
CERN

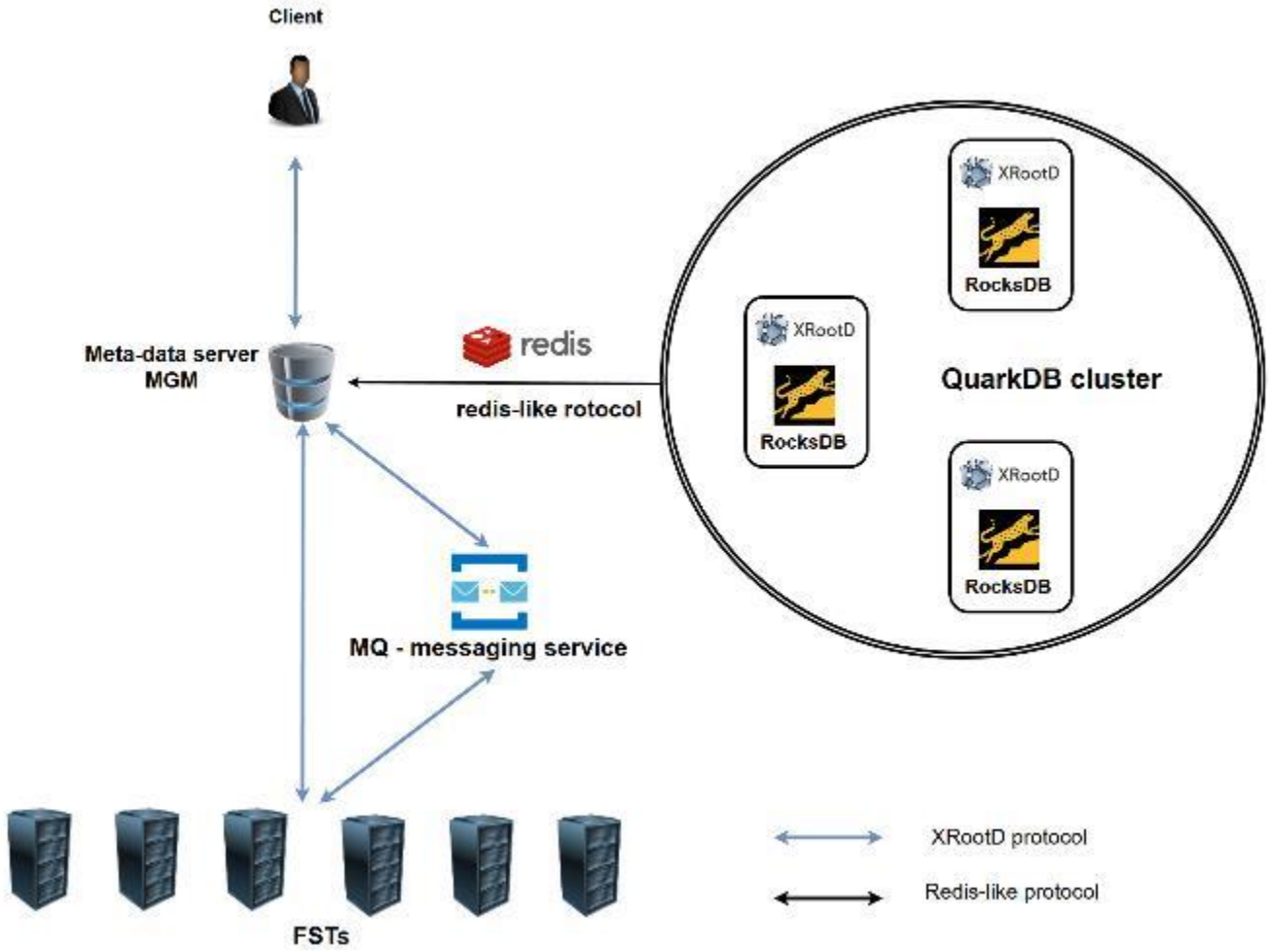
EOS status and strategic directions

Elvin Sindrilaru
on behalf of the **EOS team**

Outline

- EOS architecture overview
- New namespace and FUSEx
- Central draining
- Recycle-bin structure changes
- QuarkDB configuration and HA setup
- LRU and FSCK refactoring
- Packaging changes and Kubernetes testing
- Plans for the future

