel!)
  - Many SMB3 fixes (found by newly updated automated fs tests: “xfstests”)
  - Directio allowed on cache=strict mounts
  - Fallocate/sparse file support for SMB3
  - Fixed SMB 2.1 mounts to MacOS
- 3.18 (Some highlights of what to expect in next kernel)
  - SMB3 Emulated symlinks: Mfsymlink support for smb2.1/smb3 (complete).
  - SMB3 POSIX Reserved Character mapping: support for reserved characters e.g. * : ? < > etc. (complete)
  - Workaround MacOS problem with CIFS Unix Extensions from Linux

Improvements by release (continued)

- 3.19 26 changesets
  - Fix Oplock bug, inode caching bug and ioctl clone bug
  - Fix conflicts between SecurityFlags (which allowed CONFIG_MUST_LANMAN and CONFIG_MUST_PLNTXT
  - Improve fallocate support
- Linux 4.0 (!) 21 changesets
  - Various minor stability fixes
- Linux 4.1
  - Stability fixes: Mapchars fix, fix to allow Unicode surrogate pairs (improved character conversion for some Asian languages), DFS fix, inode number reuse fix
- Linux 4.2
  - Partial support for SMB 3.11 (Windows 10) dialect (improved security)
  - Duplicate extents support added (copy offload with block duplication to REFS e.g.)
  - Get/Set Integrity ioctl added
  - Stability fixes

Cifs-utils

- The userspace utils: mount.cifs, cifs.upcall,set/getcifsacl,cifscreds, idmapwb (idmap plugin),pam_cifscreds
  - thanks to Jeff Layton for maintaining cifs-util
- Various improvements proposed and under development (additional userspace utils for smb3 features like copy offload and statistics being prototyped)

Using SMB3

- Practical tips
  - Use -o vers=3.0 to Samba or Windows (or vers=3.02 to latest Windows, consider vers=2.1 to MacOS or 3.0 to most recent Mac)
  - Mount options to consider
    - “mfsymlinks” (3.18 or later kernel)
    - “sfu” option enables creation of FIFOs and char devices
    - Consider experimenting with default rsize/wsize (which is 1MB) to improve large file I/O performance
- Restrictions
  - Case sensitivity
  - POSIX vs. Windows byte range locks, and unlink behavior

SMB3 Kernel Client Status

- SMB3 support is solid (and large file I/O FAST!), but lacks many optional features
  - Metadata performance expected to be slower (need to add open/query compounding)
- Badly need to prototype Apple's SMB2.1/SMB3 “AAPL” create context” to determine if adequately addresses a few remaining POSIX compatibility issues)
- Can mount with SMB2.02, SMB2.1, SMB3, SMB3.02 (or even SMB3.11)
  - Specify vers=2.0 or vers=2.1 or 3.0 or 3.02 (or 3.11) on mount
  - Default is cifs but also mounting with vers=1.0 also forces using smb/cifs protocol
  - Default will change to SMB3 soon (likely with new “mount -t smb3” ie using new “/sbin/mount.smb” and/or “mount.smb3” symlink – to avoid changing “mount -t cifs” behavior for existing users)

Testing … testing … testing

- One of the goals last summer was to improve automated testing of cifs.ko
  - Multiple cifs bugs found, test automation much improved, approximately 5 bugs/features remain to be fixed for full xfstest compatibility
  - See https://wiki.samba.org/index.php/Xfstesting-cifs
- Functional tests:
  - Xfstest is the standard file system test bucket for Linux
    - Runs over local file systems, nfs, and now cifs/smb3
      - Found multiple bugs when ran this first (including Samba bug – with times before Epoch e.g.)
    - Challenge to figure out which tests *should* work (since some tests are skipped when run over nfs and cifs)
  - Other functional tests include cthon, dbench, fsx. Cthon also has recently been updated to better support cifs
- Performance/scalability testing
  - Specsfs works over cifs mounts (performance testing)
  - Big recent improvements in scalability of dbench (which can run over mounts)
  - Various other linux perf fs tests work over cifs (iozone etc.)
  - Need to figure out how to get synergy with iostats/nfsstats/nfsometer

XFSTEST current status for cifs.ko

- Multiple server bugs found too
- Client bugs:
  - As with NFS, there are some intractable mtime consistency problems due to server/client last write time differences/delays, but these tests could be skipped
  - Generic tests: 011 (dirstress), 023 and 245 (rename), 075/091/127/263 (fsx failures fallocate related), 239 (need ACLs), 313 (timestamps)

- The Future of SMB3, NFS and Linux is very bright
- Let's continue their improvement!

Additional resources to explore for NFS and SMB3

- NFS:
  - https://tools.ietf.org/wg/nfsv4/
  - http://www.snia.org/sites/default/files/NFS_4.2_Final.pdf
  - http://nfsv4bat.org/
- SMB3:
  - https://msdn.microsoft.com/en-us/library/gg685446.aspx
    - In particular MS-SMB2.pdf at https://msdn.microsoft.com/en-us/library/cc246482.aspx
  - http://www.samba.org
  - Linux CIFS client https://wiki.samba.org/index.php/LinuxCIFS
  - Samba-technical mailing list and IRC channel
  - And various presentations at http://www.sambaxp.org and Microsoft channel 9 and of course SNIA … http://www.snia.org/events/storage-developer

Thank you for your time