# RNTuple: Towards First-Class Support for HPC data centers

GIOVANNA LAZZARI MIOTTO ⋆
Universidade Federal do Rio Grande do Sul (BR)

JAVIER LOPEZ-GOMEZ †
EP–SFT, CERN

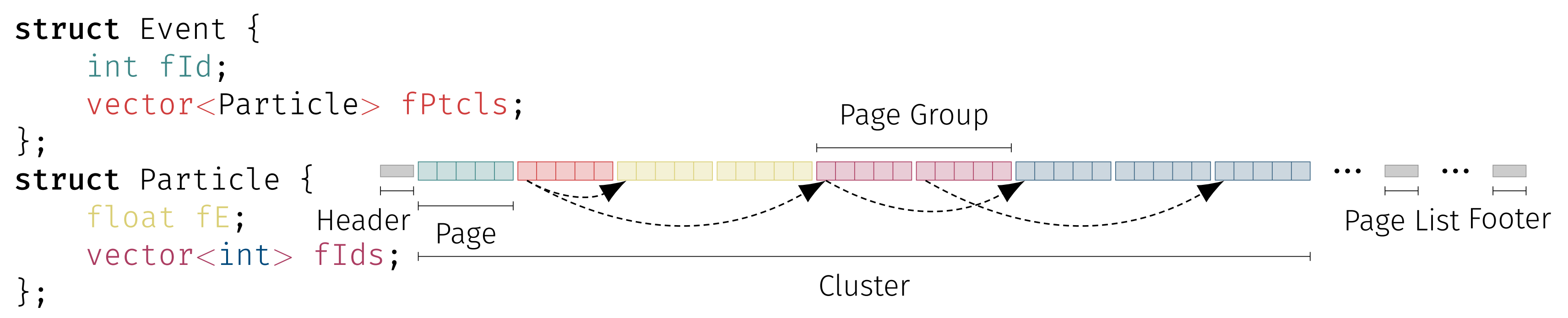
## RNTuple and Intel DAOS

With future colliders projected to generate **10×** as much event data, **RNTuple** is ROOT's I/O subsystem for next-generation HENP analysis, targeting:

- Low-latency, high-bandwidth NVMe devices
- Asynchronous and concurrent bulk I/O
- Distributed object stores

RNTuple's storage layer can be specialized for different storage systems to leverage **HPC data centers**, e.g., **object stores** [2]. In this work, we present significant improvements to RNTuple's **DAOS backend**.

On-disk, RNTuple has a columnar data storage model organized in *pages*, *page groups* and *clusters* (Figure 1). For key-value stores, we provide a remapping of RNTuple *pages* to DAOS `objects`.



```
struct Event {
    int fId;
    vector<Particle> fPtcls;
};
struct Particle {
    float fE;
    vector<int> fIds;
};
```

Fig. 1: RNTuple's on-disk format

### What is DAOS?

- Open-source object store for massively distributed NVM devices
- Highly scalable storage due to simple objects (Figure 2)
  - `dkey` impacts data co-locality in pool shard
  - `akey` specifies value within `{object, dkey}`
- Storage system for Argonne's Aurora exascale supercomputer

