RESEARCH UNIVERSITY
EXCELLENCE INITIATIVE

Akademia Górniczo-Hutnicza
im. Stanisława Staszica w Krakowie

AGH University of Krakow

Utilizing RDataFrame for Data Preservation and Open Publishing Data and Analysis Software for HEP

Paweł Kruczkiewicz[1] Kamil Burkiewicz[1] Leszek Grzanka[1]
Valentina Avati[1] Maciej Malawski[1]

[1]AGH University of Krakow

Section 1

Introduction

Motivation

- OpenData is a portal that publishes LHC data.
  - provides **data storage** but *does not* provide computational power
  - analyses and software along with the data → VMs, Dockers and CERN related software – **can be troublesome for the end user**
  - Who is the end user?:
    - people outside of CERN
    - little IT knowledge
    - education and outreach
  - http://opendata.cern.ch

Motivation - OpenData example

Figure: Example of an OpenData record. http://opendata.cern.ch/record/14014; available 05-03-24)

TOTEM

- *TOTal cross-section and Elastic scattering and diffraction dissociation Measurement*
- *Roman Pots* - movable detectors
- The data from 2011 along with:
  - **an article** – *Measurement of proton-proton elastic scattering and total cross-section at* $\sqrt{s} = 7TeV$, The TOTEM Collaboration *et al* 2013 EPL **101** 21002 https://iopscience.iop.org/article/10.1209/0295-5075/101/21002
  - **C++ scripts** made by Jan Kaspar, one of TOTEM Collaboration members.

Section 2

Tools for data processing at CERN

Experiment-centered software

- *CMS Software (CMSSW)*, *DaVinci* for LHCb, *ATLAS Software*, and *ALICE O2*
- requiring users to overcome a **steep learning curve** to effectively utilize them
- **pitfalls**: difficult configuration, version compatibility, and potential errors during installation
- a primary method of publishing data on CERN Open Data Portal

ROOT

- primary framework for data analysis at CERN and beyond
- functionalities: statistical description, histogram drawing
- active **ROOT forum** – accessibility for users at all levels
- **PyROOT** – official Python extension
- **RDataFrame** – declarative programming, partial-column read

Serverless Engine for HEP

- **Serverless computing** – function based dynamic management of resources
- suitable for event-driven analyses of HEP
- *A Serverless Engine for High Energy Physics Distributed Analysis* 2022 22nd IEEE International Symposium on Cluster, Cloud and Internet Computing (CCGrid), Taormina, Italy, 2022, pp. 575-584, doi: 10.1109/CCGrid54584.2022.00067. – Kusnierz et al.
  - RDataFrame, AWS Lambda, AWS S3 and EOS
  - promising results in terms of scalability and processing time reduction
  - experimental phase

AGH

Section 3

Data preservation process

From archives

- Data stored on CERN Tape Archive (CTA)
- It can be requested and downloaded within a day.
- Downloading ∼260 GB of data in 42 files
- *ntuple* - a tree like structure where branches have their own list of columns
- Used custom Python scripts for automation

Data Exploration

- Describing the ntuples with RDataFrame
- 3 different Roman Pots' setup [fig.2]
- ∼12.5 million entries
- ∼1500 columns
  - many of those columns did not contribute directly to the original *Measurement of proton-proton...*[1] paper
  - in the analysis we focus only on a small fraction of those columns

[1] https://iopscience.iop.org/article/10.1209/0295-5075/101/21002

Data Exploration

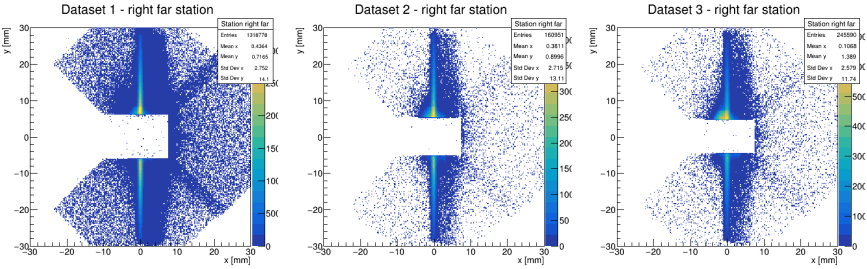


Figure: Hit points of the local tracks on Roman Pots in the right far station from 3 different datasets. Note the difference in width of the middle gap

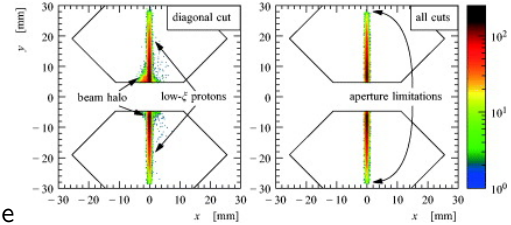
Translation to RDataFrame

- Most challenging and demanding part.
- Iterative process with the TOTEM collaboration
- Reproduction of histograms from the *Measurement of proton-proton...*[2] paper [see figure 3]
- Greater readability [see figure 4]
  - Vast code reduction
  - No *event loop* which leads to **less nesting**
  - Declarative programming in a form of **plain mathematical equations**

