Thread-safe N-tuple Writing in `Gaudi` with `TTree` and Migration to `RNTuple`

Silia Taider *on behalf of the LHCb Collaboration*
Supervised by Marco Clemencic

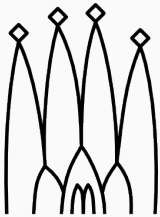
CHEP - 22 October 2024

CERN - LHCb - CPE Lyon

Gaudi

A cross-experiment **framework** and architecture for data processing for **HEP experiments**

- Multithreading support since 2013

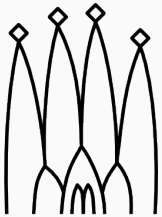


A cross-experiment **framework** and architecture for data processing for **HEP experiments**

- Multithreading support since 2013

But not everything in **Gaudi** is thread-safe ...

... in particular n-tuple writing



N-tuple writing, in Gaudi

- Gaudi uses TTree for writing n-tuples, which is not thread-safe

N-tuple writing, in Gaudi

- Gaudi uses TTree for writing n-tuples, which is not thread-safe
- The successor: RNTuple
  - Evolution of the currently used TTree columnar format
  - Naturally thread-safe interfaces
  - Ref.: ROOT RNTuple and EOS: The Next Generation of Event Data I/O
    Plenary session - 23 Oct 2024, 11:00

Thread-safety, in Gaudi

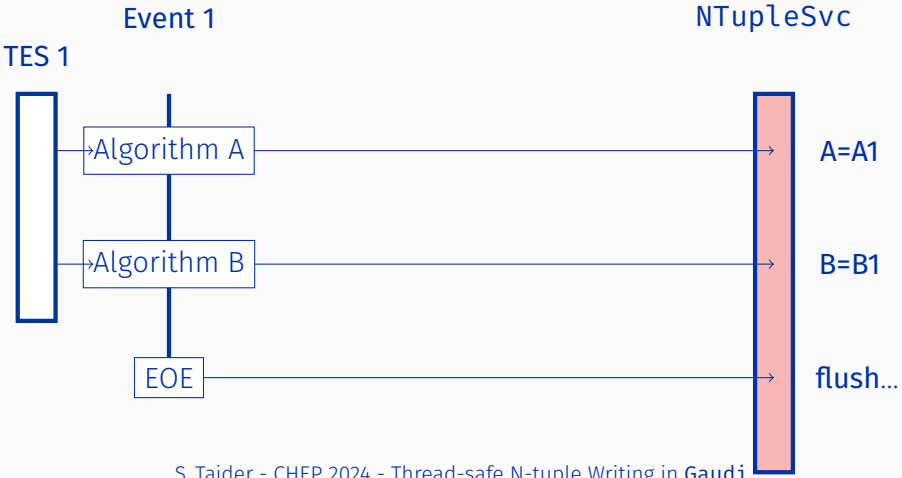
Moving from TTree to RNTuple will not be beneficial right away

