EOS architectural evolution and strategic development directions

Georgios Bitzes, Fabio Luchetti (CERN), Andrea Manzi (CERN) Mihai Patrascoiu, Andreas Joachim Peters (CERN), Michal Kamil Simon (CERN), Elvin Alin Sindrilaru (CERN)
Overview

- Introduction
- Architecture Evolution
- New Features
- Directions
- Summary & Outlook
Project History

2009
- CASTOR
- XRootD
- castor-xrootd
- IF
- lustre-xrootd
- IF

2010
- EOS
- IPv4
- IPv4
- Lustre-xrootd
- IF

2011
- EOSATLAS
- EOSCMS
- EOSALICE
- EOSLHCB
- EOSPUBLIC
- IPv4
- v 0.2
- Amber
- v 0.3
- Beryl

2012
- EOSATLAS
- EOSCMS
- EOSALICE
- EOSLHCB
- EOSPUBLIC
- IPv4
- v 0.2
- Amber
- v 0.3
- Beryl
- v 0.3.x
- Aqua Marine

2013
- EOSATLAS
- EOSCMS
- EOSALICE
- EOSLHCB
- EOSPUBLIC
- IPv4
- v 0.2
- Amber
- v 0.3
- Beryl
- v 0.3.x
- Aqua Marine
- EOSUSER

2014
- EOSATLAS
- EOSCMS
- EOSALICE
- EOSLHCB
- EOSPUBLIC
- IPv4
- v 0.2
- Amber
- v 0.3
- Beryl
- v 0.3.x
- Aqua Marine
- EOSUSER

2015
- EOSATLAS
- EOSCMS
- EOSALICE
- EOSLHCB
- EOSPUBLIC
- IPv4
- v 0.2
- Amber
- v 0.3
- Beryl
- v 0.3.x
- Aqua Marine
- EOSUSER

2016
- EOSATLAS
- EOSCMS
- EOSALICE
- EOSLHCB
- EOSPUBLIC
- IPv4
- v 0.2
- Amber
- v 0.3
- Beryl
- v 0.3.x
- Aqua Marine
- EOSUSER

2017
- EOSATLAS
- EOSCMS
- EOSALICE
- EOSLHCB
- EOSPUBLIC
- IPv4
- v 0.2
- Amber
- v 0.3
- Beryl
- v 0.3.x
- Aqua Marine
- EOSUSER

2018
- EOSATLAS
- EOSCMS
- EOSALICE
- EOSLHCB
- EOSPUBLIC
- IPv4
- v 0.2
- Amber
- v 0.3
- Beryl
- v 0.3.x
- Aqua Marine
- EOSUSER

2019
- EOSATLAS
- EOSCMS
- EOSALICE
- EOSLHCB
- EOSPUBLIC
- IPv4
- v 0.2
- Amber
- v 0.3
- Beryl
- v 0.3.x
- Aqua Marine
- EOSUSER

Architecture evolution & strategic development directions

CERN 2019
What is EOS used for ...

- **disk storage**
  - raw data
  - analysis data
  - cernbox home & project spaces
  - cloudstore AARNet, Joint Research Centre JRC
  - Tier 2 & universities
  - online systems

- **tape storage cache**
  - Cern Tape Archive
Development Work Areas

- namespace architecture (MGM)
- storage consistency (FST)
- filesystem access (eosxd/ACLs)
- tape integration (CTA)
- protocols/API (ProtBuf, XrdHttp, GRPC)
- tokens & authorisation
Architectural Evolution

EOS 2017

Master-Slave Architecture

stateful meta-data service

Active-Passive Architecture

almost stateless meta-data service + Service Sharding scale-out meta-data performance

EOS 2019
Architectural Evolution

CERNBOX 2017

- EOSUSER
- 1TB RAM
- MGM
- 600M files
- at namespace scalability limit
- availability constrained by infrequent long boot time of 2h

