



ESCAPE

European Science Cluster of Astronomy &
Particle physics ESFRI research Infrastructures

GammaLearn

Deep Learning for CTA event reconstruction

T. Vuillaume, M. Jacquemont

E-OSSR Onboarding Presentation

16/10/2020



Introduction/Instructions

Aim:

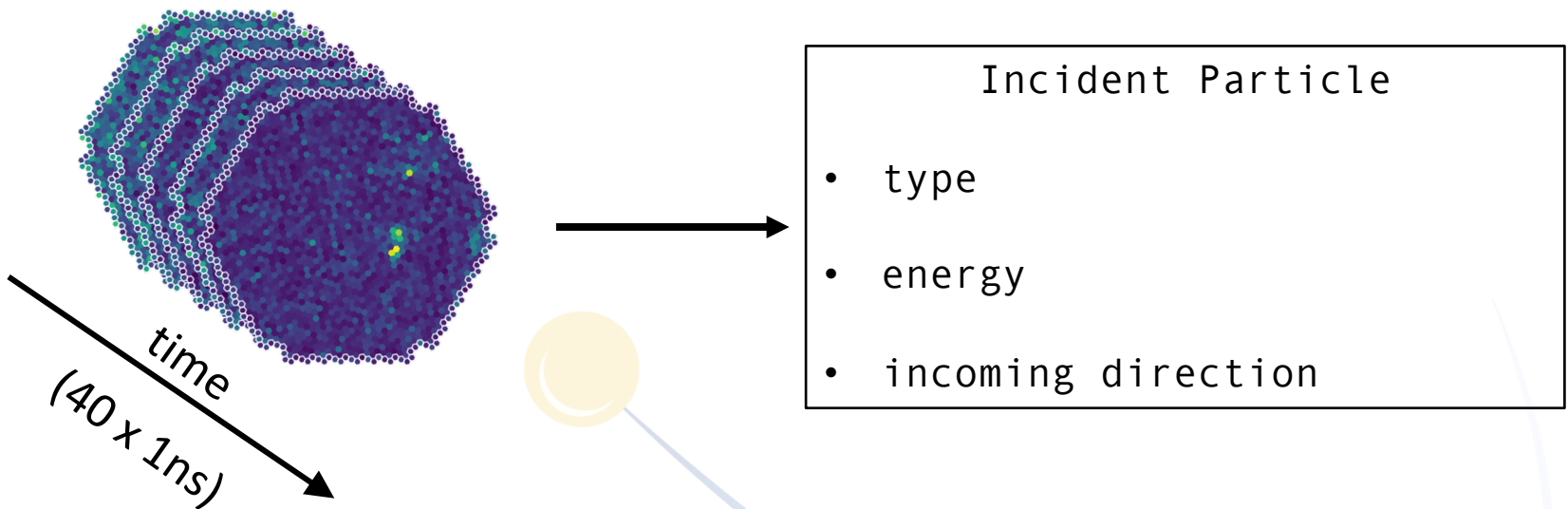
tech reports (~30 min talk and 2-3 page summary documents) on community software for OSSR

- Content:

- science case and "user story"
- added value of OSSR
- update on questions from [OSSR's first questionnaire](#) and [software registration survey](#)
 - Both replies will be provided before the talk by FG1 lead
- Discussion on on-boarding: open points, requirements...

