Curation and metadata - concepts for data irreversibility

Hans Dembinski[1], Hermann Hessling[2], Ramesh Karuppusamy[3], Michael Kramer[4], Vladimir Lenok[5], Jakob Nordin[6], Susanne Pfalzner[7], Andreas Redelbach[8], Dominik Schwarz[9], Arno Straessner[10], and Vadim Vybornov[11]

[1]Technische Universität Dortmund
[2]Hochschule für Technik und Wirtschaft Berlin
[3]Max-Planck-Institut für Radioastronomie Bonn
[4]Max-Planck-Institut für Radioastronomie Bonn
[5]Universität Bielefeld
[6]Humboldt-Universität Berlin
[7]Forschungszentrum Jülich
[8]Frankfurt Institute for Advanced Studies, Universität Frankfurt
[9]Universität Bielefeld
[10]Technische Universität Dresden
[11]Forschungszentrum Jülich

Preamble

The curation of data and the concept of the associated metadata are relevant for all TAs in PUNCH4NFDI and, obviously, also very much relevant beyond our own consortium for the whole of NFDI. A number of specific challenges arrive with the focus on TA5, caused by the huge data streams and the needs for heavy on-line processing. Solutions to address these challenges must not, however, be designed in isolation of TA5 but must find the applicability also in other TAs, if not now then certainly in the future. Vice-versa, concepts and implementations in other TAs must be flexible enough to accommodate TA5 requirements in the future. The aim of this document is therefore *not* to provide a general and complete description of metadata in all fields of PUNCH sciences, but to start a discussion of the relevant topics by highlighting some of the specific TA5 challenges. Consequently, the document is naturally biased towards TA5 needs to convey our *current* thinking. That thinking will evolve with time as part of a process including ongoing and future TA5 work and discussions with other TAs. This document is a snapshot of this process.

Contents

1 Introduction

Data are of little use unless there is knowledge about what the data represent. A simple one-dimensional sequence of numbers could be a coded message or a measurement of events during certain (unknown) time intervals. A two-dimensional data set may represent a fundamental scaling law, like the expansion of the Universe measurable from a correlation of galaxy distance and velocity, or a

ledger with the balance of customers. A three dimensional data cube may be weather data as a function of location and time, or something else entirely. In other words, data are only as useful as the appropriateness and completeness of their "metadata" that provide a description of the data. But metadata describing a measurement or experiment should ideally not only describe a data set and its relevant parameters, but they should also contain information about the experiment itself, environmental conditions and, in particular, also any relevant information about how and why certain information was selected and, ideally, why other were not. Increasingly, metadata by themselves can become very large as a consequence.

While the volume of data and metadata is increasing by our ability to measure more details and larger parameter spaces, we also need to increase our ability to reduce these data in order to handle, manage and store these larger volumes. This results in a loss of information, and the degree of information contained in the metadata has direct consequences for the degree of data irreversibility. Task Area 5 is dedicated to develop methods to keep the degree of information loss as low as possible.

This document is considered to be a "living" attempt to develop the best concept of metadata in light of the challenges addressed by TA5. These concepts are likely to differ between different case studies, but lessons from one area are useful to overcome challenges in others. As we interact with the other task areas, and as our work in TA5 progresses, new strategies and methods are developed. This document seeks to capture the status of our thoughts, attempting to be a reference for current and future discussions.

2 Concepts

The Merriam Webster dictionary describes *metadata* as "data that provides information about other data". This relational definition as data about data might appear as rather technical, however a consistent and efficient system for organising and accessing metadata is a crucial aspect for data analysis, particularly for long-term analysis preservation. We adopt the following operational definition.

We call *data* everything that is fundamentally obtained from an experiment or observatory, whatever format (binary numbers on tape, recorded analog signals, photographic plates). If data are lost, the information cannot be restored. In that sense, these data already contain metadata, which we call level 0. Level 0 metadata describes and gives appropriate context to the raw data, for example, when the data was taken, units in which numbers were recorded, etc. If level 0 metadata is lost, the associated data becomes less informative since important context is missing, or entirely useless. Since level 0 metadata is deeply connected with data, we will implicitly include when talking about data, and only mention it explicitly when necessary.

We further distinguish between *transient* and *persistent* data (and associated level-0 metadata). Transient data is not archived for later processing. Most

experiments search only for specific events in the stream of events provided by the experimental apparatus. Storing all events would be uneconomic or unfeasible (data processing rates are limited), and thus online algorithms select which events are stored and the others are discarded. Transient data is data stored in temporary buffers processed by these algorithms.

Before describing various technical aspects of metadata in the context of data irreversibility we underline that they are instrumental to enable *reproducibility* of results at different levels of processing.

3 Data irreversibility and metadata

3.1 Short overview of work in TA5

In the case of on-line filtering or processing, only a (limited) version of the original data can be stored. Sufficient metadata then need to be provided on how this selection process was implemented and executed. This is crucial to evaluate the content of an archive, to present new choices of criteria for online filtering or to judge the discovery potential in light of selections made. Therefore, in addition to metadata that describe the basic properties of the data and the instrumentation used, we need to provide additional metadata that describe the first selection of data. Especially, we need to capture in some way a description also of why or which data have *not* been captured. We need extra metadata including the complete chain of algorithms needed to enable, in principle, the reproducibility of selections.

In a dynamical life cycle of data (see Figure 1), the filtering process is not static but dynamic. Any scheme must be flexible enough to accommodate different types and numbers of decision processes.

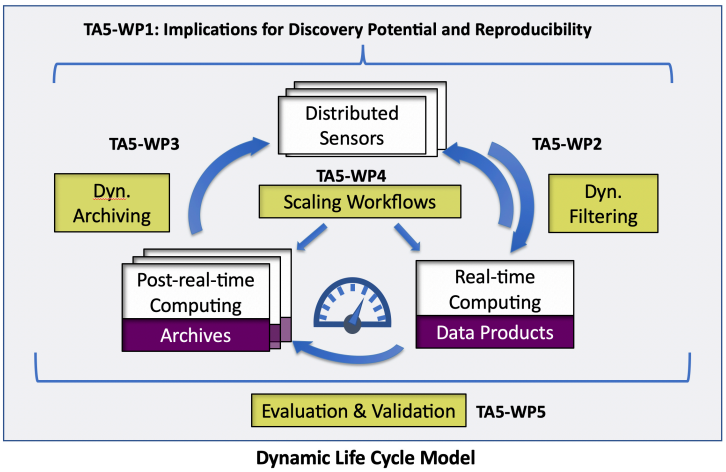

Figure 1: Structure of TA 5 as dynamic life cycle of data

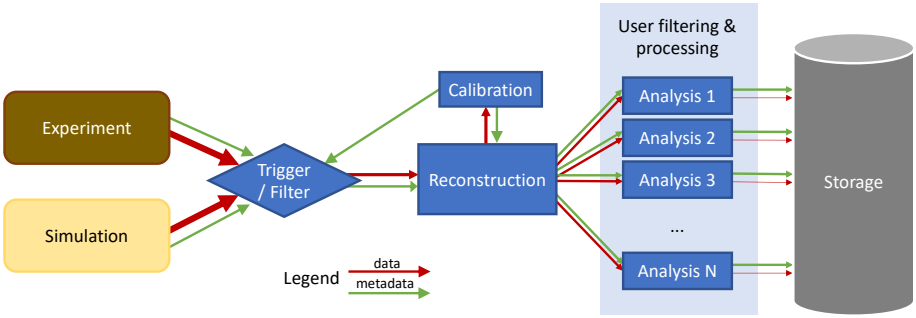

Figure 2: General data processing graph for particle and astroparticle experiments. Variations of the data flow and triggering scheme are possible. The arrow width qualitatively indicates the data rate.

A detailed description of the tight interconnection between all work packages (WPs) of TA5 has been presented in the corresponding section in the PUNCH proposal [1]. Our concepts and developments aim at common solutions for processing of dynamic metadata in the fields of PUNCH sciences and potentially beyond. Here it is important to note one main distinction between high energy physics experiments and astrophysical observations: In collider experiments of high energy physics one can practically increase the number of metadata by improving and repeating experimental setups. However, this is not the case for astronomical observations where the observed object, the universe, is literally out of control. One can only partially overcome this problem by turning to statistically large samples of objects. In contrast, in large data streams originating from smart cities, for instance, a similar situation occurs, i. e. the "experiment" cannot be controlled, but large samples are also not available. Developing techniques for such situations are among the intended tasks for TA5, but not in scope of this document.

3.2 Data reduction and the challenges for metadata

A generic data processing graph for a particle or astroparticle experiment is sketched in Figure 2. Data is generated either by the experiment or its Monte-Carlo simulation. The trigger step keeps only data of interest. The reconstruction step converts low-level data into higher-level objects, for example, hit coordinates and collected energy deposits into tracks or showers. It is followed by analysis-specific processing and filtering, which may happen in parallel. Some data is used to calibrate the reconstruction and the trigger.

In a general sense metadata describe other data thus supporting the F.A.I.R. principles in a direct way. In the broad context of NFDI, metadata are information (context) that describes an object such as a dataset as richly and complete as possible. As data is further analysed, processed or related to other data,

metadata can only grow in time. In other words, additional layers of metadata will be added, corresponding to the increasing levels of metadata as described in Table 3.3. This table is one example of how to establish structured meta. Another example is being used by NASA [2], as also outlined in section 5.4.

