CCE-SOP RNTuple Work

Dr Christopher Jones
CCE All-Hands Meeting
23 July 2024

Overview

- What is RNTuple
- CCE-IOS API Review
- Early Measurements

What is RNTuple

- ROOT is a library used by much of HEP for storage

- RNTuple is ROOT's new file format
  - expected to replace the old format on ~5 year time scale

- ROOT's goals for RNTuple
  - take what they have learned over the last ~30 years
  - more time/space efficient than old format
  - API more inline with modern C++ practice

- Is in an alpha release now
  - ROOT wants a 1.0 release end of this year

API Review

- ROOT developers asked CCE to do an API review on RNTuple
  - Specifically if API is sufficient for use in experiment's production applications

- CCE-SOP held dedicated meetings on the new APIs
  - ATLAS and CMS developers provided feedback

- CCE-SOP will write a final report to ROOT developers on the findings
  - due this Fall
  - ROOT developers have already been given the initial feedback

Evaluation for Review

- ATLAS evaluation
  - Incorporated reading/writing RNTuple into their application (ATHENA)
  - All ATLAS storable data can be handled by RNTuple

- CMS evaluation
  - CMS storage data model uses more features of C++ than ATLAS
    - e.g. storage of polymorphic objects
  - Support for the data model required new features be added to RNTuple
    - initially thought CMS would have to do a major overhaul of part of its data model
  - Initially used the CCE-SOP I/O testing framework to evaluate the new features
  - Just completed prototype reading/writing RNTuple into their application (cmsRun)
  - All CMS storable data can be handled by RNTuple *without any modifications*

Evaluation Findings

- ATLAS and CMS find the APIs sufficient for their processing applications

- Some minor API modifications were suggested to make usage easier

- Request a way to be able to tune storage of fields
  - a field is the atomic unit of storage for data in RNTuple
  - from experience storing data in the old ROOT format, experiments have needed to help ROOT improve the storage size by providing hints to the storage API
    - e.g. how much memory to accumulate for the field before compressing and storing it

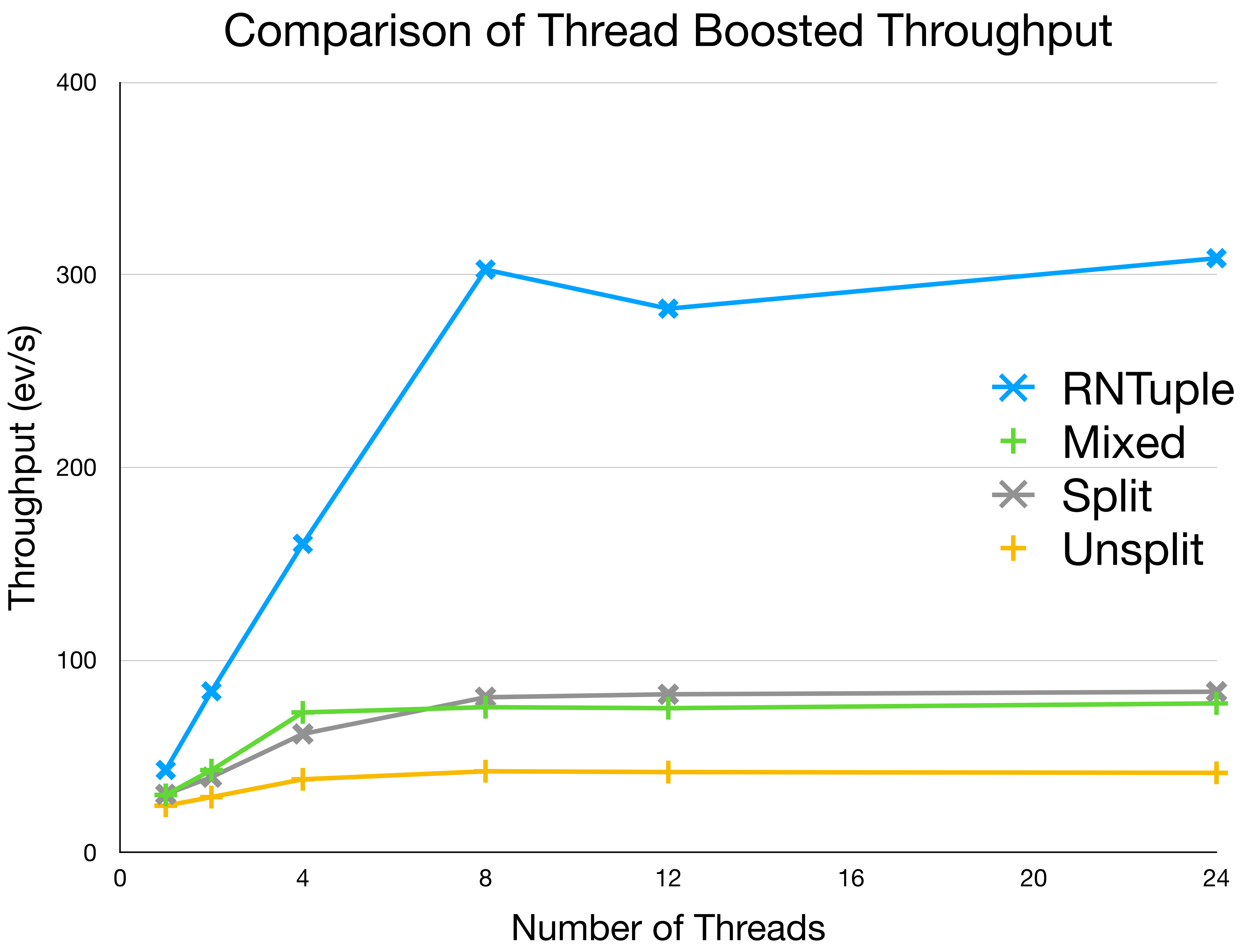
Early RNTuple Writing Performance Measurements

