



# Russian Federated Data Storage System Prototype

Andrey Kiryanov, Alexei Klimentov, Andrey Zarochentsev

on behalf of BigData lab @ NRC "KI" and  
Russian Federated Data Storage Project



# Russian federated data storage project

In the fall of 2015 the "Big Data Technologies for Mega-Science Class Projects" laboratory at NRC "KI" has received a Russian National Science Foundation grant to evaluate federated disk storage technologies.

This work has been started with creation of a storage federation for geographically distributed data centers located in Moscow, Dubna, St. Petersburg, and Gatchina (all are members of Russian Data Intensive Grid and WLCG).

This project aims at providing a usable and homogeneous service with low requirements for manpower and resource level at sites for both LHC and non-LHC experiments.



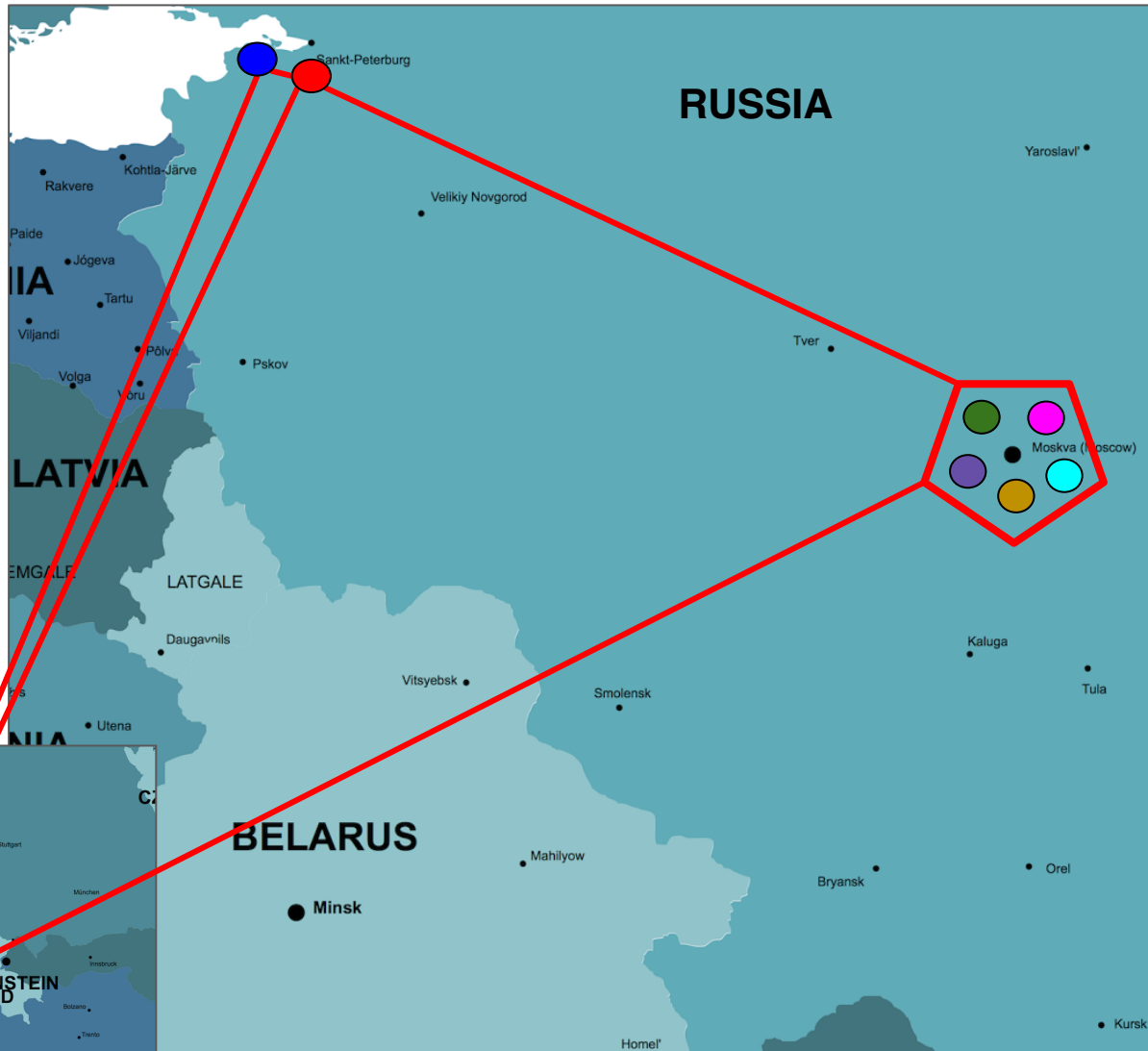
# Federated Storage Basics

1. Single entry point;
2. Scalability and integrity: it should be easy to add new resources;
3. Data transfer and logistics optimisation: transfers should be routed directly to the closest disk servers avoiding intermediate gateways and other bottlenecks;
4. Stability and fault tolerance: redundancy of core components;
5. Built-in virtual namespace, no dependency on external catalogues;
6. Strong security and mutual authentication of all components.

