ceph

NOW and LATER

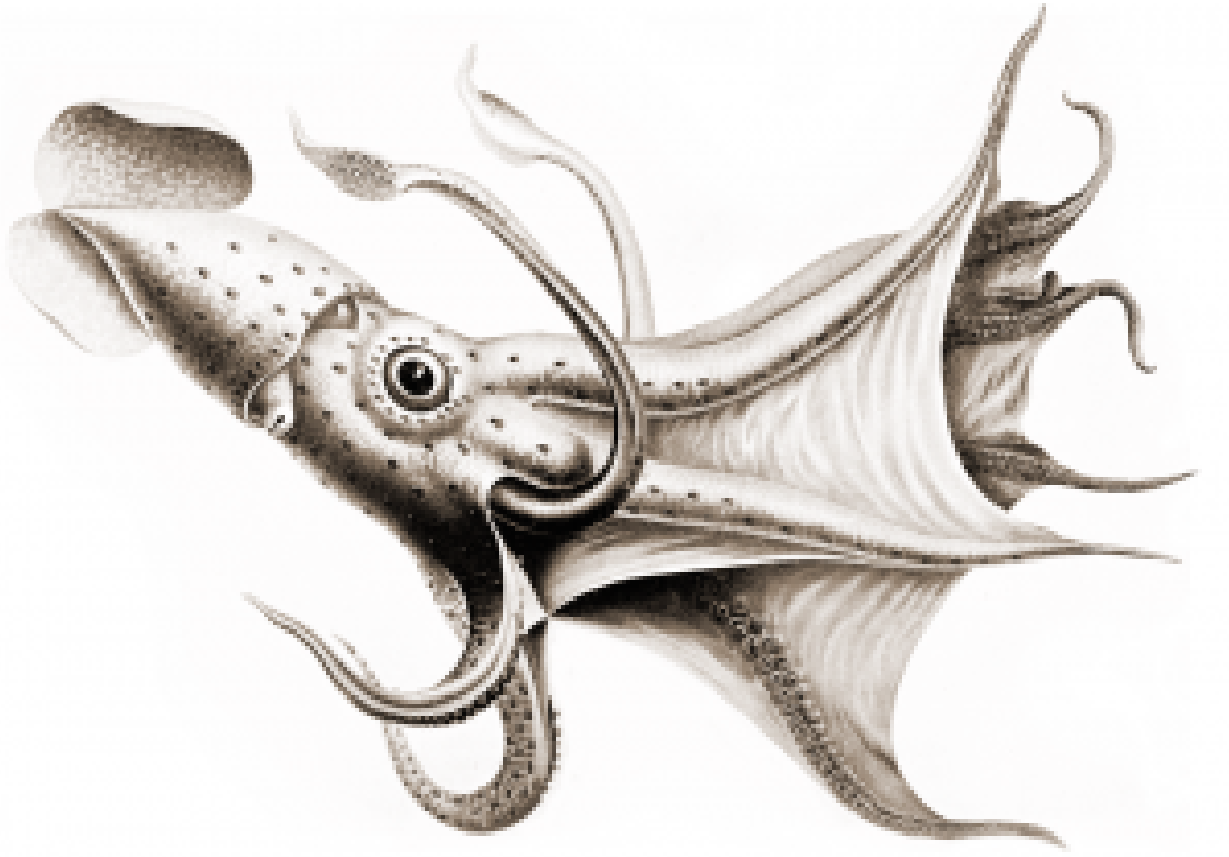
OUR VISION FOR OPEN UNIFIED CLOUD STORAGE

SAGE WEIL – RED HAT

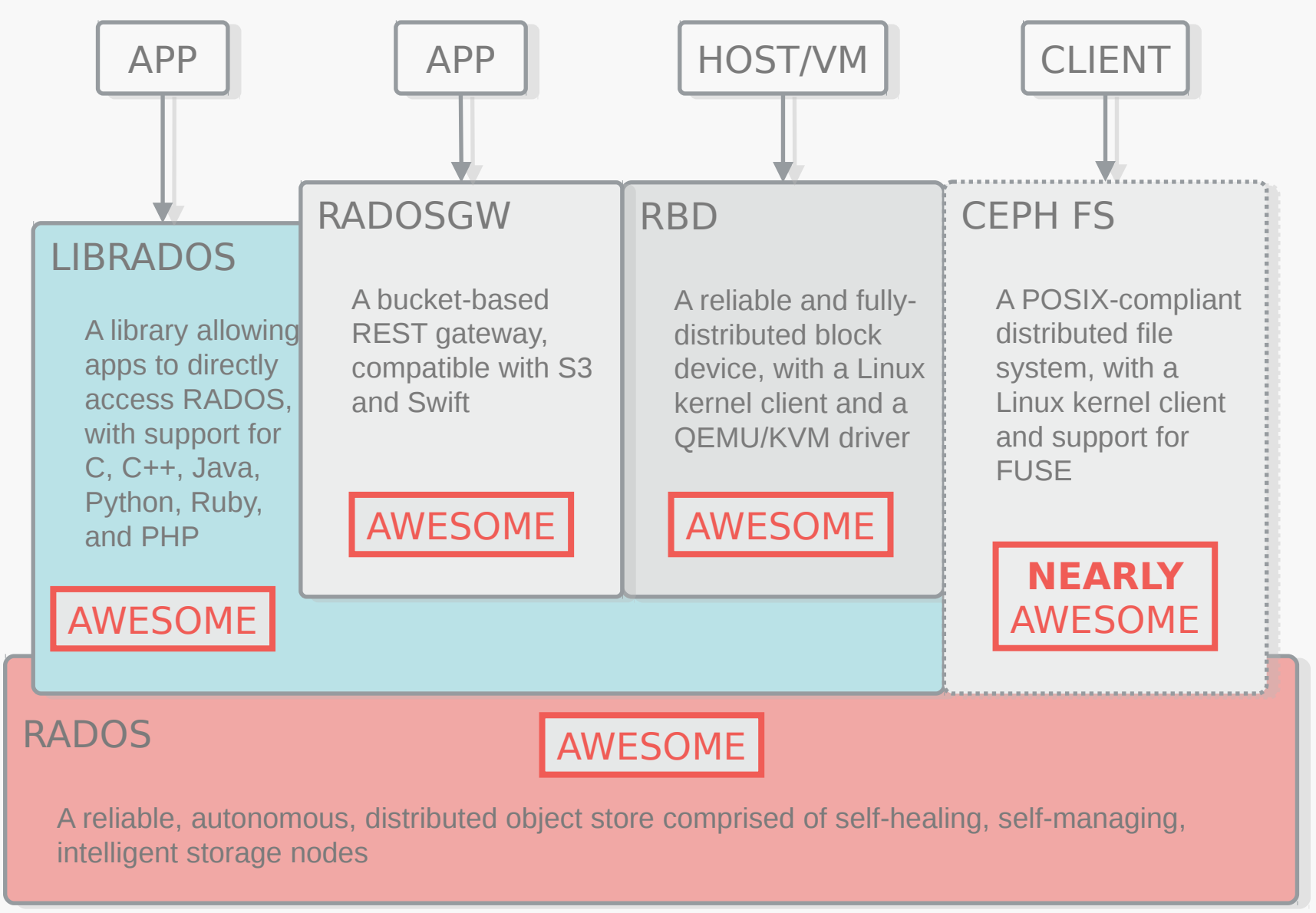
OPENSTACK SUMMIT BARCELONA – 2016.10.27

OUTLINE

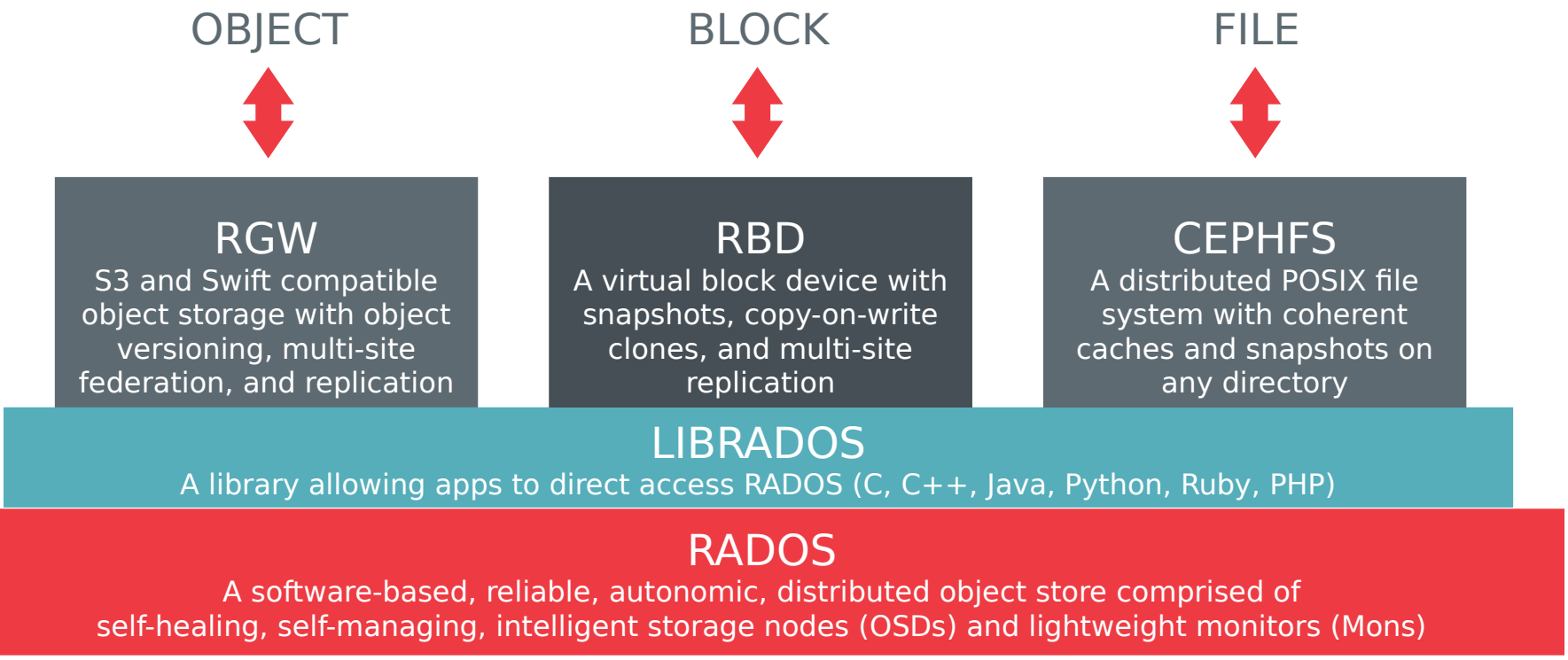
- Now
  - why we develop ceph
  - why it's open
  - why it's unified
  - hardware
  - openstack
  - why not ceph
- Later
  - roadmap highlights
  - the road ahead
  - community



