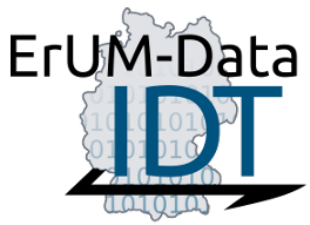
Status report Munich

E. Schanet, G. Duckeck, N. Hartmann, C. A. Mitterer, R. Walker

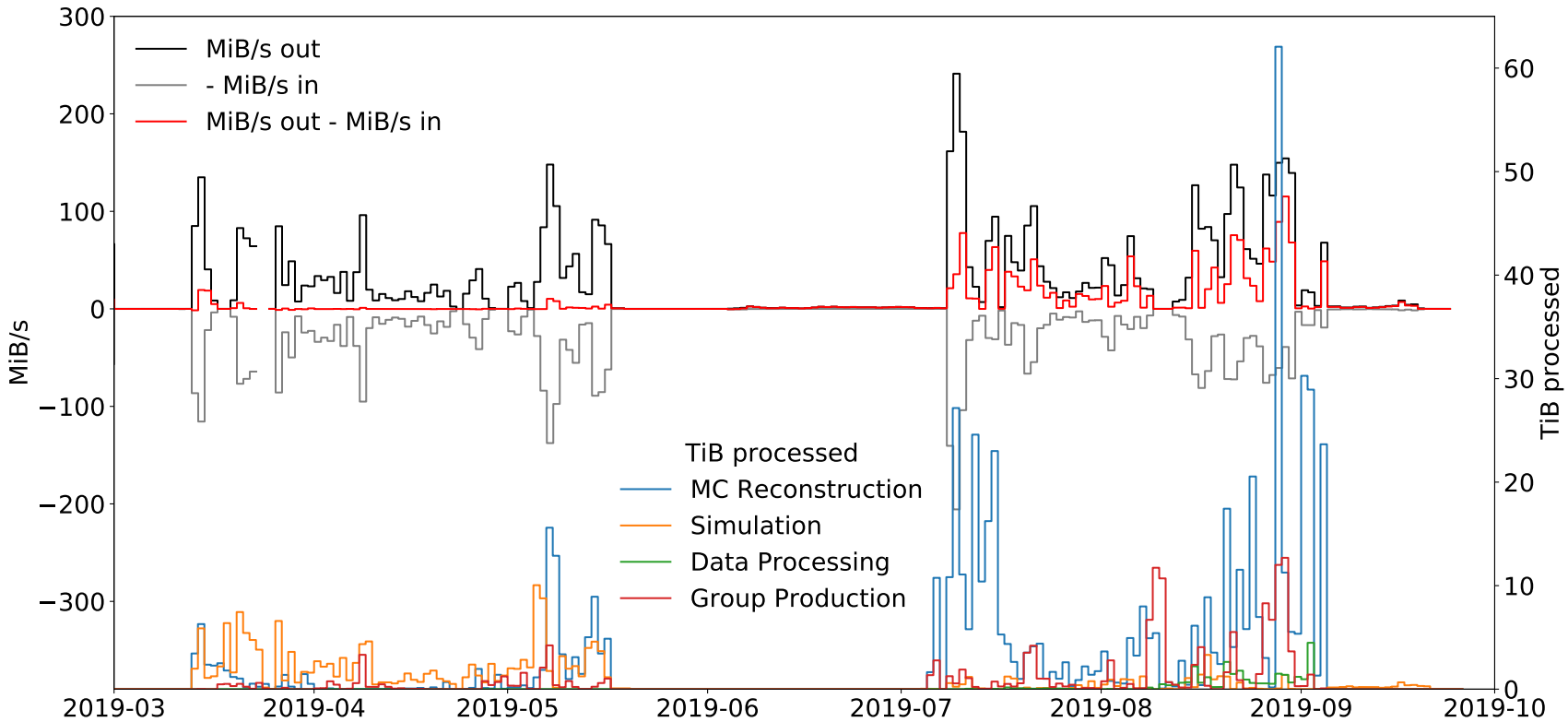
LMU Munich