EOS has been designed to satisfy all of these requirements.



# Federation topology



## SPb Region

- SPbSU
- PNPI

## Moscow Region

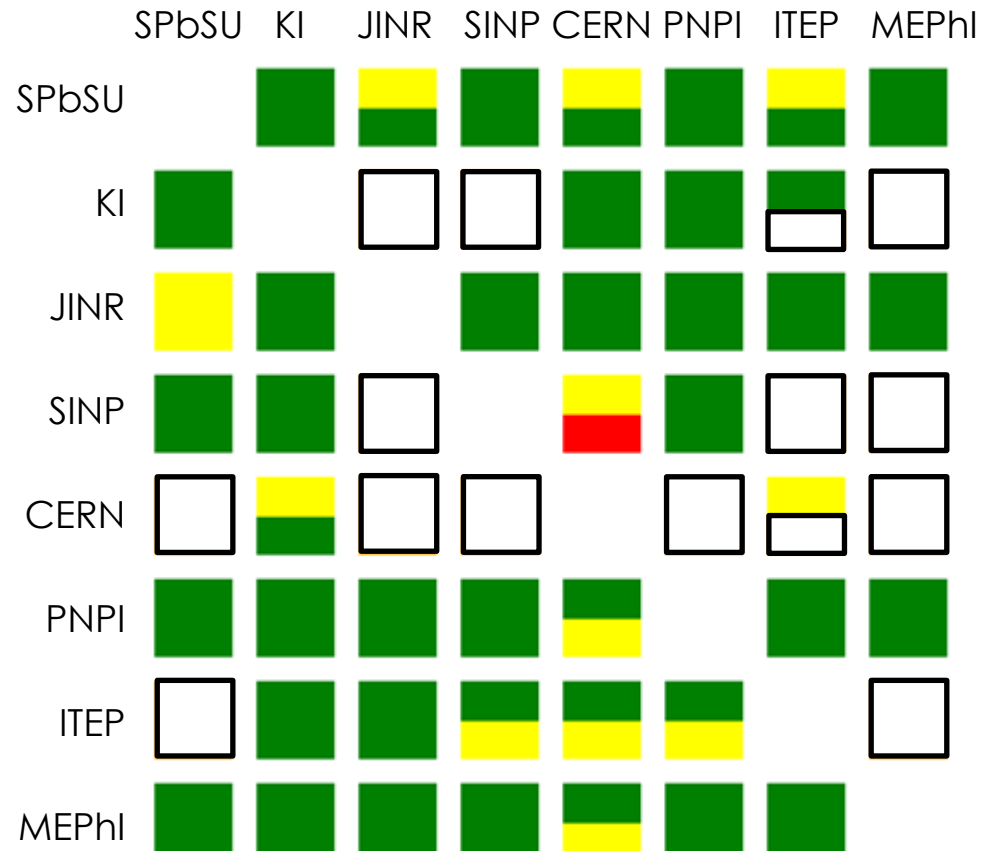
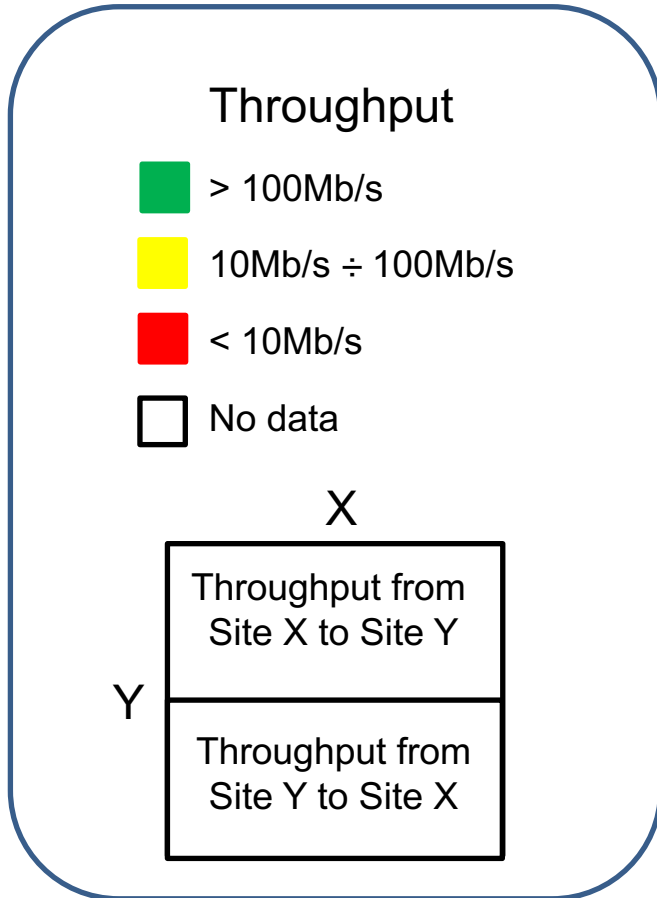
- JINR
- NRC "KI"
- MEPhI
- SINP
- ITEP