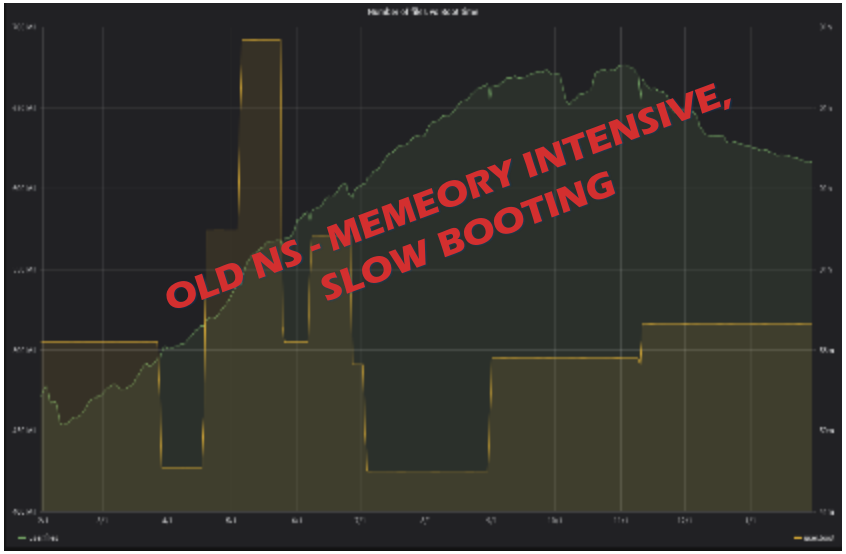
EOS architecture



QuarkDB namespace

- **QuarkDB in production**:
  - **EOSHOME** instance acting as backend for CERNBOX
  - **EOSBACKUP** holding > **1.5 B** files
  - **EOSPPS** > **3.5 B** files
  - All **LHC** experiment instances

- Designed and implemented **QuarkDB**, a highly available datastore for the namespace:
  - **Redis protocol**, supports a small subset of Redis commands
  - **RocksDB** as the underlying storage backend
  - High availability: **Raft consensus**

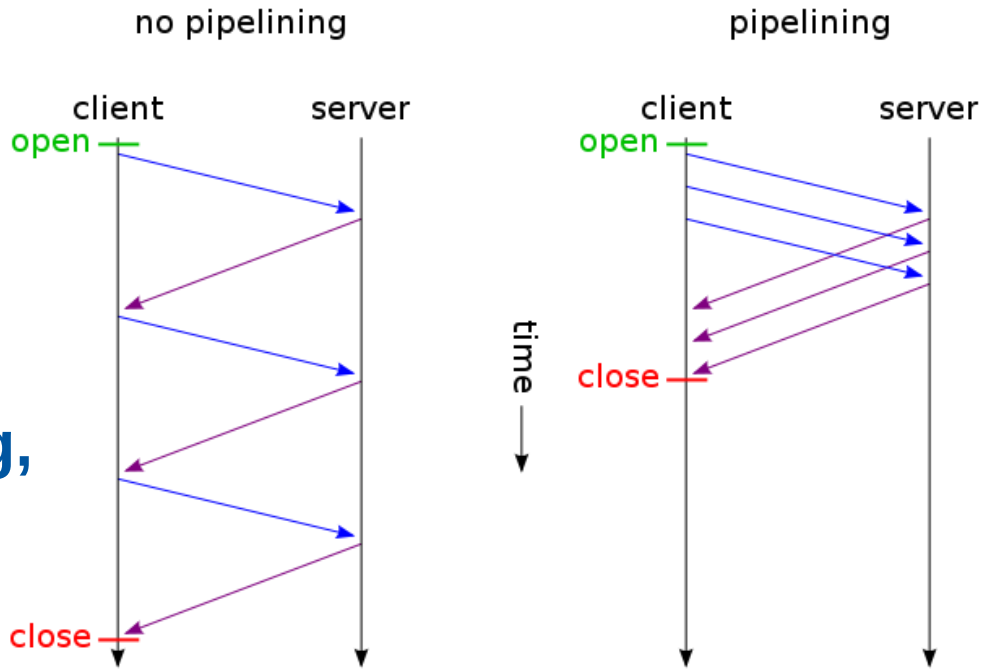


NEW NS – MEMORY FRIENDLY,
FAST BOOTING

```
[root@eoshome-ns-i01-01 (mgm:master mq:master) ~]$ eos ns
# ------------------------------------------------------------------------------------
# Namespace Statistics
# ------------------------------------------------------------------------------------
ALL      Files                            139603441 [booted] (0s)
ALL      Directories                      14767987
ALL      Total boot time                  4 s
# ------------------------------------------------------------------------------------
```

Latency optimization

- No previous notion of **asynchronous namespace operations** in the MGM
- Impossible to achieve reasonable performance (many **kHz**) without it
- Optimization through MGM **metadata caching, prefetching, pipelining**, plus an **asynchronous write queue**…



