



cherenkov
telescope
array



Gammappy update

Gammappy is a prototype for the CTA science tools

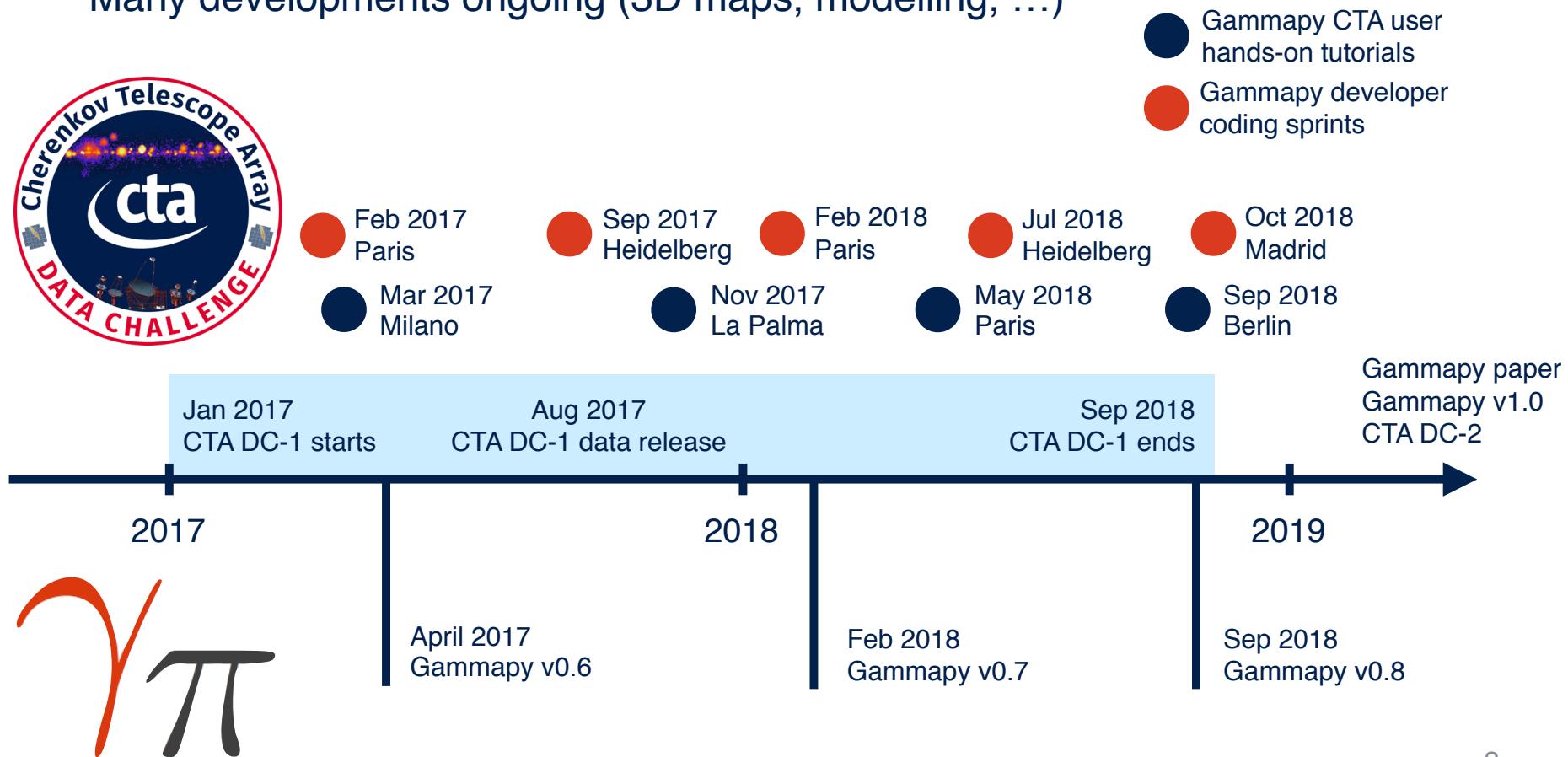
Sep 26, 2018, Berlin
CTA consortium meeting

Christoph Deil for the Gammappy team

Gammappy activities



Gammappy is very active now in CTA
DC-1 was a major effort (tools comparison, user support)
Many developments ongoing (3D maps, modelling, ...)



Gammappy team



Development team



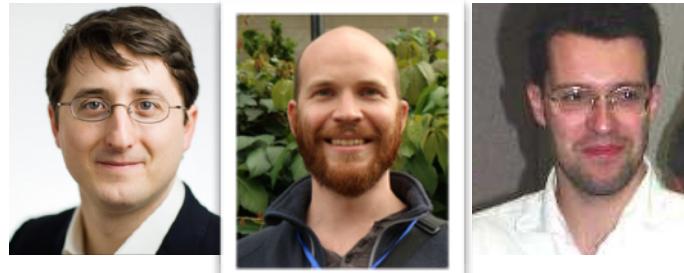
Project managers:

- Roberta Zanin (MPIK)
- Bruno Khelifi (APC)



Lead developers:

- Christoph Deil (MPIK)
- Axel Donath (MPIK)**
- Régis Terrier (APC)



Coordination committee:

