RNTuple Testing Using CMSSW Updated

Dr Christopher Jones
CCE SOP
25 September 2024

Update

- RNTuple was changed to do dynamic sizing for the pages
  - tail page optimization was also removed

- How did this affect the performance?

Testing Recap

- Use a standard ROOT MiniAOD input file
  - 84,000 events
- Have prototype components that can read/write RNTuple TFiles
  - have various options to control performance
- Testing procedure
  - Read the MiniAOD file
  - Write either TTree or RNTuple based file containing full content of the input

File Sizes

Format	File	Size	Relative Size
TTree	**Standard**	4.69GB	100.0%
	Fully Split	4.8GB	102.4%
RNTuple	**Fully Split (standard)**	4.75GB	**101.3%**
	Fully Unsplit	5.77GB	**123.2%**
	Partially Split	4.54GB	**96.8%**

New File Sizes

- RNTuple standard split is now better than standard TTree

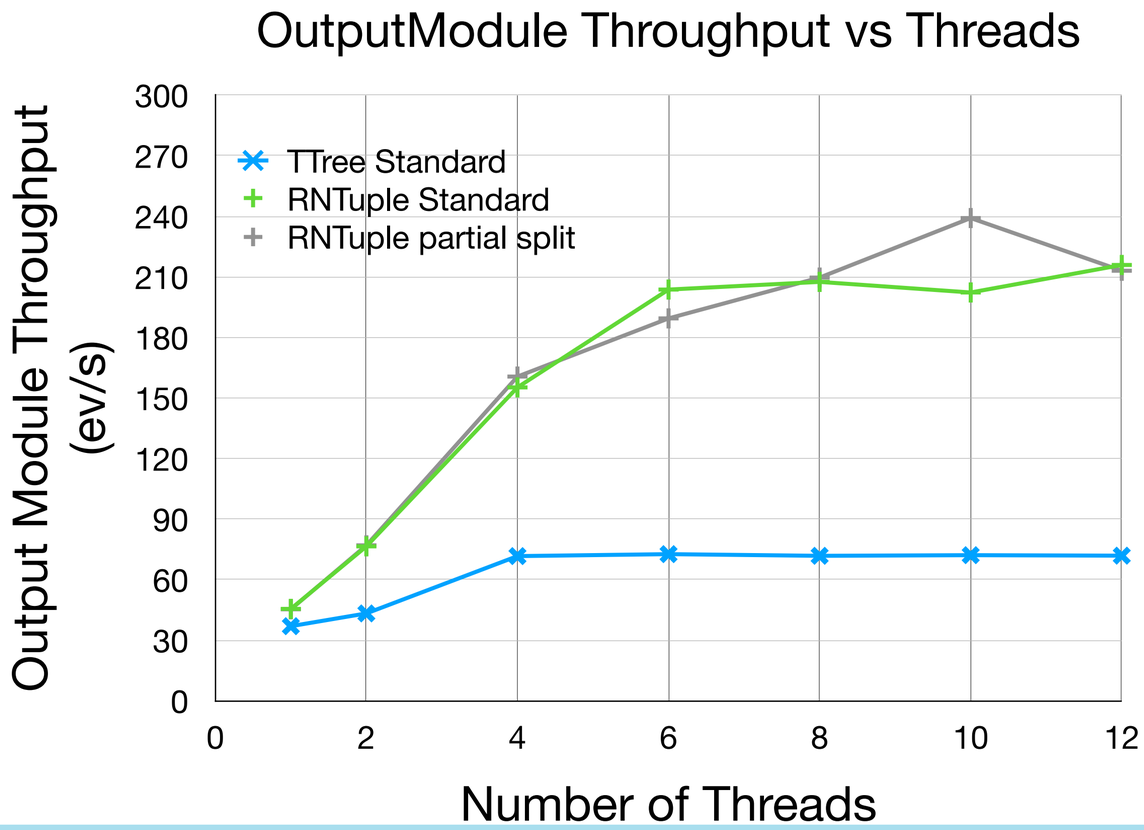
Format	File	Size	Relative Size
TTree	**Standard**	4.69GB	100.0%
	Fully Split	4.8GB	102.4%
RNTuple	**Fully Split (standard)**	4.6GB	**98.1%**
	Fully Unsplit	5.16GB	**110.0%**
	Partially Split	4.39GB	**93.7%**

Partially Split Recap

- Wrote two RNTuple files
  - Fully split (standard)
  - Fully unsplit
- Compared size on disk for each top level Fields
- Force those which are smaller unsplit to be written unsplit
- The three fields which contribute the most to unsplit are still the same

