

# Metadata for the protoDUNE raw data: Initial ideas

Maxim Potekhin (Brookhaven National Laboratory)

ProtoDUNE Measurements Group Meeting

Aug 9, 2016

# Overview

- Well-designed metadata is instrumental in achieving the goals of protoDUNE.
- Note that presently it is about the *raw data* only (we'll have to deal with other types of data later, and of course there will be an overlap).
- The protoDUNE measurements WG appears to be the most appropriate forum to discuss this item since it includes people working on different research topics.
- Need to cooperate/coordinate with NP02 since a large part of data infrastructure is shared between NP02 and NP04
- What's presented today is not complete and is meant to seed the discussion!

# Why do we need to discuss this now?

- Metadata for raw data is created immediately after the data is taken so that the files can be handed over to the Raw Data Handling System. It is necessary to design, test and deploy requisite infrastructure elements for this to happen.
- The content and sources of the metadata may have a major effect on both computing requirements for the DAQ/offline boundary (cf. I/O and CPU, example: calculation of checksums) and interfaces between the various system, e.g. DAQ, Run Control, Conditions Database, Online buffer etc.
- Work on the Online Buffer has started and understanding these requirements is necessary to ensure its scalability and performance. Bandwidth is tight!
- Integration of these systems takes a while, so we need to get started on this asap.

## Where are we now?

- E-mail discussions have started in the past week and have been lively and informative. My intention is to capture this information and get more input from interested parties, then create a collaborative document signed-off by the group. This will inform our systems development.
- Design of the metadata can be factorized as follows:
  - *content (i.e. what specific information is included)*
  - *sources (i.e. a precise list of where it's coming from)*
  - *placement*
    - *the data file itself vs the database, and declaration of which is definitive during data access and processing*
    - *metadata/file catalog vs references to conditions DB*
- In terms of the main file catalog component we shall rely on Fermilab SAM.
- At the online systems level, interaction with SAM will be mainly via the Fermi FTS system which is tasked with the chain of data transmission from the online buffer to EOS/CASTOR and then on to FNAL/BNL. SAM is instrumental in keeping the state of each file transmission.

# Content1: run-specific information

- Run number.
- “Good Run” flag.
- Trigger type.
- **artdaq Configuration:** buffer sizes, parameters for modules in EB etc
- **DAQ Configuration:** this will be mapped to the run number in the run database, but due to importance will probably be included in the file as well.

## Content2: file-specific information

- We are close to giving up on enforcing strictly sequential trigger numbers in strictly sequentially numbered files. Due to multiple event builders and resulting multiple files being written the “strictly sequential” mode would require an extra “serializer” layer in the online system which is most likely not cost-efficient.
  - *“Min/Max trigger No. and timestamps”*: assuming the events in a given run receive numbers from the same sequence it is helpful to have the “min” and “max” event numbers for a given file — after the signal processing stage the data size will shrink and it will be possible to re-introduce the “strictly sequential” ordering of data in files
  - *Number of events*
- **Checksum**: needed to ensure data integrity during a few phases of transmission and storage.

## Content3: conditions

- **TPC conditions:**
  - voltages (drift, bias, etc)
  - temperature(s), pressure(s)
  - electron lifetime (if available)?
  - What else?
- **Beam instrumentation conditions.**
- **Beamline conditions:**
  - Beam type
  - Misc (target, magnet settings etc)
- **Electronics (FE):** gain, shaping time

# Sources

- **Content1/2 (run and file-specific):** filled by the DAQ and also by scanning/reading the file header once it's closed
- **Content3, Beam/target:**
  - need to learn about interfaces to the CERN Accelerator Division DB
  - **Beam instrumentation conditions.**
- **Content3, Run control/Slow control.**

# Placement

- There should be a definitive and documented source for each piece of metadata, i.e. the header vs the database
- Using the header is not practical for keeping all metadata that will be required in the production chain, so it can't be the primary container
- Some conditions-type metadata clearly have the “period of validity” characteristics and need to be store accordingly
- From the recent discussion it appears that the we need to pay attention to following issues
  - SAM is great but we won't be able to change the schema often or at all once the files start coming in
  - We won't be able to cram all the Content3 information we need into SAM
  - Therefore optimal factorization of functionality between SAM and the “other” databases is necessary
- A possible solution could be in an application layer above SAM and conditions databases that could do an effective “join” (in a limited sense). Also cf. NP02 with potentially different content.

# Summary

- This presentation was not meant to be very precise or exhaustive, it's just a start of an important discussion
- Someone needs to take charge of this topic
- A Metadata specification document would be a very helpful first deliverable
- This has a significant impact on software development and the effort required, so needs to be prioritized
- Liaison with NP02