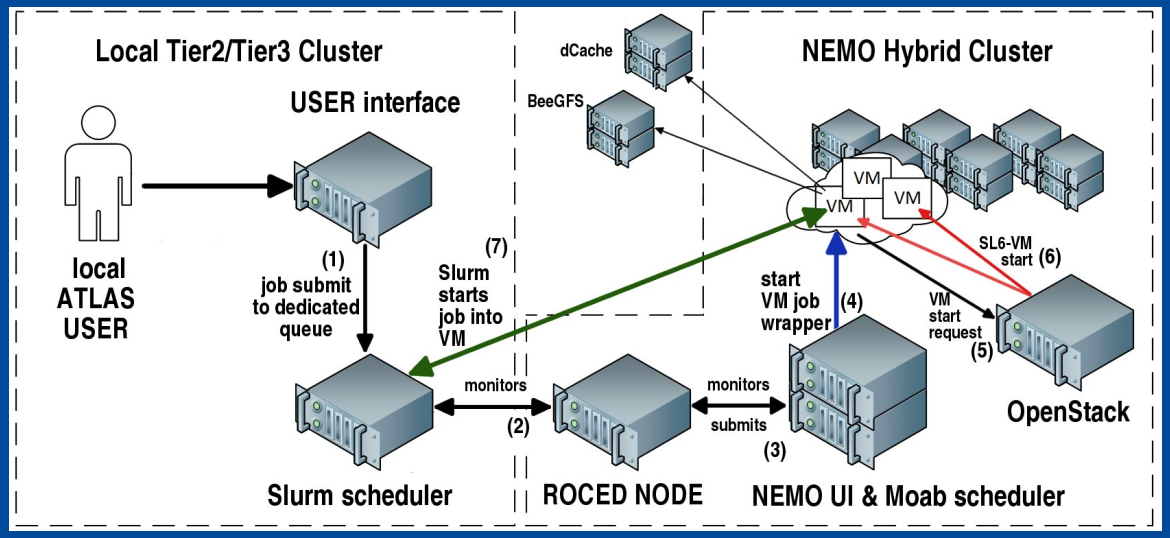
# Plans and Status in Freiburg




Entwicklung und Optimierung der Nutzung heterogener Rechenressourcen

Albert-Ludwigs-Universität Freiburg


Markus Schumacher

Physikalisches Institut
Albert-Ludwigs-
Universität Freiburg


Kick-Off-Meeting of Research Compound
„Innovative Digitale Technologien für ErUM“
Munich, 21./22. February 2019

UNI FREIBURG

# Planned contributions

UNI FREIBURG

**Work area A: Development for provisioning of technologies for the usage of hetereogenous ressources**

WP1: Tools for Integration of heterogenous ressources in scientific computing

WP2: Efficient use of heteregogenous ressources

WP3: Identification & steering of workflows on hetereogenous ressources

**Work area B: Application and test of virtualised software components in the environment of heterogenous compute ressoruces**