It is interesting to note that a section focusing on metadata and data provenience has been established within the NFDI [3]. However, the general concept of metadata implicitly assumes that metadata are assigned only to data that are permanently stored. During the huge data reduction in real-time filtering one can however distinguish between *transient data* typically constituting the majority of processed data and a fraction of permanently stored data. The latter are those filtered and thus *usually* represent *persistent data* that should be described by metadata.

In the context of time-critical data processing and the dynamic life cycle of data we want to sharpen and extend this common concept of metadata:

Metadata characterise the data during the complete, dynamic process of data taking, as sketched in Figure 2. Due to the intrinsic time-dependency of the processing, one first has metadata describing the input or starting phase. These metadata typically include status and bookkeeping information of detector systems participating in the data taking, e.g. the current time, how many sensors are active at the start of data taking, or which set of calibrations is loaded, or which telescopes are participating at an observation at which locations or frequencies. Furthermore, as a result from online algorithms extra metadata will be derived during the actual data taking. One important example is capturing the triggering or filtering activities: How often has a specific trigger 'fired' thus selecting a reconstructed event in high energy physics or time-domain astronomy to be stored permanently. What is the background level of noise-like signals, and what threshold has been established and applied during the experiment or observations? Such a threshold is likely to change within time, even during the duration of an experiment (e.g. during sunrise and sunset, the ionosphere changes rapidly, changing the level of Faraday rotation which may be monitored, whereas the interstellar component can be expected to remain constant).

Another kind of metadata that could arise during processing and storing data might be unwanted back-pressure when a system is not capable of processing data in the required rate. The time indicating the onset of a phase of back-pressure would therefore also constitute an example of relevant metadata. From the viewpoint of the dynamic life cycle of data metadata are thus strongly related to the process of (dynamical) filtering.

Additional important metadata are necessary to describe the logic or conditions why specific filters or workflows have been chosen. This implies a mapping of the algorithms underlying the decision process executed in the context of real-time data taking and the dynamic life cycle of data as outlined in section 7. Deriving this from, for instance, a neural network is highly non-trivial, especially in real-time.

This extended concept of metadata is therefore a consequence of the complex interplay of our highly dynamic data model. These high-level metadata arise when addressing questions such as: Why are other filters disregarded? Which

Data level	Content	Sample content
L0	A sensor measurement.	Count level in CCD detector.
L1	Annotations referencing L0	CCD temperature and readout gain.
L2	Operation on Level 0/1.	Photon flux reaching telescope.
L3	L2 annotations ("metadata").	Reference to calibration algorithm / ancillary data.
L4	Operation on L2/3.	Brightness of astronomical source.
L5	L4 annotations.	Reference to astronomical catalog.
L4	Operation on L4/5.	Source classification.
L5	L4 annotations.	Reference to astronomical catalog.
L6	Analysis output (Operation on L4).	Real-time follow-up announcement.
L7	L6 annotations.	Publication reference.

Table 1: A description of data levels suitable to a sample analysis of astronomical data. Here, data and metadata levels are interleaved to emphasize that higher level data can depend on lower level metadata.

set of input data/conditions/archives triggered a specific filter selection? Can we classify a trigger/filter as anomaly-based?

The layered metadata approach presented above should be able to address some of these challenges, as a hierarchy of processing (and decision) steps can be captured in principle. We discuss this further in the following.

3.3 Hierarchical dynamic metadata

Based on data (and level-0 metadata), higher-level data and metadata are built, which form a natural data hierarchies. Metadata is of a higher level if its construction depends on metadata of lower level or if it describes data of a lower level, otherwise it is of the same level. With the level of metadata, the abstraction level increases and the distinction between data and metadata can become blurred as high level data directly depends on lower level metadata. In practice, low level metadata is often automatically created and centrally processed to higher levels, and most analyses operate entirely on higher level metadata.

As an illustrative example from astrophysics, Table 1 shows an hierarchical view of data and metadata for optical astronomy. In this case data levels range from level-0 to level-7. Initial data are from a telescope where an observation results in a number of counts being registered in a pixel when reading out a CCD array (L0), which are recorded together with information regarding current observing conditions (L1). L0 and L1 data are then used to derive a physical measurement (L2) through comparison with a sensitivity curve (L3), converted into an intrinsic brightness (L4) after using a distance found in an external cat-

alog (L5), labeled (L4) based on a classification algorithm (L5) and, finally, this information is communicated to an external receiver (L6) after fulfilling some publication criteria (L7). From this viewpoint, data is not divided into "data" and "metadata", but rather into levels containing data of increasing complexity, and dependency on other sources of information. Information at each level requires lower level to have existed in order to obtain meaning. Data hierarchies can also branch, both in the sense of higher levels combining lower level data from multiple sources (sensors) as well as different higher level processes using the same lower level data. Data irreversibility emerges when some parts of this hierarchy is not available for subsequent analysis. First, the capacity to store lower level data might not exist. For example, it is conceivable that L1+ data can be kept to indicate that measurement were made but not the raw observations themselves, or that a higher level analysis decides whether lower level data is stored. Secondly, some measurements might never have existed. This could be due to an observation being cancelled due to cloud coverage or having been superseded (replaced) by a different target.

In a collider experiment, data are the raw event information, hits in the tracking system, ADC counts in the calorimeter system, etc. In an astronomical observation, they are ADC counts in a camera, the digitized amplitude trace of a radio antenna, etc. Level-1 metadata would be reconstructed tracks in a collider experiments, air showers in an astroparticle observatory, or a transient or persistent signal in astronomy. Level-2 would be reconstructed decays in a collider experiment, a light-curve in optical astronomy, or a dispersion track of a pulsar or Fast Radio Burst (FRB) candidate in radio astronomy. Even higher level metadata could be analysis results obtained, like histograms, or catalogues of lower level objects that pass certain filters, or the statistics of similar analyses on adjacent positions or epochs, giving means to decide the reality of a detected signal. In this definition, the level is defined by the number of intermediate processing steps and not necessarily comparable at higher levels between different experiments. An experiment which requires more intermediate steps for processing would introduce more intermediate levels of metadata.

However, it is also common that complementary experiments are performed simultaneously during the same data-taking process. As a use-case, consider a pulsar search observation with the a radio interferometer such as MeerKAT [4]. While searching for pulsars, a high-volume high-speed data stream from the telescope's digitiser is processed by a beam-former, two subsequent pipelines work on the same stream at $\sim$300 Gb/s. One pipeline searches for accelerated binary pulsars, expecting very weak, Doppler-modulated periodic signals, while a second pipeline looks simultaneously for single FRBs [5]. In case of an FRB detection, a buffer containing the dispersed short-duration (few ms) signal is replayed and fed into an imaging pipeline that forms an image to localise the FRB. In parallel, the pulsar search pipeline searches a parameter space of binary orbits to detect a signal. When something is found, a fraction of the data set is marked for permanent storage, together with information on the candidate (and anything additionally found). The metadata related to such pulsar searches are already quite large in size (a few hundred GBs), and the sizes of the stored

items become a crucial consideration. One should therefore avoid duplication of storage. In both cases of this example use-case, the same Level-0 data stream is being considered and the two pipelines may even use the same formed de-dispersed time-series for their analysis (constituting a Level-1 data product). However, from then on-wards, there are two branches that describe a completely different decision process: why was the FRB signal considered to be real? The statistics used for a decision here is a vastly different one from that why a signal is considered to be periodic and hence potentially a pulsar.

This use-case (for more details see also Section 4.3) highlights two different aspects of our discussions and the methods that we need to develop. Firstly, it is clear that our concepts must be flexible. Already in this rather specific example, a degree of variation in the requirements is visible. This becomes even more challenging, when for instance deriving schemes for astronomy or particle physics experiments, or experiments in general and in simulations. Secondly, we need to make decisions, how to handle different branches that have the same Level-0 or Level-1 origin. It seems inadequate to keep copies of the same low-level items, but in order to avoid duplication lower levels in our metadata structure may be simply a reference, pointing to a single physical location of those items.

Every commensal observation or experiment will face similar questions. In the course of TA5, we will study various implementations already used in the communities to offer adequate answers. It is unlikely that we find a single best solution, but the propose of TA5 will be to provide a guide and eventually a set of tools, depending on the actual requirements.

3.4 Recursive metadata

Another even more general method for describing the interplay between data and metadata would to use recursive relations, where each record can link to other records in a process with no definite end. This chain would contain references to both data, metadata and algorithms.

The cause of a specific dataset, as in why a specific piece of record exists, is not immediately obvious in e.g. a dynamical archive. [It might be said to be obvious in a standard archive which records all output from an experiment.] The cause can be seen as a union of several things, including "physics" but also experiment setup, software and ancillary data like weather. Data and metadata are here even less distinguishable, and every record contains link to other parent records.

Items such as dynamical filters and weather measurements would be seen as data-records in different archives (e.g. `filters_used_by_SKA_DB`, and the structure of data viewed as a recursive tree based on pointers to different collections rather than data — metadata — abstracted-metadata. A record in the "SKA-filters" collection could then refer to a set of previous measurements that were used for training together with a link to a specific ML algorithm. Metadata items would then frequently look like: `{'link':'url_of_origin_db', 'collection':'hash or set of elements in origin db', 'type': 'input`

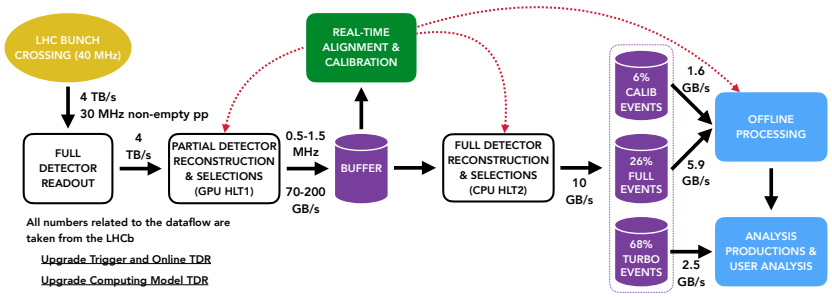

Figure 3: Current data processing pipeline of the LHCb experiment for proton-proton collisions [6, 7]. Arrows indicate data flow, which are annotated with event and data rates.

