CERN

EOS, DPM and FTS developments and plans

Andrea Manzi - on behalf of the
IT Storage Group, AD section

HEPIX Fall 2016 Workshop – LBNL

Outline

- CERN IT Storage group, AD section
- EOS
  - Namespace on Redis
- DPM
  - DOME
- FTS
  - New optimizer
  - Object Stores Integration

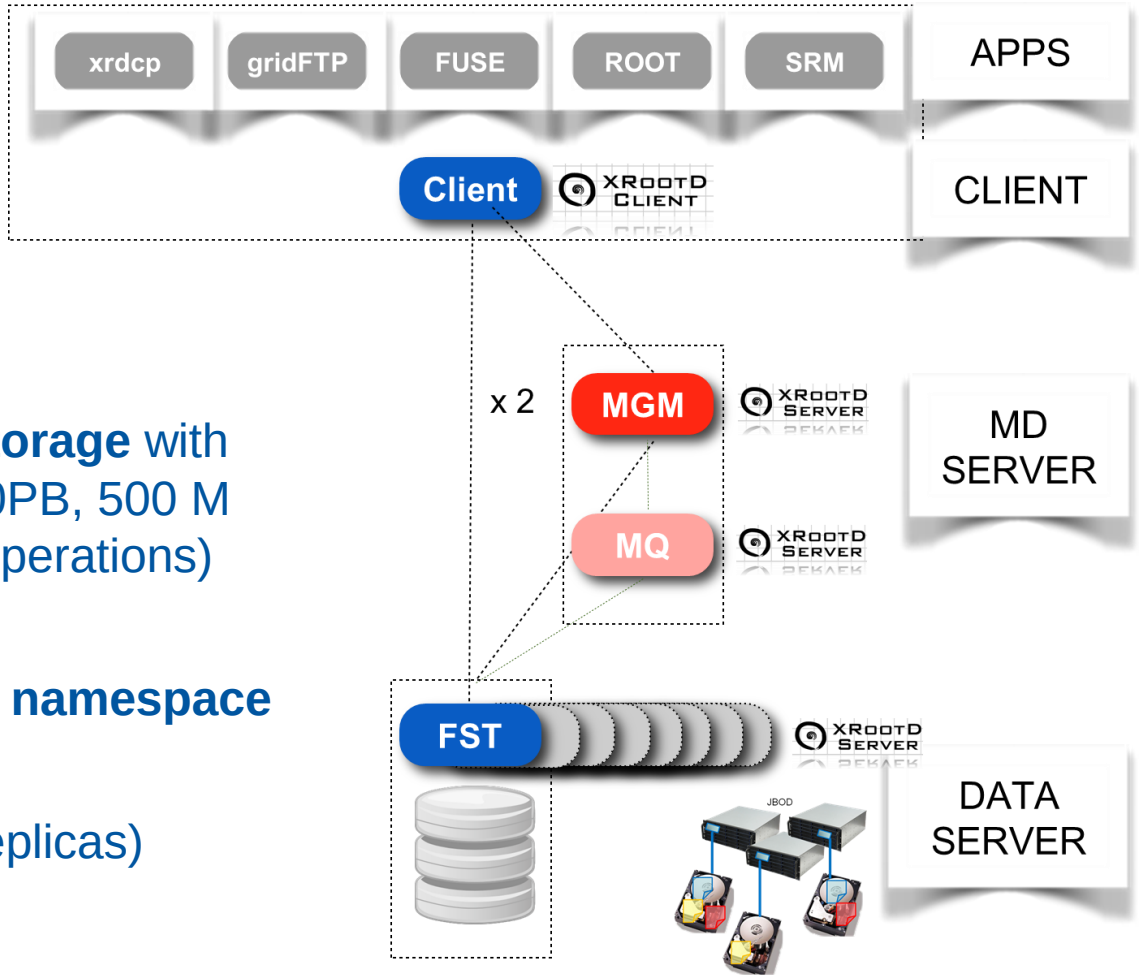
CERN IT-ST group, AD Section

- 16 members
- Main activities
  - Development
    - **EOS, DPM, FTS**, Data management clients (Davix**,** gfal2, Xrootd client)
  - Operations
    - FTS
  - Analytics WG
  - Effort in WLCG ops

