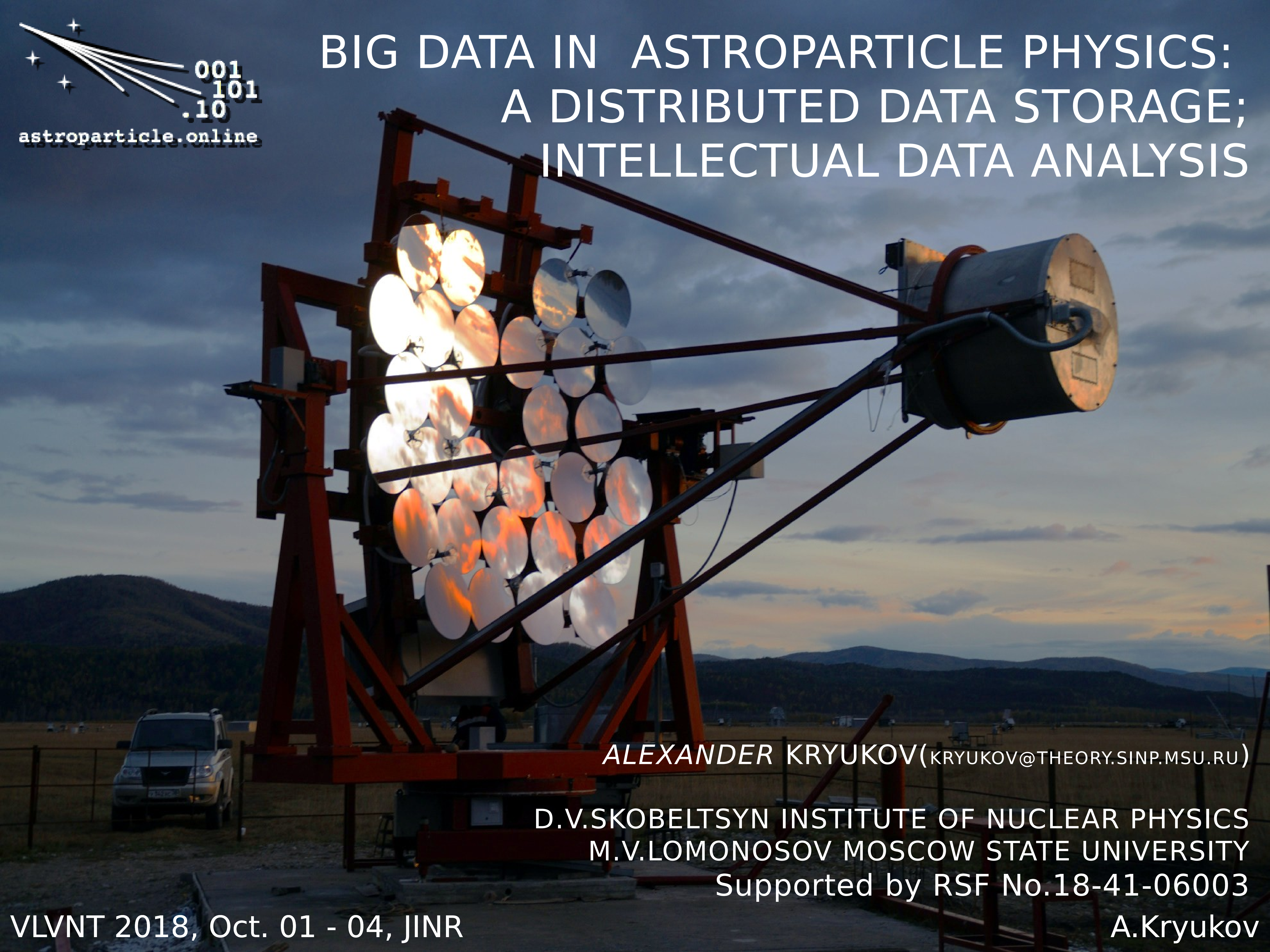


BIG DATA IN ASTROPARTICLE PHYSICS: A DISTRIBUTED DATA STORAGE; INTELLECTUAL DATA ANALYSIS

ALEXANDER KRYUKOV(KRYUKOV@THEORY.SINP.MSU.RU)

D.V.SKOBELTSYN INSTITUTE OF NUCLEAR PHYSICS
M.V.LOMONOSOV MOSCOW STATE UNIVERSITY

Supported by RSF No.18-41-06003

astroparticle.online

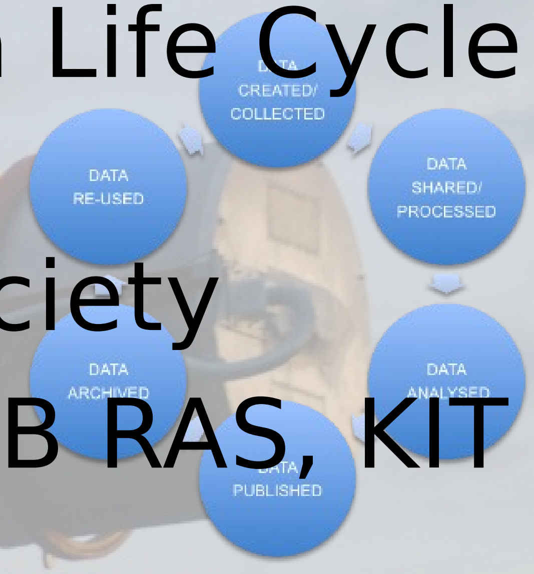
OUTLINE

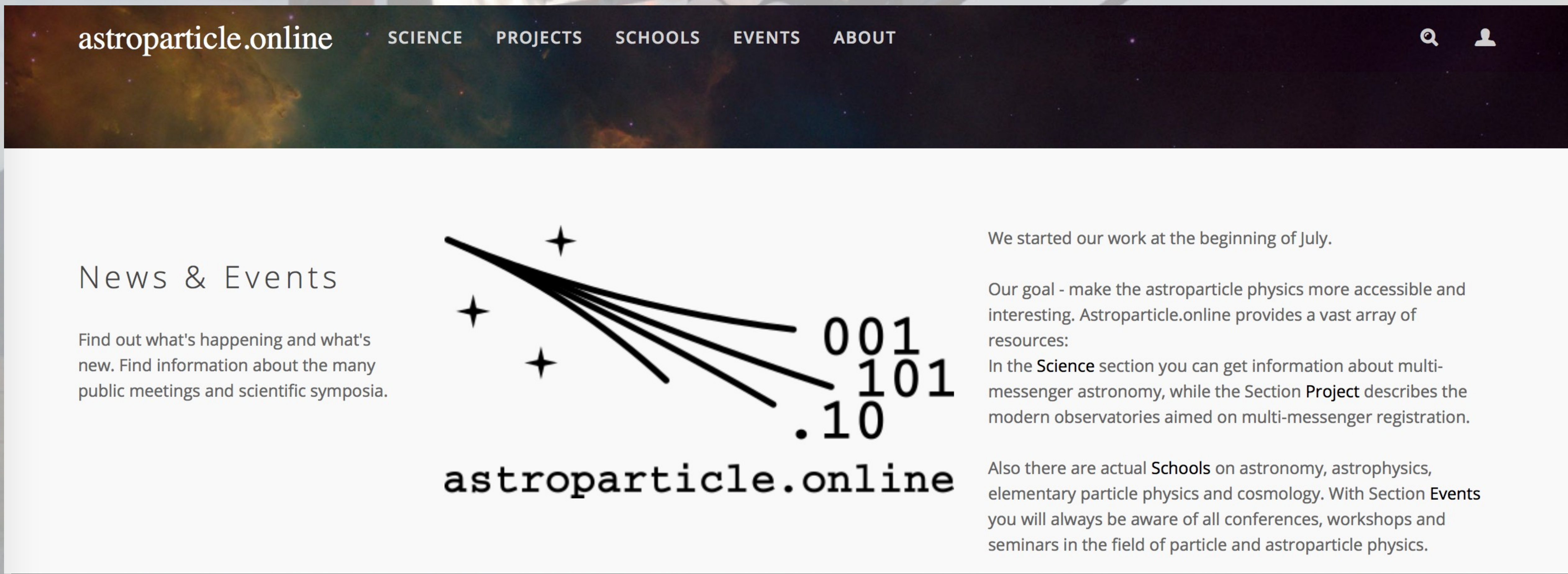
- **Introduction. Karlsruhe-Russian Astroparticle Data Life Cycle Initiative**

  Part I. A distributed data storage for astroparticle physics
- **Part II. Convolution Neural Network for particle identification**
- **Conclusions**

