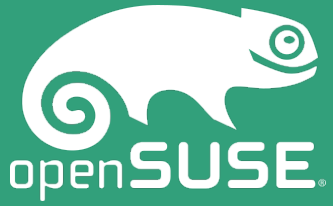


The CephFS Gateways Samba and NFS-Ganesha

David Disseldorp
ddiss@samba.org
Supriti Singh
supriti.singh@suse.com

Agenda

- Why
  - Exporting CephFS over Samba and NFS-Ganesha
- What
  - Architecture & Features
    - Samba
    - NFS-Ganesha
- How
  - Interoperability of Samba and NFS-Ganesha

Ceph Architecutre

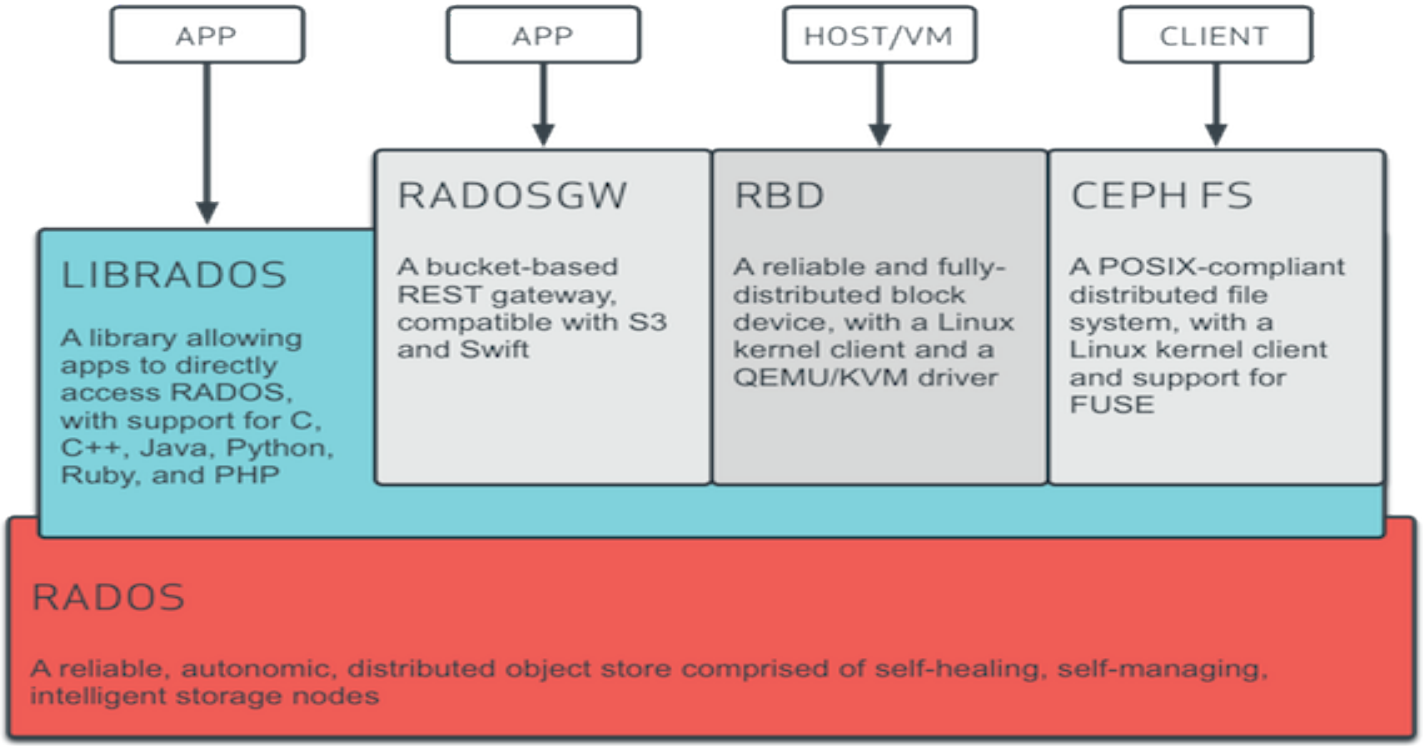
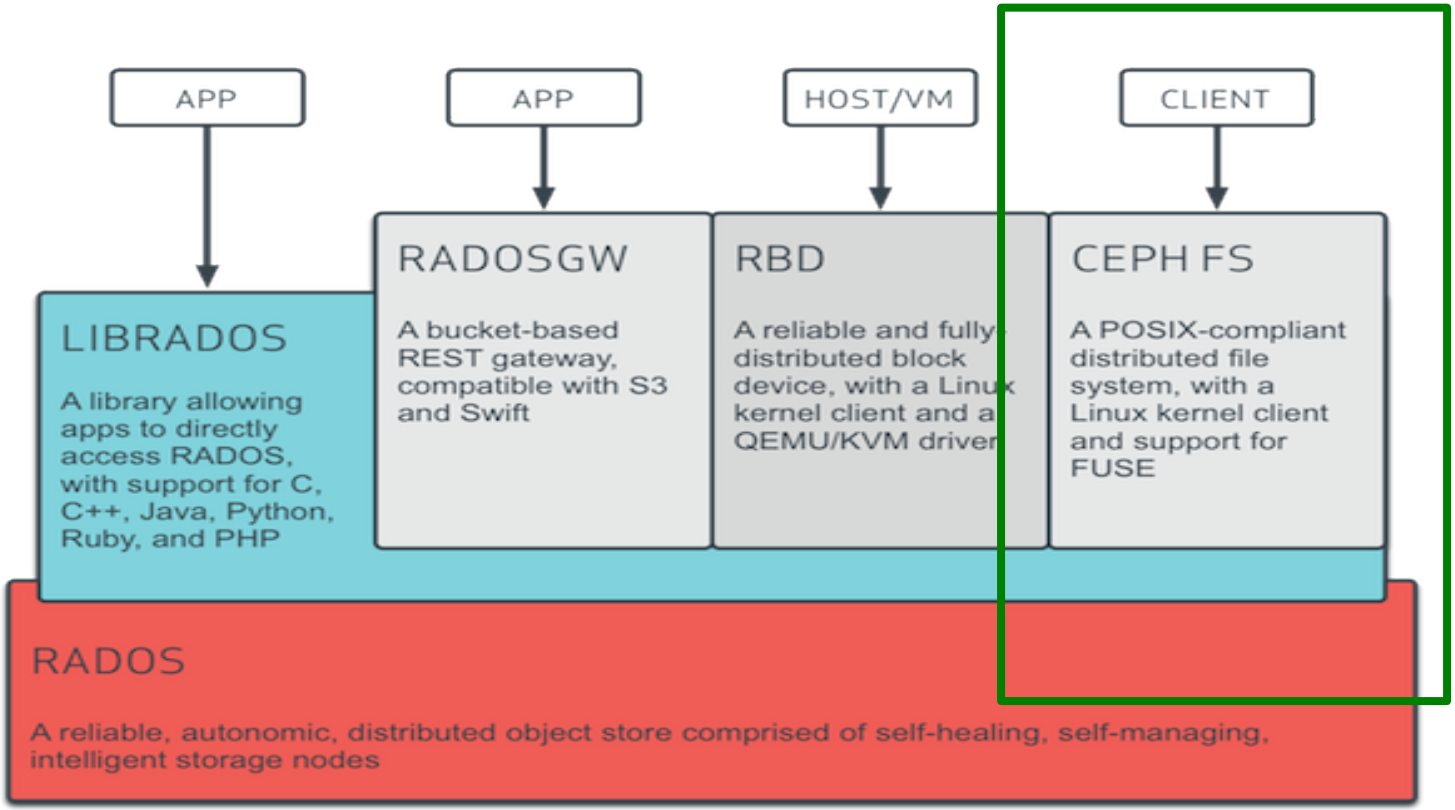
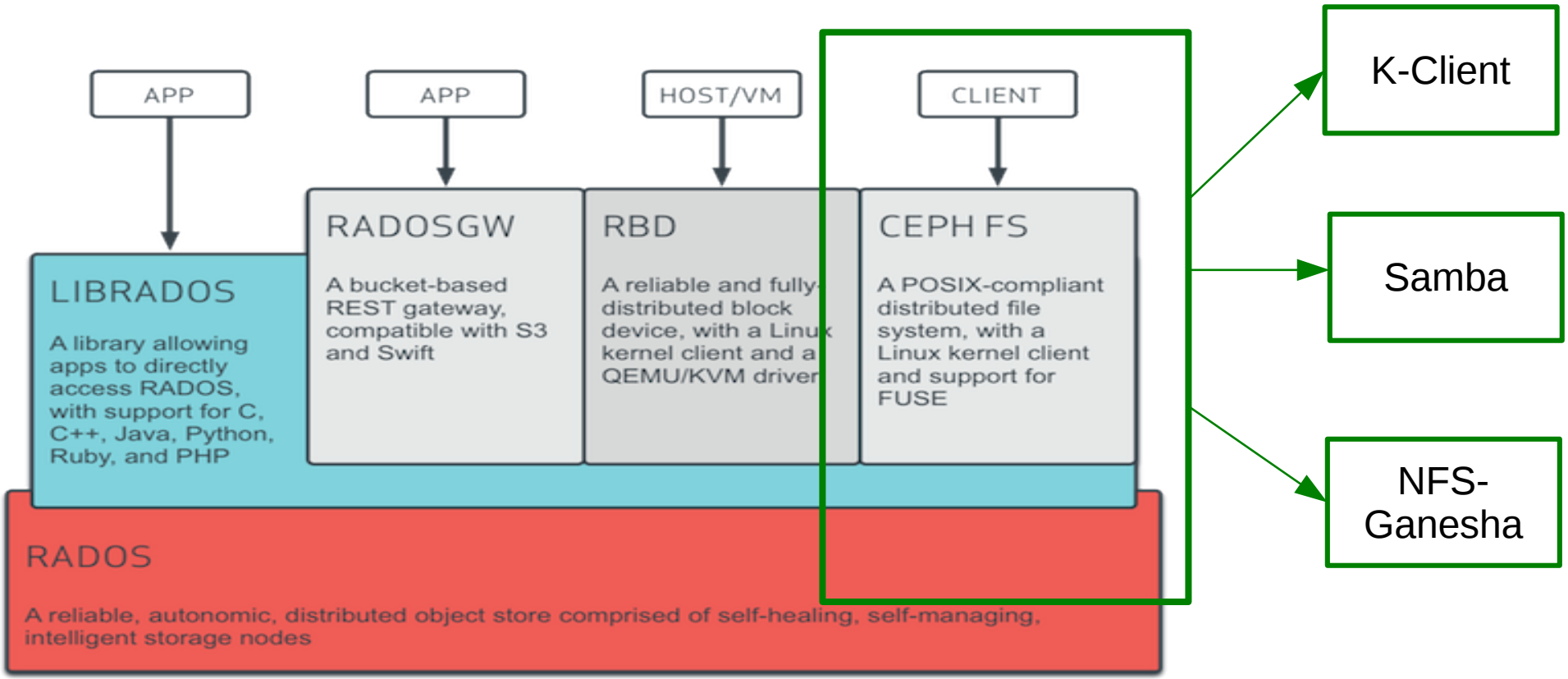


