High performance storage pools for LHC

Łukasz Janyst

on behalf of the CERN IT-DSS group

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

Introduction

- The goal of EOS is to provide a relatively simple and scalable solution for the analysis users

- In this presentation I will:
  - Introduce EOS
  - Discuss some of the implementation details
  - Present test results

EOS requirements

- RW file access (random and sequential reads, updates)
- Fast Hierarchical Namespace
  - Target capacity: 10^9 files, 10^6 containers (directories): we're not quite there yet although things look promising
- Strong authorization
- Quota
- Checksums
- Redundancy of services and data
- Dynamic hardware pool scaling and replacement without downtimes

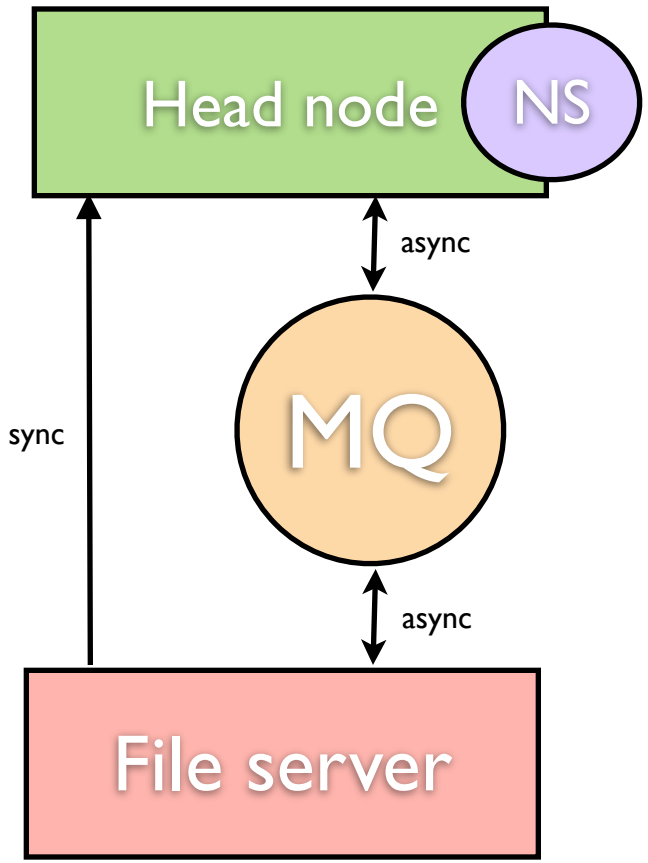
EOS timeline

- Started in April 2010 with storage discussions within the IT-DSS group
- The development started in May
- Prototype is under evaluation since August
  - Tested within the ATLAS Large Scale Test (pool of 1.5PB)
  - Will be open to individual users from Nov 15
  - Will be run by the CERN operations team
- Still much work left to do
- Currently we're focusing on ironing out the operational procedures

Selected features of EOS

- Is a set of XRootd plug-ins
- Speaks XRoot protocol
- Just a Bunch Of Disks (JBOD - no RAID arrays)
- Network RAID within node groups
- Tunable per-directory settings
  - User have a choice, by choosing the number of replicas, to achieve arbitrary level of availability/performance
- Fault tolerant
  - From client's point of view, all files are always readable and writable

EOS Architecture



Head node

Namespace, Quota
Strong Authentication
Capability Engine
File Placement
File Location

Message Queue

Service State Messages
File Transaction Reports

File Server

File & File Meta Data Store
Capability Authorization
Checksumming & Verification (adler,crc32,md5,sha1)
Disk Error Detection (Scrubbing)

EOS Namespace

- In the typical HEP use case a number of active files at any given moment is rather small even though the overall number of files stored in the system may be huge.
- EOS tries to leverage this fact to provide a memory-based namespace

EOS Namespace

Version 1 (current implementation)

- In-memory hierarchical namespace using google hash
- Stored on disk as a changelog file
- Rebuilt in memory on startup
- Two views:
  - hierarchical view (directory view)
  - view storage location (filesystem view)
- very fast, but limited by the size of memory
  - 1GB = ~1M files

