Evaluating distributed EOS installation in Russian Academic Cloud for LHC experiments

A.Kiryanov[1], A.Klimentov[2], A.Zarochentsev[3].

1. Petersburg Nuclear Physics Institute - PNPI
2. National Research Center “Kurchatov Institute” - NRC KI
3. Saint-Petersburg State University - SPbSU

This work is supported by RFBR grant 15-29-07942

Motivation

1) There’s an increasing demand in storage resources for LHC experiments, especially in view of HL LHC

2) Operation of a large Tier-2 site is quite complicated and requires unique expertise of attending personnel
3) Smaller Tier-3 sites may help, but it’s not easy to centrally coordinate their activity

Our solution to aforementioned problems is a federation of small sites that looks like a large site (“cloud”) from outside

We evaluate federated storage as a first step towards this idea

Requirements for a federated storage

Single entry point

Universality: should be usable by at least four major LHC experiments

Scalability: it should be easy to add new resources

Data transfer optimality: transfers should be routed directly to the disk servers avoiding intermediate gateways and other bottlenecks

Fault tolerance: replication/redundancy of core components

Built-in virtual namespace, no dependency on external catalogues

Finding possible solutions

We had to find a software solution that supports federation of distributed storage resources. This very much depends on a transfer protocol support for redirection. Two protocols that are good at it are xroot and HTTP.

HTTP-based federation is implemented in DynaFed software developed by IT/SDC group at CERN. This software is highly modular and only provides a federation frontend while the storage backend(s) have to be chosen separately. It would be interesting to try it out but we were looking for more all-in-one solution.

xroot-based solution is EOS. It’s also developed at CERN (we knew where to ask for help), has characteristics closely matching our requirements, and is already used by all major LHC experiments. We decided to give it a try.

Prototype structure: software and tests

Base OS: SL6/x64

Storage system: EOS Aquamarine

Authentication scheme: GSI

Tests

- Bonnie++ (file I/O test on FUSE-mounted file system)
- ATLAS test: standard ATLAS event reconstruction workflow with Athena (thanks to Dmitrii)
- ALICE test: sequential ROOT event processing (thanks to Peter)

Prototype structure: architecture

Initial plan was to test 3 sites (NRC KI, PNPI, SPbSU) with storage servers + “head” at CERN

Due to various reasons most of the presented tests were done with only two sites: PNPI and SPbSU

NRC KI - MGM (slave/master) + FST + PerfSONAR + UI

PNPI - MGM (slave/master) + FST + PerfSONAR + UI

SPbSU - MGM (slave/master) + FST + PerfSONAR + UI

CERN - MGM (master) + PerfSONAR (?) + UI (lxplus)

(green means it’s already there)

Prototype structure: testing schemes

Proof-of-concept test: install and configure distributed EOS, hook up GSI authentication, test basic functionality (file/directory create/delete, FUSE mount, access permissions)

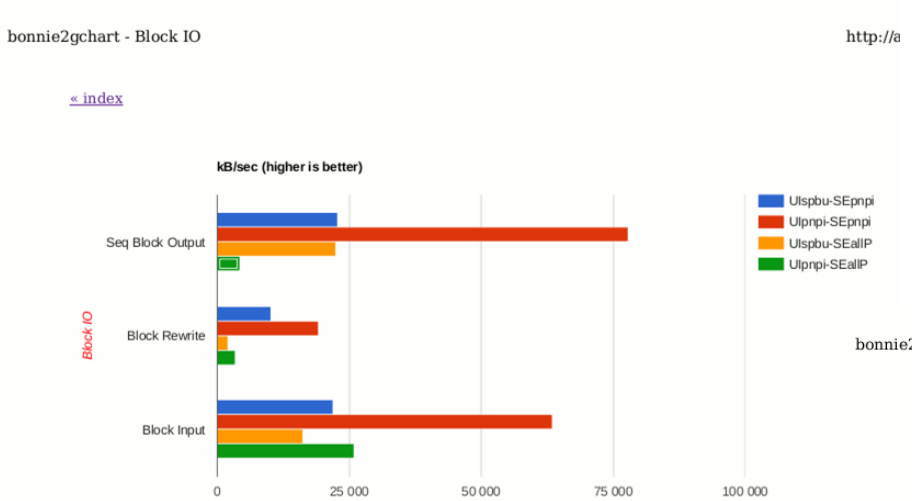
Reliability test: MGM master-slave migration

Performance tests: file and metadata I/O, real-life experiment software, network

Redirection impact test: check if there’s performance degradation with remote “head” node

