

Xcache plans and studies at LRZ/LMU

Nikolai Hartmann, Guenter Duceck, Rodney Walker

LMU Munich

February 12, 2019

Use cases for caching

Caches can supplement manual/automatic data placement. Interesting scenarios:

- Processing on local batch systems without the need to replicate data
- Sites with small storage but available computing resources
- “Hospital” queues where jobs can be processed for which all slots on sites where data is available are full
- Caching for additional data - metadata, containers (http?)

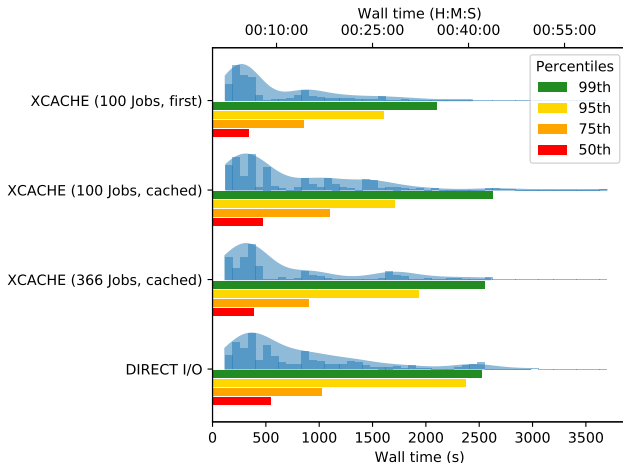
→ Testing in Munich with an Xcache server

Testing the Xcache server (local batch)

- Test cases: user analysis job and ATLAS derivation
- Analysis job I/O: $\approx 0.5 - 1.5$ MB/s
- Derivation job I/O: ≈ 3 MB/s
- Analysis job test dataset: ≈ 340 GB in 366 Files
(reading via DESY Hamburg)
- Derivation job test dataset: ≈ 33 GB in 10 Files

Job run times

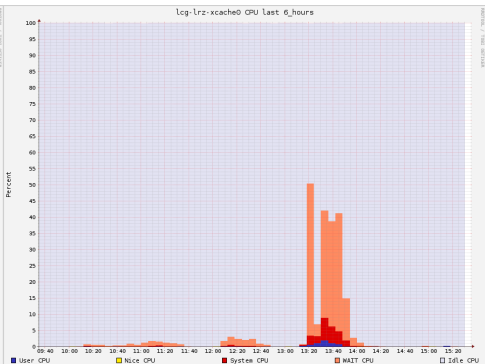
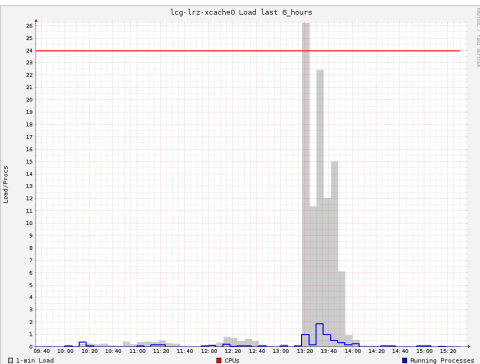
Analysis Job, data stored on well-connected DESY Hamburg site



No large difference between first and second processing via cache, direct I/O has similar performance

Xcache server monitoring

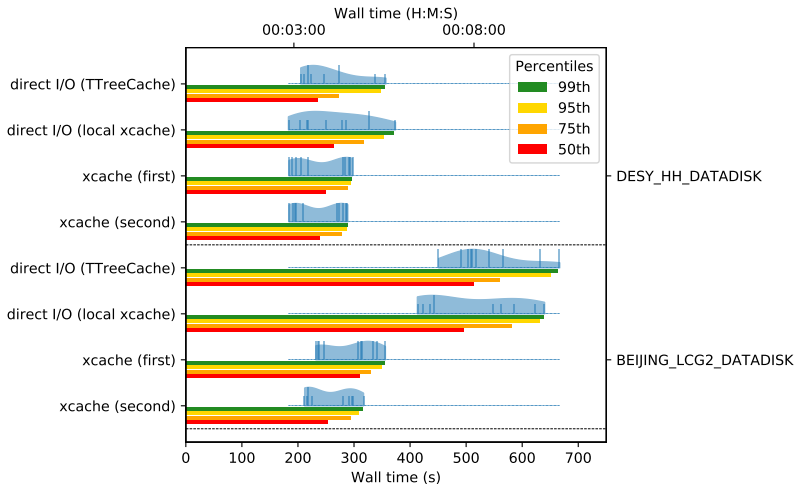
366 Analysis Jobs at once - "stress test"