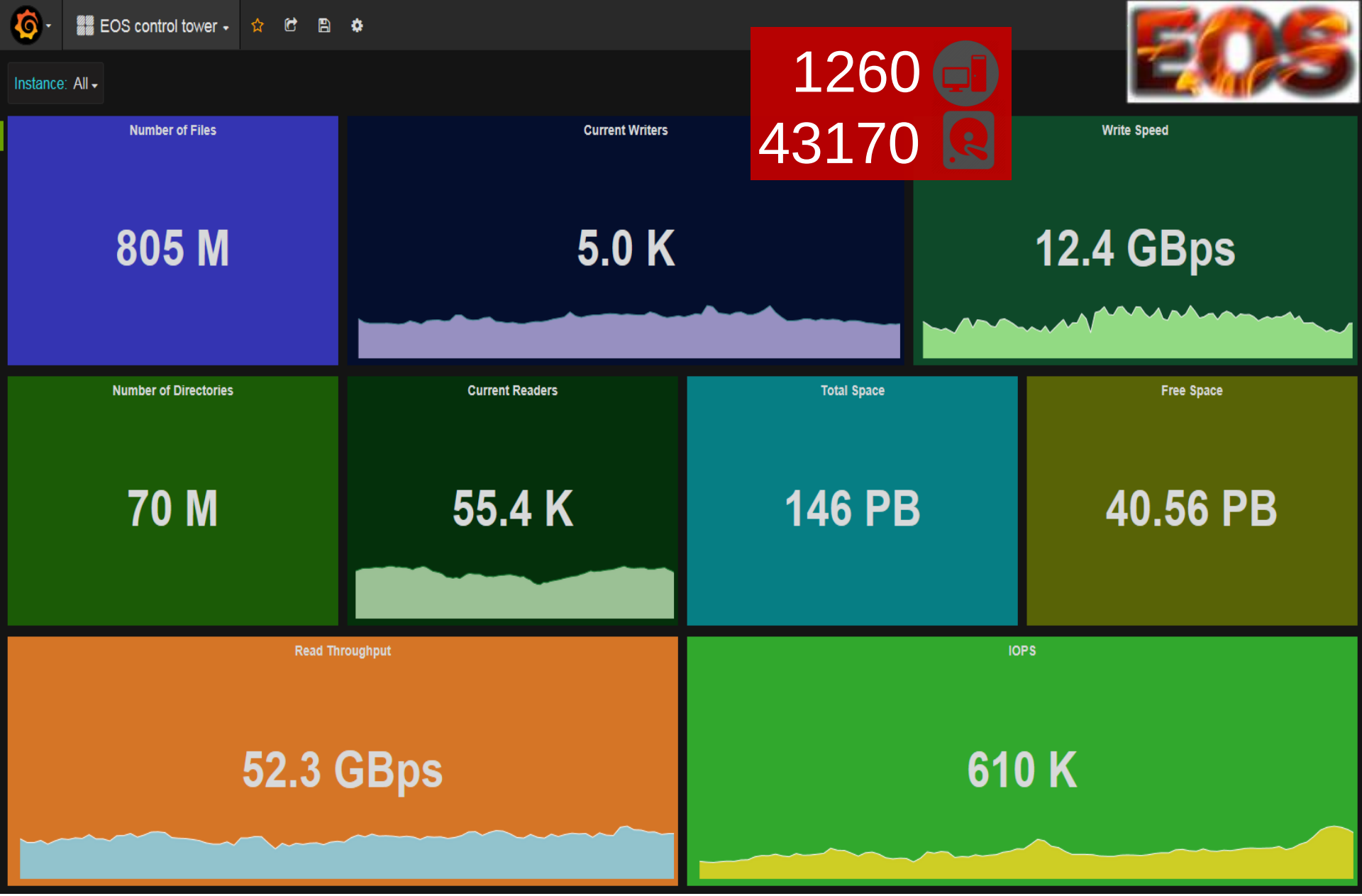
Outline

- CERN IT Storage group, AD section
- **EOS**
  - **Namespace on Redis**
- DPM
  - DOME
- FTS
  - New optimizer
  - Object Stores Integration

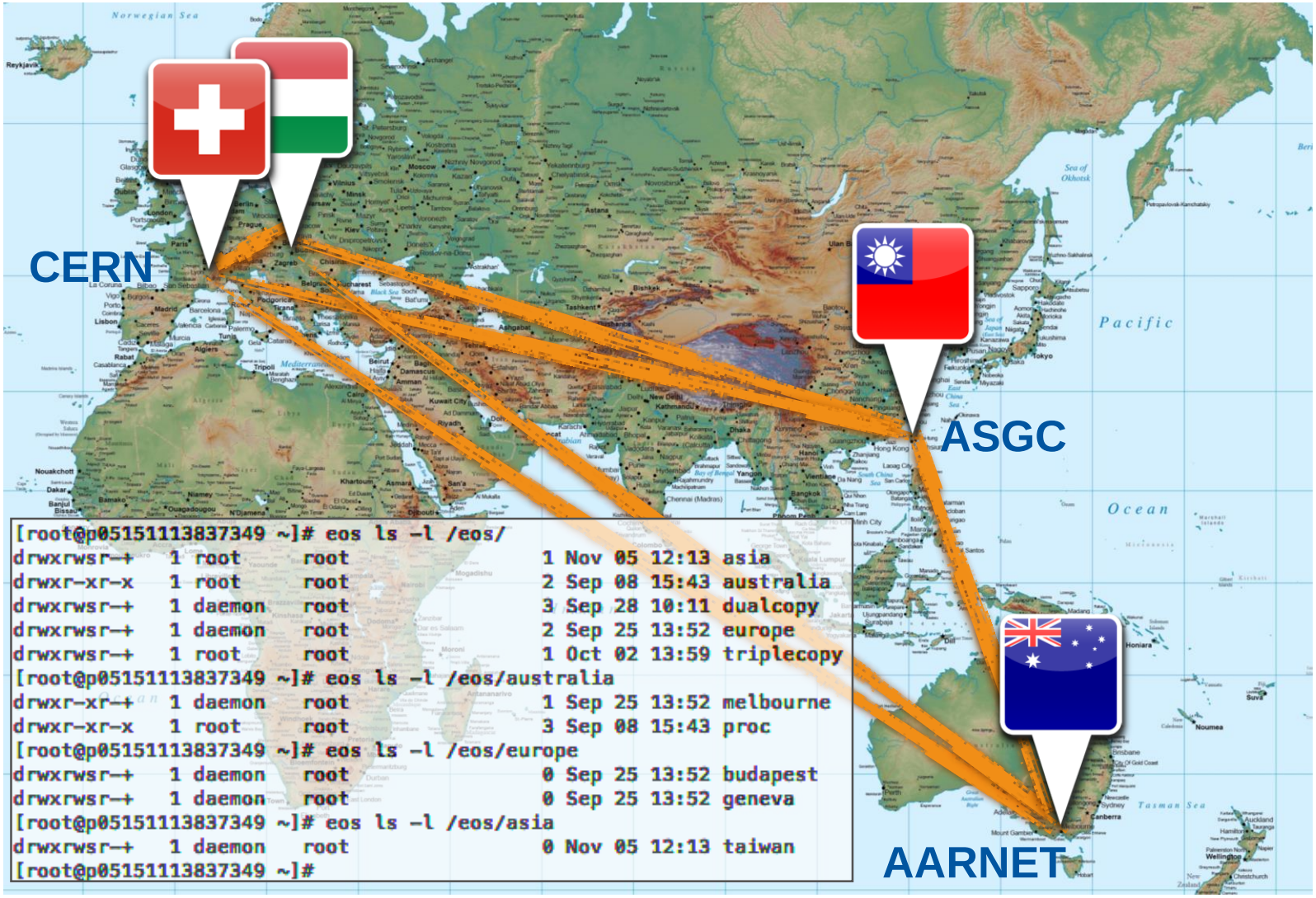
EOS architecture



- **Disk only** file storage
- Designed for **Massive storage** with high performances (> 50PB, 500 M files and Khz metadata operations)
- **In memory hierarchical namespace**
- **File layouts** (default 2 replicas)
- **Low latency access**


EOS control tower
Instance: All
1260
43170
Number of Files
805 M
Current Writers
5.0 K
Write Speed
12.4 GBps
Number of Directories
70 M
Current Readers
55.4 K
Total Space
146 PB
Free Space
40.56 PB
Read Throughput
52.3 GBps
IOPS
610 K

EOS World-Wide Deployment



EOS releases and branches

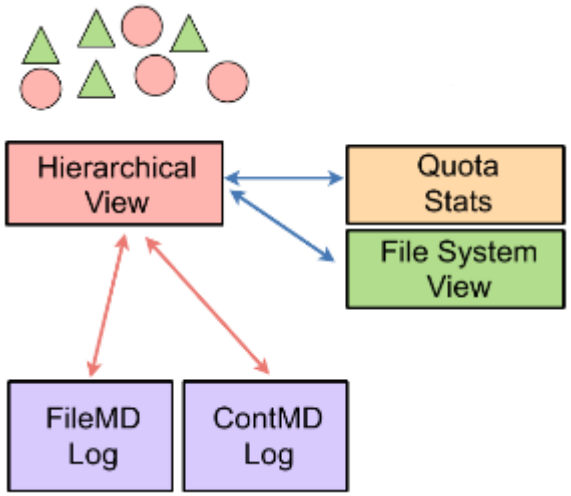
- **Production** version
  - Branch: **beryl_aquamarine**
  - Release number: >= 0.3.210

- **Development** version (master)
  - Branch: **citrine**
  - Release number: >= 4.1.4
  - Requires **XRootD** 4.4.0

- **Feature branches** get merged into master e.g. kinetic, geo-scheduling, namespace devel. etc.