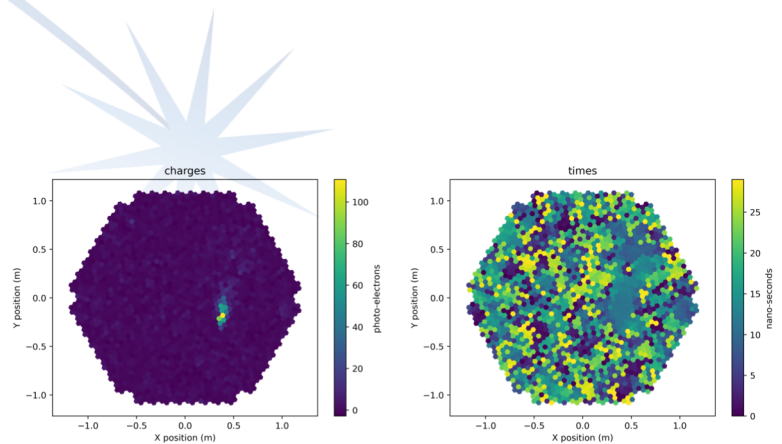
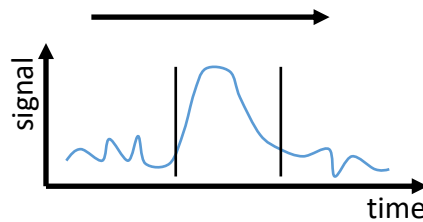
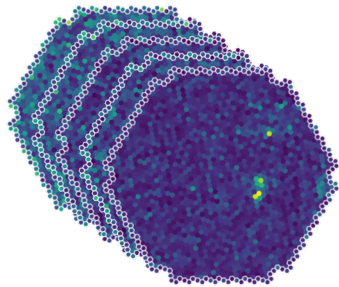


- CTA
 - event reconstruction
- GammaLearn
 - Apply deep learning to CTA event reconstruction

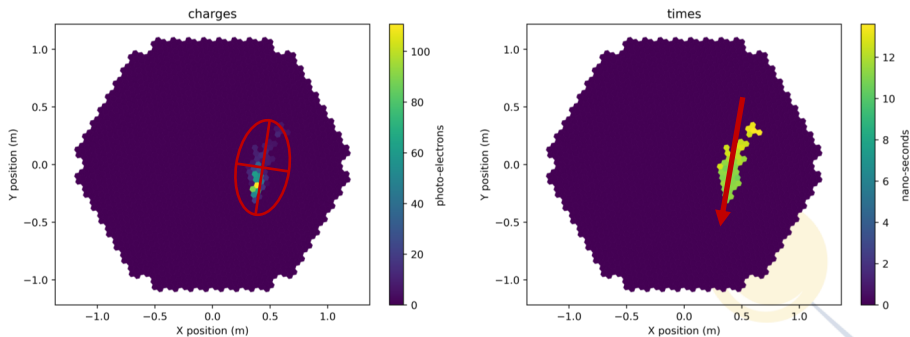


Standard approach

1. Calibration, Integration

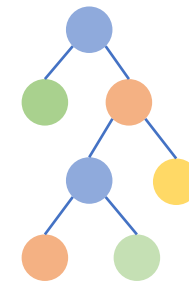


2. Noise removal, features extraction



3. Monte-Carlo + Random Forest → Physics parameters

Image features

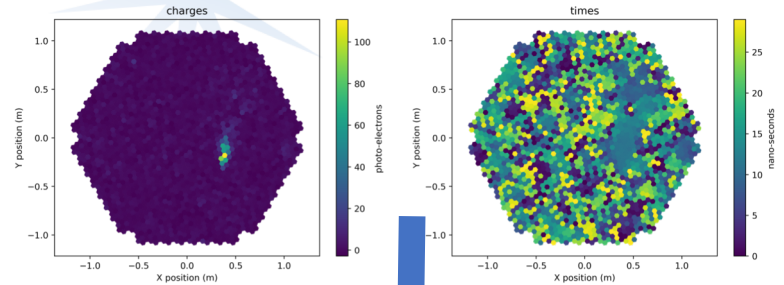
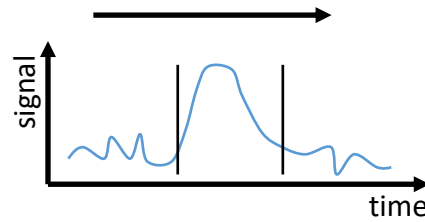
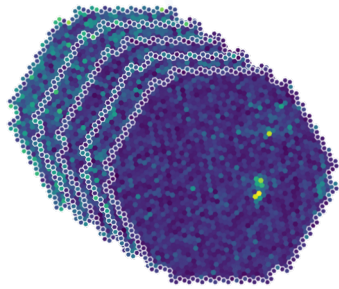