Image Source: http://docs.ceph.com/docs/giant/architecture/

CephFS Clients: Samba and NFS-Ganesha



CephFS Clients: Samba and NFS-Ganesha



NFS-Ganesha

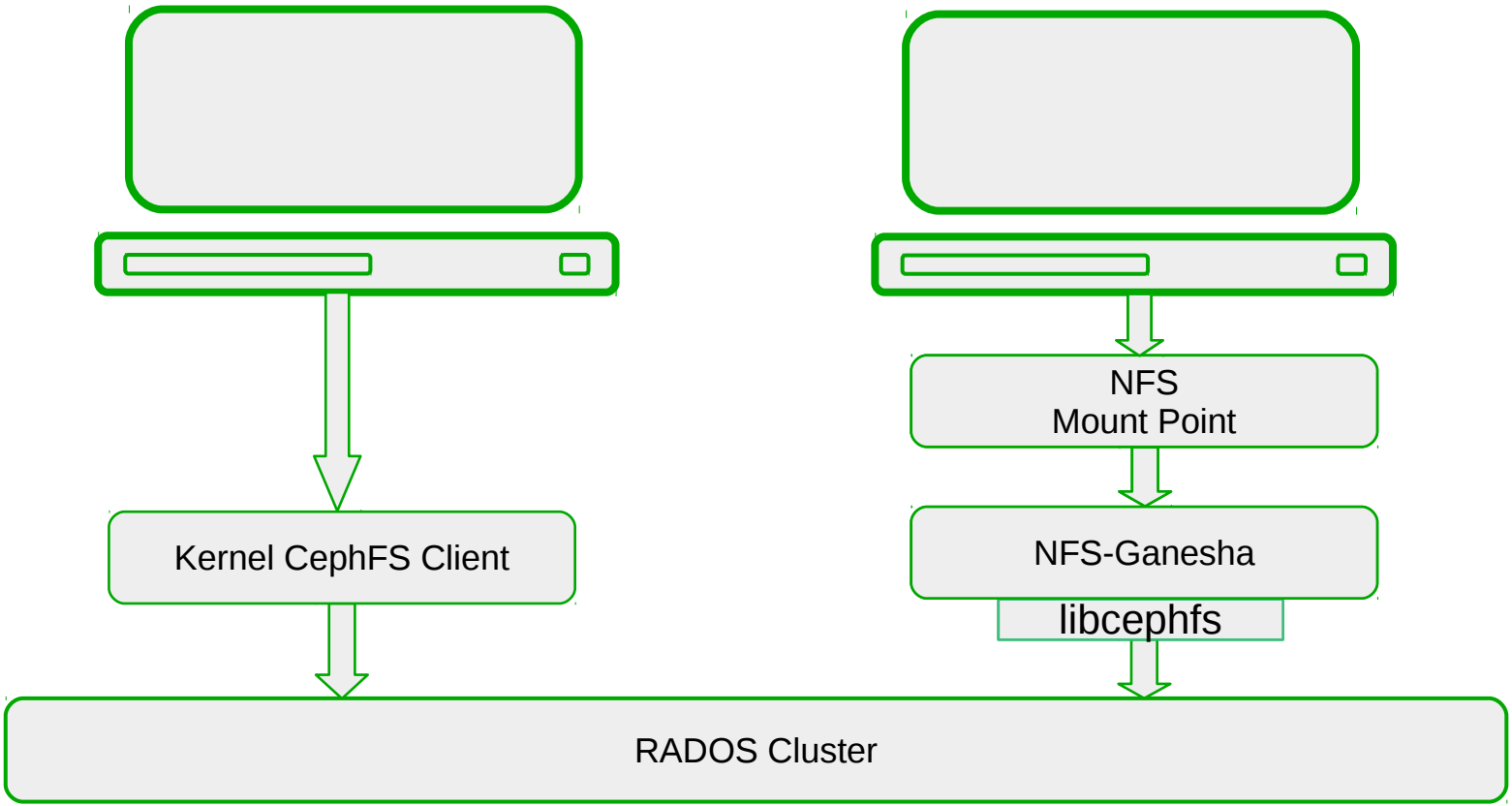
NFS-Ganesha

- Open source
- User space NFS server
- Supports multiple Filesystem Backends:
  - CephFS
  - RGW (RADOS Gateway)
  - Gluster
  - GPFS