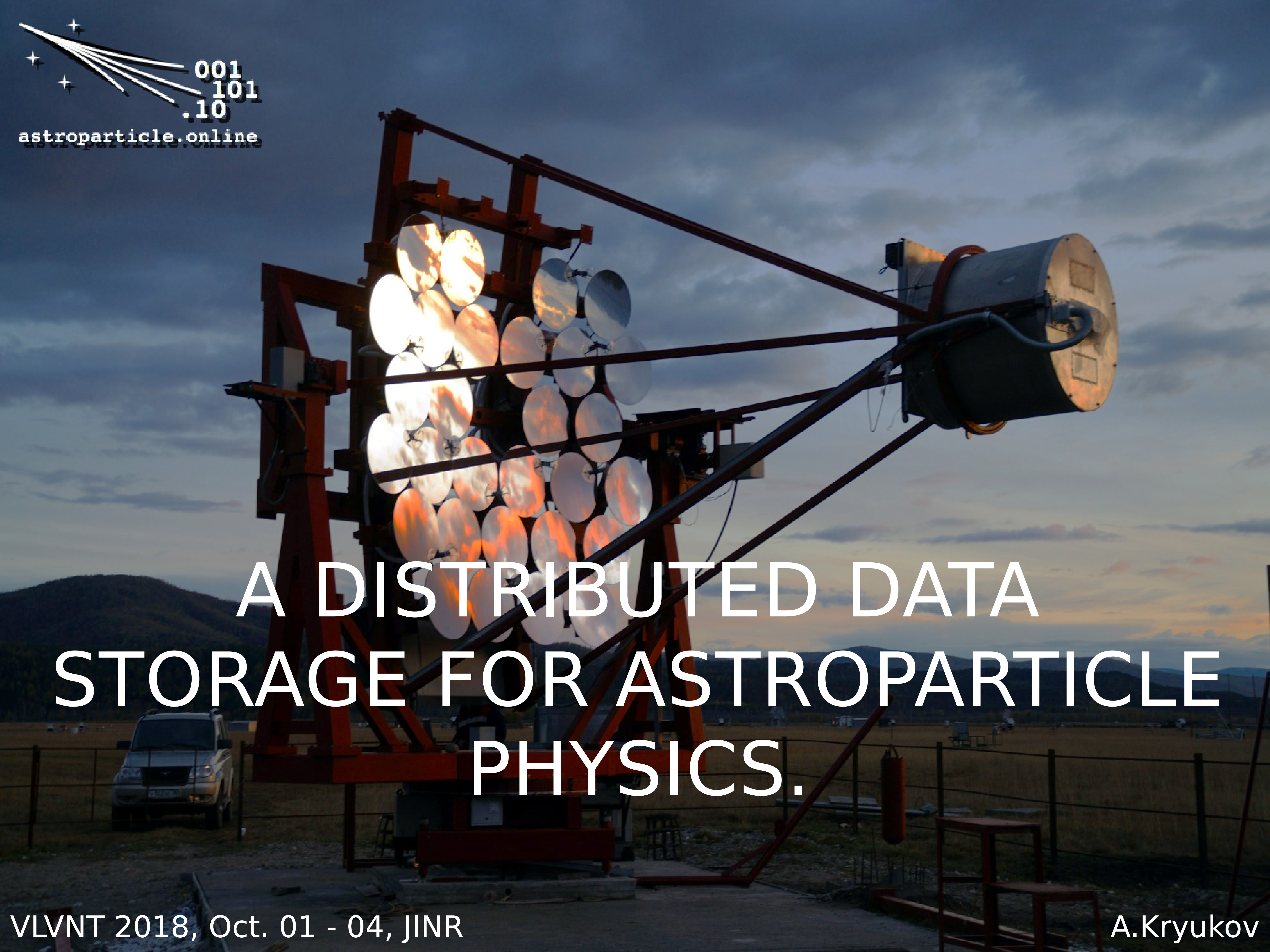
ASTROPARTICLE.ONLINE

- Karlsruhe-Russian Astroparticle Data Life Cycle Initiative
- Supported by RSF and Helmholtz Society
- Participants: SINP MSU, ISU, ISDCT SB RAS, KIT





astroparticle.online



A DISTRIBUTED DATA STORAGE FOR ASTROPARTICLE PHYSICS.

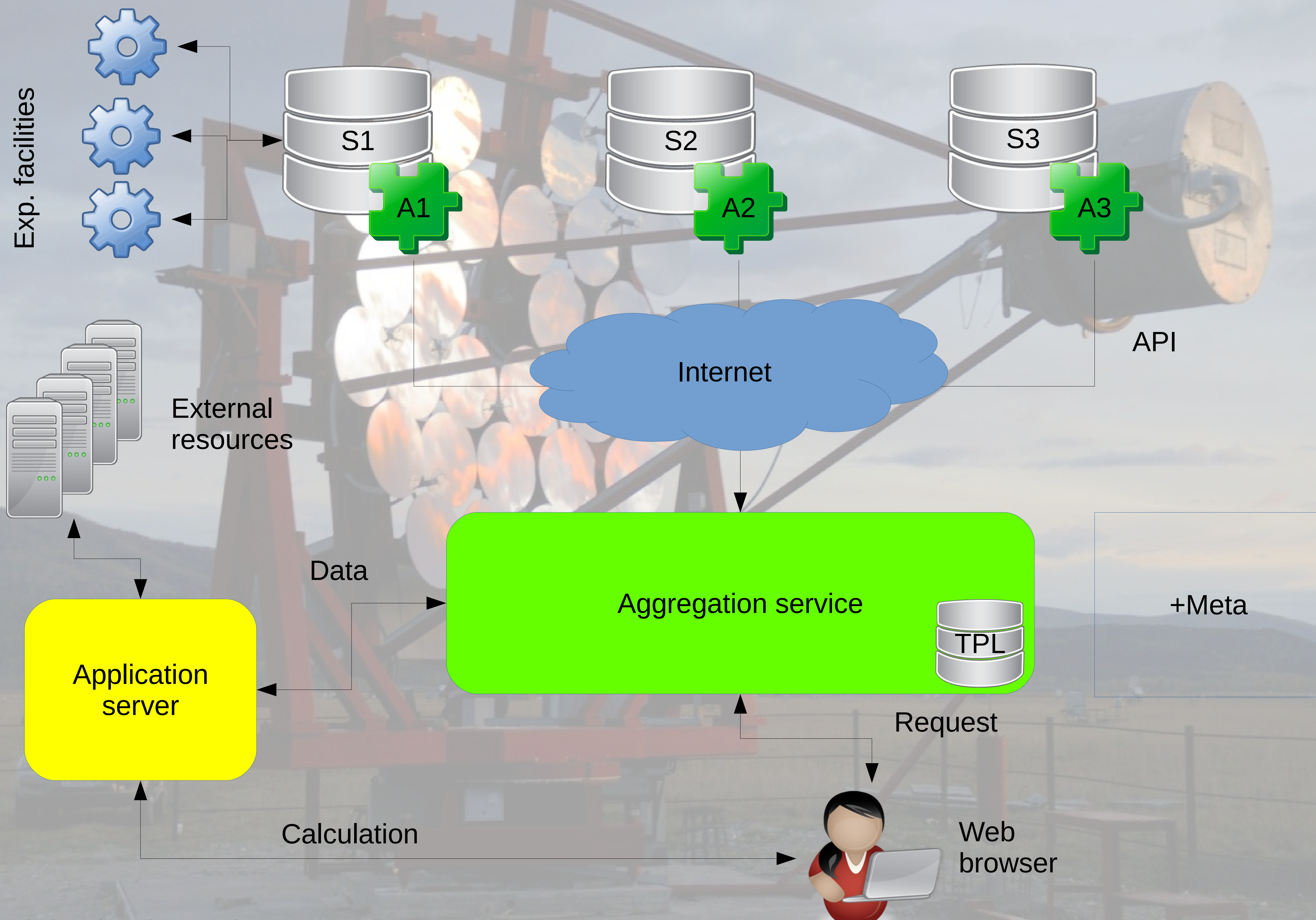
VLVNT 2018, Oct. 01 - 04, JINR

A.Kryukov

REQUIREMENTS FOR THE DATA STORAGE

- Multiple experiments (TAIGA, KASCADE, etc.)
- Hundreds of terabytes and more of raw data at each site
- Remote access to data as local file systems
- On-demand data transfer by requests only
- Automatic real-time updates
- No change to existing site infrastructure, only add-ons

Storage architecture



POSSIBLE SOLUTIONS

CernVM File system

APACHE hadoop

OpenAFS

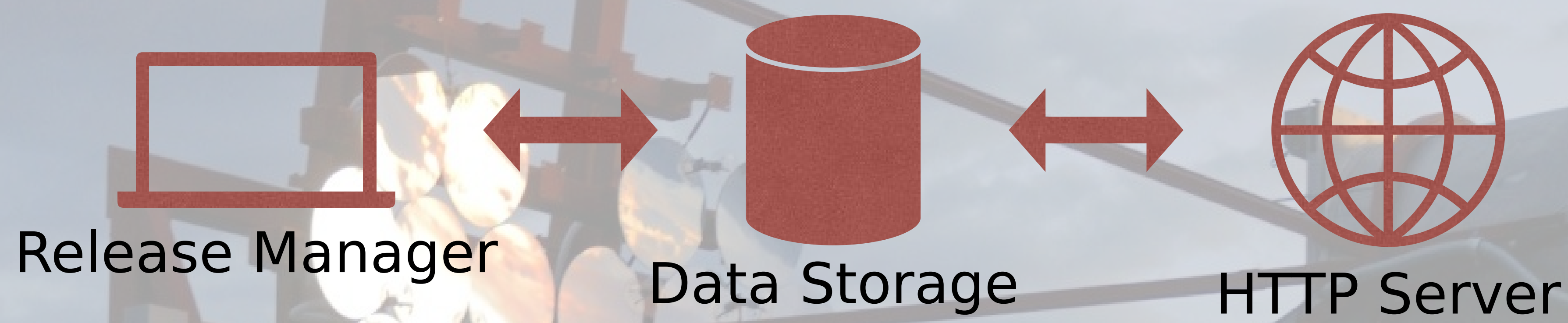
CERNVM-FS

- Data are left untouched in their own file system
- CernVM-FS indexes the data and changes, stores only the metadata (indices, checksums, locations, etc.) and data tree
- CernVM-FS uses HTTP as the data transfer protocol, so there’s no firewall problem
- Data transfer starts only on actual reads
- Multilevel cache-proxy servers