$E = 0,189 \text{ TeV}$

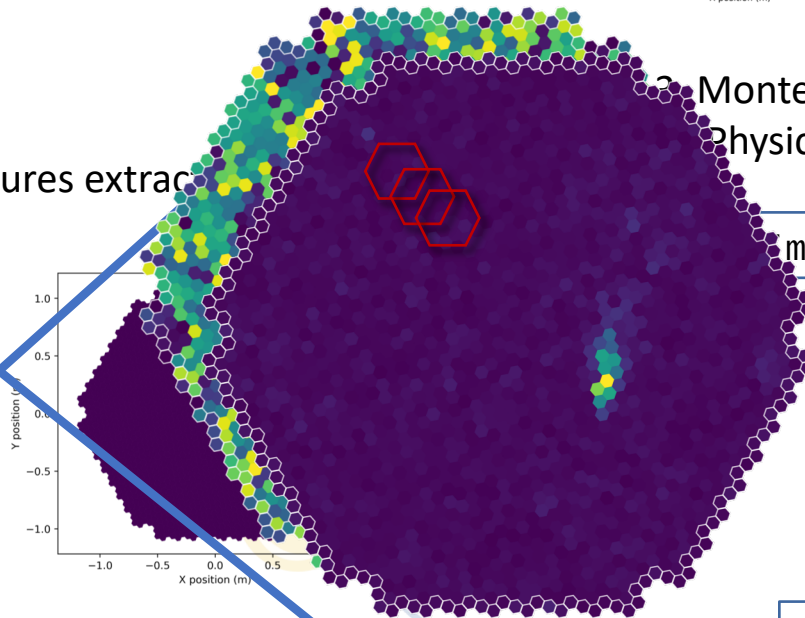
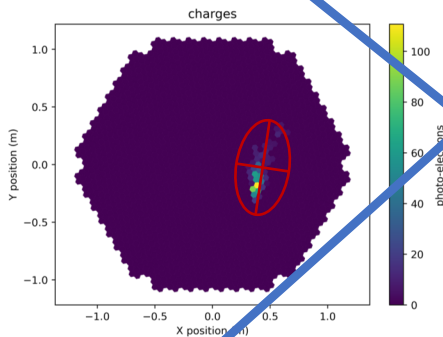


