Demonstrating interactive, large-scale High Energy Physics data analysis workflows in distributed computing environments

EP-SFT Group

Andrea Ola Mejicanos

Supervisor: Marta Czurylo, Vincenzo Eduardo Padulano

12/08/24

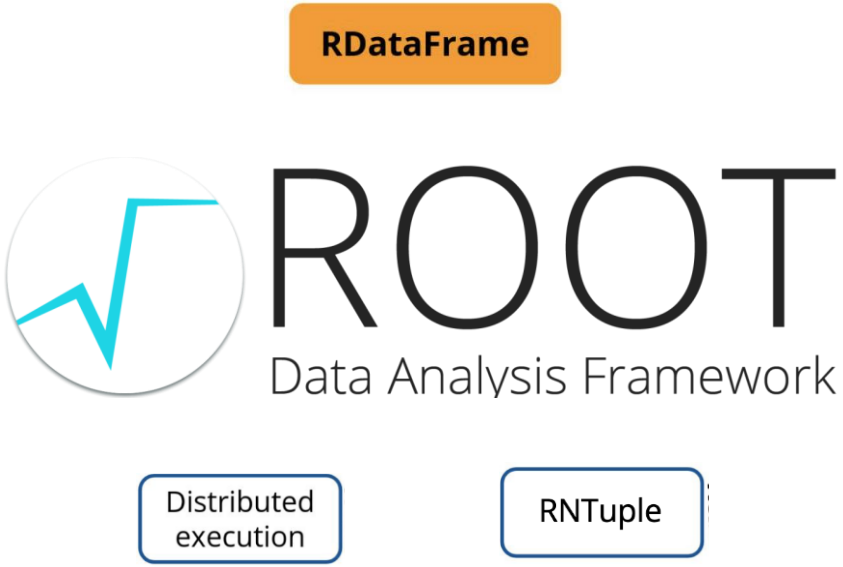
CERN openlab

Introduction

Large amounts of data are needed for new discoveries

HL-LHC will produce 30 times more data than the LHC has produced as of today.

Creates more computing needs and requires technological evolution

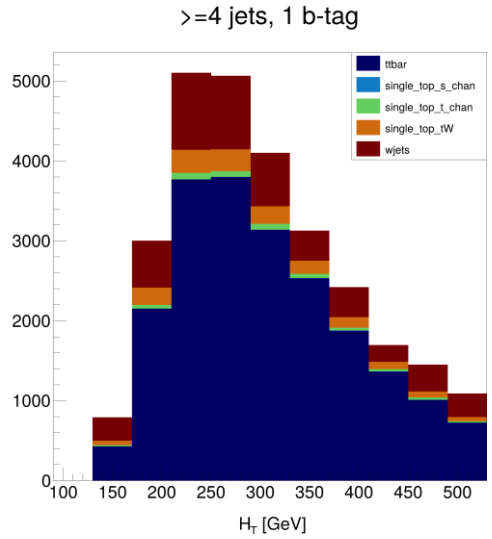


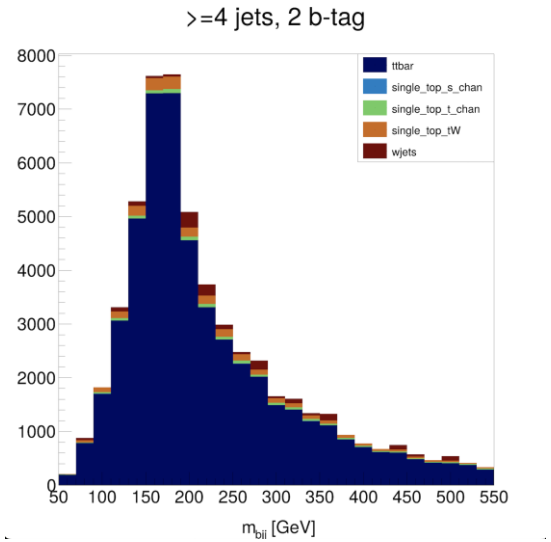
ROOT, including RDataFrame, is being upgraded to address new computing challenges