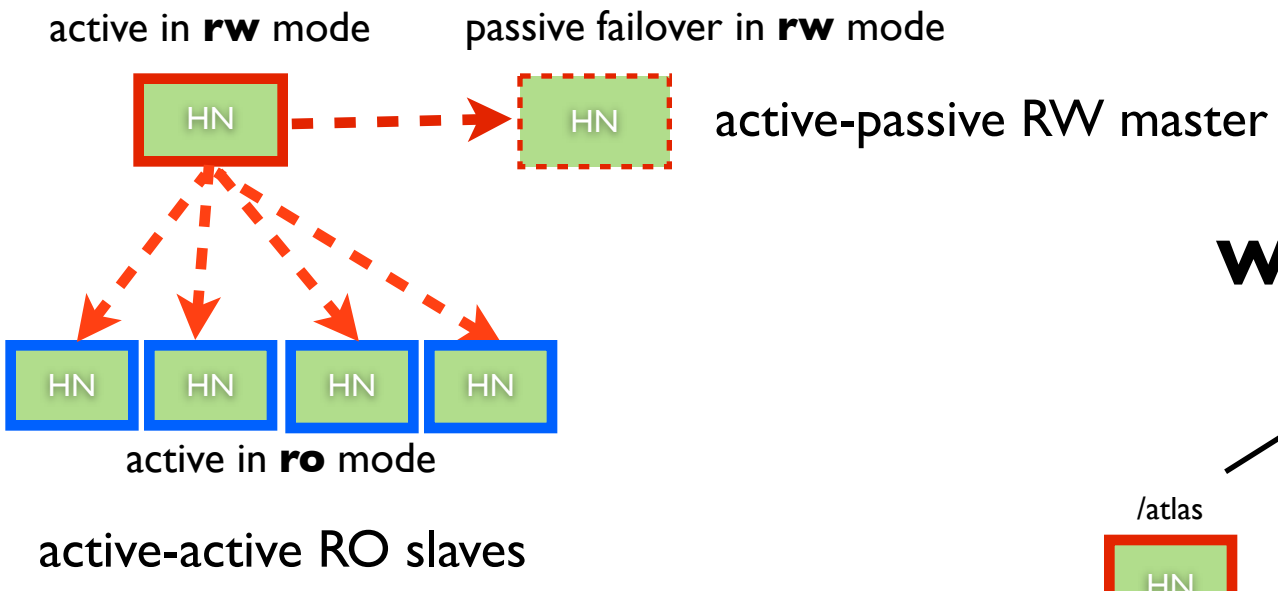
EOS Namespace

Version 2 (current development)

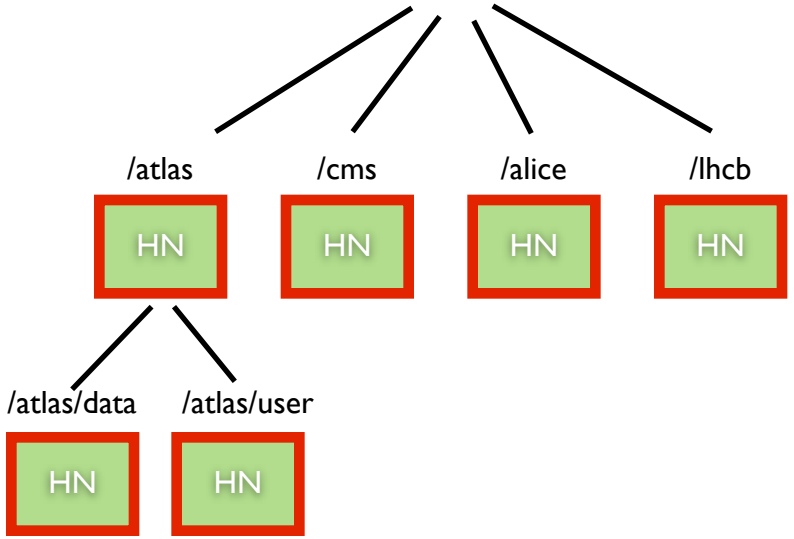
- Only view index in memory
  - Metadata read from disk/buffer cache
  - Perfect use case for SSDs (need random IOPS)
- 10^9 files = ~20GB per view