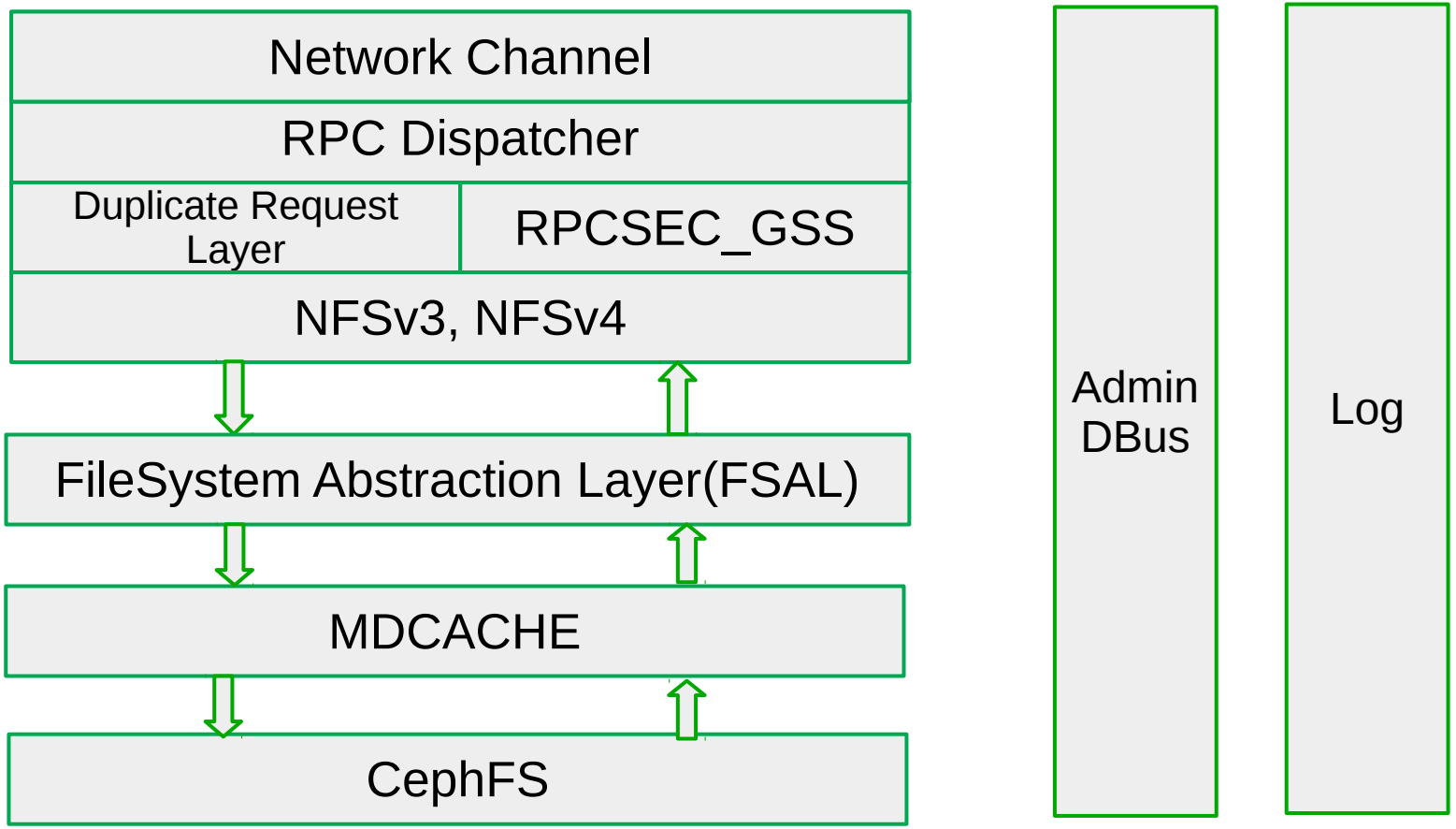
NFS-Ganesha

- Open source
- User space NFS server
- Supports multiple FilSystem Backends:
  - **CephFS**
  - RGW (RADOS Gateway)
  - Gluster
  - GPFS

NFS-Ganesha and CephFS



NFS-Ganesha Architecture



NFS-Ganesha Modular Architecture

- RPC layer:
  - uses libntirpc
- Filesystem Abstraction layer (FSAL)
  - Provides API for exported namespace
- Metadata cache (MDCACHE)
  - Chunked dirent cache (version 2.6)
- Dbus Interface
  - System management and communication
- Log Management
  - Support for internal logging

NFS-Ganesha key features

- Single nfs-ganesha instance can support:
  - Multiple exports
  - Multiple filesystem backend
  - Multiple Protocols
- RPCSEC_GSS with krb5 authentication
- Dynamically export/unexport entries using DBus

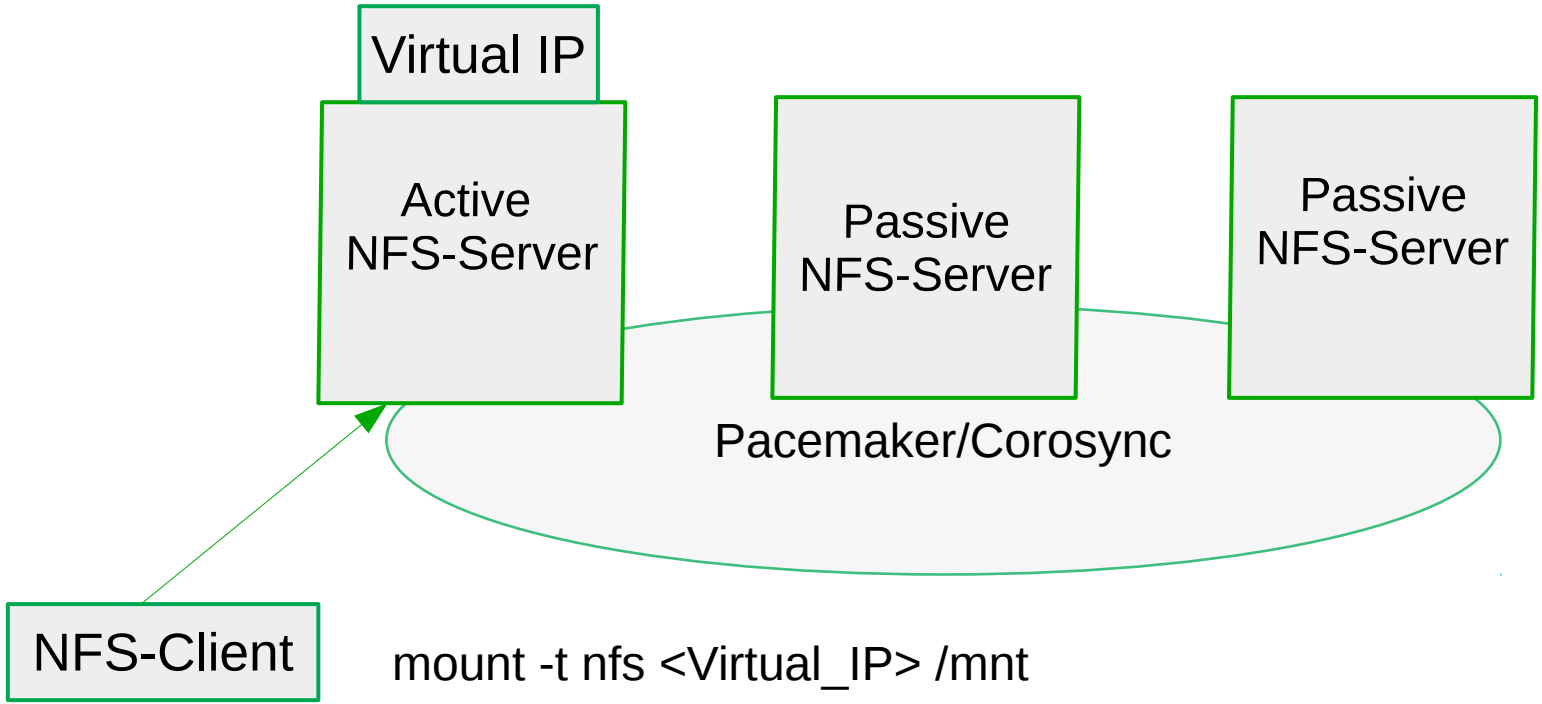
NFS-Ganesha CephFS features

- Cephx authorization
- Read delegations
- Export subdirectories
  - Load balancing