WP2: Managment of jobs and ressources

WP4: Combined tests

**Positions not filled yet. 2nd round of annoucement of open positions ongoing**

# NEMO Hybrid Cluster at Freiburg


In operation since August 2016

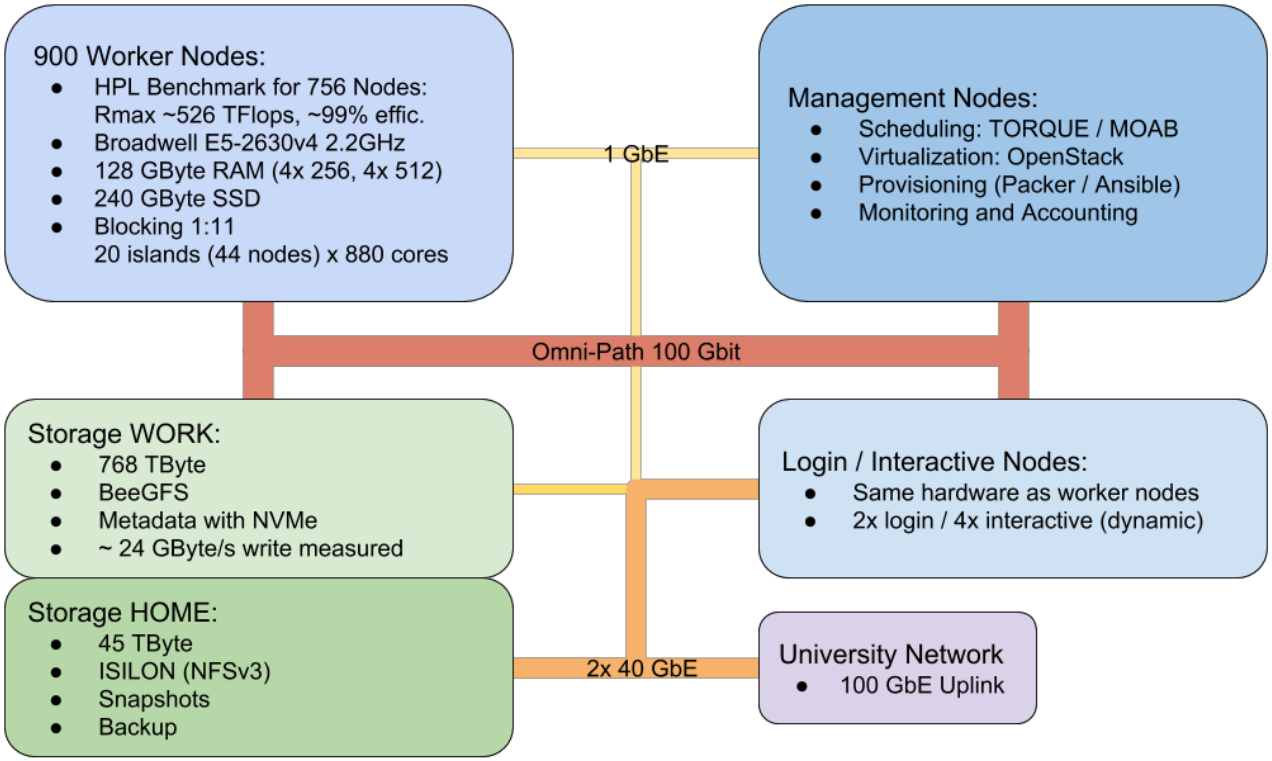
Serving communities of **N**euro Science, **E**lementary Particle Physics, and **M**icro**s**ystem Engineering in state of Baden-Württemberg

## Compute:

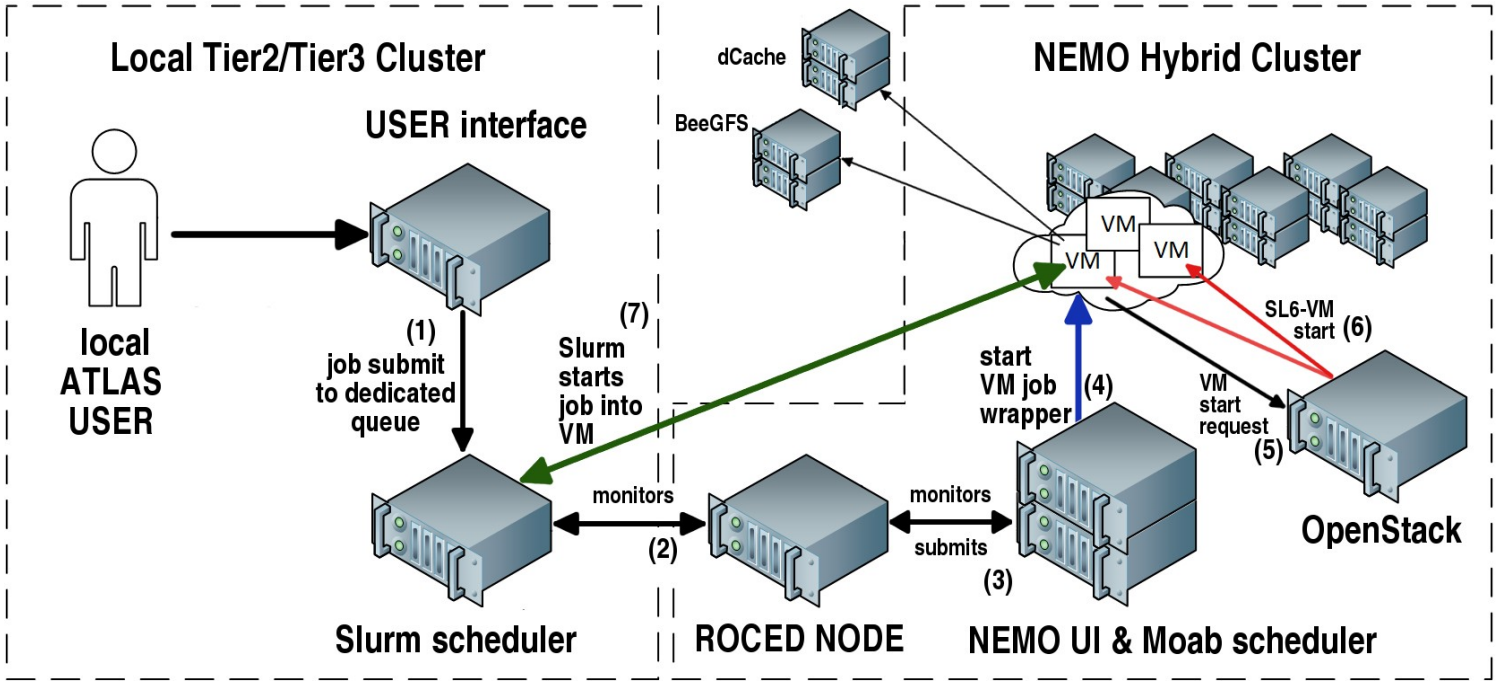
900 nodes
with 20 CPUs,
(Intel Xeon E5-2630)
128 GB RAM per node
Omni Path 100Gb/S