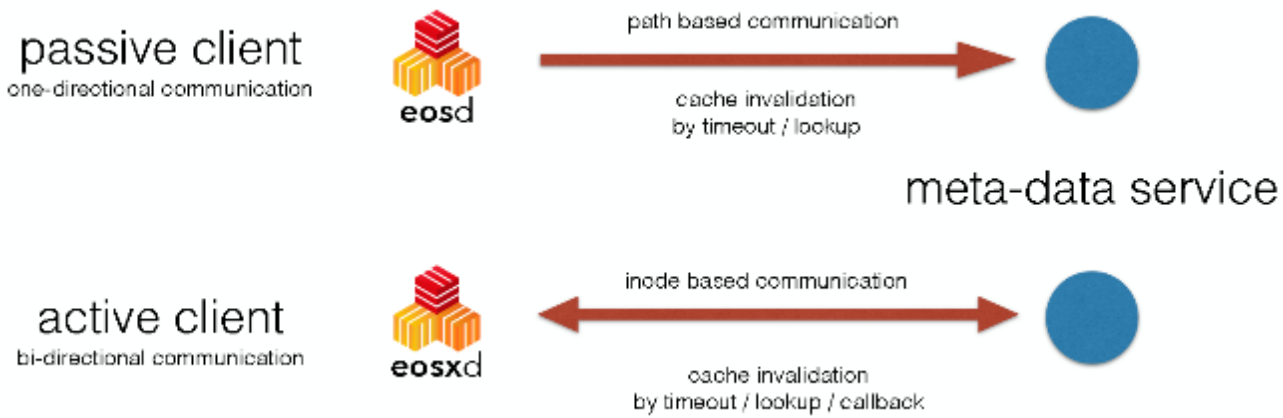
Practical matters for operators

- How to move to the new namespace?
  - Use the **conversion tool**:
    - http://eos-docs.web.cern.ch/eos-docs/quickstart/ns_quarkdb.html
- Will the old namespace stay around forever?
  - There's really no use-case which benefits anymore from the old NS, so **it will be deprecated**
  - **Support at least until 2020**, depending on how quickly the migration happens
- Using **SSDs** for QuarkDB data is ***necessary*** for good performance
- MGM and QuarkDB daemons can be **co-located or running on different nodes**

eos-ns-inspect tool

- Tool used for getting information about namespace entries
- Can be use for:
  - **dumping an entire hierarchy** – for example for **backup recovery** operations
  - getting **specific file information**
- For minimal operational impact can display info from:
  - **QDB snapshots**
  - **QDB slaves**
- Handy tool for:
  - displaying “offline” info about the state of the namespace
  - **post-incident inspection** for the namespace
  - scanning for different **error condition or inconsistencies**

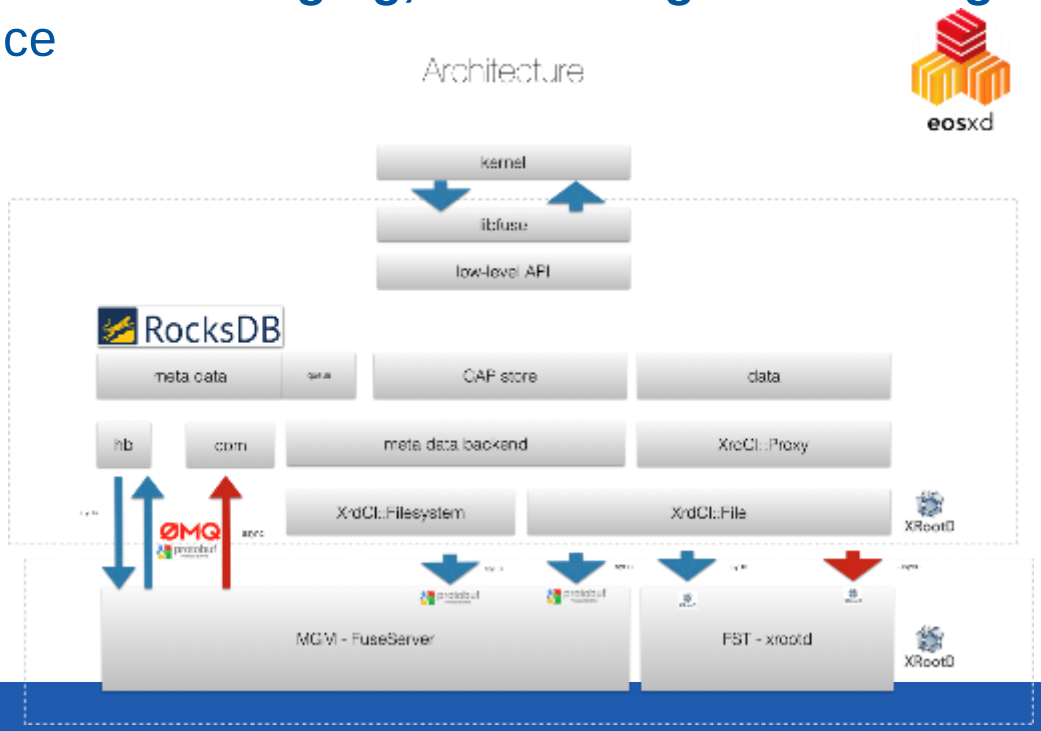
FUSEx (eosxd)



- Why **eosxd**
  - Better **POSIX**-ness
  - File locks, **byte-range locks**
  - **Hard links** within directories
  - **Rich ACL** client support
  - Local **caching**
  - **Bulk deletion**/protection
  - Strong security/mount-by-key