JEWEL


APP
APP
HOST/VM
CLIENT
LIBRADOS
A library allowing apps to directly access RADOS, with support for C, C++, Java, Python, Ruby, and PHP
AWESOME
RADOSGW
A bucket-based REST gateway, compatible with S3 and Swift
AWESOME
RBD
A reliable and fully-distributed block device, with a Linux kernel client and a QEMU/KVM driver
AWESOME
CEPH FS
A POSIX-compliant distributed file system, with a Linux kernel client and support for FUSE
NEARLY AWESOME
RADOS
AWESOME
A reliable, autonomous, distributed object store comprised of self-healing, self-managing, intelligent storage nodes

2016 = FULLY AWESOME



WHAT IS CEPH ALL ABOUT

- Distributed storage
- All components **scale horizontally**
- No single point of failure
- Software
- **Hardware agnostic**, commodity hardware
- Object, block, and file in a single cluster
- Self-manage whenever possible
- **Open source** (LGPL)

WHY OPEN SOURCE

- Avoid software **vendor lock-in**
- Avoid hardware lock-in
  - and enable hybrid deployments
- Lower total cost of ownership

- Transparency
- Self-support
- Add, improve, or extend

WHY UNIFIED STORAGE

- Simplicity of deployment
  - single cluster serving all APIs
- Efficient utilization of storage
- Simplicity of management
  - single set of management skills, staff, tools

...is that really true?

WHY UNIFIED STORAGE

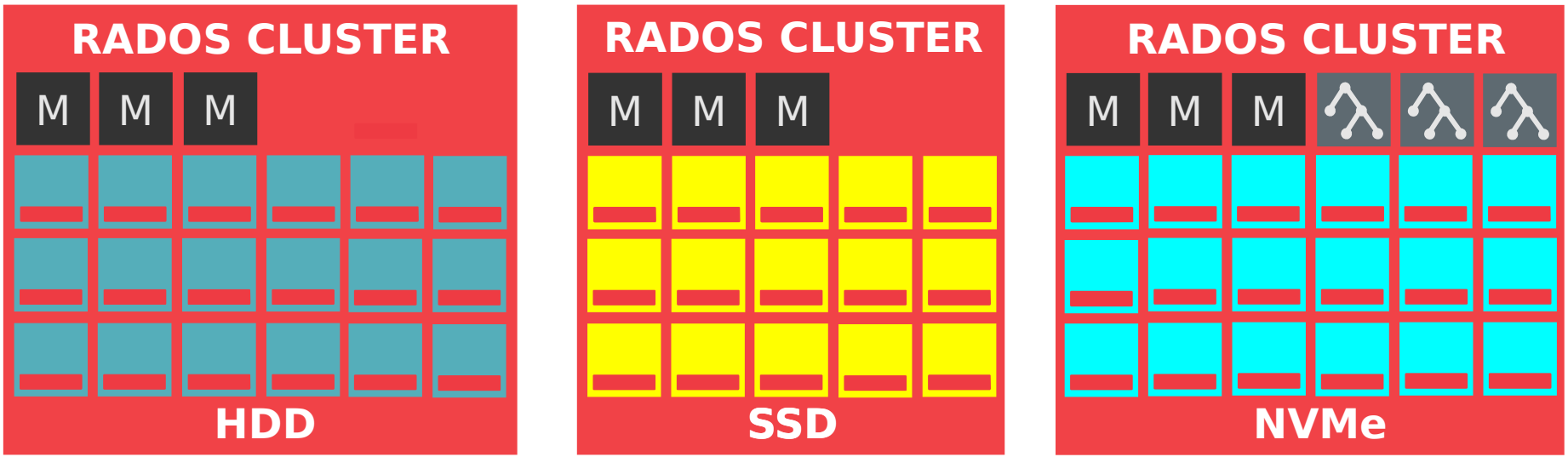
- Simplicity of deployment
  - single cluster serving all APIs
- Efficient utilization of space

- Only really true for small clusters
  - Large deployments optimize for workload

- Simplicity of management
  - single set of skills, staff, tools

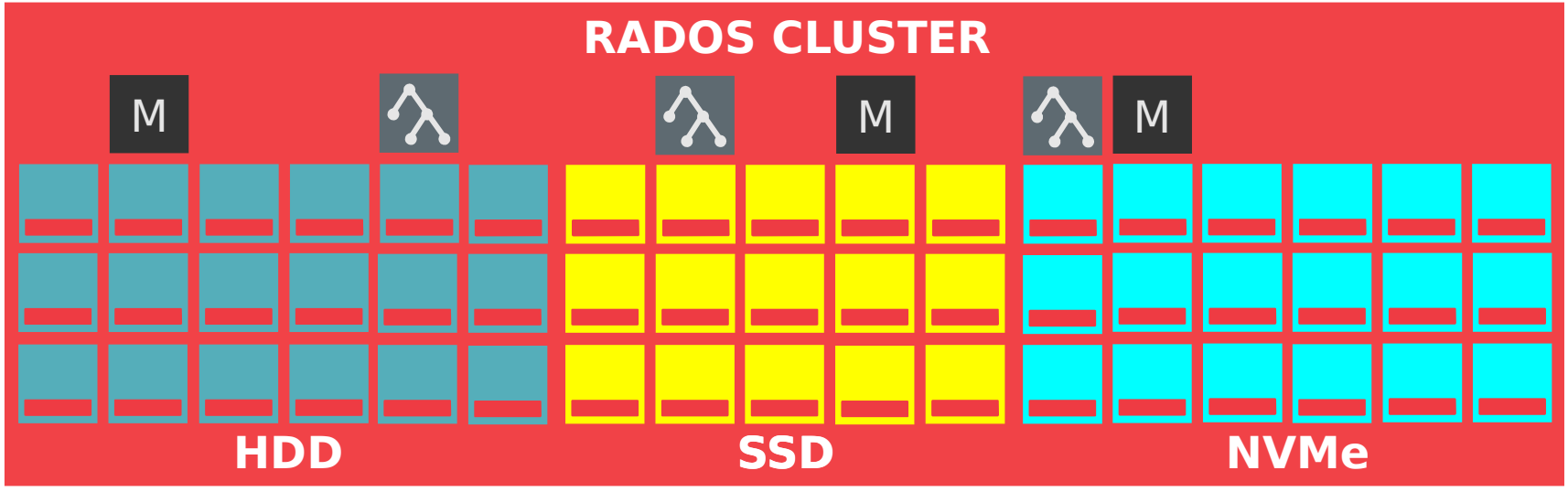
- Functionality in RADOS exploited by all
  - Erasure coding, tiering, performance, monitoring

SPECIALIZED CLUSTERS



- Multiple Ceph clusters per-use case

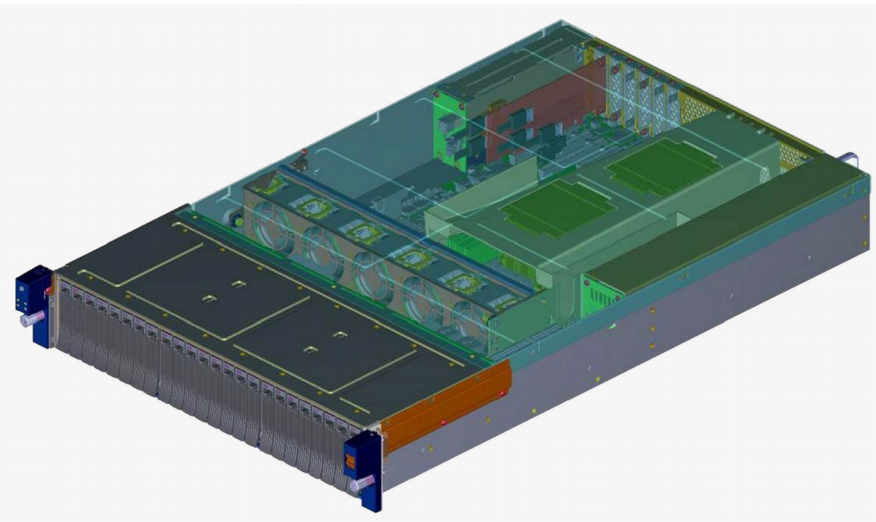
HYBRID CLUSTER



- Single Ceph cluster
- Separate RADOS pools for different storage hardware types

CEPH ON FLASH

- Samsung
  - up to 153 TB in 2u
  - 700K IOPS, 30 GB/s



- SanDisk
  - up to 512 TB in 3u
  - 780K IOPS, 7 GB/s

CEPH ON SSD+HDD

- SuperMicro
- Dell
- HP
- Quanta / QCT
- Fujitsu (Eternus)
- Penguin OCP

CEPH ON HDD (LITERALLY)