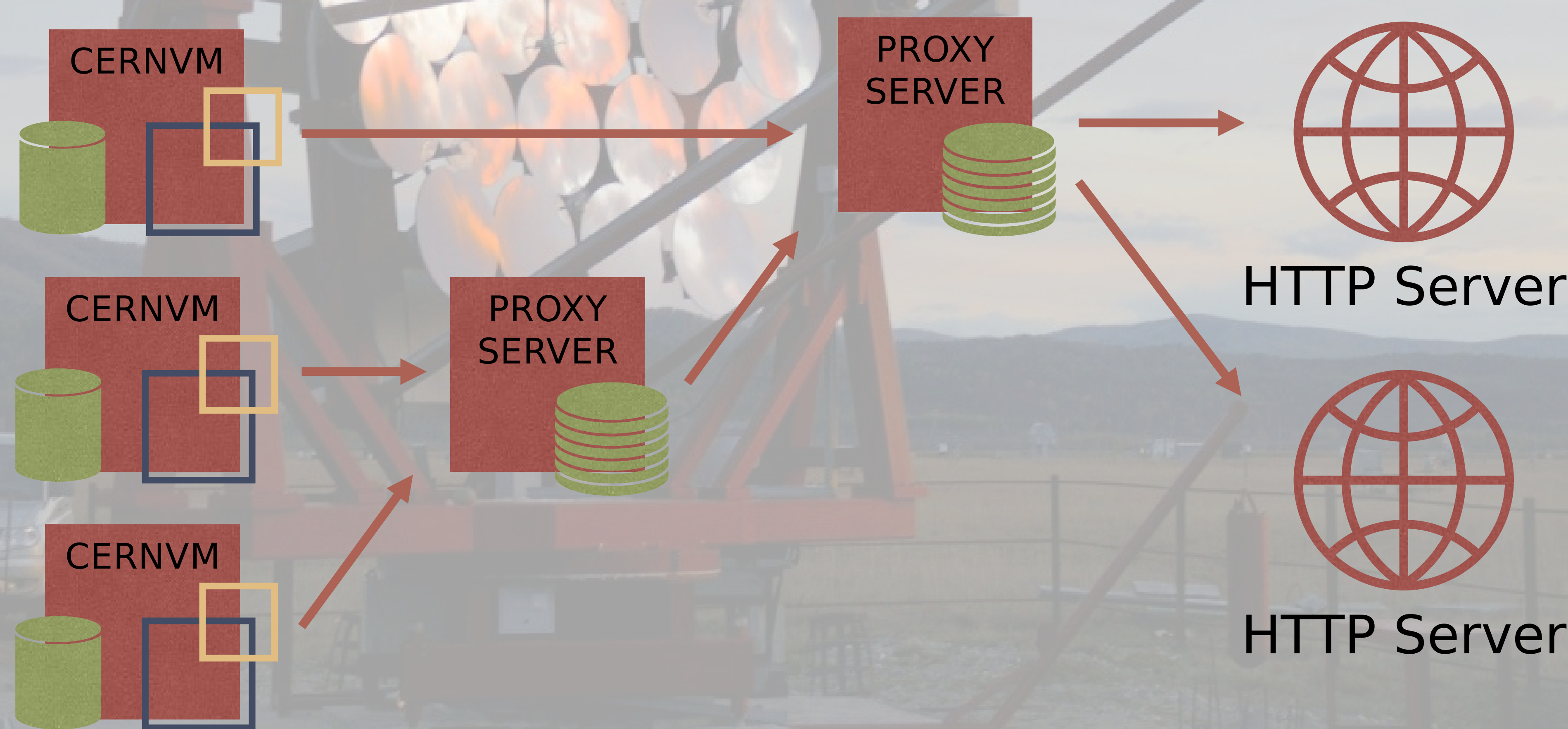
astroparticle.online

CERNVM-FS

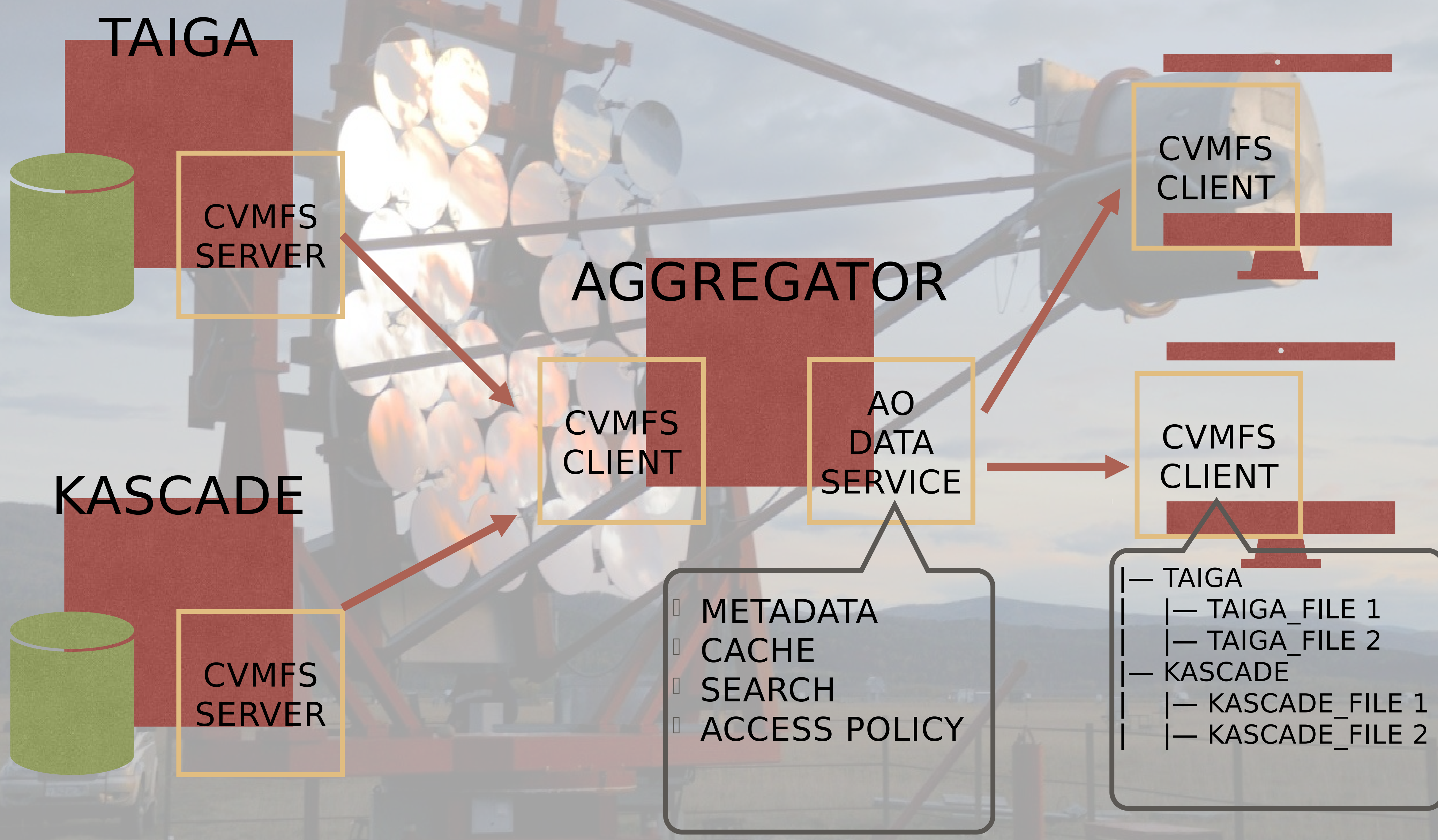
DATA UPDATE



DATA DISTRIBUTION



CERNVM-FS



TAIGA
CVMFS SERVER
KASCADE
CVMFS SERVER
AGGREGATOR
CVMFS CLIENT
AO DATA SERVICE
METADATA
CACHE
SEARCH
ACCESS POLICY
CVMFS CLIENT
CVMFS CLIENT
|— TAIGA
| |— TAIGA_FILE 1
| |— TAIGA_FILE 2
|— KASCADE
| |— KASCADE_FILE 1
| |— KASCADE_FILE 2

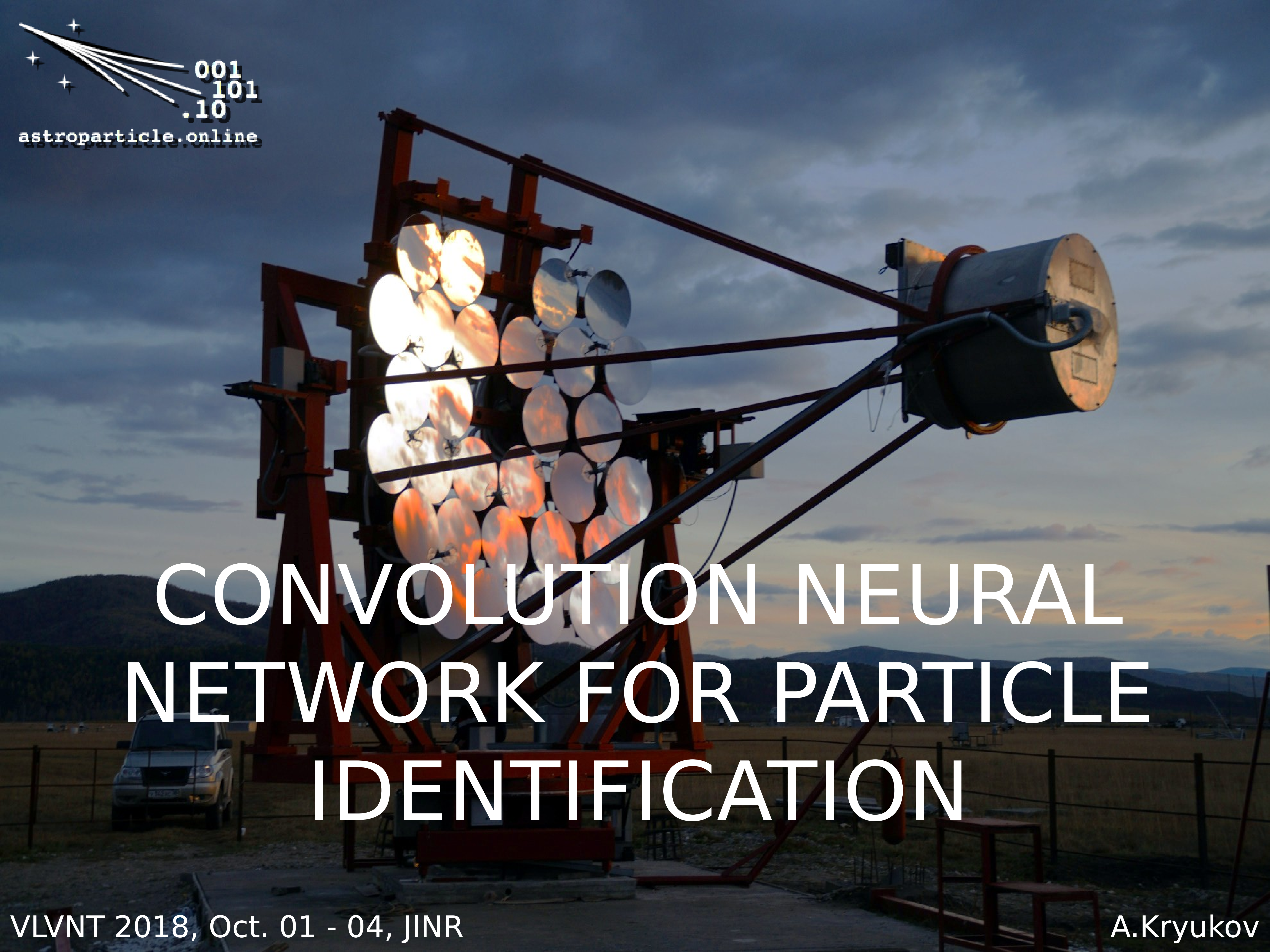
CURRENT STATUS

- ✓ Used CernVM-FS to export the existing data storage of each site as is without changing the file system
- ✓ Merged different data trees to a single one at the aggregation server level
- Metadata search and API (in progress)
- Access policy (in progress. Currently, the whole data tree is accessible for everyone)

