DSS

Data & Storage Services

CERN IT Department

EOS - CERN Disk Storage
EXPLORATION OF STORAGE!

The EOS Storage System

Andreas J. Peters

CERN IT-DSS group

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

Alice Offline Week – 10.11.2011

CERN

Outline

- EOS status and production experience
  - EOS Architecture
  - Operations at CERN
  - Benchmarks
  - License
  - Code Base & Pointers
  - Roadmap/Outlook

EOS: What is it ...

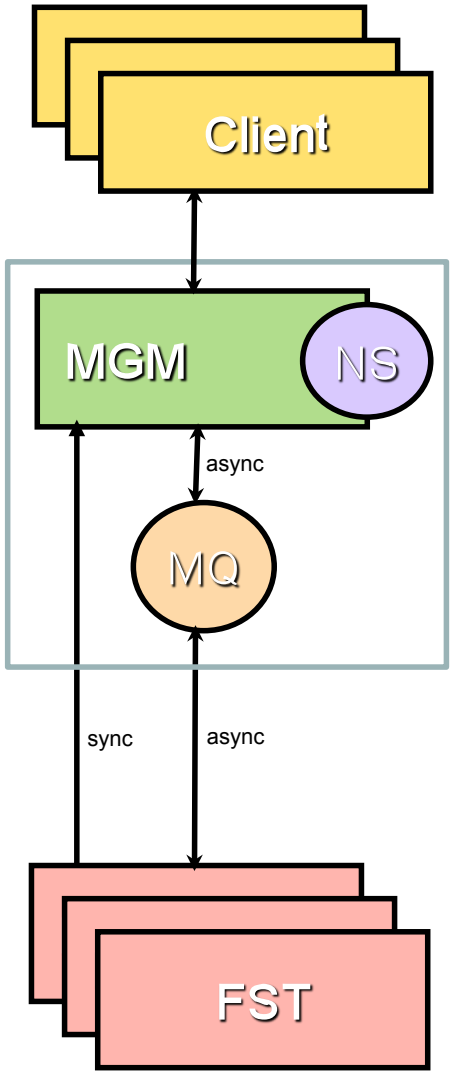
- Easy to use standalone **disk-only** storage for user and group data with **in-memory** namespace
  - **Few ms** read/write open latency
  - Focusing on end-user analysis with chaotic access
  - Based on ***XROOT*** server plugin architecture
  - Adopting ideas implemented in *Hadoop*, *XROOT*, *Lustre* et al.
  - Running on low cost hardware
    - no high-end storage
  - At CERN: Complementary to CASTOR

Architecture

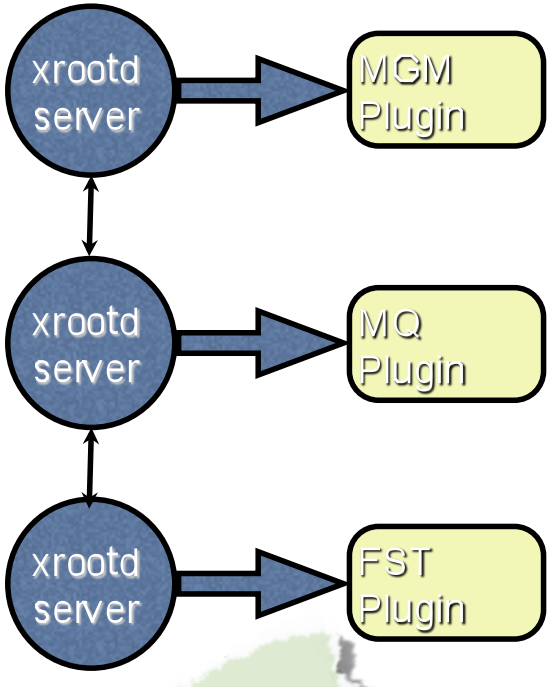
Management Server
Pluggable Namespace, Quota
Strong Authentication
Capability Engine
File Placement
File Location

Message Queue
Service State Messages
File Transaction Reports
Shared Objects (queue+hash)