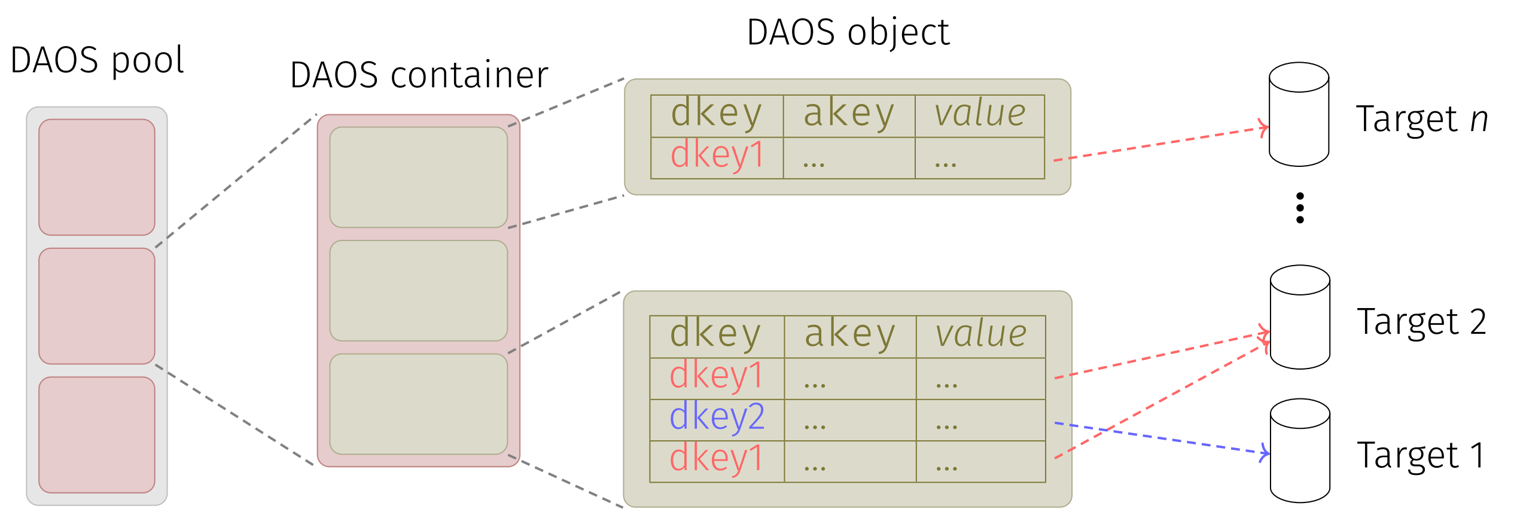


Fig. 2: Simplified DAOS top-level organization

## Efficient Storage of HENP data in DAOS

One of the expected uses of object stores is as a transient storage for distributed analysis. Thus, our improvements [1] are designed to **shorten the data import stage**.

1. **Vector writes**, i.e., committing a vector of buffered *page groups* in a single call, allowing for parallelized writes
2. **Coalesced R/W requests** by `{object, dkey}` to minimize I/O calls and exploit target parallelization
3. **Improved RNTuple ↔ DAOS mapping** preserving *page* co-locality, tuned for typical HENP analysis patterns:

   *cluster* ↦ `object`, *column* ↦ `dkey`, *page* ↦ `akey`

4. Server-side concatenation of row-adjacent values into page chunks targeting **near-optimal throughput independently of native page size** by associating multiple IOVs per *akey* (Figure 3).
5. And more: better event management, many ntuples per container