## Storage:

786 TB BEEGFS



# Virtualisation on NEMO




Successfully operated as virtualized Tier3 for CMS groups at KIT and ATLAS groups in FR

https://arxiv.org/abs/1812.11044

For ATLAS use case:

- frontend: scheduler=SLURM OS=SL6 backend: scheduler=MOAB OS= CentOS7
- cloud meta-scheduler: ROCED (Responsive On-Demand Cloud Enabled Deployment) developed at KIT adapted for ATLAS use case
- currently: specific queue for sending jobs to NEMO

# Topic Area A: Work Package 1

UNI FREIBURG

## WP1: Tools for Integration of heterogeneous resources in scientific computing

- development of scheduling of cloud-jobs with virtual machines
  - adaption of meta scheduler for "unified queue"
    (meta scheduler "ROCED" (or successor COBalD/TARDIS) developed in G.Quast' group)
    to SLURM/ MOAB frontend/backend combination
  - extension of meta scheduler to other backend-systems and Cloud-APIs
  - extension of meta scheduler to provide different configurations
    of VMs depending on job requirements and available resources

- development of container solutions
  - inclusion of container solution in workflow e.g. based on "Singularity"
  - develop interface btw. front- and backend systems respecting limited rights in container
  - develop monitoring for/at interface between batch systems

# Topic Area A: Work Packages 2 and 3

UNI FREIBURG

## WP2: Efficient use of heterogeneous resources

- contribute to development of fast „on the fly“ data caches
  - adapt prototype based on XRootD (developed by groups of G. Quast and K. Schwarz ) for ATLAS use case
  - implement Dynafed ansatz and prototype and check scalability
  - develop benchmarks and compare different approaches

## WP3: Identification & steering of workflows on heterogeneous resources

- development of monitoring and accounting tools for different configurations, available number and kind of resources, kind of jobs (1st for ATLAS use case)
  - development of standardized interfaces for automised tests (experiment overarching, for monitoring and benchmarking)
  - development and test of benchmarks (for I/O load, cpu load, simulation, user job, ..)
  - evaluate performance, investigate long-term efficiency and reliability
  - analysis and storage of monitoring data via Elastic Search and Kibana

# Topic Area B: Work Packages 2 and 4



## WP2: Management of jobs and resources

- optimise parameters for job orchestration for different combinations of heterogeneous resources (WLCG, HPC, clouds, ...; permanent and short time availability) based on monitoring and unified queues developed in WP1 (1st for ATLAS use case)
  - deployment and transfer of VMSs and Containers to CPU nodes
  - handling of VM/Container and Jobs meta data
    also/in particular for resources which are not available as long as expected

## WP4: Combined tests

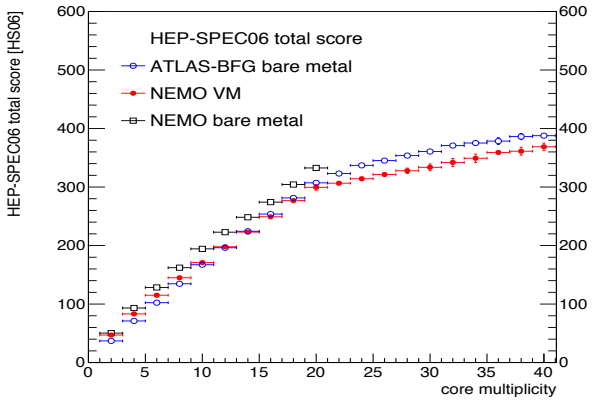
- test of complete workflow based on tools for monitoring, benchmarking and accounting developed in work area A on complex systems
  - evaluation and optimisation of performance
  - investigation of reliability, scalability and maintenance of solutions