Threading Performance Previous

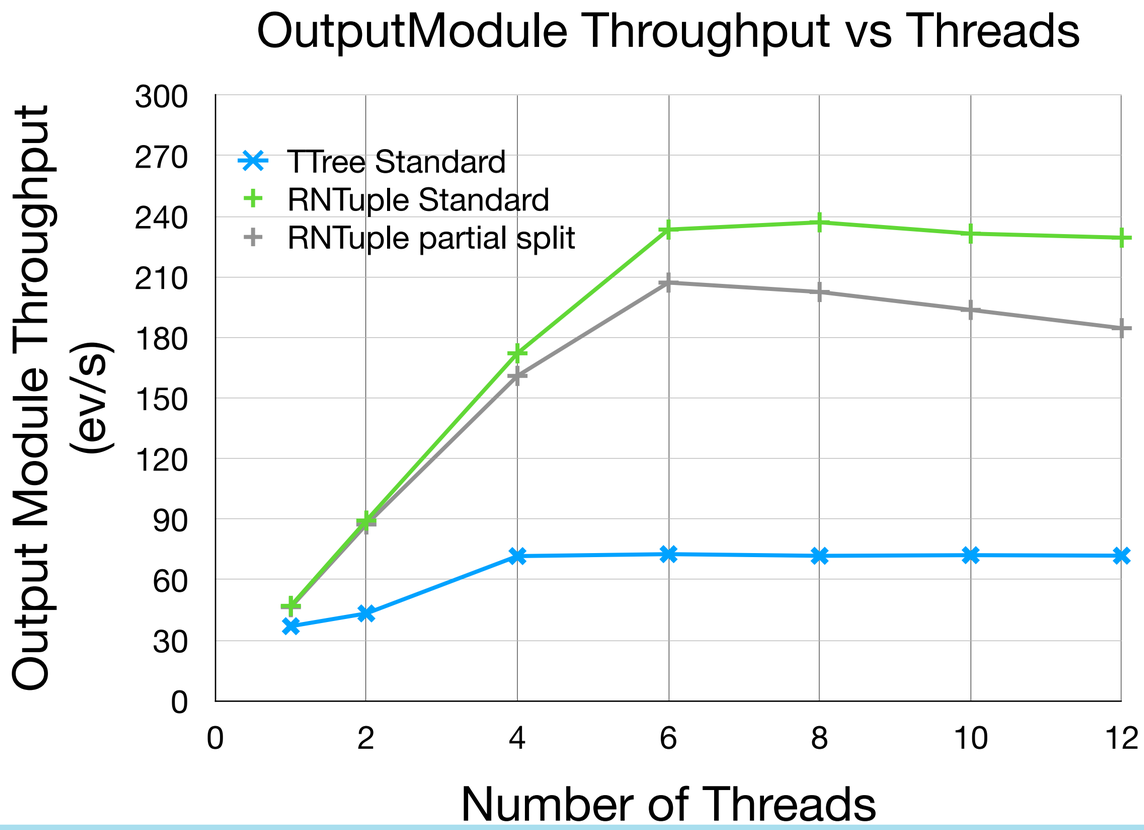
- Ran the copy jobs at different thread counts
  - Number of concurrent events always kept at 1
  - Use only time spent in the code that interacts with output file



Fermilab

Threading Performance Previous

- Throughput appears to be approximately the same



Fermilab

Max Memory Usage Old

- Monitor calls to new and delete
  - track amount of memory presently allocated and record the max for the job
- Run job with a dummy outputer to get baseline memory
- RNTuple output uses the partial split setting

Job	Max Memory	Output Overhead	Relative Overhead
Dummy	1.34GB	0GB	
TTree standard	2.33GB	0.99GB	1.00
RNTuple	4.27GB	2.94GB	**2.96**