- Microserver on 8TB He WD HDD
- 1 GB RAM
- 2 core 1.3 GHz Cortex A9
- 2x 1GbE SGMII instead of SATA

- 504 OSD 4 PB cluster
  - SuperMicro 1048-RT chassis
- Next gen will be aarch64

FOSS → FAST AND OPEN INNOVATION

- Free and open source software (FOSS)
  - enables innovation in software
  - ideal testbed for new hardware architectures

- Proprietary software platforms are full of friction
  - harder to modify, experiment with closed-source code
  - require business partnerships and NDAs to be negotiated
  - dual investment (of time and/or $$$) from both parties

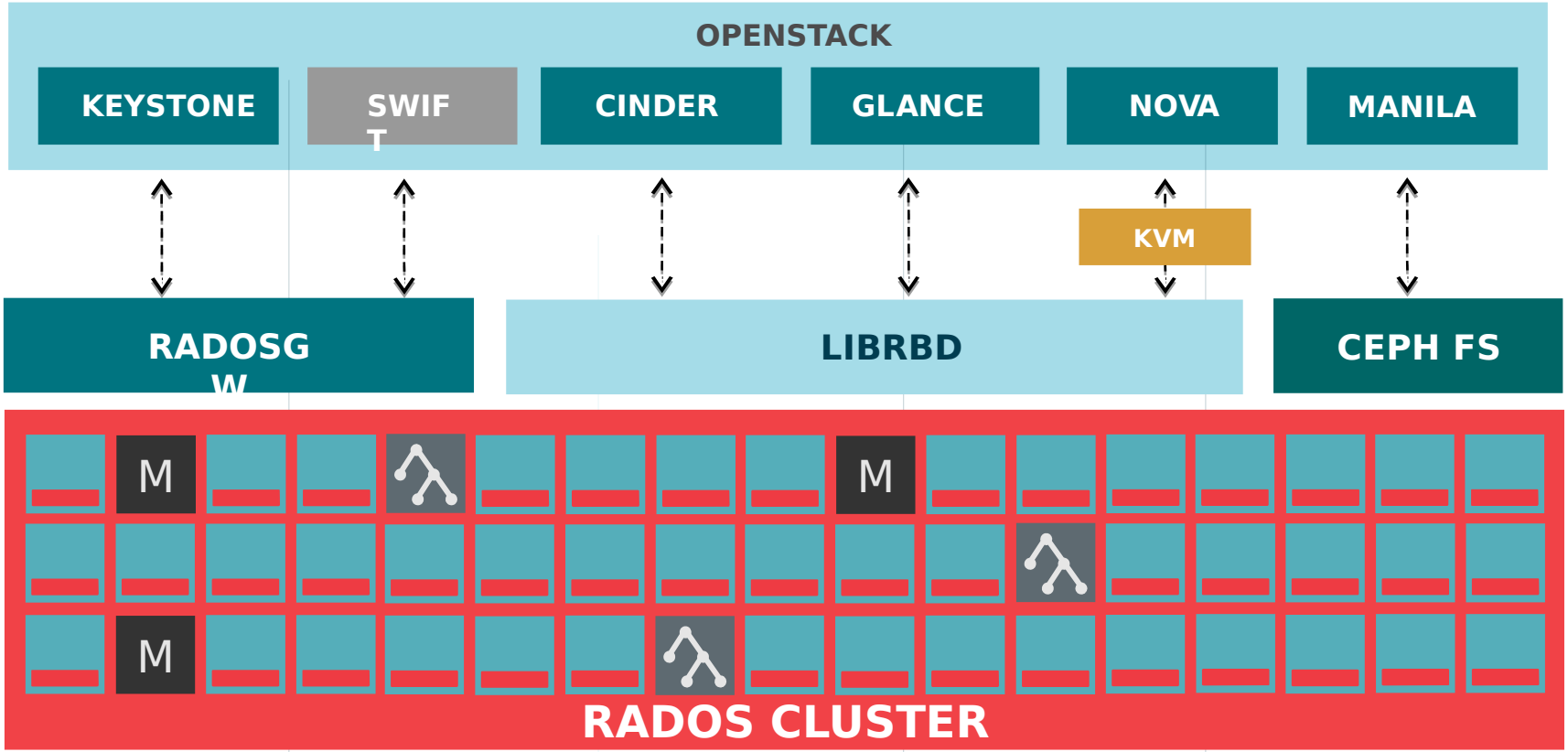
PERSISTENT MEMORY

- Persistent (non-volatile) RAM is coming
  - e.g., 3D X-Point from Intel/Micron any year now
  - (and NV-DIMMs are a bit $$$ but already here)
- These are going to be
  - fast (~1000x), dense (~10x), high-endurance (~1000x)
  - expensive (~1/2 the cost of DRAM)
  - disruptive
- Intel
  - PMStore – OSD prototype backend targetting 3D-Xpoint
  - NVM-L (pmem.io/nvml)