Server got busy for a short period of time, but analysis job runtimes basically unaffected

Processing from different sites

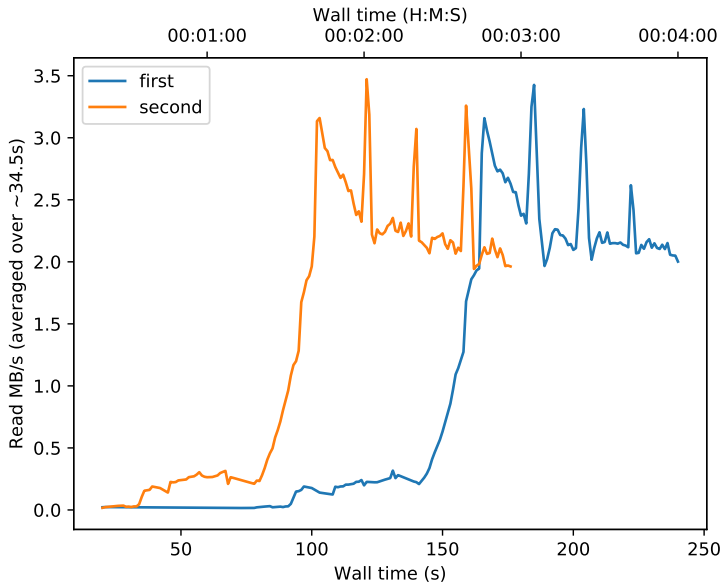
Derivation Jobs ($\approx 3\text{MB/s}$) - process 500 Events



- Differences for direct I/O and cached visible for far away sites
- Local Xcache (on each node) can serve as alternative to TTreeCache