and

- CERN



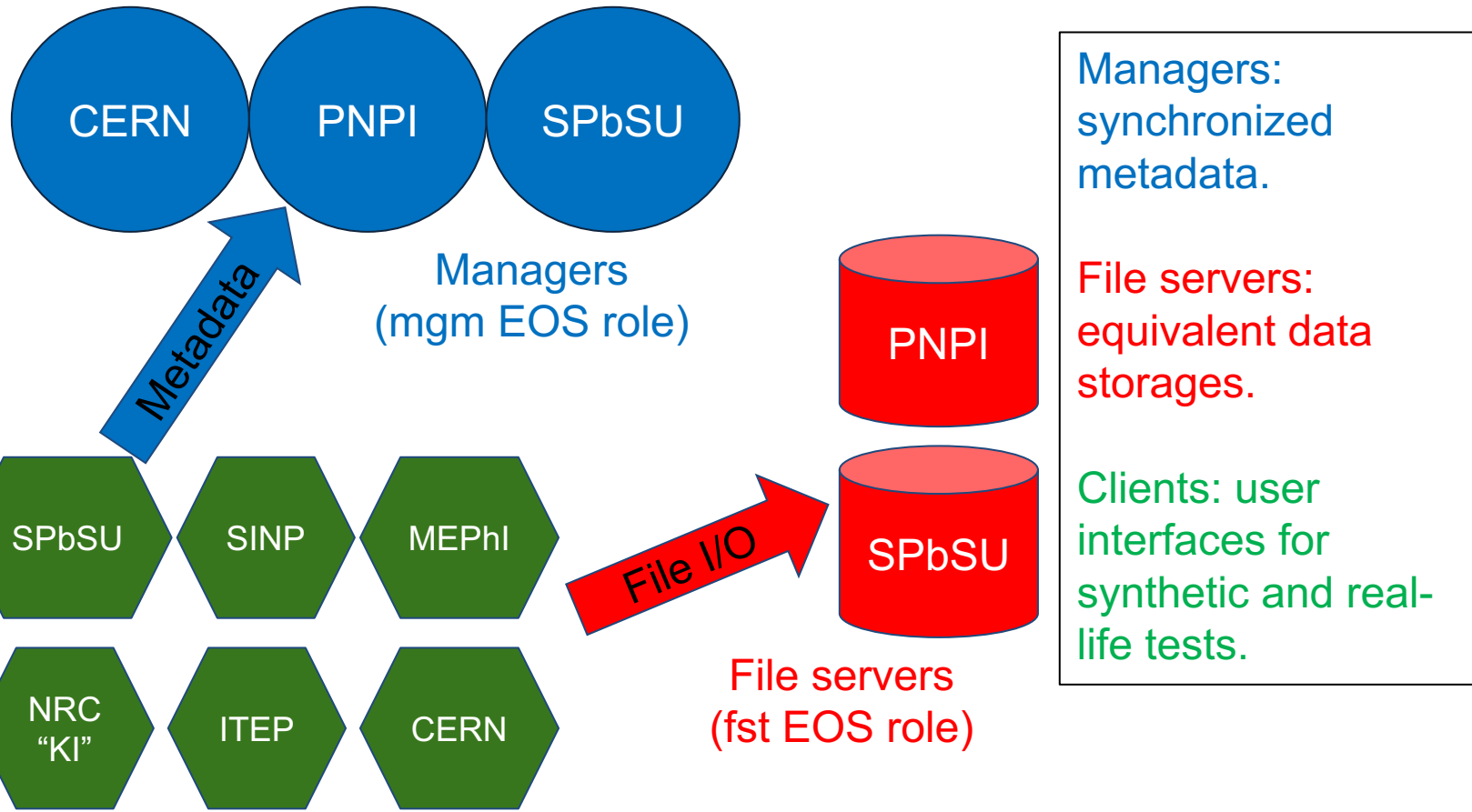
# Inter-federation network performance





# Initial testbed

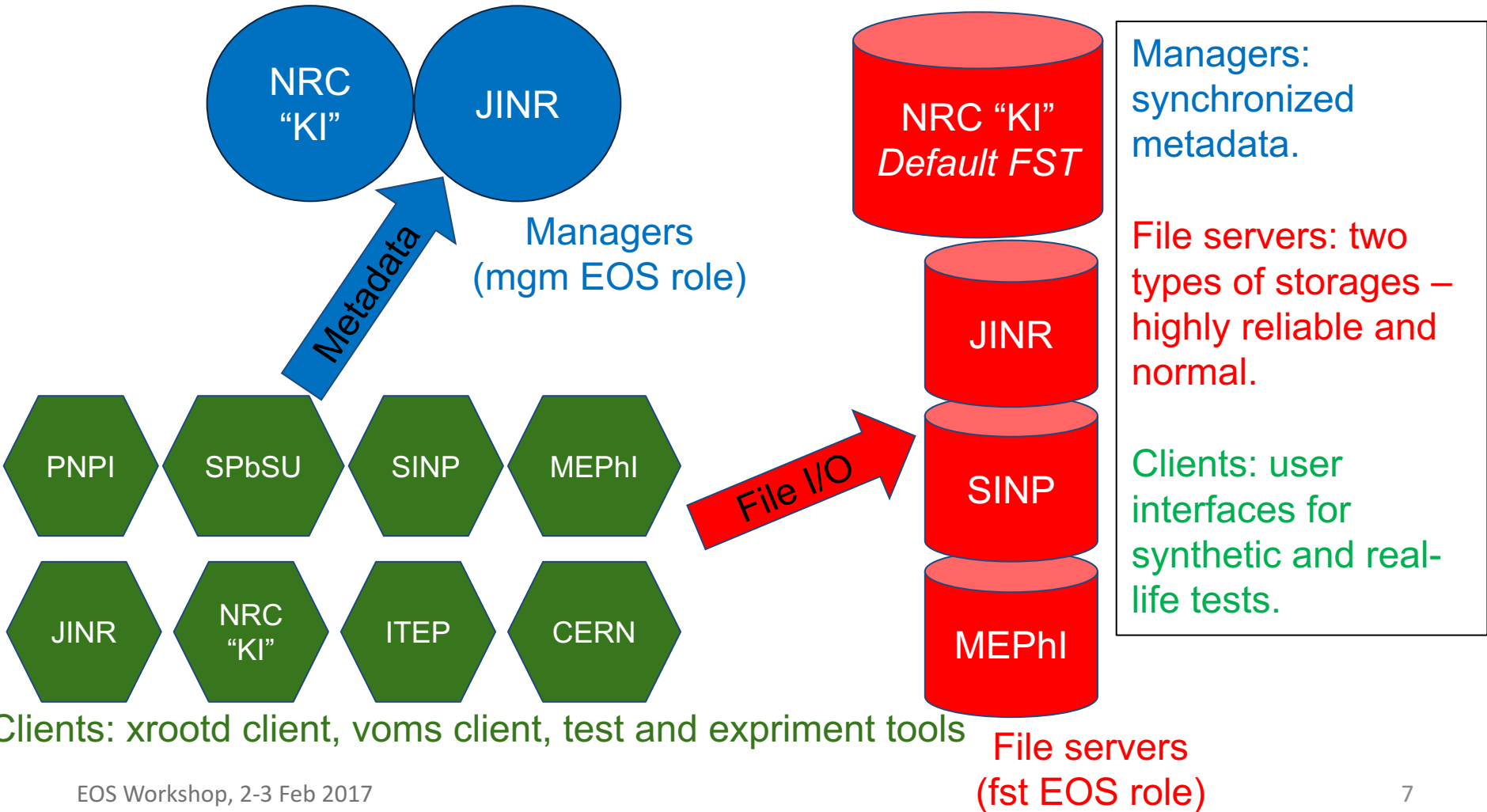
proof of concept tests and optimal settings evaluation



Clients: xrootd client, voms client, test and experiment tools



# Extended testbed for full-scale testing





# Test goals, methodology and tools

## Goals:

Set up a distributed storage and verify basic properties:

- Data access
- Reliability
- Data replication

## Tools:

*Synthetic tests:*

- Bonnie++: file and metadata I/O test for mounted file systems (FUSE)
- xrdstress: EOS-bundled file I/O stress test via xrootd protocol

*Experiment-specific tests:*

- ATLAS test: standard ATLAS TRT reconstruction workflow with Athena
- ALICE test: sequential ROOT event processing

*Network monitoring:*

- Perfsonar: a widely-deployed and recognized tool for network performance measurements

## Software components:

Base OS: CentOS 6, 7 (64bit)