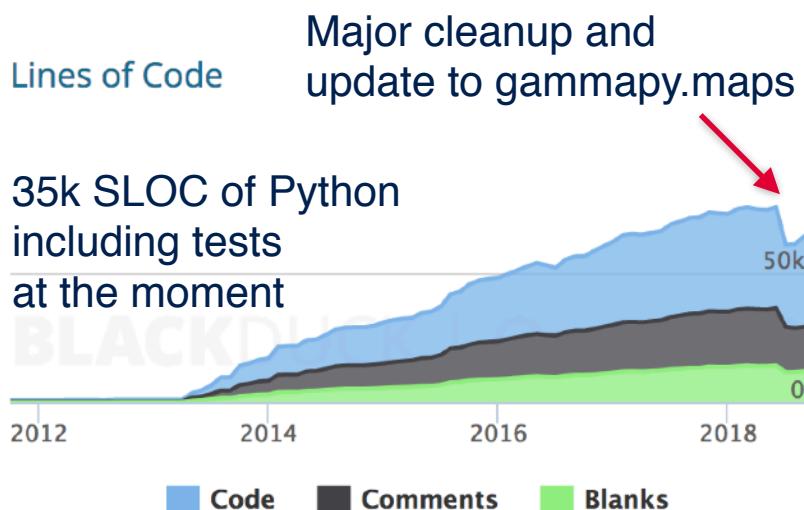
- Catherine Boisson (LUTH)
- Emma de Ona Wilhelmi (DESY)
- Fabio Acero (DAP/CEA)
- Jose Luis Contreras (UCM)
- + Project managers and lead developers
- + one representative for each group
that joins the development team

see <http://gammappy.org/team.html>

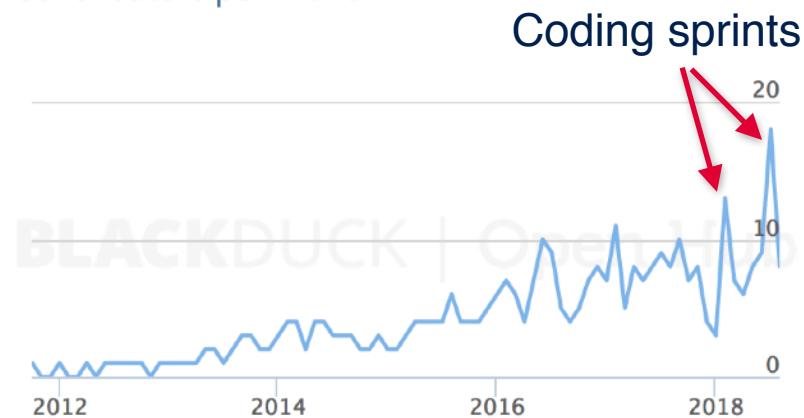
Gammappy developments



Gammappy development activity very high now
Contributor and user base is growing



Contributors per Month



Next week (Oct 1-5) next Gammappy coding sprint at UCM Madrid ([LINK](#)). Contact: Jose-Luis Contreras



UNIVERSIDAD
COMPLUTENSE
MADRID



Gammapy code and test quality



- Major effort to improve code and test quality in Gammapy ongoing
- Refactor, clean up, delete old code
- Improving automated tests
 - Use simple datasets for unit tests
 - Use curated datasets (CTA, HESS, Fermi-LAT) for science verification
- New Gammapy code style: Black
“the uncompromising Python code formatter”
- Several other standard tools mentioned in previous presentations:
pull request code review, test coverage, static analysis, continuous integration, ...



“Any color you like.”



Gammapy dependencies

- Gammipy now moving from the prototyping towards the production-quality phase and simpler maintenance.
- Only core dependencies are Numpy and Astropy, and PyYAML if we continue using YAML
- astropy-regions advanced, will become astropy.regions in 2019
- healpy will be replaced with astropy.healpix in 2019
- Sherpa now an optional fitting backend, not a core dependency
- Completely removed dependencies in v0.8:
scikit-image, photutils, pandas, aplpy
Were used just in a few functions in Gammipy,
e.g. 20 lines of code in [`gammipy.detect.find_peaks`](#)
replace the call to the `find_peaks` that we made to photutils.

CTA first data challenge



- CTA DC-1 was a major focus
- Tool comparison now with 3D analysis results from Gammapy
- Write-up of our experiences and feedback in the close-out document at the moment
- Next slides give summary of user feedback from the questionnaire



*Thank you Roberta & Fabio and everyone
for improving Gammapy via DC-1!
And thank you Luigi & Stefan for your
patience with the Gammapy 3D results
coming late in the DC-1 tools check.*