File Storage
File & File Meta Data Store
Capability Authorization
Check-summing & Verification
Disk Error Detection (Scrubbing)



Implemented as plugins in **xrootd**



Access Protocol

- EOS uses XROOT as primary file access protocol
  - The **XROOT framework** allows flexibility for enhancements
- Protocol choice is not the key to performance as long as it implements the required operations
  - Client caching matters most
    - Actively developed, towards full integration in ROOT (rewrite of XRootD client at CERN)
- SRM and GridFTP provided as well
  - BeStMan, GridFTP-to-XROOT gateway
- Single & Multi User FUSE Mount (experimental)

Features

- Storage with single disks (JBODs, no RAID arrays)
  - redundancy by s/w using cheap and unreliable h/w
- Network RAID within disk groups
  - Currently file-level replication
- Online file re-replication, re-organization
  - Aiming at reduced/automated operations
- Tunable quality of service
  - Via redundancy parameters (directory based)
- Optimized for reduced latency
  - Limit on namespace size and number of disks to manage
    - Currently operating with hardware limit of **40M** files and **10K** disks
- Achieving additional scaling by partitioning the namespace
  - Implemented by deploying separated instances per experiment

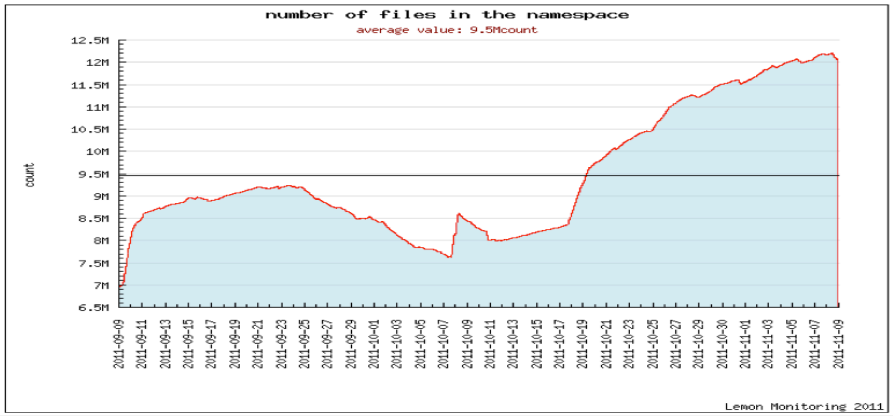
Self-healing

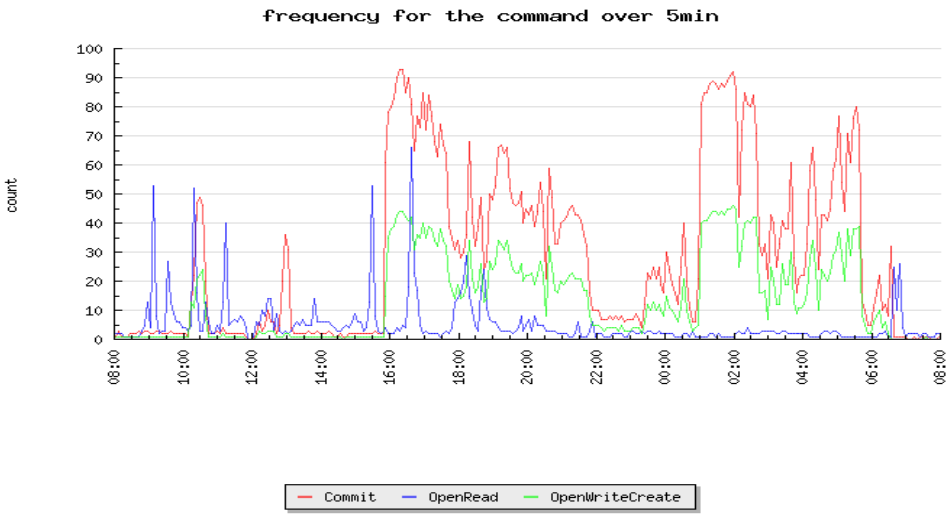
- Failures don't require immediate human interventions
  - Metadata server (MGM) failover
  - Disks drain automatically triggered by I/O or pattern scrubbing errors after a configurable grace period
- Drain time on production instance < 1h for 2 TB disk (10-20 disks per scheduling group)
  - Sysadmin team replaces disks 'asynchronously', using admin tools to remove and re-add filesystems
    - Procedure & software support is still undergoing refinement/fixing
- Goal at CERN: run with best effort support