What is the EOS namespace?

- C++ library used by the EOS MGM node single-threaded
- Provides API for dealing with hierarchical collections of files



- **Filesystem elements**
  - Containers & files
- **Views**
  - Aggregate info about filesystem elem.
  - E.g QuotaView, FileSystemView etc.
- **Persistence objects**
  - Objects responsible for reading and storing filesystem elements
  - Implemented as binary change-logs

Namespace architectures pros/cons

- **Pros:**
  - Using hashes all in memory → **extremely fast**
  - Every change is logged → **low risk of data loss**
  - Views rebuilt at each boot → **high consistency**

- **Cons:**
  - For big instances it requires **a lot** of RAM
  - Booting the namespace from the change-log takes long

EOS Namespace Interface

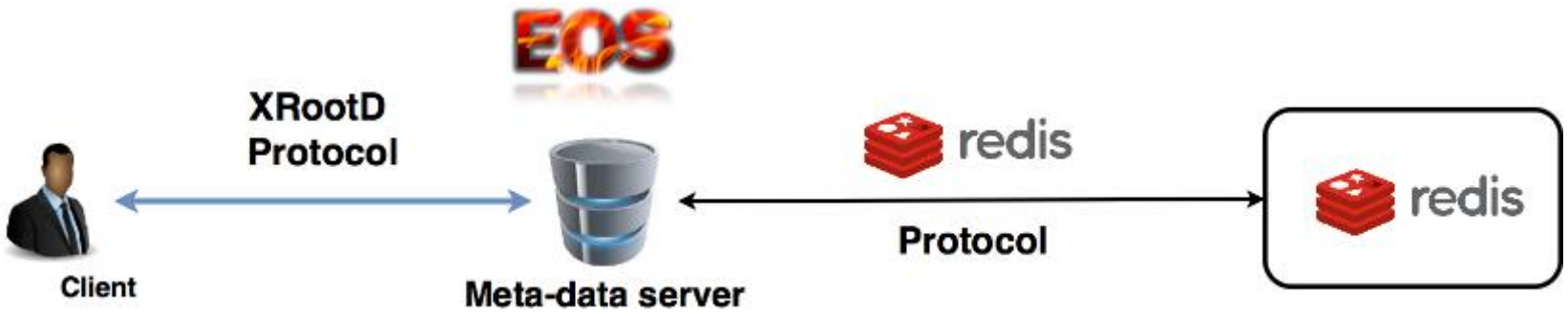
- Prepare the setting for different namespace implementations
- Abstract a **Namespace Interface** to avoid modifying other parts of the code

- **EOS citrine 4.***
  - **Plugin manager** – able not only to dynamically load but also stack plugins if necessary
  - **libEosNsInMemory.so** – the original in-memory namespace implementation
  - **libEosNsOnFilesystem.so** – not existing based on a Linux filesystem

Why Redis?

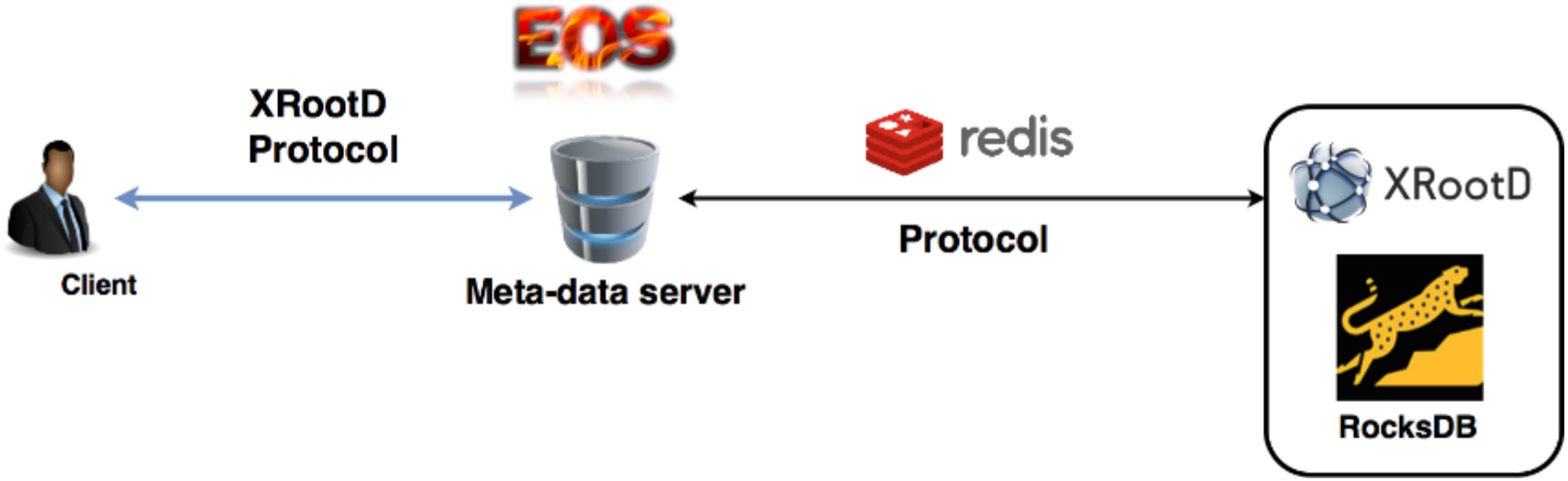
- **Redis** – in-memory **data structure store**
- Separate data from the application logic and user interface
- Supports various data structures: strings, hashes, lists, sets, sorted sets etc.

