UNIVERSITY OF LIVERPOOL

QuMAP
Quantum Materials and Astroparticle Physics

Artwork by Inez Froehlich

Inspiration for data management practices from large collaborations

Tetiana ("Tania") Kozynets

QUEST-DMC Collaboration Meeting
17 October 2025

Outline

- Data storage and access
- Data (and simulation!) file format
- Hierarchical / staged data (and simulation!) processing
- Action items

Data storage and access

- Ideally, would have a dedicated resource allocation (both storage & computing) for QUEST – on a shared HPC cluster

  - All collaboration members should be able to easily request an account (perhaps through their PI)
  - Data and simulation would be stored centrally on the cluster, with read access to everyone
    - Write access for the selected few (who other produce new MC or collect data)

Shared HPC resources – what are our options?

- On Thursday, Mark mentioned the great Finnish resources (e.g. LUMI) – can we request resources as a collaboration?
  - EU vs non-EU policies?

- Other options through **EuroHPC**?
  - Continuous call for proposals – the next deadline is too soon (27/10), but perhaps could make the one after?

CALL STATUS: OPEN

EuroHPC JU Call for Proposals for Extreme Scale Access Mode

The call is open to all fields of science, industry and public sector justifying the need for and the capacity to use extremely large allocations in terms of compute time, data storage and support resources.

#EuroHPC Joint Undertaking

The European High Performance Computing Joint Undertaking (EuroHPC JU) pools together resources of the European Union (EU), European countries and private partners to develop a world class supercomputing ecosystem in Europe, boosting European competitiveness, innovation and improving European citizens' quality of life.

Member countries are Albania, Austria, Belgium, Bulgaria, Croatia, Cyprus, Czechia, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Israel, Italy, Latvia, Lithuania, Luxembourg, Malta, Moldova, Montenegro, Netherlands, North Macedonia, Norway, Poland, Portugal, Romania, Serbia, Slovakia, Slovenia, Spain, Sweden, Türkiye, United Kingdom.

EuroHPC Joint Undertaking

UK is included ✓

Data storage & file format

- Dealing with multidimensional time series data – want to extract the quantities of interest in the time window of interest

  - At **high level**, could have a few large .hdf5 files with the following columns:

Unique run/ subrun ID	`t_start`	`t_end`	Quantity X recorded	Quantity Y recorded	…	Is this a calibration/ “special conditions” run
int	float (UNIX timestamp)	float (UNIX timestamp)	True/False (bool), units (str), summary stats	True/False (bool), units (str), summary stats		True/False (bool), type of run (str)

Data storage & file format

- Dealing with multidimensional time series data – want to extract the quantities of interest in the time window of interest

  - At **high level**, could have a few large .hdf5 files with the following columns:

Unique run/ subrun ID	`t_start`	`t_end`	Quantity X recorded	Quantity Y recorded	…	Is this a calibration/ "special conditions" run
int	float (UNIX timestamp)	float (UNIX timestamp)	True/False (bool), units (str), summary stats	True/False (bool), units (str), summary stats		True/False (bool), type of run (str)

- To come up with the numbering scheme / short str flags for runs, need a list of all possible "types of runs" / conditions so we can easily classify them. Let's make such a list!
- E.g. **1**XX**0**YY could be all tracking at RHUL, **2**XX**1**YY could be all calibration at Lancaster

Data storage & file format

- Dealing with multidimensional time series data – want to extract the quantities of interest in the time window of interest

  - At **high level**, could have a few large .hdf5 files with the following columns:

Unique run/ subrun ID	`t_start`	`t_end`	**Quantity X** recorded	**Quantity Y** recorded	…	Is this a calibration/ “special conditions” run
int	float (UNIX timestamp)	float (UNIX timestamp)	True/False (bool), units (str), summary stats	True/False (bool), units (str), summary stats		True/False (bool), type of run (str)

- Similarly, want a comprehensive list of the main quantities of interest to high-level analyses – let’s make such a list in the coming weeks. A bool could indicate whether a measurement for this quantity is available

Data storage & file format

- Dealing with multidimensional time series data – want to extract the quantities of interest in the time window of interest

  - At **high level**, could have a few large .hdf5 files with the following columns:

Unique run/ subrun ID	`t_start`	`t_end`	Quantity X recorded	Quantity Y recorded	…	Is this a calibration/ “special conditions” run
int	float (UNIX timestamp)	float (UNIX timestamp)	True/False (bool), units (str), **summary stats**	True/False (bool), units (str), **summary stats**		True/False (bool), type of run (str)

min/max, median, some other quantiles?, std of the measured quantity for high-level analyses/quick lookup

Data storage & file format

- Dealing with multidimensional time series data – want to extract the quantities of interest in the time window of interest

  - At **high level**, could have a few large .hdf5 files with the following columns:

Unique run/ subrun ID	`t_start`	`t_end`	Quantity X recorded	Quantity Y recorded	…	Is this a calibration/ “special conditions” run
int	float (UNIX timestamp)	float (UNIX timestamp)	True/False (bool), units (str), summary stats	True/False (bool), units (str), summary stats		True/False (bool), type of run (str)

  - At **low level**, look up the entire time series between `t_start` and `t_end` from smaller compressed files with unique run ID in the file name
    - No need to duplicate other metadata in the time series as available from high-level files

Simulation storage & file format

- In the long run, want to achieve a state where our simulation files (signal + background + noise, from particle generation to readout) come in the same format as data files

Simulation storage & file format

- In the long run, want to achieve a state where our simulation files (signal + background + noise, from particle generation to readout) come in the same format as data files