March 11, 2020



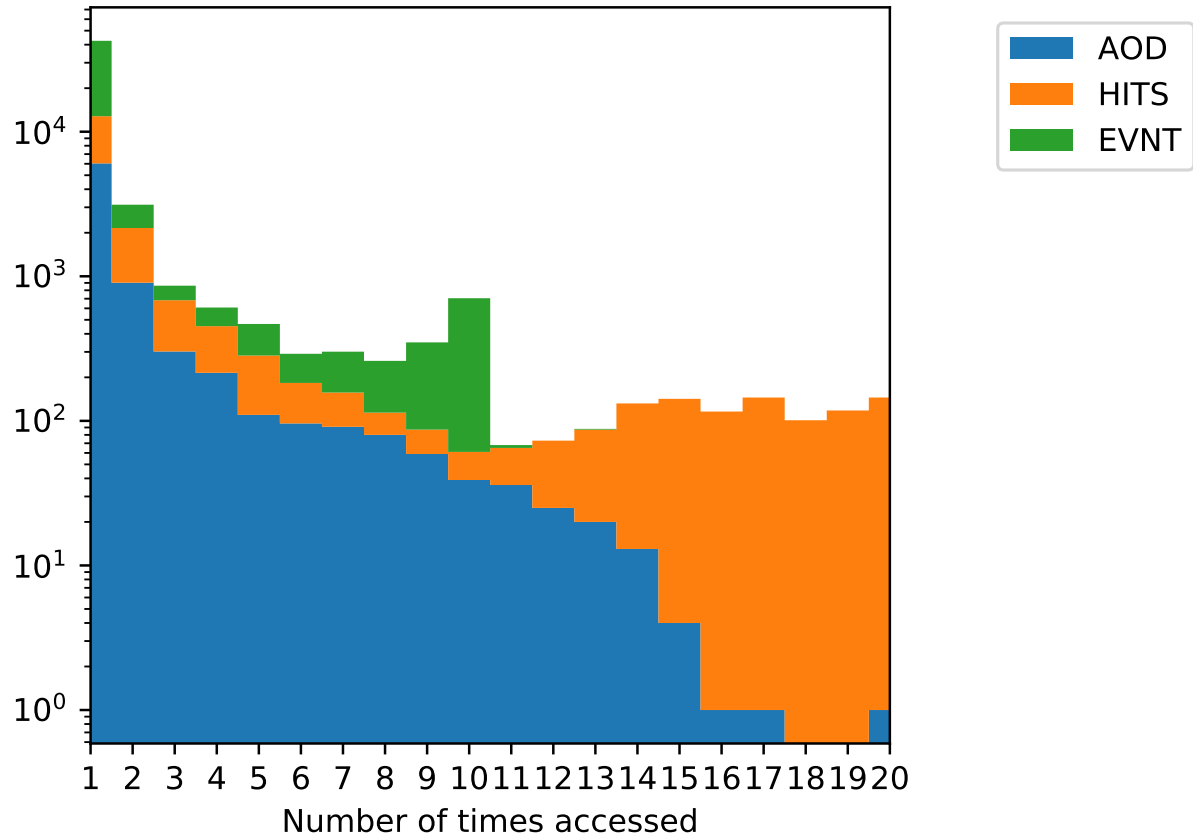
Status on Xcache

Reminder: testing xcache in production queue



→ largest hit rate for MC Reconstruction (here mainly pileup overlay)