CERNBOX 2019

- EOSHOME
- tested with >5B files
- namespace scalability limit by size of SSDs on QDB nodes
- automatic built-in HA mechanism for MGM failover

Architectural evolution & strategic development directions
Introduction of QuarkDB as persistent KV store for namespace meta-data

- based on REDIS protocol, RocksDB & RAFT consensus algorithm
- high-available, high-performance, scalable, low-latency
- extremely positive production experience

QDB api
- kv
- sets
- hashes
- pub-sub
- lease

C++ client library
https://gitlab.cern.ch/eos/qclient

QDB performance example: retrieve KV@200kHz

https://github.com/gbitzes/QuarkDB
QuarkDB Namespace

- Service startup time was major source for service downtime for in-memory namespace.

<table>
<thead>
<tr>
<th>Service startup time [s]</th>
<th>10000</th>
<th>100</th>
<th>10</th>
<th>3'600</th>
</tr>
</thead>
<tbody>
<tr>
<td>In-memory NS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QuarkDB NS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Meta-data service MGM stateless with configurable cache.

Meta-data persistency with QuarkDB.

Storage server FSTs.

Architectural evolution & strategic development directions

CHEP 2019
QuarkDB for HA

QDB provides support for leases to automatically fail-over meta-data server

- leases renewed every 5 seconds
- default validity is 10 seconds
- if a lease is required configuration is automatically reloaded and namespace becomes active
- service fail-over within few seconds

Lease Management

meta-data service MGM stateless with configurable cache

meta-data persistency with QuarkDB
File System Consistency
EOS v4.6

- re-engineering of FSCK functionality
  - over the past 9 years accumulated replication inconsistencies in EOS instances
  - with transition to QuarkDB filesystem consistency check & repair broken
- FSCK components
  - **backward consistency check**: compare filesystem contents to namespace - size, checksum, layout
    - data scanner with inconsistency flagging & checksumming for each filesystem - by default all data scanned within one week
  - **forward consistency check**: compare namespace to filesystem contents
    - MGM scanner identifying missing replicas on filesystems
  - **repair engine** error collection & automatic repair actions
File System Access

**eosxd**

avg. > 20k mount clients @ CERN for CERNBOX

### Latest developments

- **OIDC** support as kerberos/x509 replacement
- **Snapshot** support with COW functionality for consistent backups
- **Squashfs** integration for software distribution

### Callback Architecture

- **MGM**
- **FuseServer**
- **ZeroMQ**
- **callback network**
- **modification**
- **client 1**
- **client 2**
- **client 3**
- **/dir**

### Web App

- **OpenID**
- **/eos**
- **eosxd**
- **Web App**

### Architectural evolution & strategic development directions
Tape Integration

EOS + Tape = EOSCTA

integrated support for tape into EOS
file on tape=offline replica
- loose service coupling between EOS and CTA via protocol buffer interface & notification events - everything is synchronous
- no SRM, using XRootD protocol only - integrated with FTS

Operation Model:
- High disk capacity
- Low disk capacity

EOSATLAS

EOSATLASCTA

short file lifetime

Cern Tape Archive

Architectural evolution & strategic development directions
Protocol Support

**GRPC support** with token and x509 support
- mapping applications identity using GRPC token=>(uid,gid) or DN=>(uid,gid) mapping

**Namespace interface**
- metadata injection - used for Castor=>CTA meta-data migration
- mkdir|rmdir|touch|rm|unlink|ls|find|rename|symlink|setxattr|chown|chmod|acl|token|create-version|list-version|purge-version with streaming support for large responses

**HTTP(S) support** with token and x509 support
- using XrdHttp and external handler

**HTTP TPC / XRootD with delegation support**
- using default proxy server in front of EOS instances on gateway machines

**S3 support** with MINIO gateway
- via plug-in for MINIO developed by AARNet - *currently not deployed at CERN*
EOS Tokens

Bearer Token Support preparing coming WLCG authz changes

Proprietary format
- Serialized PROTOBUF structure + ZLIB Compression + Base64URL encoding

