# XRootD caching for Belle II

**Moritz Bauer***, Max Fischer, Manuel Giffels, Günter Quast, Matthias Schnepf

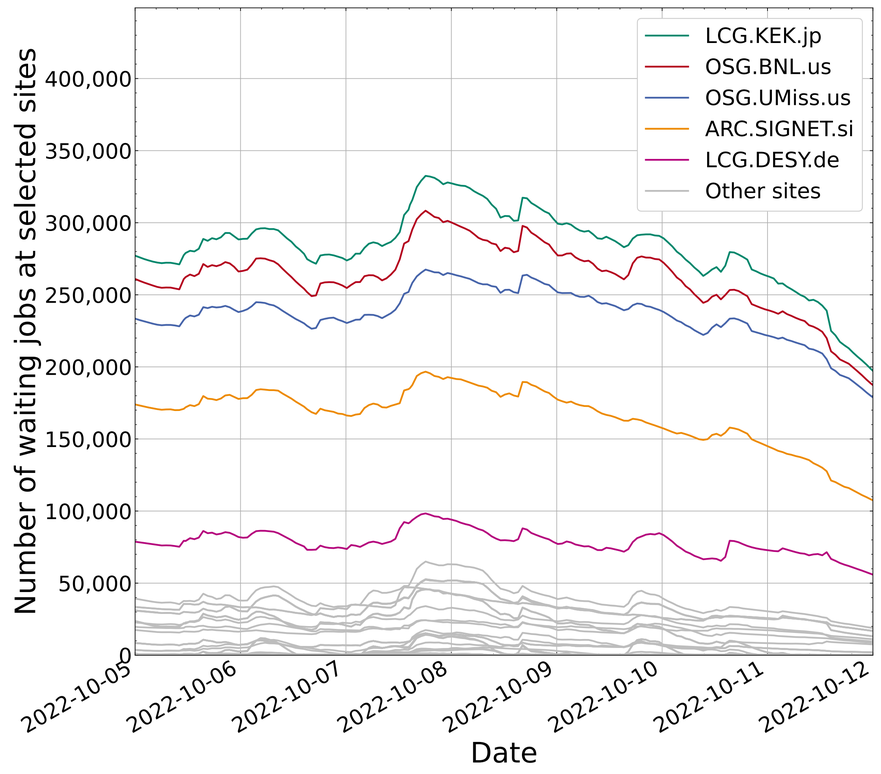
## Distributed Computing at Belle II

### Computing Infrastructure

- Approximately 30 computing sites of varying size in America, Europe and Asia.
- All sites are required to have managed Grid storage which is expensive to support.
- `DIRAC` and `Rucio` distribute workloads and files to the different sites.
  - Analysis jobs depend on input datasets which are not available on every site.

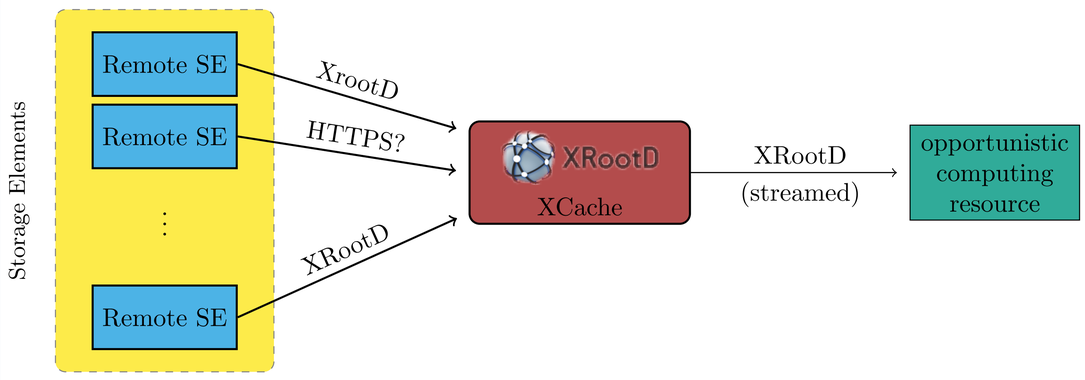
### Challenges

- Belle II follows a Grid-based analysis workflow: Local analysis & MC production is possible but discouraged.
- Many statistically-dominated analyses which process the entire dataset.
- Smaller sites with CPU resources but without managed storage cannot contribute easily.
- Varying popularity of sites and datasets can lead to uneven workloads and high numbers of waiting user jobs on individual sites.
  - Rebalancing of replicas using `Rucio`'s dataset popularity feature can help but requires manual intervention.