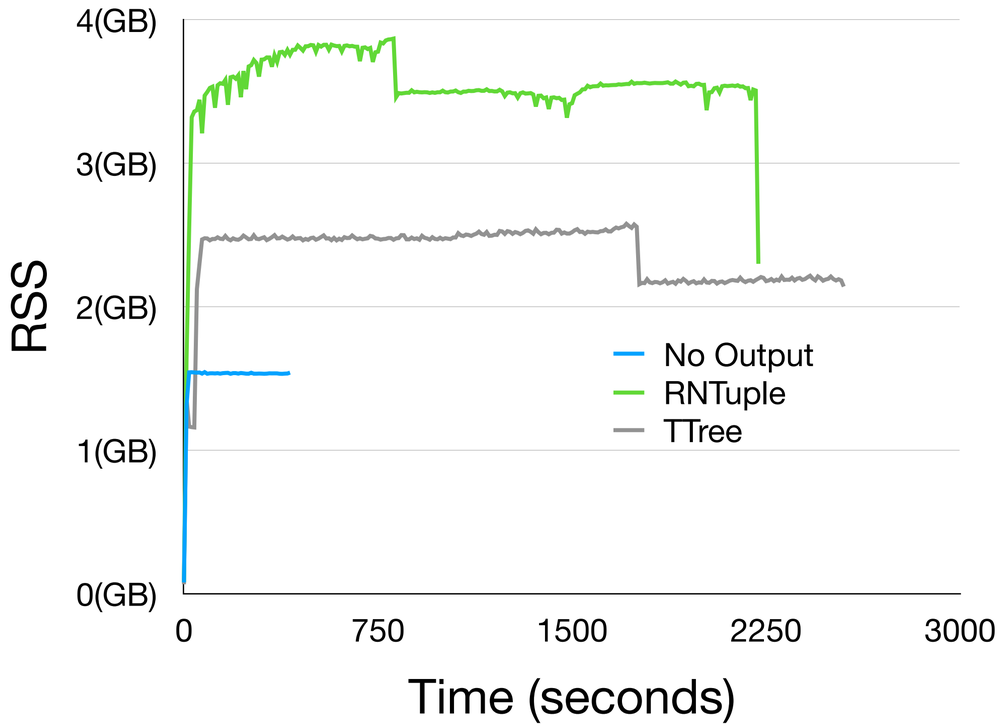
Max Memory Usage New

- Substantially less memory requested
  - decreased by 1.3GB

Job	Max Memory	Output Overhead	Relative Overhead
Dummy	1.34GB	0GB	
TTree standard	2.33GB	0.99GB	1.00
RNTuple	3.03GB	1.69GB	**1.70**

RSS Usage Old

- Have an external process which periodically samples RSS of running job

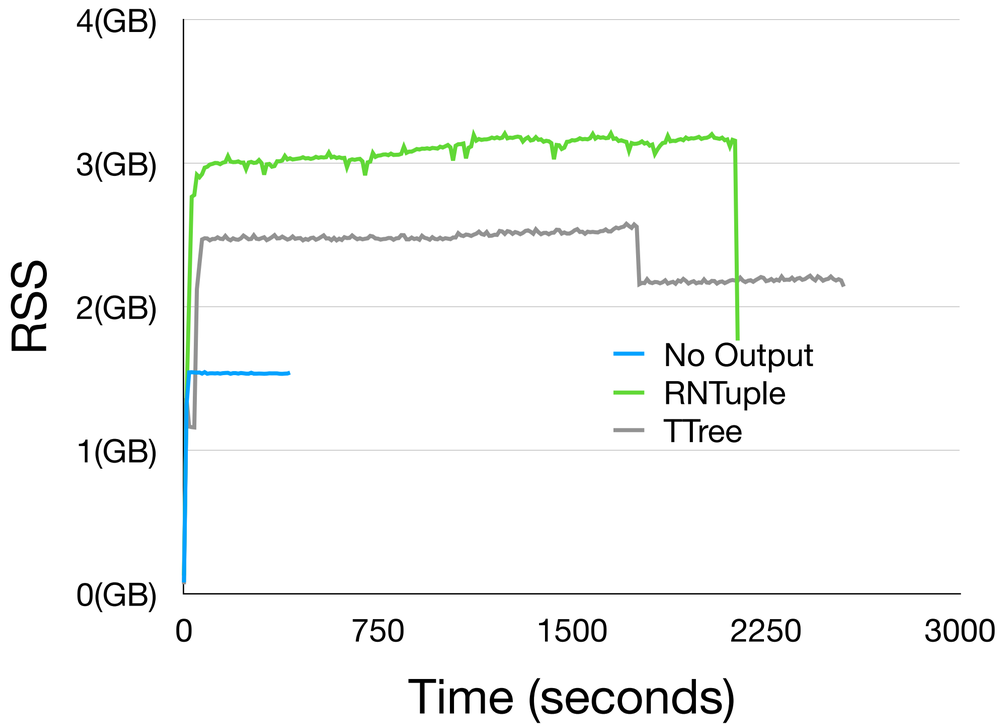


Job	Max RSS	Overhead
No Output	1.58GB	0GB
TTree	2.64GB	1.06GB
RNTuple	3.96GB	**2.38GB**

Fermilab

RSS Usage New

- Memory improvements also reflected in RSS usage



Job	Max RSS	Overhead
No Output	1.58GB	0GB
TTree	2.64GB	1.06GB
RNTuple	3.29GB	**1.7GB**

Fermilab