Deep Learning

1. Calibration, Integration



2. Noise removal, features extraction



3. Monte-Carlo + Random Forest Physics parameters

image features

$E = 0,189 \text{ TeV}$



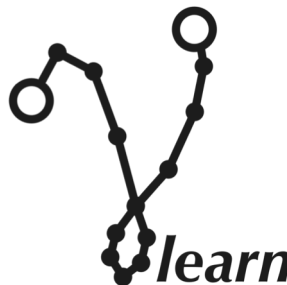
GammaLearn



Mikaël Jacquemont
thesis



- Patrick Lambert
- Alexandre Benoit



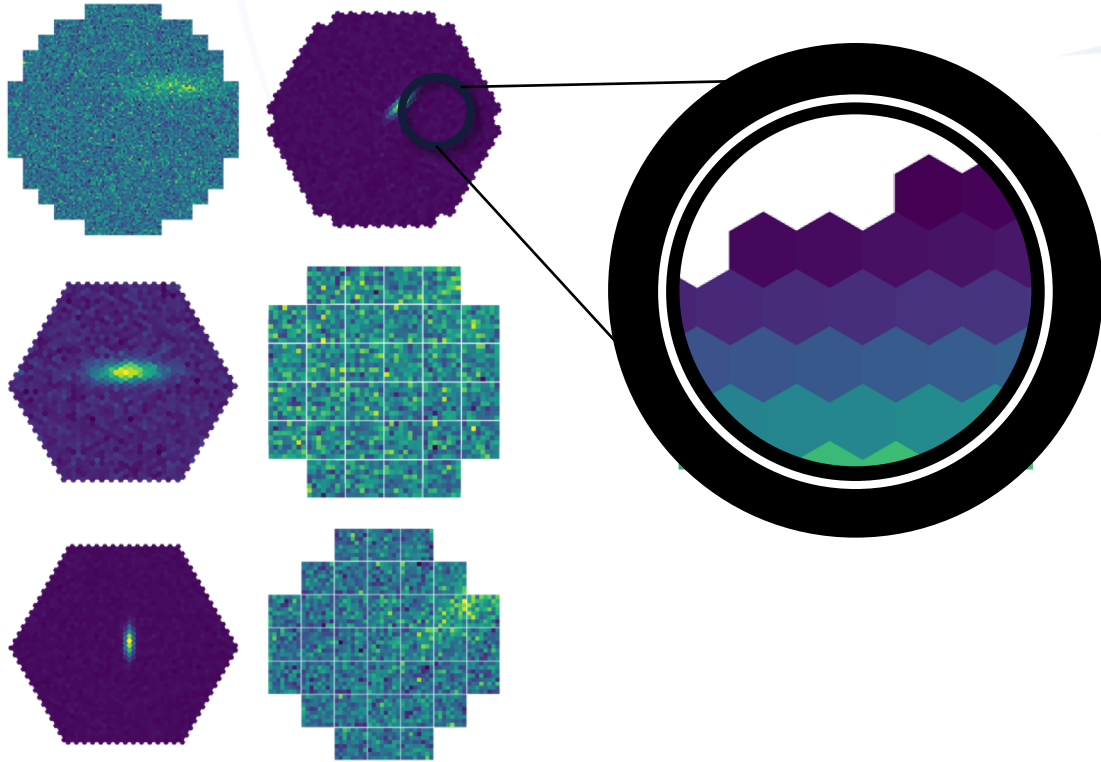
- Thomas Vuillaume (PI)
- Gilles Maurin



IndexedConv



IndexedConv



- Not standard images
- Traditional frameworks and methods inadapted
- Development of a generic solution to apply **Indexed Convolution and Pooling**

- *IndexedConv* library for PyTorch
- Publication: M. Jaquemont et al, VISAPP 2019



IndexedConv

Developments on GitHub:

<https://github.com/IndexedConv>

- Pull requests, release, versioning...
- CI: Travis and unit tests
- Documentation: <https://indexed-convolution.readthedocs.io/en/latest/>
- MIT License



Software/Service Requirements

- Requirements: Python 3, PyTorch
- Hardware requirements
 - GPU is not mandatory but a strong suggestion
- Containerisation
 - Never built a container for IndexedConv



Integration

- Fully integrated in GammaLearn framework (see after)
- Can be use as a standalone lib with PyTorch



OSSR Integration

- What is available?
 - Source code
 - pip package
- What will be onboarded (source code, container, test workflow incl. data)?
 - Source code
 - Container/Image is possible
 - A working test and the dataset used for the publication are included
- Already on OSSR : DOI 10.5281/zenodo.3734091



- What is the “user story” of a EOSC user taking on the software/service?

From the IACT community

- Use GammaLearn framework
- IndexedConv already included and related functions for CTA images