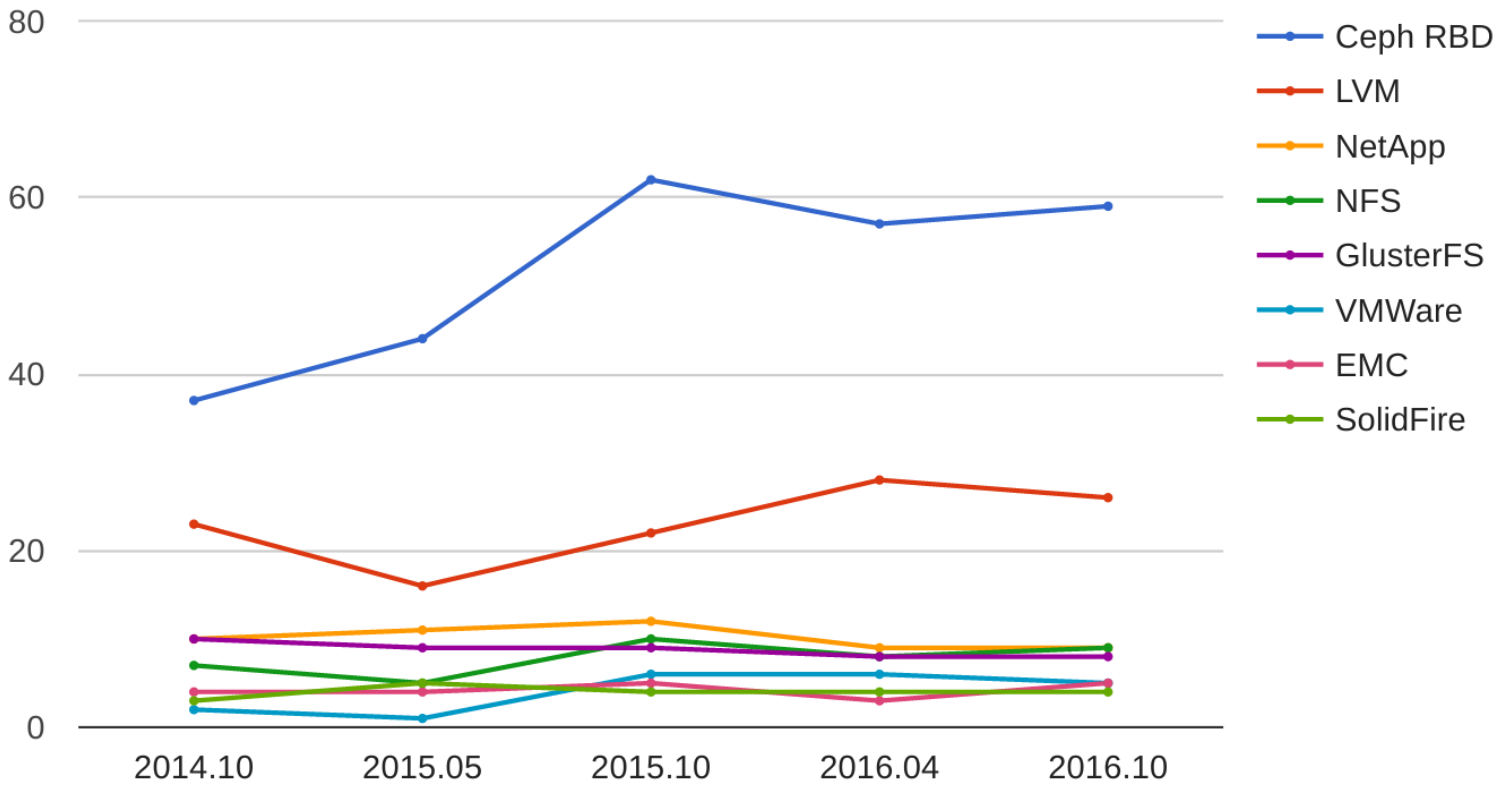
CREATE THE **ECOSYSTEM**
TO BECOME THE **LINUX**
OF DISTRIBUTED **STORAGE**

OPEN COLLABORATIVE GENERAL PURPOSE

CEPH + OPENSTACK



CINDER DRIVER USAGE

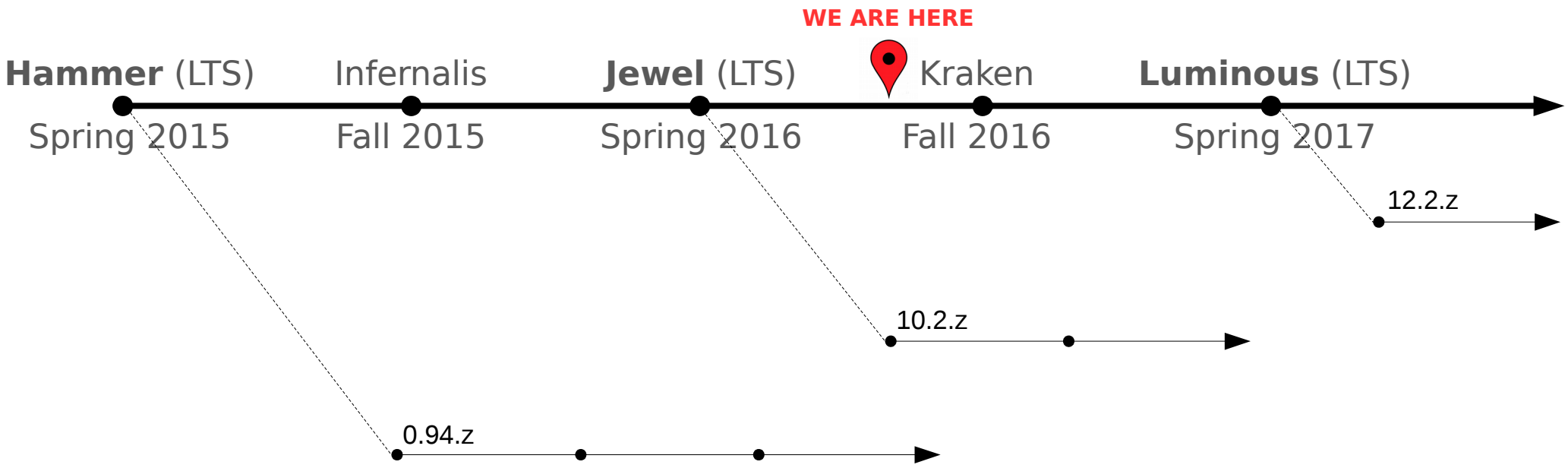


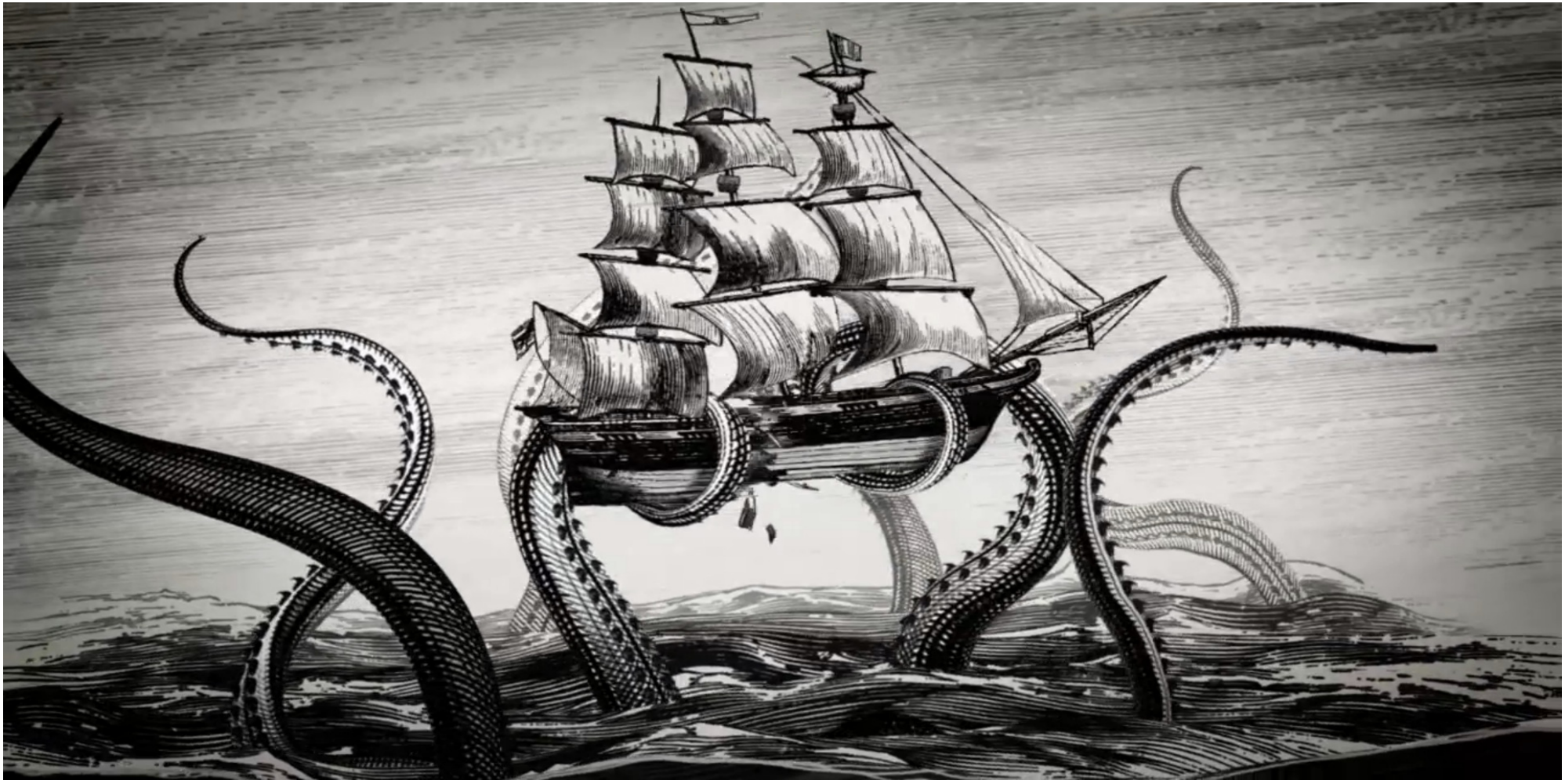
WHY NOT CEPH?

- inertia
- performance
- functionality
- stability
- ease of use

WHAT ARE WE **DOING** ABOUT IT?

RELEASE CADENCE

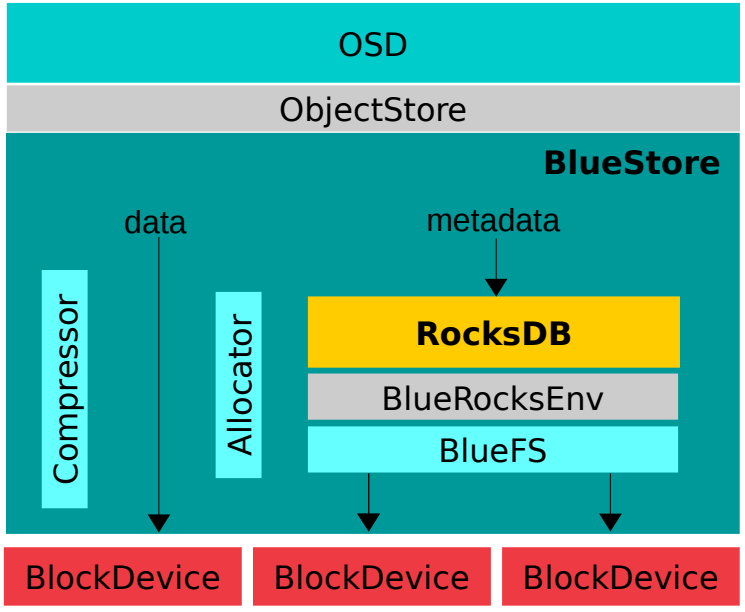




KRAKEN

BLUESTORE

- BlueStore = **Bl**ock + N**ewStore**
  - key/value database (RocksDB) for metadata
  - all data written directly to raw block device(s)
  - can combine HDD, SSD, NVMe, NVRAM
- Full data checksums (crc32c, xxhash)
- Inline compression (zlib, snappy, zstd)
- ~2x faster than FileStore
  - better parallelism, efficiency on fast devices
  - no double writes for data
  - performs well with very small SSD journals



BLUESTORE – WHEN CAN HAZ?

- Current master
  - nearly finalized disk format, good performance
- Kraken
  - stable code, stable disk format
  - probably still flagged 'experimental'
- Luminous
  - fully stable and ready for broad usage
  - maybe the default... we will see how it goes

- Migrating from FileStore
  - evacuate and fail old OSDs
  - reprovision with BlueStore
  - let Ceph recovery normally

ASYNCMESSENGER

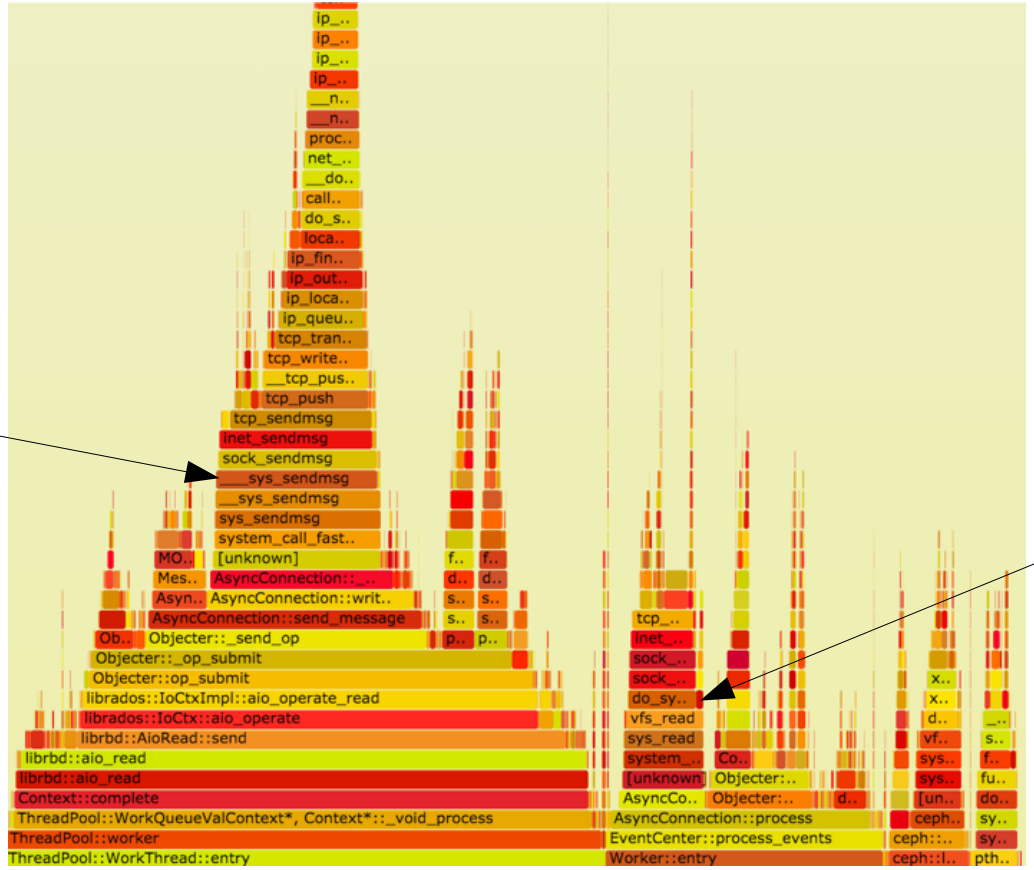
- New implementation of network layer
  - replaces aging SimpleMessenger
  - fixed size thread pool (vs 2 threads per socket)
  - scales better to larger clusters
  - more healthy relationship with tcmalloc
  - now the default!