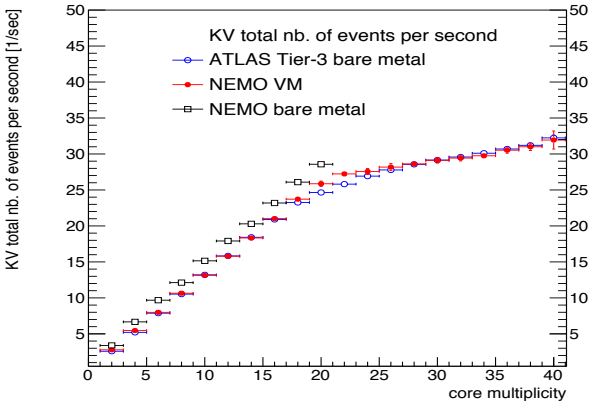
# Benchmarking of VM performance

UNI FREIBURG

- Studies performed by Benoit Roland with support from Frank Berghaus (Victoria), Anton Gamel and Felix Bührer
- Compare fast benchmarks as representative of CPU load for various configurations
  → develop monitoring tool
- Configurations:
  - Freiburg Tier2/3: Bare metal, SL6, Hyper-threading
  - NEMO HPC Cluster: Virtual machine, SL6, Hyper-threading
  - NEMO HPC Cluster: Bare metal, CentOS7, No Hyper-threading
- Architecture: Intel Xeon E5-2630v4 @ 2.2 GHz, 20 cores
- Benchmarks considered (CERN benchmark suite (from HEPIX CPU Benchmarking WG)
  - HEP-SPEC06 (HS06)
  - Whetstone floating-point arithmetic operations (MWIPS)
  - Dirac 2012 : operation on random numbers from Gaussian PDF (HS06)
  - KiT Validation (KV): Geant4 simulation of single muon events in ATLAS detector. More realistic HEP workload (Events per second)
- Results stored in Elastic Search Instance at CERN, Visualised with Kibana Dashboard

# Benchmarking of VM performance (2)

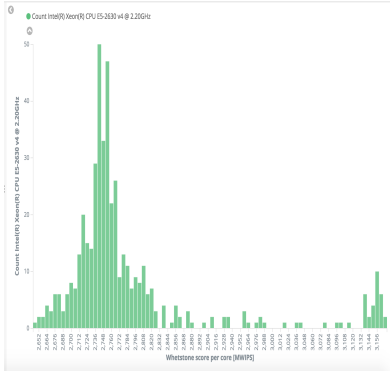


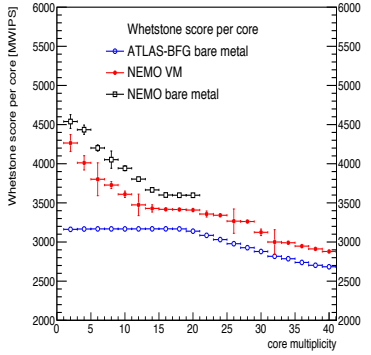




- HEP-SPEC06: performance loss due to virtualisation ≤ 5%
- Kit Validation (KV): no impact from virtualisation observed

Results consistent with data stored from automatic tests in MWT2 Elastic Search instance

- main peak at 2750 MWIPS (40 cores locally)
- small peak at 3150 MWIPS (1-20 cores locally)