Access statistics from cinfo files

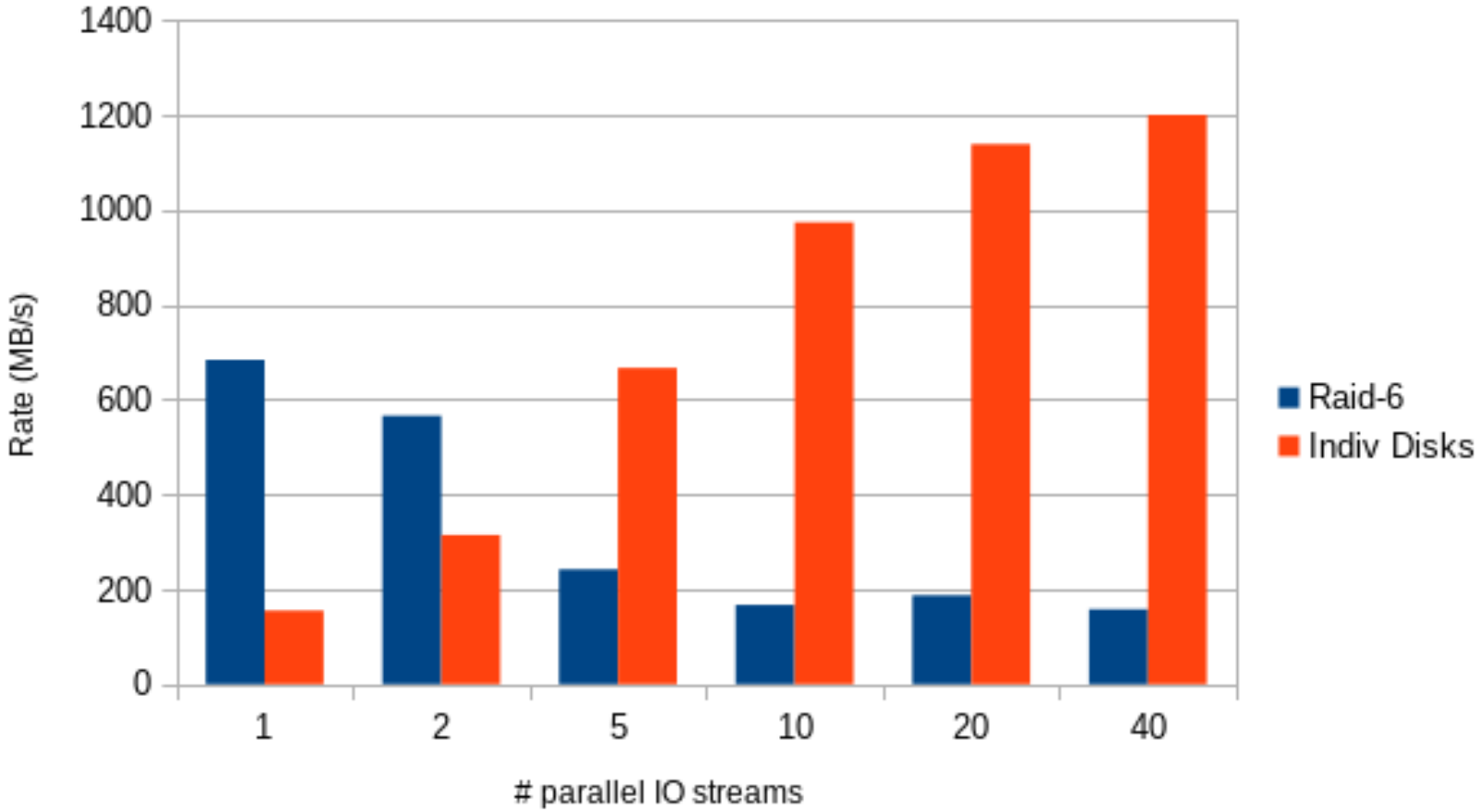


- Most reused files are HITS (pileup)
- EVNT files get reused when one file is processed via multiple jobs
- AOD files get reused for DAOD production (?)