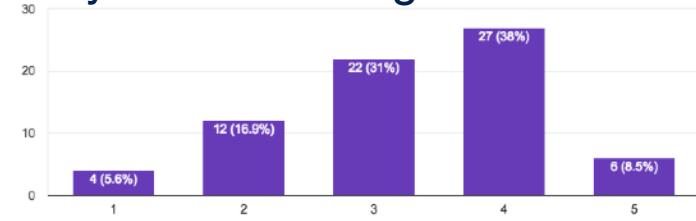
CTA DC-1 user feedback



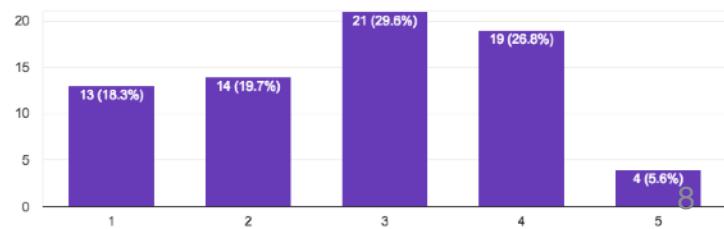
- 50 responses for Gammapy in the questionnaire from August 2018.
- 80% use Python scripts, 80% use Jupyter notebooks (i.e. most use both)
- 92% install Gammapy in less than one hour, 58% in less than 10 min
- 40% are still using Python 2.7, 60% are using Python 3.
- Wide spectrum of user familiarity with Python and Jupyter
- Variety of user support channels used (hands-on tutorials, online docs, Gammapy Slack, mailing list, Github)



Python knowledge



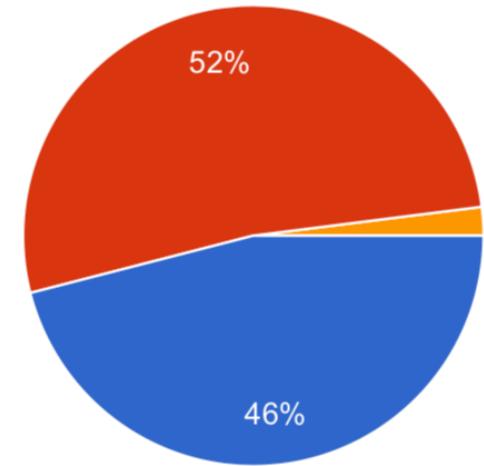
Jupyter knowledge



Platform support



- Same number of Gammapy users on Linux and MacOS, those are fully supported.
- Only a few Windows users so far.
- Why? Either little interest overall or missing full support in Gammapy so far.
- Is Windows support for CTA ST planned?



- Linux
- Mac OS X
- Windows

CTA 1DC user feedback summary



What did you like ?

- Easiness to understand each analysis step.
- Transparency of the code and understanding how it is made (not a black-box).
- Flexibility of implementing new features.
- Accessing intermediate results in interactive notebooks.
- Large community, openly developed.
- Interfaces nicely with stable packages widely used in astronomy (numpy, astropy).
- Use of astropy.units.

What did you not like/problems encountered ?

- Inconsistency in the API and deprecated methods.
- Organization in the documentation and lack of it for some tools.
- Young and immature package.
- Slow on large datasets (> 100 runs).
- No 3D analysis for complex regions.
- Issues in the DC1 IRFs.

What would you improve/new functionalities ?

- Documentation, tutorials, and more notebooks once the API are stabilised.
- A solid 3D analysis.
- Run-wise background normalisation.
- Configuration file (YAML) for high-level analysis (traceability).
- Command line interface to have a step by step analysis.
- Pulsar tools.

Thank you for using Gammapy in this early phase of CTA!
Several issues addressed already in Gammapy v0.8.
More improvements coming in the next releases.

Gammapy version 0.8



- Gammappy v0.8 release this week. See [changelog](#).
- Next slides give summary of major new features.

0.8 (2018-09-23)

Summary

- Released on September 23, 2018 ([Gammappy 0.8 on PyPI](#))
- 24 contributors (6 new)
- 7 months of work (from Feb 28, 2018 to Sep 23, 2018)
- 314 pull requests (not all listed below)

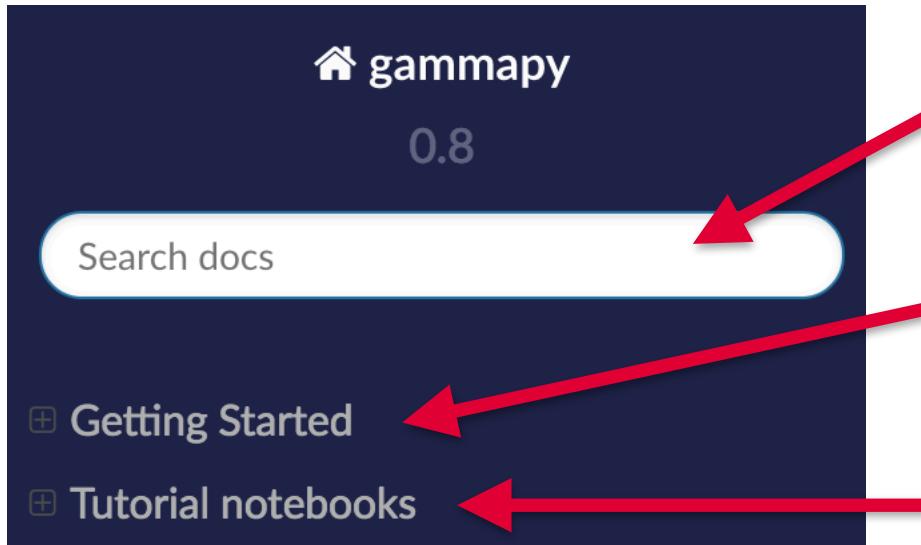
What's new?