`data'}` , where 'type' would be one of a limited set of known keywords like

$$'inputdata', 'algorithm', 'trainingdata', 'configuration', 'ancillary'.$$

Based on a metadata file one could then in principle traverse this tree.

4 Use cases

4.1 Data from tracking in high-energy physics

In collider-based experiments, it is common to work almost entirely with metadata. A sketch of the LHCb data processing pipeline is shown in Figure 3. A full detector readout at the revolution frequency of the LHC leads to rate of 4 TB/s, which is infeasible to store permanently. Instead, raw data is processed online on GPUs and CPUs to form higher level objects such as tracks and even fully reconstructed decay chains. Analysts define upfront which decays are of interest and which subset of the full event information is needed for each analysis. Only this information is written to the disk and the rest is discarded. This reduces the initial rate by a factor 500 to manageable 10 GB/s through efficient filtering and because high-level objects such as tracks and decays can be represented with just a few numbers.

The most basic objects that analysts work with are typically particle tracks in the detector. Tracks are metadata, a dynamic interpretation of the raw detector data, the space-time points of hits in its tracking system and corresponding energy deposits. The reconstruction software identifies point patterns that match tracks of charged particles and assigns a momentum and charge based on the curvature of the particle in a magnetic field, or an energy and direction, for neutral particles that hit the calorimeter system. To assign the correct momentum, the software will take the exact magnetic field into account, continuous energy loss in the detector material, and detector alignment parameters, that need to be regularly updated.

The reconstruction software identifies hit patterns that are consistent with a particle moving in the magnetic field, but this identification is not perfect. Some particles are missed. For muons that neither decay nor interact strongly with the detector material, this is only a few percent or less [8]. Some fake tracks are formed as well, which do not correspond to a real particle. This happens if the software connects track segments of two unrelated particles, for example. In the LHCb experiment, the fraction of fake tracks is non-negligible at very low and very high momentum [9, 10]. If the tracking system is well calibrated and understood, the chance probability that a track corresponds to a real particle can be calculated and provided as additional metadata for filtering. An analyst may use a hard requirement on this probability to select only tracks which are very likely real particles, at a cost of some efficiency, or achieve higher efficiency at the cost of admitting more fake tracks.

A detector also provides particle identification information for tracks. This may be based on energy loss, time-of-flight, the fact that the particle penetrated shielding, and the presence or absence of showers in the calorimeter system. As with the tracking, the identification is typically not perfect, and thus metadata is assigned to the track which allows analysts to dynamically filter tracks. In case of the LHCb experiment, a neural network uses as input all various sub-detectors to assign a probability that a charged track is originating from a pion, kaon, proton, muon, or electron [11].

Tracks that almost meet in a common vertex (space-time point) are further combined by the tracking software to identify a primary vertex, the point where the beam particles collided, or to identify a secondary vertex, where a particle decayed into other particles. These decay candidates are higher-level metadata. The chance probability that unrelated tracks are merged in this way can again be calculated and is provided as additional metadata for filtering. In LHCb, many analyses work entirely on this level of metadata and most of the trigger lines are based on reconstructed decay candidates [6]. A part of the LHCb data is made available for the general public in this way [12]. As part of the LHCb Open Data Project (see also section 5.5), scientists can interact with the LHCb processing pipeline via a website interface. They select decays of interest and then submit a request to extract and process this selection from internal storage for download.

4.2 Data from the ground-based air-shower observations

Similar to the tracking in high-energy physics described in the previous section, research in ground-based air-shower observations is based on metadata with an event being the most basic object. The event is the lowest possible high-level interpretation of the data obtained from the individual detectors of an air-shower instrument. These events, as collections of individual detector signals, are subject to further reconstruction which finds appropriate values of the shower impact point into the detector, incoming direction, depth of the air-shower maximum, the energy of particle initiating it, etc. Each of these derived quantities is essentially metadata. The set of the metadata varies for different

instruments. An overview of measurements of cosmic rays can be found e. g. in [13].

The event identification from the initial, individual detector data, is not straightforward due to the occasional coincidences of the background counts appearing at the same time that can form a false event, or malfunctions of the individual detectors of the instrument hampering the event identification because of the partial loss of the data. The reconstruction procedure attempts to recognize the true events based on the best available knowledge about the instrumentation and performs reconstruction of the air-shower and primary-particle parameters with the best available methods. The result of the reconstruction procedure can be called metadata of the low level.

These low-level metadata are initial data for any high level analysis. With appropriate description of the detector, the found energies are the initial data for reconstruction of the cosmic-ray energy spectrum, the depth of tho shower maximum for the mass composition analysis, and the incoming directions for the anisotropy studies.

The modern approaches to the air-shower data analysis attempt to use machine learning methods to improve reconstruction of some of the low-level metadata and make new types of metadata available (such as the primary particle type). The entry information for some of these approaches is a mixture of data and metadata to achieve the best performance. The identified particle type, which is also metadata, is a complex, non-linear function (formed by the machine learning method) connecting a set of metadata or metadata+data to a new type of metadata. This more generic approach to the metadata organization, in contrast to a simple leveling, can be a promising approach for the future.

4.3 Metadata in pulsar searches

As mentioned in Section 3.3, pulsar/FRB searches are done on data at Level-1, and therefore have an additional layer of metadata. Primarily this consists of the original fast-sampled time domain data from multiple telescopes which are transformed to have frequency and time-resolution. The transformation is typically done by means of an FFT either on FPGAs or GPUs. The beamformer will then compute multiple "pixels" or beams on the sky that are much smaller but are more more sensitive than the wide-beam from a single telescope in the interferometer. Once the beams are computed, the beamformer further reduces time and/or frequency resolution. Additional products maybe computed at this time, e.g the four Stokes representation of the electric field sensed by the antennas. At this point, one encounters several hundred to thousands of beams (or independent data streams that contain both frequency and time information). These are the beams that are subjected to FRB or pulsar searches, as also described in more detail e. g. in [14].

At this stage, Level-1 metadata contains information on the beam position, start epoch, time and frequency resolutions, band width, receiver frequency and type of data (full Stokes, total intensity or any other Stokes). One can imagine that additional information at this level can benefit future search pipelines,

such as RFI information, antennas that are masked (due to maintenance, malfunction, or bad local RFI). Similarly, some details on the way FFT was done (e.g so-called poly-phase filterbanks and what response was used, as this can introduce leakages in other parts of the band).

The above data is then subjected to searches for periodic or transient astrophysical signal both of which have the hallmark dispersion effect. However the dispersion is not known *a priori* and therefore searches over a range of dispersion trials are carried out both for FRBs and pulsars. For pulsars, the search for binary systems need a further trial in the acceleration and/or jerk dimension. Following this a FFT search and harmonic summing of the resultant data is done, in order to identify the presence of a pulsar candidate. At this stage the newly generated information like period, signal-to-noise ratio, dispersion/acceleration/jerk trial has to be captured. The presence of any residual RFI (that potentially shows up as a candidate at multiple dispersion and acceleration trials) results in a large number of pulsar candidates, which were in the past vetted by an expert, and recently machine learning schemes have been put in to use.

4.4 Concepts for related data from simulations

While most data are published in observational astronomy, and the FITS format is widely accepted, the situation is very different for astrophysical simulations. Here the open data policy is much less applied than in observations. While some sub-communities publish their code and resulting data, many do not follow such an open-access policy. The reasons are not necessarily proprietorial, but the groups often lack the human resources to provide adequate code and data documentation. Thus there is the general trend that extensive numerical collaborations tend to publish their codes, but smaller groups and single researchers do to a lesser degree. During the last decade, the situation has improved in making codes open access; however, publishing the resulting simulation data rare in many sub-fields. Consequently, a shared culture of publishing data with standards in formats and metadata provision still needs to be developed.

When publishing simulation results, the most apparent metadata to provide is the code that has been used to produce these data. Usually, these codes include an input file where the main variables are defined. Such input files include scientific as well as numerical information. For example, an astrophysical variable could specify the used potential for modelling an astrophysical object. An example of a numerical variable is the timestep used. Providing a comprehensive list of the astrophysical and numerical parameters is quintessential to reproducing the results.

In high-energy physics, the detectors used to measure particles are static, i.e., their operating conditions do not change. In astroparticle physics, on the other hand, the situation is different because important conditions can change within a short time, such as weather conditions (clouds) or observation directions. This is accompanied by enormous challenges with respect to simulation: much larger simulation data is required to adequately account for these dynamic influences.

It is to be expected that the associated metadata can no longer be kept in relational database systems due to their sheer size.

However, there are also technical aspects that should be contained in the information provided by the metadata. The metadata should contain information about the computer used and the technical setup. This information is critical; for example, the same code run with different precision can give different results. Similarly, the number of cores used for producing the data is potentially essential. As many versions of a code exist in parallel, providing a version identifier would be essential.

Usually, astrophysical data contain no personal data. However, for reproducibility purposes, it might be necessary to include the name and contact details of the person(s) who produced the results. In addition, it is also helpful to provide the identity of the code developer(s). Some codes already contain an automatic protocol for this information. However, providing this information is often not the standard procedure.