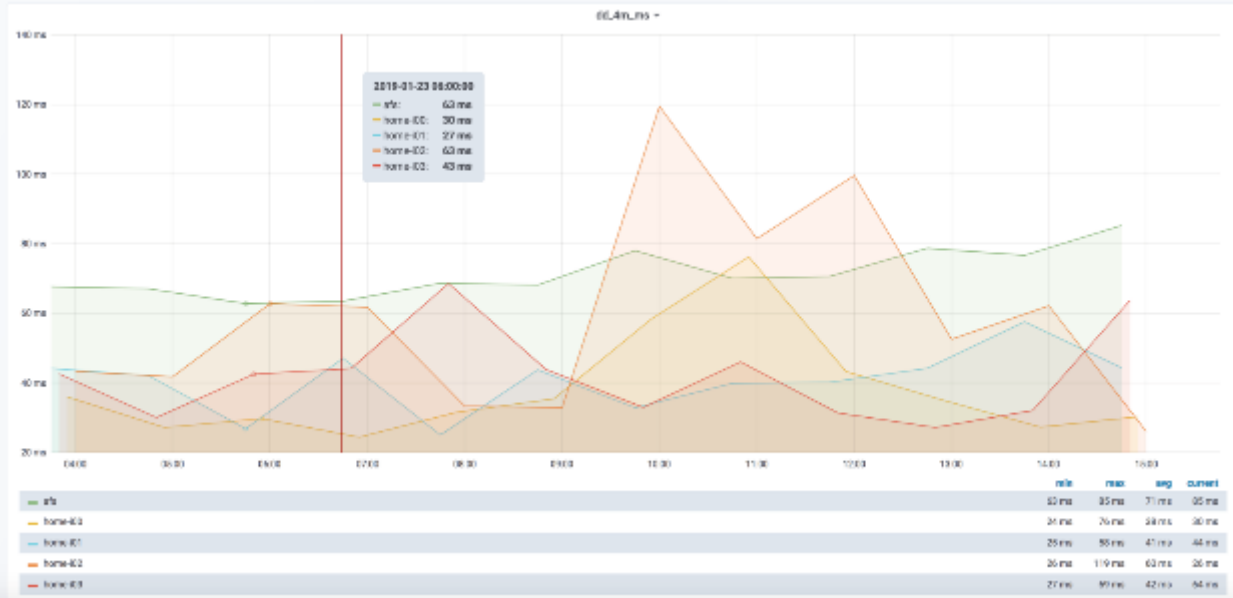
eosxd – improvements

- **Completely new architecture** with aggressive caching and strong consistency guarantees
- Functional improvements to **multi-client cache-invalidation protocol, mtime consistency, negative kernel cache management**
- Enhancement of **managing, monitoring and limiting client access** per instance



eosxd – many more improvements …

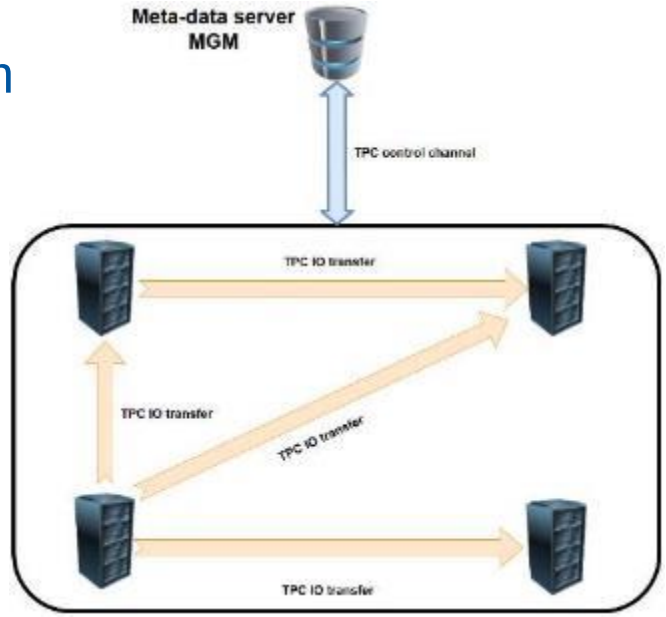
- Evolution of **client driven recovery** to hide common hardware unavailability
- Support for **containerized application** and simplification of strong authentication
- Performance *in general* on par with AFS depending on the workload – AFS implemented as a kernel module!



Central draining

- **Old distributed draining** model not scalable for the new namespace
  - Each FST was querying repeatedly the namespace for the list of files to be drained

- **Central draining** now steers transfer from the MGM using **XRootD TPC transfers**
  - Simplify the code on the FST side
  - Automatic retries and fallback to other replicas if first attempt failed
  - Handles any type of layout: plain, replica, RAIN
  - Dedicated/configurable pool of threads doing the draining
  - Queue for pending file-systems to be drained



Central draining configuration

- Dynamically configurable drain thread pool

eos ns max_drain_threads <num>

- Other configuration saved as space attributes
  - **drainer.node.fs** – max number of file-system in draining per node
  - **drainer.fs.ntx** – max number of parallel transfers per file system
  - **drainer.retries** – max number of retries if failed transfers

- **Monitor** performance:

```
[root@eosbackup-ns-00 (mgm:master mq:master) ~]$ eos ns stat | grep Central
 all DrainCentralFailed          678.79 K     7.00    58.29    11.54     8.38     -NA-     -NA-
 all DrainCentralStarted          18.53 M    47.75   118.61    50.85    35.08     -NA-     -NA-
 all DrainCentralSuccessful       17.83 M    40.75    16.66    21.32    25.21     -NA-     -NA-
```

Recycle bin structure changes

- Existing recycle path convention:

/…/proc/recycle/gid/uid/dir1#:#dir2#:#file1.dat.hex_fid

- **Drawbacks:**
  - Flattens the entire recycle history for a user
  - Leads to extremely large directories (100k – 1M)
  - Considerable scalability issues when using the QuarkDB namespace
- **New recycle path convention**