- The Gaudi abstraction over TTree is not thread-safe

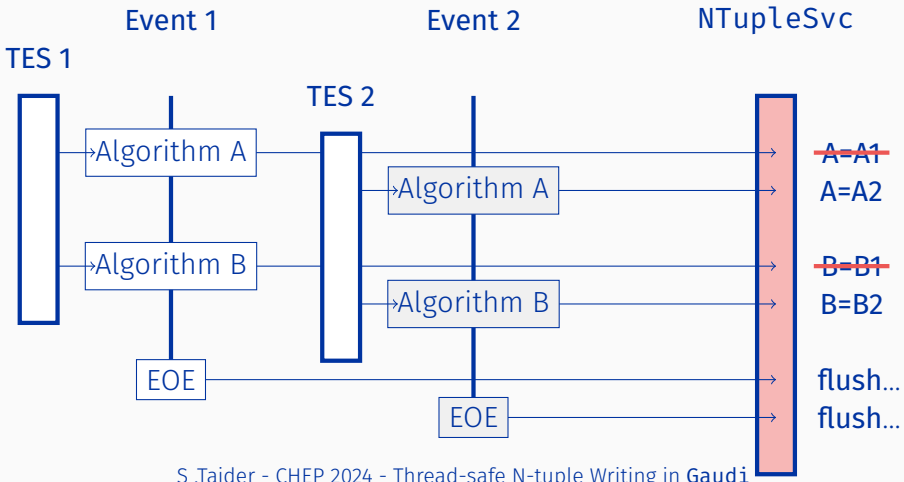
Table of contents

The NTupleSvc

The NTupleSvc



The NTupleSvc



The NTupleSvc

We have:

- One **shared** buffer that gets overwritten
- **Simultaneous** invocations of the tree-filling process

The NTupleSvc

We have:

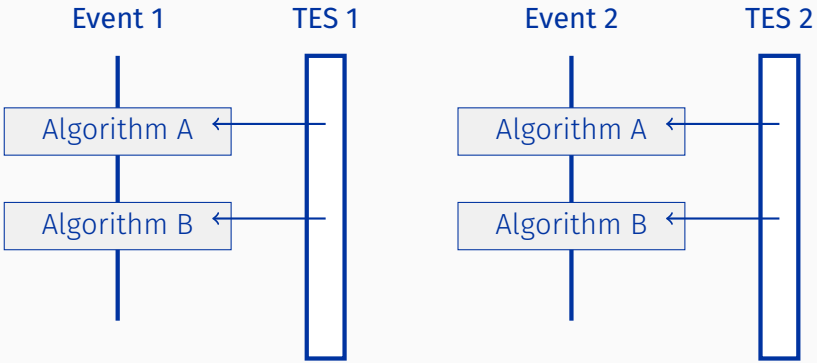
- One **shared** buffer that gets overwritten
- **Simultaneous** invocations of the tree-filling process

We need:

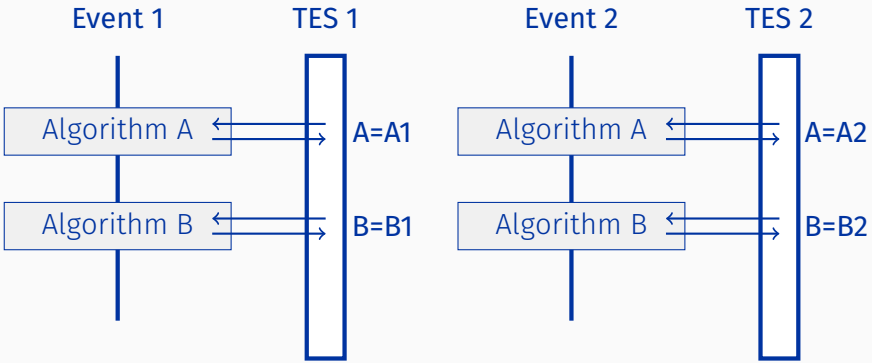
- One buffer **per event**
- **Synchronized** invocations of the tree-filling process