- Pluggable backends
  - PosixStack – Linux sockets, TCP (default, supported)
  - Two experimental backends!

X·SKY

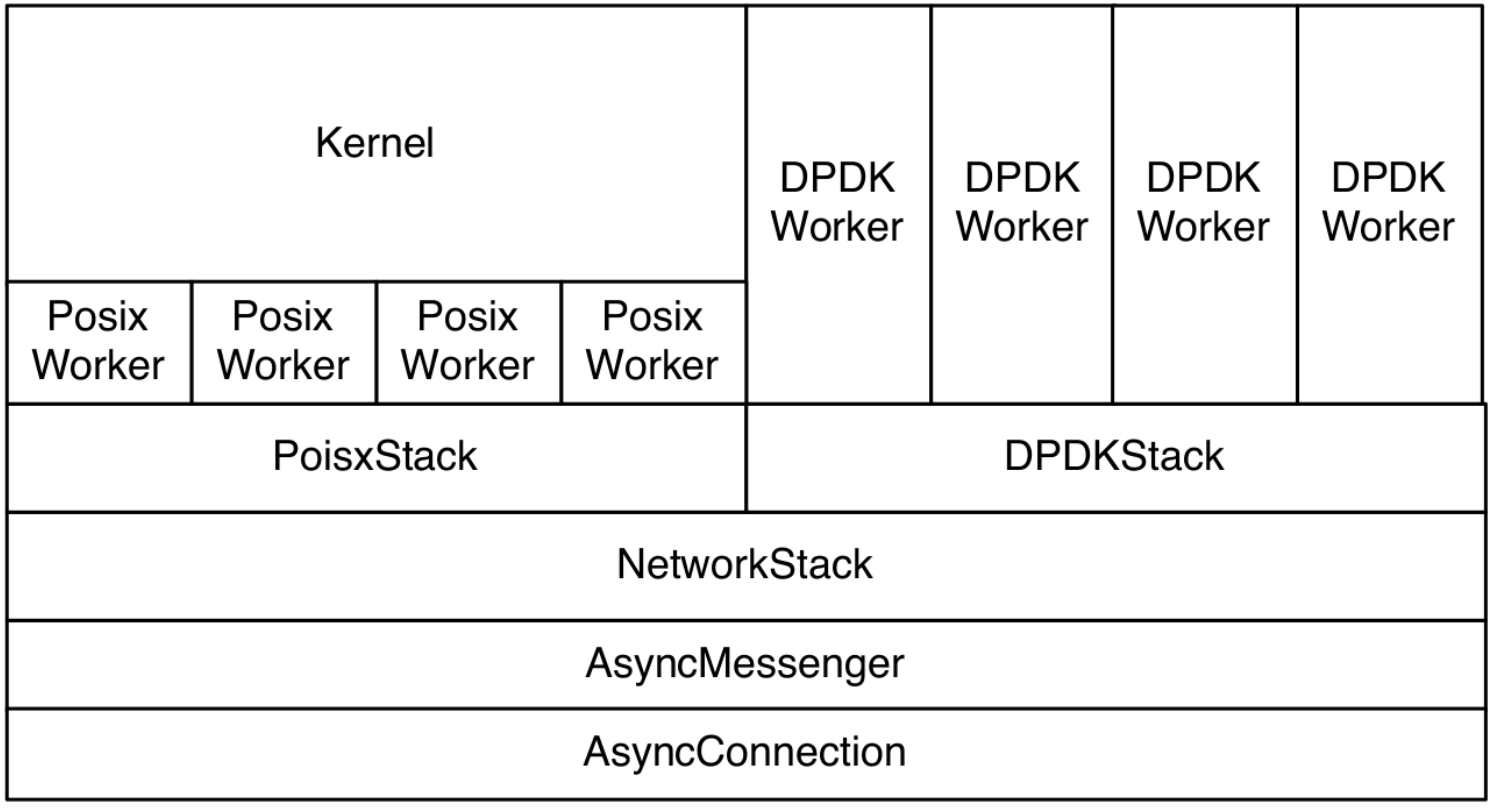
TCP OVERHEAD IS SIGNIFICANT

writes



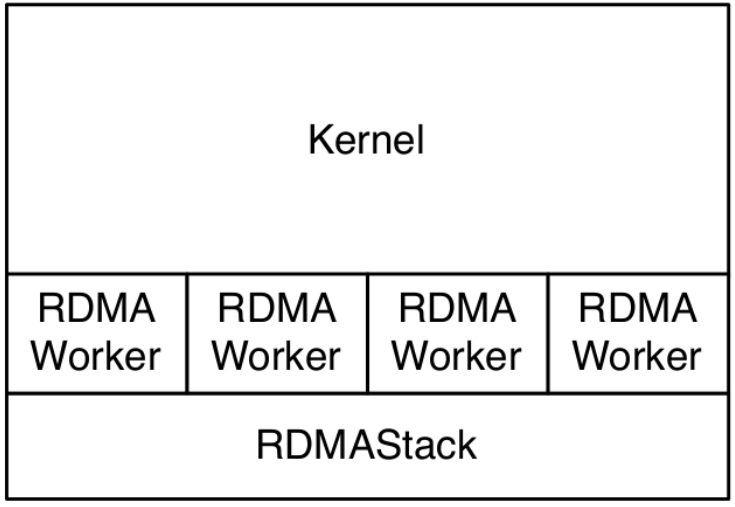
reads

DPDK STACK


Kernel
DPDK Worker
DPDK Worker
DPDK Worker
DPDK Worker
Posix Worker
Posix Worker
Posix Worker
Posix Worker
PoisxStack
DPDKStack
NetworkStack
AsyncMessenger
AsyncConnection

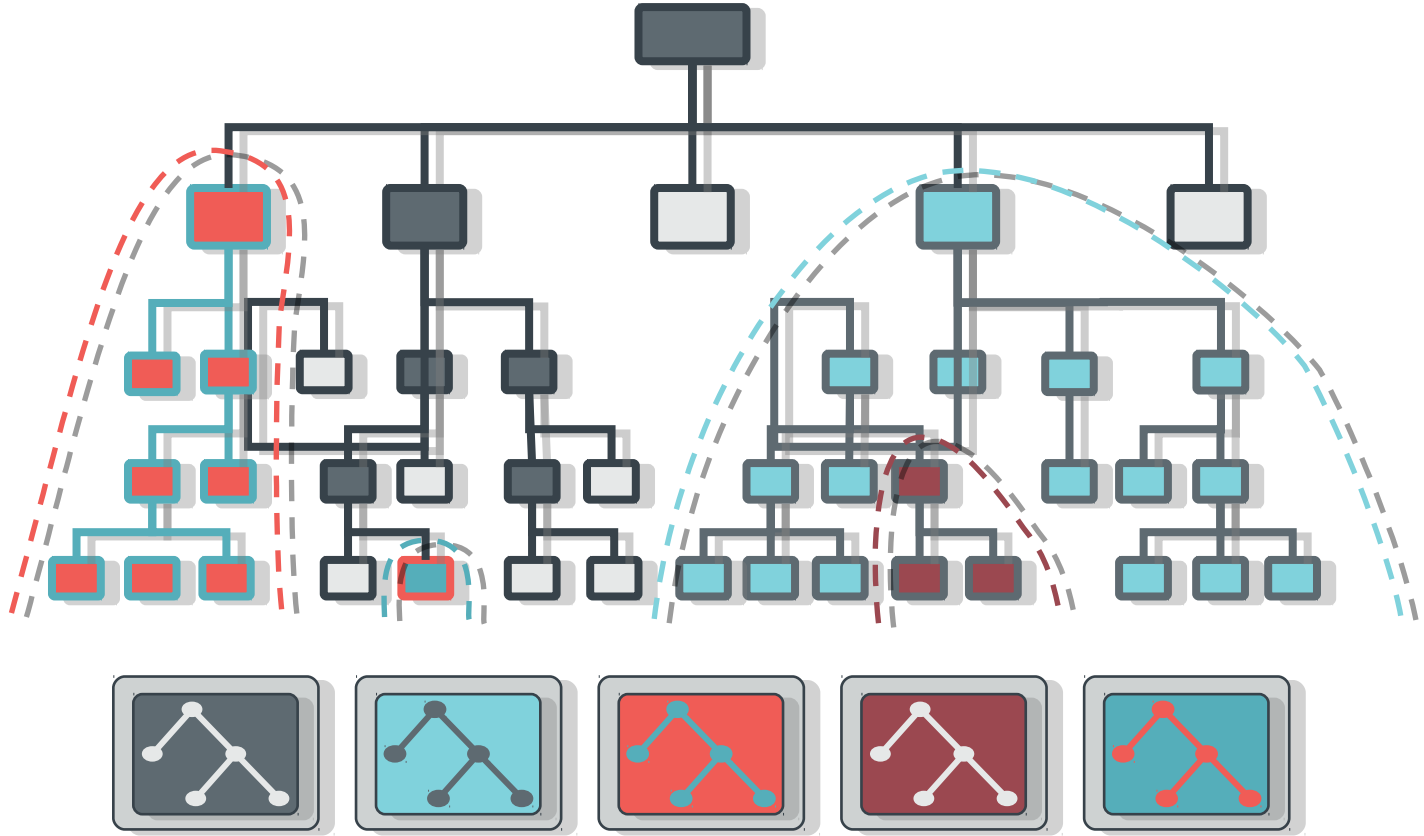
RDMA STACK

- RDMA for data path (ibverbs)
- Keep TCP for control path
- Working prototype in ~1 month