Token carries
- a namespace scope file, directory or tree
- an ACL entry replacing locally stored ACLs - no need to invent new syntax like UPLOAD, DOWNLOAD…
- an optional role e.g. the owner when creating a file
- an optional set of origin restrictions - which clients can use this token and how do they have to be authenticated
  - we can enforce additional strong authentication if a bearer wants to use a token
- a generation value allows immediate token revocation of a given generation
- an expiration time
EOS Tokens

JSON representation

```

{  
  "token": "mytoken",  
  "clientid": "myclient",  
  "expires": "2019-12-31T23:59:59Z",  
  "owner": "",  
  "group": "",  
  "generate": false,  
  "path": "/eos/myusername/",  
  "filename": "example.txt",  
  "token": "mytoken",  
  "macro": "mymacro",  
  "requester": "username@example.com",  
  "priority": 1,  
  "verification": "myverify",  
  "signature": "mysignature",  
  "format": "myformat",  
  "timeout": 300
}
```

Usage
token as filename or CGI authz=<token> usable with XRootD, HTTP, GRPC, eosxd (fuse)

Creation

```
# as a filename
xcdcp root://myeos//zteos64:MDAwMDAwNzR4nONS4WIuKq8Q-Dlz-ltWI3H91Pxi_cSsAv2S_OzUpp2SeAgtPMAY7fie31Ts-od-rgcLZ_a2_bhwcZ09cracy /tmp/

# via CGI
xcdcp "root://myeos//myfile?authz=zteos64:MDAwMDAwNzR4nONS4WIuKq8Q-Dlz-ltWI3H91Pxi_cSsAv2S_OzUpp2SeAgtPMAY7fie31Ts-od+rgcLZ_a2_bhwcZ09cracy" /tmp/

eos token --path /eos/myfile --expires $LATER
zteos64:MDAwMDAwNzR4nONS4WIuKq8Q-Dlz-ltWI3H91Pxi_cSsAv2S_OzUpp2SeAgtPMAY7fie31Ts-od-rgcLZ-a2-bhwcZ09cracyhmb3c6jpRIeWWOws710x6xAABeTC8I
```
EOS Tokens

How can they be used?

- usable by applications for restricted on-behalf access via any supported access method - even fuse mounts
- can be used by CERNBOX services to provide shares and delegate permissions
- as internal format for external tokens WLCG/ALICE tokens
- as single file token like signed S3 URLs are used

http://eos-docs.web.cern.ch/eos-docs/using/tokens.html
General Directions

consolidation of new architecture, improvement of reliability & consistency and optimisation of internal storage services to profit from QuarkDB

look at MD Scale-out without service sharding subtree assignment to MGMs

support HTTP eco-system: establish GRPC as MD API, DAV as Data API for front-end CERNBOX possibly also GRPC+flatbuffers as DATA API

establish/support tokens for applications and GRID access

focus on erasure coding
pre-defined conversion policies for files from/to EC layouts
light-weight object storage for sequential access & archiving use-cases - client-driven
Invitation

4th edition EOS workshop 2020

3rd to 5th of February @ CERN

platform for exchange between developers, users, sites and people interested in storage technology

disk - tape - cloud - sync & share - devops

Architectural evolution & strategic development directions
Code health in EOS: Improving test infrastructure and overall service quality

EOS architectural evolution and strategic development directions

Erasure Coding for production in the EOS Open Storage system

Evolution of the filesystem interface of the EOS Open Storage system

Seeking an alternative to tape-based custodial storage

Using the RichACL Standard for Access Control in EOS

CERN Tape Archive: production status, migration from CASTOR and new features

CERN Disk Storage Services: report from last data taking, evolution and future outlook towards Exabyte-scale storage

Migration of user and project spaces with EOS\CERNBox: experience on scaling and large-scale operations

Converging to Kubernetes for on-premise and hybrid clouds for CERNBox, SWAN, and EOS

https://eos.cern.ch