The NTupleWriter

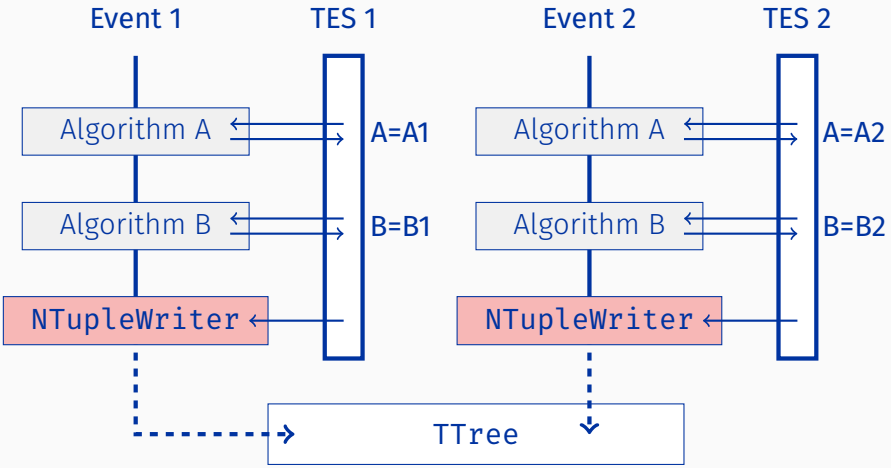
The NTupleWriter



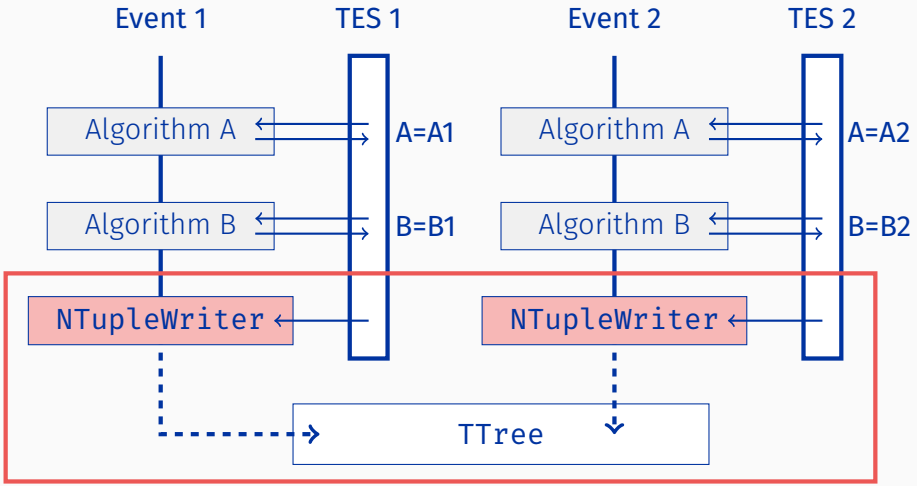
The NTupleWriter



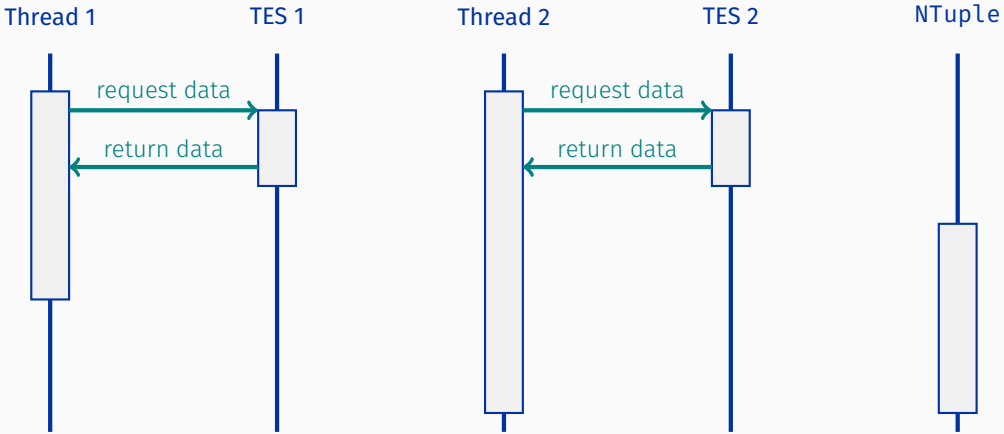
The NTupleWriter



The NTupleWriter

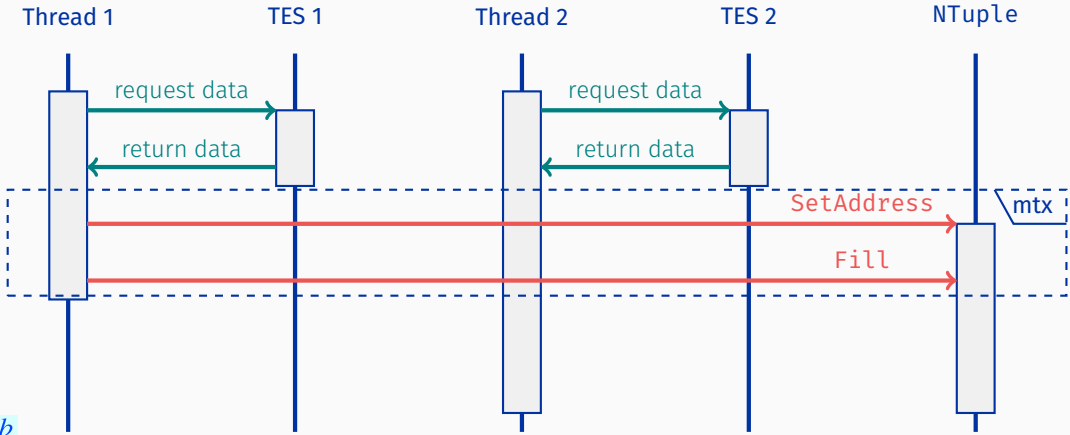


The NTupleWriter



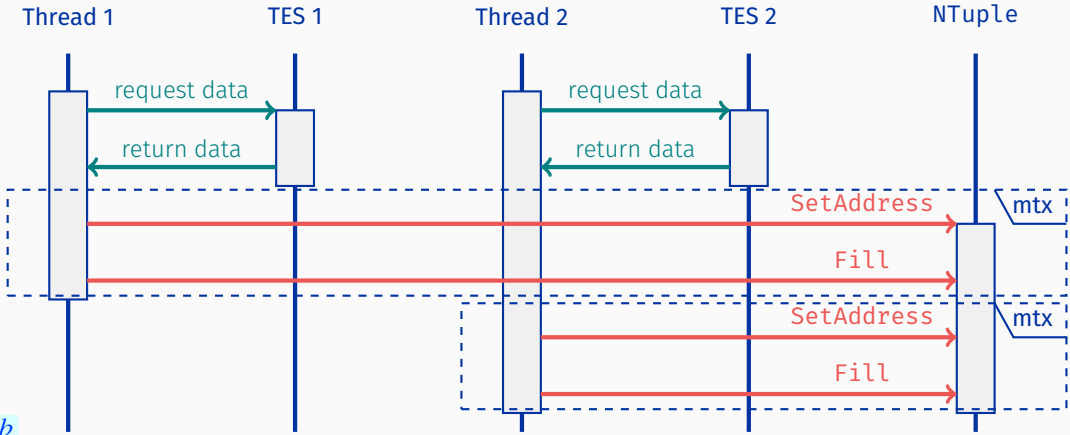
The NTupleWriter