FUTURE WORK

- Sub-tree export (build a CVM-FS middleware module or an independent bridging module?)
- Data access policy and API (RESTful API or GraphQL?)
- Metadata indexing and parameterized search (RDBMS (PostgreSQL) or NoSQL (column-based or row-based)?)
- HDFS-prototype and AFS-prototype
- Benchmark

astroparticle.online

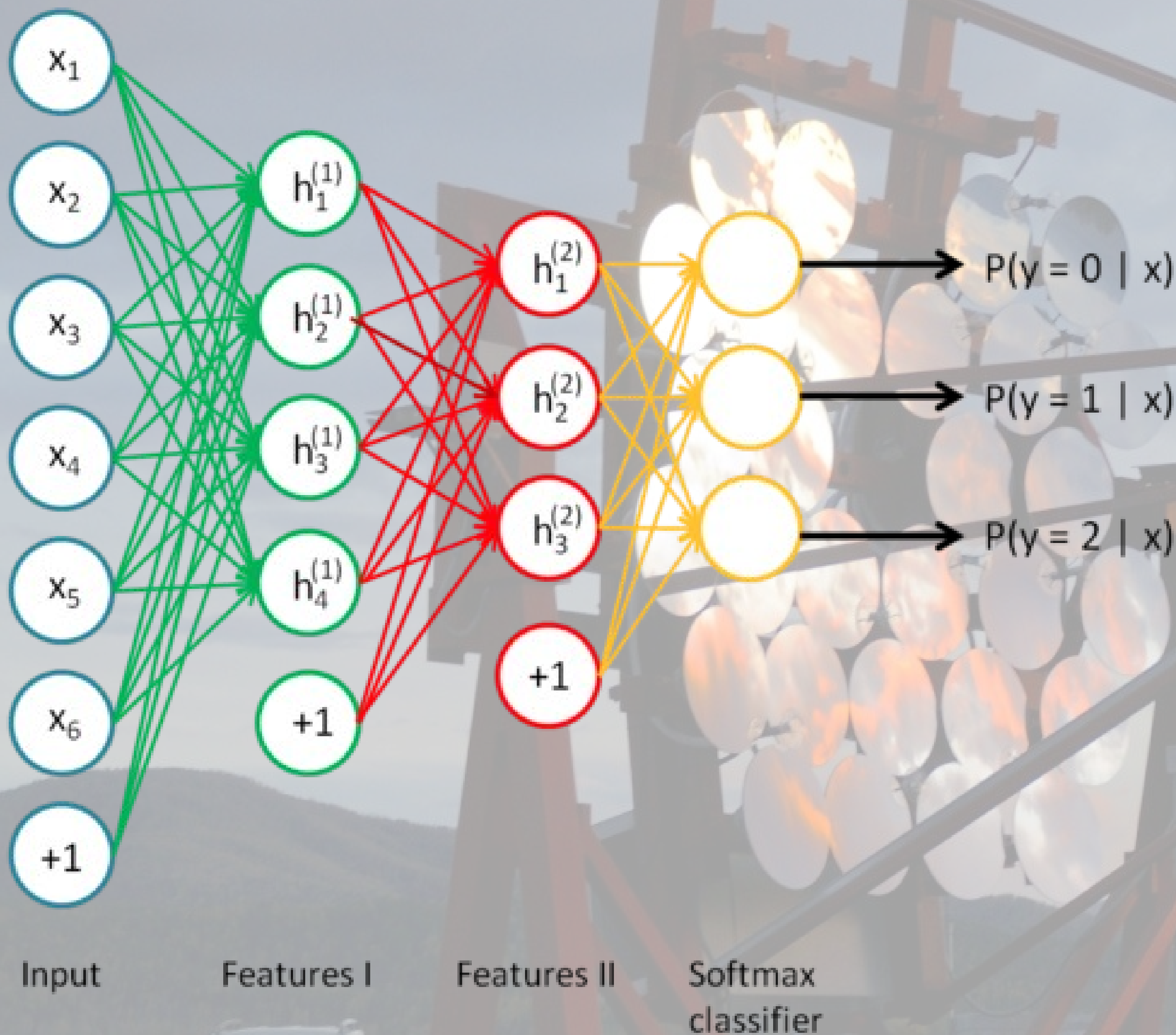


CONVOLUTION NEURAL NETWORK FOR PARTICLE IDENTIFICATION

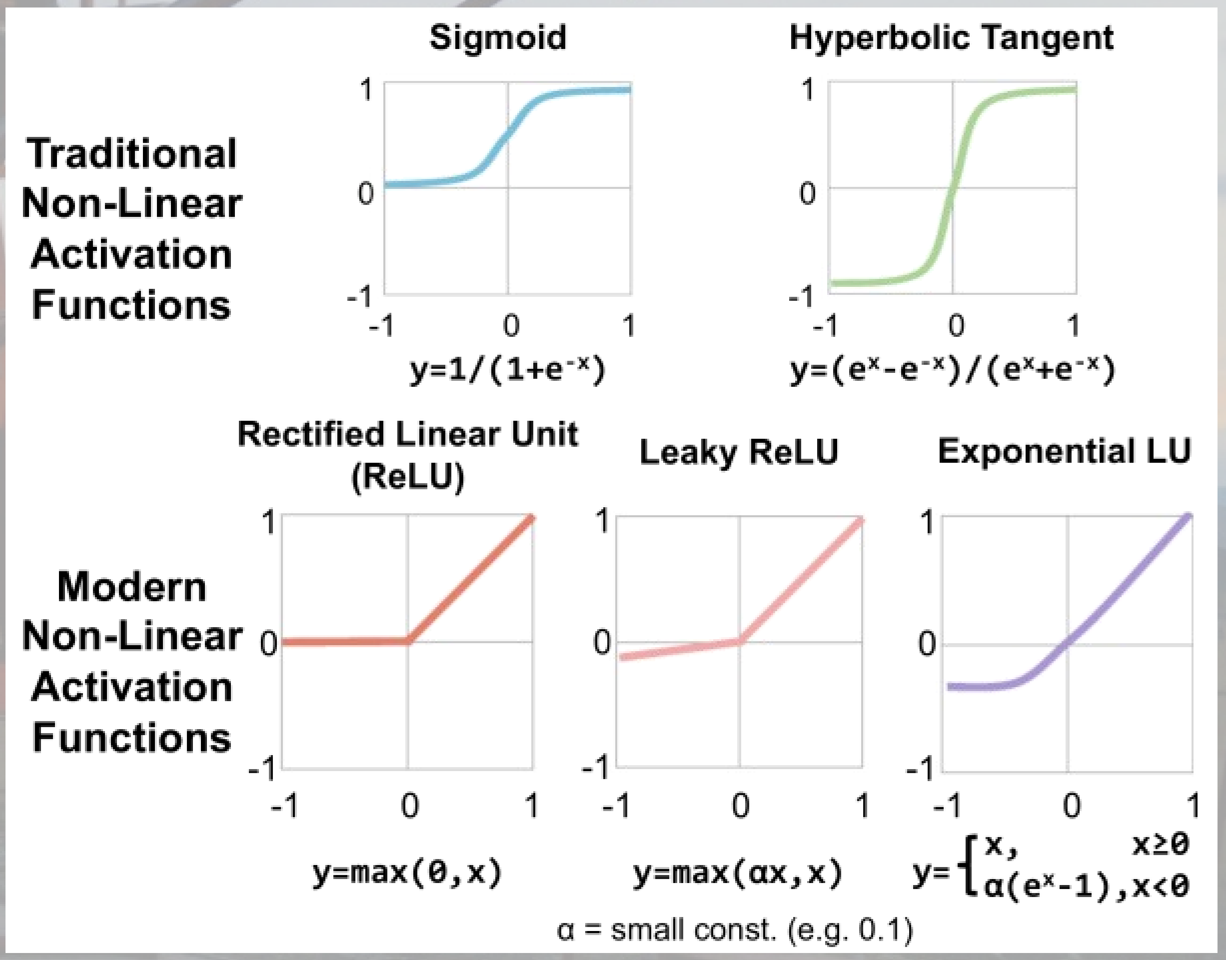
VLVNT 2018, Oct. 01 - 04, JINR

A.Kryukov

Deep Learning. Neural Networks

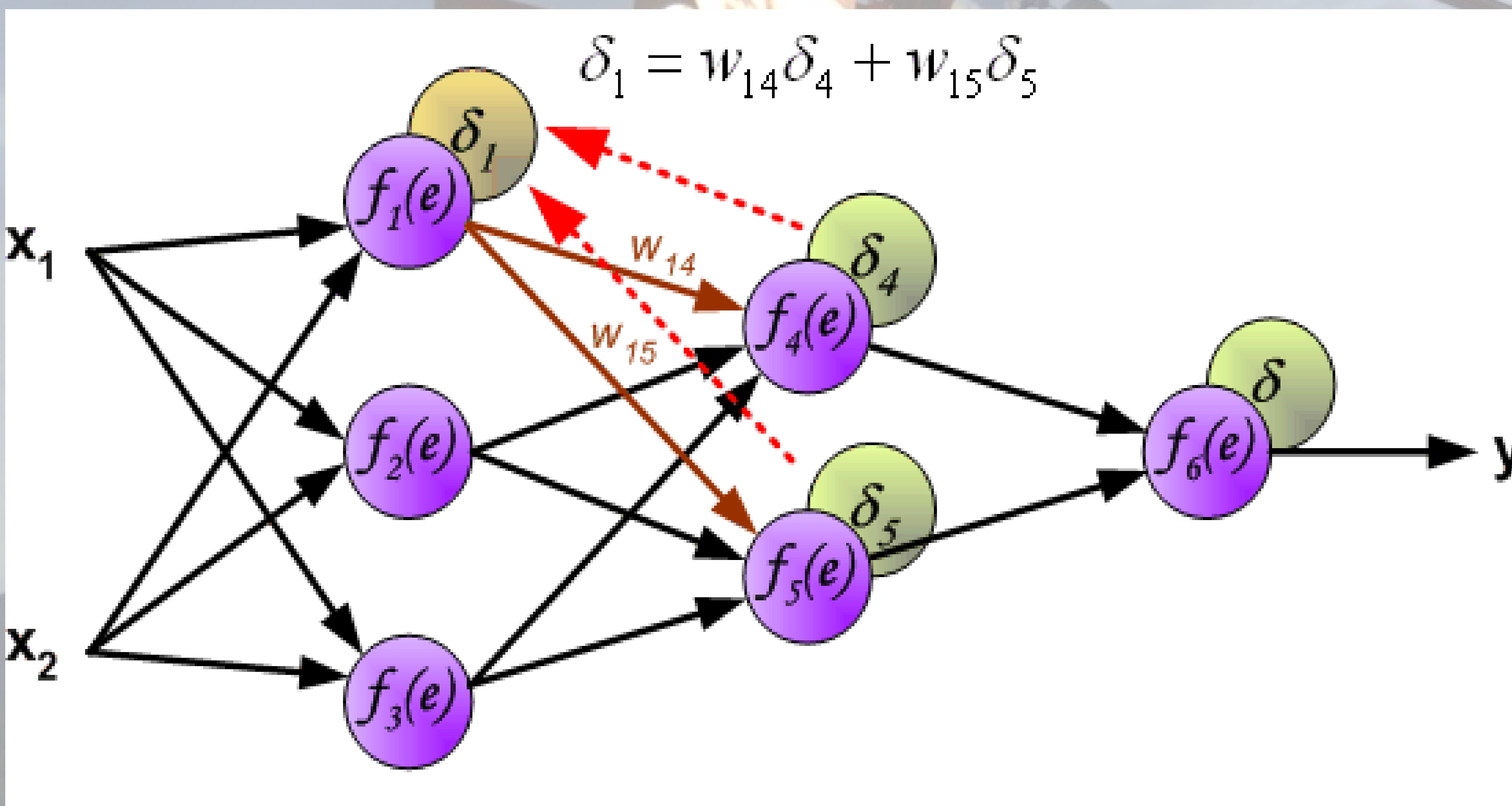


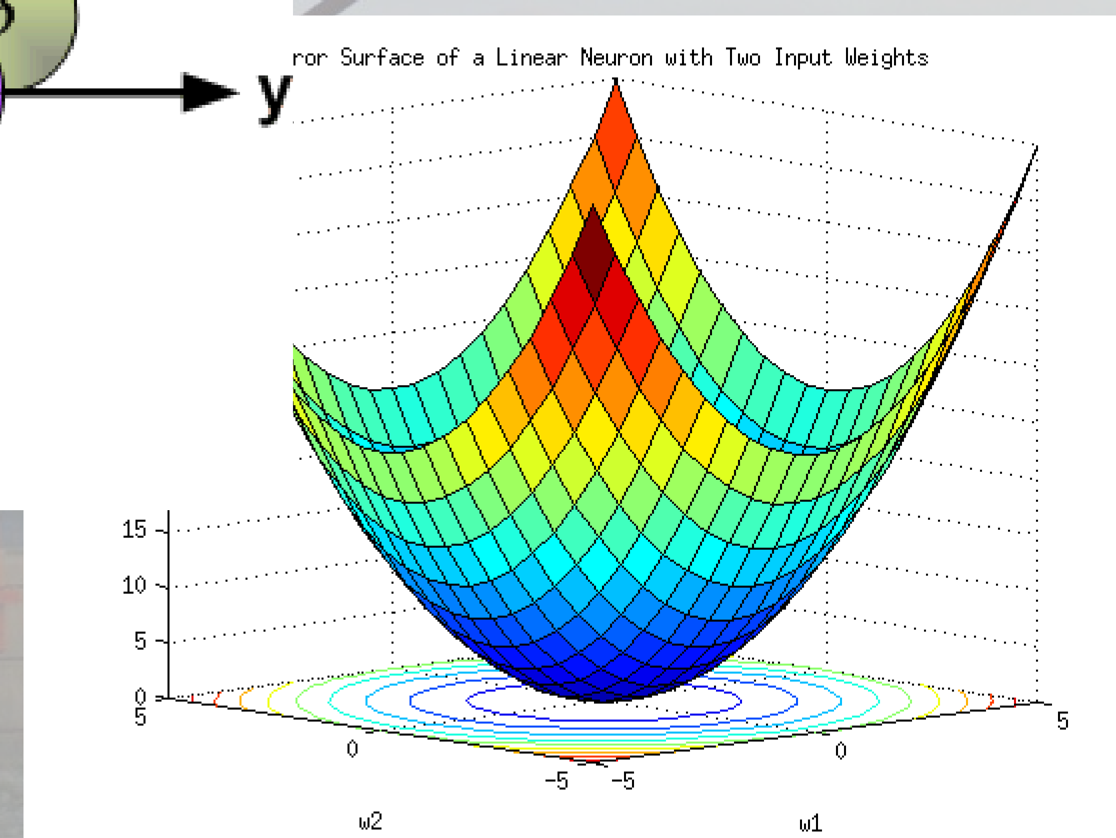
- Deep learning use large NN with some hidden layers.
- A lot of activation functions are used.