Entering production

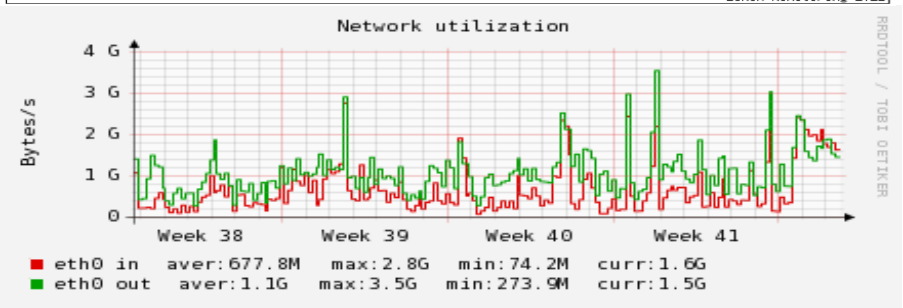
- Field tests done (Oct 2010 – May 2011) with ATLAS and CMS, production since summer
- EOS 0.1.0 currently used in EOSCMS/EOSATLAS
  - Software in bug-fixing mode, frequent releases though
- Pools migration from CASTOR to EOS mostly done
  - Currently at **2.3 PB** usable in CMS, **2.4 PB** in ATLAS
  - Required changes in ATLAS/CMS experiment frameworks
    - User + quota management, user mapping
    - Job wrappers
    - => no change in ALICE model
  - Several pools already decommissioned in CASTOR
    - E.g. CMSCAF

Usage Statistics

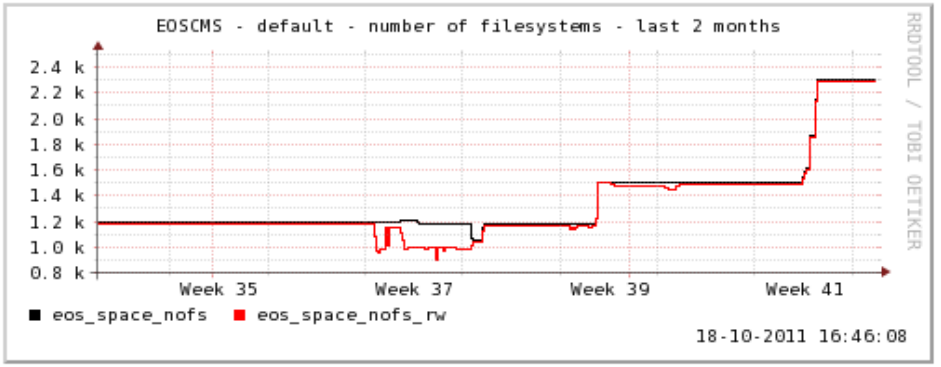




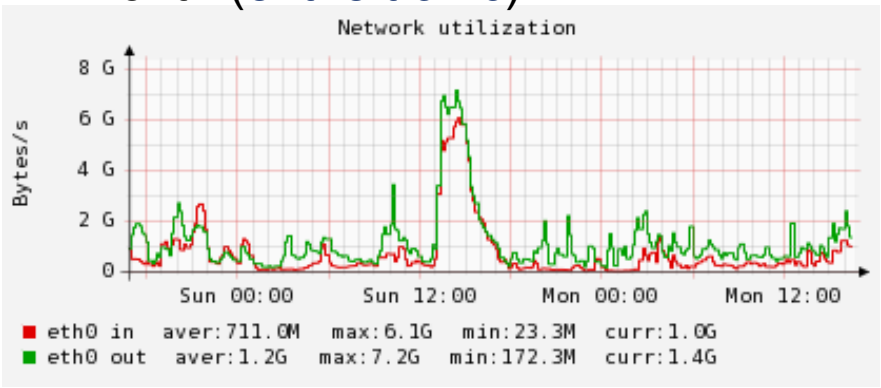
ATLAS instance: file ops per second



ATLAS instance: throughput over 1 month (entire traffic)