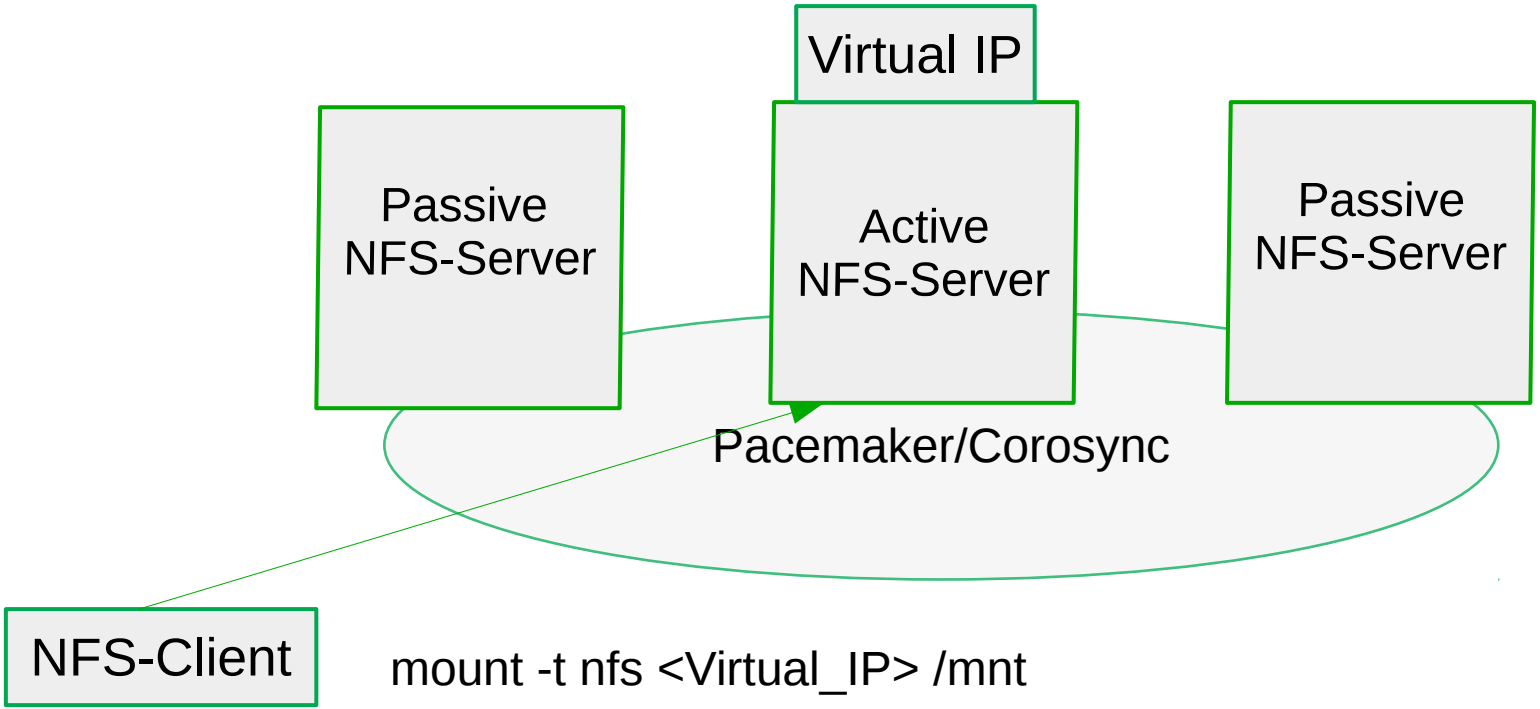
NFS-Ganesha High Availability

- Problem: Single NFS-Ganesha server
  - Single point of failure
  - Bottleneck
  - Cannot scale with backend filesystem.
- Solution: Clustering
  - High availability
  - Load balancing

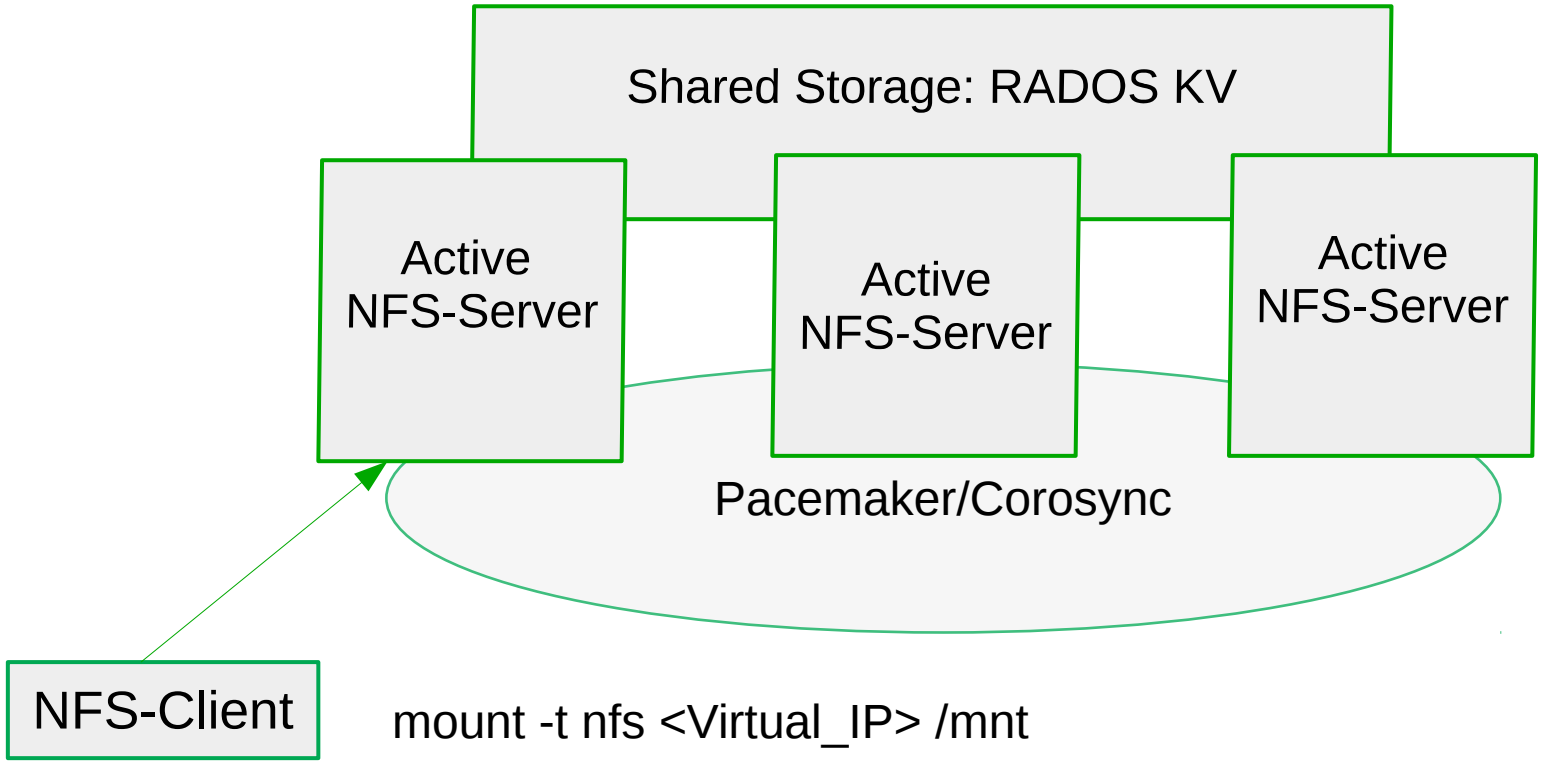
NFS-Ganesha HA: Active-Passive



NFS-Ganesha HA: Active-Passive



NFS-Ganesha HA: Active-Active



Samba

Samba

- File and print sharing
  - SMB / CIFS, SMB2 and SMB3+ dialects
- Authentication
  - NTLMv2 and Kerberos
- Identity mapping
  - Windows *SIDs* to *uids* and *gids*
  - Active Directory domain member or domain controller

Protocol

- SMB / CIFS
  - Legacy dialect
  - Hundreds of commands and subcommands
  - UNIX extensions
- SMB2
  - Clean break from old dialects
  - Modern, simplified protocol with improved performance

Protocol (continued)

- SMB2.1 → SMB3.1.1
  - Most recent protocol revisions
  - Lease improvements
  - RDMA extensions
  - Multichannel
  - Witness Protocol
  - End-to-end encryption