LUMINOUS

MULTI-MDS CEPHFS



ERASURE CODE OVERWRITES

- Current EC RADOS pools are append only
  - simple, stable suitable for RGW, or behind a cache tier
- EC overwrites will allow RBD and CephFS to consume EC pools directly
- It's hard
  - implementation requires two-phase commit
  - complex to avoid full-stripe update for small writes
  - relies on efficient implementation of “move ranges” operation by BlueStore
- It will be huge
  - tremendous impact on TCO
  - make RBD great again

CEPH-MGR

- ceph-mon monitor daemons currently do a lot
  - more than they need to (PG stats to support things like 'df')
  - this limits cluster scalability
- ceph-mgr moves non-critical metrics into a separate daemon
  - that is more efficient
  - that can stream to graphite, influxdb
  - that can efficiently integrate with external modules (even Python!)
- Good host for
  - integrations, like Calamari REST API endpoint
  - coming features like 'ceph top' or 'rbd top'
  - high-level management functions and policy

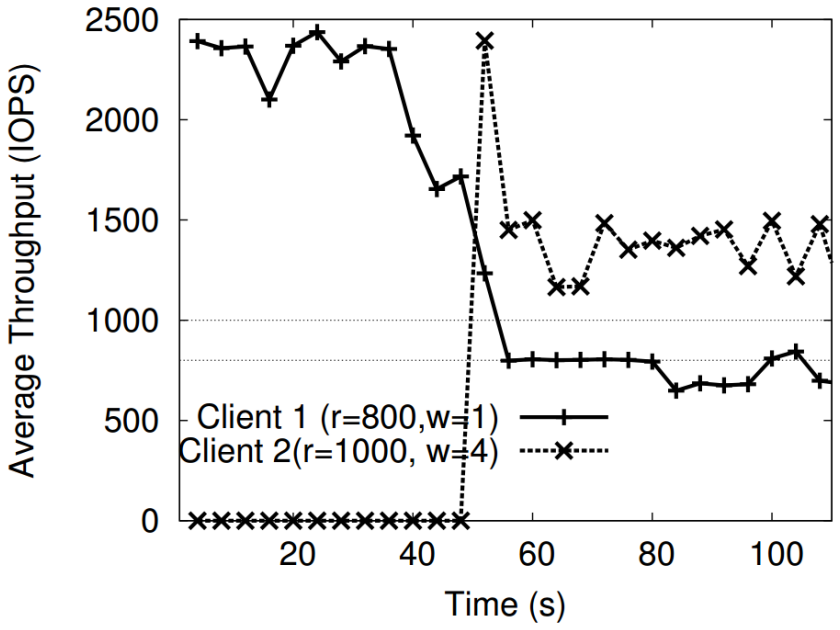
???

(time for new iconography)

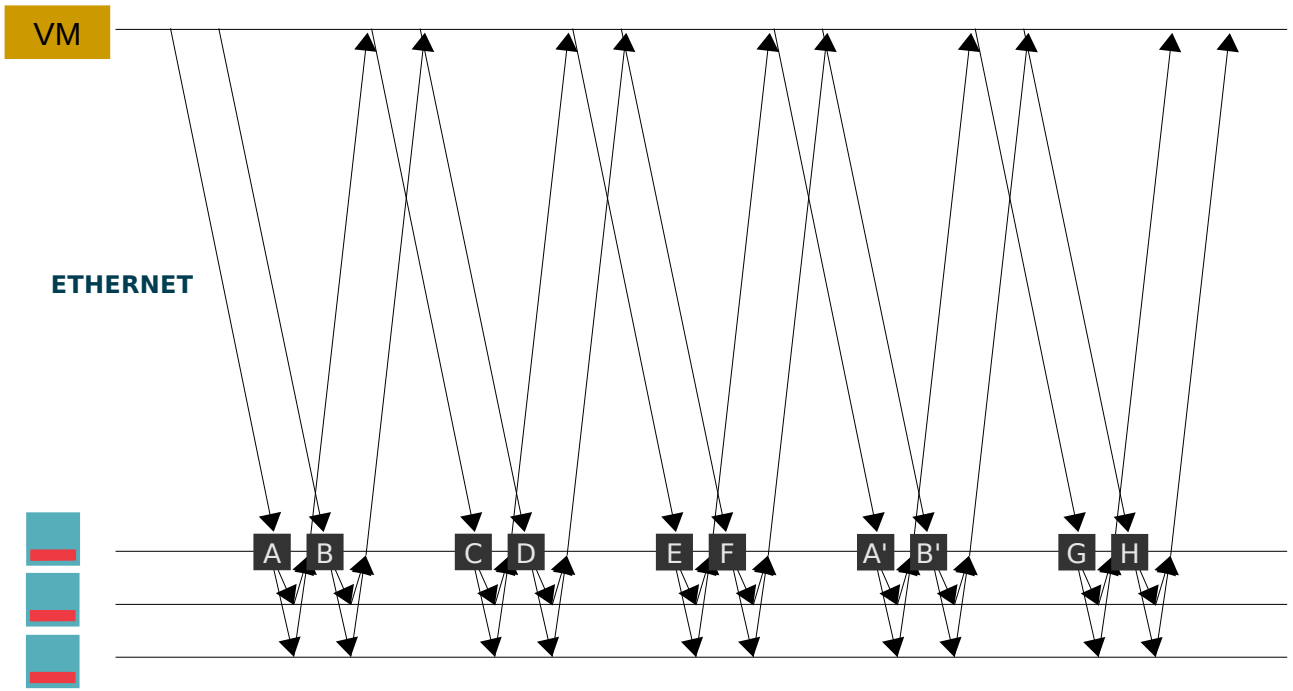
redhat. SUSE

QUALITY OF SERVICE

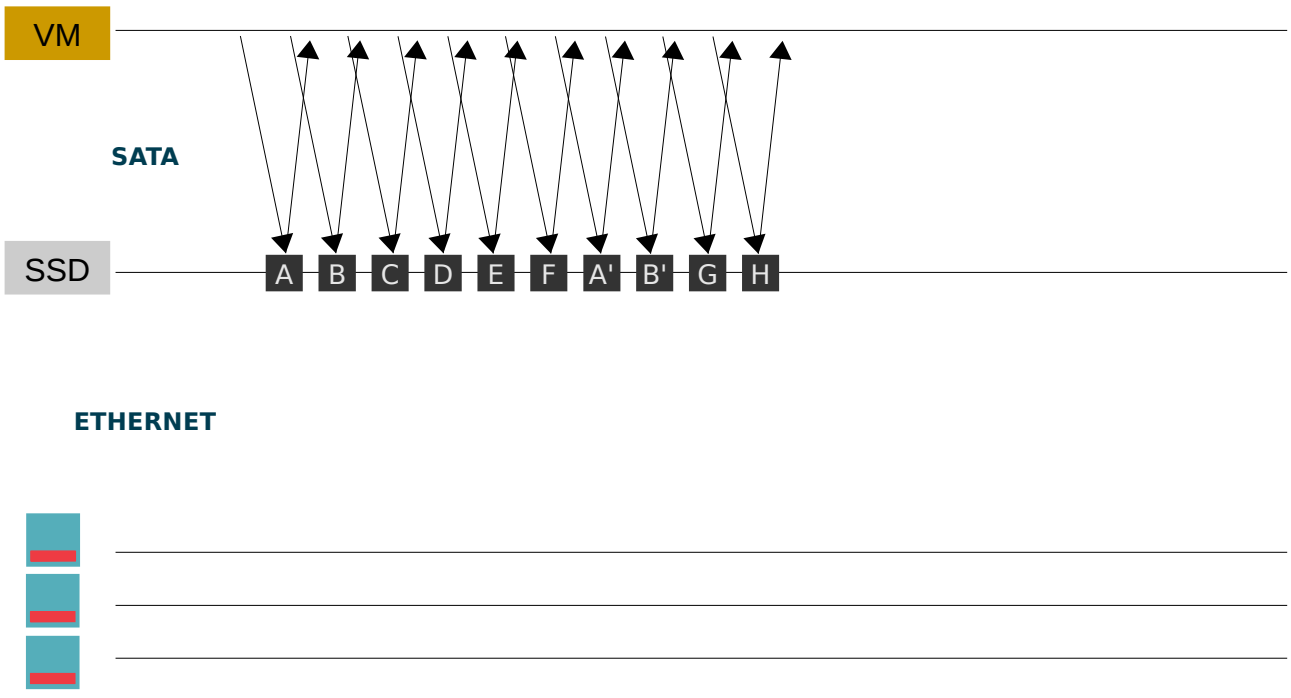
- Set policy for both
  - **reserved**/minimum IOPS
  - **proportional sharing** of excess capacity
- by
  - type of IO (client, scrub, recovery)
  - pool
  - client (e.g., VM)
- Based on mClock paper from OSDI'10
  - IO scheduler
  - distributed enforcement with cooperating clients