Performance for parallel reads - Raid6 vs single disks

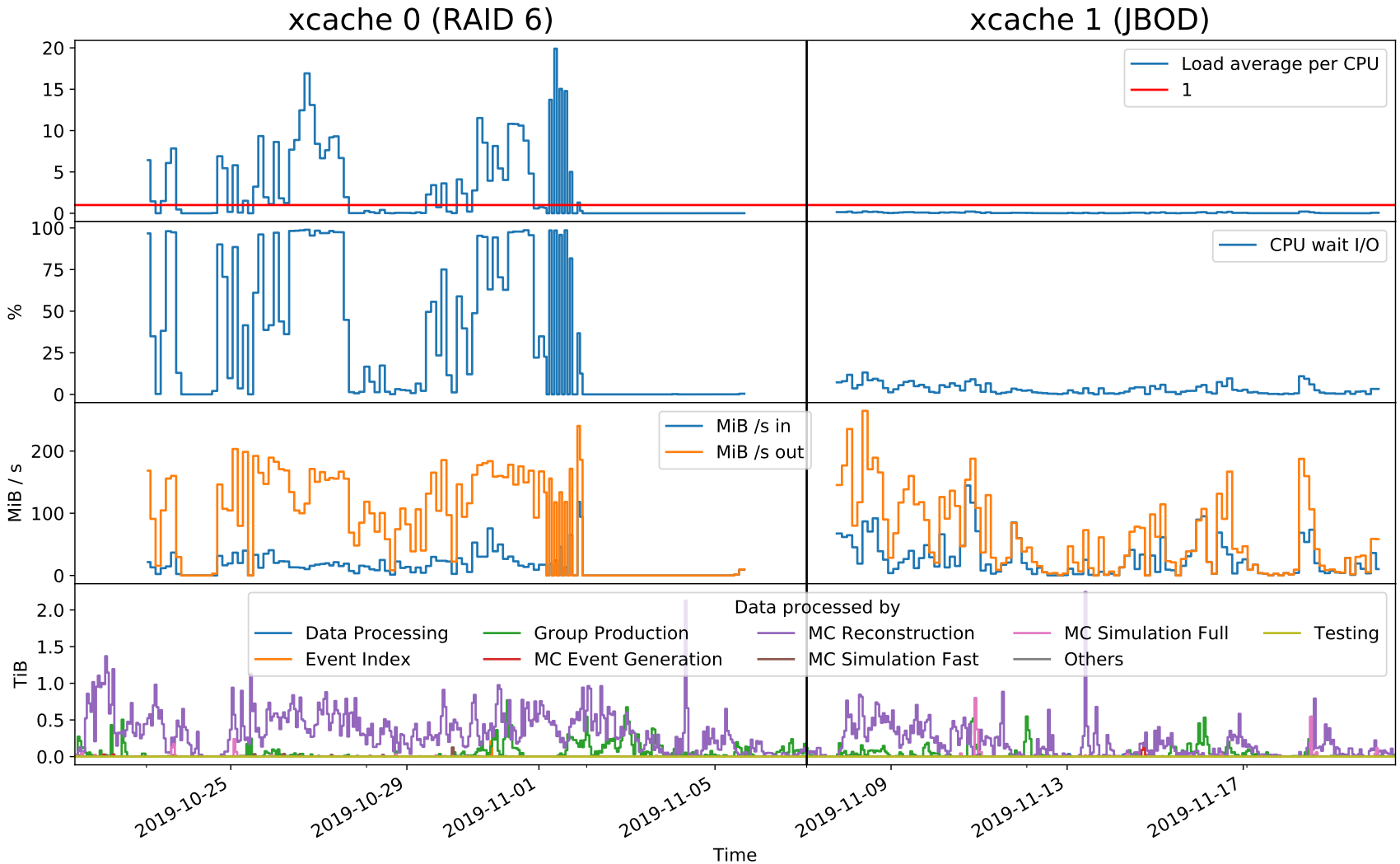
Feedback from xrootd developers: Use multidisk-mode instead of Raid
(see slides from Matevž at XRootD workshop)

Raw reading tests at LRZ:



→ multi-disk mode might perform better than Raid for caching system

Multidisk XCache in ATLAS production queue



→ load and wait CPU drastically reduced for multidisk mode setup!

Xrootd development - checksum tests

Planned to work on checksum test within xrootd:

- currently no verification of checksums for cached files (in terms of checking if the file was received correctly from remote)
- could lead to corrupted files ending up in cache
- Long term plan of xrootd developers: check crc blockwise, receive blockwise checksums from remote together with file
  $\rightarrow$ needs to be implemented by storage systems as well
  $\rightarrow$ advantage: also ensures consistency for partially cached files
  (also see report on DOMA/ACCESS meeting)
- therefore decided not to work on short-term solution for fully cached files within xrootd
  $\rightarrow$ could implement regular checks outside of xrootd instead (compare to checksums from rucio)
  $\rightarrow$ alternative for ROOT files: try to decompress, most corruptions should show up there