Note: Red arrows indicate operations that need to be protected by a mutex



The NTupleWriter

Note: Red arrows indicate operations that need to be protected by a mutex



The NTupleWriter

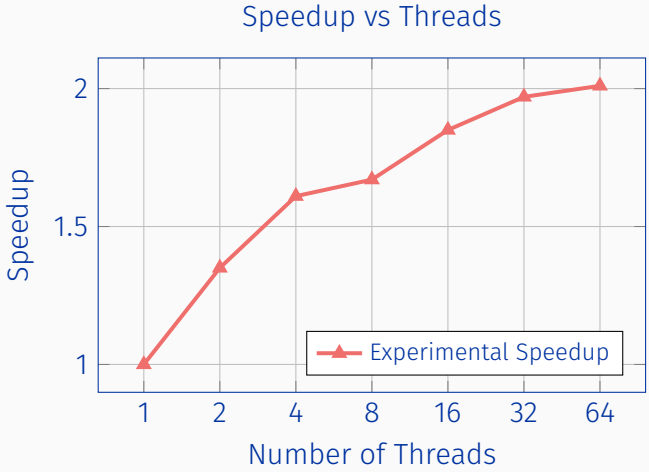
- Released in Gaudi v38
- Tested within Gaudi
- Currently being tested within DaVinci, LHCb's Analysis Framework