- Namespace implementation: **libEosOnRedis.so**



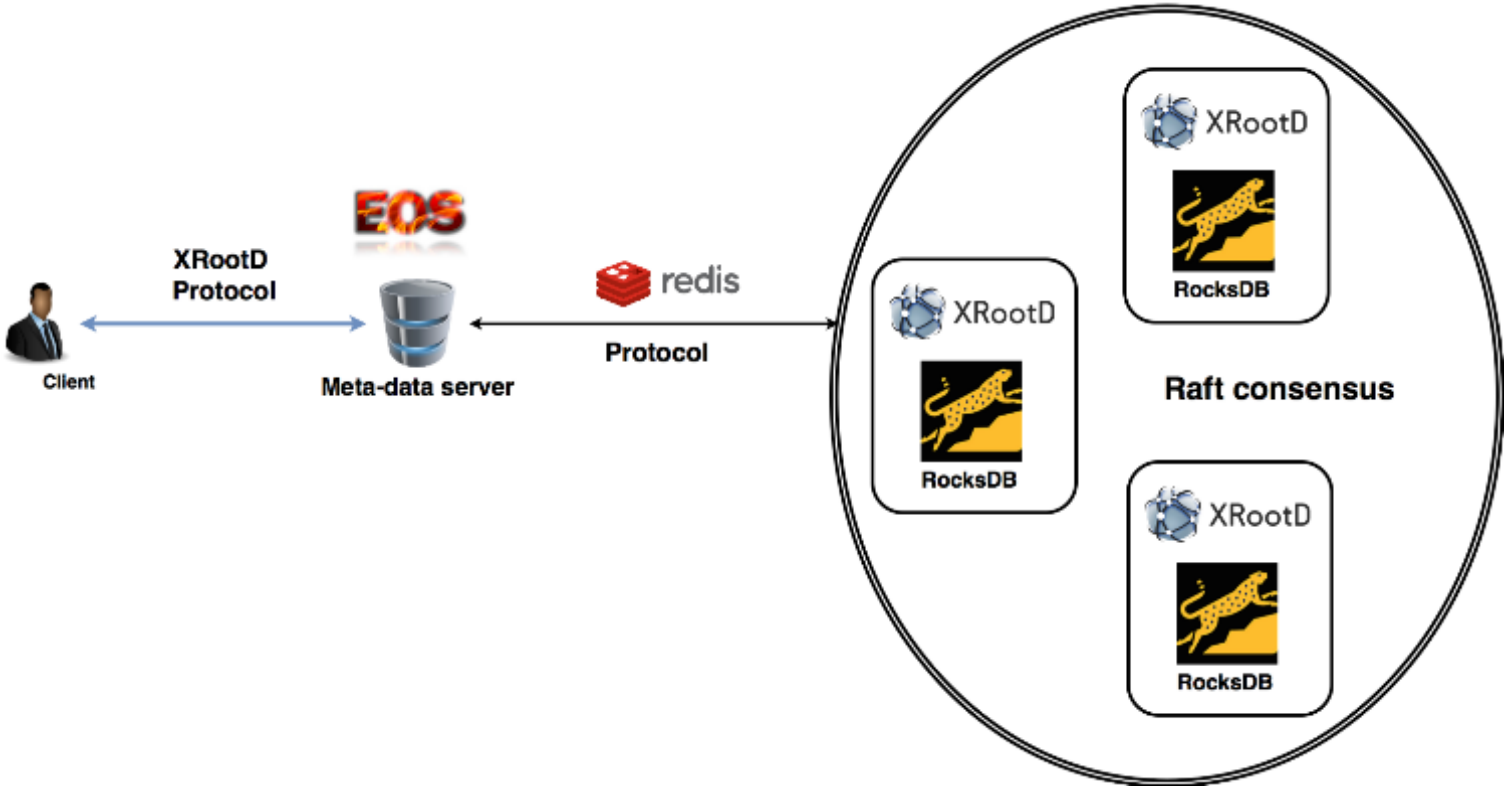
XRootD and Redis

- Replace Redis backend with XRootD
- Implemented as an XRootD **protocol plugin** – to be contributed upstream
- XRootD can use **RocksDB** as persistent key-value store



Namespace HA

- Ensure high-availability using the **Raft consensus algorithm**



Outline

- CERN IT Storage group , AD section
- EOS
  - Namespace on Redis
- **DPM**
  - **DOME**
- FTS
  - New optimizer
  - Object stores Integration