It is interesting to note that TA3 has started to classify metadata related to simulations in PUNCH.

5 Previous approaches and frameworks

Here we present an overview of existing approaches and solutions related to processing metadata. At this point we can confirm the observation of the authors of [15] that designing analysis metadata systems has historically received little direct attention. That analysis also investigates requirements for processing metadata in future HEP experiments. In this section we want to summarise several other relevant references to give an overview of some lessons learned before and thus also form a basis for future developments.

5.1 Data provenance

An example of a generic approach is DataCite's Metadata schema [16], intended to support citation and discovery of data in a broad range of research datasets, without being customised to the needs of data-intense PUNCH sciences.

Astronomical use cases have been evaluated in the VAMPIRA project [17]. This analysis can serve as a basis to discuss the requirements, challenges, and opportunities involved in designing both a tool for automated provenance generation and the associated provenance model.

Standards for protocols and data schemes defining the technologies behind the Virtual Observatory are summarised in the article [18]. This includes also the Table Access Protocol (TAP) enabling flexible and powerful dataset discovery largely independently of their types.

The FITS (Flexible Image Transport System) format is probably the most commonly used digital file format in astronomy [19]. This format consists of multi-dimensional arrays including extra calibration information, together with image origin metadata. The format has meanwhile been extended to also contain

different types of data, e.g. time domain pulsar data ("PSRFITS format"). A list of officially registered FITS formats is maintained by NASA (see `https://fits.gsfc.nasa.gov/fits_registry.html`). In all cases, it is important to note that all metadata are stored in a human-readable ASCII header as a table of keywords and values.

Within the scientific community of lattice QCD the Metadata Archives at Fermilab and the International Lattice Data Grid have been developed since many years [20]. Here the description of metadata follows flexible XML-based formats. See also the recent paper [21] for a status update.

The design of the ATLAS Metadata Interface (AMI) 2.0 metadata ecosystem is summarised in [22, 23] explaining also the underlying Metadata Querying Language (MQL) - a domain-specific language allowing to query databases without knowing the relation between entities.

5.2 Frameworks for Big Data

The Rucio framework for large scale data management [24] is well-tested for data distributed across heterogeneous data centers at widely distributed locations. Originally developed for the needs of the ATLAS experiments, it is utilised by several other experiments in the field of PUNCH sciences. Rucio offers also advanced features like dynamic data placement or automated data rebalancing and extension modules which can access internal instrumentation data.

Various experiences using Rucio in exascale scientific data management are discussed in [25] focusing also on other experiments beyond the original ATLAS community. Here the autonomous declarative way of handling dataflows, the transparent handling of data incidents, and the capability to monitor the flows in Rucio are emphasized.

In this context it is interesting to note that the CMS experiment at CERN has decided to perform a transition to Rucio data management [26] for run 3 data.

An overview of the ESCAPE data lake including SKA Rucio ESOC is given in [27]. This document describes the corresponding challenges of the pilot Data Lake and presents both the continuous monitoring and the experiment-led tests in detail.

Modern BigData technologies to store and access metadata for the ATLAS experiment are addressed in [28] including the EventIndex application that was entirely developed having in mind the usage of modern structured storage systems as back-end instead of a traditional relational database.

The performance of time-series databases is in the focus of [29] investigating benchmarks for data management and data analysis systems of large-scale scientific facilities. The authors show relevant benefits of relaxing the consistency constraints for performance. Also the columnar format of databases in conjunction with data partitioning into multiple parts boosts corresponding ingestion rates and leads to improved performance of data queries.

The overview of HEP Data Frameworks in [30] has concluded that it is essential for the HEP community to invest in the development of its data-processing

frameworks as fundamental building blocks for exploiting the available computational resources.

A general-purpose framework for big data processing has been introduced in [31]. Thrill is based on C++ to improve performances and it uses arrays as primary data structures to enable operations like sorting or combing of fields and elements.

Another framework for large-scale data processing that should be mentioned is Apache Spark. While Spark is written in Scala, it provides high-level APIs for Scala, Java, Python and R. Spark includes different modules allowing to execute data processing, streaming, SQL or machine learning workloads. Although it is slightly inferior to the Message Passing Interface (MPI) in performance and resource consumption in some tests on HPC clusters, Spark is better in fault tolerance and easier to use [32]. Spark is often used in conjunction with Apache Kafka (a distributed event streaming platform), which provides a powerful tool to build complex data pipelines such as dynamic filtering.

5.3 PUNCH4NFDI

Recently, a comparative analysis of two metadata curation use cases from the field of astroparticle physics has been presented [33]. The projects KASCADE Cosmic-ray Data Center (KCDC) and German-Russian Astroparticle Data Life Cycle Initiative (GRADLCI) have been analysed regarding the requested functionality, chosen data architectures, technical solutions and, especially, metadata management approaches.

Over many years PUNCH sciences have pioneered scientific projects based on open data. Some of these approaches are described in [34]. Here we shortly summarise two projects related to open data from experiments at CERN or HERA.

5.4 Data Processing Levels in NASA/EOSDIS

As outlined in [2], a hierarchy of basically 5 layers has been established in the project EOSDIS being NASA's Earth Observing System Data and Information System. It is interesting to note that in that context level 0 products are raw data at full instrument resolution. At higher levels, the data are converted into more useful parameters and formats. Moreover, all EOS instruments must have Level 1 products. This corresponds to the following scheme:

- Level 0: Reconstructed, unprocessed instrument and payload data at full resolution, with any and all communications artifacts (e.g., synchronization frames, communications headers, duplicate data) removed.
- Level 1A: Reconstructed, unprocessed instrument data at full resolution, time-referenced, and annotated with ancillary information, including radiometric and geometric calibration coefficients and georeferencing parameters computed and appended but not applied to Level 0 data.

- Level 1B: Level 1A data that have been processed to sensor units (not all instruments have Level 1B source data).
- 2: Derived geophysical variables at the same resolution and location as Level 1 source data.
- 3: Variables mapped on uniform space-time grid scales, usually with some completeness and consistency.
- 4: Model output or results from analyses of lower-level data (e.g. variables derived from multiple measurements).

5.5 CERN open data and preservation

The CERN Open Data portal [35] provides an access point to a growing range of data produced through the research performed at CERN. The long-term preservation of data and the corresponding open data concept at CERN have been decided a decade ago, see e. g. [36]. In this context it is important to note that data produced by LHC experiments are usually categorised in four different levels:

- Level 1 data provides additional information on published results in publications, such as extra figures and tables
- Level 2 data includes simplified data formats for outreach and analysis training, such as basic four-vector event-level data
- Level 3 data comprises reconstructed collision data and simulated data together with analysis-level experiment-specific software, allowing to perform complete full scientific analyses using existing reconstruction
- Level 4 data covers basic raw data (if not yet covered as level 3 data) with accompanying reconstruction and simulation software, allowing the production of new simulated signals or even re-reconstruction of collision and simulated data

In order to validate workflows based on real-time data selections, level 4 data are relevant. However the numbers of these raw-level datasets in open data are small in comparison to datasets corresponding to more abstract data levels. One example of level 1 data is a CMS open data in raw format [37]. Also OPERA detector data available, e.g. http://opendata.cern.ch/record/10101.

Recently LHCb has described how to select decays of interest in LHCb Open Data [12]. Using a website interface the corresponding evaluations are performed essentially based on Level 3 data, corresponding to the aforementioned classification.

It is interesting to note that the ordering of these four categories are to some extend *inverted* and less flexible as compared to the definition in this document in section 3.3.

5.6 Data preservation for the HERA experiment

Also the H1 collaboration at the HERA experiment at DESY has developed methods to maintain the corresponding data, the related software and the documentation. An overview of these efforts and experiences after many years of the end of data taking at HERA is presented in [38]. The challenges in the transitions towards modern computing platforms and an object–oriented data analysis framework are outlined. These lessons can also become relevant for data curation in PUNCH sciences.

6 Requirements for metadata in PUNCH

Hierarchical metadata are tied to a distinct (atomic) observation: Astronomy has been uniquely good in storing this kind of metadata. The FITS format has played an enormous role, and it has been adapted to many more types of astronomical data than the initialed foreseen usage for images.

Basic information in the form of metadata is for most observatories already fairly complete and compact. Further standardizing this, e.g. by adding uniform sets of generic metadata keys, runs a great risk of increasing storage sizes while still not being specific enough to fully describe how an observation was made.

6.1 WP 1 - Discovery potential and reproducibility

In the PUNCH sciences we face two types of scientific questions. In the first one we start from an already established model of the world and measure some model parameters, prominent examples are the measurement of the Higgs mass in the context of the Standard Model [39, 40], or the modern measurements of the Hubble-Lemaître expansion rate of the Universe in the context of the flat Lambda Cold Dark Matter model, see e.g. [41]. Searching for yet unknown radio pulsars or for a yet unknown nuclear isotope falls into the same category, as we know what we are looking for, so we carry out a parametric search in a (large) parameter space. But there is also a second class of scientific questions, which concerns the rare, unexpected and transformational discoveries. Here, the prior model of the world states that the so far unknown object or phenomenon does not exist and we don't even know before the fact what the relevant parameter space is and for what we should look out. From a statistical point of view these questions are not parametric tests, but rather falsify the statement that something does not exist.

Most of current activities in the PUNCH communities focus on the first question and the unexpected discoveries often happen by chance and are mostly driven by curiosity, or the lucky discoverers actually looked for something else, like Penzias and Wilson when they discovered the cosmic microwave background radiation instead of being able to further reduce the system temperature of their horn antenna [42].