Deep Learning. Error back-propagation

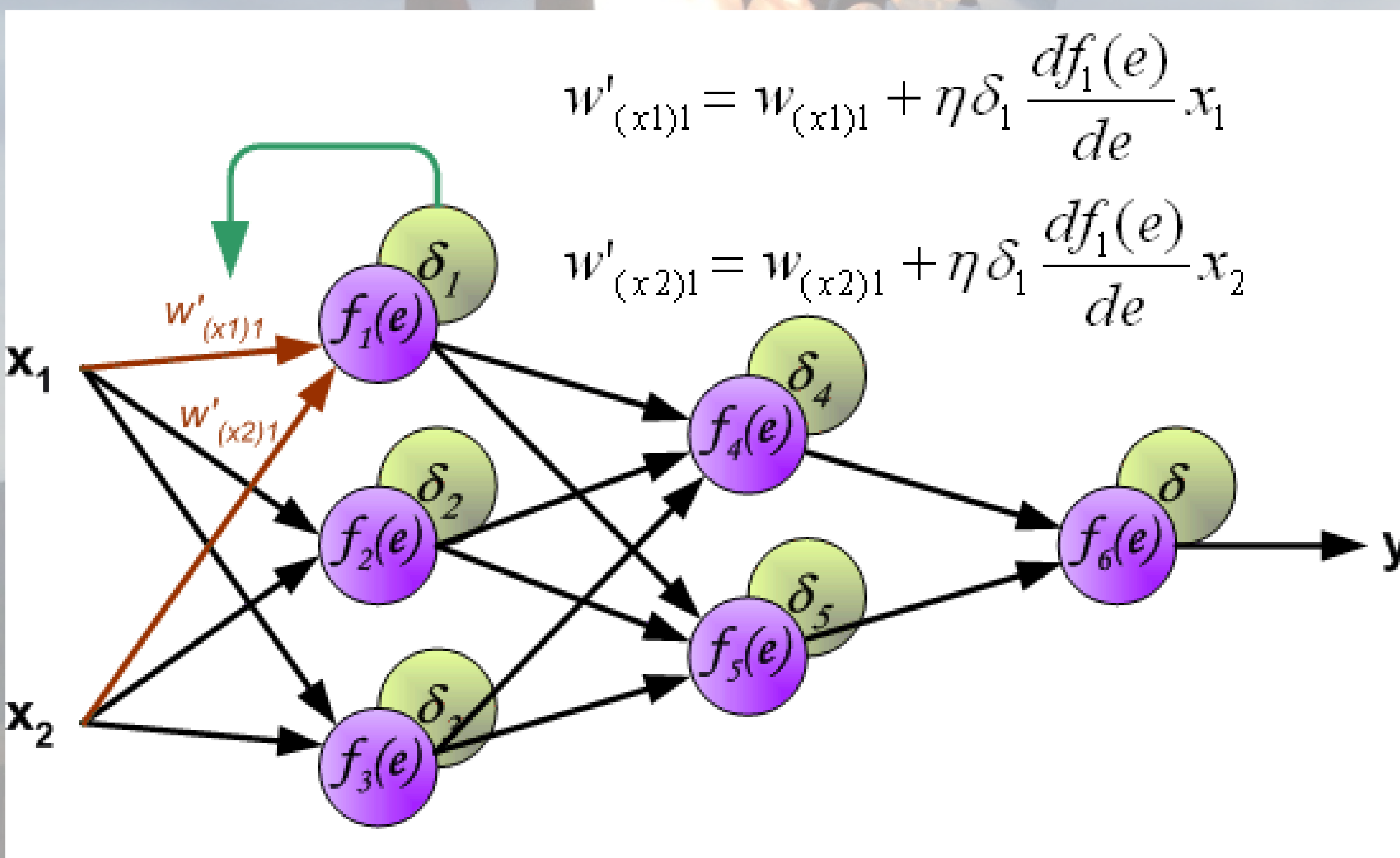
- Error back-propagation algorithm





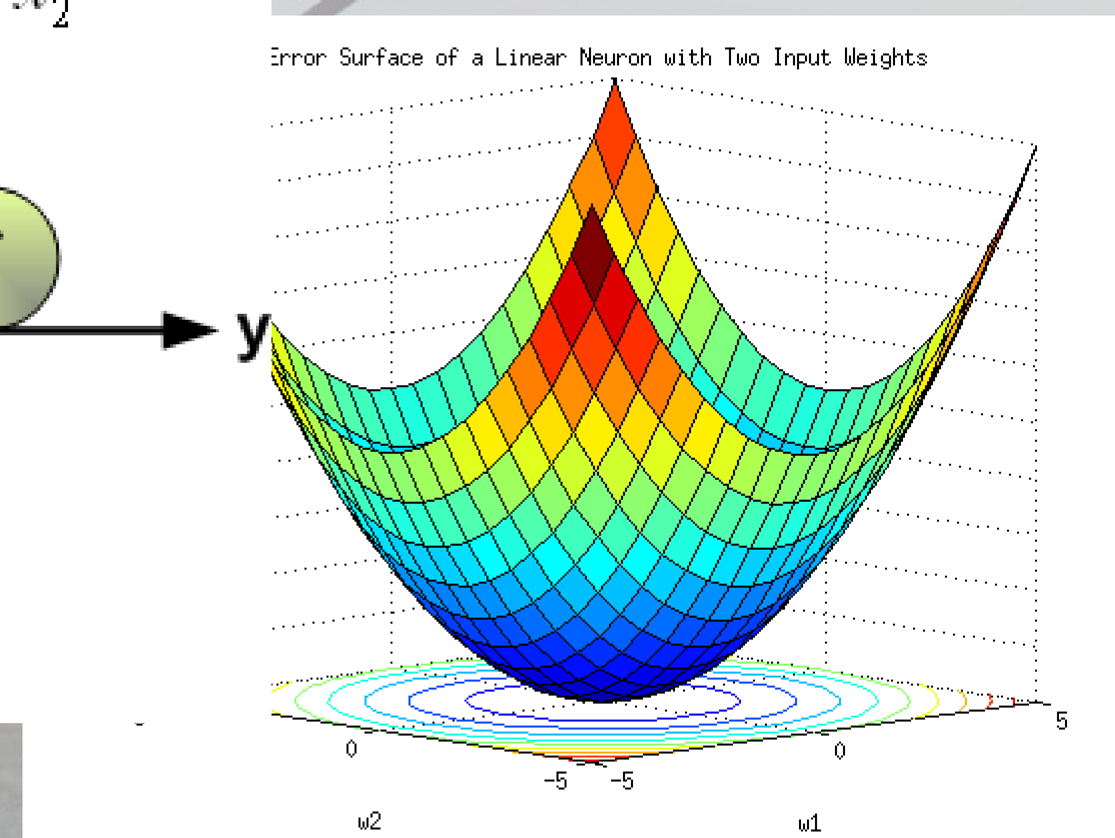
Deep Learning. Error back-propagation

- Correction of weights
- This is a gradient descent methods for NN



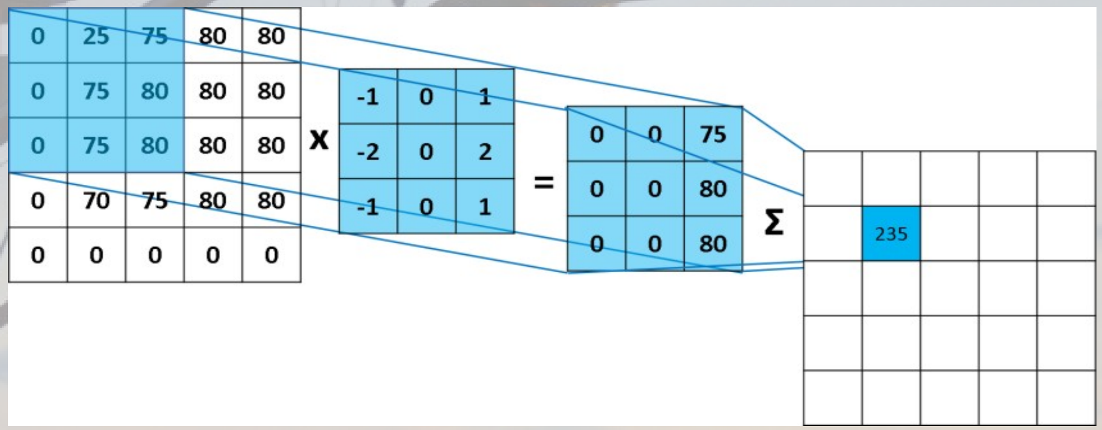
$$w'_{(x1)1} = w_{(x1)1} + \eta \delta_1 \frac{df_1(e)}{de} x_1$$

$$w'_{(x2)1} = w_{(x2)1} + \eta \delta_1 \frac{df_1(e)}{de} x_2$$



Deep Learning. Convolution Neural Networks

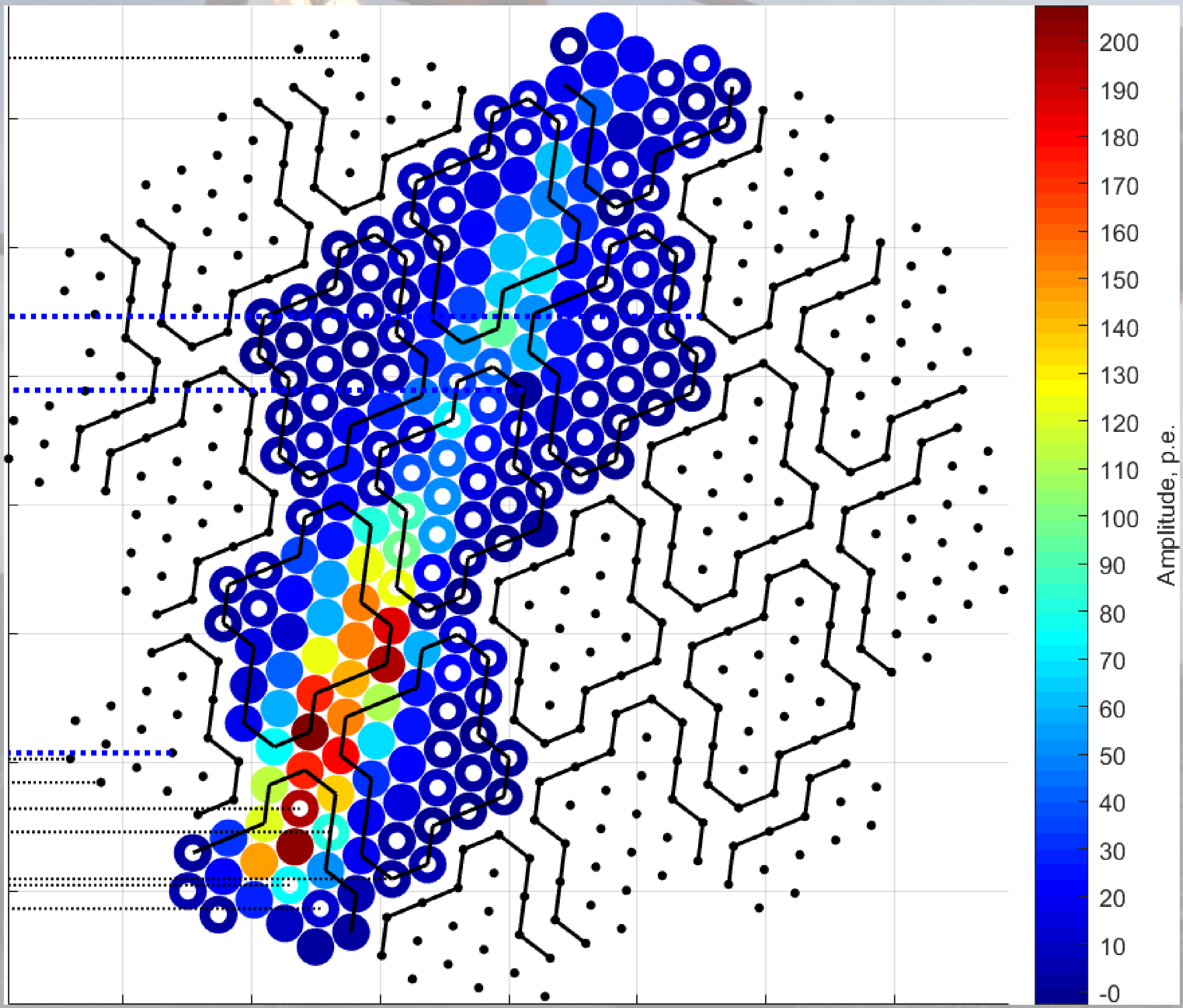
- Convolution layers apply a convolution operation (cross-correlation, or simply filtering) to the input, passing the result to the next layer, and so on.
- Special features of feedback avoid overfitting that was the problem for conventional ANN.