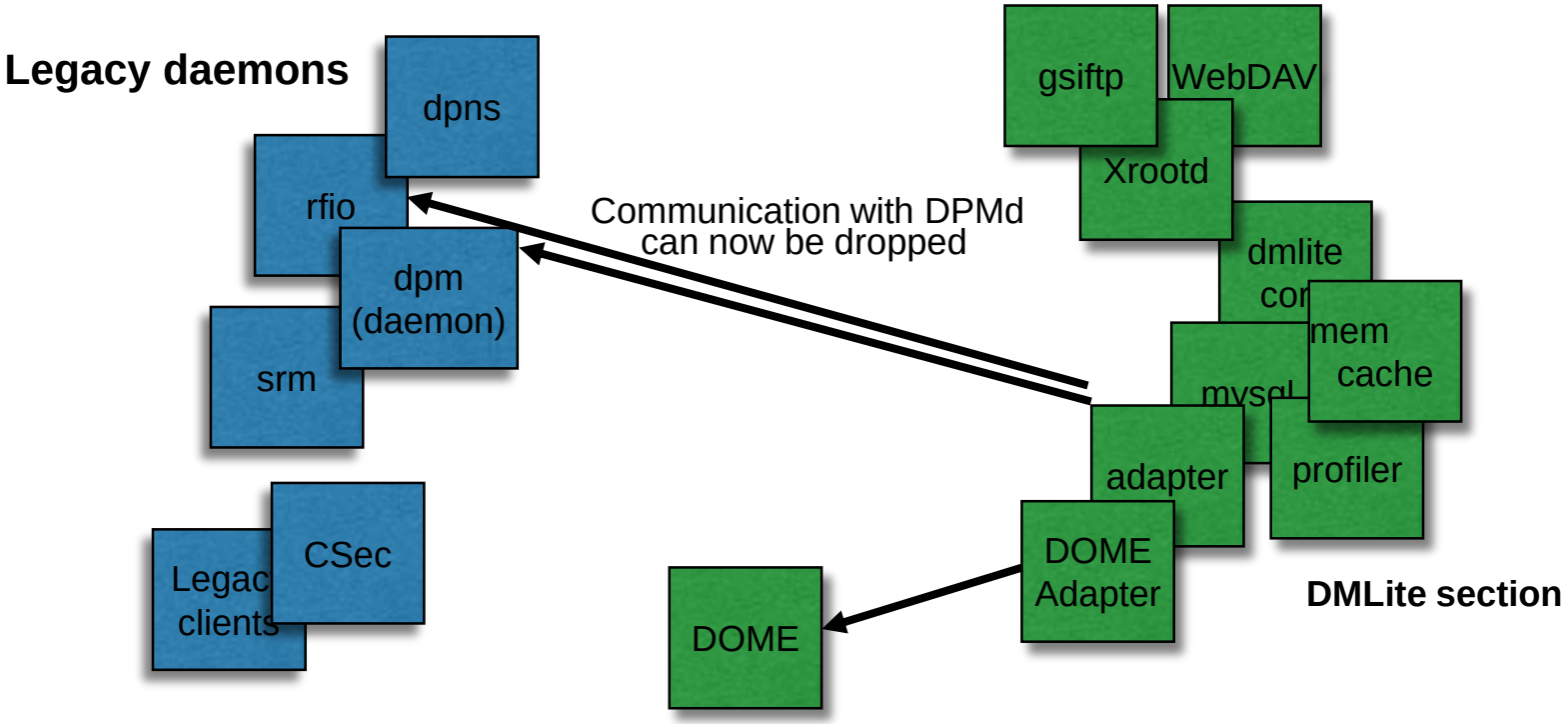
Disk Pool Manager

- DPM is a system for managing disk storage at small and medium sites
- Over 70PB stored in the system at around 150 sites
- DPM has just finished a little revolution.
  - We now have our definitive platform for the future
    - **SRM-less operations**
    - **Caching**

The 3rd Generation DPM

- **1st generation** – derived from **Castor**
- **2nd generation** – introduced **dmlite**
  - Internal framework abstracting many functions, enabling multiple new frontends
- **3rd generation** – introduces **Dome**
  - DPM 1.9 “Legacy flavour”
    - Legacy services still running
  - DPM 1.9 “Dome flavour”
    - Legacy services retired
      - No more dpmd, dpns, rfio, srm, csec

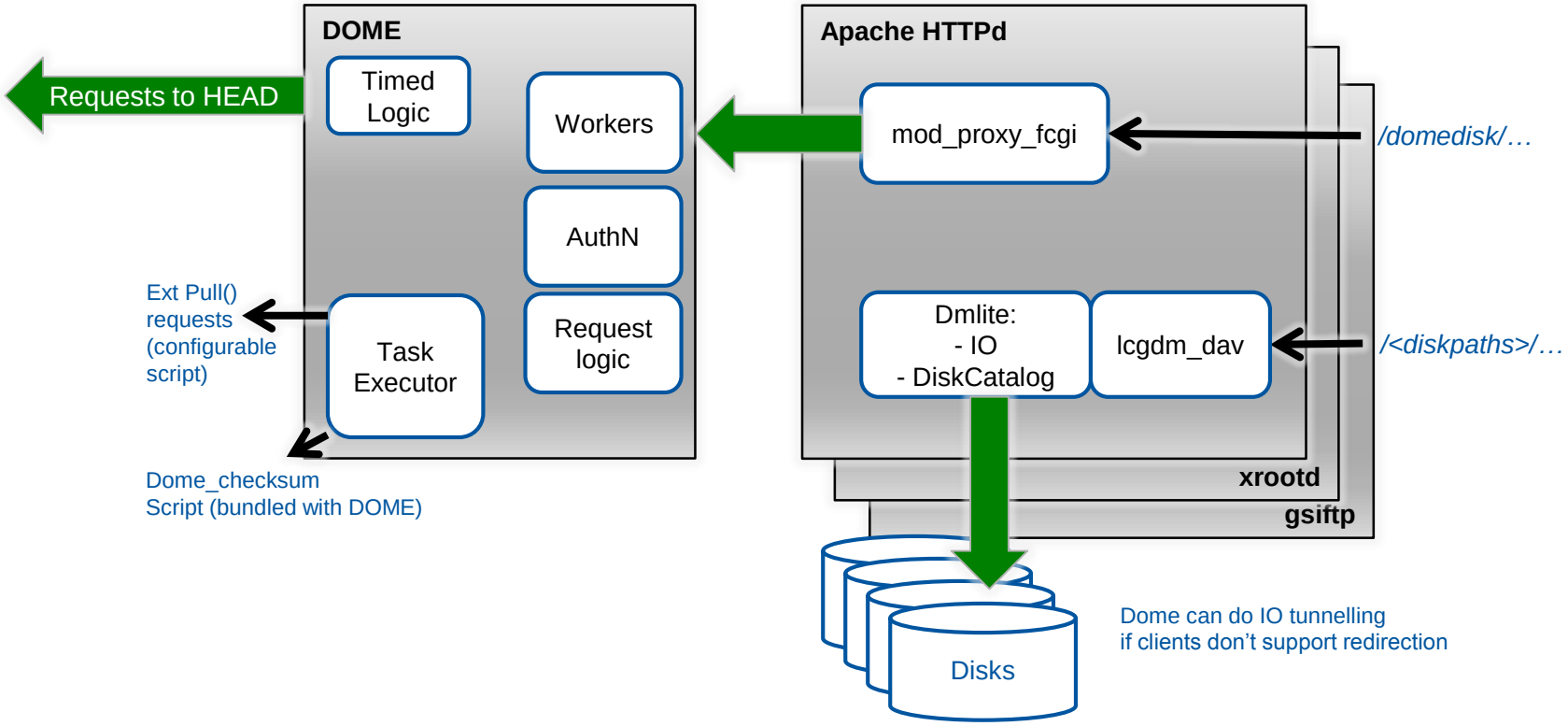
DPM Headnode architecture



DPM 1.9 “Dome flavour”

- Activate Dome, what do you get?
  - **SRM free operation**
    - **Quotas**
      - “space token” concept generalised and mapped into namespace
    - **Space reporting** – used/free via HTTP/DAV
      - Reporting on “space tokens” and subdirectories
    - **GridFTP redirection** enables scalability
  - **Caching hooks**
  - **Simplified system**
    - All internal communication over HTTP
      - Control and data tunneling
  - **Improved dmlite-shell**

Disk server (simplified)