Gammappy v0.8 features major updates to maps and modeling, as well as installation and how to get started with tutorial notebooks. It also contains many smaller additions, as well as many fixes and improvements.

Gammappy documentation



- Development version at <http://docs.gammappy.org/dev/>
- Stable version at <http://docs.gammappy.org/stable/>
- Currently: <http://docs.gammappy.org/0.8/>



Search Gammappy docs
(high-level and API)

Step-by-step instructions for
installation, setup and first steps

Tutorials Jupyter notebooks
Integrated in docs via nbsphinx

Installation



```
curl -O http://gammapy.org/download/install/gammapy-0.8-environment.yml  
conda env create -f gammapy-0.8-environment.yml  
conda activate gammapy-0.8
```

- [Getting started](#) page now is opinionated recommending conda to new users
- Other options still exist of course: pip, Macports, now also packaged for Debian
- For v0.8, we try shipping a conda env with completely pinned dependencies
- Please report issues and have a little patience as we figure out the best way to distribute Gammapy and reproducible execution environments → important experiences with conda for CTA

```
name: gammapy-0.8  
  
channels:  
  - conda-forge  
  - sherpa  
  
dependencies:  
  - gammapy==0.8  
  - python==3.6  
  - ipython==6.5.0  
  - jupyter==1.0.0  
  - jupyterlab==0.34.12  
  - jupyterlab_launcher==0.13.1  
  - cython==0.28.5  
  - numpy==1.15.1  
  - astropy==3.0.4  
  - regions==0.3  
  - click==6.7  
  - pyyaml==3.13  
  - scipy==1.1.0  
  - reproject==0.4  
  - uncertainties==3.0.2  
  - naima==0.8.1  
  -iminuit==1.3.3  
  -sherpa==4.10  
  -healpy==1.11.0  
  -matplotlib==2.2.3  
  -pandas==0.23.4
```

gammapy info



```
$ gammapy info
```

System:

python_executable	:	/Users/deil,
python_version	:	3.6.0
machine	:	x86_64
system	:	Darwin

Gammapy package:

path	:	/Users/deil,
version	:	0.8

Other packages:

numpy	:	1.15.1
scipy	:	1.1.0
matplotlib	:	2.2.3
cython	:	0.28.5
astropy	:	3.0.4

Gammapy installation puts the Python package your site-packages and the “gammapy” executable on your PATH

Type “gammapy info” to get infos about your Gammapy installation

Of course, you can always check from Python, IPython or Jupyter:

```
$ ipython
Python 3.6.0 | packaged by conda-forge
Type 'copyright', 'credits' or 'license'
IPython 6.5.0 -- An enhanced Interactive

In [1]: import gammapy

In [2]: gammapy.__version__
Out[2]: '0.8'

In [3]: gammapy
Out[3]: <module 'gammapy' from '/Users/
```

gammapy download



```
gammapy download tutorials  
cd gammapy-tutorials  
export GAMMAPY_DATA=$PWD/datasets
```

- With v0.8, we introduce “gammapy download tutorials”
 - fetch tutorial notebooks
 - fetch example datasets
- Uses web server gammapy.org, not git or Github any more.
- Resolves long-standing usability and versioning issues with the previous setup using the gammapy-extra git repo

```
[\$ ls -1  
datasets  
gammapy-0.8-environment.yml  
notebooks-0.8  
\$ du -hs datasets/*  
17M     datasets/catalogs  
19M     datasets/cta-1dc  
3.7M    datasets/dark_matter_spectra  
868K    datasets/ebl  
6.4M    datasets/fermi_2fhl  
516K    datasets/fermi_3fhl  
3.0M    datasets/fermi_survey  
43M     datasets/hess-dl3-dr1  
2.2M    datasets/images  
292K    datasets/joint-crab
```

Stable datasets, no version
Versioned notebooks

Gammappy tutorials



- Gammappy tutorials written as Jupyter notebooks, integrated into Sphinx docs using nbsphinx
- “Launch in binder” option to try online using <https://mybinder.org/>
- Many new tutorials:
 - maps intro
 - 3D analysis
 - 3D simulation and fitting
 - light curve
 - background models
 - H.E.S.S.
 - Fermi-LAT
 - dark matter

This screenshot shows a web browser displaying the 'Tutorial notebooks' section of the Gammappy documentation. The URL is docs.gammappy.org/dev/tutorials.html. The page has a dark blue header with the 'gammappy' logo and version '0.9.dev7470'. A search bar is at the top. On the left, a sidebar lists categories: Getting Started, Tutorial notebooks (which is expanded), Notebooks, Extra topics, Basics, Installation, data - Data and observations, irf - Instrument response functions, maps - Sky maps, image - Map image analysis, cube - Map cube analysis, detect - Source detection, and background - Background modeling. The main content area is titled 'Notebooks' and contains sections for 'First steps with Gammappy' and 'Introduction to gammappy.maps'. It also includes links for CTA data challenges and analysis, H.E.S.S. data analysis, and 3D cube analysis.