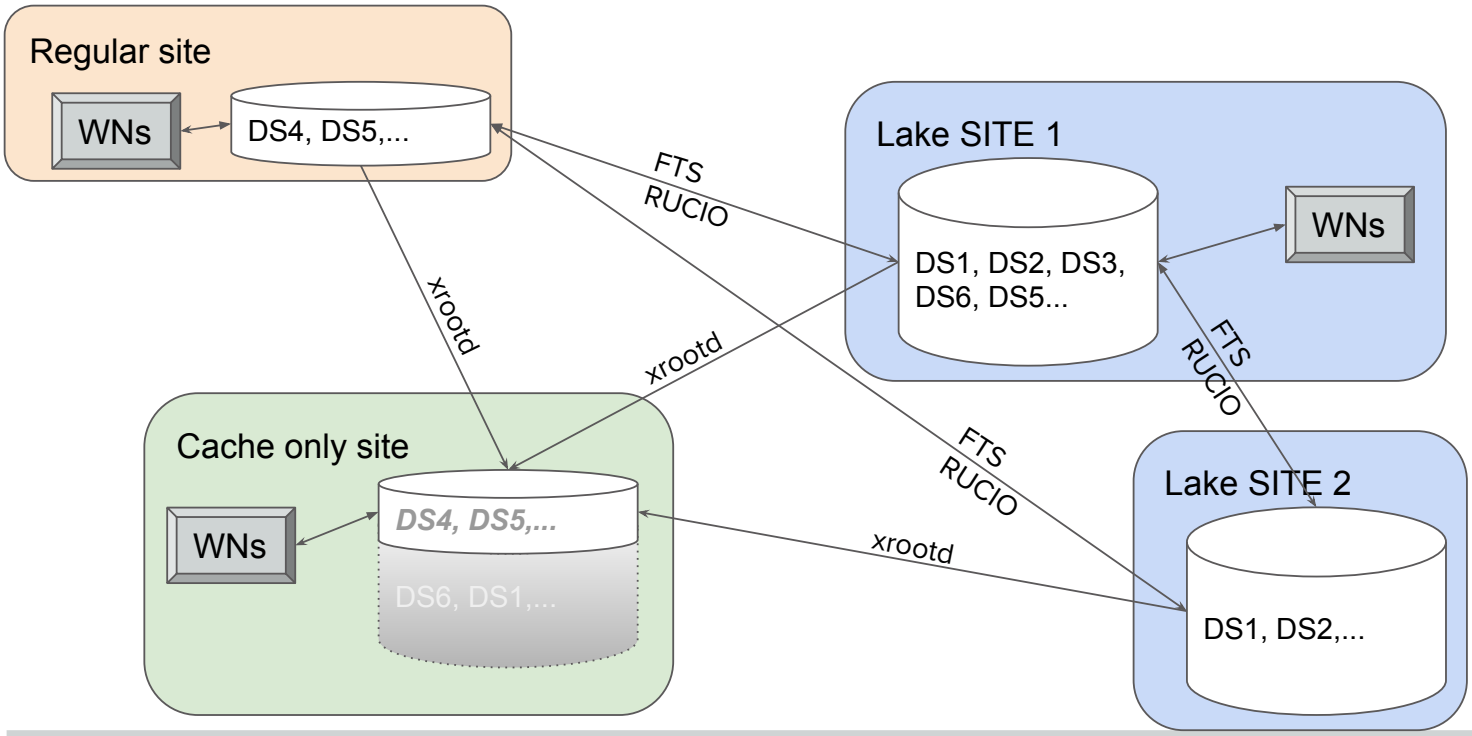
Virtual placement

What is it actually ?

DSX - primary copy

DSX - virtual copy fully or partially cached data

DSX - virtual copy - not there at all until needed



(Slide by Ilija Vukotic)

- "virtually" place datasets to cache-only sites
- expected to ensure high hit rates
- want to test in Munich
- study/simulate hit rate from rucio logs

Collaboration

- DOMA/ACCESS: Contributing to a document that is supposed to become a white paper with recommendations for HL-LHC
- New analysis formats play a role in the discussion (MiniAOD, NanoAOD)
  $\rightarrow$ unclear where caching will play a role
  $\rightarrow$ will smallest formats will be stored on institute disks, will we have "analysis facilities"?
- Different contexts for caching
  - Analysis facilities, very small formats: Caching in addition to storage for fast access on local computing resources (Karlsruhe?)
  - Caching for "diskless" sites (context we studied so far) in grid

Other projects

Queue-based job monitoring

(for details, see slides from meeting in Karlsruhe)

- Set up monitoring system based on ATLAS queue-level information
  $\rightarrow$ provides low latency, high-granularity monitoring
- One application: Suspicious site dashboard
- Investigating whether we can setup a similar system at Belle II
- Also plan to create a dataset to investigate the usage of ML techniques, e.g anomaly detection

Summary

- Successful running of xcache in ATLAS production environment
- Most reused files in current workflow from pileup overlay jobs
- Running XCache with individual disks beneficial (compared to RAID6)
  - significantly reduces load and wait times
  - peak I/O also increased for parallel disk reads/writes