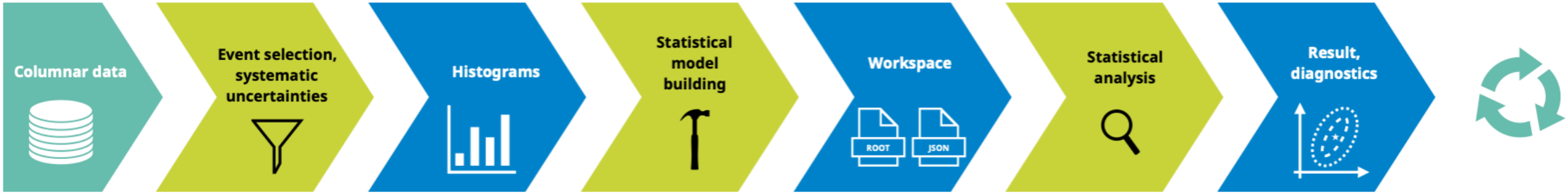
Analysis Grand Challenge

Our task:

- Test the performance of the distributed AGC RDataFrame implementation with TTree and RNTuple
- Validate the consistency of results
- AGC is a common open and reproducible analysis benchmark of different implementations





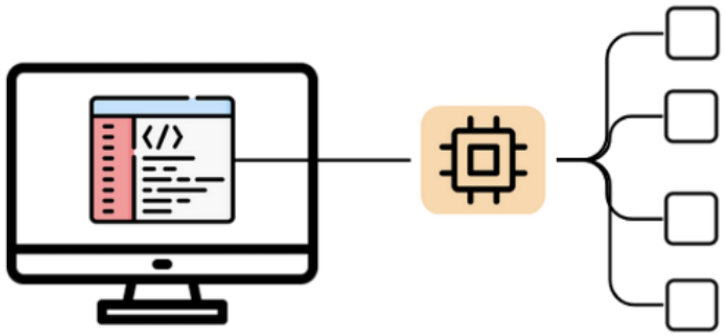


Multithreaded and Distributed RDF

Scaling tests on SWAN distributed environment via Dask

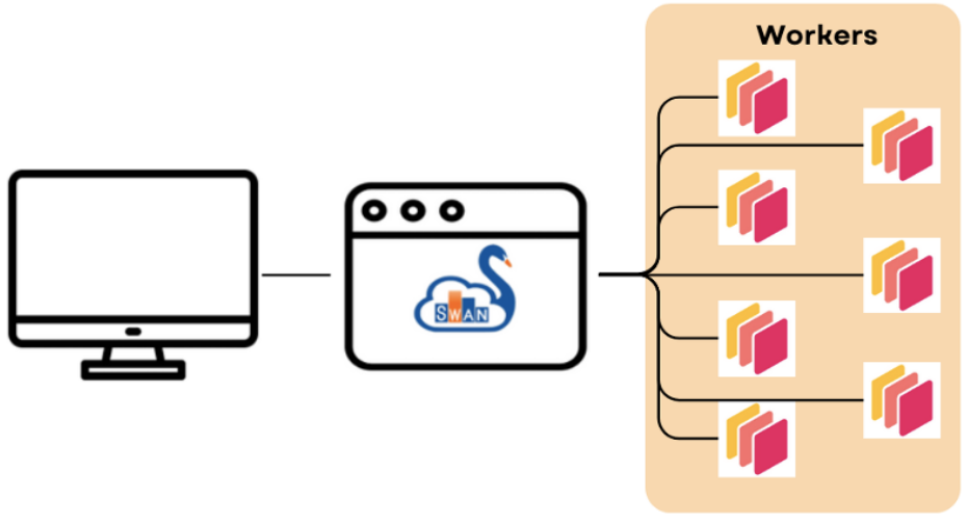
Multi-threaded RDF

```
import ROOT

ROOT.EnableImplicitMT(args.ncores)

df = ROOT.RDataFrame('treename', 'filename.root')

# Rest of the analysis code . . .
```