This screenshot shows a web browser displaying the 'Getting started with Gammappy' notebook. The URL is docs.gammappy.org/dev/notebooks/first_steps.html. The page title is 'Getting started with Gammappy'. At the top, it says 'This is a fixed-text formatted version of a Jupyter notebook.' Below that, there's a note about contributing to the GitHub repository. The main content is the 'Introduction' section of the notebook, which includes text and several code snippets. The sidebar on the left is identical to the one in the previous screenshot, showing the 'Getting started with Gammappy' section is currently selected.

Tutorial notebook maintenance



- Major improvements in tutorial Jupyter notebook maintenance workflow
- Versioning: store in main Gammapy repo, ship via gammapy download
- Authoring and review: store only markdown and code cells, not the output → pull request diff that can be reviewed
- Testing: check that notebooks execute without error in continuous integration

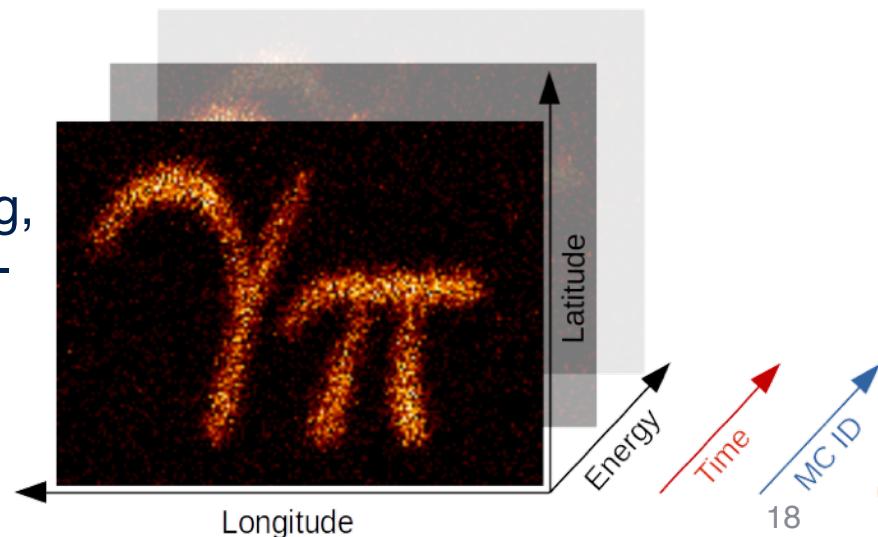
A screenshot of a GitHub pull request page. The repository is "gammapy / gammapy". The branch is "master". The pull request is titled "Bultako Fix broken link in hess.ipynb". The pull request has 139 issues and 13 pull requests. The commit message is "Fix broken link in hess.ipynb".

File	Description
images	Fix Travis CI Sphinx war
analysis_3d.ipynb	Run black on tutorials/a
astro_dark_matter.ipynb	Copy published tutorial
astropy_introduction.ipynb	Fix astropy_introduction
background_model.ipynb	Update tutorials/backgr
cta_1dc_introduction.ipynb	Run black on tutorials

gammapy.maps



- gammapy.maps: maps, geometry and axis classes to work with sky image pixel data (2D, 3D, ND)
- A corner-store for developments and analyses in Gammapy
- Design by Matthew Wood, now developments continue by Atreyee Sinha (APC), Régis, Axel,...
- [Tutorial: intro_maps.ipynb](#)
- See also: [In-depth docs page](#)
- Next steps: coordinate unit handling, unify maps & IRFs, HEALPix, multi-resolution and sparse maps, map-based model fitting



Modeling and fitting



- New `gammapy.utils.fitting` with `Parameter`, `Parameters`, `Fit`
- Flexible fitting backends:
 - `iminuit` (default at the moment)
 - `Sherpa` (work in progress)
 - `scipy.optimize`, `emcee` (planned)
- Heavily inspired by `Sherpa`
<https://sherpa.readthedocs.io>
- A first version, will significantly improve in the coming months

```
from gammapy.image.models import SkyGaussian
from gammapy.spectrum.models import PowerLaw
from gammapy.cube.models import SkyModel

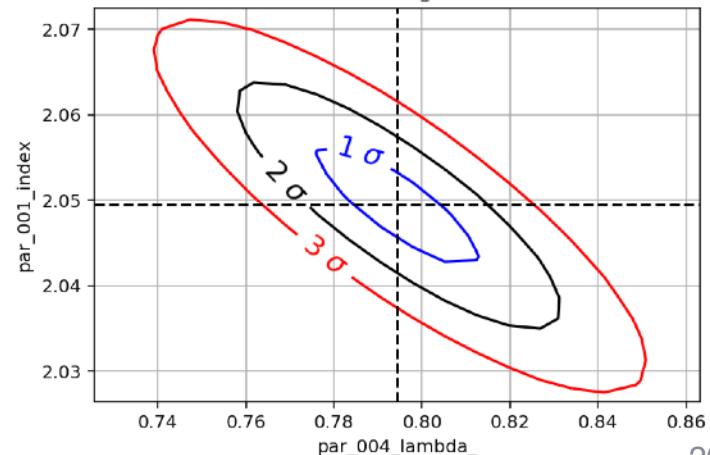
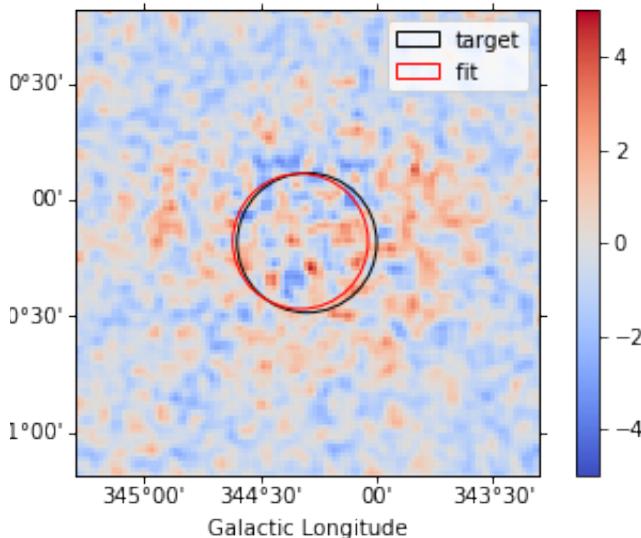
spatial_model = SkyGaussian(
    lon_0="0.2 deg",
    lat_0="0.1 deg",
    sigma="0.3 deg",
)
spectral_model = PowerLaw(
    index=3,
    amplitude="1e-11 cm-2 s-1 TeV-1",
    reference="1 TeV",
)
sky_model = SkyModel(
    spatial_model,
    spectral_model,
)
print(sky_model)
```