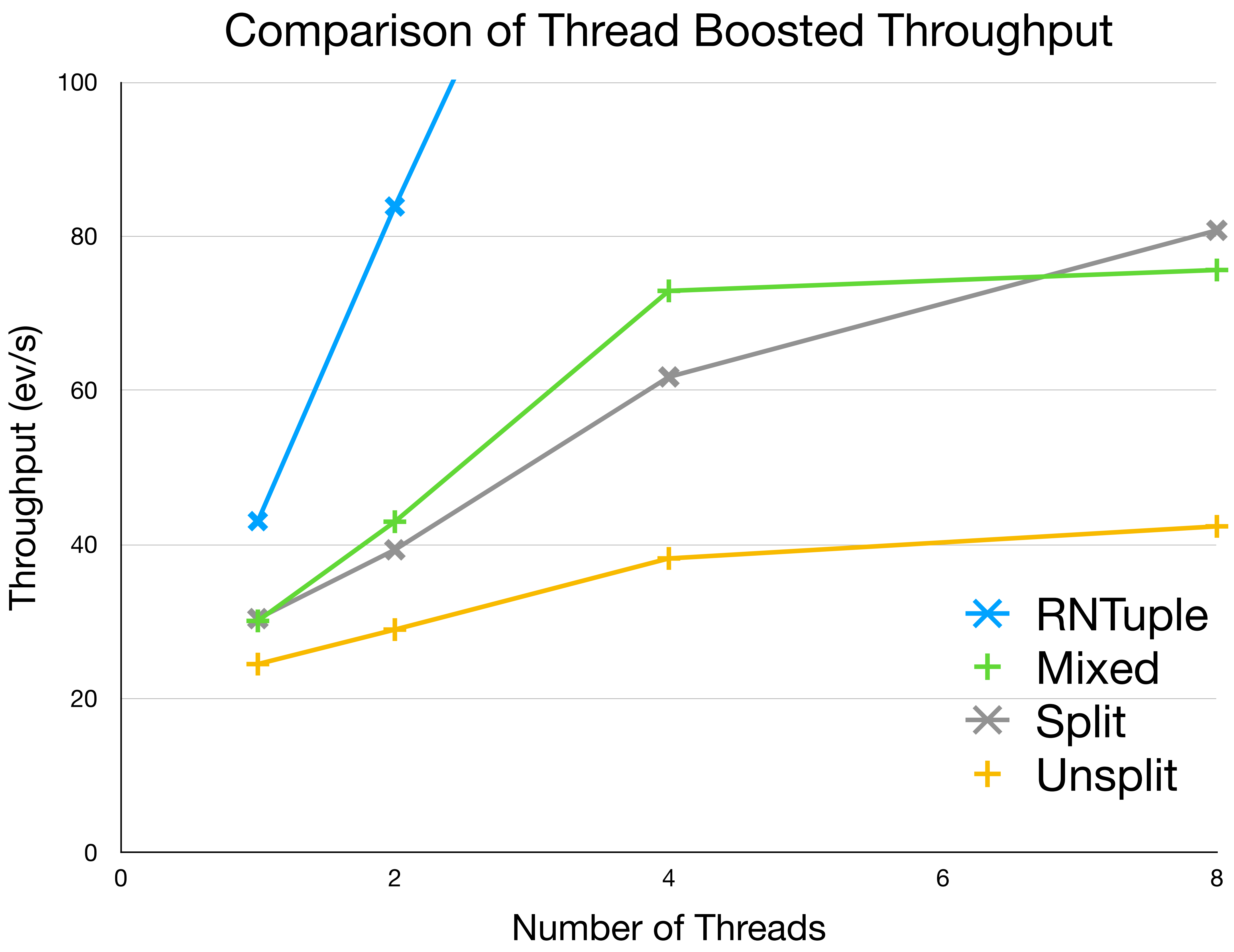
- Used the SOP I/O testing framework

- Read a subset of CMS smallest data format
  - done before new features added to RNTuple

- Comparing event throughput and memory usage compared to three different storage options available in older ROOT format
  - **unsplit**: store data product in one buffer
  - **split**: store data product into multiple buffers grouping like data
  - **mixed**: choose split or unsplit on what works best for the given data product