/…/proc/recycle/uid:<val>/<year>/<month>/<day>/<hash>/path.hex_fid

EOS configuration in QuarkDB

- Necessary step in providing high-availability setup
  - Move file-based config (default.eoscf) to **QDB**

- MGM setup requirements (**xrd.cf.mgm**):
  - **mgmofs.cfgtype quarkdb**
  - **mgmofs.qdbcluster <qdb1> <qdb2> …**
  - **mgmofs.qdbpassword_file <some/file>**
- **Configuration export** done using:

eos config export <path_to_config_file>

- Inspect the configuration directly from QuarkDB

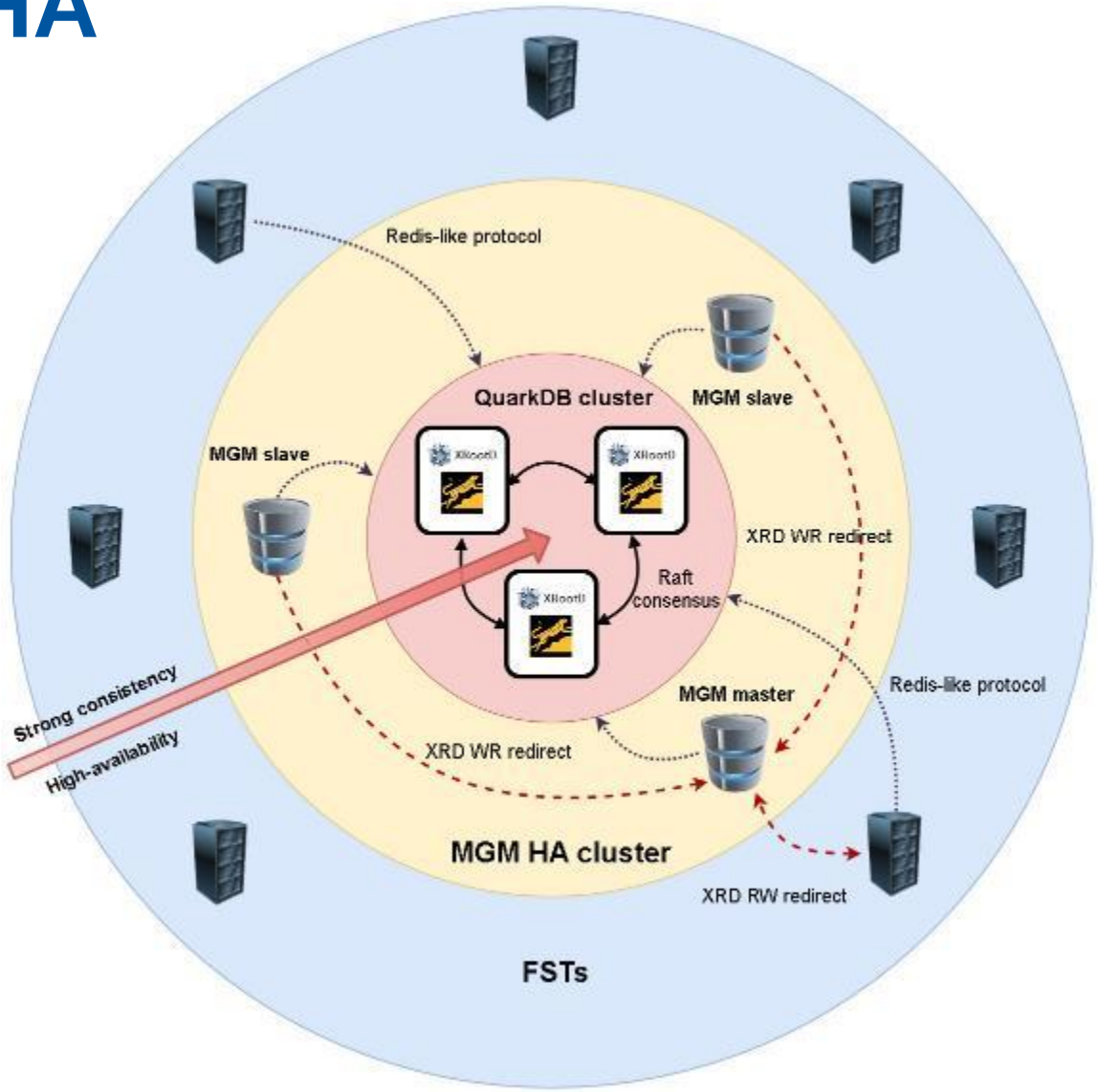
```
[root@eospps-ssd-ns1 ~]# redis-cli -p 7777 keys "eos-config:*"
1) "eos-config:backup1"
2) "eos-config:default"
[root@eospps-ssd-ns1 ~]# redis-cli -p 7777 hgetall "eos-config:default" | head -n 2
fs:/eos/lxfsre03a02.cern.ch:1095/fst/data01
```