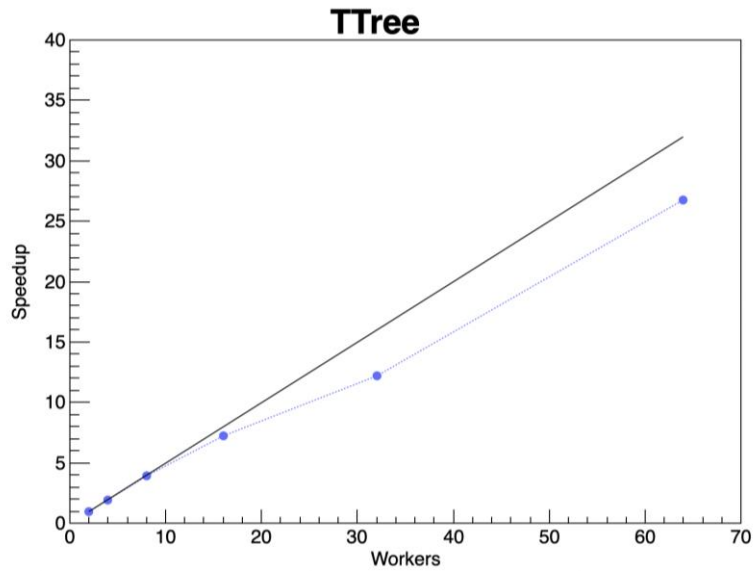


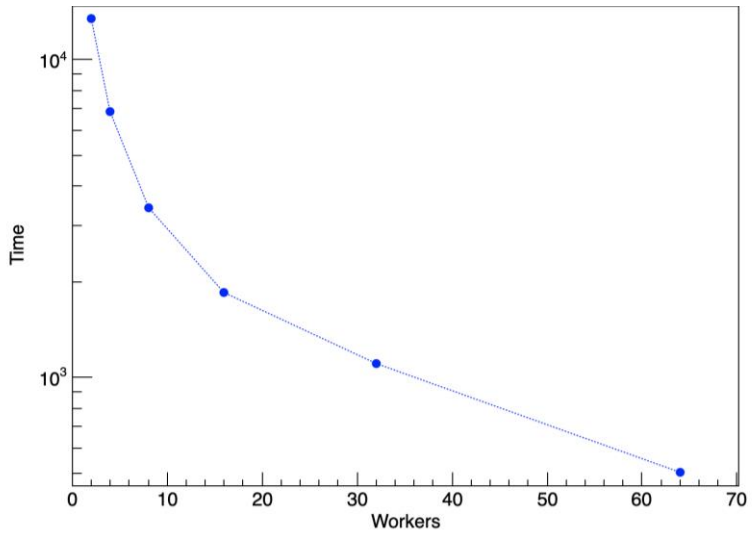
Distributed RDF

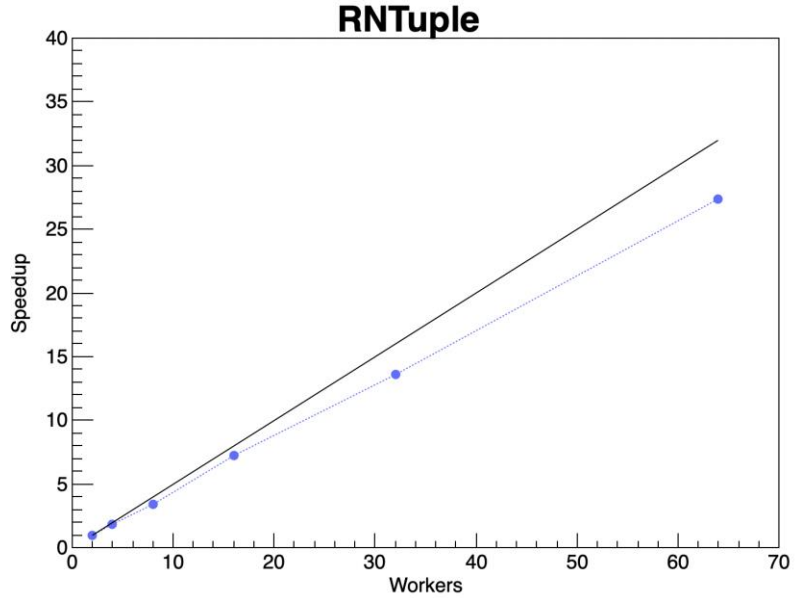
```
import ROOT

from distributed import Client

df = ROOT.RDF.Experimental.Distributed.Dask.RDataFrame
     ('treename','filename.root', daskclient =
     Client('tcp://hostname:port'))

# Rest of the analysis code . . .
```

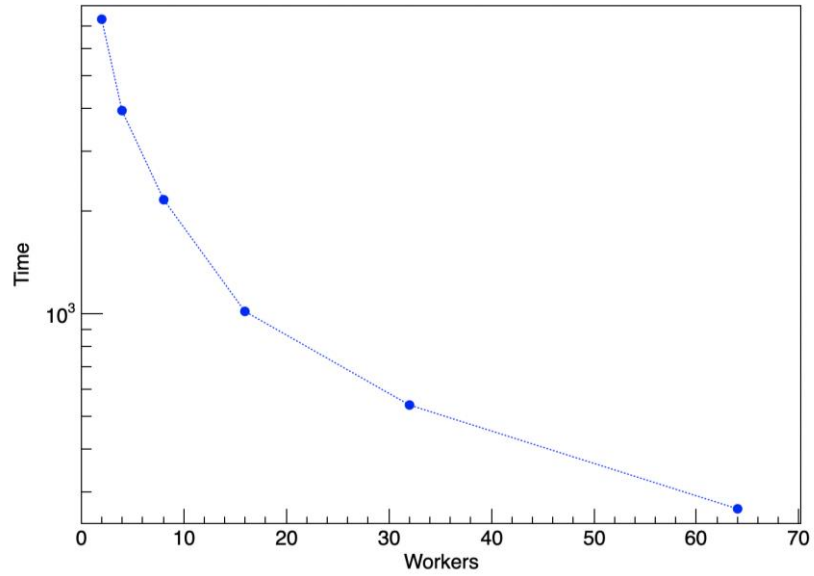


Benchmarks & validation









Validation shows perfect agreement between Distributed and Multithreaded modes!

Challenges

- Using user-installed packages in a distributed environment, specifically on CERN-SWAN
  - ✓ Learned how to packages are managed and distributed across multiple workers
- Navigating the SWAN interface can be problematic sometimes due to network issues.
  - ✓ Gained insight into navigating the SWAN interface.
- Identifying and resolving small bugs in the physics analysis.
  - ✓ Gained a broader knowledge of key physics analysis elements, such as handling of systematic uncertainties.
- Runtimes for lowest number of cores are long making the process time consuming.
  - ✓ Gained context of the challenges faced in the HEP community.

Future work

- Provide a demonstration analysis using the recently released 2016 CMS Open Data
- Perform memory profiling of distributed workers
- Optimize performance by tunning the data partitioning parameter

Thank you!

Questions?