DaVinci

- The LHCb offline analysis application, built on top of **Gaudi**
- Produces the **tuples** in which the relevant information of the reconstructed particles of the decay of interest are stored

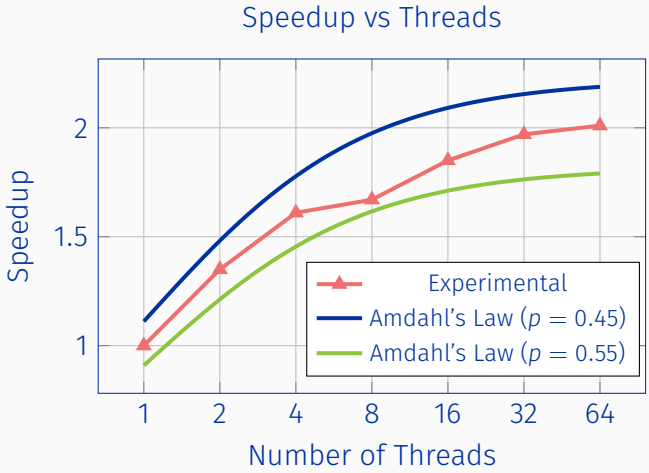
Benchmarks

Speedup of a representative DaVinci job comparing the `NTupleWriter` to a sequential run with the `NTupleSvc`



Benchmarks

Speedup of a representative DaVinci job comparing the `NTupleWriter` to a sequential run with the `NTupleSvc`



Benchmarks

Takeaway: We can run multithreaded jobs in `DaVinci`

The NTupleWriter

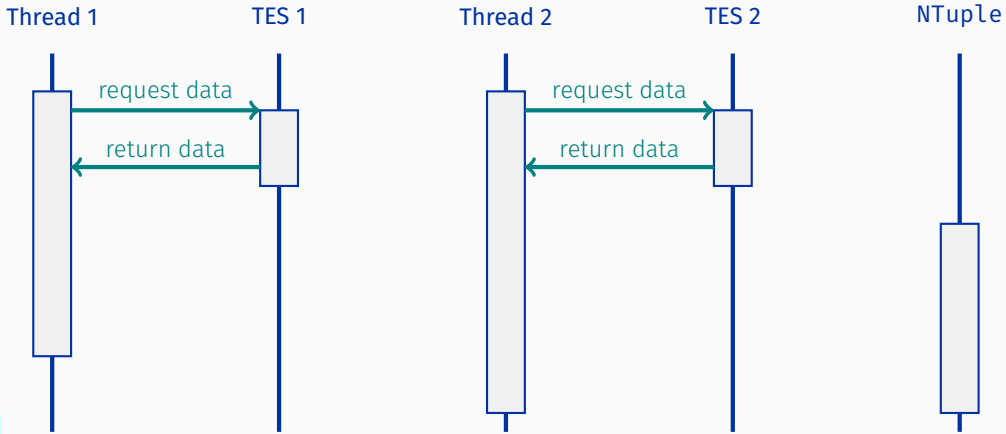
- A technology-independent Gaudi interface
- Encapsulated access to the storage backend
  - Maintainable
  - Configurable