Next plans

- Test virtual placement
- Study caching policies
  $\rightarrow$ can we do better than "least recently used"?
  $\rightarrow$ potentially use ML/reinforcement learning for this (some research on this exists)
  $\rightarrow$ try to simulate with rucio log data
- Combine the 2 xcache servers to a cluster
- Queue-based monitoring (ATLAS):
  - Investigate setup for Belle II
  - Investigate usage of ML (anomaly detection)

Backup

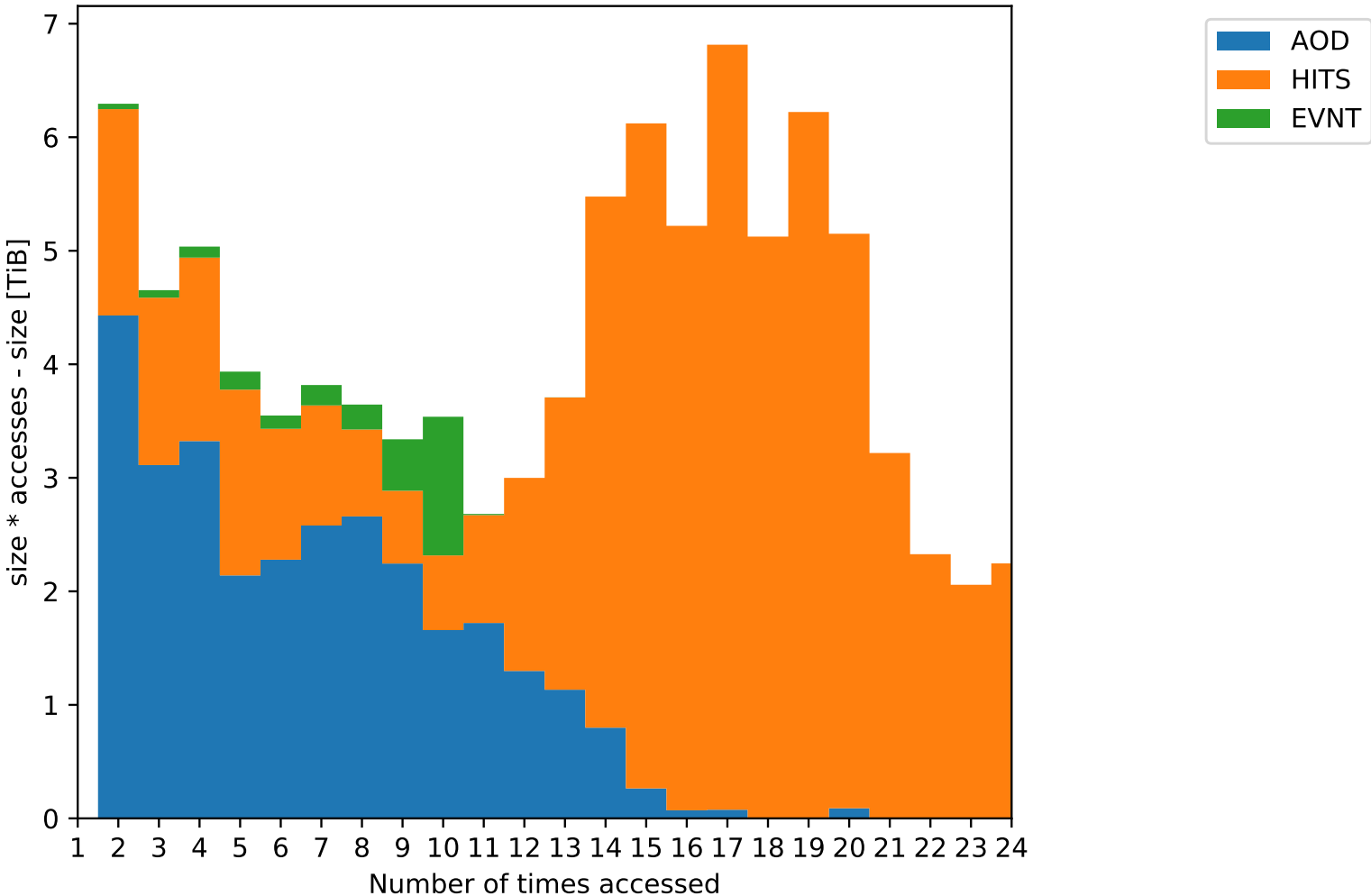
Setup

- Hardware: Old dCache pool node (from 2012):
  - Dell R710, 2x6 core Xeon L5640, 32 GB RAM, 10 Gb Ethernet
  - 60 TB Raid-6 (2x12x3TB HDD)
    $\rightarrow$ second node with individual disks since November 2019
- Xrootd version 4.11.2
- Setup w/ singularity SL6 image. Full configuration: https://gitlab.physik.uni-muenchen.de/Nikolai.Hartmann/xcache-singularity-lrz/
- XCache settings:

```
pfc.ram 14g
pfc.blocksize 1M
pfc.prefetch 10
```