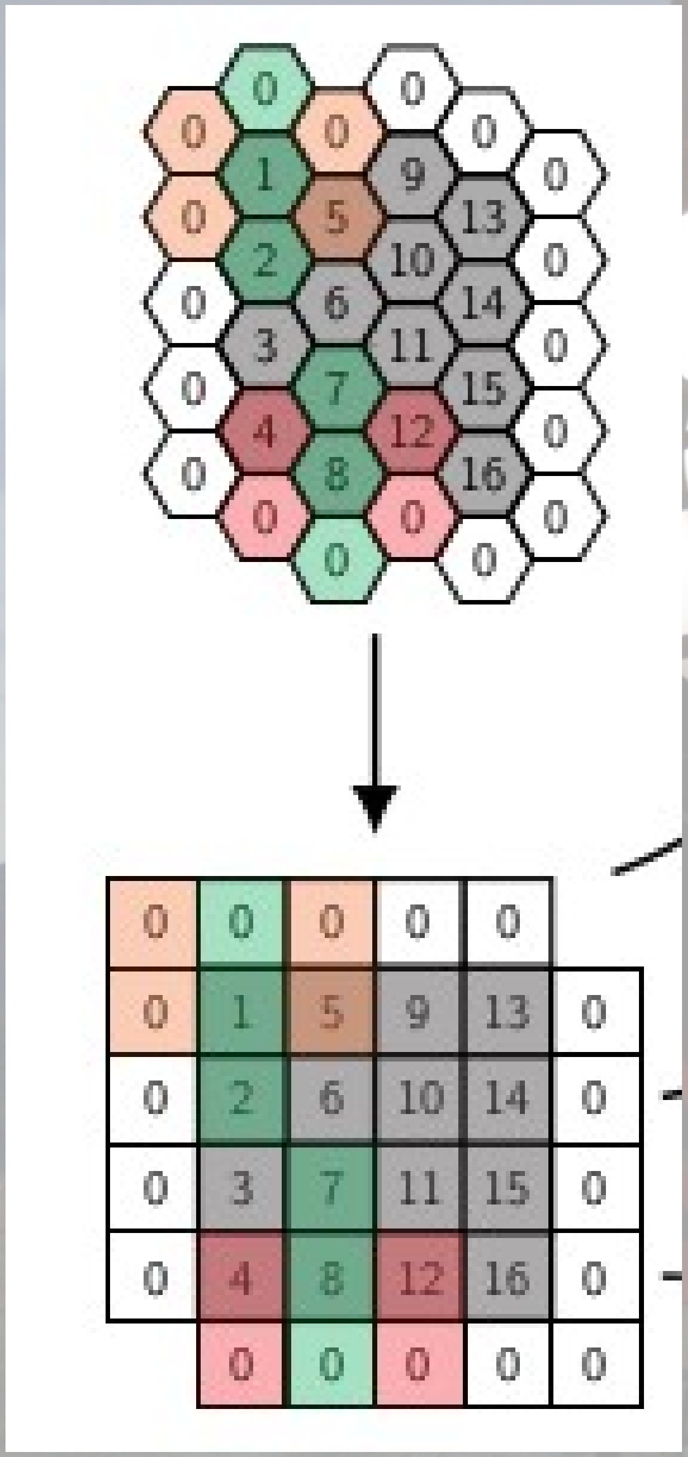
How CNN is realized

Free libraries: TensorFlow PYTORCH

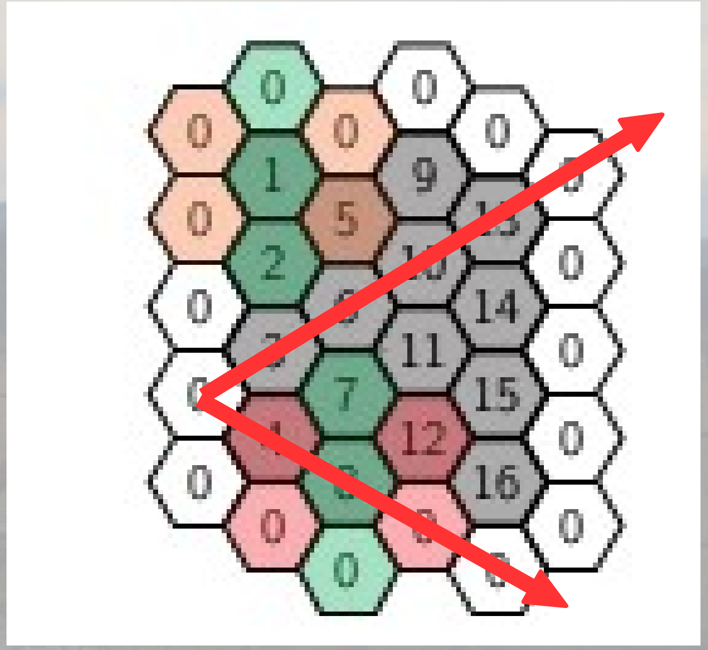
TAIGA telescope image example



Hexagonal to square grid tranformation



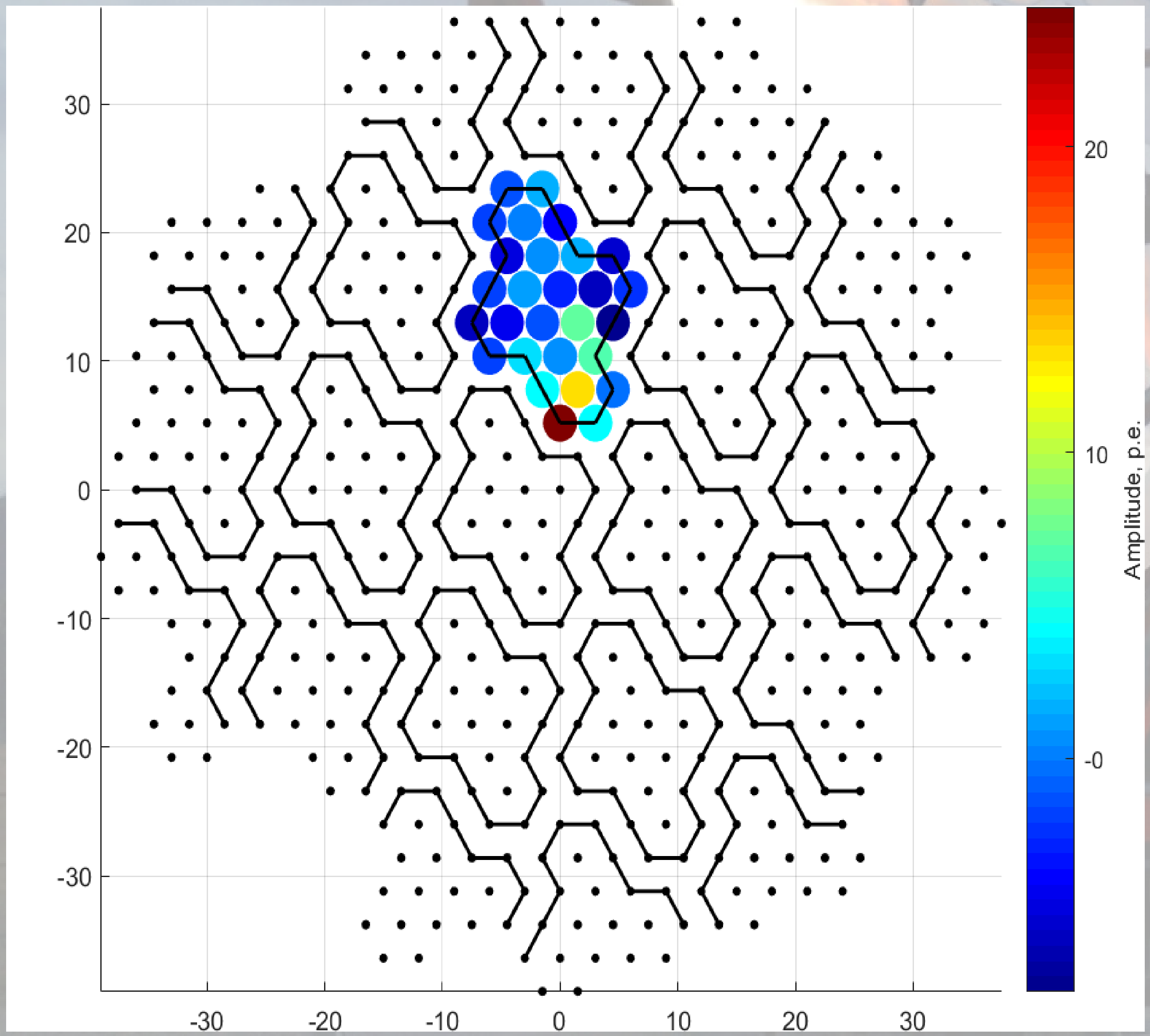
- There are many ways to map a hexagonal grid on a square one.
- We used an inclined coordinate system for preliminary researches.

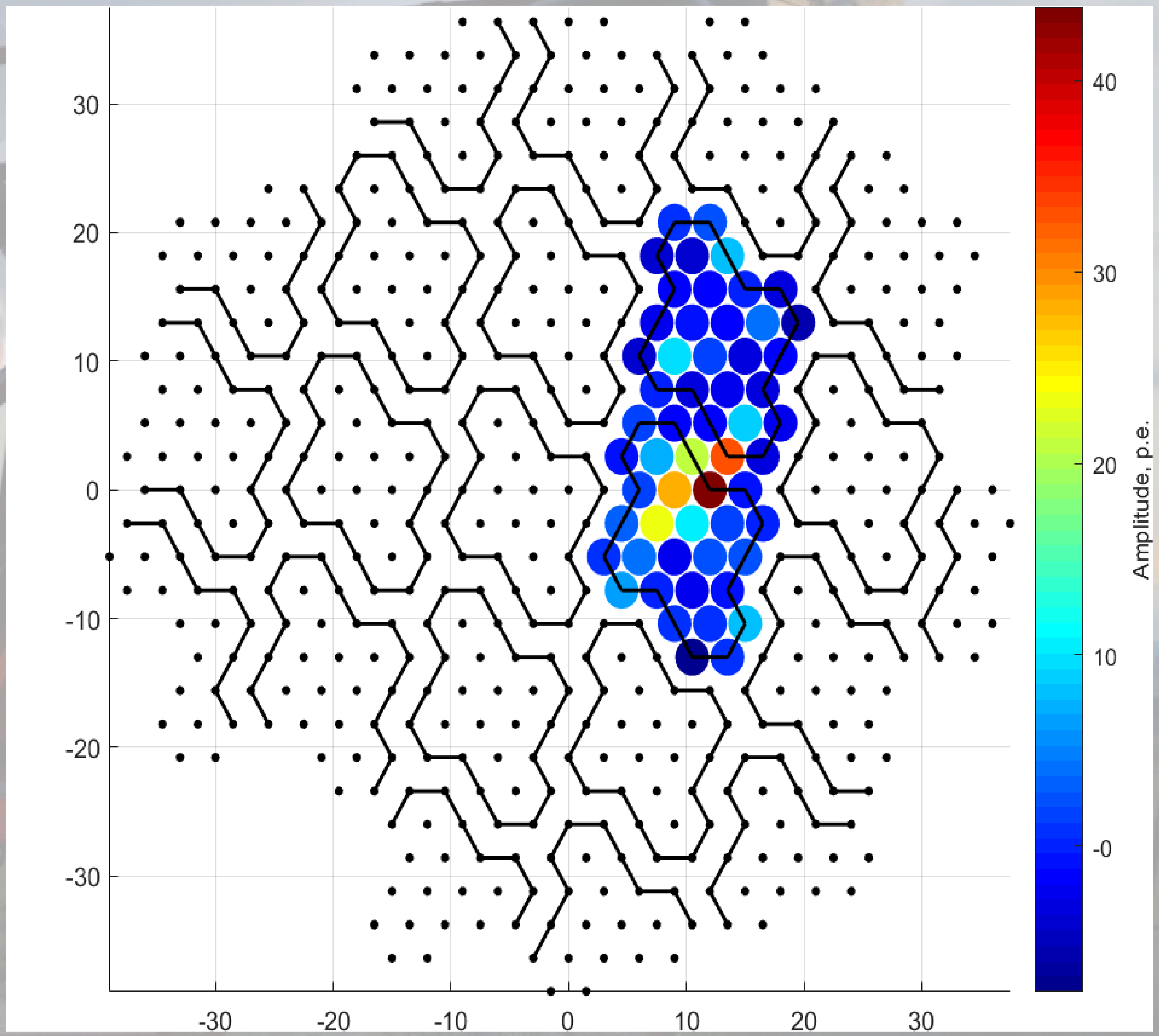


Monte Carlo and blind analysis