QuarkDB leases

- Building block for providing **HA** for the MGMs
- Stores information concerning:
  - **Current owner** of the lease
  - **Validity** of the lease
- Operations on leases:
  - **lease_acquire**
  - **lease_release**
  - **lease_get** -> display information about the lease
- Master-slave MGMs synchronize using the lease key "**master_lease**"

```
[root@eospps-ssd-ns1 ~]# redis-cli -p 7777 lease-get "master_lease"
1) HOLDER: eospps-fe1.cern.ch:1094
2) REMAINING: 9023 ms
```

EOS HA


Redis-like protocol
QuarkDB cluster
MGM slave
MGM slave
XRD WR redirect
Raft consensus
Strong consistency
High-availability
MGM master
Redis-like protocol
XRD WR redirect
MGM HA cluster
XRD RW redirect
FSTs

EOS master-slave HA

- Rely on the **QuarkDB** lease to decide who is the master
  - Lease is **valid for 10 seconds (configurable)**
  - Master **renews** the lease every **5 seconds**
- During a slave->master transition **reload the configuration from QuarkDB**
- Automatically **enforce/disable stall rules**
- Force a master to abandon the lease

```
eos ns master other
```

- Master-slave info displayed in the “ns” command
  - **EOS-MGM-1: ALL Replication is_master=true master_id=eos-mgm-1.cern.ch:1094**
  - **EOS-MGM2-: ALL Replication is_master=false master_id=eos-mgm-1.cern.ch:1094**

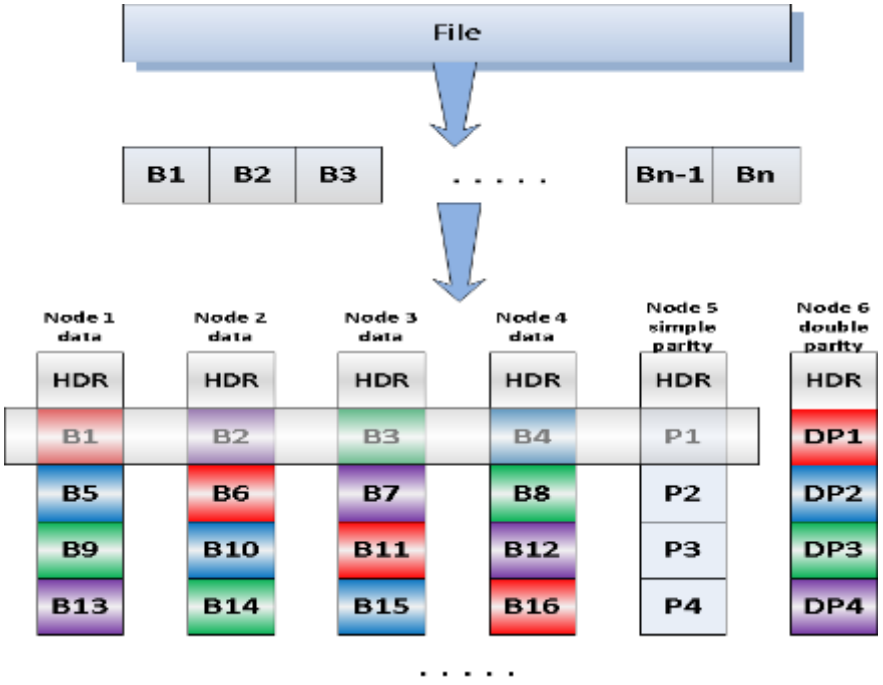
LRU and FSCK refactoring

- **LRU** requires scanning through the entire directory hierarchy
  - NS intensive operation which can **trash the directory cache** at the MGM
  - Does not require strong consistency → **avoid taking the global namespace lock**
  - Rewritten to take advantage for the QDB interface and not impact normal user activity
  - Fully functional since **EOS 4.4.35** version
- **FSCK** is also an NS intensive operation
  - State can change considerably between runs
    - EOS has 30-300 servers i.e. 1000 → 15000 disk ok, partially failing or broken
    - Some hardware not working is the standard case
  - Needs to be **redesigned to take advantage of the QDB backend**
  - No need to trash the MGM cache
  - Plan to integrate it with a **QoS workflow** which will address transient failures
- **Final goal** → Improve user experience and availability
  - **MGM** behaves rather binary – available or not
  - **FST** (storage server) – can exhibit **transient failures**
    - These can crash the applications and transient behavior can be *frustrating* for the users

EOS RAIN support

- EOS supports by default different types of RAIN layouts:
  - **RAID Double Parity** (4 data +2 parity stripes) – uses XOR
  - **Reed Solmon** (4+2), Archive (6+3) or other combinations

- File layout type is set as an **extended attribute** of the directory containing the files
- **Other attributes:** checksum, block checksum, number of replicas/stripes
- Preferred block checksum type: **CRC32C** uses SSE if the HW supports it