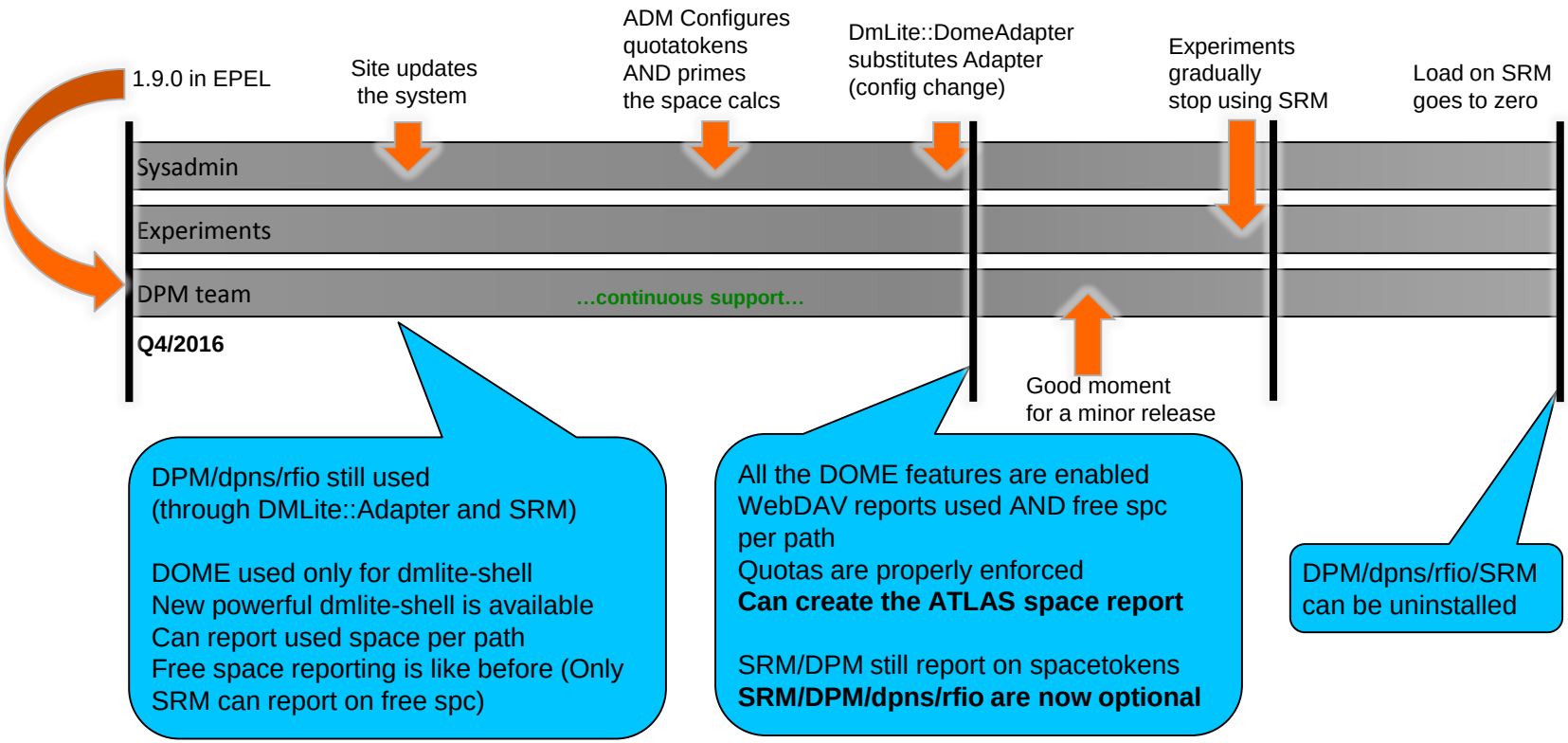


Used/free space: WebDAV

```
$ davix-http -P grid -X PROPFIND --header 'Depth: 0' --header 'Content-Type: text/xml;
charset=UTF-8' "https://domehead-trunk.cern.ch/dpm/cern.ch/home/dteam" --data '<?xml
version="1.0" ?><D:propfind xmlns:D="DAV:"><D:prop><D:quota-used-bytes/><D:quota-
available-bytes/></D:prop></D:propfind>'

<?xml version="1.0" encoding="utf-8"?>
<D:multistatus xmlns:D="DAV:" xmlns:ns0="DAV:">
<D:response xmlns:lp1="DAV:" xmlns:lp2="http://apache.org/dav/props/"
xmlns:lp3="LCGDM:">
<D:href>/dpm/cern.ch/home/dteam/</D:href>
<D:propstat>
<D:prop>
<lp1:quota-used-bytes>24677181319</lp1:quota-used-bytes>
<lp1:quota-available-bytes>75322818681</lp1:quota-available-bytes>
</D:prop>
<D:status>HTTP/1.1 200 OK</D:status>
</D:propstat>
</D:response>
</D:multistatus>
```

Going SRM-less with your DPM



Caching laboratory

- DPM 1.9 with Dome will allow investigation of operating **WLCG storage as a cache**
- Scenarios
  - Data origin a local federation of associate sites
  - Data origin the global federation
  - Hybrid cache/conventional setup

- A **volatile pool** can be defined which calls out to a stager on a miss
  - Caching logic implemented in a pluggable way

- Questions to investigate
  - Cache management logic
  - Different client strategies on miss
    - blocking read, async read, redirection to origin
  - Authentication solutions
  - Workflow adaptation for locality