To solve another problem

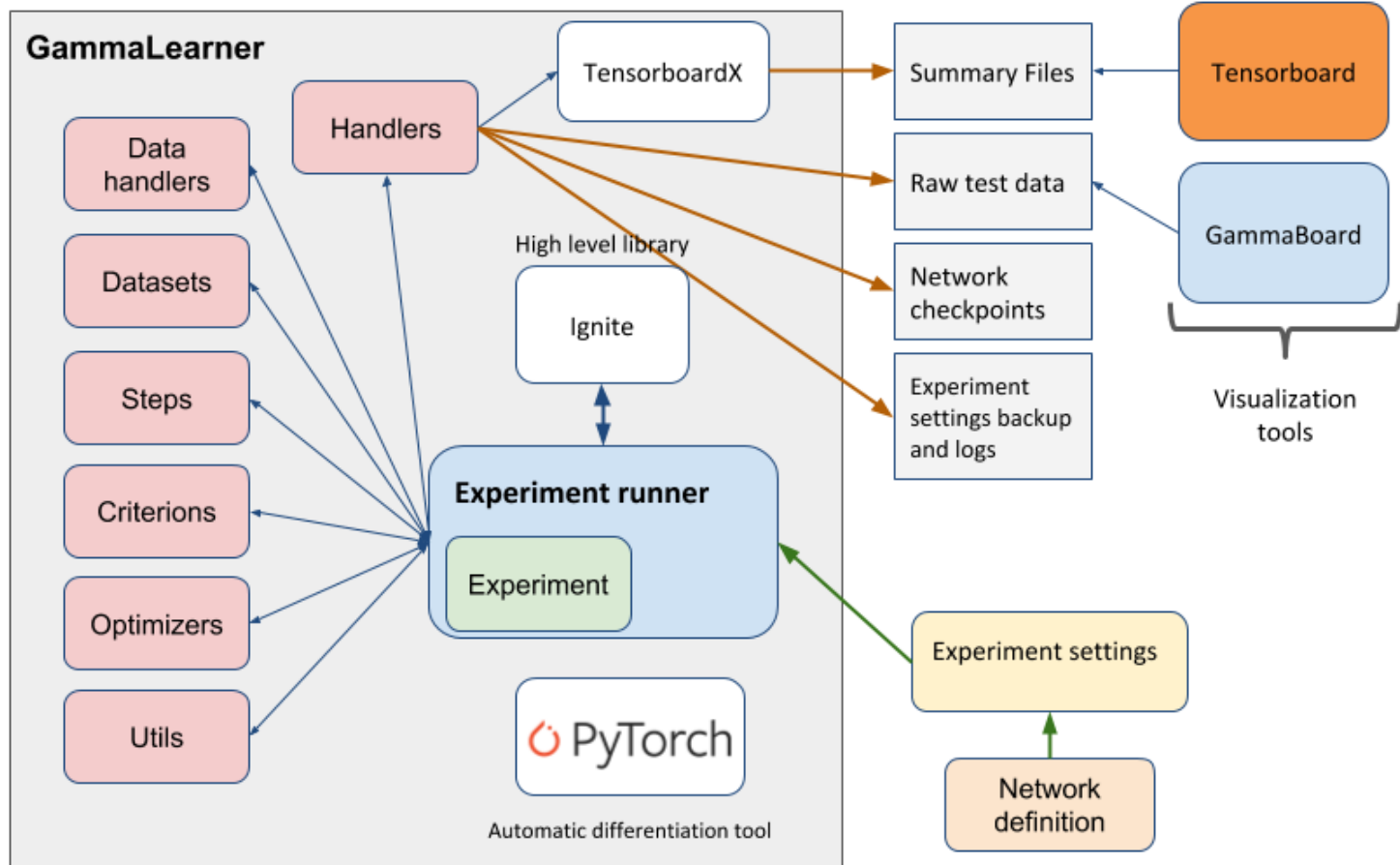
- pip install
- build it's custom image index matrix
- prototype with PyTorch