The enormous data streams of many of the experiments, observations, and simulations in the PUNCH communities, don't allow, for technical and economic

reasons, to store and save all data and especially the sustainable and responsible use of resources calls for energy efficient algorithms and data storage. However, sustainability might be in contrast to the scientific need for reproducibility and the ability to make unexpected discoveries. In fact, the most efficient algorithm to test a hypothesis or to measure a parameter would of course be a routine that just returns a logical value (true or false) or a posterior probability distribution for a parameter, but if nothing else is stored the result is no longer reproducible and unexpected discoveries would become impossible.

The process of data reduction must therefore first of all be documented such that it is transparent how the data where taken and how they were processed, which forms a huge set of metadata (mainly in the form of instrument descriptions, software packages), at all possible data levels. This is essential to even call the data reduction scientific and not storing any of the information on these two aspects would be a violation of the scientific method. Let us refer to that as the essential description of the experiment or observation. This could still allow to drastically reduce the amount of transient sensor data, either before or after processing them.

Let us first turn to the aspect of reproducibility in the light of the first type of a scientific question.

The above mentioned essential data are just enough to allow others to understand what was done. In an ideal experimental set up, that is actually almost all that we need to make it reproducible, as one could just repeat everything that has been described. From that point of few, radical data reduction and compression, e.g. by means of a trigger system, is possible and might be efficient, as long as mainly questions of the first type are addressed. Here however, it might be that rebuilding the experiment is expensive, which then calls for the storage of intermediate data levels, such that the results can at least be partially reproduced starting from a higher data level, which also implies that the very first data reductions steps are highly trusted and have been extensively tested, such that storing the lowest data levels is not necessary.

The situation is more complex for observations. An observation is a measurement or a set of measurements of a system whose state we cannot control at all. That is certainly the case is astronomy and cosmology, but, depending on the experimental set up, it could also be the case in the laboratory. In such a situation it is simply not enough to document the procedure, as e.g. any specific supernova (SN) will never repeat again and the specific details can be different, e.g. the line of sight differs from SN to SN, such that cosmic dust could absorb some light from one direction and gravitational lenses could magnify the light from a SN in another direction. Therefore, a key aspect here is to figure out for each type of observation what data levels must be kept, such that an observation can be reproduced in the sense of being checked by an independent colleague.

Also data that are not of direct interest to the scientific question, but affect the performance of our sensors that target the science question at hand and thus the quality of the senor data must be analysed and taken into account in order to reproduce the science results. But as described above, in principle it might be fine to run them just in an alarm mode where we flag (or even delete)

data that are not trustworthy, but for precision science we actually would like to keep track continuously. The cadence of sensor readings that should be kept must be adopted to the science question and the environment of the experiment or observation at hand.

Let us finally turn to the question of the discovery potential. For that question, experiments and observations are much more alike. While for a data intense experiment, a super efficient filter system would not harm the reproducibility (at least in principle – say all we would have noted down from LEP is that there are only three lepton generations, we could decide to rebuild LEP or some ILC and measure the Z peak again, in that sense reproducibility is not an issue), but such a super effective filter definitely would get rid of any unexpected discovery. An example of an unexpected discovery from astronomy are fast radio bursts (FRBs), unexpected flashes of highly dispersed radio pulses with origin at extragalactic distances [43]. They have been discovered in archival data of the Parkes radio telescope. It was found much later on, that at least some of them seem to repeat, however, the details, especially the emission mechanism, are still unclear. One of the reasons that the discovery was possible was that data have been stored at high enough time and frequency resolution and sufficient metadata of the observation existed such that one could verify that the event found was not due to an yet unknown radio pulsar.

There are also examples of claims, which could not be verified later on, a prominent example is the claimed discovery of a magnetic monopole (postulated first by Dirac [44]) on Valentine's day in 1982 [45]. No similar event has been found since and to the best of my knowledge it was never figured out what actually caused the event. It is of course speculation, but perhaps a more detailed storage of data at various data levels would have allowed to sort this out. It is impossible to proof that it was not a monopole, all later searches however where not successful and have put stringent constraints on the cosmic abundance of such objects (see e.g. [46]).

To summarise, extensive filtering and reduction of data bears a serious threat to the discovery potential and the reproducibility of new discoveries. On the other hand massive filtering and data reduction, if applied thoughtfully, should not hamper the reproducibility of parametric measurements. The storage of all essential metadata is key for guaranteeing reproducibility, but it seems to us that a general recipe for unexpected discoveries does not exist (or is unknown to us).

6.2 WP 2 - Dynamic Filtering

We first discuss the concept of dynamic filtering in high-energy physics, before addressing an example from astrophysics.

Dynamic filtering at high-energy physics collider experiments happens at multiple levels. The PUNCH4NFDI project is supported by and linked to a number of particle or nuclear physics experiments including ATLAS, ALICE, CMS, PANDA, CBM, Belle II and LHCb. We discuss dynamic filtering primarily in the context of the LHCb experiment, in order not to overload the

description of main concepts with fine-grained comparisons. Indeed, from a conceptual point of view, the following approach is also followed also by other collider experiments to a large extend.

In the LHCb experiment, dynamic filtering is applied by the high-level trigger system. The trigger system selects events to store based on a high-level physics interpretation of the event. For example, physicists may program the system to select events which contain a class of b-hadron decay candidates. A b-hadron decay candidate is metadata, it is a dynamic interpretation of the raw event data, because it depends on detector calibration and alignment parameters that change during the data taking and are continuously adjusted. Physicists statistically define a list of decays of interest before run, but the actual filtering is dynamic.

Dynamic filtering in LHCb grew out of the need to search for rare processes of interest (signal) in a vast amount of uninteresting common events (background). The LHCb experiment employs a trigger system which consists of hard- and software that analyses the temporary detector recording of the current collision to find signal candidates. Collisions without interesting candidates are discarded. A trigger decision depends on the static physics of the process, and on the dynamic properties of the detector hardware and the beam (focus and intensity), which vary over time. Because of the latter, the trigger is a dynamic filter. The total event rate is dominated by the background rate, and limited by the computing resources of the experiment. The goal of dynamic filtering is to optimise the signal signal rate for a given background rate.

To approach the optimum, the LHCb experiment moved away from static hardware-driven trigger stages to dynamic software-driven trigger stages. The LHCb experiment [47, 48, 49] is the first LHC experiment to eliminate the hardware trigger completely, with large performance gains. The trigger efficiency for certain hadronic b-hadron decays is roughly doubled, for example [50].

Traditional hardware-driven triggers were limited to event parameters close to the capabilities of the detector hardware and only allowed comparably simple decisions (e.g. select an event which contains a particle with large transverse momentum or a large energy deposit in the calorimeter). Software-driven triggers operate directly on the physical interpretation of the raw event. They form a decision based on fully reconstructed particle decays (e.g. accept the event if it contains a reconstructed decay in a given mass window). The decision is based entirely on metadata. Hits in the detector are combined into tracks, which are further classified as particle candidates based on the particle identification data. Particle candidates are further combined to decay candidates. The association of hits with particle tracks and everything that follows is metadata.

In summary, the LHCb experiment filters events based on static rules (which rarely change) that directly describe particular classes of decays. The filtering is based entirely on metadata (decay candidates), which is extracted by a software online from the raw event data, and calibration and alignment parameters that are continuously updated online [6]. The latter are needed to obtain the sharpest peaks in the reconstructed invariant mass distribution of the decaying particle, which in turn boosts the background rejection factor.

In astrophysics experiments, dynamic filtering is mostly occurring in time-domain astronomy, but also during astroparticle experiments or when operating gravitational wave detectors. Staying with the usecase introduced above in Section 4.3, the background level in pulsar or FRB search experiments is usually highly dynamic. It consists of an extraterrestrial background and one of man-made, so called "radio frequency interference" (RFI) signals. The trigger level for single bursts will have to be raised, as the RFI level increases. Similarly, in the search for periodic pulsar signals, an increasing number of positions in the maintained list of identified candidates will be occupied by RFI signals. In such a case, RFI signals will need to be identified and removed or, alternatively, the length of the candidate list will need to be extended. Either method represents a dynamical adjustment of the applied filters that need to be recorded in the metadata.

6.3 WP 3 - Dynamic Archiving

While metadata is routinely distributed with the data, the *interpretation* is often nontrivial. Experiments producing large data volumes will endeavour to minimize this through compressing metadata to a minimal set of parameters describing the state of the detector and any processing pipeline carried out. When processing archived data it will therefore be necessary to also have access to tabulators / transformation methods which can "translate" metadata. There needs, in principle, be enough encoded information to rerun the full pipeline from decision to carry out observation, scheduling, experimental sequence, processing and selection. In other words, the purpose of dynamic archives is to give the scientists an opportunity to re-investigate archival data after having obtained additional or new knowledge, guided by the available metadata of the stored data sets.

In order to stay with the FRB usecase once more, after the first FRB was discovered in archival data [43], different archives were re-analysed and re-evaluated to discover more FRBs. This was not immediately successful, but the lessons learnt from those archival studies, and in particular from the information provided by the metadata, led to changes in the experiments and then eventually to new FRB discoveries [51].

6.4 WP 4 - Scalability

For efficient scaling of these metadata one needs to consider:

- Drastic increase of metadata volumes: For reproducibility of online selections one needs to know the underlying software chain running the dynamic filtering. Also constant updates of quality measures of archived data are expected to increase the actual size of metadata. In future astrophysics experiments like SKA, base metadata for an observation can be of the order of PB.

