Efficient Analysis Facilities

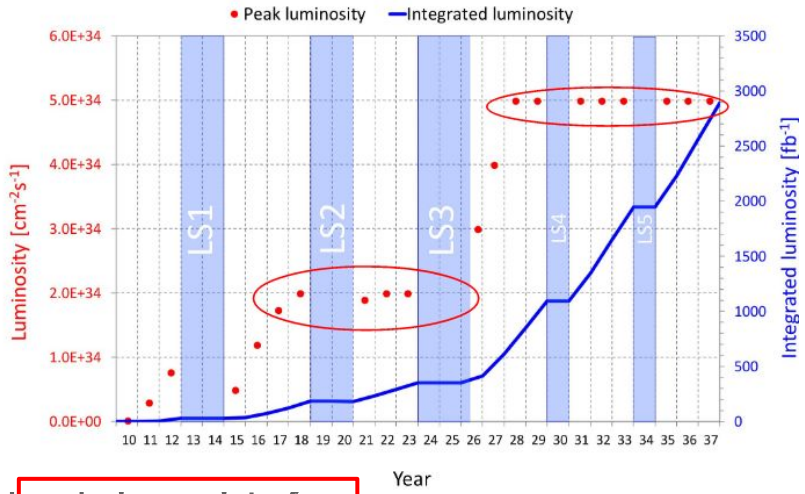
EP R&D - Software WP

Jakob Blomer

2021-02-10

Reminder: Analysis Challenge

- HL-LHC challenge: first major milestone on the way towards future accelerators and detectors
  - From 300fb^{-1} in run 1-3 to 3000fb^{-1} in run 4-6
  - 10B events/year to 100B events/year
  - Real analysis challenge depends on several factors: number of events, analysis complexity, number of reruns, etc.
    - **As a starting point, let's prepare for ten times the current demand**
- Developments in our favour
  - Experiment R&D on central, compact AODs, e.g. CMS nanoAOD, ATLAS DAOD_PHYSLITE 1kB - 10kB per event
  - R&D on ROOT I/O throughput
    - Currently 100kB - 10MB/s per core
    - In the lab: 100MB/s per core
      - Google: 200MB/s per core
- Faster storage devices: NVMe (~10x faster), 100GbE
- Object store distributed storage systems



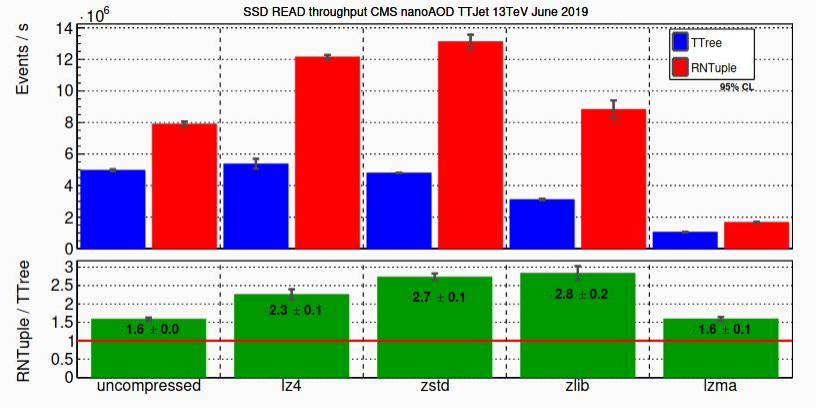
- **Calls for major R&D and engineering on I/O subsystem and analysis user interface**

Increase data throughput

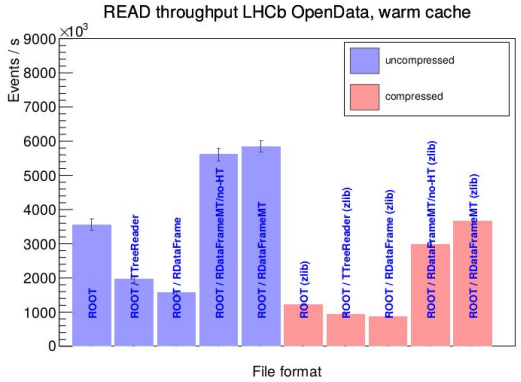
Increase analyst's throughput

Overview: R&D Programme on Faster Analysis

1) Increase data throughput
   - Data format R&D
   - ROOT RNTuple is a research prototype for next-generation event I/O
   - Expect ~25% smaller files, x2-5 better single-core throughput on SSD
   - Prototype available in `ROOT::Experimental`
   - Entering first exploitation phase