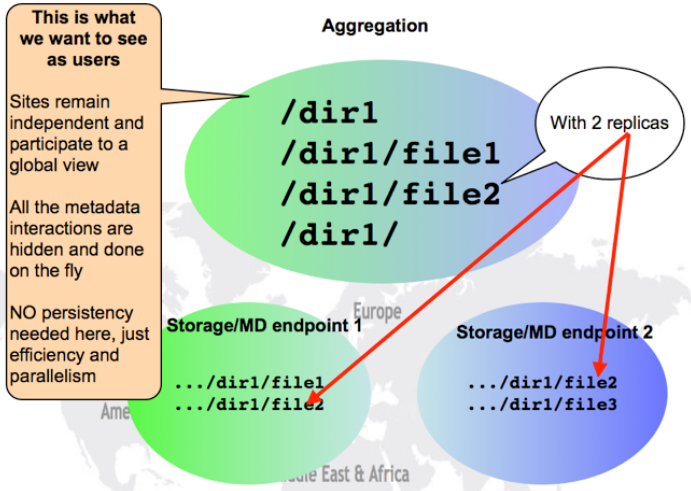

Next step: implement tests (in particular KV benchmark) in Hammercloud framework

# Dynfed@FR and 1st Comparison

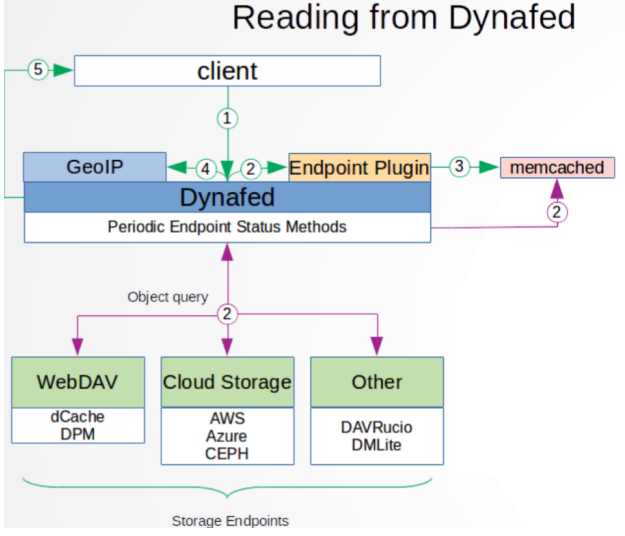
BURG

- Studies performed by Benjamin Rottler with support of Frank Berghaus (Victoria), Anton Gamel amd Felix Bührer

- Dynafed server set set up at Freiburg
- Comparison of reading files with different protocols (XrootD, SRM, WebDAV, Dynafed) from local machines in Freiburg (done) on Grid (to be done)




- Results stored in Elastic search instance at CERN
- Visualisation via Kibana Dashboard (or custom plotting script shown next slide)

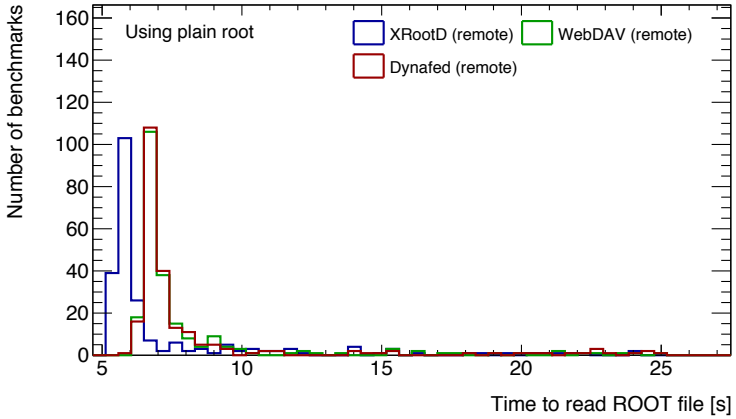


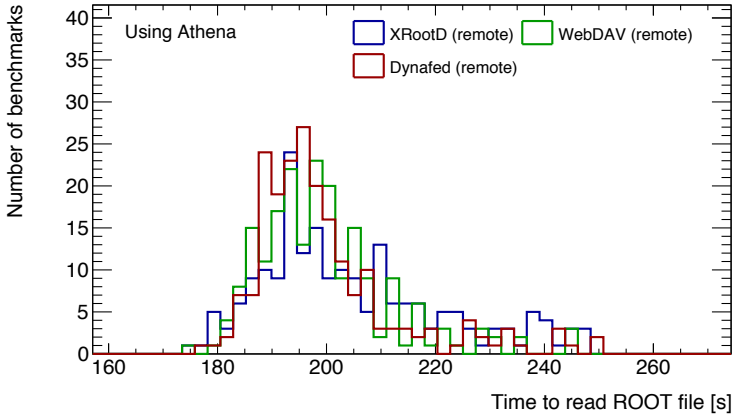

- test with one DxAOD file stored at CERN (or locally) (~ 1,5 Gbyte, ~ 30 000 ttbar event)