Clients

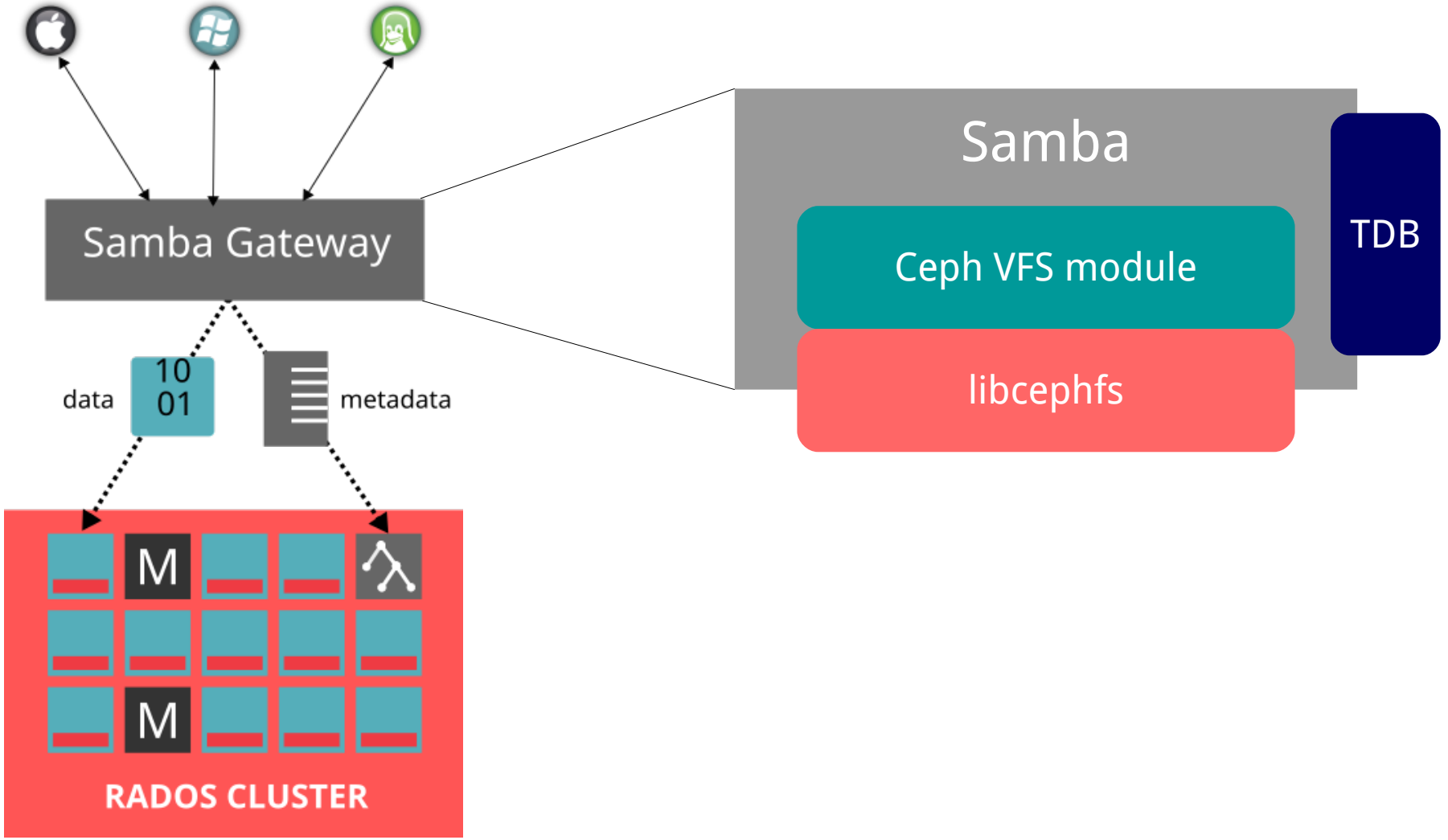
- Windows
  - Reference client
  - Protocol specification publisher
- macOS
  - AFP replaced by SMB2 as default for Mavericks (2013)
- Linux
  - CIFS kernel module and Samba smbclient

Clustered Trivial Database (CTDB)

- Samba persistent state stored in TDB key-value store
- Share some of this state across multiple nodes
  - Cluster consistent database
  - Reliable messaging
- HA features bolted on
  - Monitoring and failover

CTDB

- Nodes participate in election of recovery master
- Recovery master monitors state of cluster
- Performs database recovery if necessary
  - Cluster-wide mutex used to prevent split brain
- “Tickle” clients on IP failover


Samba Gateway
data
10
01
metadata
M
M
RADOS CLUSTER
Samba
Ceph VFS module
libcephfs
TDB

Samba Ceph Integration

- CephFS module for Samba: v*fs_ceph*
  - Maps SMB file and directory I/O to libcephfs API calls
- Static cephx credentials
  - Regardless of Samba authenticated user
- POSIX ACLs

Samba Ceph Integration

- Ceph RADOS clustered mutex helper for CTDB
- Ceph librados service integration *(coming soon)*

Testing

- Samba smbtorture
- cifs.ko fstests
- Interoperability
  - MacOS, Windows, Hyper-V, etc.

Performance

Benchmark Setup

- Ceph setup on 8 nodes
  - 5 OSD nodes – 24 cores – 128 GB RAM
  - 3 MON/MDS nodes – 24 cores – 128 GB RAM
  - 6 OSD daemons per node – Bluestore – SSD/NVME journals
- 10 client nodes
  - 16 cores – 16 GB RAM
- Network interconnect
  - Public network 10Gbit/s
  - Cluster network 100Gbit/s

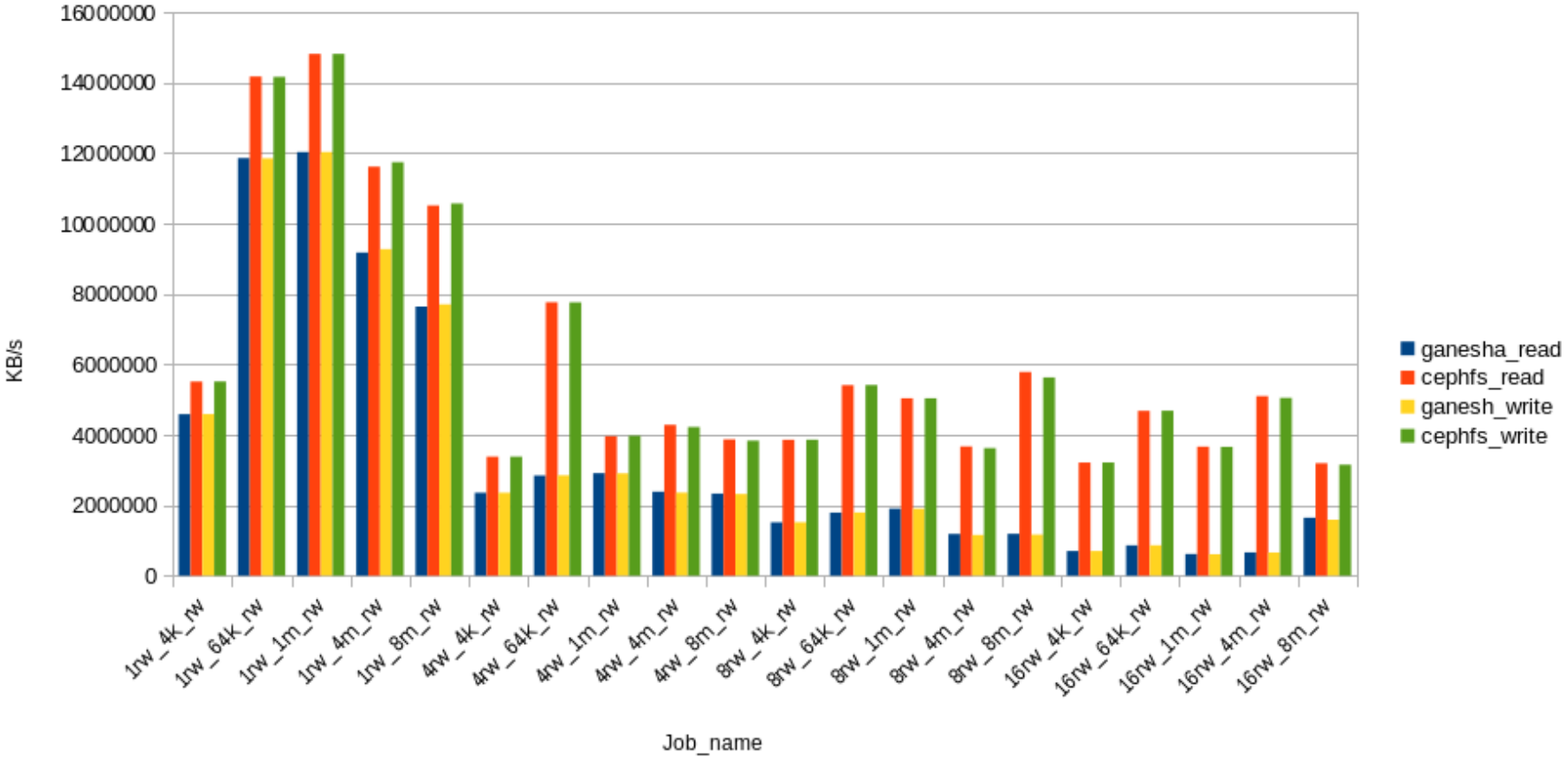
FIO Job

- Read/write data to #SIZE file for #TIME
- 10 client nodes. On each client node, jobs are executed
- A single job is of type:
  - { number_of_workers}rw_{ block_size }_{ op }, where:
    - Number of worker threads:
      - 1, 4, 8, 16
      - Each worker thread creates a file of 1g size
    - Block Sizes:
      - 4k, 64k, 1m , 4m, 8m
    - Op:
      - rw

