Karlsruhe Institute of Technology

Astroparticle Data Life Cycle

Physical aspects of data format

Dmitriy Kostunin

March 6, 2018

INSTITUT FÜR KERNPHYSIK

Disclaimer

- This talk describes the problem from the phisicist's point of view (unrelated to stucture of metadata, etc)
- Hereafter only air-shower data is discussed (e.g. particles, light, radio, etc)
- Many things are already implemented, we only need to formalize them and write corresponding software
- The discussed approach will be presented at ECRS2017 conference by D. Kostunin
- This talk is the mixture of physics, math, programming, etc and might be not very clear

The problem

Karlsruhe Institute of Technology

Initial idea

- APPDS is aimed to operate with big data, i.e. it is neccessary to provide flexible interface for selection and joins (otherwise end-user have to download more data and perform cuts locally)
  - Database should understand sophisticated queries
  - End-user should be able to upload custom macroses

Two main requirements from physicist

- More sophisticated cuts (in most interfaces only $\Theta(x)$ cut is implemented)
- More sophisticated left joins (virtual observables, recalibration on the fly, etc)

The example of custom cut

- The Earth's magnetic field $B(x, y, z, t)$ plays important role in radio measurements of cosmic rays
- People use coordinate system transfomations or geometrical operations to handle this

Very common query in Tunka-Rex database:

```
select * from reconstruction where \
energy*sin(geomag_angle(71.7571,-2.7625,zenith,azimuth))>=0.1;
```

where `geomag_angle` is a custom SQL macros

```
create function geomag_angle(incl double, decl double, zen double, az dou
begin
        declare th1 double;
        declare th2 double;
        declare ph1 double;
        declare ph2 double;
        declare x1 double;
        declare y1 double;
        declare z1 double;
        declare x2 double;
        declare y2 double;
        declare z2 double;
        declare scalar double;
        declare angle double;
        set th1 = 3.0*pi()/2.0 + incl*pi()/180.0;
        set ph1 = pi()/2.0 - decl*pi()/180.0;
        set th2 = zen*pi()/180.0;
        set ph2 = az*pi()/180.0;
        set x1 = sin(th1)*cos(ph1);
        set x2 = sin(th2)*cos(ph2);
        set y1 = sin(th1)*sin(ph1);
        set y2 = sin(th2)*sin(ph2);
        set z1 = cos(th1);
        set z2 = cos(th2);
        set scalar = x1*x2 + y1*y2 + z1*z2;
        set angle = acos(scalar);
        return angle;
end;
//
```

Description of data

It is suggested to consider full description of air-shower using CORSIKA conventions:

- air-shower consists of a number of layers $X_0, X_2, ..., X_g$, where X_0 and X_g is a first interaction and the ground, repectively
- each layer is described by the set of particles $P(E, p, t)$, where E, p, t is an energy, momentum and type, respectively

The other observables are derived from the basic ones via *mapping*:

- electromagnetic energy $\int P(E, p, t = e, \gamma)\mathrm{d}E$
- detected muons $P_{X=X_g}(E > E_{\mathrm{th}}, p, t = \mu)$
- Top-down mapping in case of simulations, bottom-up in case of real measurements

Types of data and mapping (arbitrary)

Observables

- "Direct" D (e, μ, γ, etc)
- "Reconstructed" R (E, $X_{\max}$, $\ln A$, etc)
- "Virtual" V ("ideal measurements", EM component, hadr.component, neutrino, etc)

Mapping

- Calibration (hardware response) $D \to R$: provided by experiments, but can be change on the flight by the end-user
- Models $V \rightleftharpoons$: provided by experiments (reconstruction methods), theorists (proposed methods), end-users.
  This mapping can be implemented in the frame of our system, indexed with DOI and be cited in papers.
  The external reader can test this mapping (methods) using DOI from paper and simulations (for example)

Example of application

KG-Tunka joint paper "A comparison of the cosmic-ray energy scales of Tunka-133 and KASCADE-Grande via their radio extensions Tunka-Rex and LOPES"

`DOI: 10.1016/j.physletb.2016.10.031`

- Radio is left joined to host detector with `timestamp1=timestamp2`
- LOPES and Tunka-Rex data are recalibrated "on the fly": $D_1 \to D_1'$, $D_2 \to D_2'$ to be consistent
- LOPES and Tunka-Rex events were re-simulated with CORSIKA: $D_1 \to V_1$ and $D_2 \to V_2$
- Finally, KG and Tunka-133 were joined via modified radio observables: `D1c=D2c or V1=V2`

Summary

- The idea is to follow CORSIKA conventions for air-shower description (and to be more consistent with simulations)
- It is suggested to integrate calibration and reconstruction into aggregation service and call this *mapping*
- Allow end-user to implement his/her own macroses for cuts and joining
- Optimize APPDS to support and execute these macroses