The NTupleWriter

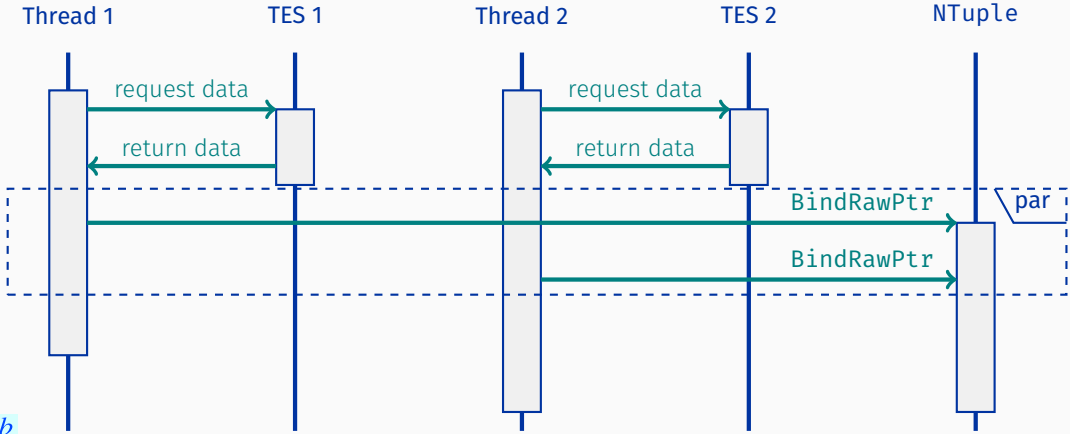
- A technology-independent Gaudi interface
- Encapsulated access to the storage backend
  - Maintainable
  - Configurable
  - **Interchangeable**