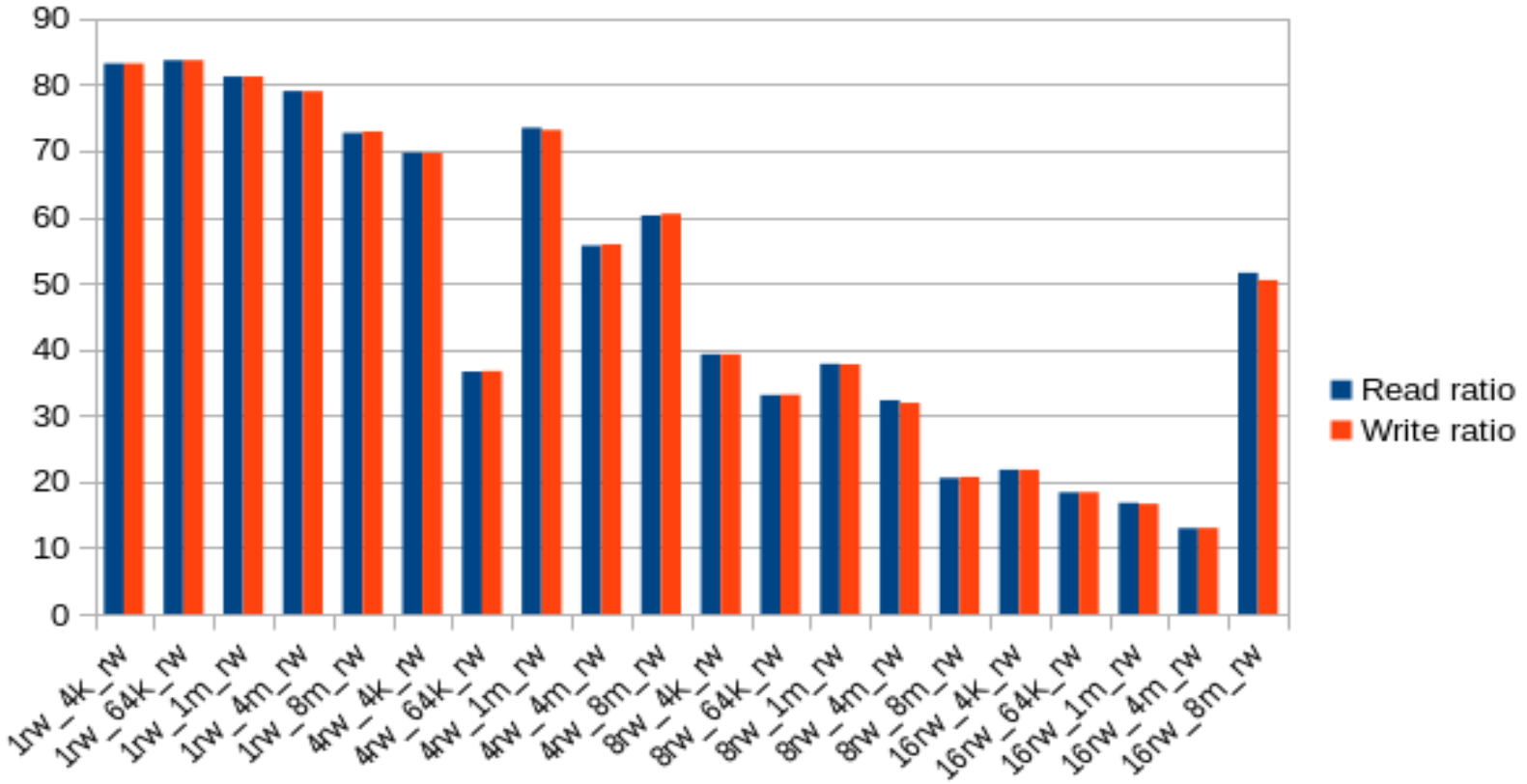
Performance: NFS-Ganesha vs CephFS

- Benchmarking was performed for:
  - NFS-Ganesha v2.5.2
  - Ceph Version 12.2.1
  - Single NFS-Ganesha server
  - NFS version 4.0
  - Read/Write data 1GB file for 2 mins

NFS-Ganesha vs CephFS Kernel client: Aggregated B/W over 10 clients

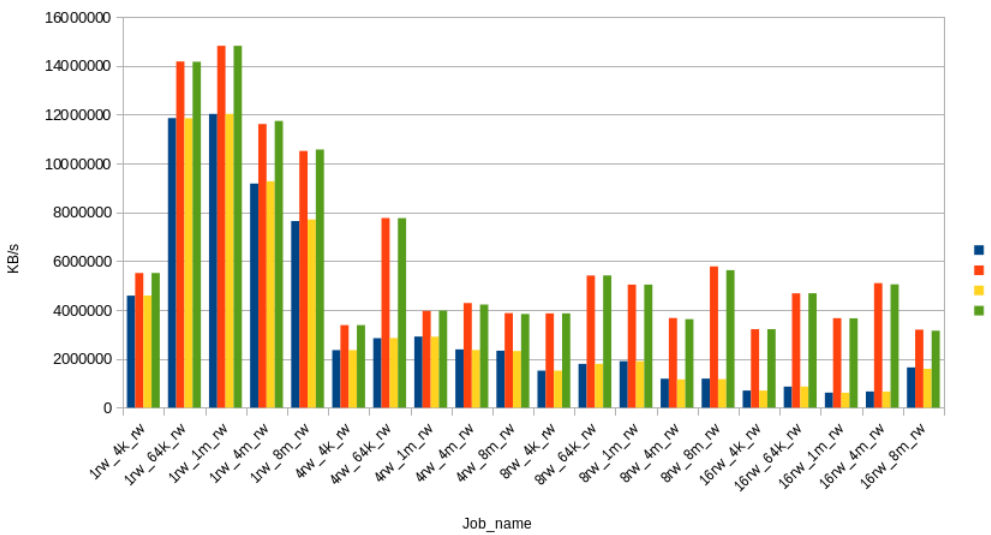


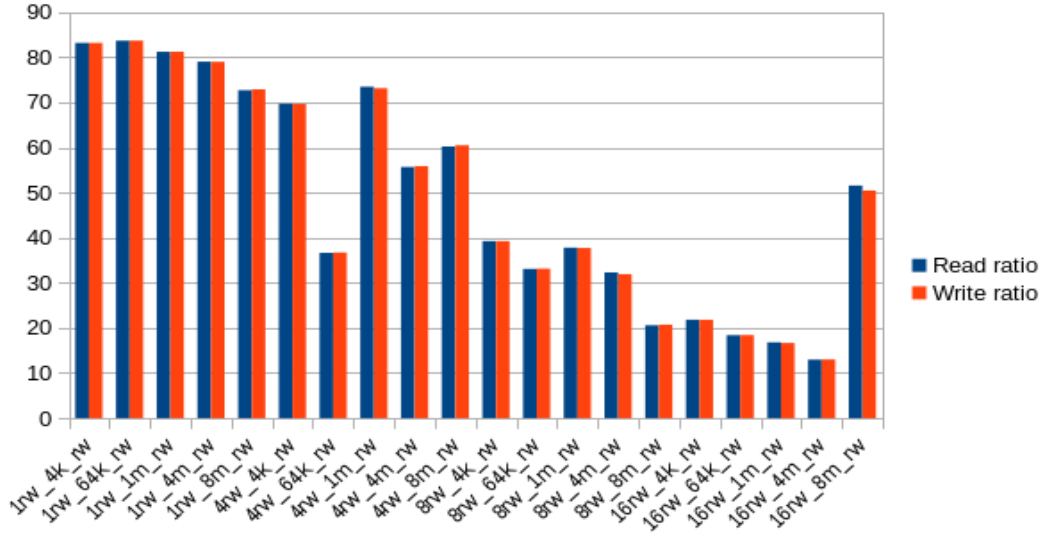
NFS-Ganesha vs CephFS Kernel client: Read/Write ratio B/W comparison (in %)