- Training datasets: gamma-ray and proton images (Monte Carlo of TAIGA-IACT, real energy spectrum); night sky background, trigger procedure and detector response added, but neither cleaning nor preselection applied.
  Test datasets: after CNN training, datasets (different from training ones) of gamma-ray and proton images in random proportion (blind analysis) were classified by each of the packages: TensorFlow and PyTorch.
  Each package output was ‘probability’ of any image to be gamma-ray of proton.

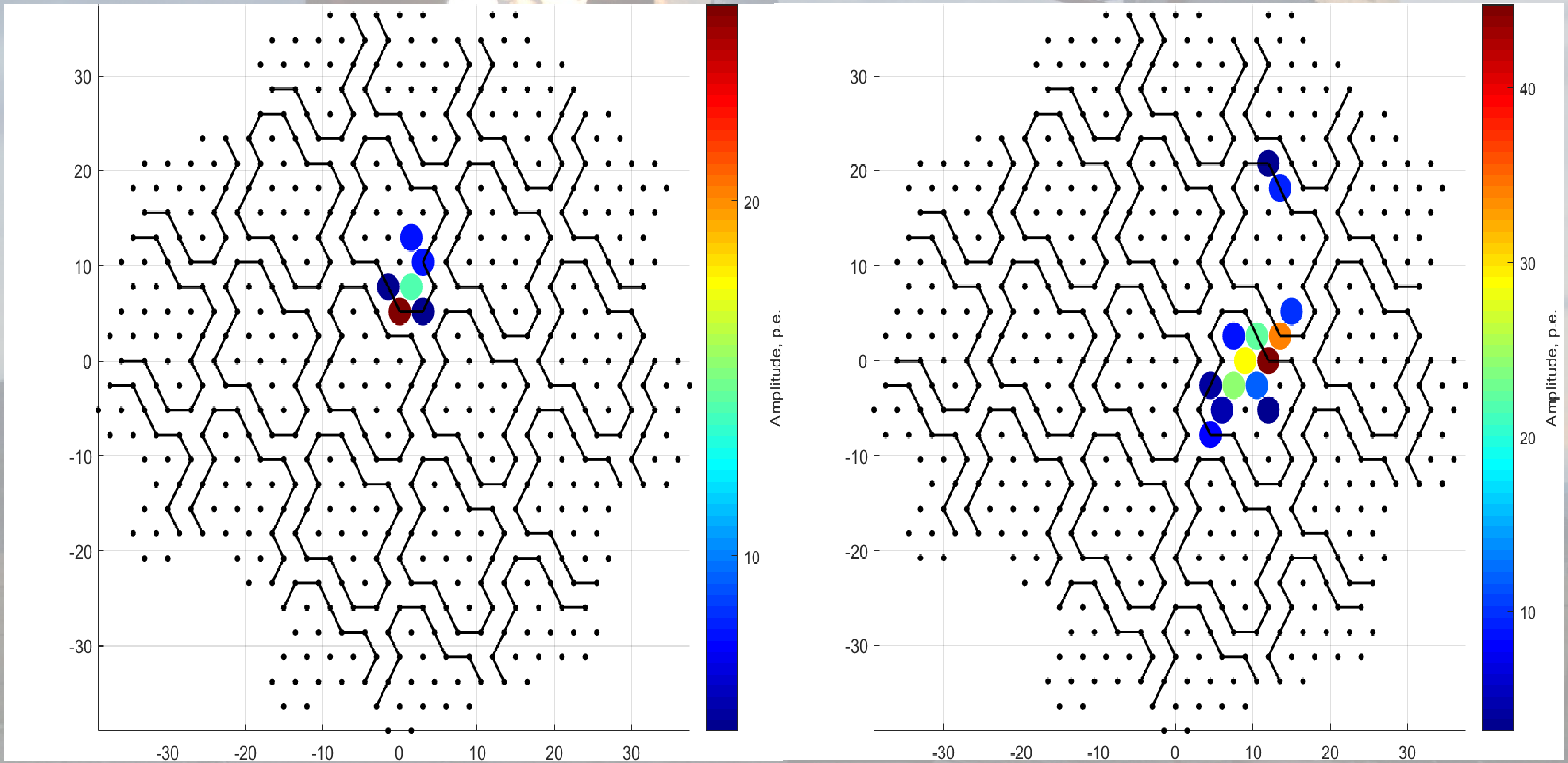
Simulated gamma-ray image example: ‘as is’, no cleaning