3D analysis: MapMaker and MapFit



- First version of high-level classes in Gammapy v0.8 to make maps (MapMaker) and fit models (MapEvaluator and MapFit).
- Example using CTA DC-1:
[Tutorial: analysis_3d.ipynb](#)
- You can do many studies for CTA already using binned simulation:
[Tutorial: simulate_3d.ipynb](#)
- A first version, will significantly improve in the coming months.

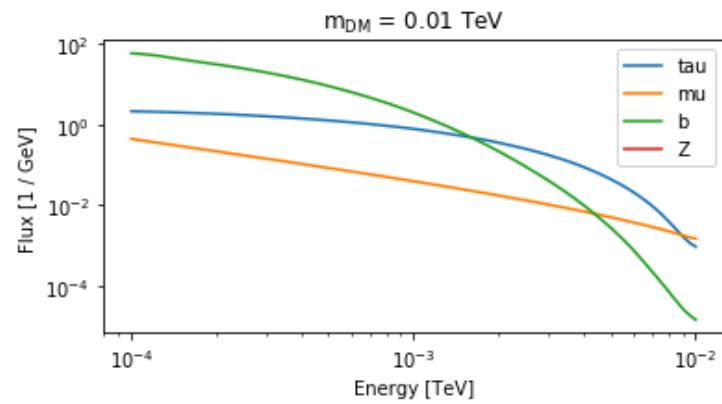
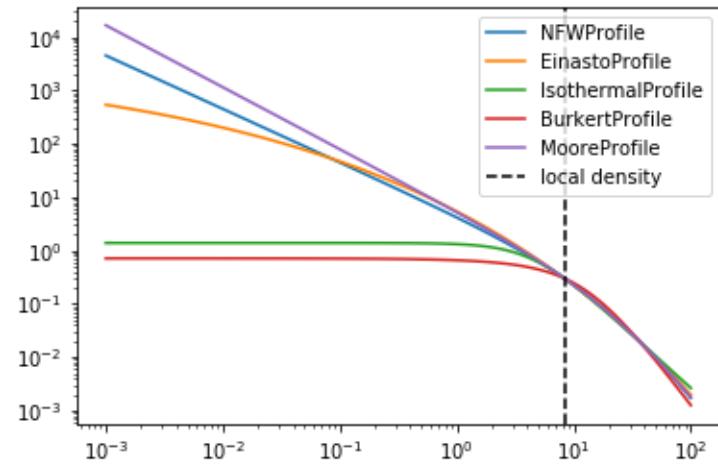
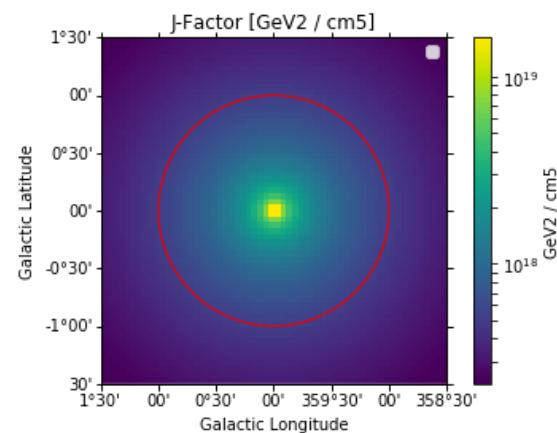
J1702+Gal IEM+PWN8+PWN42