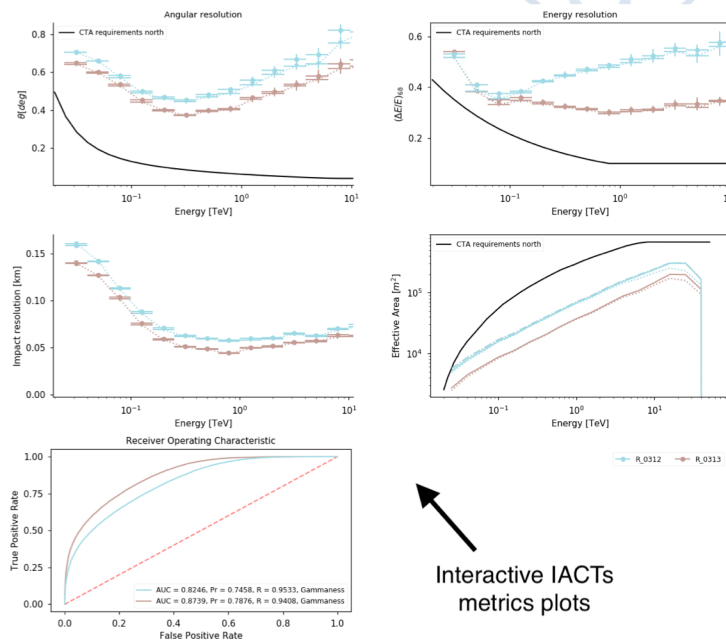
GammaLearn framework



GammaLearn framework



GammaBoard



Interactive IACTs metrics plots

Interactive selection of the experiment to display

gb.exp_box

Experiment	Selected
R_0313	Selected
R_0312	
R_0310	
R_0309	
R_0308	
R_0307	
R_0306	
R_0305_test_dif	
R_0305	
R_0304_checkpoint00_dif	
R_0304	
R_0303	
R_0302	
R_0301	
R_0300_test_dif	
R_0300	
R_0299	

```

exp_name : R_0312
files_folders : /...
dataset_class : GLEAMDataset
dataset_parameters :
  camera_type : GLEAM
  group_by :
  use_time :
  particle :
  num_epochs :
  batch_size :
filters :
  telescope_id_filter :
    tel_id :
  cleaning_filter :
    apply_cleaning :
    picture_thresh :
    boundary_thresh :
    keep_isolated_pixels :
    min_number_picture_neighbors :
network :
  GLEAMMonNTShallowEnergy
  n_features :
  init :
  num_channels :
  batchnorm :
  drop_rate :
  bias :
  num_class :
  regressor :
  energy :
  impact :
  direction :
  
```

Selected experiments information



GammaLearn

- Development on GitLab :
<https://gitlab.lapp.in2p3.fr/GammaLearn>
- Merge requests, releases, versioning...
- MIT License
- Documentation: docstring and examples