Storage system: EOS Aquamarine, Citrine

Access protocol: xrootd

Authentication scheme: GSI

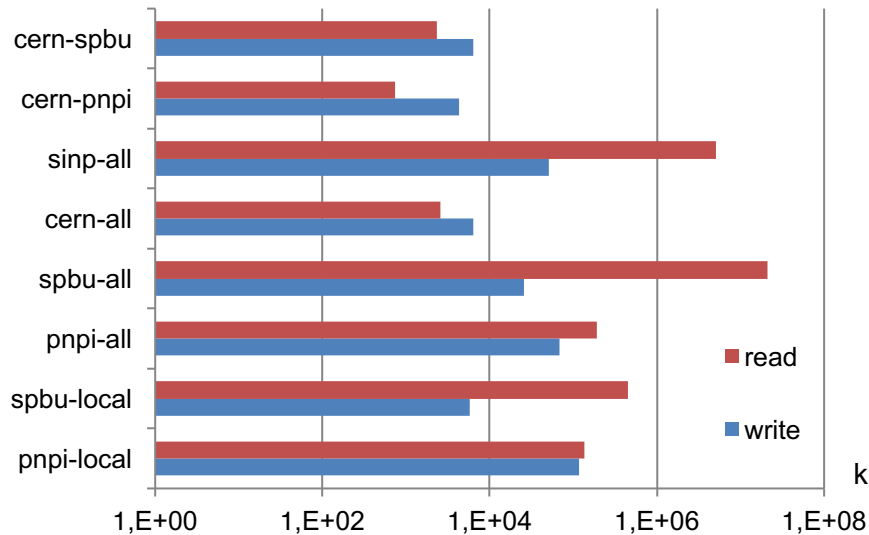
Network monitoring: perfSONAR





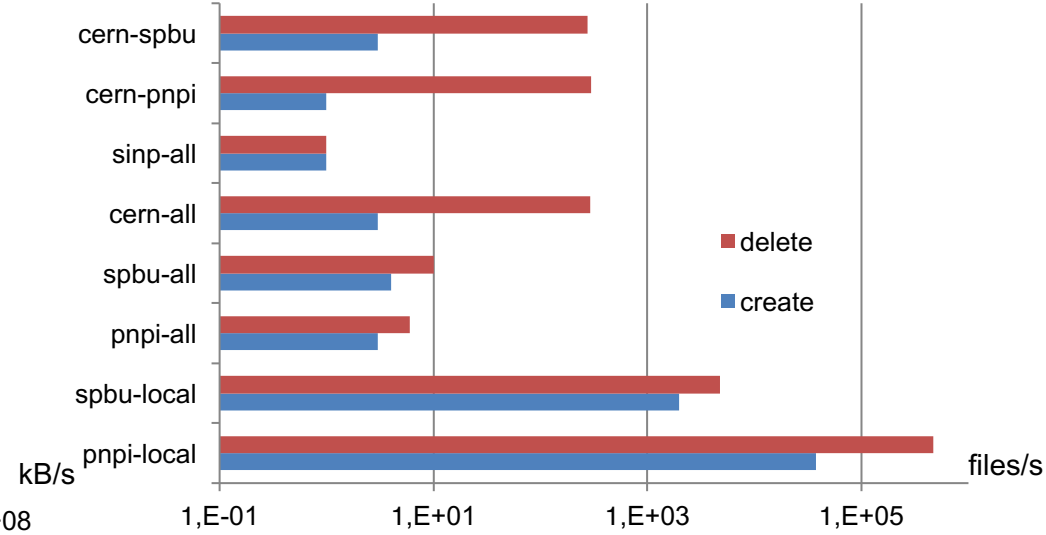
## Bonnie++ test for initial testbed: MGM at CERN, FSTs at SPbSU and PNPI

### Data read-write



pnpi-local – local test on PNPI FST  
 spbu-local – local test on SPbSU FST  
 pnpi-all – UI at PNPI, MGM at CERN, Federated FST  
 spbu-all – UI at SPbSU, MGM at CERN, Federated FST

### Metadata create-delete



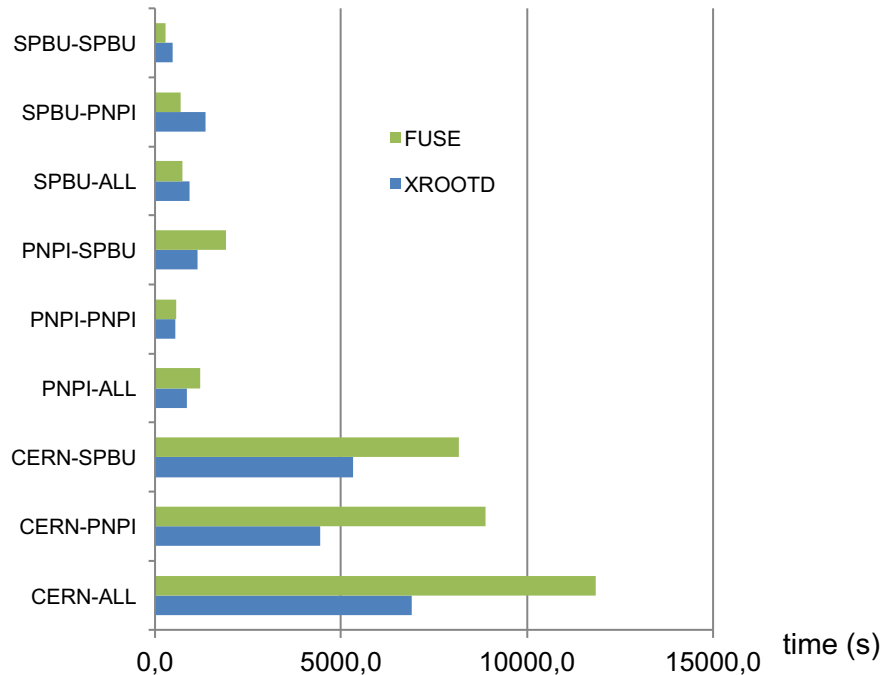
cern-all – UI at CERN, MGM at CERN, Federated FST  
 sinp-all – UI at SINP, MGM at CERN, Federated FST  
 cern-pnpi – UI at CERN, MGM at CERN, FST at PNPI  
 cern-spbu – UI at CERN, MGM at CERN, FST at SPbSU