gammapy.astro.darkmatter



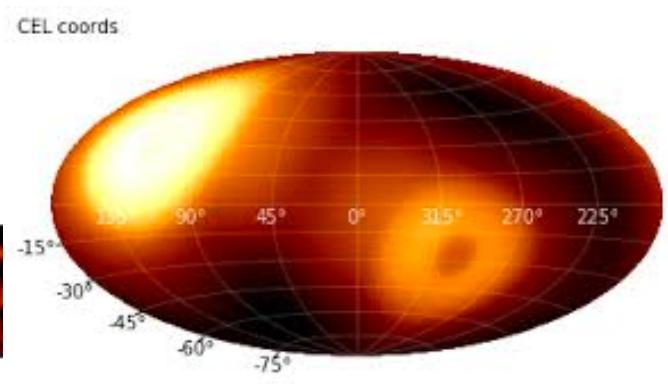
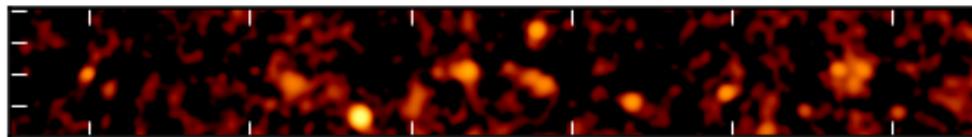
- New [`gammapy.astro.darkmatter`](#) with spatial and spectral models for indirect dark matter searches
- [Tutorial: astro_dark_matter.ipynb](#)
- From Johannes King (MPIK). Thank you for this contribution, and all your work on Gammapy over the past years!



Fermi-LAT with Gammapy



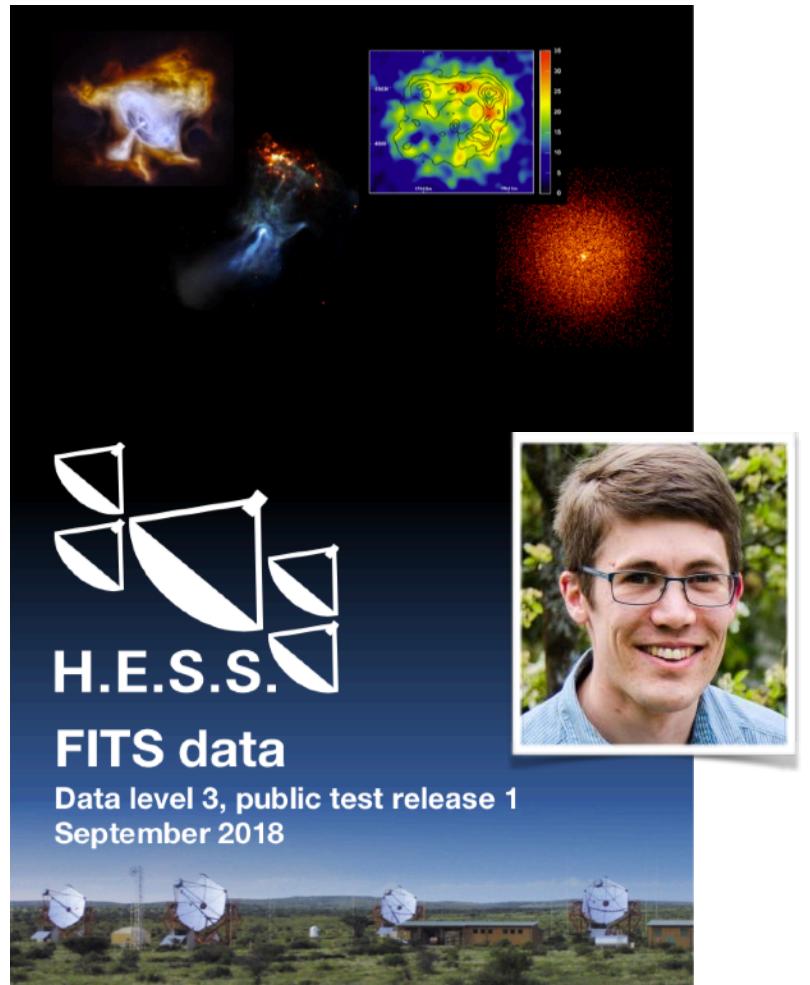
- Gammapy has currently some support for Fermi-LAT
- Use Fermi ST to select events, compute exposure and PSF
→ Model and fit with Gammapy
- For most analyses Fermi ST and [Fermipy](#) are the right choice.
Gammapy can be useful for multi-mission or special analyses
- [Tutorial: fermi_lat.ipynb](#)