First and second processing via Cache

Difference visible only for far away sites - Example: BEIJING

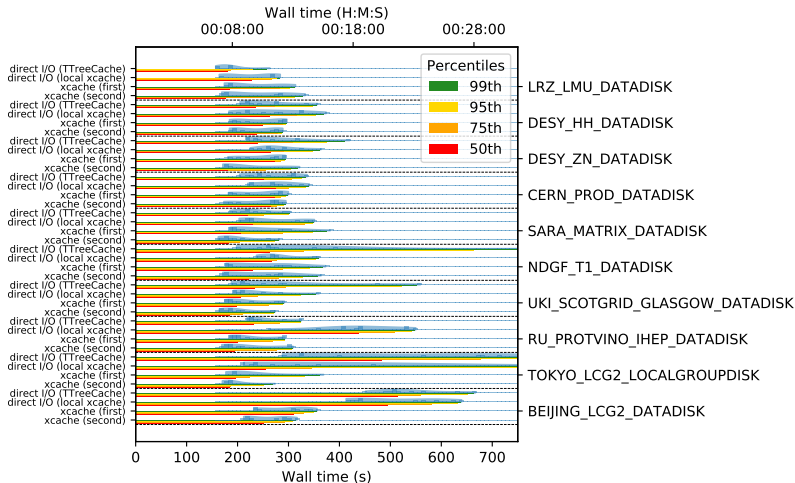


Backup

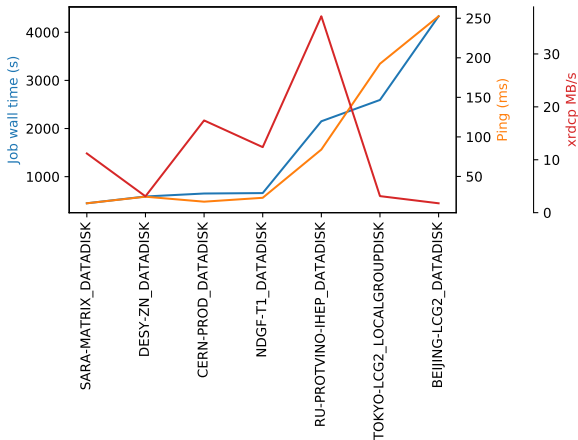
Xcache hardware/software at LRZ-LMU

- 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)
- Xrootd version 4.8.5
- Setup w/ singularity SL6 image following Wei's instructions:
<https://github.com/wyang007/rucioN2N-for-Xcache/wiki/Deploy-Xcache-via-a-Singularity-Container>
- Xcache settings:
`pfc.ram 14g`
`pfc.blocksize 1M`
`pfc.prefetch 10`

All sites

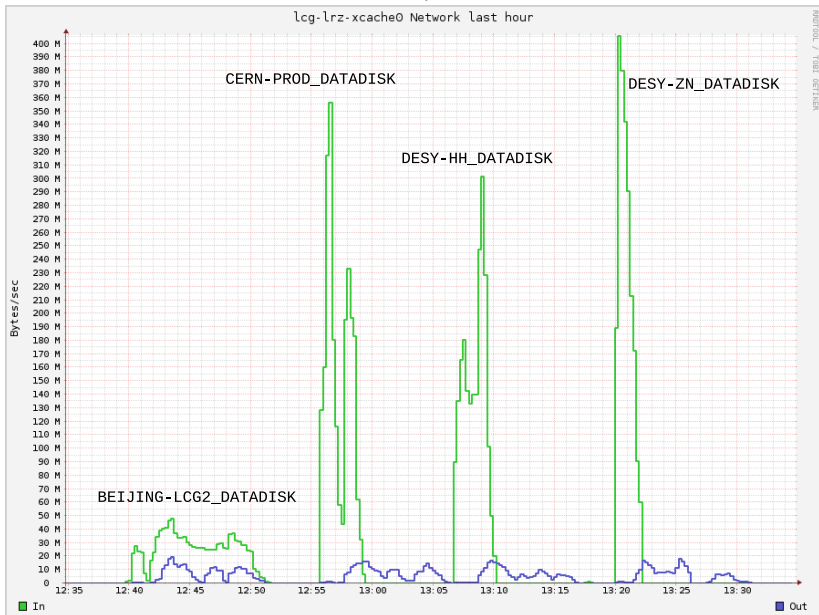


Run times without TTreeCache vs ping vs speed

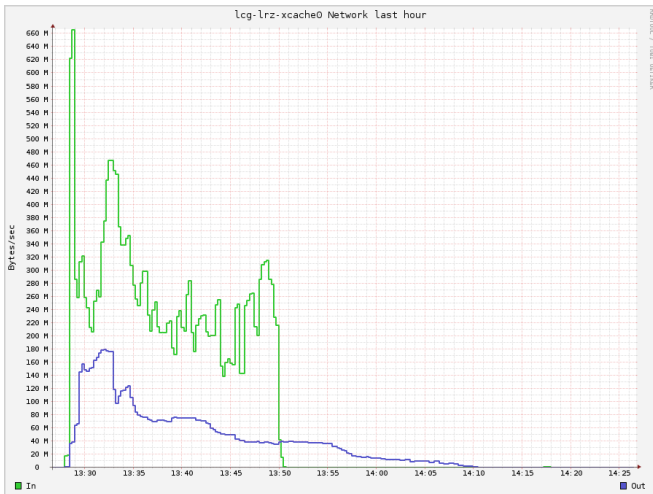


- processing time correlates with ping, not (xrdcp) transfer speed
- reading in larger blocks better (use TTreeCache)

Xcache server I/O monitoring



Analysis job “stress test” via lrz batch system



366 Jobs (one per file) at once still work fine
(total dataset size: \approx 340 GB)