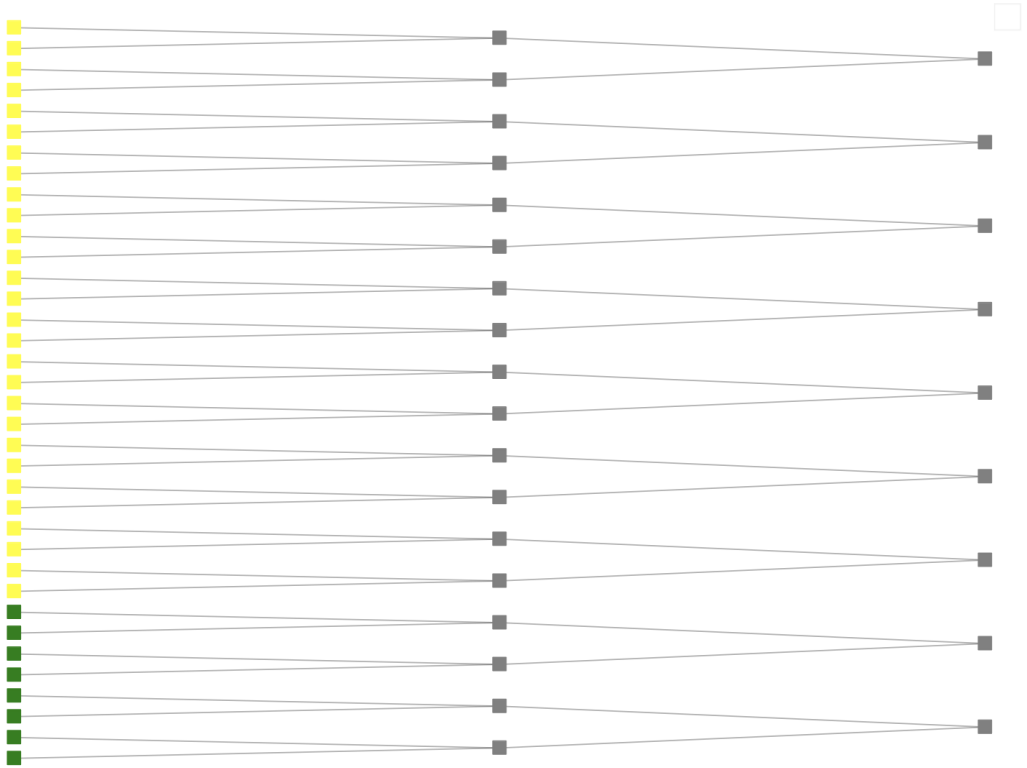
Email: olaa@berea.edu

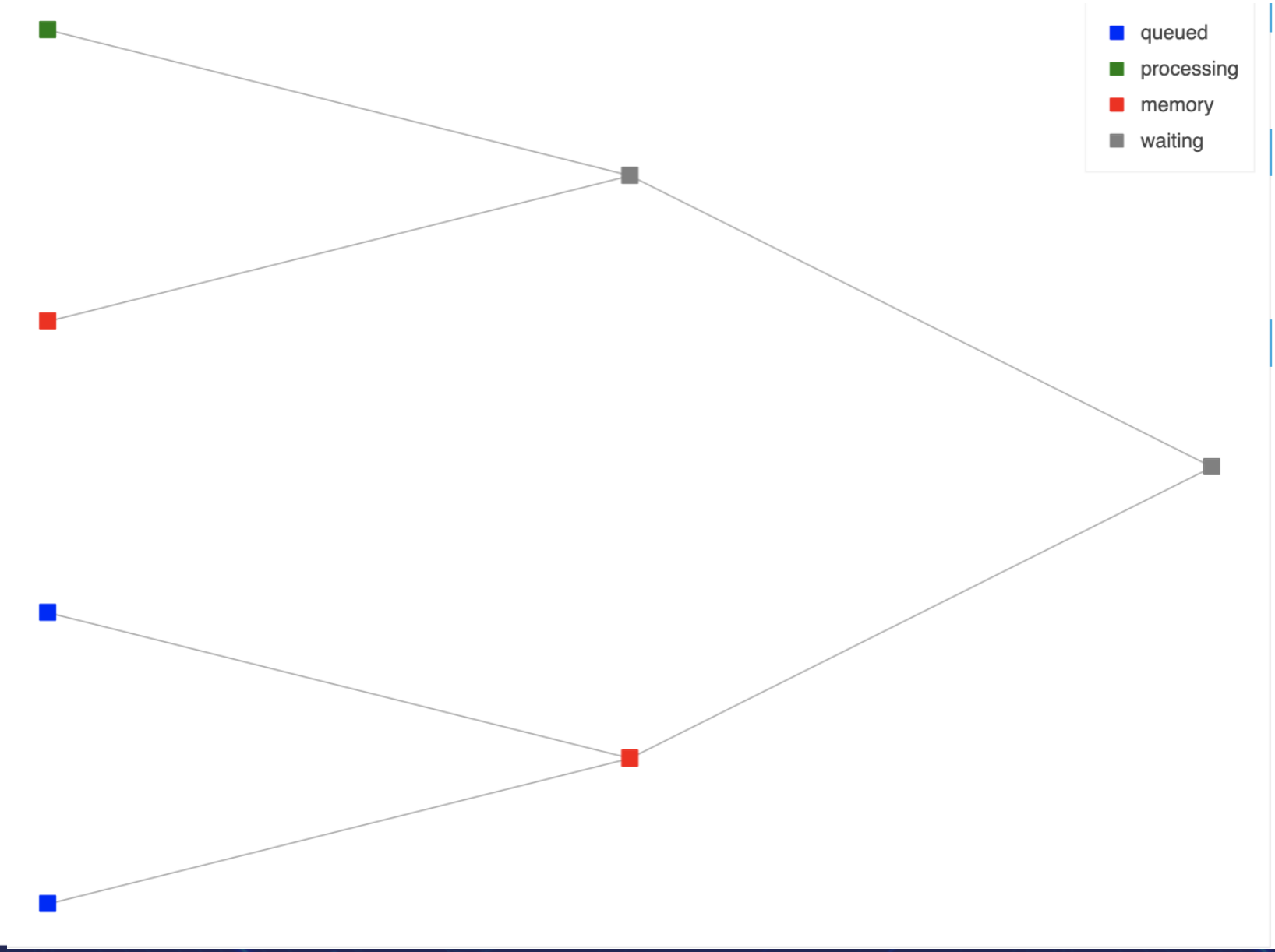
Github: https://github.com/andrea010901

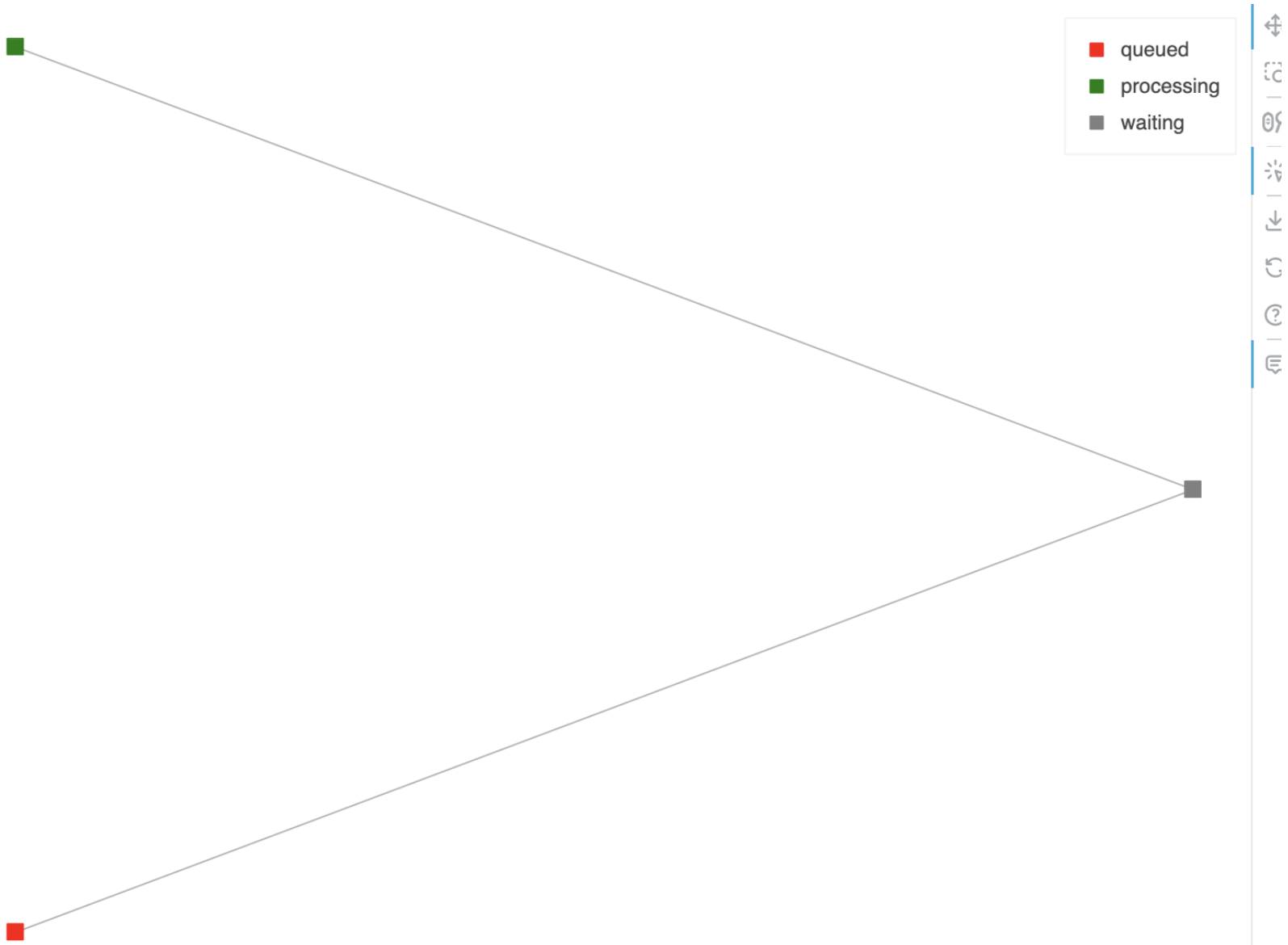
Backup

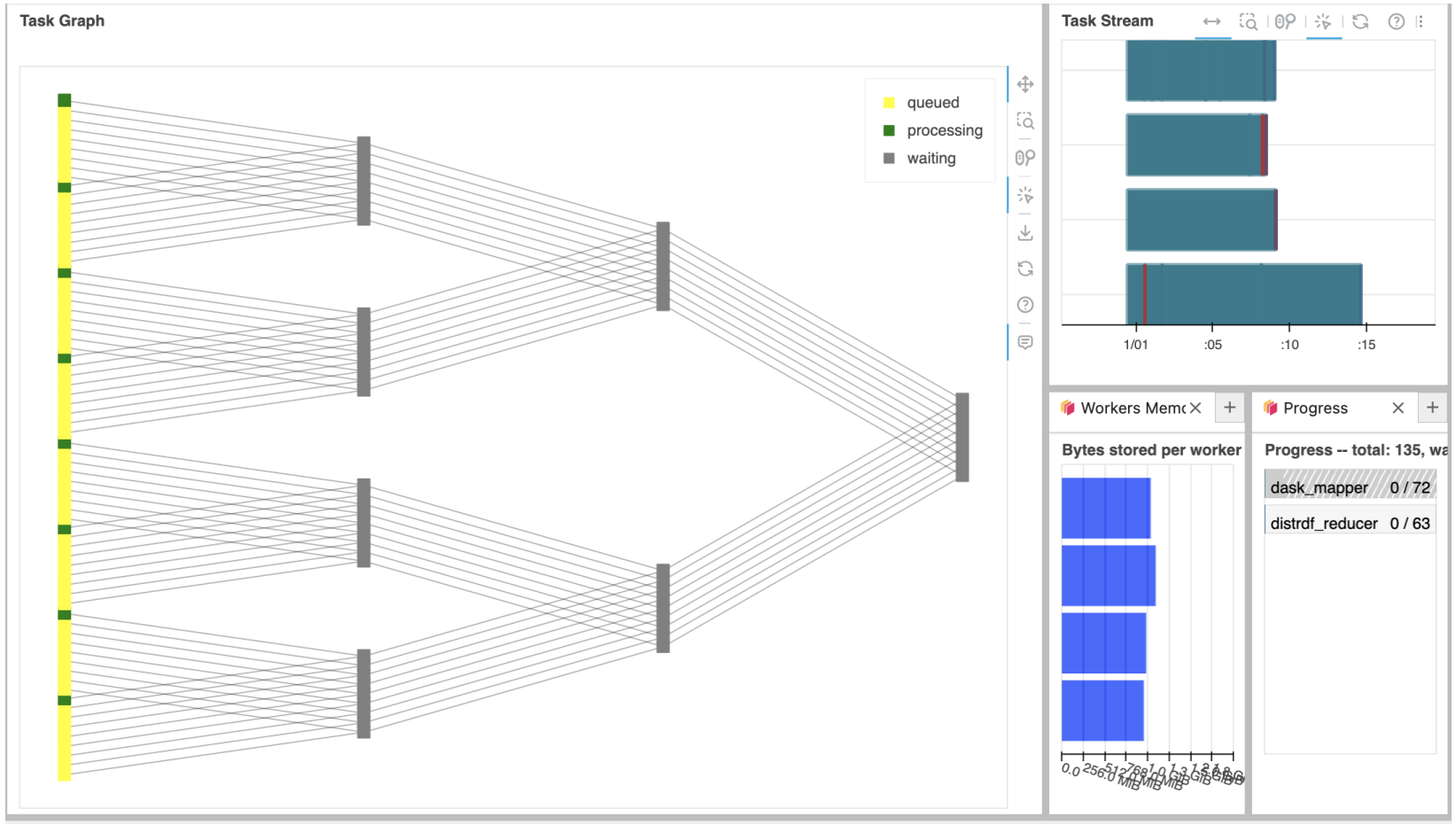
CERN openlab

Computation Graph