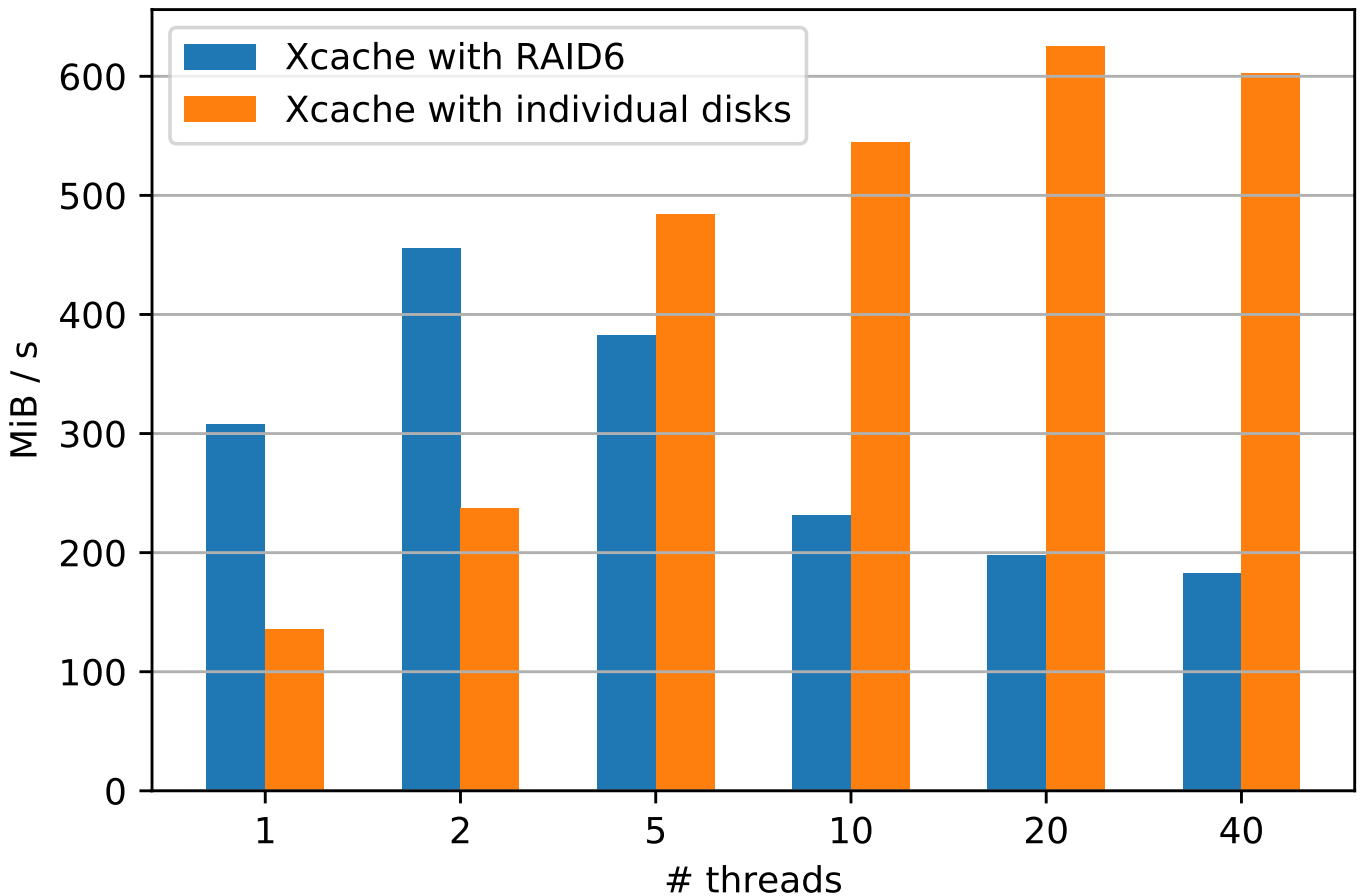
Weighted by `size * accesses - size`

Corresponding reduction in WAN traffic
(w.r.t reading everything from remote without cache)