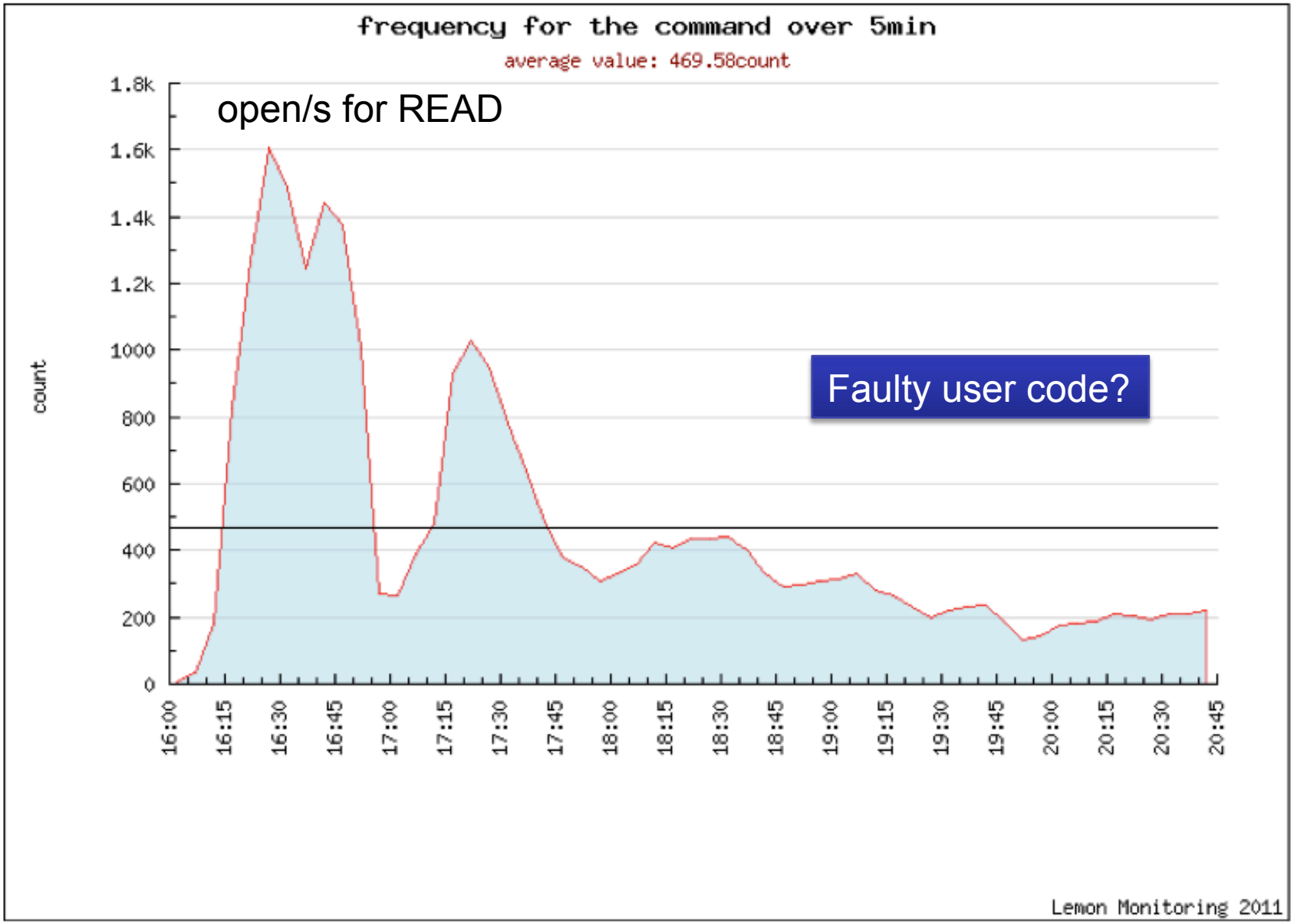
CMS instance: hardware evolution



Pool throughput during a node drain

Production Performance

EOSATLAS yesterday



Performance – Why EOS?