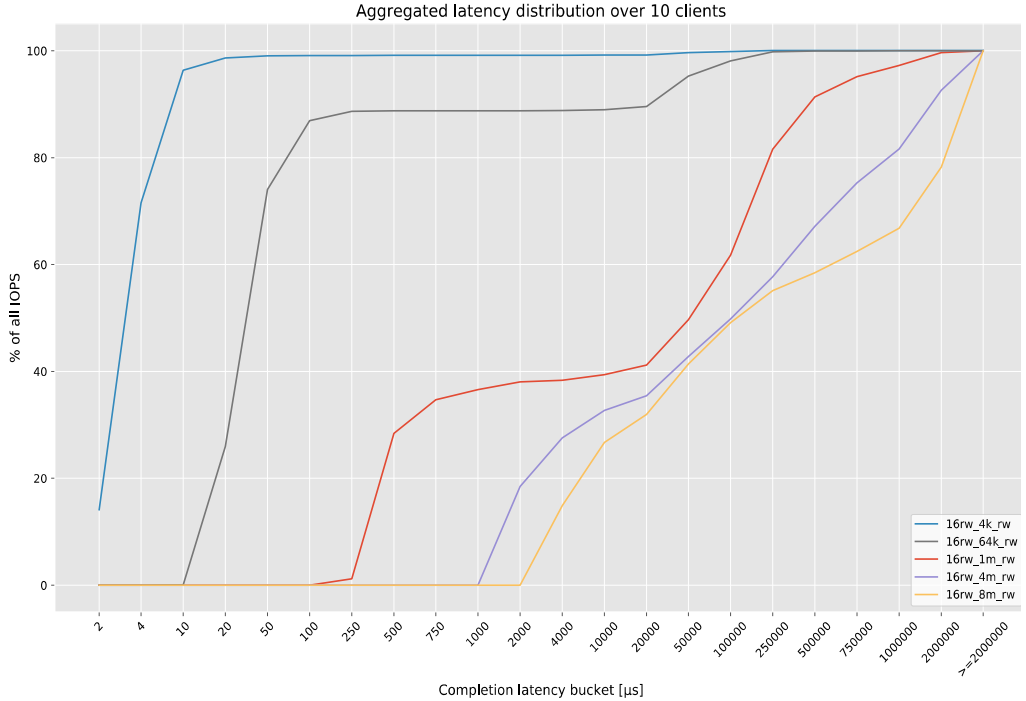
Obeservations

- For single thread, nfs b/w is 80% of cephfs b/w.
- Performance degrades as no. of threads increase
- Single nfs-ganesha server is bottleneck.