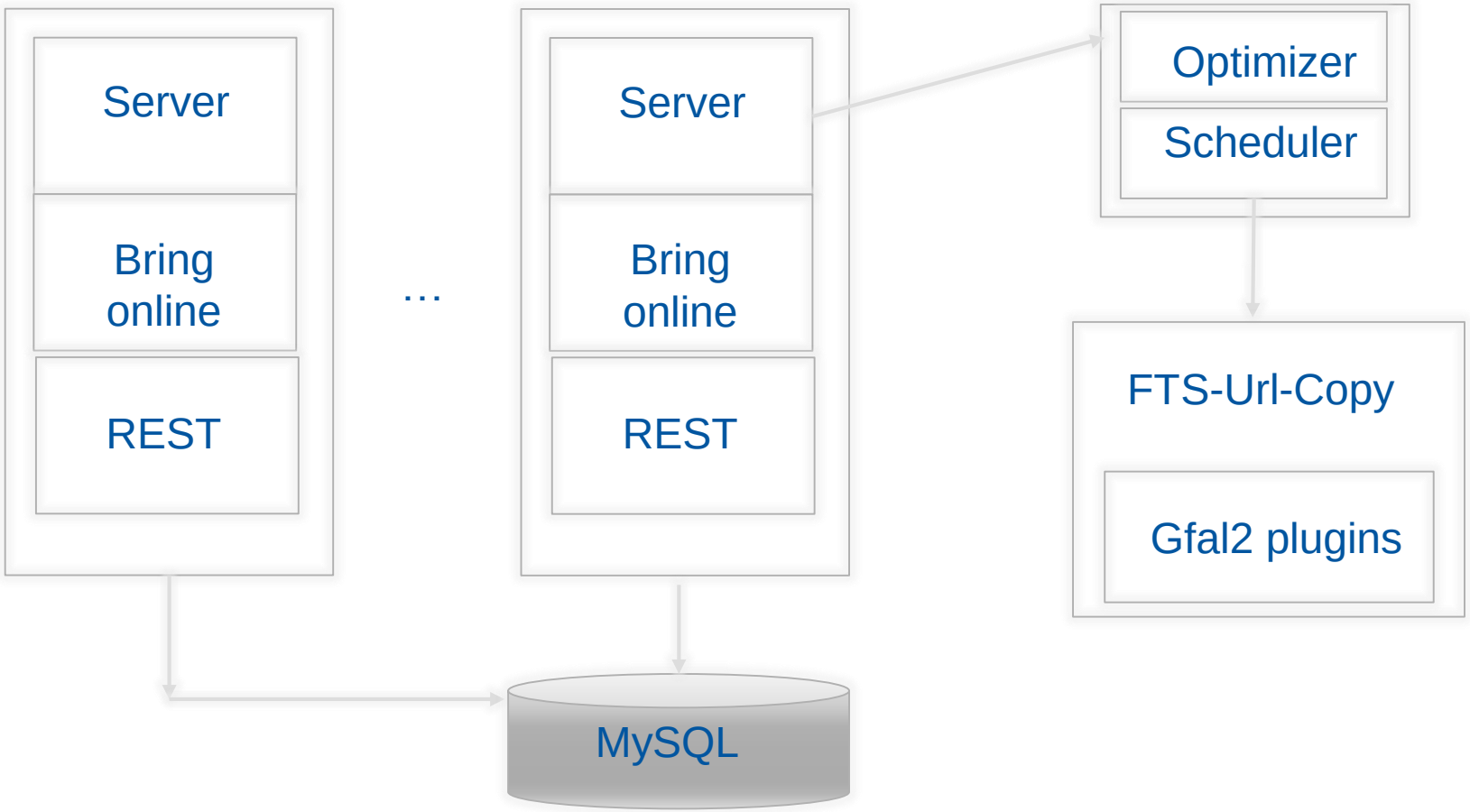
Outline

- CERN IT Storage group , AD section
- EOS
  - Namespace
- DPM
  - DOME
- **FTS**
  - **New optimizer**
  - **Object stores Integration**

FTS

- **FTS** is the service responsible for distributing the majority of LHC data across the WLCG infrastructure
- Is a low level data movement service, responsible for moving sets of files from one site to another while allowing participating sites to control the network resource usage
- WLCG stats:
  - Installed at: **CERN, RAL, BNL, FNAL**
  - **~20PB monthly volume / ~2.2M files per day**

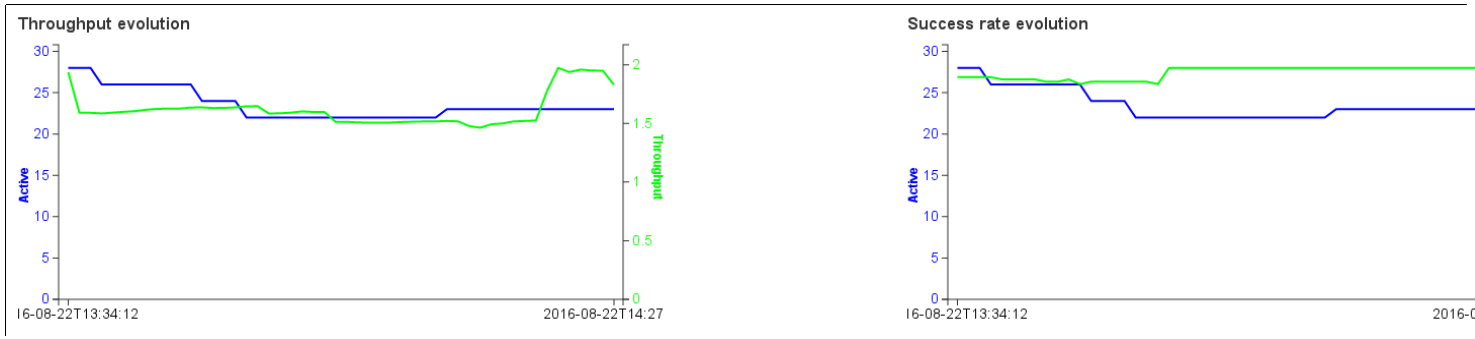
FTS Architecture

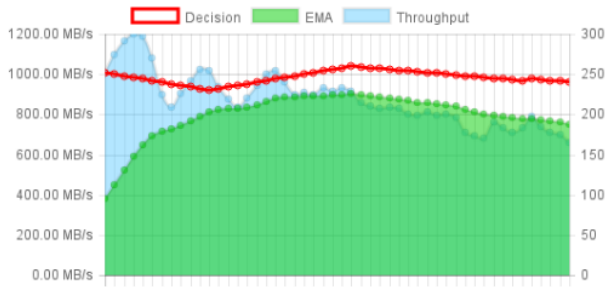


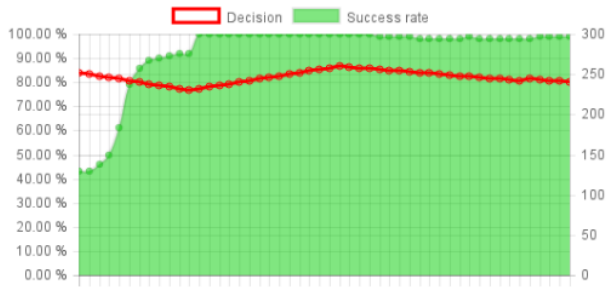
FTS 3.5: Optimizer changes

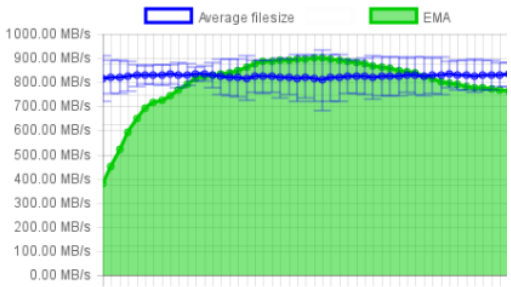
- **Min/max active ranges** can be configured per link
- Improved **throughput calculation**
- Softener ***Exponential Moving Average*** to reduce noise effects
- Throughput taken into account even with high success rates
- **1 stream per transfer by default**
  - Reduces resource consumption on the storages