2) User interface R&D
   - RDataFrame introduced as ROOT's declarative analysis toolkit
   - Two major R&D challenges
     i) Optimal translation to low-level I/O routines
     ii) Distributed execution engine: how to run the analysis on my laptop on O(1000) cores

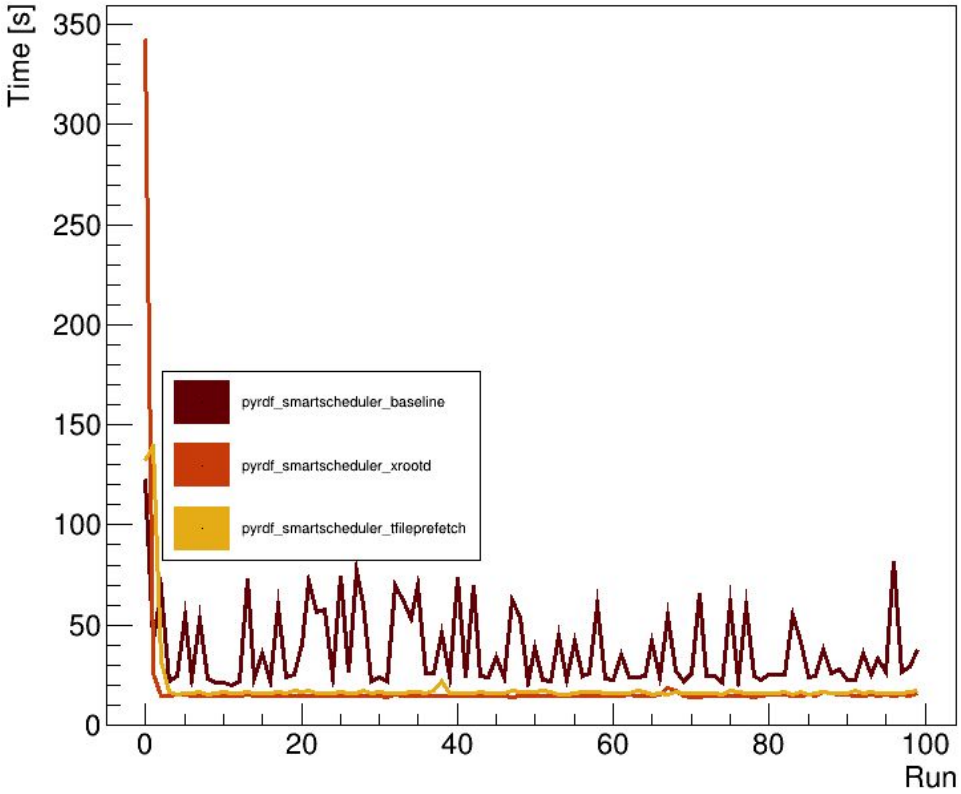


RNTuple Goals

Based on 25+ years of TTree experience, we redesign the I/O subsystem for

- Less disk and CPU usage for same data content
  - 25% smaller files, x2-5 better single-core performance
  - 10GB/s per box and 1GB/s per core sustained end-to-end throughput (compressed data to histograms)
- Native support for object stores (targeting HPC)
- Lossy compression
- Systematic use of exceptions to prevent silent I/O errors