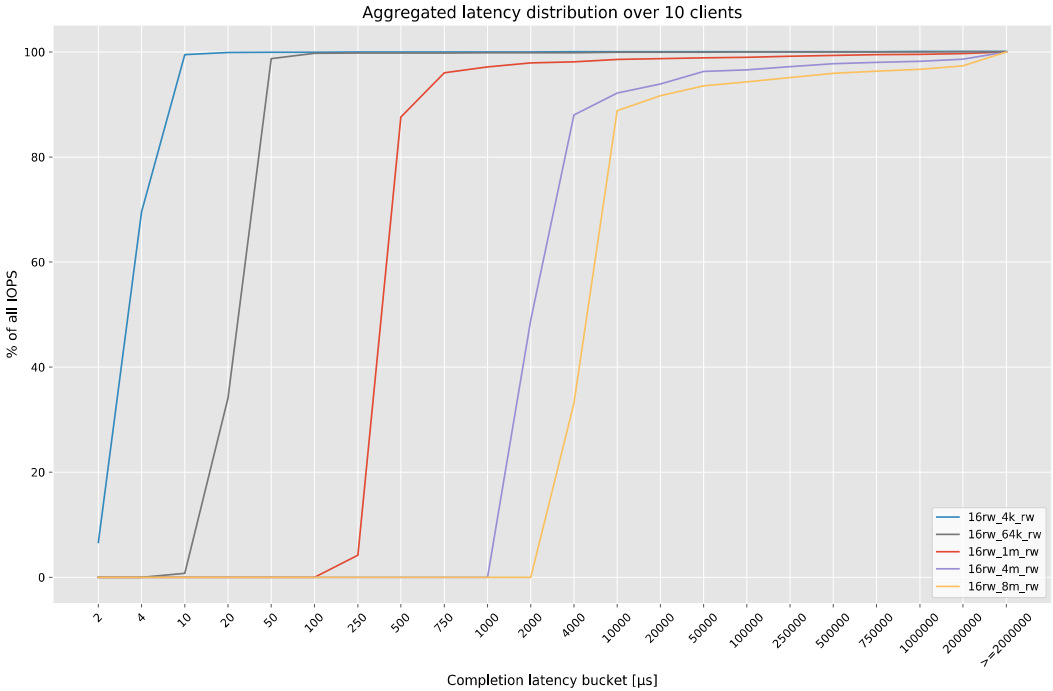




NFS-Ganesha vs CephFS: 16 threads latency



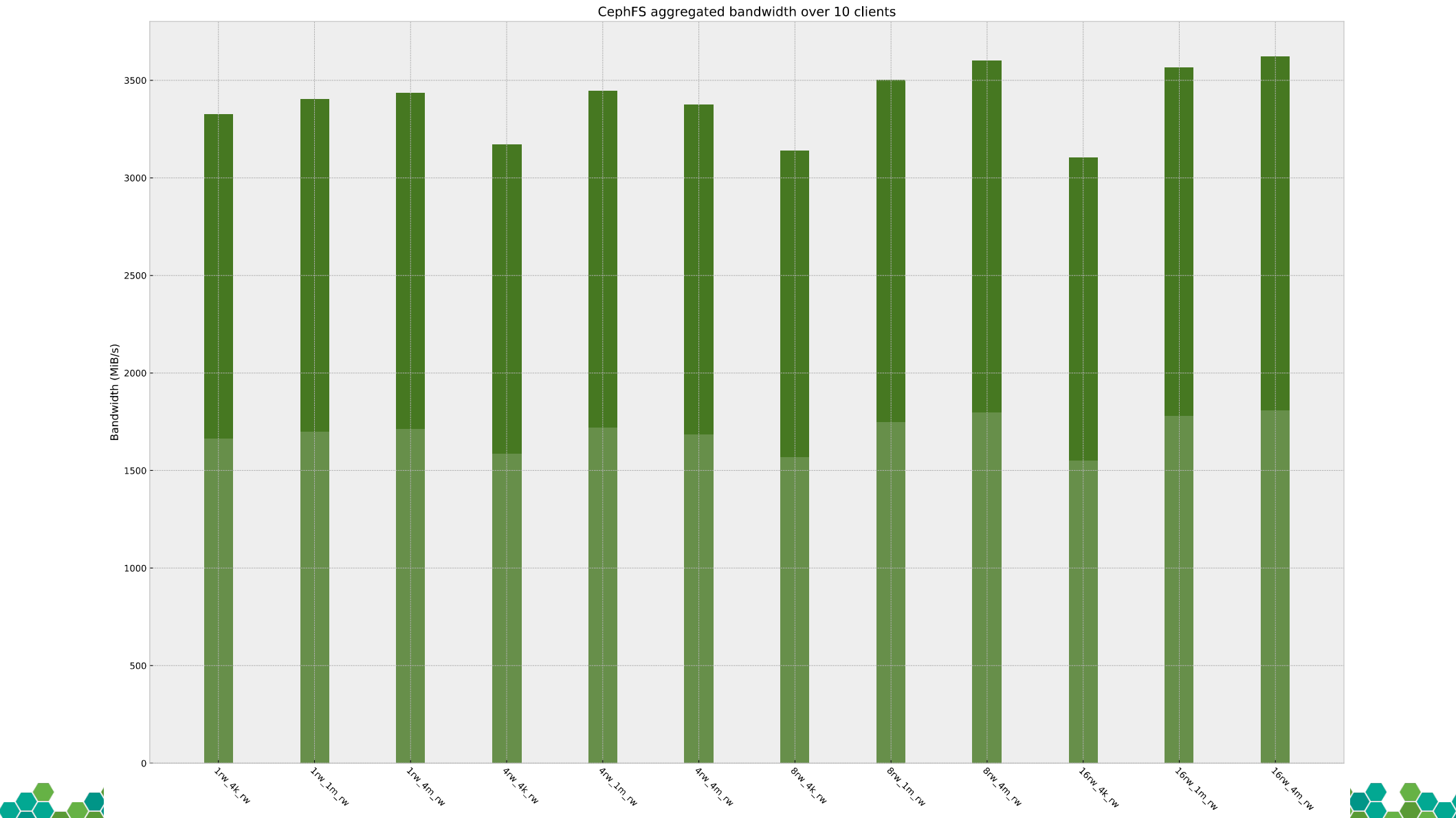
NFS-Ganesha

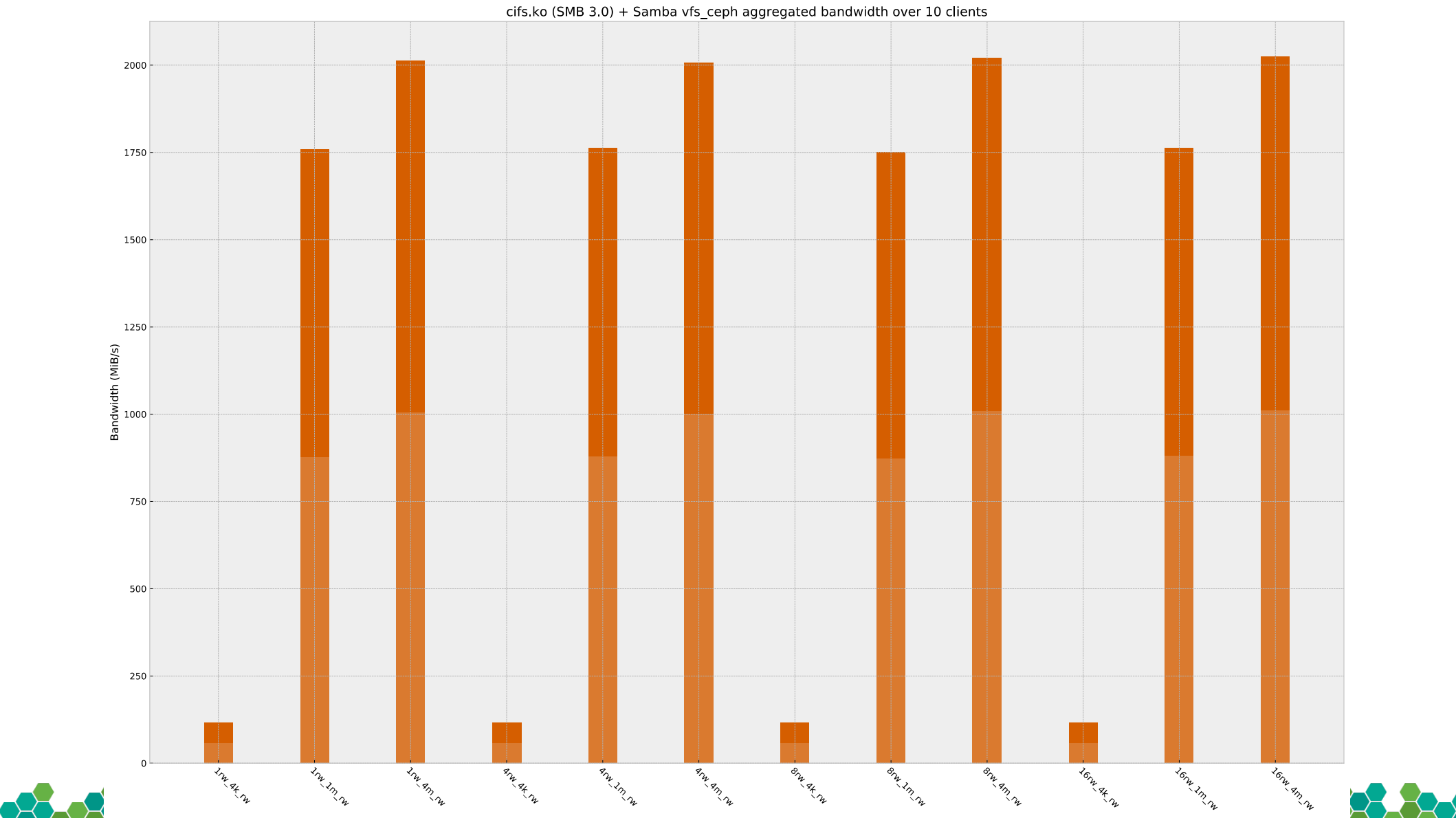


CephFS

Performance: Samba vs CephFS

- Preliminary results!
- Environment:
  - Ceph Version 12.2.2
  - Samba 4.6.9
    - Three Samba gateways
    - *vfs_ceph* with oplocks / leases disabled
    - Non-overlapping share paths
  - Linux cifs.ko client
    - 4.4 kernel with many backports
    - SMB 3.0 mount


CephFS aggregated bandwidth over 10 clients
Bandwidth (MiB/s)
0
500
1000
1500
2000
2500
3000
3500
1rw_4k_rw
1rw_1m_rw
1rw_4m_rw
4rw_4k_rw
4rw_1m_rw
4rw_4m_rw
8rw_4k_rw
8rw_1m_rw
8rw_4m_rw
16rw_4k_rw
16rw_1m_rw
16rw_4m_rw


cifs.ko (SMB 3.0) + Samba vfs_ceph aggregated bandwidth over 10 clients
Bandwidth (MiB/s)
0
250
500
750
1000
1250
1500
1750
2000
1rw_4k_rw
1rw_1m_rw
1rw_4m_rw
4rw_4k_rw
4rw_1m_rw
4rw_4m_rw
8rw_4k_rw
8rw_1m_rw
8rw_4m_rw
16rw_4k_rw
16rw_1m_rw
16rw_4m_rw

Challenges and Future

Challenges

- Cross-protocol client support
  - Shared (NFS, CephFS) ACL model
    - RichACLs or POSIX draft ACLs
  - Coherent client caching
    - Map leases to CephFS *FILE* and *AUTH* capabilities

- Unified authentication and user mapping
  - Use Kerberos / AD for Samba gateway and cephx

Challenges

- libcephfs asynchronous I/O
- Multichannel support
  - Experimental in upstream Samba
  - Not integrated with CTDB
- Automated deployment

Challenges

- Witness protocol
  - Continuous availability of SMB shares
  - Advertise Samba cluster state to clients
  - Transparent client failover
  - Load balancing

Samba: Future

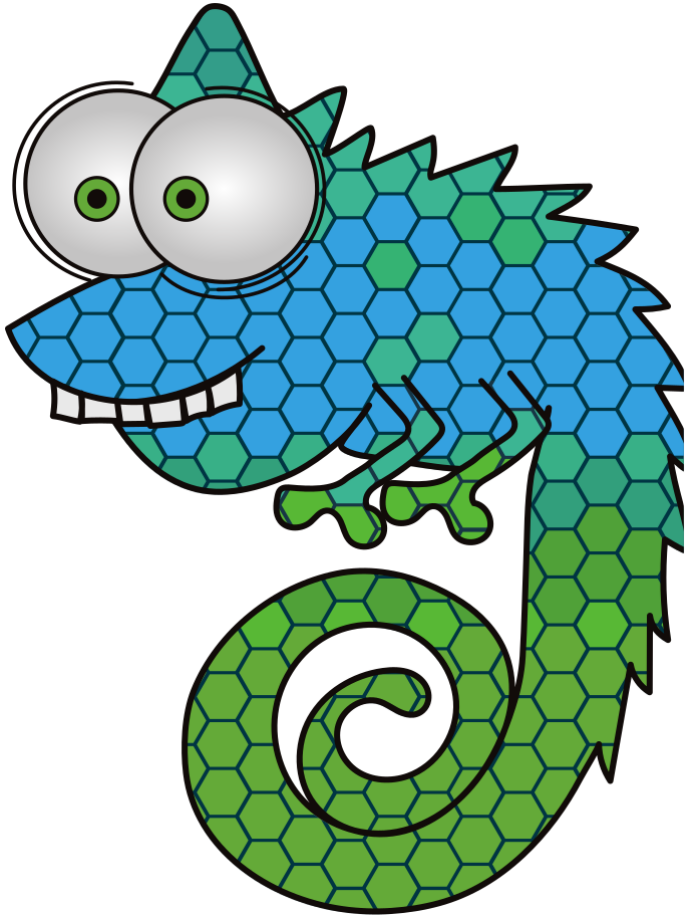
- Ceph backed key-value store for Samba
- Replace or modify CTDB
  - Rocksdb?
  - Samba database API demanding
    - Multiple processes and writers
    - Record locking and transactions

NFS-Ganesha: Future

- clustering without Linux HA
- NFS v4.1 support
- Librados service integration

References

- Samba: https://samba.org/
- CTDB: https://ctdb.samba.org/
- SMB 3.1.1 encryption: https://technet.microsoft.com/en-us/library/dn551363(v=ws.11).aspx
- Multichannel deployment: https://technet.microsoft.com/en-us/library/dn610980(v=ws.11).aspx
- Witness Protocol: http://www.sambaxp.org/archive_data/SambaXP2015-SLIDES/wed/track1/sambaxp2015-wed-track1-Guenther_Deschner-ImplementingTheWitnessProtocolInSamba.pdf
- Samba Multichannel Blocker Bug: https://bugzilla.samba.org/show_bug.cgi?id=11897
- CephFS cache flags: https://jtlayton.wordpress.com/2016/09/01/cephfs-and-the-nfsv4-change-attribute/



Join Us at www.opensuse.org

License

General Disclaimer

Credits

Template
Richard Brown
rbrown@opensuse.org

Design & Inspiration
openSUSE Design Team
http://opensuse.github.io/branding-guidelines/