- metadata I/O performance depends solely on a link between client and manager
- data I/O performance does not depend on a link between client and manager

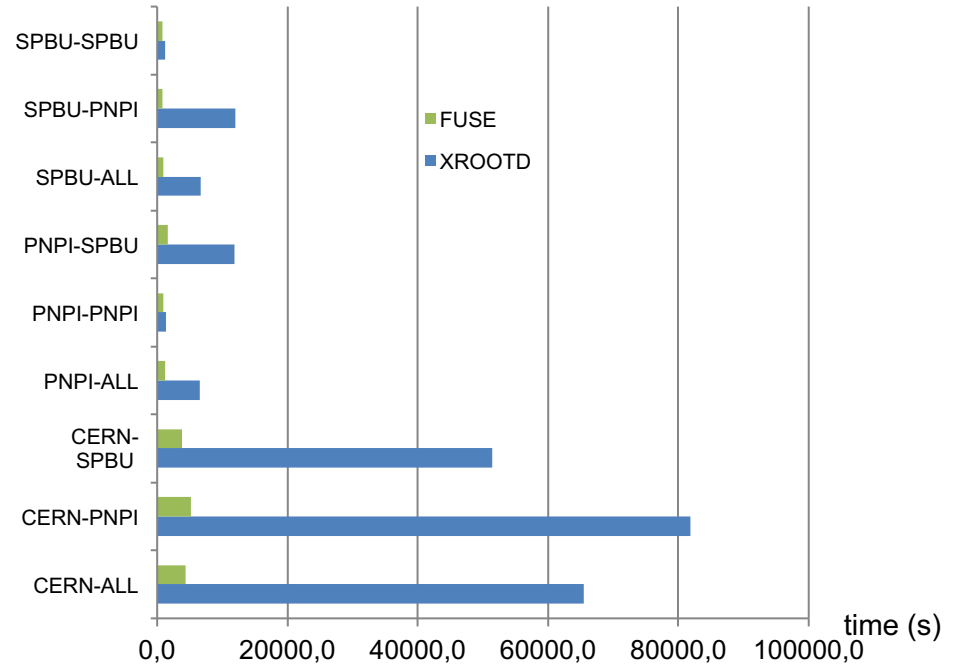


# Experiment-specific tests for two ways of access: native xrootd and locally-mounted file system (FUSE)

ALICE



ATLAS



Experiment's applications are optimized for different protocols (remote vs. local)



# Our First experience with EOS and intermediate conclusion

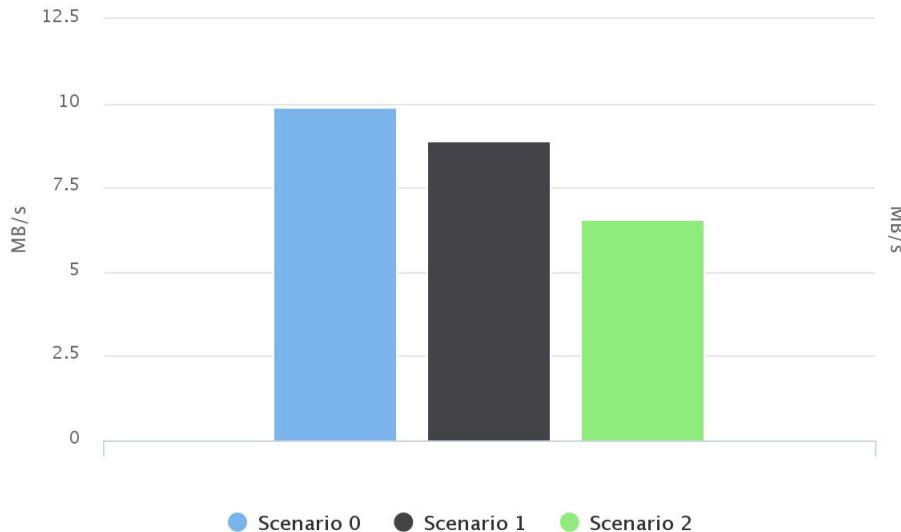
1. Basic stuff works as expected;
2. Some issues were discovered and communicated to developers;
3. Metadata I/O performance depends solely on a link between client and manager while data I/O performance does not depend on a link between client and manager;
4. Experiment-specific tests for different data access patterns have contradictory preferences with respect to the way of data access (native xrootd vs. FUSE-mounted filesystem).