Latest Namespace Benchmark EOS 0.1.1-1

0-Byte Files

Namespace 1 Mio entries - 512 Server Threads

1x8-core MGM + 8xStorage FSTs

350 & 10.000 ROOT Clients:
3kHz File Open O_CREAT 1 rep
2kHz File Open O_CREAT 2 rep
8kHz File Open O_RDONLY (2ms)
26 & 20 kHz stat

Find

Namespace 1 Mio entries

190 kHz returning
Lfn, Size & Checksum

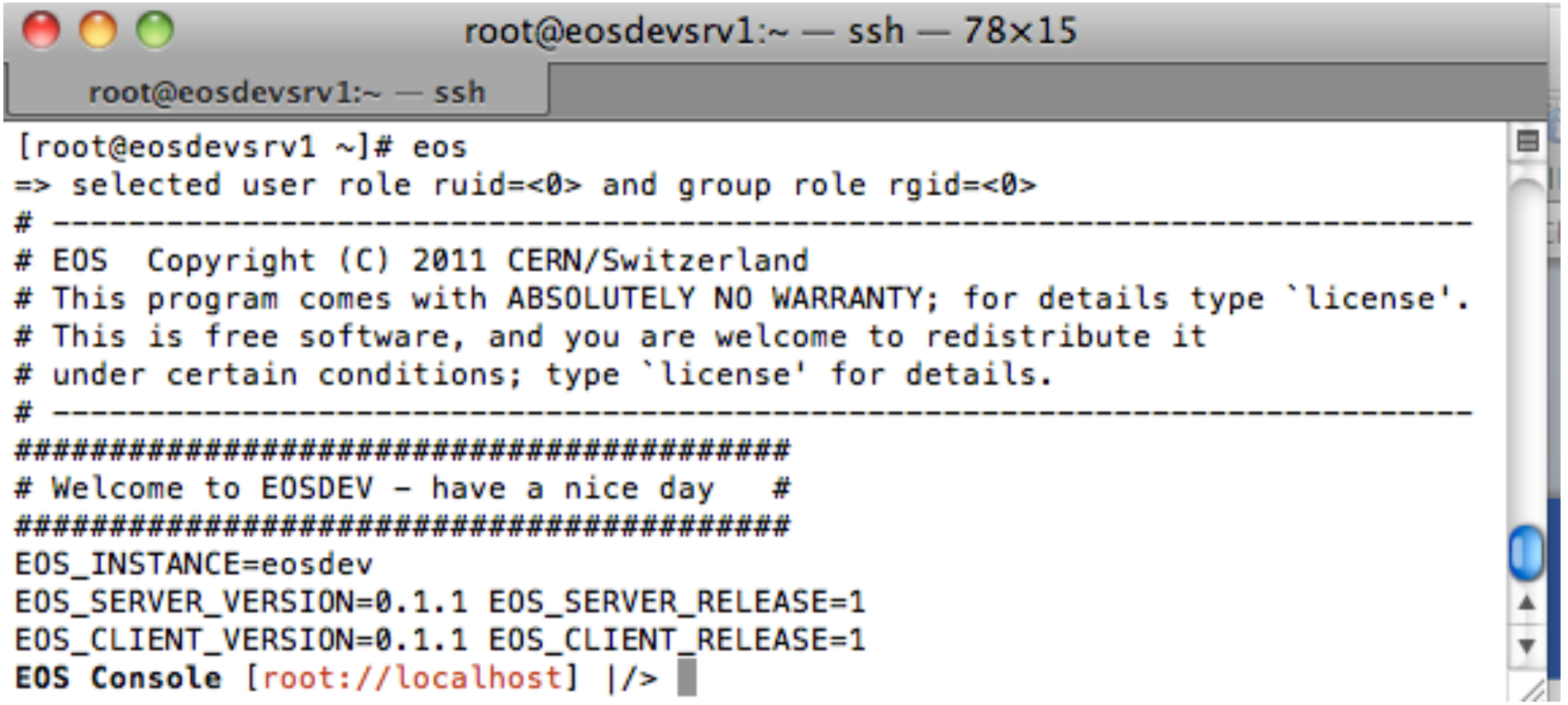
Boot

12M Entries Namespace

170 kHz = 70s Boot Time
1kB per Entry in Memory

License – GPL3

Included since Release 0.1.1-1



EOS Project

Contact

EOS Project

Name	Position	E-mail
Project Mailing List		project-eos@cern.ch

EOS Development

Name	Position	E-mail
Andreas Peters	Core Developer	Andreas.Joachim.Peters@cern.ch
Lukasz Janyst	Core Developer	Lukasz.Janyst@cern.ch
Elvin Alin Sindrilaru	Core Developer	Elvin.Alin.Sindrilaru@cern.ch

EOS Operation

Name	Position	E-mail
Jan Iven	Operation Manager	Jan.Iven@cern.ch
Luca Mascetti	Operation Manager	Luca.Mascetti@cern.ch

IT-DSS Group

Name	Position	E-mail
Alberto Pace	Group Leader	Alberto.Pace@cern.ch
Dirk Duellmann	Development Section Leader	Dirk.Duellmann@cern.ch
Massimo Lamanna	Operations Section Leader	Massimo.Lamanna@cern.ch

EOS Code Base

- cmake project C++ (75k lines)
- requires gcc >=4.4
- supported platforms
  - server: SLC5/SCL6 64 bit
  - client: SLC5/SLC6 32+64 bit + OSX
- Dependencies
  - build: cmake, xrootd dev, sparsehash, cppunit, attr …
  - runtime: xrootd,crypto,ncurses,uuid,z

EOS Software Distribution

- Source: GIT http://eos.cern.ch/cgi-bin/cgit.cgi/eos/
- RPMS:
  - Releases http://eos.cern.ch (not yet open)
  - HEAD Build Server https://teamcity-dss.cern.ch:8443/
- XRootD RPMS via http://xrootd.org
- EOS AFS Client Installation