[2] https://iopscience.iop.org/article/10.1209/0295-5075/101/21002

Translation to RDataFrame – Track distribution

From the article



Reconstructed

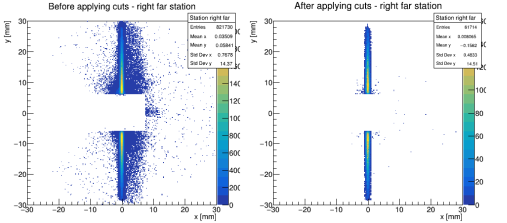


Figure: Comparison between histograms presented in the original *Measurement of proton-proton...* article and reconstructed

Transferring to RDataFrame – code snippet

Before

```cpp
// load files
TFile *inF = new TFile((string("distill_") + argv[1] + ".root").c_str());
(...)

// event loop
for (int ev_idx = 0; ev_idx < inT->GetEntries(); ++ev_idx) {
    inT->GetEntry(ev_idx);
    (...)

// cut evaluation
(...)
for (map<unsigned int, double>::iterator cit = csi.begin(); cit != csi.end(); ++cit) {
    unsigned ci = cit->first;
    cv[ci] = cca[ci]*cqa[ci] + ccb[ci]*cqb[ci] + ccc[ci];
    ct[ci] = (fabs(cv[ci]) <= n_si * csi[ci]);
}
(...)
```

After

```python
### Loading the data
totem_data = ROOT.ROOT.RDataFrame("TotemNtuple", "tuple_with_optics.root")

### Appplying
elastic_data = totem_data.Filter( f"abs(th_y_R - th_y_L) < 3 * 3.5E-6", "Cut 2")\
                         .Filter(f"abs(vtx_x_R) < 3 * 0.2")\
                         .Filter(f"abs(vtx_x_L) < 3 * 0.2")
```

Figure: Code in C++ and its counterpart in RDataFrame

OpenData

- (bibliographic) **record** - single page on OpenData Portal consisting of author, title, description, files, unique DOI number etc.
- The outcome – **2 records**:
  1. Dataset – skimmed ntuple
  2. Analysis – Jupyter Notebook

OpenData – Dataset Record

- Skimmed ntuples (table 1)
  - **Hit distributions**
  - **Metadata**
  - **Kinematics**
- Reduction of size due to better compression and refined number of columns

Name	Original	Skimmed
Size [GBs]	260	less than 1
Ntuple files	42	1
Number of columns	around 1500	56

Table: Comparison of original and skimmed ntuples

OpenData – Analysis Record

- Exemplary usage of the data
- Includes histograms and selections from the original *Measurement of proton-proton...*[3] analysis
- Descriptions of presented research
- Technological stack of Python, PyROOT (RDataframe), Jupyter Notebook
  - Common among Data Analysts
  - Easy to download and set up (`root --notebook` in the console)
  - **No need for Virtual Machines, Dockers, or specialized software**

[3] https://iopscience.iop.org/article/10.1209/0295-5075/101/21002

Performance optimisations

- We have tested if the skimmed ntuple results in reduced execution time of the notebook.
- The tests compare the original (260 GB) dataset with the skimmed one (< 1 GB).
- In the test we perform *cuts* - semantic filtering of the data that the physicist do at this point of data processing and drawing 5 histograms.
- Conducted on a personal laptop.
- Skimmed ntuples give three time shorter execution time compared to the original dataset (See fig 5).
- The original dataset is still comparatevely fast due to the partial read of the RDataFrame.

Performance optimisations – time measurements

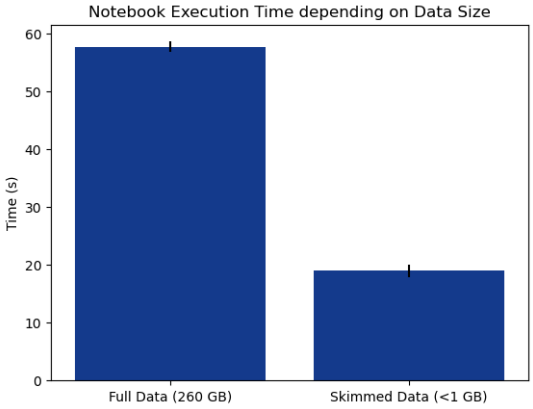


Figure: Comparison between execution time of the notebook with regards to the used dataset. Using the skimmed ntuples reduces this metric by three.

Conclusions

- **Collaborative Restoration**: Efforts to restore archived data foster accessibility in HEP.
- **Technological Access**: Innovative tools enable LHC data processing on personal desktops, broadening HEP research access.
- **Cloud Computing Potential**: Current form of the data and software enables us to leverage the power of serverless computing which will be further researched in the future.

Acknowledgements

The project was partially funded by the Polish Ministry of Education and Science, project 2022/WK/14.

AGH

www.agh.edu.pl

Thank you!