# Data placement policy

1. Number of data replicas depends from data type (replication policy has to be defined by experiments / user community);
2. Federated storage may include reliable sites ("T1s") and less reliable sites ("Tns");
3. Taking aforementioned into account we have three data placement scenarios which can be individually configured per dataset:
  - Scenario 0:** Dataset is randomly distributed among several sites;
  - Scenario 1:** Dataset is located as close as possible to the client. If there's no close storage, the default reliable one is used (NRC "KI" in our tests);
  - Scenario 2:** Dataset is located as in scenario 1 with secondary copies as in scenario 0.

All described tests have been performed on extended testbed.



Populate dataset of 21 ROOT files (~30 GB) from CERN with three scenarios. Plot above shows mean write speed per dataset per scenario.

## Data population performance test from CERN

Data population procedure is as follows:

**Scenario 0:** Files are copied to several file servers;

**Scenario 1:** All files are copied to the default file server at NRC "KI", because there's no storage close to CERN in our testbed;

**Scenario 2:** All files are copied to the default file server at NRC "KI" with secondary replicas on several servers.

There's a slight increase in transfer speed with distributed write.  
Replication cost is less than 20%



## ALICE read test

Read procedure is as follows:

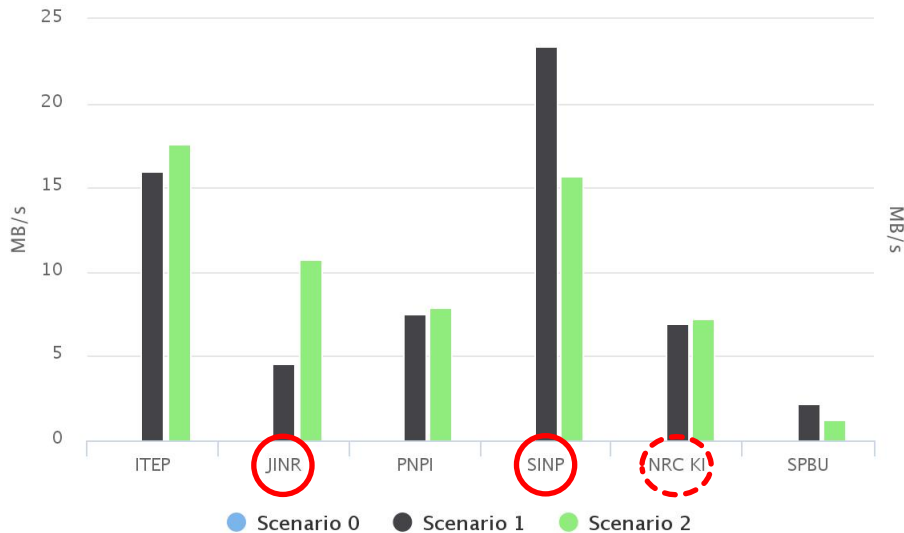
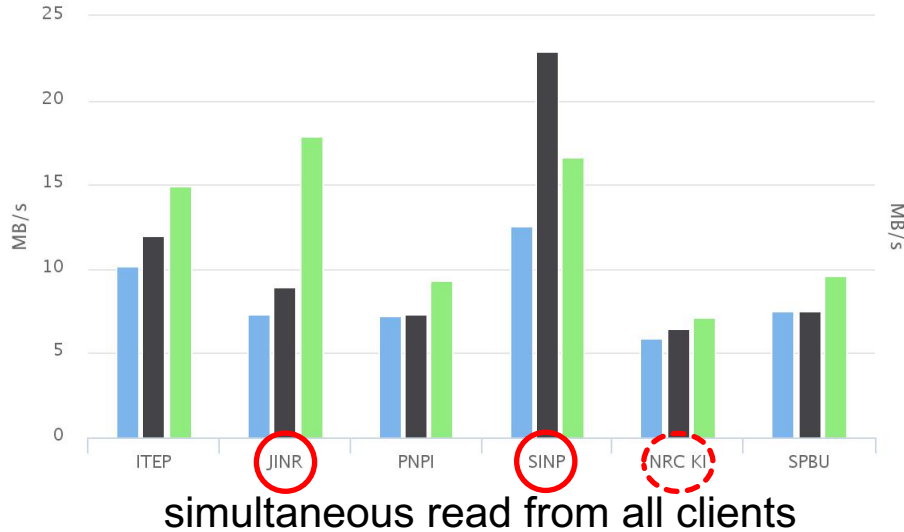
**Scenario 0:** Files are scattered among several file servers;

**Scenario 1:** All files are on the default file server at NRC "KI";

**Scenario 2:** All files are on the default file server at NRC "KI" with replicas that may end up on a closest file server.

○ – this client can find dataset on the closest file server.

On both plots clients are shown on X axis.



Impact of a system load is negligible at this scale.  
Logistics optimization only makes sense for sites with a proper infrastructure.