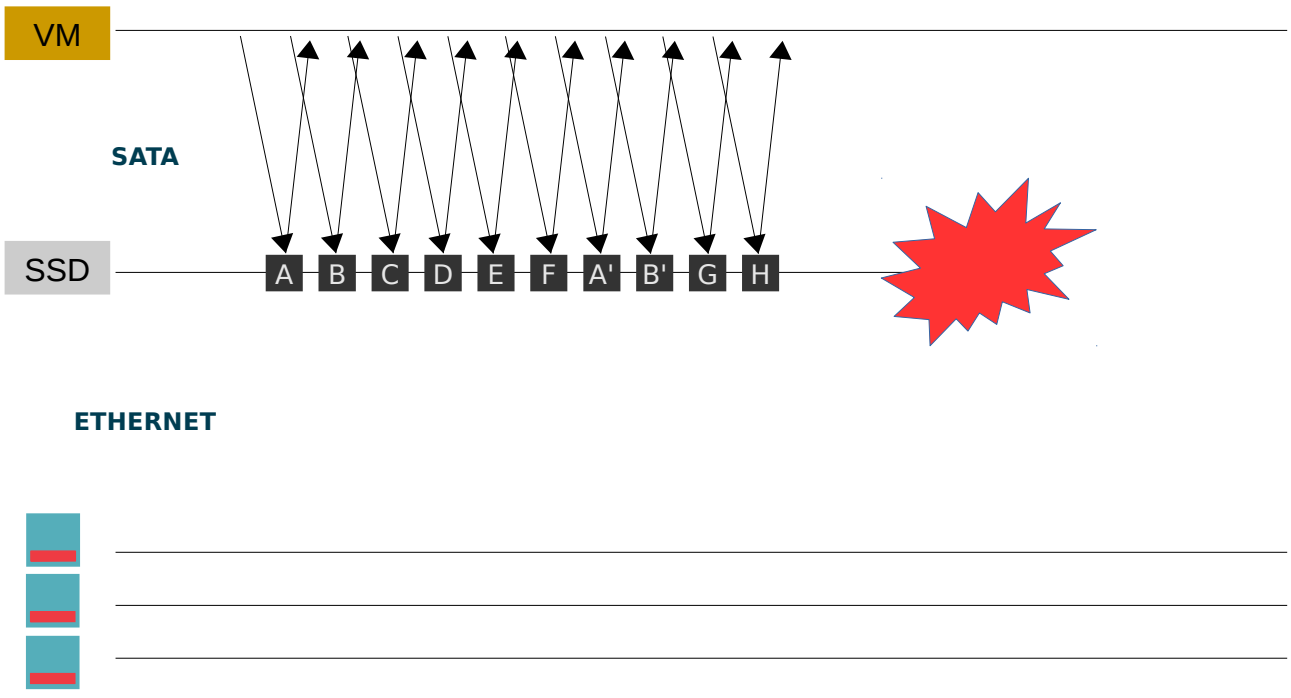
SHARED STORAGE → LATENCY



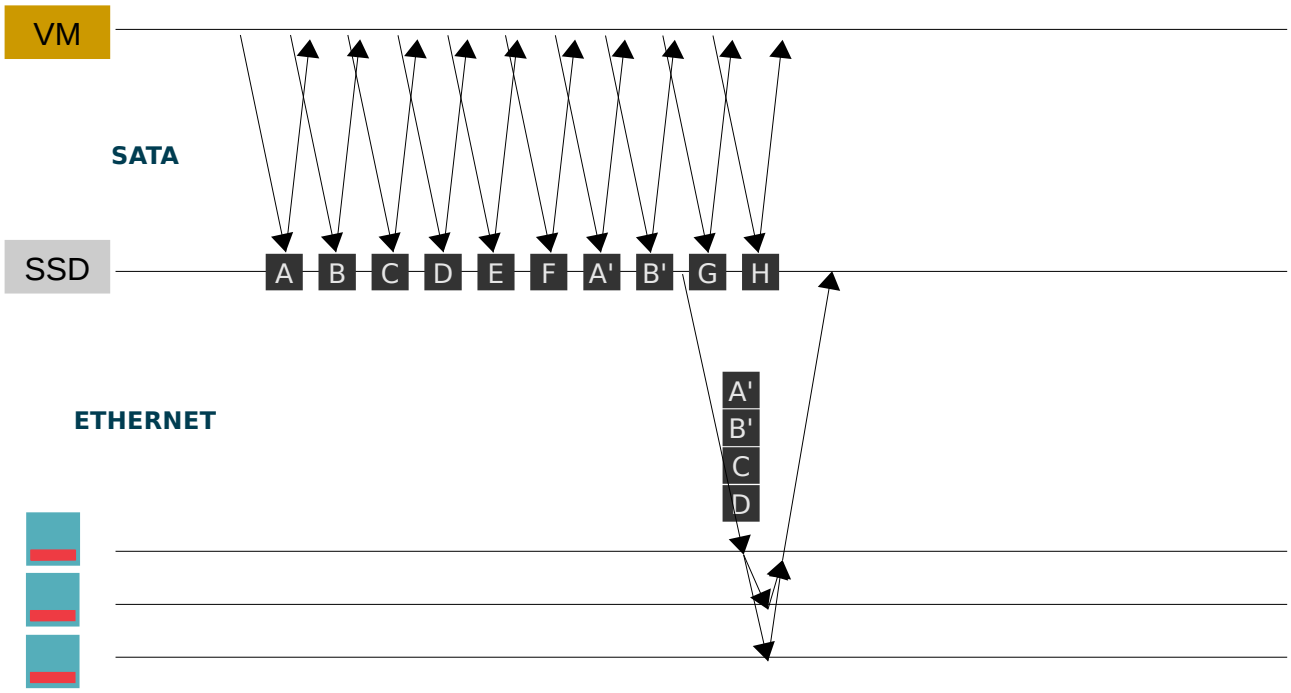
LOCAL SSD → LOW LATENCY



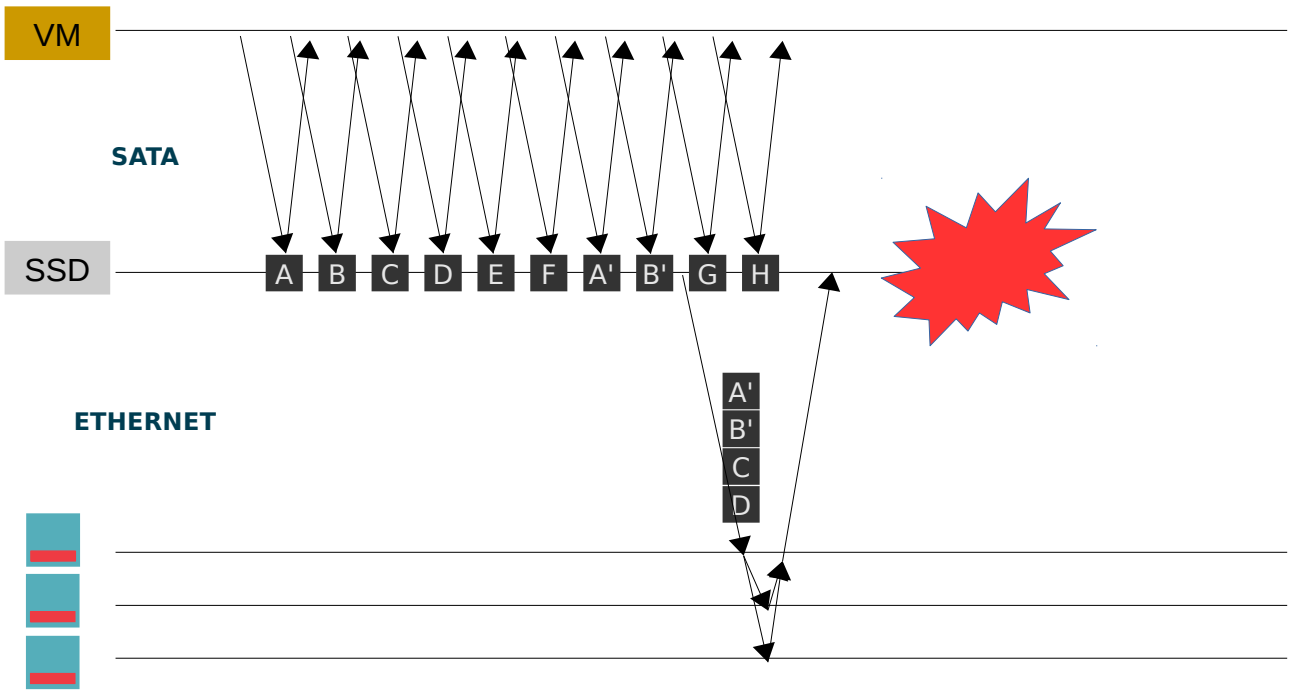
LOCAL SSD → FAILURE → SADNESS



WRITEBACK IS UNORDERED

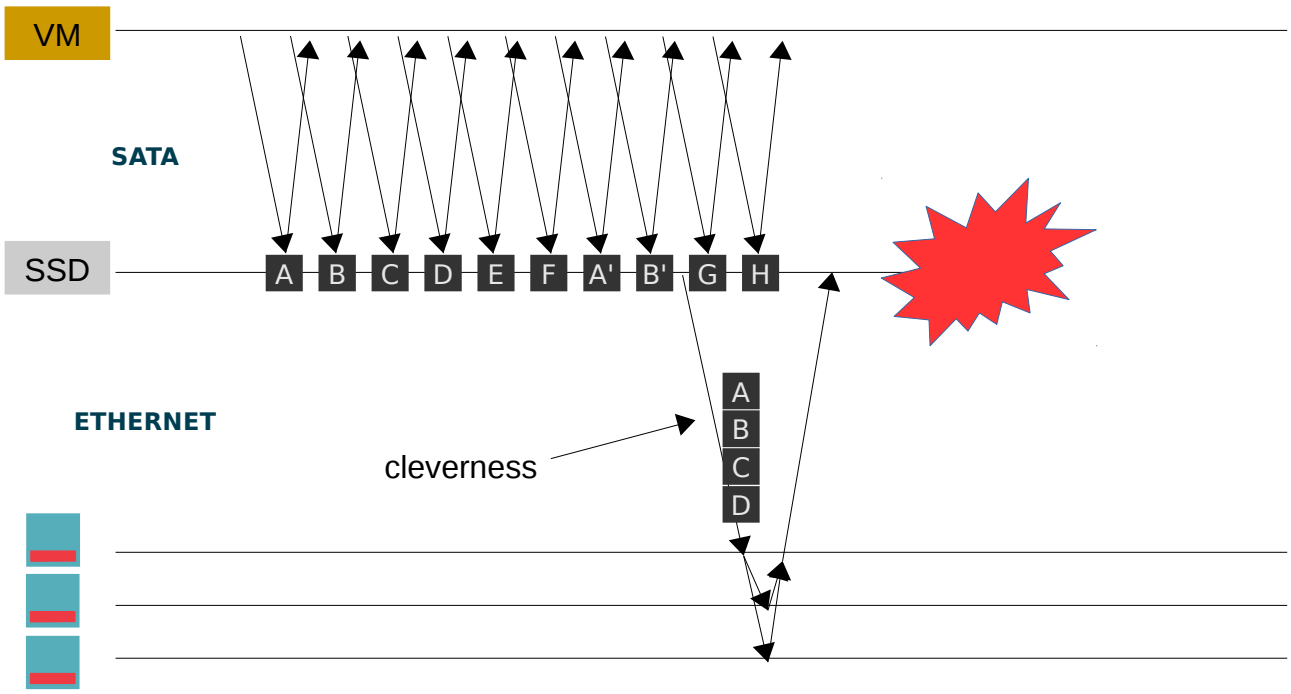


WRITEBACK IS UNORDERED



A' B' C D → never happened

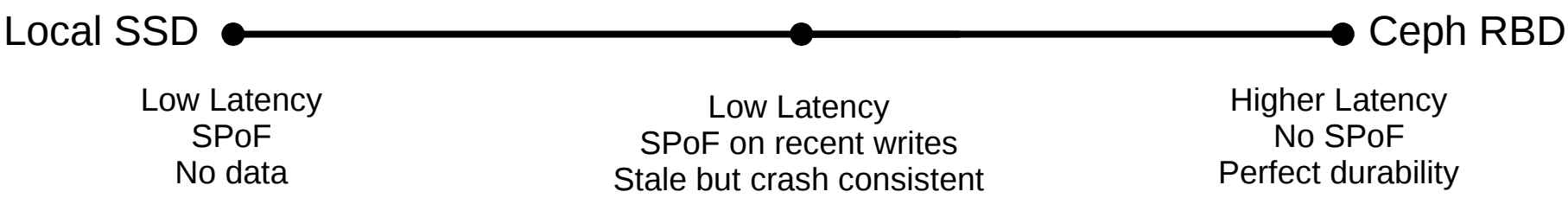
RBD ORDERED WRITEBACK CACHE



A B C D → CRASH CONSISTENT!

RBD ORDERED WRITEBACK CACHE

- Low latency writes to local SSD
- Persistent cache (across reboots etc)
- Fast reads from cache
- *Ordered* writeback to the cluster, with batching, etc.
- RBD image is always point-in-time consistent



IN THE FUTURE
MOST BYTES WILL BE STORED
IN **OBJECTS**

WORM

S3

Swift

Tiering

Erasure coding

Compression

Deduplication

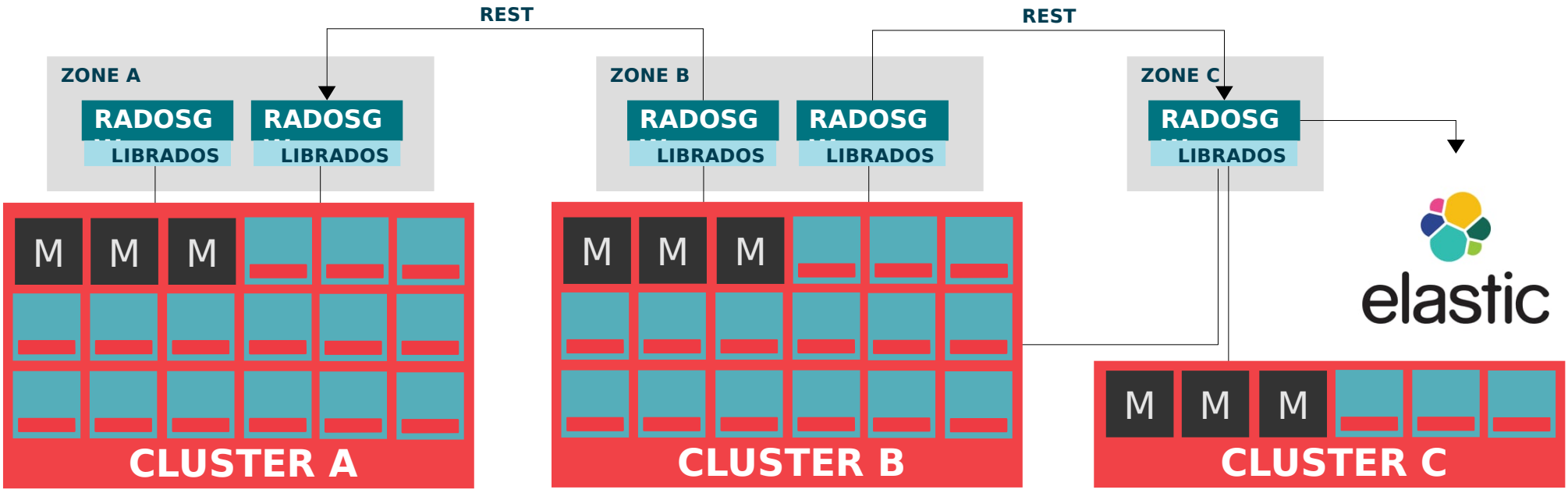
Multisite federation

Encryption

Multisite replication

RADOSGW

RADOSGW INDEXING



redhat.

GROWING DEVELOPMENT COMMUNITY

GROWING DEVELOPMENT COMMUNITY

- Red Hat
- Mirantis
- SUSE
- SanDisk
- XSKY
- ZTE
- LETV
- Quantum
- EasyStack
- H3C
- UnitedStack
- Digiware
- Mellanox
- Intel
- Walmart Labs
- DreamHost
- Tencent
- Deutsche Telekom
- Igalia
- Fujitsu
- DigitalOcean
- University of Toronto

GROWING DEVELOPMENT COMMUNITY

- **Red Hat**
- **Mirantis**
- **SUSE**
- SanDisk
- XSKY
- ZTE
- LETV
- Quantum
- **EasyStack**
- H3C
- **UnitedStack**
- Digiware
- Mellanox
- Intel
- Walmart Labs
- DreamHost
- Tencent
- Deutsche Telekom
- Igalia
- Fujitsu
- DigitalOcean
- University of Toronto

GROWING DEVELOPMENT COMMUNITY

- Red Hat
- Mirantis
- SUSE
- SanDisk
- XSKY
- ZTE
- LETV
- Quantum
- EasyStack
- H3C
- UnitedStack
- Digiware
- Mellanox
- Intel
- Walmart Labs
- **DreamHost**