HA and Namespace scaling

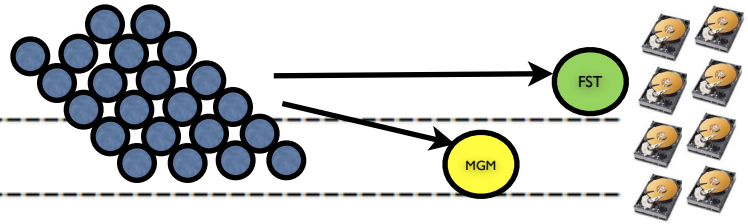
HA & Read Scale out



Write Scale out



File creation test



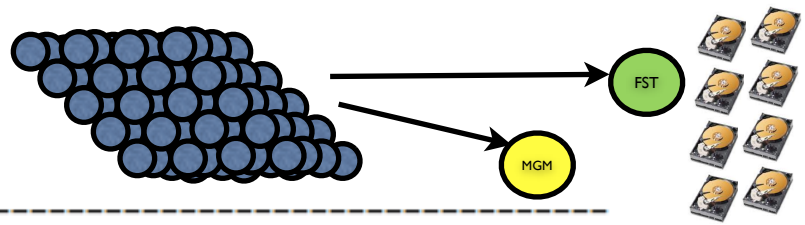

```
EOS Console [root://localhost] |/> ns stat
# ------------------------------------------------------------------------------------
# Namespace Statistic
# ------------------------------------------------------------------------------------
ALL      Files                            5008898
ALL      Directories                      5073
# ------------------------------------------------------------------------------------
who      command                          sum          5s      1min      5min       1h
# ------------------------------------------------------------------------------------
ALL      Commit                           5006939    926.00   1104.64   1054.88   914.63
ALL      Exists                           5007435    926.00   1104.66   1054.88   914.63
ALL      Find                             0000005      0.00      0.00      0.00     0.00
ALL      Mkdir                            0005022      1.25      1.10      1.05     0.92
ALL      Open                             5007195    926.00   1104.66   1054.88   914.63
ALL      OpenDir                          0000010      0.00      0.00      0.00     0.00
ALL      OpenFailedQuota                  0000240      0.00      0.00      0.00     0.00
ALL      OpenProc                         0000151      0.25      0.03      0.01     0.02
ALL      OpenWriteCreate                  5006955    926.00   1104.66   1054.88   914.63
ALL      OpenWriteTruncate                0000240      0.00      0.00      0.00     0.00
ALL      Rm                               0000240      0.00      0.00      0.00     0.00
ALL      Stat                             0000413      0.00      0.00      0.00     0.11