## Synthetic data placement stress test

Stress test procedure is as follows:

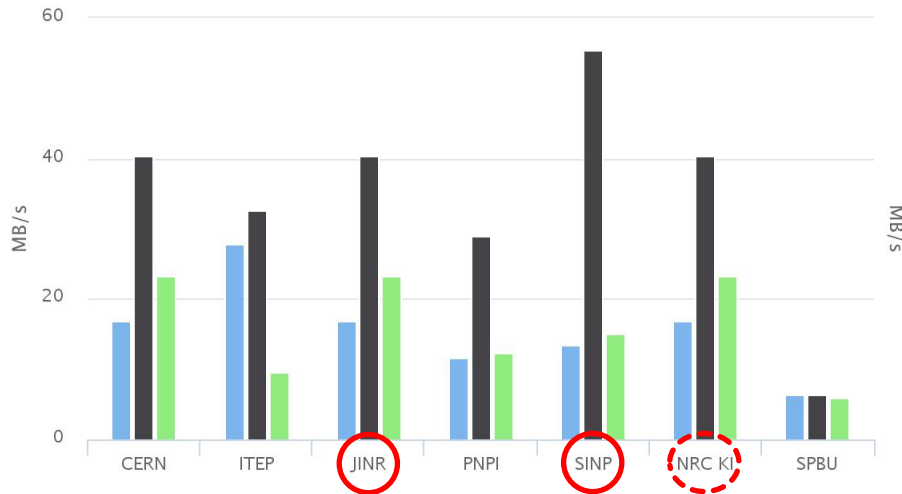
**Scenario 0:** Files are written to and read from random file servers;

**Scenario 1:** Files are written to and read from a closest file server if there is one or the default file server at NRC "KI";

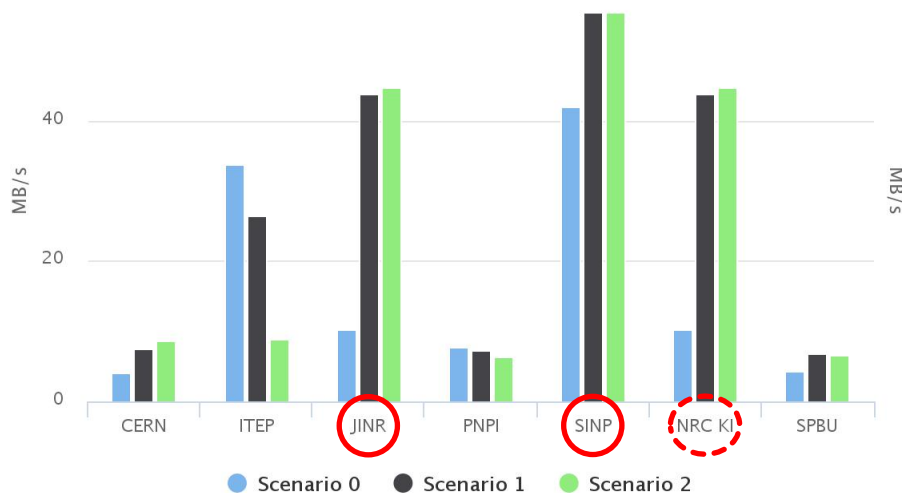
**Scenario 2:** Primary replicas are written as in Scenario 1, secondary replicas as in Scenario 0. Reads are redirected to a closest file server if there is one or to the default file server at NRC "KI".

○ – this client can find dataset on the closest file server.

On both plots clients are shown on X axis.



Xrootd write stress test



Xrootd read stress test

In contrast with the data population test, here distributed write is a bit slower than write to the default storage. At low transfer speed replication cost is almost negligible. With many small files there's almost no difference in transfer speed between close and remote datasets. Network fluctuations have more impact.



# Summary

- We have set up a working prototype of federated storage:
  - Seven Russian WLCG sites organized as one homogeneous storage with single entry point
  - All basic properties of federated storage are respected
- We have conducted an extensive validation of the infrastructure using:
  - Synthetic tests
  - Experiment-specific tests
  - Network monitoring
- We have exploited EOS as our first technological choice and we have enough confidence to say that it behaves well and has all the features we need
- We're looking forward to building an xroot or HTTP storage federation with more than one backend storage technology





# Contribution

- During our project we have contributed to the EOS project by discovering and helping to fix various issues with:
  - Startup scripts for CC7 (eossync)
  - Online MGM migration
  - Multiple geotags
  - Single user FUSE mounts
  - Compatibility of xrdstress with non-EOS xrootd implementations



# Acknowledgements

This talk drew on presentations, discussions, comments and input from many. Thanks to all, including those we've missed.

P. Hristov, D. Krasnopevtes for help with experiment-specific tests.

A. Peters for help with understanding EOS internals.

This work was funded in part by the Russian Ministry of Science and Education under Contract No. 14.Z50.31.0024 and by the Russian Fund of Fundamental Research under contract "15-29-0794\_2 офи\_м"

Authors express appreciation to SPbSU Computing Center, PNPI Computing Center, NRC "KI" Computing Center, JINR Cloud Services, ITEP, SINP and MEPhI for provided resources.



National Research Centre (NRC)  
"Kurchatov Institute"



*Big Data Technologies Laboratory*  
<http://bigdatalab.nrcki.ru/>

# Thank you!