2020 Main Results I: Distributed Caching

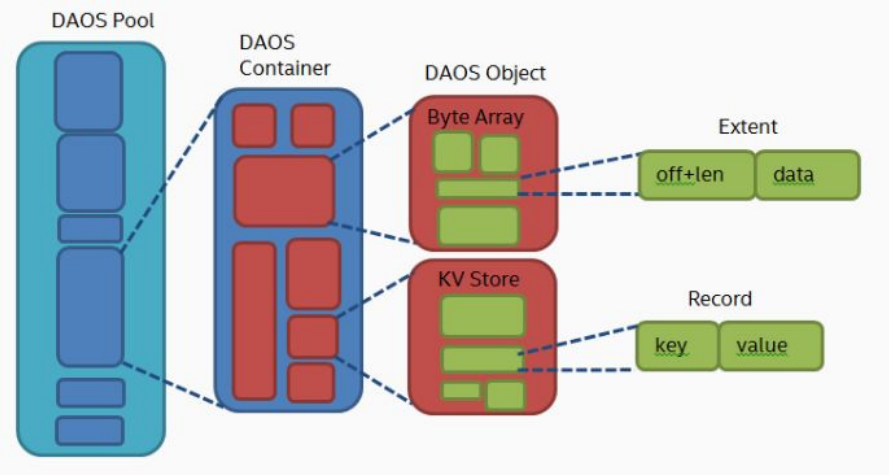


- Stability issues with TFilePrefetch seem to be solved. The workers almost always get assigned the same range
- TFilePrefetch and XRootD cache scenarios have very similar execution time
- XRootD starts off quite slower, the workers trigger caching different portions of the dataset separately.

→ **Vincenzo's PPP presentation:** https://indico.cern.ch/event/954326/

2020 Main Results II: Mapping RNTuple ➜ DAOS

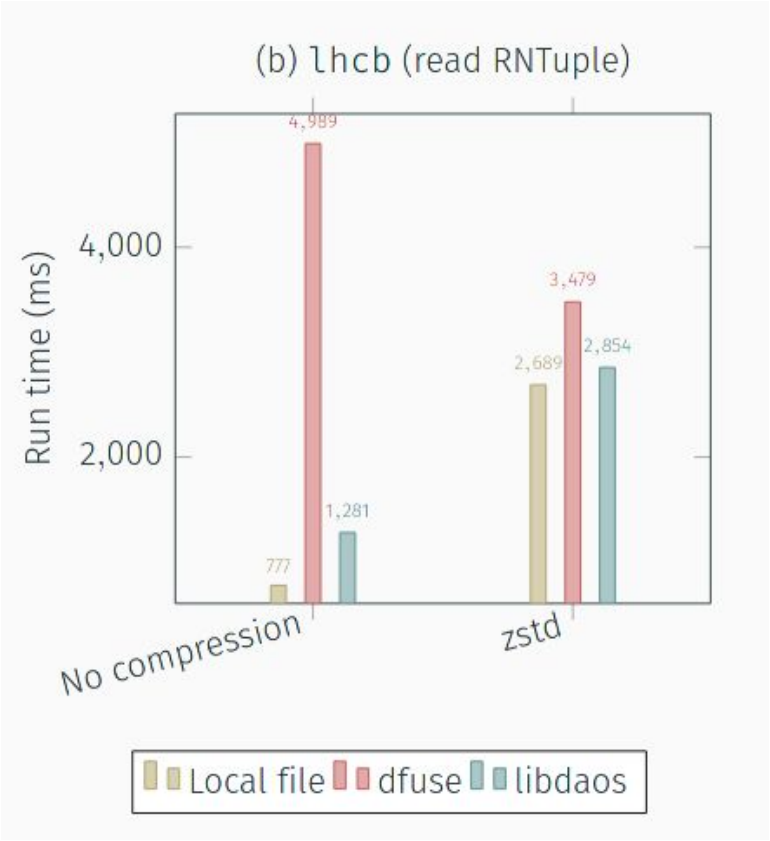
```
auto ntuple = RNTupleReader::Open("DecayTree", "daos://41adf800b537...");
```



DAOS object: a key–value store with locality. The key is split into **dkey** (distribution key) and **akey** (attribute key).

RNTuple → Container
Cluster → Object
Page group → dkey
Page → akey

2020 Main Results II: DAOS Preliminary Read Results



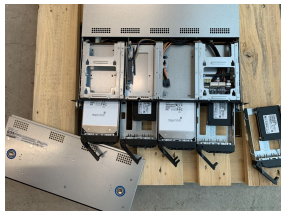
Preliminary results

- 1 client, 3 servers (provided by CERN openlab)
- Significantly faster read with RNTuple than with dfuse compatibility layer
- ~1GB/s throughput (should be even higher, still under investigation)