```

1 kHz

NS Size: 10 Mio Files
* 22 ROOT clients 1 kHz
* 1 ROOT client 220 Hz

```
  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
23681 daemon    20   0 8173m 7.9g 4356 S    0  6.3   0:48.55 xrootd
```

File read test




```
EOS Console [root://localhost] |/> ns stat
# ------------------------------------------------------------------------------------
# Namespace Statistic
# ------------------------------------------------------------------------------------
ALL      Files                            10079235
ALL      Directories                      10139
# ------------------------------------------------------------------------------------
who      command                          sum          5s      1min      5min       1h
# ------------------------------------------------------------------------------------
ALL      Commit                           0001742      0.00      0.00      0.00      0.00
ALL      Exists                           0003220      0.00      0.00      0.00      0.00
ALL      Open                             100069124    6785.00   6764.90   6697.50   7096.28
ALL      OpenFailedQuota                  16645985       0.00      0.00      0.00    751.41
ALL      OpenProc                         0000527      0.50      0.19      0.05      0.02
ALL      OpenRead                         100066929    6784.75   6764.90   6697.50   7096.28
ALL      OpenWriteCreate                  0000262      0.00      0.00      0.00      0.00
ALL      OpenWriteTruncate                0001479      0.00      0.00      0.00      0.00
ALL      Rm                               0001479      0.00      0.00      0.00      0.00
```

7 kHz

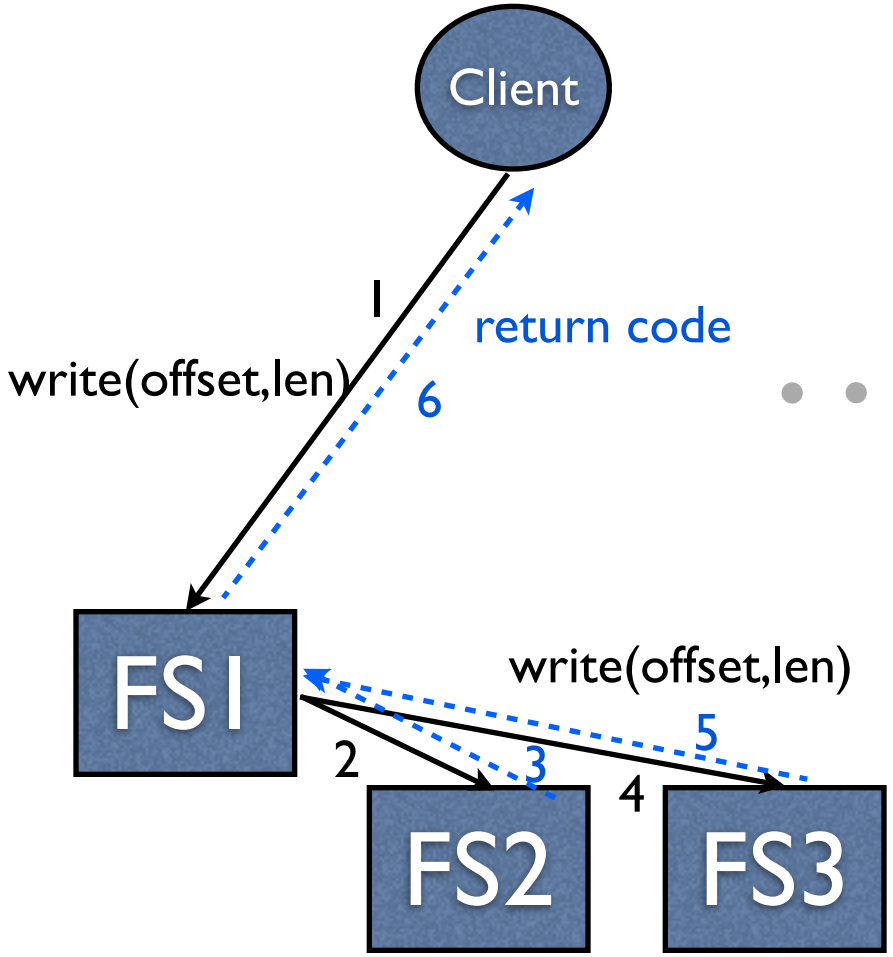
NS Size: 10 Mio Files
* 100 Million read open
* 350 ROOT clients 7 kHz
* CPU usage 20%