Data locality test: evaluate EOS geo-tags role in data distribution

Tests: Bonnie++



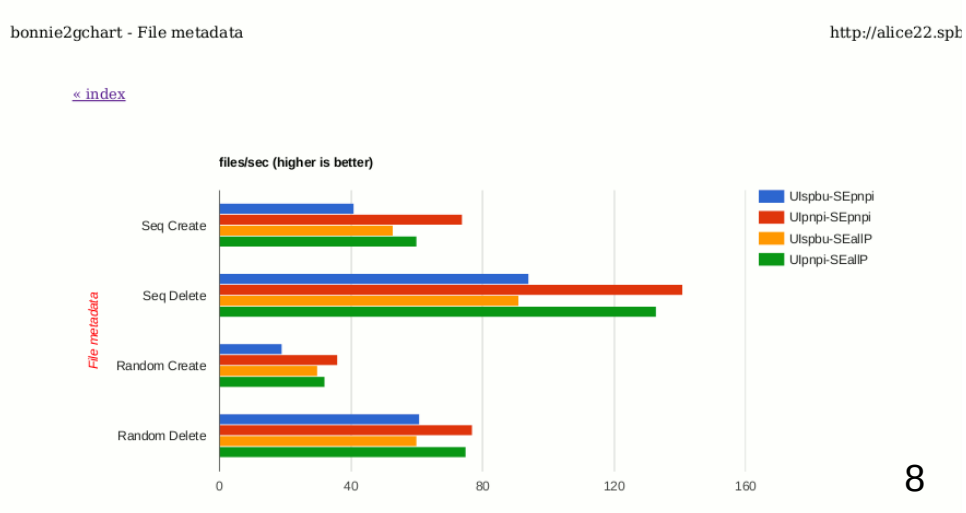
From PerfSonar:

Speed SPbSU - > PNPI 810Mb/s

PNPI->SPbSU 570 Mb/s

Latency SPbSU - > PNPI 0.9 ms

PNPI->SPbSU 3.6 ms

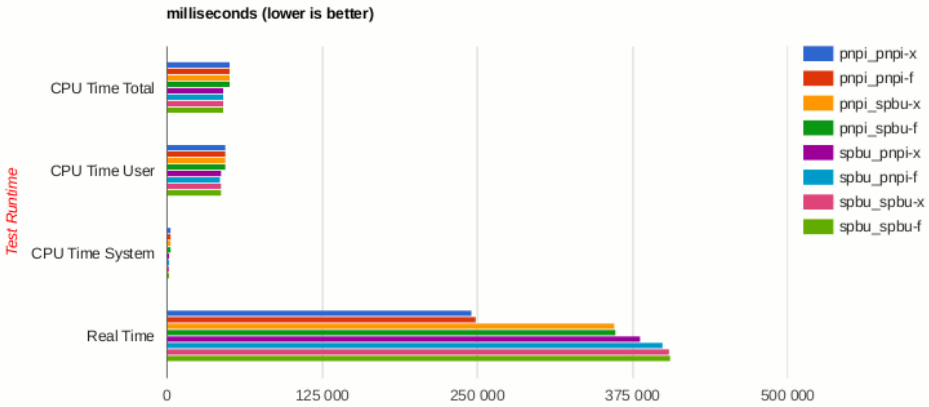


ATLAS

bonnie2gchart - Test Runtime

http://alice22.spbu.ru/test_suite0/?t=test-results

« index

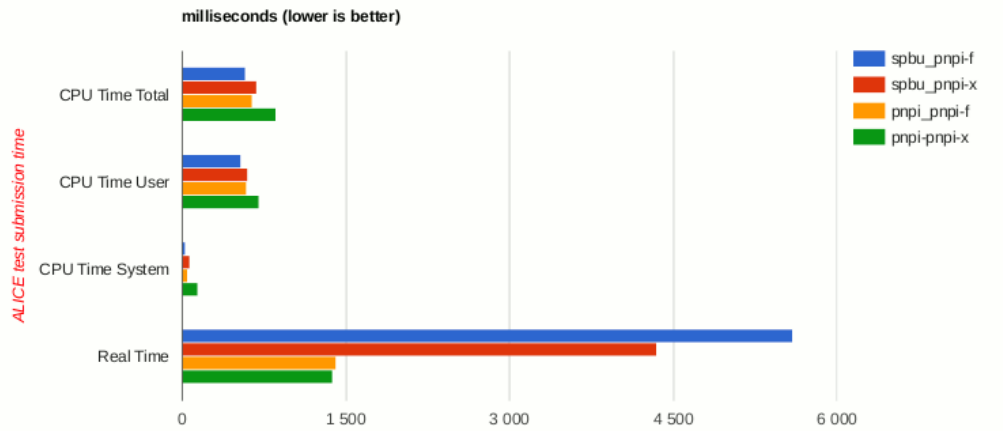


ALICE

bonnie2gchart - ALICE test submission time

http://alice22.spbu.ru/alice_test/?t=time

« index



Conclusions

Successful proof of concept

Redirection impact is relatively low (definitely not a show-stopper)

There’s a visible but not huge difference between direct xroot and FUSE access, however only with some specific software (ROOT) probably optimized for direct xroot access

Geo-tags and data locality tests and optimizations are work in progress

There are some roughs in FUSE handling and MGM master-slave migration, we hope for further improvements in EOS concerning these areas

Acknowledgements

Peter Hristov for ALICE tests

Dmitrii Krasnopevtsev for ATLAS tests

Eygene Ryabinkin for support on NRC KI side

THANKS