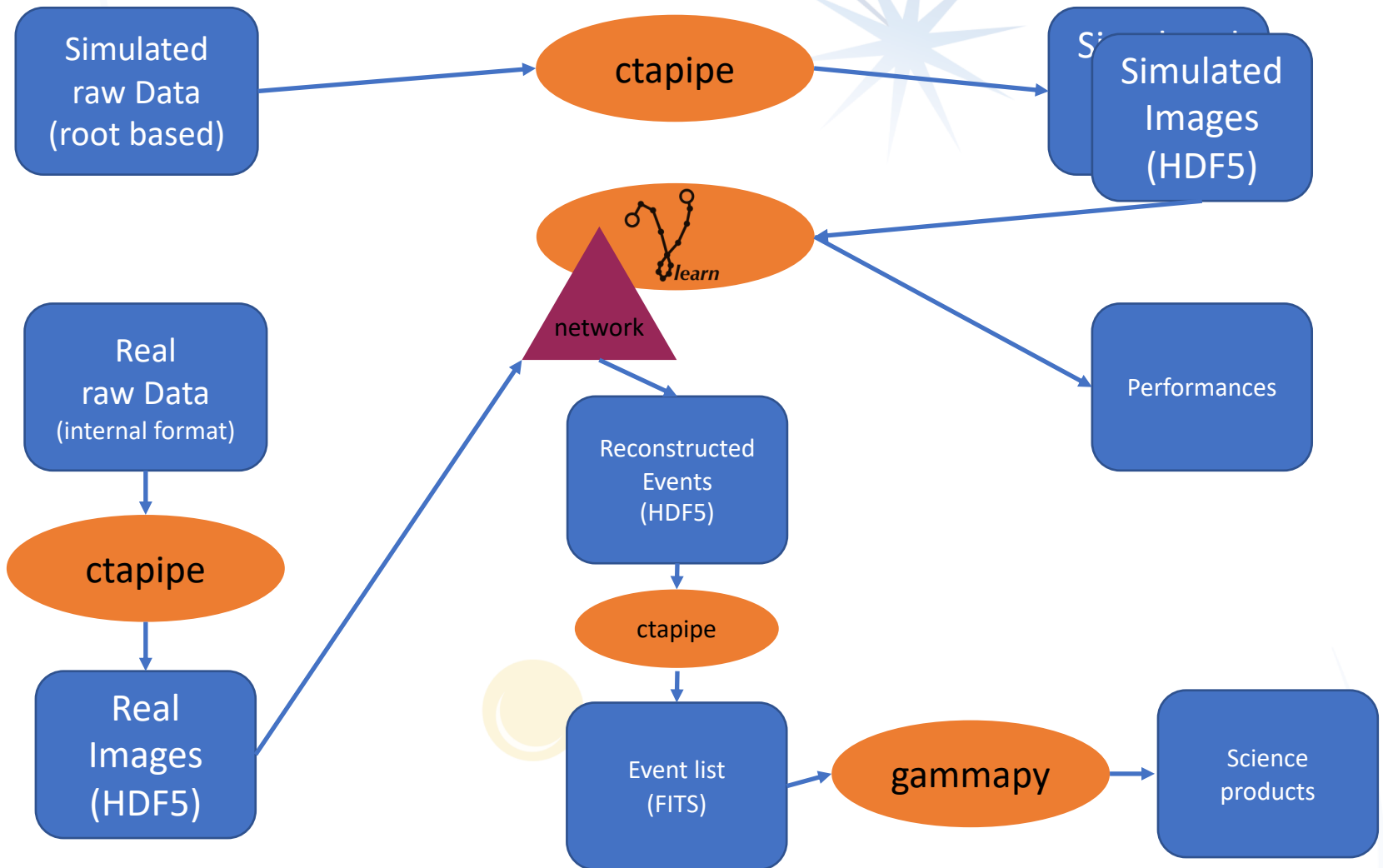
Software/Service Requirements

- Python 3, conda
- Hardware requirements
 - GPU is not mandatory but a strong suggestion
 - GammaLearn has been deployed and used on laptops and computing clusters (MUST, CC-IN2P3, Jean Zay)
- Containerisation and portability requirements
 - Tried singularity images but came back to package install which is easier when developing rapidly
 - Available as conda package:

```
conda install gammalearn -c  
http://conda.anaconda.org/gammalearn
```



Integration into CTA pipeline



- What is available?
 - Source code
 - Conda package
- What will be onboarded (source code, container, test workflow incl. data)?
 - Source code
 - Conda package ?
 - Container/Image is possible
 - Working example included
 - Larger test and workflow will depend on the availability of test data from CTA



- What is the “user story” of a EOSC user taking on the software/service?

Human who wants to develop a new NN for IACT

- Source code
- Install on laptop/small server
- Rapid prototyping and testing
- Scientific interest → publication

Organisation (CTA, other IACT?) who wants to easily use DL in production, especially for the training step

- Container or released version as conda/pip package
- Install in production environment (GRID ?)
- Modify only the config files
- Integrates in a larger workflow



Time for a short demo (~10 min)

- Show how the software is used and what is the outcome
- What should and can a EOSC user do with the software?



Questions and discussion

Links

- <https://github.com/IndexedConv>
- <https://gitlab.lapp.in2p3.fr/GammaLearn>
- <https://github.com/cta-observatory/ctaplot>
(gammaboard)



TOC of Tech Report

- Introduction
 - ESFRI/RI and Partner, Science Case
 - Software and Service Name
- Software/Service Development Strategy
- Software/Service Requirements
- OSSR Integration
 - Status
 - Content
 - User Story