File layouts

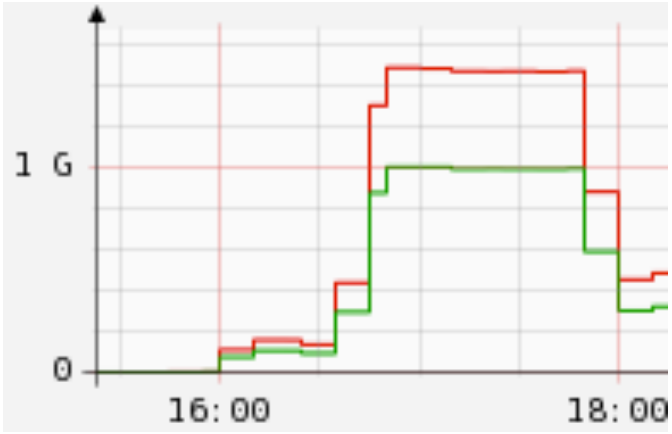
EOS supports layouts

- plain
- replica (synchronous replication)
- RAID5 (untested prototype)

Replica layout

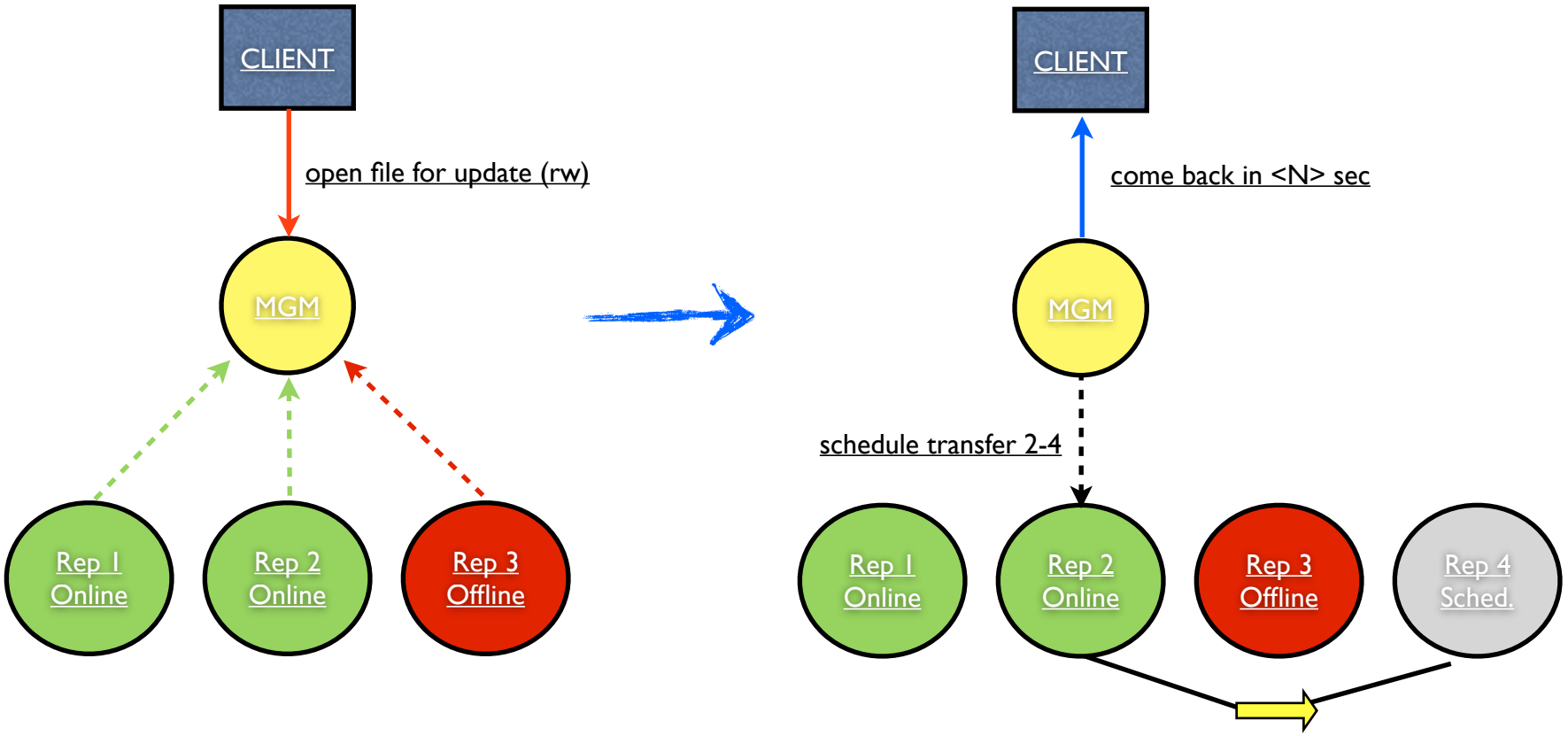


Network IO for file creations with 3 replicas:



500 MB/s injection result in
- 1 GB/s output on eth0 of all disk servers
- 1.5 GB/s input on eth0 of all disk servers

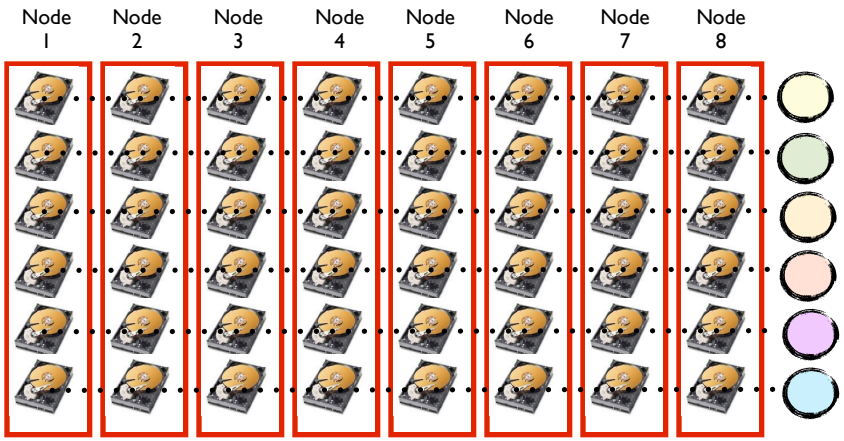
Replica healing



Client **rw** reopen of an existing file triggers

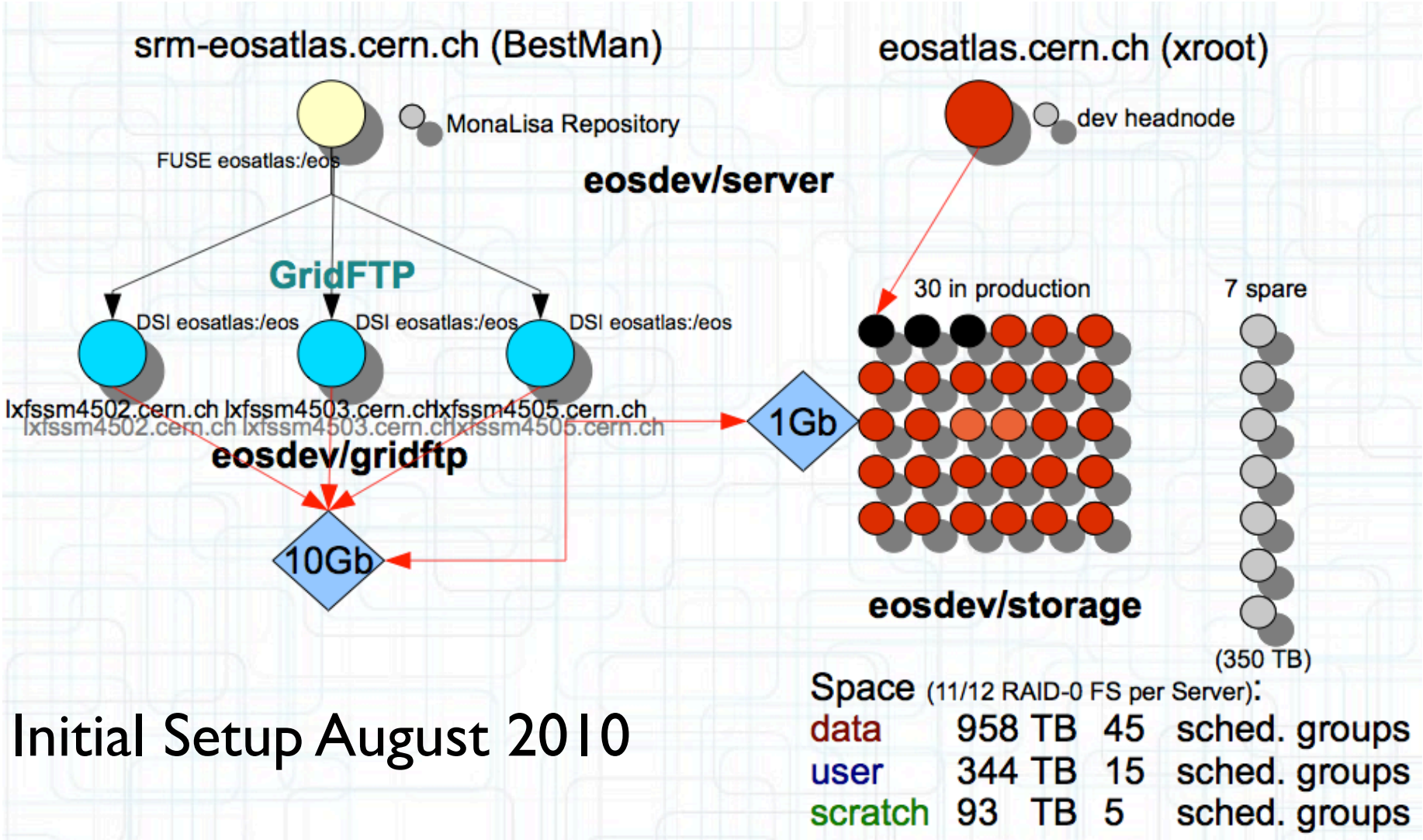
- creation of a new replica
- dropping of offline replica

Replica placement



- In order to minimize the risk of data loss we couple disks into scheduling groups (current default is 8 disks per group)
- The system selects a scheduling group to store a file in in a round-rubin manner
- Then all the other replicas of this file are stored within the same group