- Based on a number of physics conditions ($p, B, T_0 \ldots$) and systematic parameters (BG rates for each component, CR flux, thickness of detector components etc…),
  want to be able to simulate the expected temperature time series (*"end-to-end"*)

Simulation storage & file format

- In the long run, want to achieve a state where our simulation files (signal + background + noise, from particle generation to readout) come in the same format as data files

- Based on a number of physics conditions ($p, B, T_0 \ldots$) and systematic parameters (BG rates for each component, CR flux, thickness of detector components etc…),
  want to be able to simulate the expected temperature time series (***"end-to-end"***)

  - Store in the same format as data (many low-level compressed files + high-level hdf5),
    but with additional truth information about what is simulated
  - **Aim at "nominal" simulation (nominal values for each systematic) + simulation sets where each parameter is offset by +- X%**

Simulation storage & file format

- In the long run, want to achieve a state where our simulation files (signal + background + noise, from particle generation to readout) come in the same format as data files

- Based on a number of physics conditions ($p, B, T_0 \ldots$) and systematic parameters (BG rates for each component, CR flux, thickness of detector components etc…),
  want to be able to simulate the expected temperature time series (*"end-to-end"*)

- While the simulation is still in active development, we can at least **number and version everything**, and make all produced MC accessible to everyone

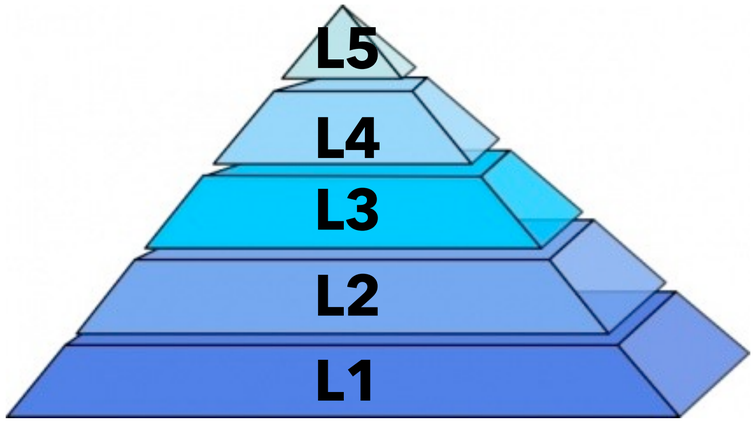
Hierarchical (staged) data processing

- People involved in different types of analyses might need different levels of fidelity / pre-processing to make their life easier.
  - Raw vs cleaned time series
  - Peaks removed vs baseline removed
  - …
  - …

Hierarchical (staged) data processing

- People involved in different types of analyses might need different levels of fidelity / pre-processing to make their life easier.

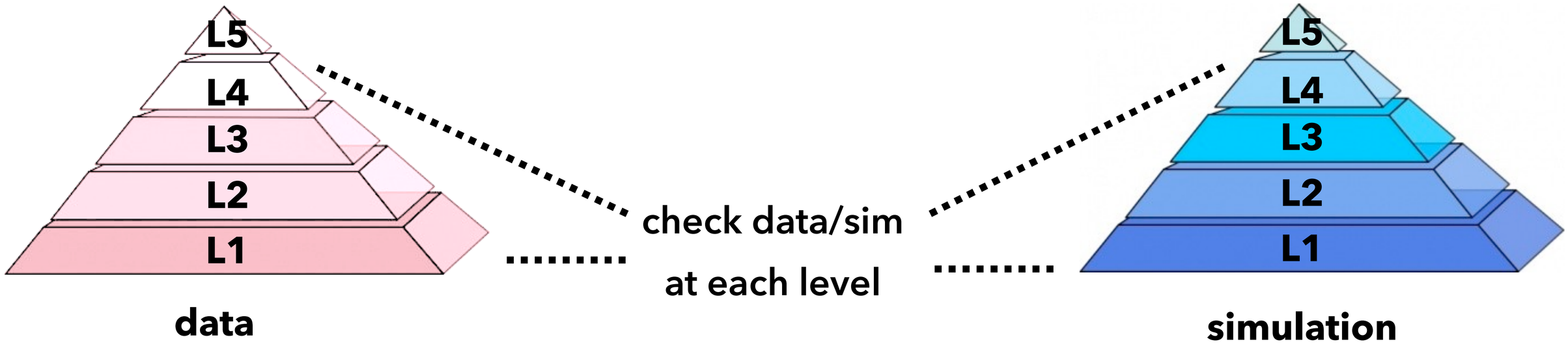
- One idea could be to process and store data hierarchically, such that one could easily grab a “level” they want. ***E.g.:***



- **Level 1** is everything (the low-level files contain raw time series, no high-level files)
- **Level 2** has “basic crap” removed (some time series will be unusable from Level 1 due to conditions / not passing basic quality checks → remove those)
- **Level 3**: preliminary filtering applied to the time series; start making high-level files…

Unifying data processing between data and MC

- Want our simulation to reflect the data as well as possible, so apply the same processing to both data and simulation.



- Example checks: could be count rates for peaks with a certain amplitude; distribution of peak widths; … **The more low-level checks, the better!**

Action items

Let’s try and identify…

- People who want to contribute to this effort :)
- Shared computing / storage resources we can apply for as a collaboration
- Quantities we want to keep in both the high-level and low-level files
- Any relevant categories of metadata / conditions to keep track of
- Reasonable sequence of “stages” of data processing, and which stage would be useful for which analysis