astroparticle.online

Simulated gamma-ray image example: after soft cleaning



Particle identification quality

Quality factor

$$Q = \frac{\text{Significance of a } \gamma\text{-source after } \gamma \text{ separation}}{\text{Significance before separation}}$$

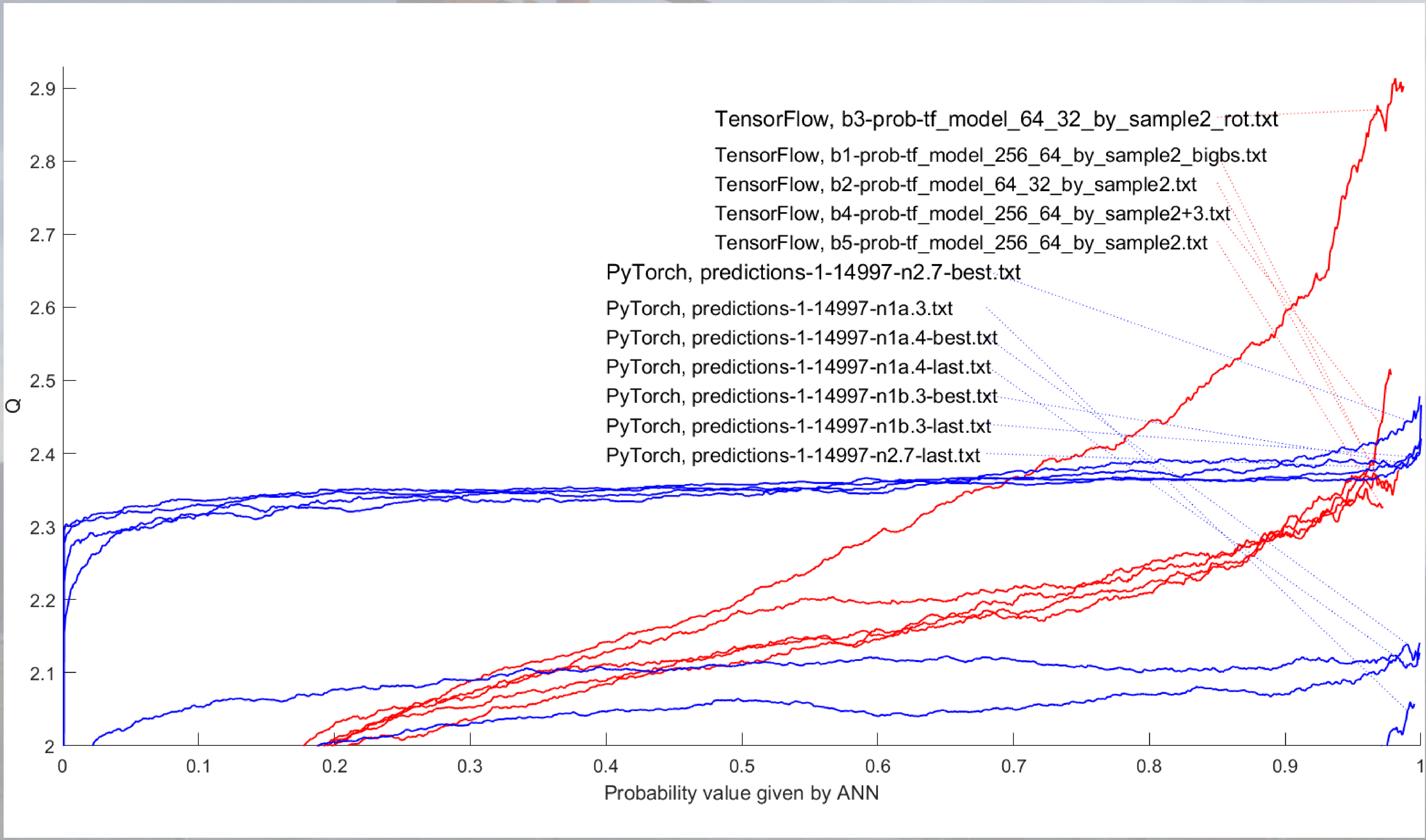
For Poisson distribution of hadron fluctuations:

$$Q = \frac{N_{\gamma \to \gamma} / N_{\gamma}}{\sqrt{N_{hadron \to \gamma} / N_{hadron}}}$$

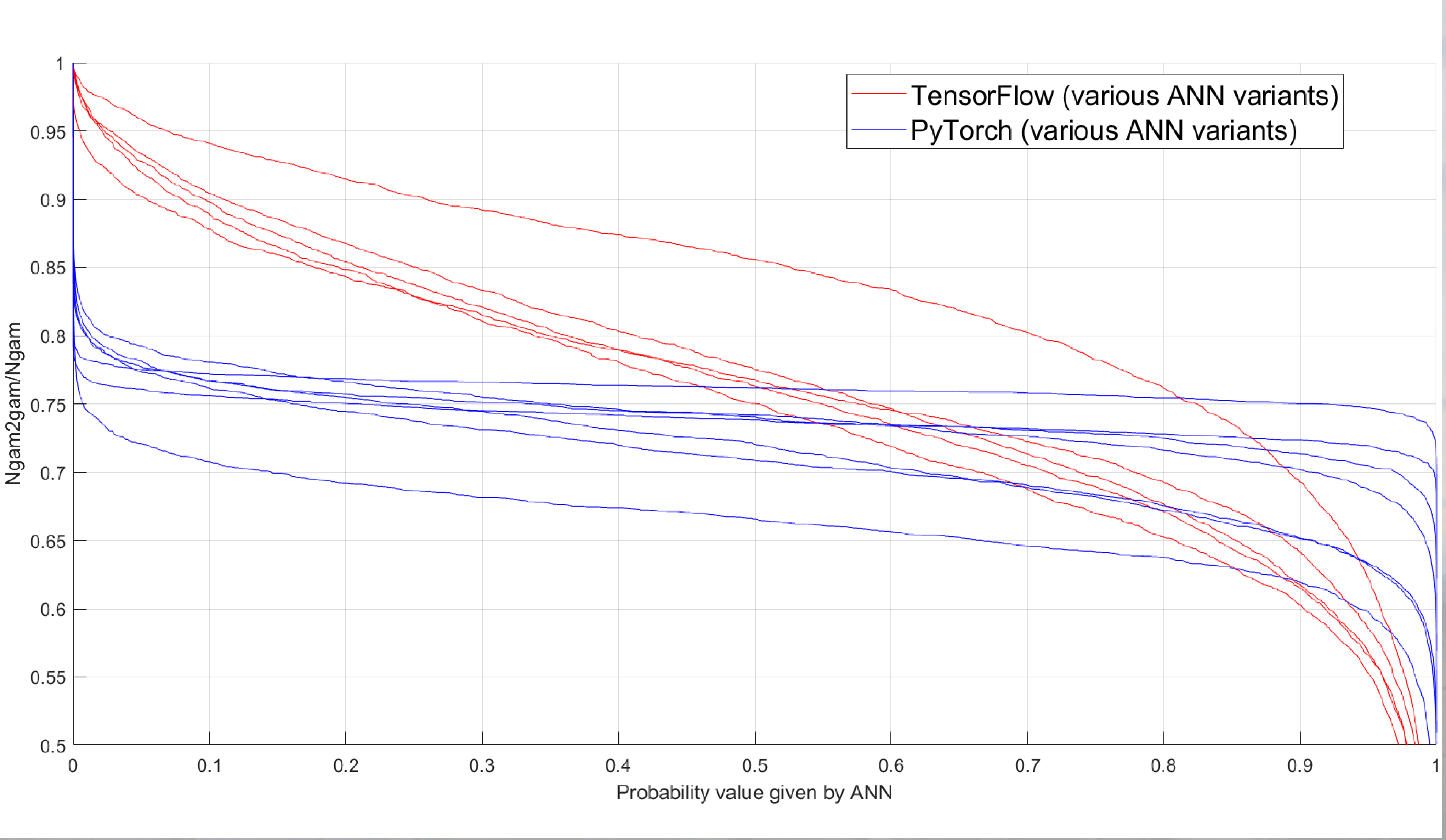
Particle identification quality

	Simple 2-D technique	PyTorch	TensorFlow
Without cleaning	1.76	1.74	1.48
With cleaning	1.70	2.55	2.99

Q vs CNN output parameter (*various CNN after same soft cleaning*)



Number of correctly identified γ-rays vs CNN output parameter (*Problem of the 'cut value' choice*)



astroparticle.online

Conclusions

- **The distributed storage provide unified access to astroparticle data of many collaborations which permit to make multi-messenger analysis.**
- **Modern deep learning analysis techniques permit to get more high quality of the analysis in particular for**
  - **Particle classification**
  - **Parameters of the showers and so, the properties of primary particles.**


astroparticle.online
THANK YOU!
QUESTIONS?