- Flexible data models: The data base scheme of SQL is typically good for complex queries while it is not designed for (rapidly) changing metadata nor for huge data volumes. Data volumes beyond 100 TB are suited for non-SQL schemes. The PostgreSQL format tends to have some advantages with respect to performance in comparison to SQL. Also a mixture of NoSQL and SQL databases could be an option, as for the ATLAS experiment using the Rucio framework. PostgreSQL is probably a good choice for structured persistent metadata. In case of large data, Greenplum may be used. This is an MPP architecture based on PostgreSQL. A logical database in Greenplum is an array of individual PostgreSQL databases working together to present a single database image. As a non-SQL alternative, Cassandra may be considered – a distributed, wide-column store, NoSQL database. For transient (dynamical) metadata, a NoSQL database such as MongoDB is probably preferred.

Both of these requirements have not been investigated at the scale anticipated for some future PUNCH experiments. Up to now, the size of metadata even for large datasets of LHC experiments tends not to exceed the scale of GB. Therefore we also have to analyse the scalability of existing solutions towards the metadata sizes well beyond TB to come to consistent solutions.

Referencing the pipeline of decisions, their underlying data and all related dependencies of software needs to be transferred into efficient and scalable workflows. The requirements for the scalability of metadata processing are actually complemented by those from allowing reproducible filtering workflows. Technically, we have to investigate the performance of the most relevant workflows including large sets of metadata. The objective is the processing of the latter implemented such that the runtimes and usage of resources are still acceptable. The scheme for referencing the components recursively is also outlined in 3.4. Ideally, the metadata representation of decisions should be based on pointers, zero-copies of data and scalable approaches.

6.5 WP 5 - Evaluation and validation of instrument response & characteristics

Similar to section 6.2 main concepts are exemplified here for the LHCb detector and astrophysics workflows. For the former, they can basically also be generalised to other high energy collider experiments. The LHCb detector response is validated and calibrated with measurements in special control channels and using simulation [8]. As part of the instrument response we discuss the trigger system, the tracking system (including the muon tracker), the particle identification system, and the calorimeter system.

The LHCb trigger system consists of trigger lines. Most trigger line selects candidates for a particular decay channel, e.g. $B^+ \to J/\psi K^+$ or $D^0 \to K^+\pi^-$. A few lines also select minimum-bias events with at least one reconstructed track with reduced rate (only every N-th triggered event is written to storage) or look for generic signatures of potential exotic decays (particles with high

transverse momentum). The efficiency of a trigger line could be calculated with the minimum-bias line, but the small fraction of events with b- and c-hadron decays makes this approach unfeasible. The TIS-TOS method is used instead [52]. An event is classified as Trigger on Signal (TOS) if the trigger objects (the reconstructed tracks associated to the selected decay of interest) are sufficient to trigger the event. An event is classified Trigger Independent of Signal (TIS) if it has been triggered entirely by objects not associated with the decay of interest. Some events are classified as TIS and TOS simultaneously. The efficiency of the trigger line can be calculated from these numbers [52] using data and is validated on simulation. A small bias remains since small residual correlations between TIS and TOS cannot be completely avoided, but the bias can be reduced to sub-percent level by counting the TIS/TOS events in kinematics bins of the decay candidate.

The efficiency of the tracking system for charged particles is studied with muons from the decay $J/\psi \to \mu^+\mu^-$, which has naturally low background. The LHCb tracking system consists of several sub-detectors, which are partially redundant. The efficiency of individual sub-stages is measured with a tag-and-probe technique; a partially reconstructed track (without using data from the sub-detector) and a fully reconstructed track are combined into a J/ψ candidate. The number of J/ψ is obtained from the mass distribution of the candidates with a fit of a signal-and-background mixture model. The resolution of the mass peak of J/ψ candidates is reduced when using partially reconstructed tracks, but still good. The efficiency of the sub-stage is the ratio of the number of J/ψ candidates obtained with two fully reconstructed tracks divided by the number obtained with a partially reconstructed track. The total tracking efficiency of the tracking system is the product of the efficiencies of the sub-detectors.

On a lower level, tests of hit efficiency, resolution, occupancy, and radiation damage are also performed for each sub-detector. Hit efficiencies and resolutions are obtained by extrapolating tracks reconstructed without a segment of a sub-detector into this segment. One then either tests whether the expected hit is present or measures the distance between actual and predicted hit location. The low-level tests are performed with muons or hadrons with large momentum to minimize multiple scattering.

The momentum resolution obtained for tracks from charged particle is also based on the observed width of the mass peak of $J/\psi \to \mu^+\mu^-$ decay candidates. The momentum scale is calibrated using large samples of $J/\psi \to \mu^+\mu^-$ and $B^+ \to J/\psi K^+$ decays. The central locations of the peaks are compared to world average values. The primary vertex resolution is studied by randomly splitting sets of tracks which point to the same primary vertex and reconstructing the vertex with each set. The resolution is computed from the difference of these two vertex locations.

Regular alignment of the LHCb tracking system is important to maintain high vertex and momentum resolution. Re-alignment is necessary after moving sub-detectors during maintenance, after temperature variations, when the dipole field of the LHCb magnet is reversed about twice every month, and when the central tracker is opened or closed (the central tracker is opened when the LHC

beams are defocussed to avoid sensor damage). The spatial alignment of the tracking detectors is based on optical and mechanical surveys and on the study of residuals of reconstructed tracks. A model of the effect of a misalignment on the residuals between predicted and actual hit locations in the tracking system is fitted to thousands of tracks simultaneously. These fits were initially performed with the Millipede method [53] and now with an algorithm [54] that correctly takes the effect of the magnetic field, multiple scattering and energy loss into account.

The response of the ring-imaging Cherenkov detectors, used for particle identification, is calibrated with decays that can be identified well based on their topology alone [50]. Suitable decays contain a vertex that is sufficiently separated from the primary collision vertex to reduce combinatorial background, for examples, the decays $K_s^0 \to \pi^+\pi^-$ and $\Lambda^0 \to p\pi^-$. The pure samples of pions, kaons, and protons are then used to measure the identification efficiency and misidentification rate. On a lower level, the Cherenkov angle resolution and the photoelectron yield are also monitored with well isolated tracks.

The LHCb calorimeter system consists of four components. The first two components, SPD and PS, are separated by a thin lead layer and consist each of a plane of scintillator tiles. They are used to distinguish between electrons and photons. The last two components are segmented traditional electromagnetic (ECAL) and hadronic calorimeters (HCAL). The SPD and PS are calibrated using the distinct peak in the charge response distribution that originates from minimum ionizing particles. Their efficiency is monitored with reconstructed tracks of particles with sufficient momentum to reach the calorimeter system. The ECAL calibration is performed in two steps, an initial calibration is followed by an iterative refinement. The initial calibration was original performed with a test-beam and uses a built-in LED system now. Relative calibrations between cells are improved similar to tracker alignment by identifying and adjusting consistent offsets between cells over many hits. Calibration constants are also improved by reconstructing $\pi^0 \to \gamma\gamma$ decays, in which a photon in a cell to be calibrated is combined with a photon from another cell. Calibration of the HCAL is done during bimonthly beam-stops, when two $137^C s$ sources are moved through the calorimeter with a hydraulic system. During data taking, the performance of ECAL and HCAL is monitored with the built-in LED system.

Also in astrophysical experiments, calibration and monitoring of the system parameters are essential. The additional complication – and essential difference to most high energy experiments – is that the Universe cannot be controlled. We can only watch the event, but we cannot repeat its processes. At the same time, the conditions of our experiments do change in terms of our local surrounding. In optical astronomy, the reflection of sunlight at megaconstellation satellites will provide a disturbing unwanted background signal. In radio astronomy, RFI may become severe, such that our methods of detecting celestial signals are thrown off and made useful. It is therefore important to continuously evaluate and validate the proper working conditions of our equipment and experiments. Usually, this can only be tested properly during the observations, so that corresponding

information must be captured in metadata.

6.6 Metadata and workflows in the dynamic life-cycle

There are three fundamental considerations that one can make first.

a) There is currently no standard methodology for how to handle missing data, i.e. where no metadata exists. What we have discussed would be a tiered metadata structure: observation-metadata, instrument-metadata, observatory-metadata etc. If no observations are found one instead determines whether an instrument/pipeline/algorithm was active and an observatory even open. This would require facilities to produce and share this metadata, and it would require standards for how to link metadata, i.e. how one would go from a level to the one above or below.

b) It is often hard to interpret metadata from different instruments for non-experts. For efficiency, different instruments use different fields that can capture the required information as compactly as possible. The two ways of making this data accessible by non-experts would be: Either add a lot of standard fields to produce common standard values Or distribute software together with metadata which can translate the core metadata into common properties (target, field, energy sensitivity etc). The first path has the drawback of "polluting" stored data with additional fields which might still not be sufficient for all information, while the second creates requirements for a full additional software component to be distributed and maintained.

c) The domain of irreversibility directly ties selection algorithms (whether real-time ML or human selection based) to metadata. Data which is lost (not saved) cannot be recreated, but we should be able to rerun the selection algorithm that was active at that time on e.g. simulated data.

Rather than constructing a list of metadata fields that should be present (which will always be incomplete),we could focus on two conceptual/standardization questions:

- How should different metadata tables reference each other? Could be directed to lower or higher level archives, or to e.g. follow-up observations. Can we suggest some scheme where a metadata query would automatically follow this link and gather all metadata pertinent to a query?

- How should metadata reference software, either to be used for parsing the metadata itself or for recreating a pipeline? In principle we would like a metadata file to itself suggest the software (including a specific release/tag) which should be used to parse it correctly and e.g. use in a larger scale off-line analysis.