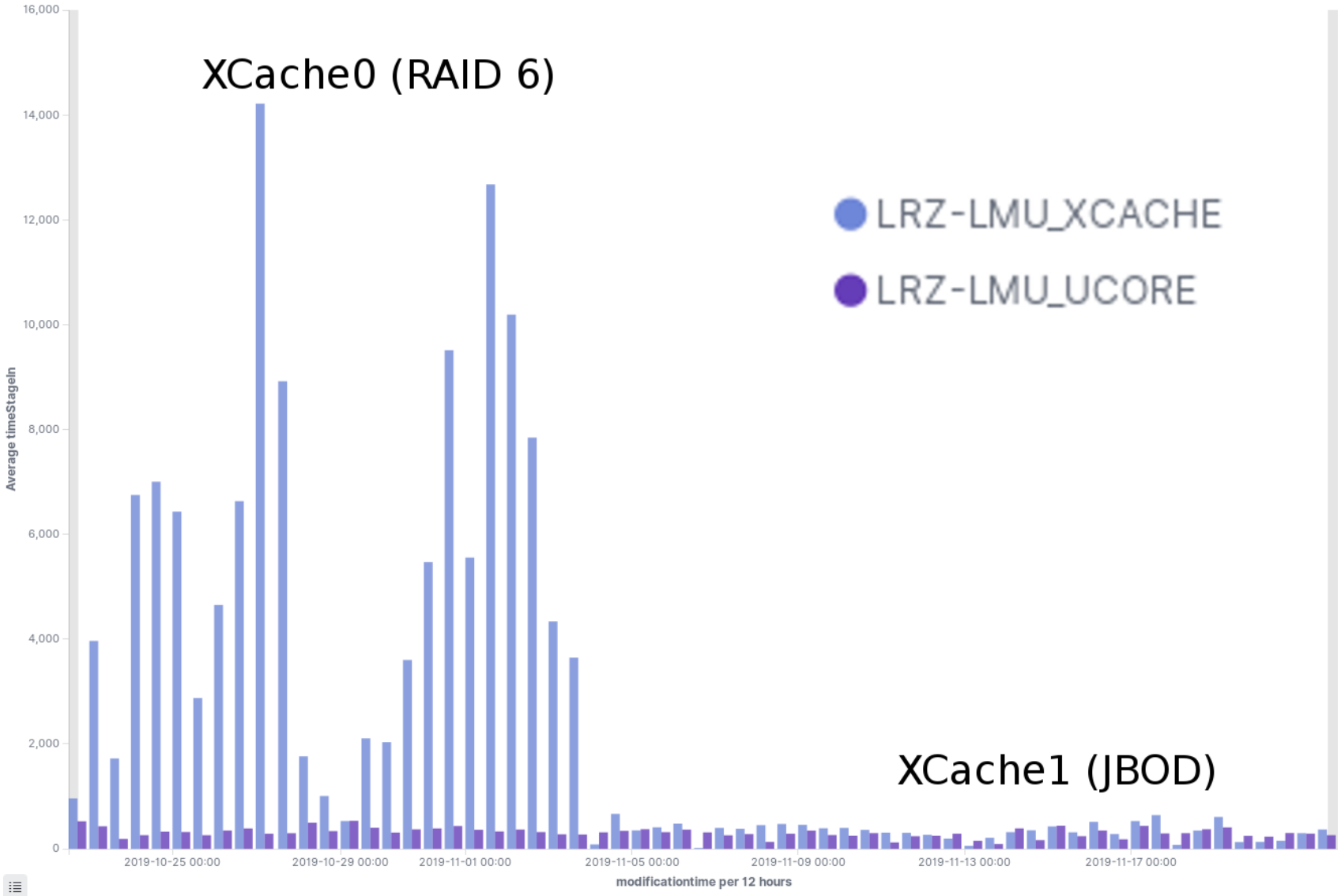
Performance for parallel reads - Raid6 vs single disks

Now similar test with an actual xcache setup:
(read random cached files through xcache, read from server)



→ same conclusion - individual disks outperform RAID for parallel reads

Stage-in times



$\rightarrow$ comparable stage-in times (with JBOD) as for non-xcache queue