



RNTuple in LHCb

M. Clemencic *on behalf of LHCb Collaboration*

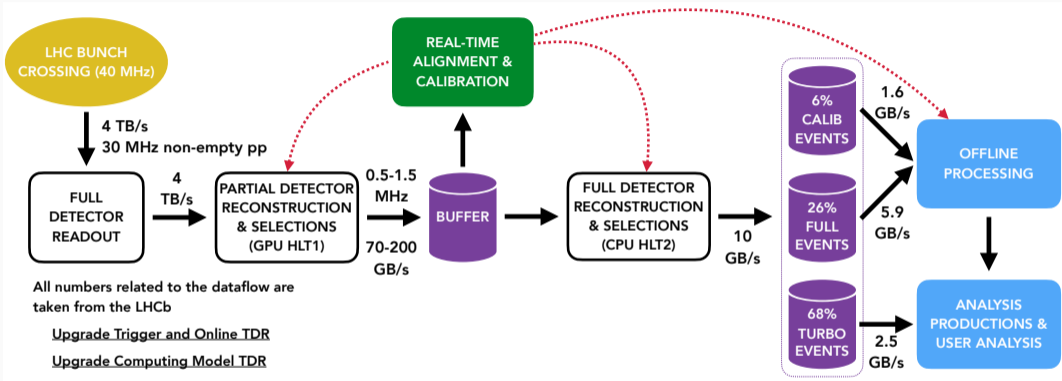
November 5, 2023

CERN - LHCb

1. Data flow in LHCb
2. Use of TTree in LHCb
3. Requirements for RNTuple

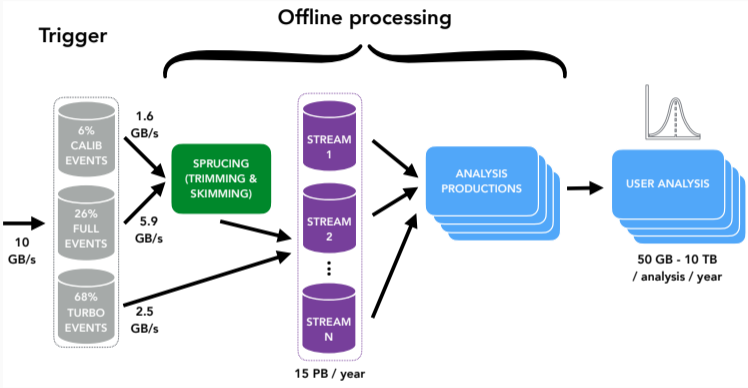
Data flow in LHCb

Data flow in LHCb: RTA



custom file format designed for streams of raw events

Data flow in LHCb: DPA



Sprucing and Analysis Productions write ROOT files

Use of TTree in LHCb

- 2 use cases for TTree
 - data streams (DSTs), output of Sprucing
 - n-tuples for user analysis, output of Analysis Production
- DST
 - one branch: the raw event BLOB
- n-tuples
 - mostly flat n-tuples
- User analysis implemented in Python

Requirements for RNTuple

- From the software framework point of view
 - LHCb software is multithreaded
 - I/O is lagging a bit behind
 - thread-safe n-tuple library is highly appreciated

Requirements for RNTuple: analysis

- From analysis jobs point of view
 - we mostly need flat n-tuples
 - nested branches are useful sometimes
 - standalone library with Python bindings
 - we rely a lot on uproot
 - uproot support allows for transparent migration
 - *friend trees* would be greatly appreciated

Conclusions

- LHCb does not have very complicated requirements on n-tuples
 - apart from *friend trees*, maybe
- Very strong need of standalone decoder
 - to support Python based analysis
- Use of TFile/TTree for sequences of raw events (BLOBs) is under discussion