queued
processing
memory
waiting

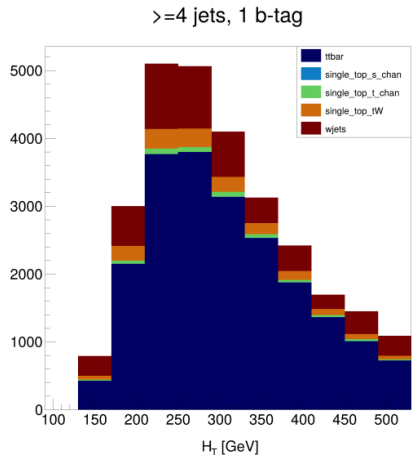

queued
processing
waiting


Task Graph
queued
processing
waiting
Task Stream
1/01
:05
:10
:15
Workers Memc
Bytes stored per worker
Progress
Progress -- total: 135, wa
dask_mapper 0 / 72
distrdf_reducer 0 / 63

Analysis Grand Challenge

Our task:

- Test the performance of the distributed AGC RDataFrame implementation with TTree and RNTuple
- Validate the consistency of results



AGC:

- AGC is a common open and reproducible analysis benchmark of different implementations
- $t\bar{t}$ cross-section measurement
- Represents a realistic analysis pipeline showing workflows envisioned for the HL-LHC including:
  - Event selection
  - Weighting
  - Systematics variations
  - Histograms production