EOS Atlas

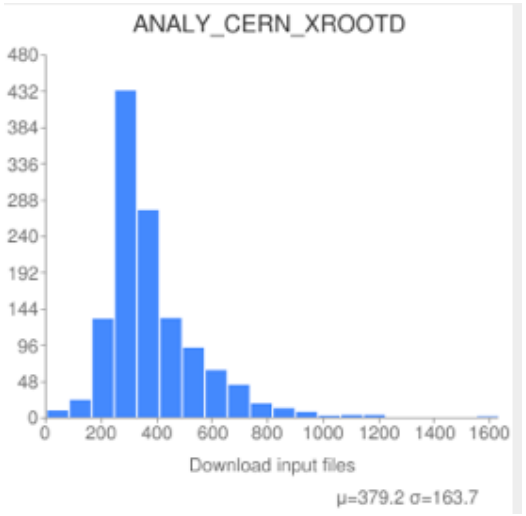


Initial Setup August 2010

Hammer Cloud Test

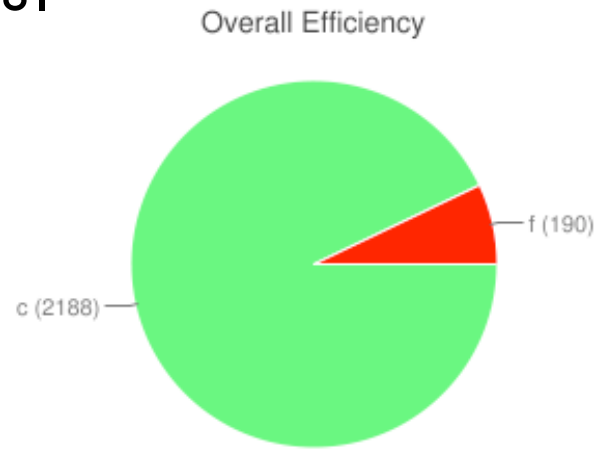
HC 10001181

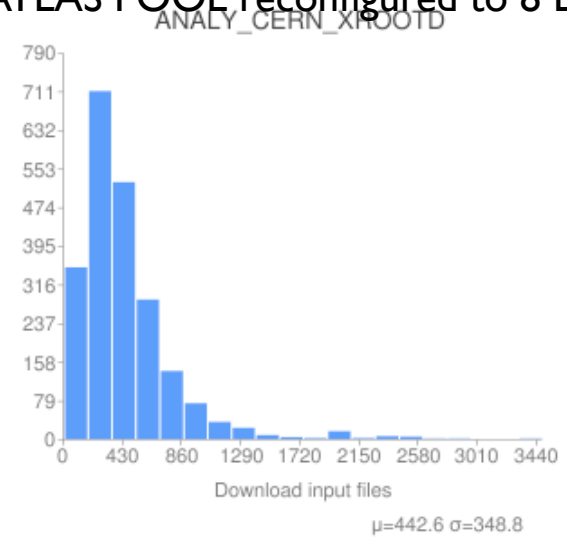
EOSATLAS POOL 27 Disk Server RAID0



HC 10001381

EOSATLAS POOL reconfigured to 8 Disk Server JBOD

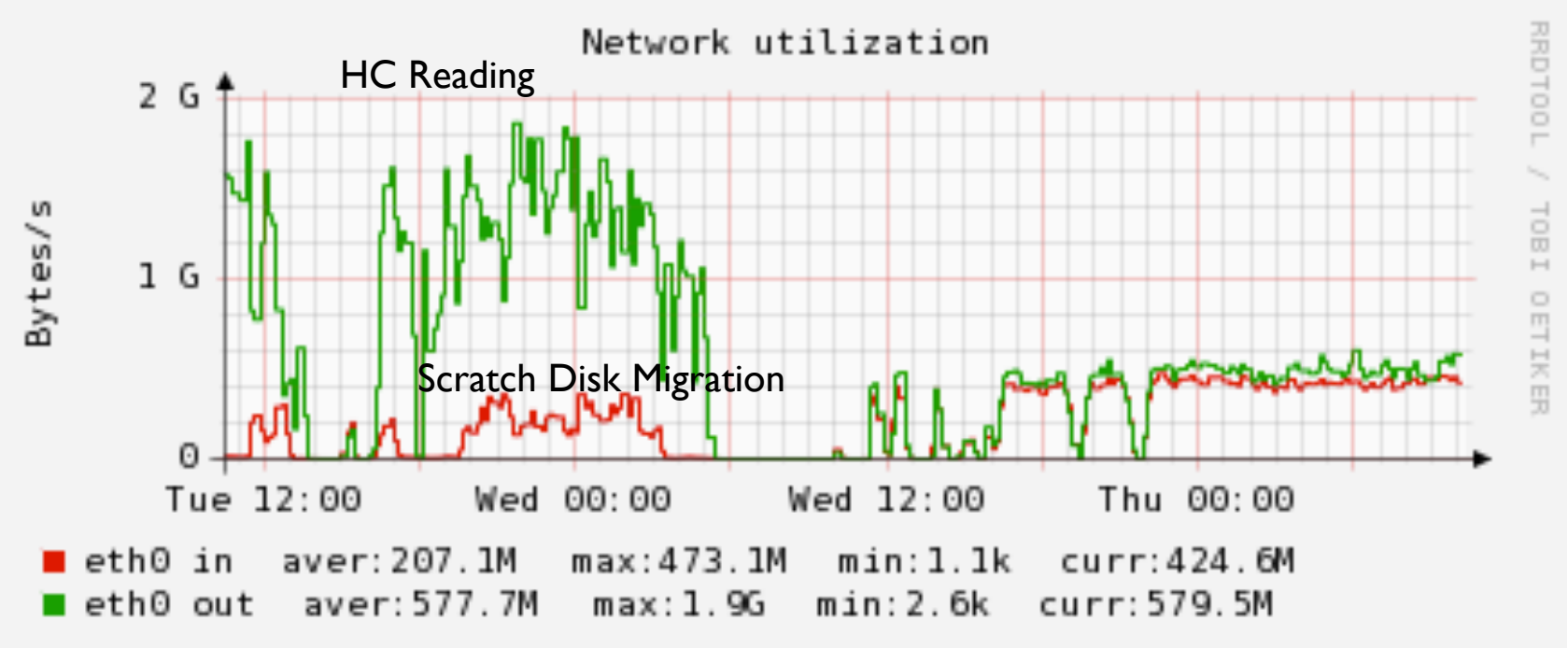




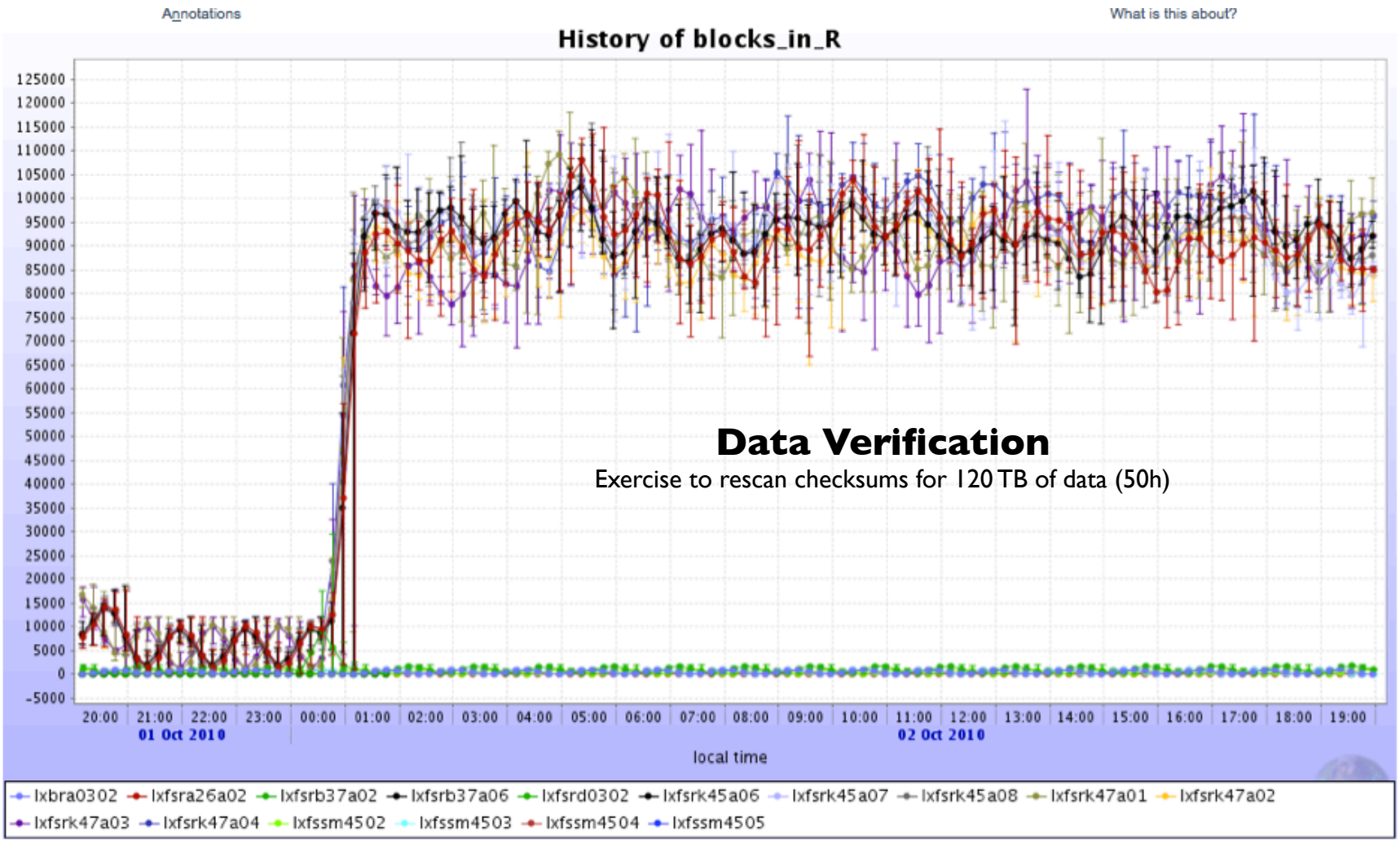
Hardware replacement test

Life Cycle Management

Exercise to migrate
27 disk server with 10 Raid-0 FS to 8 new with 20 JBOD FS
[partially overlapped with HC Tests]



Data verification test



Plans

- Operation
  - Opening EOS Atlas to Atlas users on Nov 15
  - Run by the CERN Castor operations team
- Development
  - Implement Version 2 of the namespace
- Larger Testbed (if available)
  - Scale the instance from todays 600 disks to 2000-8000 disks (4-16PB)

Summary

- We have been able to build the prototype rapidly
- We've been able to reuse some of the existing software
- The prototype has performed well during various exercises and tests
- We are looking forward to testing it with individual users