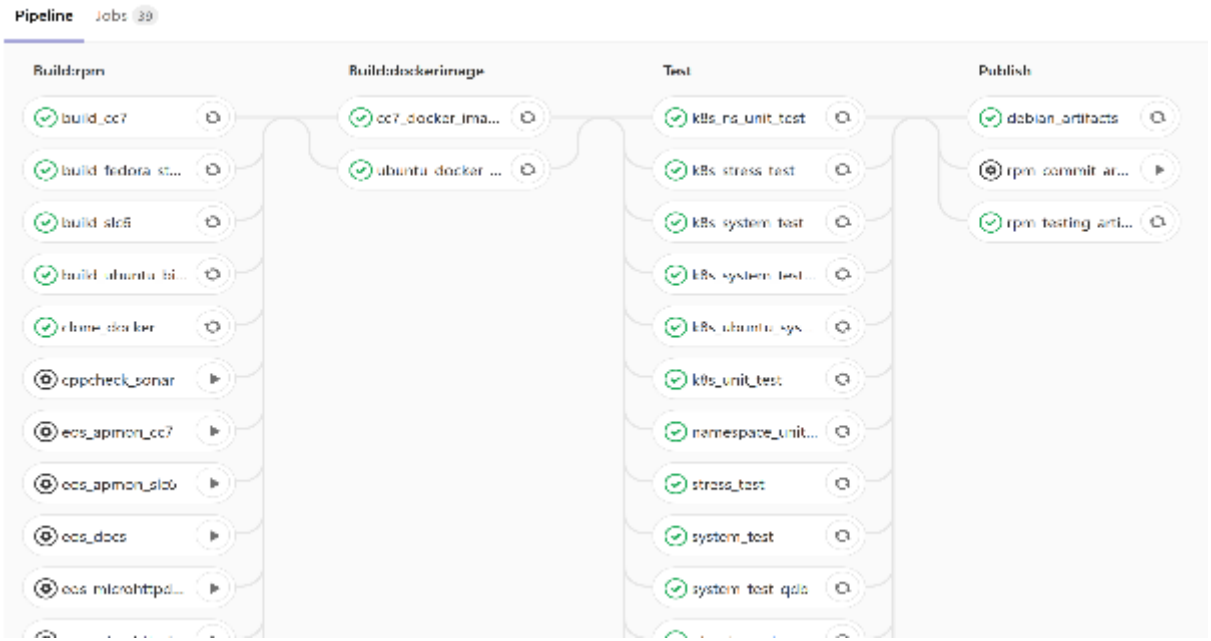
EOSALICEDAQ RAIN conversion

- Converted 1.2M files, ~4.8 PB physical size (2-replica)
- RS(10,2), **freed 2 PB** , took 84 hours

Packaging and Kubernetes testing

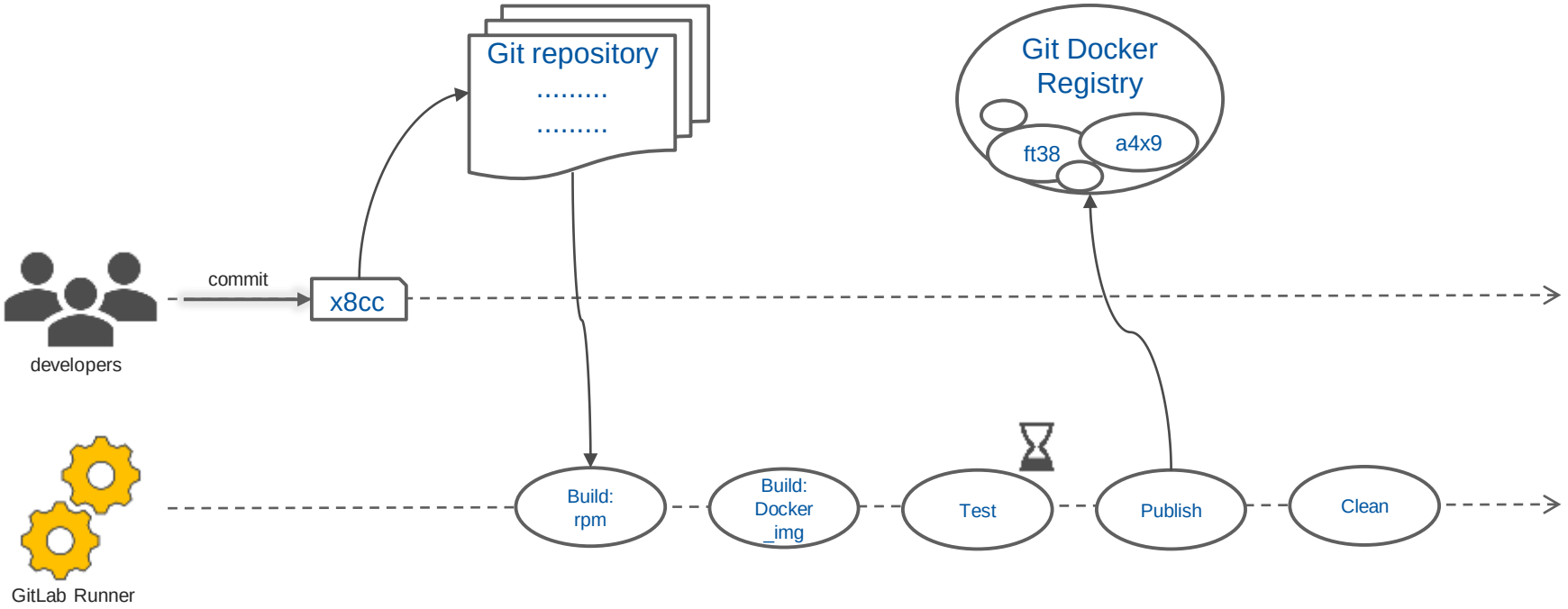
- Starting with **4.4.44** EOS brings its own dependencies for:
  - **XRootD**
  - **Libprotobuf3** – considerable improvement for eosxd performance
- These dependencies are stored in **/opt/eos/**
- **EOS server** depends on **XRootD private headers** – need to ensure compatibility
- EOS executables and libraries have **RPATH** pointing to the /opt/eos/ location
- **Testing infrastructure**
  - Docker based
  - Kubernetes extension for scalability