H.E.S.S. with Gammapy



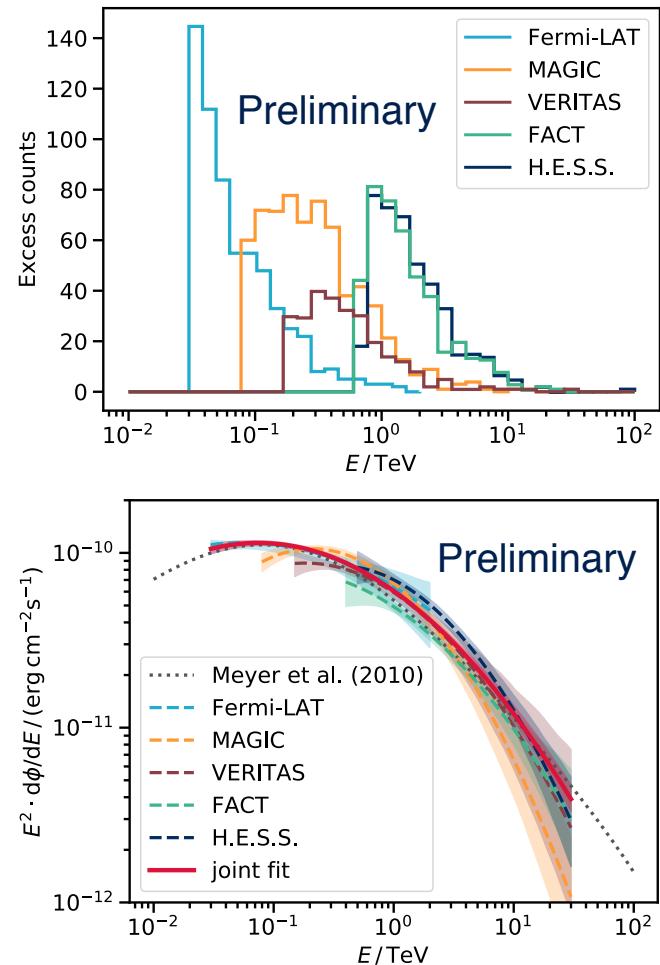
- Gammapy development and validation also with data from existing IACTs ongoing
- H.E.S.S. first public test data release last week ([webpage](#))
- Curated open test dataset for tutorials, tests, ST development
- Gammapy used to check the DL3 spec format compliance via DataStore.check() and some IRF and other data content checks
- [H.E.S.S. with Gammapy tutorial](#)
- Lars Mohrmann (Erlangen) working on a validation paper for H.E.S.S. FITS data with Gammapy & ctools



Joint Crab paper



- Paper by Cosimo Nigro, Tarek Hassan (DESY Zeuthen), Kai Brügge (Dortmund), Roberta, ... coming soon.
- Some Crab Nebula observations by H.E.S.S., MAGIC, VERITAS, FACT + Fermi-LAT
- Spectral analysis with Gammapy
- Method paper to show the power of open data and tools. Current DL3 format usable for existing IACTs and makes reproducible multi-instrument analysis possible.
- Flexibility of Python and Gammapy allows for joint likelihood, writing a custom likelihood to treat systematics is easy, [Naima](#) can be used for SED modelling.



Gammappy Twitter



- Gammappy is now on Twitter:
<https://twitter.com/gammappyST>
- Follow us to get Gammappy-related news!
- Jose Enrique Ruiz (Granada)

Thank you for tweeting, and all the work on docs, notebooks, datasets and reproducibility with Gammappy!



Jose Enrique Ruiz

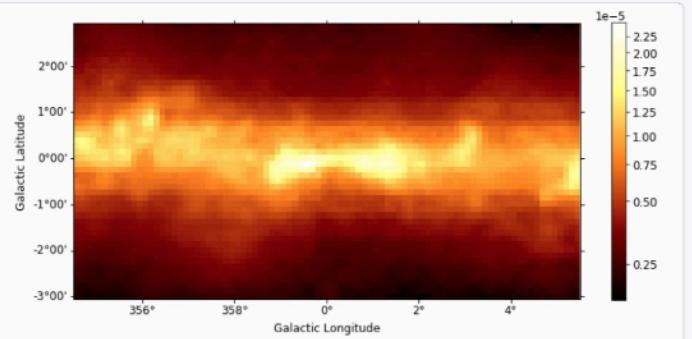
@bultako Follows you

Astro, Open Science, Jupyter Notebooks, Python, one of the best code bugs around.

Granada

iaa.es/~jer

 **Gammappy** @gammappyST · Sep 21
We will be in Berlin next Monday for #CTABerlin2018 in the Gammappy hands-on session, doing 3D analysis of @hesstelescopes and @CTA_Observatory DC1 data, and showing you how to use gammappy maps.
indico.cta-observatory.org/event/1711/tim...



A heatmap showing the distribution of gamma-ray flux in Galactic Longitude (x-axis, from 356° to 4°) and Galactic Latitude (y-axis, from -3°00' to 2°00'). The color scale ranges from 0.25 to 2.25 units of flux density, with a multiplier of 1e-5. The map shows a prominent emission feature centered around Galactic Longitude 0° and Galactic Latitude 0°.

2 3 0

 **Gammappy** @gammappyST · Aug 20
We will be in Madrid - UCM Faculty of Physics - hosted by Grupo Altas Energías for the next coding sprint. 

Gammappy Coding Sprint, October 2018, Madrid
Monday 1st - Friday 5th



gammappy/gammappy-meetings

Gammappy-related meetings (but also other topics!). Contribute to gammappy/gammappy-meetings development by creating an account on GitHub.

github.com

4

8

0

25

Summary and next steps



- Gammapy: core CTA ST functionality with 30k lines of simple Python code, flexible and extensible for advanced users
- Gammapy v0.8 is out: new maps and modelling, tutorial and dataset setup, dark matter, ...
- H.E.S.S. first test data release out, validation paper is work in progress. Joint Crab paper coming with data from other IACTs.
- Next steps:
 - Coding sprint next week (Oct 1-5) at UCM Madrid
 - Write up lessons learnt from DC-1
 - Write Gammapy roadmap and proposal for code organisation (sub-packages) and functionality for Gammapy v1.0
 - Continue work on data, IRFs, maps, modelling, ...
 - Continue to improve code and test quality, API uniformity, ...