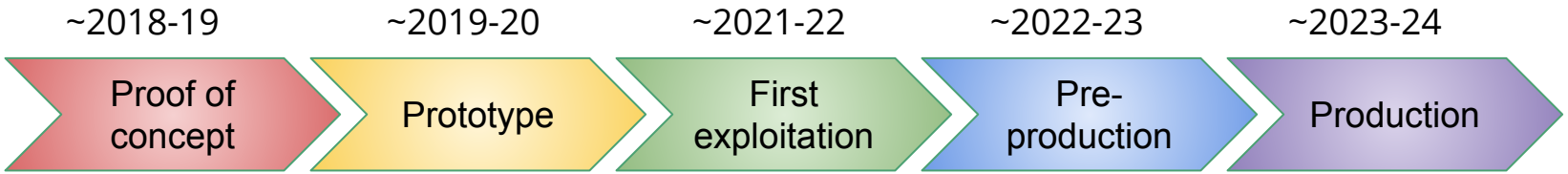
RNTuple Workforce in 2020/2021

- Vincenzo Padulano
  - Investigation on distributed data caching for analyses expressed in RDataFrame
    - pyRDF and Spark as technology basis
- Javier Lopez
  - Integration of RNTuple with Intel DAOS
    - Object store system for HPCs
    - 220PB planned for Argonne HPC (possibly delayed)
- Max Orok (IRIS-HEP)
  - CMSSW nanoAOD output module
- GSoC student (tbs)
  - TTree ➔ RNTuple disk-to-disk conversion
  - Supervised by Javier

- R&D hardware: Fast I/O development machine
  - Fast spinning disks, SSDs, 100GbE: capable of testing 10GB/s throughput
  - Should be extended by a GPU capable of direct storage
- Intel DAOS testbed provided by CERN openlab



RNTuple Development Plan



- Class design
- File format R&D

- Interplay with other ROOT classes
- Performance validation

- Interplay with experiment frameworks
- podio integration
- Schema evolution

- PB scale test cases
- Production tests with last-stage ntuples

- Ready to use for new runs

Available in `ROOT::Experimental`
Note: TTree technology will remain available for the 1EB+ existing data sets

Ongoing: NanoAOD RNTuples

IRIS-HEP is funding a project to generate NanoAODs in RNTuple format!

- Validates that RNTuple can handle complete nanoAOD EDM
- Provides first experience with framework integration
- Allows for tuning multi-threaded write
- Allows for large-scale comparison between TTree and RNTuple

Goal

A CMSSW output module to write NanoAODs in RNTuple format.

EP R&D in the broader RNTuple 2020/2021 Plan

EP R&D

1.	**Multi-threaded decompression**	Progress: done
2.	**Async reading with io_uring**	Progress: done
3.	**I/O scheduler:** cluster preloading & caching	Progress: done
4.	**Friends and chains:** virtual storage backends	Progress: almost there
5.	**Fast merging, hadd support**	Progress: almost done
6.	**TTree to RNTuple converter:** disk-to-disk conversion as in hadd	Progress: drawing board
7.	**RBrowser integration**	Progress: well underway
8.	**User-facing bulk API:** retrieve spans of entries	Progress: drawing board
9.	**RDF optimization**: use of new RVec, proper use of clusters, etc.	Progress: drawing board
10.	**DAOS (object store) backend including “data mover”**	Progress: well underway
11.	**Buffered writes:** cluster optimization, multi-threaded compression	Progress: drawing board
12.	**CMSSW nanoAOD generation in RNTuple format**	Progress: drawing board
13.	**Comparison with HDF5 on HPC**	Progress: drawing board
14.	**RNTupleLite:** C API for basic read support	Progress: well underway
15.	**Podio backend**	Progress: drawing board

Milestones 2021

Note: this covers only the *EP R&D* contribution to the complete RNTuple PoW

1) **Main Milestones**
   - Merging of the Intel DAOS object store backend.
     Includes unit and integration tests, data mover, in-depth performance validation (see dissemination plan), if possible large-scale test at Argonne
   - Performance comparison to HDF5, it time allows also Apache Arrow & Parquet
   - In collaboration with Key4HEP: RNTuple prototype generator for podio
   - GSoC project: TTree to RNTuple converter (supervised by Javier)
   - IRIS-HEP project: RNTuple nanoAOD generation (Max Orok)

2) **Dissemination**
   - vCHEP 2021: RNTuple DAOS performance evaluation
   - Towards the end of the year: RNTuple overview paper (journal to be selected,
     e.g. Computing & Software for Big Science, Frontiers in Big Data, Platform for Advanced Scientific Computing ACM-PASC)
   - ROOT I/O Workshop (tba)