# Dynfed@FR and 1st Comparison (2)



First comparison of time to read via different protocols
(file stored at CERN, metadata in Dynafed server at DESY)



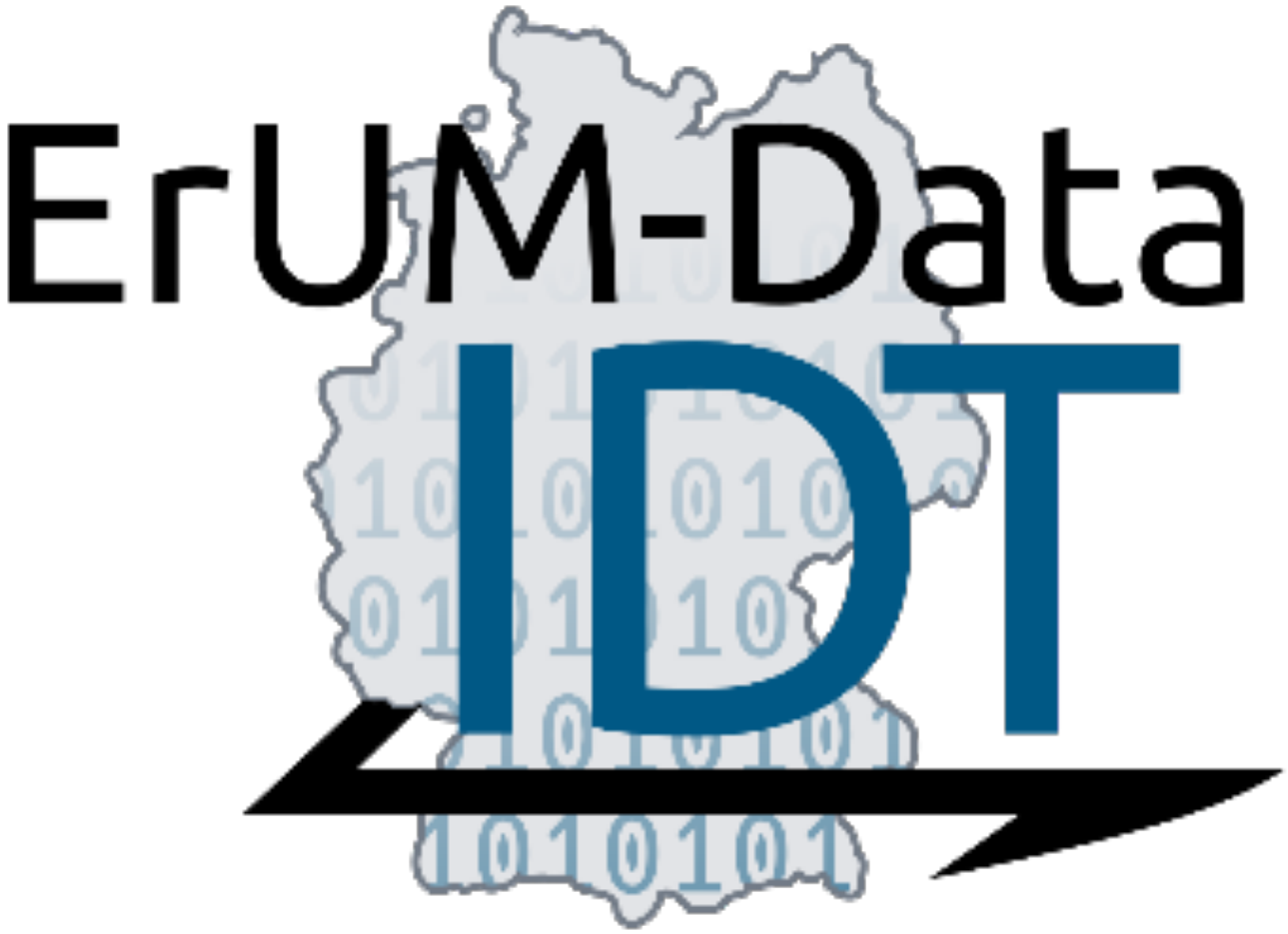




- XRootD faster than WebDAV/Dynafed for plain root (as expected)
- No significant/visible difference when using xAOD mode in ATHENA (loading objects creates overhead)

Next steps:

- compare performance with Grid jobs (needs option to select protocol)
- compare also writing of files with different protocols on different sites
- implement test in Hammercloud framework

# Thanks to Thomas for his great work!




**Looking forward to stimulating and fruitful cooperation!**