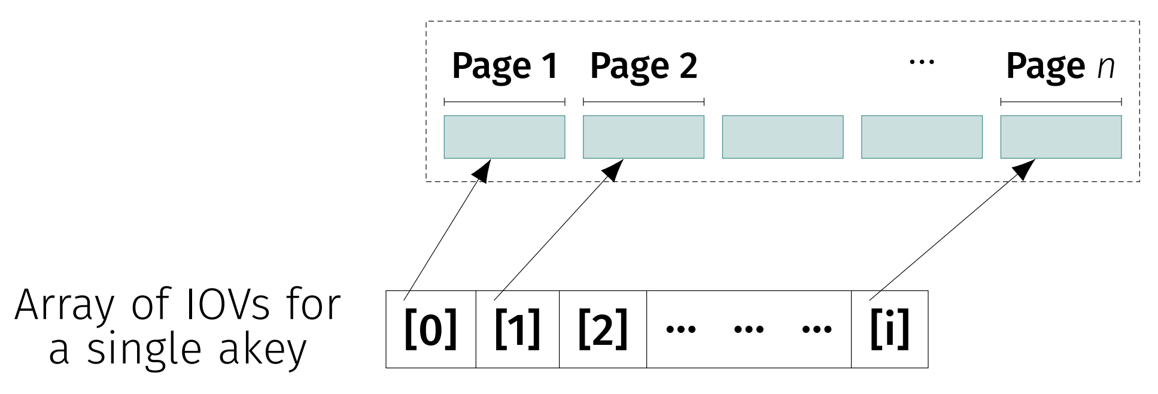


Fig. 3: Multiple IO vectors (IOVs) are associated to a key in the object store

## Performance Evaluation

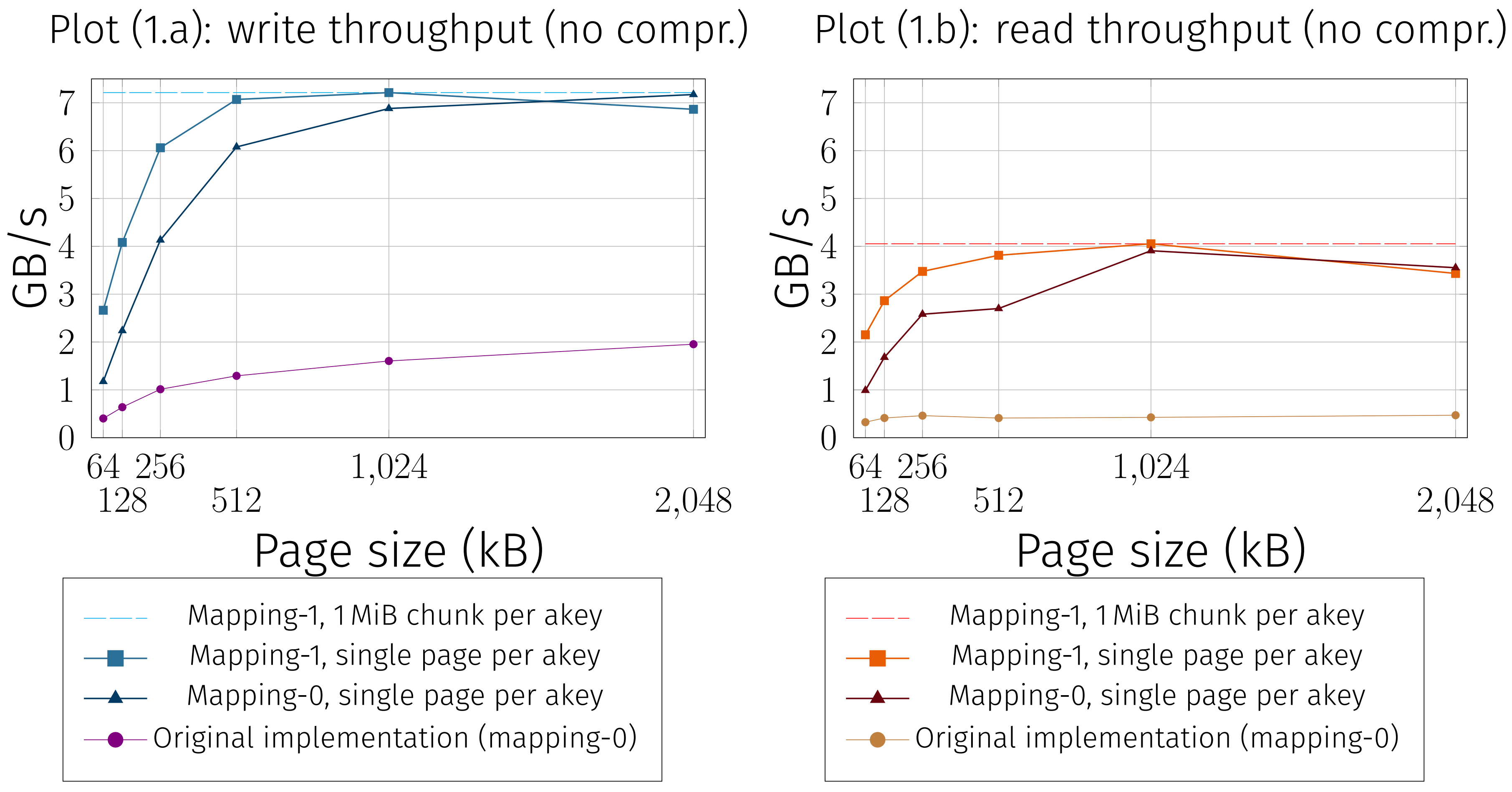
Dataset: **LHCb OpenData B2HHH**, with 8.5 million events spanning 26 branches, replicated tenfold for a total size of 15 GB.

Evaluation:

- with/out compression (`zstd`)
- Mapping-0: $\{page, k_d, k_a\} \mapsto$ `{object, dkey, akey}`
- Mapping-1: $\{cluster, column, page\} \mapsto$ `{object, dkey, akey}`