  `/afs/cern.ch/project/eos/`

Release / Roadmap

- EOS 0.1.0 (xrootd 3.0.4) deployed at CERN
  - 0.1.0-rc42
- EOS 0.1.1 (xrootd 3.1.0) waiting freeze&deployment
  - 0.1.1-1
- EOS 0.2.0 expected by end of the year
  - Main Features
    - File-based redundancy over hosts
      - Dual Parity Raid Layout Driver (4+2)
      - ZFEC Driver (Reed-Solomon, N+M, user defined)
      - Integrity & recovery tools
    - Client bundle for User EOS mounting (krb5 or GSI)
      - MacOSX
      - Linux 64bit

EOS in the ALICE Model

- ALICE authorization works like in standard XRootD setups
- EOS can be subscribed easily to a global redirector
- Third party copy
  - xrd3cp not natively supported AS
  - FTSOFS (xrd3cp) can be deployed and run via a gateway xrootd server on additional ports to push from EOS
  - EOS will provide '**eos 3pget** & **3pput**' via external transfer queues (uses already internal transfer queues between EOS disk server)
  - Still hoping for standard XRootD solution!
- For ROOT, aliensh etc. looks like any XRootD storage – no API changes
- Offers many operational advantages

Conclusions

- EOS is in production for analysis
  - Two production instances running
    - result of very good cooperation with experiments
    - today managing ~4.600 disks at CERN (9.2 PB raw space)
  - Expand usage and gain more experience
  - Outside Usage
    - No organized support for outside usage
      - But: one CMS instance at Fermilab with low effort
      - CERN deployment is very large scale => don't expect particular problems in small installations, particular when storing on RAID arrays
  - Move from fast development and release cycles to reliable production mode
- EOSALICE (?)