- **Tencent**
- Deutsche Telekom
- Igalia
- Fujitsu
- **DigitalOcean**
- University of Toronto

GROWING DEVELOPMENT COMMUNITY

- Red Hat
- Mirantis
- SUSE
- SanDisk
- XSKY
- **ZTE**
- LETV
- Quantum
- EasyStack
- H3C
- UnitedStack
- Digiware
- Mellanox
- Intel
- Walmart Labs
- DreamHost
- Tencent
- **Deutsche Telekom**
- Igalia
- Fujitsu
- DigitalOcean
- University of Toronto

GROWING DEVELOPMENT COMMUNITY

- Red Hat
- Mirantis
- SUSE
- **SanDisk**
- XSKY
- ZTE
- **LETV**
- Quantum
- EasyStack
- **H3C**
- UnitedStack
- Digiware
- **Mellanox**
- **Intel**
- Walmart Labs
- DreamHost
- Tencent
- Deutsche Telekom
- Igalia
- **Fujitsu**
- DigitalOcean
- University of Toronto

GROWING DEVELOPMENT COMMUNITY

- **Red Hat**
- Mirantis
- SUSE
- **SanDisk**
- **XSKY**
- ZTE
- LETV
- **Quantum**
- EasyStack
- H3C
- UnitedStack
- Digiware
- Mellanox
- Intel
- Walmart Labs
- DreamHost
- Tencent
- Deutsche Telekom
- Igalia
- **Fujitsu**
- DigitalOcean
- University of Toronto

TAKEAWAYS

- Nobody should have to use proprietary storage systems out of necessity
- The **best** storage solution should be an **open** source solution
- Ceph is growing up, but we're not done yet
  - performance, scalability, features, ease of use
- We are highly motivated

HOW TO HELP

Operator

- File bugs
  - http://tracker.ceph.com/
- Document
  - http://github.com/ceph/ceph
- Blog
- Build a relationship with the core team

Developer

- Fix bugs
- Help design missing functionality
- Implement missing functionality
- Integrate
- Help make it easier to use
- Participate in monthly Ceph Developer Meetings
  - video chat, EMEA- and APAC-friendly times

THANK YOU!

Sage Weil
CEPH PRINCIPAL ARCHITECT


sage@redhat.com


@liewegas