FTS 3.5: Optimizer evolution









Plans for FTS 3.6

- Targeted towards Jan 2017
  - **Remove SOAP**
    - Only CMS is using it and it's already migrating to REST
  - **Remove Bulk Deletions**
  - New algorithm for **multiple replicas jobs**
  - Database profiling and optimizations

Object stores Integration

- Advantages
  - **Scalability** and performance achieved through relaxing or abandoning many aspects of posix
  - Applications must be aware or adapted
- How can such resources be plugged into existing WLCG workflows?
  - Can apply to public or private cloud
    - NB ceph at sites
- Data transfer -> FTS integration via davix/gfal2

davix

- davix-put /etc/services https://objbkt1.s3.amazonaws.com/file01 --s3secretkey <secret> --s3accesskey <access>
- davix-cp -P grid davs://dpm.cern.ch/dpm/cern.ch/home/dteam/file01 s3s://objbkt1.s3.amazonaws.com/file01 --s3secretkey <secret> --s3accesskey <access>

3rd party copy!!

gfal2/davix

- gfal-copy file:///etc/services s3://objbkt1.s3.amazonaws.com/file01
- gfal-copy davs://dpm.cern.ch/dpm/cern.ch/home/dteam/file01 s3://objbkt1.s3.amazonaws.com/file01

3rd party copy!!

FTS: Pre-signed URL

```
fts-transfer-submit --strict-copy  -s https://fts3.cern.ch:8446 https://dpm.cern.ch/dpm/cern.ch/home/dteam/file01 'https://objbkt1.s3.amazonaws.com/tf_04?Signature=eFAyXMWlSY%2BWEVcqfvGvuxZF6ZQ%3D&Expires=2105774242&AWSAccessKeyId=AKIAJZZQ2TYSEBKNVWKA'
```

First | Previous | 1 | Next | Last

File ID	File State	File Size	Throughput	Remaining	Start Time	Finish Time	Staging Start	Staging End	
+ 357604505	FINISHED	0 bytes	0.00 MB/s	-	2016-09-23T14:2	2016-09-23T14:2			Log

https://dpmhead-trunk.cern.ch/dpm/cern.ch/home/dteam/1.txt

https://objbkt1.s3.amazonaws.com/tf_04?Signature=6qe6joRXpoSFYdAI8Hm9Bjno4%2B8%3D&Expires=1474643908&AWSAccessKeyId=AKIAJZZQ2TYSEBKNVWKA

- Transfer host: fts106.cern.ch
- Staging host:
- PID: 9601
- Hash: 20EA
- Activity: default
- Selection strategy: auto
- Attempts: 0
- Duration: 0.379 seconds
- Checksum:
- User specified size: 0
- Configuration:
- Parameters: nostreams:1,timeout:0,buffersize:0
- Job finished: 2016-09-23T14:21:14
- Finished time: 2016-09-23T14:21:14
- Error reason:
- Log file:

  https://fts106.cern.ch:8449/var/log/fts3/transfers/2016-09-23/dpmhead-trunk.cern.ch__objbkt1.s3.amazonaws.com/2016-09-23-1421__dpmhead-trunk.cern.ch__objbkt1.s3.amazonaws.com__357604505__fb01b1fe-8198-11e6-8a3f-02163e00a39b

- Metadata:

  null

FTS: key management

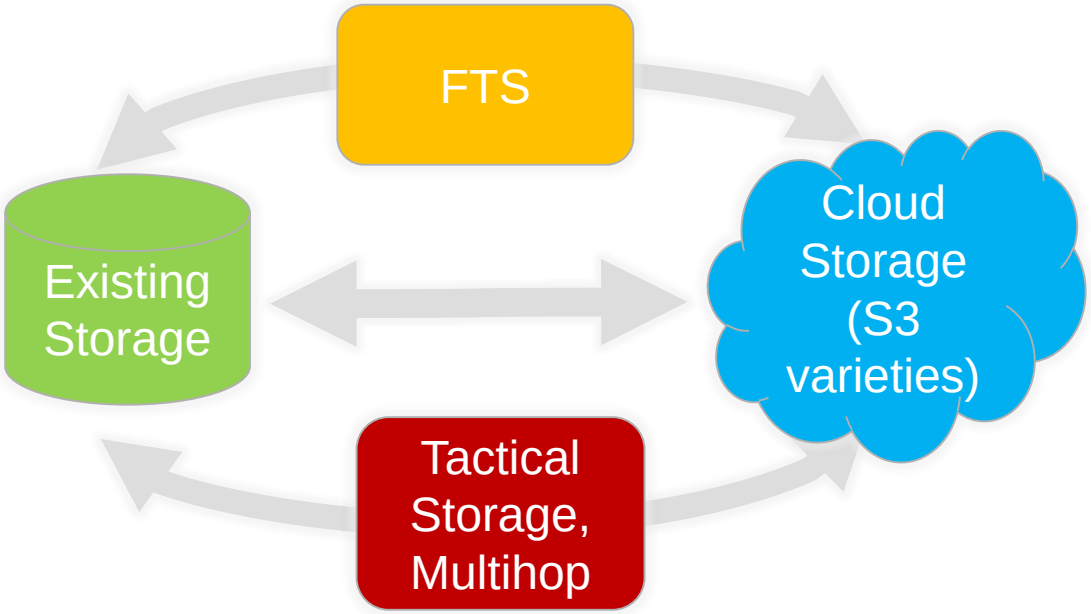
You can also allow FTS to hold the keys to your cloud storage

```
$curl[…] https://fts3devel01.cern.ch:8446/config/cloud_storage -H "Content-Type: application/json" -X POST -d '{"storage_name":"S3:s3.domain.com"}'

$curl[…]"https://fts3devel01.cern.ch:8446/config/cloud_storage/S3:s3.domain.com" -H "Content-Type: application/json" -X POST -d "${config}"

 {
    "vo_name": "dteam",
    "access_key": "ACCESS_KEY",
    "secret_key": "SECRET_KEY"
}
```

FTS: transport



- Solutions for **import to and export from clouds**
  - Several S3 variants supported
- Various architectures possible
  - **FTS gateway**
    - SRM<->S3
  - **3rd party transfer**
  - **Multi-hop with tactical storage**

References

- **EOS**
  - http://eos.web.cern.ch/
- **DPM**
  - http://lcgdm.web.cern.ch/
  - **DPM Workshop 2016, 23-24 Nov, Paris**
    - https://indico.cern.ch/event/559673/
- **FTS**
  - http://fts3-service.web.cern.ch/

CERN

www.cern.ch