The RNTupleWriter

The RNTupleWriter

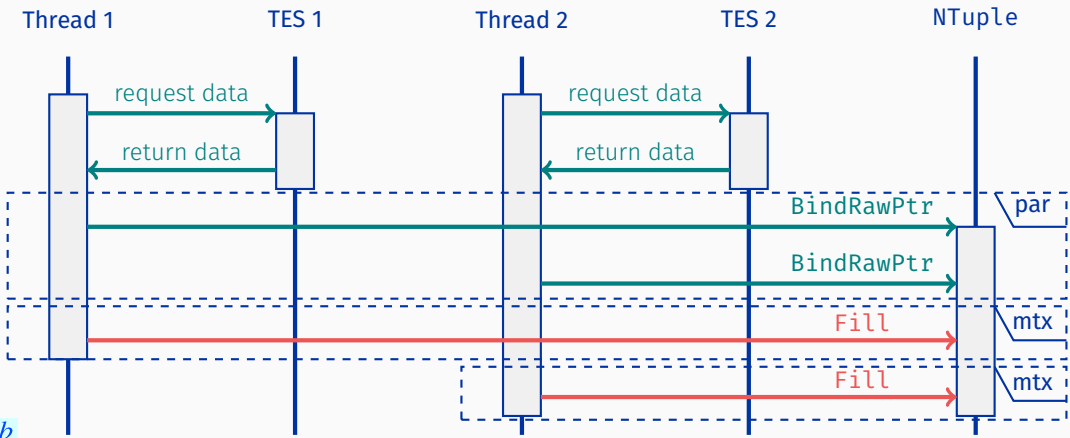


The RNTupleWriter



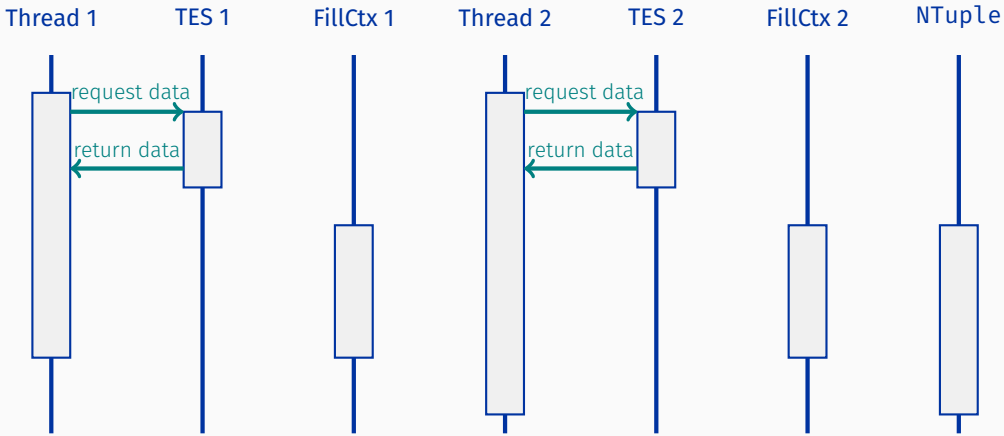
The RNTupleWriter

Note: Red arrows indicate operations that need to be protected by a mutex



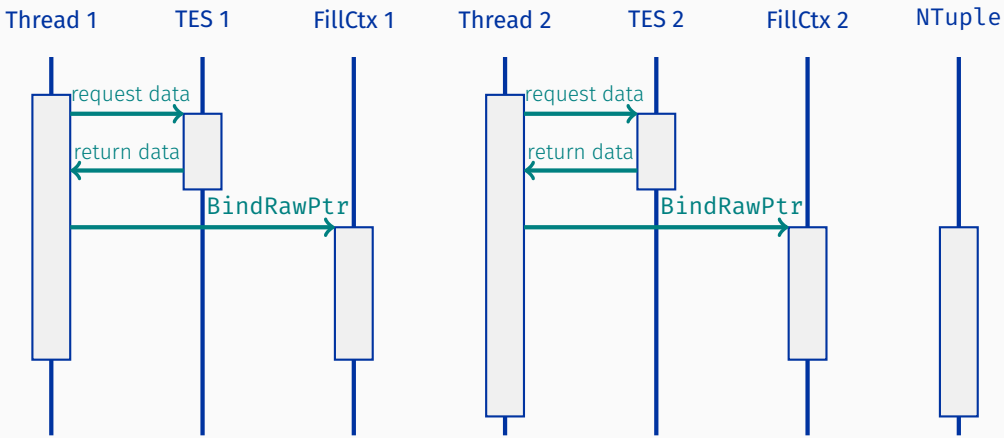
The RNTupleParallelWriter

Ref.: **Direct I/O for RNTuple Columnar Data** - Parallel Track 3 21 Oct 2024, 13:30



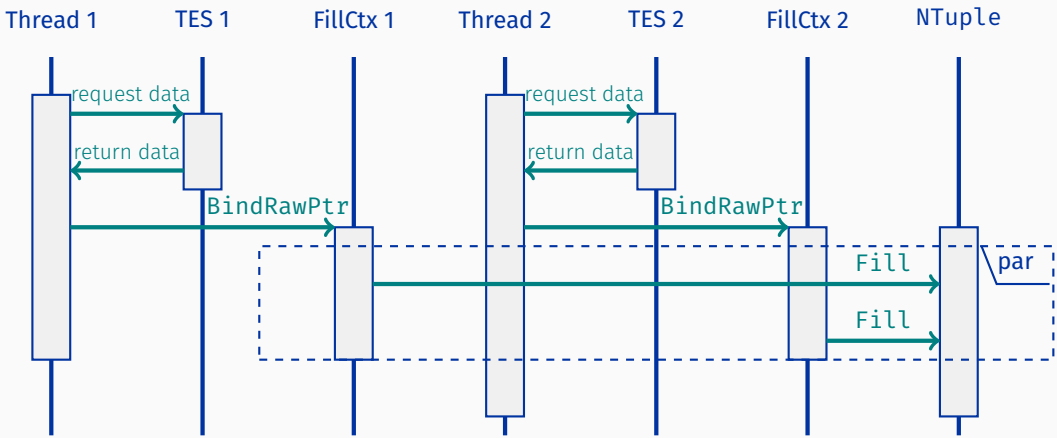
The RNTupleParallelWriter

Ref.: **Direct I/O for RNTuple Columnar Data** - Parallel Track 3 21 Oct 2024, 13:30



The RNTupleParallelWriter

Ref.: **Direct I/O for RNTuple Columnar Data** - Parallel Track 3 21 Oct 2024, 13:30



Migration to `RNTuple`

The migration was **straightforward**:

- Using the new interfaces $\rightarrow$ simpler & cleaner
- Reducing locks

Working prototype: on the dev version of `root`

Next Steps

Next Steps

- Commission the port of `DaVinci` to use the `NTupleWriter` and enable multithreaded jobs
- Integrate `RNTuple` support and plan its adoption as a persistency format

Questions