Processing data from different experiments also needs to be considered: Joining data from different instruments for which different users might have other access rights adds a layer of complexity. However, this is not another "dimension" of complexity - each archive has some authorization and this needs to be solved. Indeed there might be multiple sensors with complex relations (the existence of one measurement from one sensor might or might not depend on what was derived from another sensor).

This emphasizes the kind of dynamic that two users making the same query can get two different results (with different information loss or completeness). Such a case cannot be excluded a priori.

6.7 Extra requirements from anomaly detection workflows

Anomaly detection is a method used for identifying rare or unusual signals in large data streams. Applications can be in physics measurements or in the operation and maintenance of physics instruments. In the latter case, it is also known as predictive maintenance, i.e. the detection of deviations from the usual behaviour well before a measurement instrument is in a critical mode of operation.

The result of the anomaly detection can be a tag of the physics event or a warning message indicating problems with the measurement device. These results will presumably become part of the regular data streams, like physics data or monitoring data.

Additional metadata may however be interesting to store. If an anomaly detection algorithm continuously updates its behaviour in order to learn from from the regular data flow it will be useful to track the internal parameter settings of the algorithm. Another example are indicators which activated the anomaly detection. Specific signatures of the data or of the instrument behaviour may only be available in the real-time data stream and not in the data recorded during normal operation.

Different logging frequencies and metadata data volumes can thus be expected for anomaly detection. These should be evaluated in specific applications, so that general aspects can be identified and implemented in metadata concepts.

6.8 Metadata storage size

Storing all metadata for all observations would quickly fill up any storage space. A hierarchical, linked metadata scheme allows a piece of information to only be stored once (e.g. configuration details of a real-time pipeline) and then referenced to by every observation which made of it use this. When data is hold in computer memory, such a scheme is typically already realized, but typically not for (meta)data on mass storage. Similarly, relying on software to interpret metadata allows the stored data to be compressed (at the expense of readability!).

The amount of metadata to be stored can be reduced with online algorithms, in which only a high-level representation of the raw data is stored. An example is the online processing system of the LHCb experiment. In particle physics, at the fundamental level, the detector measures hits (space-time points) in the tracking system and energy deposits in the calorimeter. The raw event consist of this lowest level of information and is typically very large. A tracking software combines hits and energy deposits into tracks and particle candidates, which require less storage than the raw event.

The raw event was previously stored offline (at least temporarily), since the tracking software requires calibration parameters that had to be computed in an offline step. The LHCb experiment now moved to a pure online system, in which a computing farm performs the event interpretation online on GPUs and CPUs, and combined with calibration parameters that are also produced online in parallel. The pure online approach has become feasible through the availability of cost-effective computing hardware and large bandwidth networks. The online system removes the necessity to store the raw event for standard analysis, so that large temporary storage solutions are no longer required.

In this context it is also interesting that developments at the ALICE experiment have also lead to significant data compression based on entropy encoding [55]. A lossless ANS entropy encoding is employed as the last stage for all detectors during online reconstruction. Among the involved detectors, the compression in the Time Projection Chamber is the most elaborate one, involving several steps, some of which are not lossless. In particular, a clusterizing algorithms converts the raw ADC values to hits. Hits of tracks not used for physics analysis are removed, while the remaining hits are processed by entropy-reduction steps such as the track model compression, as described in detail in [56, 57, 58].

A recent study [59] of large scale metadata storage methods and frameworks has investigated methods and frameworks for storing metadata in the petabyte range. In particular, SQL and NoSQL databases have been compared and the better scaling behavior of NoSQL databases in the Big Data domain is underlined. This study also emphasizes the need for further research, especially in the aspect of performance.

7 Towards the dynamic data life-cycle

We have described concepts for highly dynamic metadata arising in the context of irreversible data processing workflows. Our main focus is on data-intensive experiments within the PUNCH sciences, however most of the aspects in the concept could be generalized towards other fields of science also processing high data rates.

Based on raw data, higher-level metadata is build, which forms a natural *hierarchy of metadata*. Metadata is of a higher level, if its construction depends on metadata of lower level, otherwise it is of the same level. A hierarchical, linked metadata scheme would allow a piece of information to only be stored once and then referenced to. A corresponding metadata scheme must be flexible enough

to accommodate different types and numbers of decision processes. Several solutions based on flexible data schemes beyond traditional relational database already exist, as also summarized in section 4. Moreover, we have described (use) cases where limitations for existing data processing levels will be too restrictive for future extended layers or branches due to more complex workflows. Capturing the workflows of dynamic filtering/archiving shall finally enable as much reproducibility and validations as possible. Therefore, metadata must include a complete description of all algorithms involved in the pipelines/workflows.

Following-up this concept, we will investigate some of the aforementioned schemes for metadata focusing on their scaling and flexibility. Furthermore, an implementation of a data base structure is foreseen reflecting requirements from future workflows. It will also be interesting to quantifying fractions of data (filtered/persistent vs. raw/transient) in some use cases, also characterising the size of corresponding metadata. This document forms the conceptual basis for other deliverables in TA5, e.g. strategy concept for identifying highly complex signals in huge data streams (D-TA5-WP2-2) and specifying the concept of a dynamic archive (D-TA5-WP3-1). These deliverables will be the starting point for an initial implementation of a dynamic archive coupled to a pipeline for filtering/triggering. Finally, the definition of an interface to the Science Data Platform and corresponding coordination with TA4 are planned.

Acknowledgements

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – project number 460248186 (PUNCH4NFDI). Special thanks to all involved PUNCH4NFDI members.

References

[1] The PUNCH4NFDI Consortium. Punch4nfdi consortium proposal, September 2020. This is the version documenting the work plan at the proposal stage. The reduction in funding led to a re-shaping of the work programme that is documented elsewhere.

[2] NASA/EOSDIS Data Processing Levels. available at `https://science.nasa.gov/earth-science/earth-science-data/data-processing-levels-for-eosdis-data-products/`.

[3] Oliver Koepler, Torsten Schrade, Steffen Neumann, Rainer Stotzka, Cord Wiljes, Ina Blümel, Christian Bracht, Tobias Hamann, Susanne Arndt, and Johannes Hunold. Sektionskonzept Meta(daten), Terminologien und Provenienz zur Einrichtung einer Sektion im Verein Nationale Forschungsdateninfrastruktur (NFDI) e.V., October 2021.

[4] B. Stappers and M. Kramer. An Update on TRAPUM. In *MeerKAT Science: On the Pathway to the SKA*, page 9, January 2016.

[5] K. M. Rajwade, M. C. Bezuidenhout, M. Caleb, L. N. Driessen, F. Jankowski, M. Malenta, V. Morello, S. Sanidas, B. W. Stappers, M. P. Surnis, E. D. Barr, W. Chen, M. Kramer, J. Wu, S. Buchner, M. Serylak, F. Combes, W. Fong, N. Gupta, P. Jagannathan, C. D. Kilpatrick, J. K. Krogager, P. Noterdaeme, C. Núñez, J. Xavier Prochaska, R. Srianand, and N. Tejos. First discoveries and localizations of Fast Radio Bursts with MeerTRAP: real-time, commensal MeerKAT survey. 514(2):1961–1974, August 2022.

[6] LHCb collaboration. LHCb Trigger and Online Upgrade Technical Design Report, 5 2014. CERN-LHCC-2014-016, LHCB-TDR-016.

[7] CERN (Meyrin) LHCb Collaboration. Computing Model of the Upgrade LHCb experiment. Technical report, CERN, Geneva, 2018.

[8] Roel Aaij et al. Measurement of the track reconstruction efficiency at LHCb. *JINST*, 10(02):P02007, 2015.

[9] Roel Aaij et al. Measurement of prompt charged-particle production in pp collisions at $\sqrt{s} = 13$ TeV. *JHEP*, 01:166, 2022.

[10] Roel Aaij et al. Measurement of the nuclear modification factor and prompt charged particle production in p-Pb and pp collisions at $\sqrt{s_{NN}} = 5$ TeV. *Phys. Rev. Lett.*, 128(14):142004, 2022.

[11] M. Hushchyn and V. Chekalina. Particle-identification techniques and performance at LHCb in Run 2. *Nucl. Instrum. Meth. A*, 936:568–569, 2019.

[12] Christine A. Aidala, Chris Burr, Marco Cattaneo, Dillon S. Fitzgerald, Adam Morris, Sebastian Neubert, and Donijor Tropmann. Ntuple Wizard: an application to access large-scale open data from LHCb. 2 2023.

[13] R. L. Workman and Others. Review of Particle Physics. *PTEP*, 2022:083C01, 2022. available at `https://pdg.lbl.gov/2023/reviews/rpp2022-rev-cosmic-rays.pdf`.

[14] Weiwei Chen, Ewan Barr, Ramesh Karuppusamy, Michael Kramer, and Benjamin Stappers. Wide Field Beamformed Observation with MeerKAT. *Journal of Astronomical Instrumentation*, 10(3):2150013–178, January 2021.

[15] T. J. Khoo, A. Reinsvold Hall, N. Skidmore, S. Alderweireldt, J. Anders, C. Burr, W. Buttinger, P. David, L. Gouskos, L. Gray, S. Hageböck, A. Krasznahorkay, P. Laycock, A. Lister, Z. Marshall, A. B. Meyer, T. Novak, S. Rappoccio, M. Ritter, E. Rodrigues, J. Rumsevicius, L. Sexton-Kennedy, N. Smith, G. A. Stewart, and S. Wertz. Constraints on future analysis metadata systems in high energy physics. *Computing and Software for Big Science*, 6(1), jul 2022.