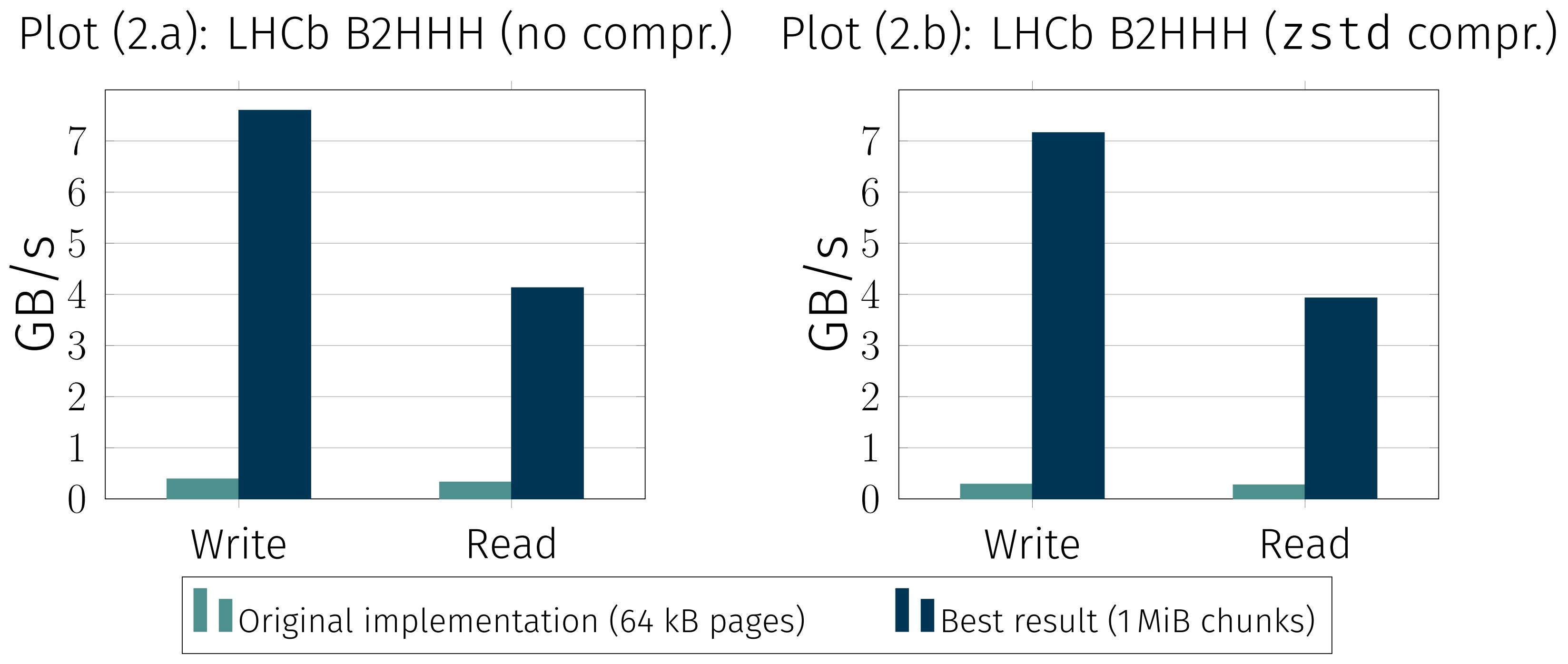
We measured (single thread, Infiniband RDMA):

- Throughput measured during data import into DAOS, in GB/s
- End-to-end analysis throughput from storage to histogram, in GB/s



Plot (1.a): write throughput (no compr.)

Plot (1.b): read throughput (no compr.)



Plot (2.a): LHCb B2HHH (no compr.)

Plot (2.b): LHCb B2HHH (`zstd` compr.)

## Conclusions

Our results show the latest improvements in the DAOS backend, which facilitate high-throughput distributed analyses in HPC centers.

- **Over 5 times the R/W throughput** for data import and end-to-end realistic HENP analysis with native 64 kB pages;
- Object-based storage made **independent of an ntuple's native configuration** (*page*, *cluster* sizes)

Next steps:

- Extended evaluation of **multi-threaded, distributed analysis** with ROOT's `RDataFrame` in order to saturate the link layer
- Storage support for **Amazon S3** object store
- Optimized vector writes for RNTuple's file backend

RNTuple is scheduled to become production grade in 2024. We appreciate the first experiments implementing RNTuple writers in their workflows, providing feedback on features and performance.

ACKNOWLEDGMENTS: *This work benefited from support by the CERN Strategic R&D Programme on Technologies for Future Experiments CERN-OPEN-2018-006. Access to the hardware for the experimental evaluation was provided by Hewlett-Packard Enterprise.*

### References

[1] root-project PRs: *#10795*, *#10860*, *#10927*, *#10944*, *#10982*, *#11466*
[2] *Exploring Object Stores for High-Energy Physics Data Storage.* `https://doi.org/10.1051/epjconf/202125102066`

⋆ <glmiotto@inf.ufrgs.br>
† <javier.lopez.gomez@cern.ch>