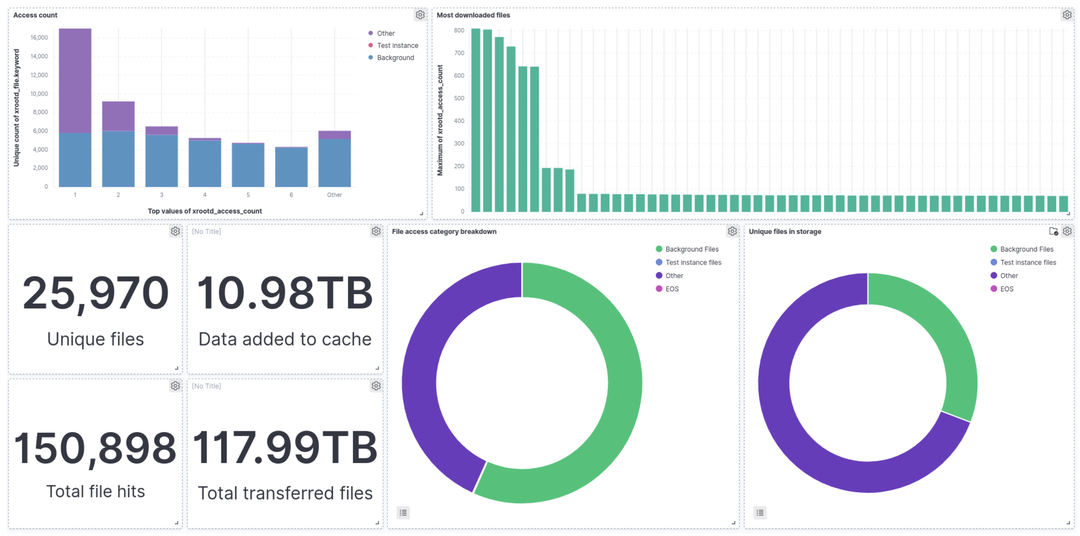

## Caching files with XRootD

- A cache server with `XRootD` ("XCache") [1] retrieves and provides files located at remote sites using the `XRootD` protocol.
- Cache server creates local file copies on demand, no manual interventions required.
- No delay is introduced as file blocks can be provided to the computing site before the entire file is downloaded.
- A redirector plugin at the computing resource redirects `XRootD` requests seamlessly to the XCache, no user interaction required.
- Using plugins, `XRootD` can not only provide files via the `XRootD` protocol but also via the more widespread `HTTP(S)/WebDAV` protocol.




## Monitoring an XRootD XCache instance

- `XRootD` emits several binary-packed monitoring streams with detailed information about file access and cache information.
- Using the `xrootdlib` package [2] to unpack the binary streams, key metrics such as access count and file size are passed into an `ElasticSearch` database and visualized with `Kibana`.
- This also allows investigating the performance of the XCache on data subsets.




## Example setup at KIT / GridKa

### Resources with COBaID & TARDIS

- These opportunistic resources are flexible and therefore perfect for caching.
- For details on the technology see poster #30.

### Current status

- Technology tested with 500TB cache in front of GridKa storage.
- Operating for 10 months, jobs running at 6 physical sites behind one Grid CE.

### Next Steps

- Cache a remote storage element at "Jozef Stefan" Institute in Ljubljana, Slovenia (`ARC.SIGNET.si`).
- Redirect HTTP(S)/WebDAV requests to the cache via a custom `DIRAC` pilot.

## A preliminary conclusion

Caching with XRootD "XCache" is an easy-to-implement approach for Belle II to decrease job waiting times and increase site utilization!

[1] XRootD: http://xrootd.org
[2] xrootdlib: https://github.com/maxfischer2781/xrootdlib

ACAT 2022, Bari, Italy

*moritz.bauer@kit.edu