[16] DataCite Metadata Working Group. DataCite Metadata Schema Documentation for the Publication and Citation of Research Data. Version 4.3. DataCite e.V. 2019.

[17] M. A. C. Johnson et al. Astronomical Pipeline Provenance: A Use Case Evaluation. *13th International Workshop on Theory and Practice of Provenance (TaPP 2021)*. available at `https://www.usenix.org/conference/tapp2021/presentation/johnson`.

[18] Markus Demleitner. *Practical Interoperability in the Virtual Observatory, E-Science-Tage 2021, Share Your Research Data*. heiBOOKS, Heidelberg, 2022.

[19] E. W. Greisen and M. R. Calabretta. Representations of world coordinates in FITS. *Astronomy and Astrophysics*, 395:1061–1075, December 2002.

[20] Eric H. Neilsen and James Simone. Lattice QCD Data and Metadata Archives at Fermilab and the International Lattice Data Grid. 2004.

[21] Frithjof Karsch, Hubert Simma, and Tomoteru Yoshie. The International Lattice Data Grid – towards FAIR data. *PoS*, LATTICE2022:244, 2023.

[22] Odier, Jérôme, Lambert, Fabian, and Fulachier, Jérôme. The atlas metadata interface (ami) 2.0 metadata ecosystem: new design principles and features. *EPJ Web Conf.*, 214:05046, 2019.

[23] Odier, Jérôme, Fulachier, Jérôme, and Lambert, Fabian. Deploying and administrating the atlas metadata interface (ami) 2.0 ecosystem. *EPJ Web Conf.*, 245:04040, 2020.

[24] Martin Barisits et al. Rucio: Scientific data management. *Computing and Software for Big Science*, 3(1), 2019.

[25] Mario Lassnig, Martin Barisits, and D Christidis. Experiences in exascale scientific data management. *IEEE Data Eng. Bull.*, 43:9–22, 2020.

[26] Eric Vaandering. Transitioning CMS to Rucio Data Managment. *EPJ Web Conf.*, 245:04033, 2020.

[27] D2.2 Assessment and analysis of performance of the first pilot data lake. available at `https://www.projectescape.eu/sites/default/files/ESCAPE-D2.2-v1.0.pdf`.

[28] D. Barberis. Modern BigData technologies to store and access metadata for the ATLAS experiment. *Afr. Rev. Phys.*, 13:0009, 2018.

[29] Jalal Mostafa, Sara Wehbi, Suren Chilingaryan, and Andreas Kopmann. SciTS: A Benchmark for Time-Series Database in Scientific Experiments and Industrial Internet of Things. 4 2022.

[30] Christopher D. Jones, Kyle Knoepfel, Paolo Calafiura, Charles Leggett, and Vakhtang Tsulaia. Evolution of HEP Processing Frameworks. In *2022 Snowmass Summer Study*, 3 2022.

[31] Timo Bingmann, Michael Axtmann, Emanuel Jöbstl, Sebastian Lamm, Huyen Chau Nguyen, Alexander Noe, Sebastian Schlag, Matthias Stumpp, Tobias Sturm, and Peter Sanders. Thrill: High-Performance Algorithmic Distributed Batch Data Processing with C++. 2016.

[32] Shen Huijie and Huang Tingwei. Spark for HPC: a comparison with MPI on compute-intensive applications using Monte Carlo method, 2018. available at `https://uu.diva-portal.org/smash/get/diva2:1347863/FULLTEXT01.pdf`.

[33] Victoria Tokareva. Metadata curation use cases in astroparticle physics, October 2022. Acknowledgement: This work was partially supported by DFG fund "NFDI 39/1" for the PUNCH4NFDI consortium.

[34] Thomas Schörner-Sadenius, Harry Enke, Andreas Haungs, Kilian Schwarz, Markus Demleitner, Achim Geiser, Lukas Heinrich, Michael Kramer, Gernot Maier, Dominik Schwarz, Hendrik Seitz-Moskaliuk, Hubert Simma, Michael Sterzik, and Stefan Typel. Survey of Open Data Concepts Within Fundamental Physics: An Initiative of the PUNCH4NFDI Consortium, March 2022.

[35] CERN Open Data Portal. available at `https://opendata.cern.ch`.

[36] Zaven Akopov et al. Status Report of the DPHEP Study Group: Towards a Global Effort for Sustainable Data Preservation in High Energy Physics. 5 2012.

[37] CMS collaboration. Doublemu primary dataset sample in raw format from runa of 2011 (from /doublemu/run2011a-v1/raw). cern open data portal. 2019. available at `http://opendata.cern.ch/record/47`.

[38] Britzger, Daniel, Levonian, Sergey, Schmitt, Stefan, South, David, and for the H1 Collaboration. Preservation through modernisation: The software of the H1 experiment at HERA. *EPJ Web Conf.*, 251:03004, 2021.

[39] Georges Aad et al. Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC. *Phys. Lett. B*, 716:1–29, 2012.

[40] Serguei Chatrchyan et al. Observation of a New Boson at a Mass of 125 GeV with the CMS Experiment at the LHC. *Phys. Lett. B*, 716:30–61, 2012.

[41] Dillon Brout, Dan Scolnic, Brodie Popovic, Adam G. Riess, Anthony Carr, Joe Zuntz, Rick Kessler, Tamara M. Davis, Samuel Hinton, David Jones, W. D'Arcy Kenworthy, Erik R. Peterson, Khaled Said, Georgie Taylor,

Noor Ali, Patrick Armstrong, Pranav Charvu, Arianna Dwomoh, Cole Meldorf, Antonella Palmese, Helen Qu, Benjamin M. Rose, Bruno Sanchez, Christopher W. Stubbs, Maria Vincenzi, Charlotte M. Wood, Peter J. Brown, Rebecca Chen, Ken Chambers, David A. Coulter, Mi Dai, Georgios Dimitriadis, Alexei V. Filippenko, Ryan J. Foley, Saurabh W. Jha, Lisa Kelsey, Robert P. Kirshner, Anais Möller, Jessie Muir, Seshadri Nadathur, Yen-Chen Pan, Armin Rest, Cesar Rojas-Bravo, Masao Sako, Matthew R. Siebert, Mat Smith, Benjamin E. Stahl, and Phil Wiseman. The Pantheon+ Analysis: Cosmological Constraints. *Astrophys. J.*, 938(2):110, October 2022.

[42] A. A. Penzias and R. W. Wilson. A Measurement of Excess Antenna Temperature at 4080 Mc/s. *Astrophys. J.*, 142:419–421, July 1965.

[43] D. R. Lorimer, M. Bailes, M. A. McLaughlin, D. J. Narkevic, and F. Crawford. A Bright Millisecond Radio Burst of Extragalactic Origin. *Science*, 318(5851):777, November 2007.

[44] Paul Adrien Maurice Dirac. Quantised singularities in the electromagnetic field,. *Proc. Roy. Soc. Lond. A*, 133(821):60–72, 1931.

[45] Blas Cabrera. First Results from a Superconductive Detector for Moving Magnetic Monopoles. *Phys. Rev. Lett.*, 48:1378–1380, 1982.

[46] M. Ambrosio et al. Final results of magnetic monopole searches with the MACRO experiment. *Eur. Phys. J. C*, 25:511–522, 2002.

[47] P. R. Barbosa-Marinho et al. LHCb online system technical design report: Data acquisition and experiment control, 12 2001. CERN-LHCC-2001-040.

[48] LHCb collaboration. Addendum to the LHCb online system technical design report, 11 2005. CERN-LHCC-2005-039, CERN-LHCC-2001-040-ADD-1.

[49] A. Augusto Alves, Jr. et al. The LHCb Detector at the LHC. *JINST*, 3:S08005, 2008.

[50] Roel Aaij et al. LHCb Detector Performance. *Int. J. Mod. Phys. A*, 30(07):1530022, 2015.

[51] D. Thornton, B. Stappers, M. Bailes, B. Barsdell, S. Bates, N. D. R. Bhat, M. Burgay, S. Burke-Spolaor, D. J. Champion, P. Coster, N. D'Amico, A. Jameson, S. Johnston, M. Keith, M. Kramer, L. Levin, S. Milia, C. Ng, A. Possenti, and W. van Straten. A Population of Fast Radio Bursts at Cosmological Distances. *Science*, 341(6141):53–56, July 2013.

[52] S Tolk, J Albrecht, F Dettori, and A Pellegrino. Data driven trigger efficiency determination at LHCb, 2014. LHCb-PUB-2014-039, CERN-LHCb-PUB-2014-039.

[53] Volker Blobel and Claus Kleinwort. A New method for the high precision alignment of track detectors, 6 2002. DESY-02-077.

[54] J. Amoraal et al. Application of vertex and mass constraints in track-based alignment. *Nucl. Instrum. Meth. A*, 712:48–55, 2013.

[55] David Rohr. Usage of GPUs in ALICE online and offline processing during LHC run 3. *EPJ Web of Conferences*, 251:04026, 2021.

[56] David Rohr. Tracking performance in high multiplicities environment at alice, 2017.

[57] David Rohr. Global track reconstruction and data compression strategy in alice for lhc run 3, 2019.

[58] Michael Lettrich and. Fast entropy coding for ALICE run 3. In *Proceedings of 40th International Conference on High Energy physics — PoS(ICHEP2020)*. Sissa Medialab, feb 2021.

[59] Tim Oelkers. Overview of petabyte-scale metadata storage methods and frameworks. available at `https://results.punch4nfdi.de/files/documents/Metadata_PUNCH_Oelkers.pdf`.