Event Writing Throughput Results

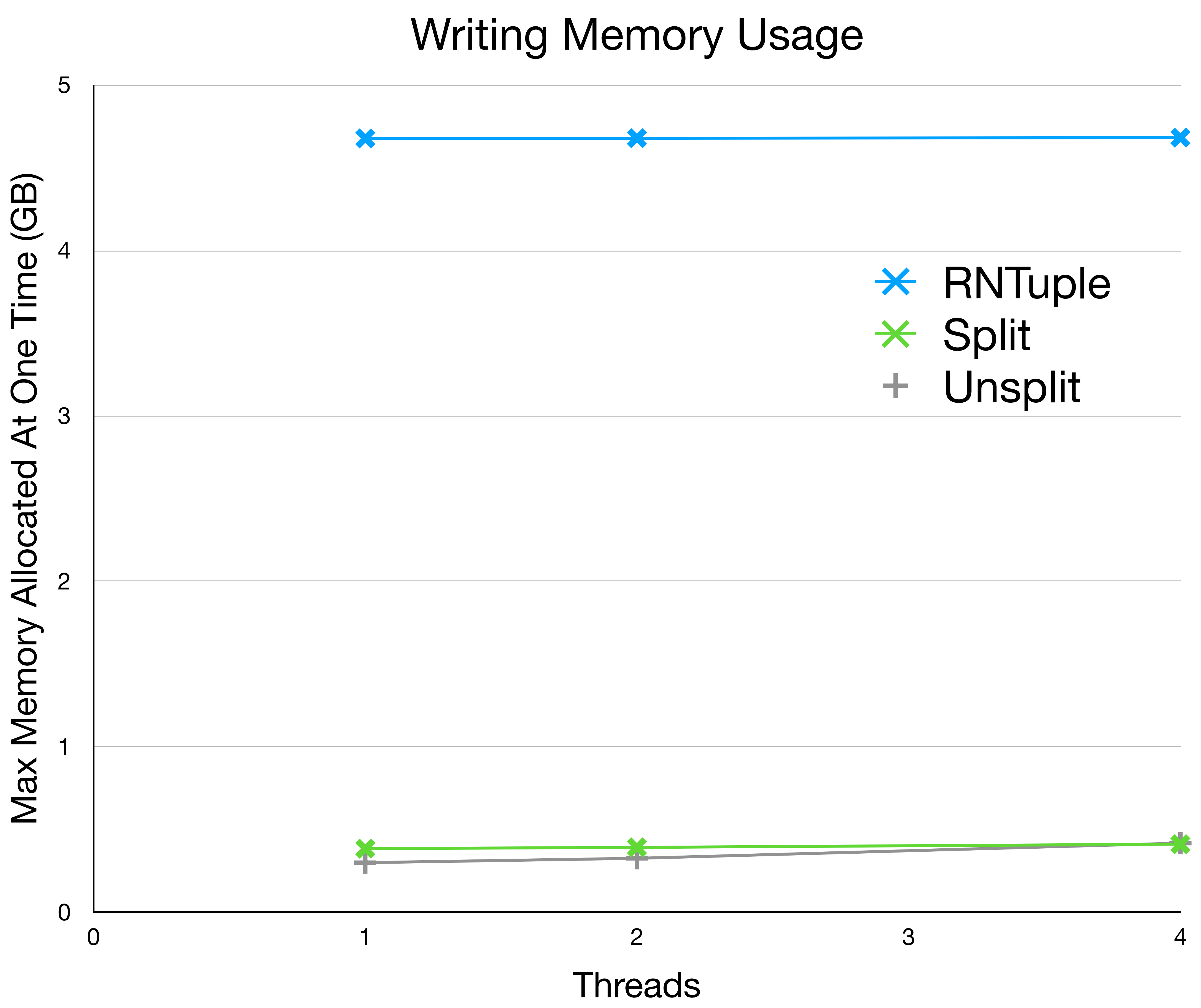




RNTuple scales perfectly up to 8 threads

Older ROOT format has much weaker scaling

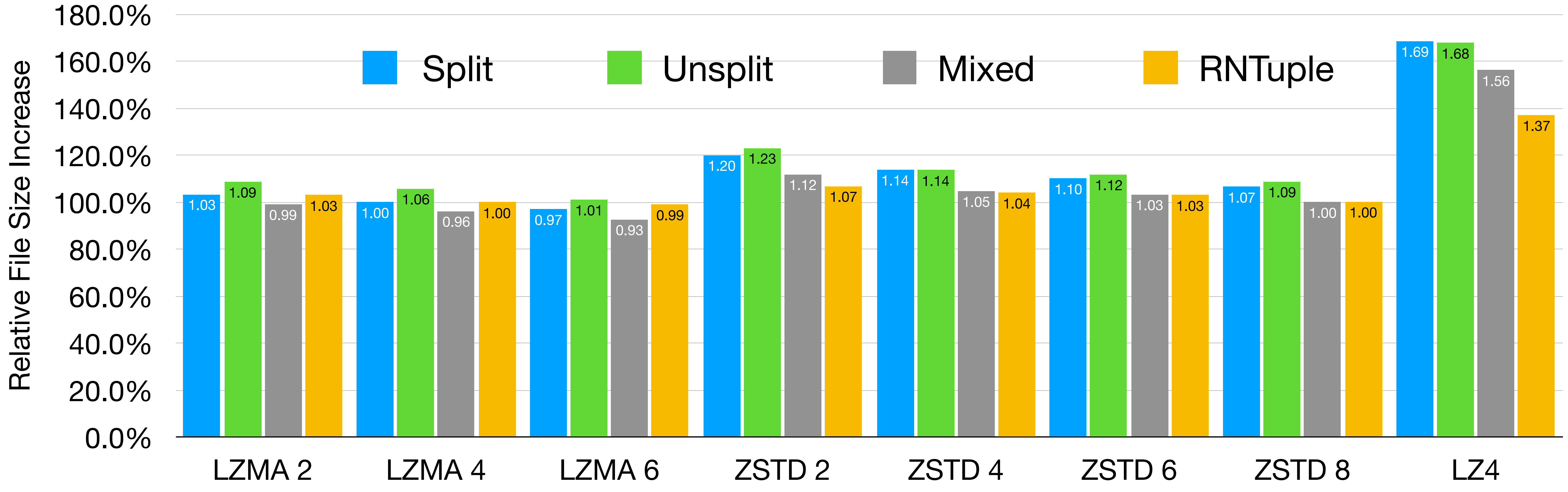
Memory Comparison



RNTuple requires ~ 10x more memory than old ROOT format

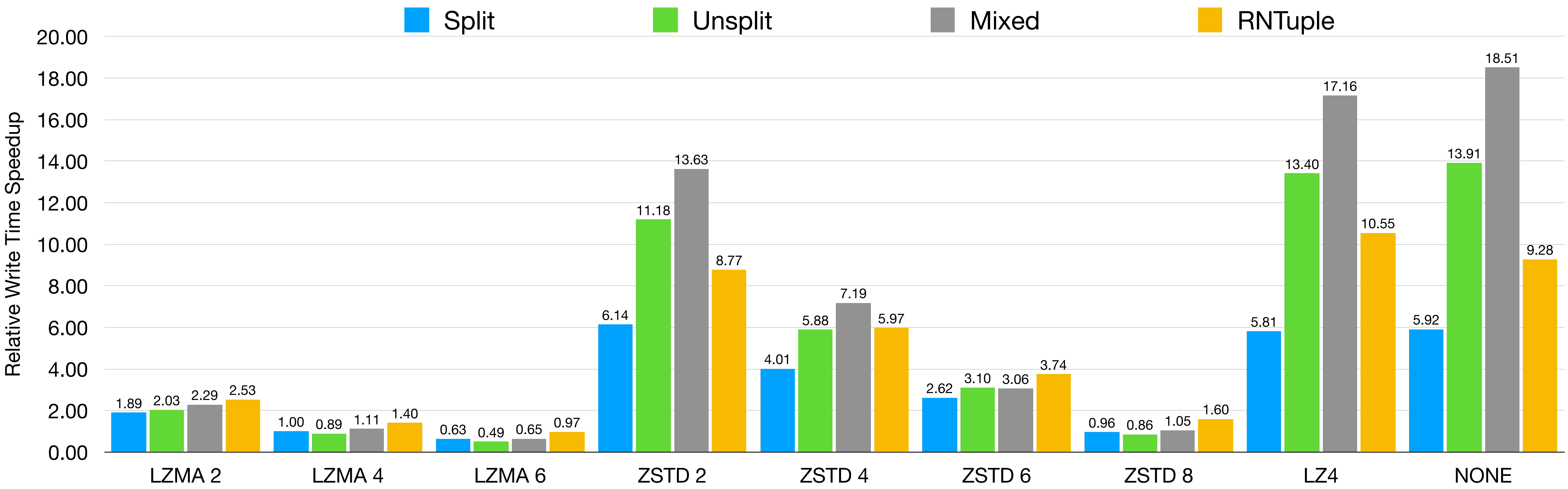
This is allocation requested, actually max RSS size is much smaller

Effect of Compression: File Size



Under high compression, difference between RNTuple and old format is small

Effect of Compression: Write Speed (1 thread)



RNTuple always writes faster than old format split option

The higher the compression time, the better RNTuple performs over the old format for all options

For lower compression times, the old format runs dramatically faster for unsplit/mixed option

Performance Measurement Feedback

- ROOT developers have been given the performance measurements

- The measurements are being used to improve RNTuple performance
  - e.g. The ROOT developers believe they can decrease the memory by 9x