EOS testing in GitLab CI

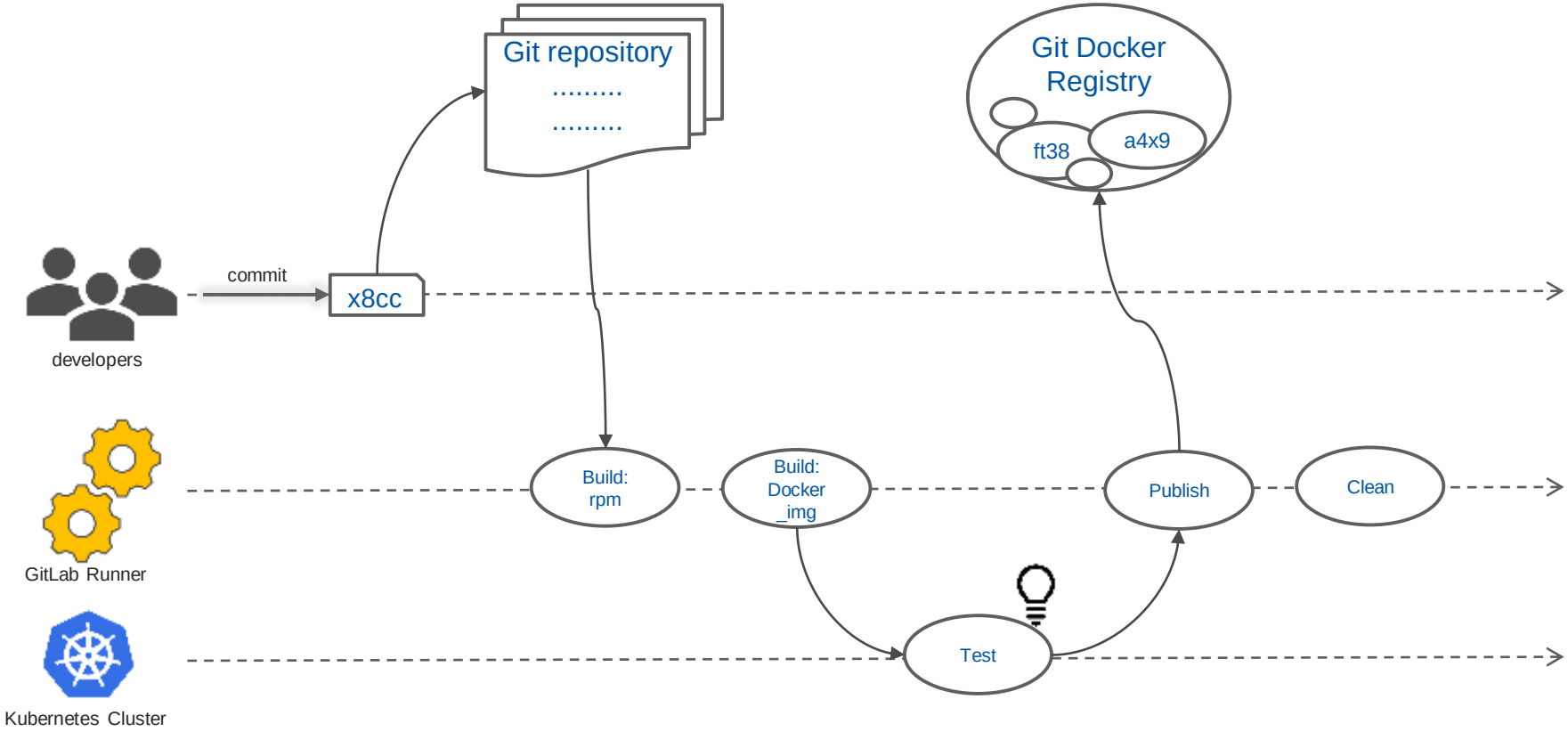


- Build and distribute Docker images for different OSes
- Create **your own fully functional EOS instance** on your laptop in a matter of seconds:
  - https://gitlab.cern.ch/eos/eos-docker

EOS testing workflow



EOS testing at scale with Kubernetes



- Repository with helper scripts and instructions
  - https://gitlab.cern.ch/faluchet/eos-on-k8s

Plans for the future

- Stop support for the **beryl_aquamarine** branch: **2019**
- Focus on **stability** and better **fault-tolerance**
- Drop the MQ daemon and move **messaging pub-sub** to QuarkDB
- Improve **availability** and **self-healing mechanisms**
  - Redesign the FSCK functionality

- **No (other) big changes** from the current model

www.cern.ch