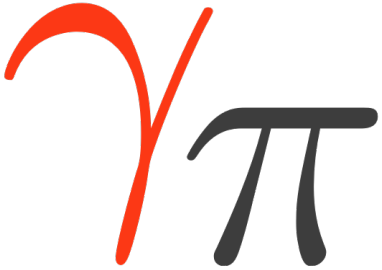


A **Python** package for
**gamma-ray** astronomy

# Gammapy in the Open Science

*Bruno Khélifi, for the Gammapy team*

CTA-France Meeting

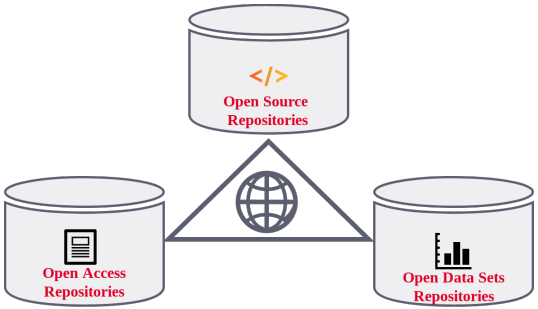
6 - 7 March 2024, Paris

# The Open Science landscape

"Making science more accessible, inclusive and equitable for the benefit of all" – UNESCO

- Berlin declaration (2003): free access to the Knowledge for Fundamental Science, Life Science, Human and Social Science

The three pillars:




Recommendations and roadmaps

EU, UNESCO, etc
States (MESRI)
NASA, CNRS, CEA, etc
Universities
Laboratories

# Software in the Open Science

I. Research Data Alliance:
   - FAIR4RS principles


II. L'Appel de Paris about the academic evaluation (2022)

# FAIR4RS principles

As with the FAIR Guiding Principles, the FAIR4RS Principles (2022) are intended to be aspirational. The application of the FAIR4RS Principles is the responsibility of the owners (who are often the creators) of the software, not the users.

- **F** Software, and its associated metadata, is easy for both humans and machines to find
- **A** Software, and its metadata, is retrievable via standardized protocols
- **I** Software interoperates with other software by exchanging data and/or metadata, and/or through interaction via application programming interfaces (APIs), described through standards.
- **R** Software is both usable (can be executed) and reusable (can be understood, modified, built upon, or incorporated into other software)

# FAIR4RS principles

**F** Software, and its associated metadata, is easy for both humans and machines to find

F1. Software is assigned a globally unique and persistent identifier.

# FAIR4RS principles

**F** Software, and its associated metadata, is easy for both humans and machines to find

F1. Software is assigned a globally unique and persistent identifier.

# FAIR4RS principles

**F** Software, and its associated metadata, is easy for both humans and machines to find

F1. Software is assigned a globally unique and persistent identifier.

F2. Software is described with rich metadata.

F3. Metadata clearly and explicitly include the identifier of the software they describe

F4. Metadata are FAIR, searchable and indexable.

# FAIR4RS principles

**F** Software, and its associated metadata, is easy for both humans and machines to find

F1. Software is assigned a globally unique and persistent identifier.

F2. Software is described with rich metadata.

F3. Metadata clearly and explicitly include the identifier of the software they describe

F4. Metadata are FAIR, searchable and indexable.

CodeMeta

Pioneer work made by ESCAPE/CEVO, Zenodo, SoftwareHeritage.

High-Energy Interest Group within IVOA is about to be created.
➲ Insertion within the Virtual Observatory underway (Observatoire de Paris, CDS, CTAO, etc)

# FAIR4RS principles

A Software, and its metadata, is retrievable via standardized protocols

A1. Software is retrievable by its identifier using a standardized communications protocol (open, free, and universally implementable)

A1.2. The protocol allows for an authentication and authorization procedure, where necessary

A2. Metadata are accessible, even when the software is no longer available

# FAIR4RS principles


Software, and its metadata, is retrievable via standardized protocols

A1. Software is retrievable by its identifier using a standardized communications protocol (open, free, and universally implementable)

A1.2. The protocol allows for an authentication and authorization procedure, where necessary

A2. Metadata are accessible, even when the software is no longer available

# FAIR4RS principles

**I** Software interoperates with other software by exchanging data and/or metadata, and/or through interaction via application programming interfaces (APIs), described through standards.

I1. Software reads, writes and exchanges data in a way that meets domain-relevant community standards.

**Principle associated with the FAIR principles of the data**

# FAIR4RS principles

**I** Software interoperates with other software by exchanging data and/or metadata, and/or through interaction via application programming interfaces (APIs), described through standards.

I1. Software reads, writes and exchanges data in a way that meets domain-relevant community standards.

VODF very-high-energy open data format

IVOA

Experimental data: GADF format is de-facto a standard. Inclusion of the future VODF format and the future IVOA recommendations: TBD.

Modelling data: the eco-system of astrophysical libraries is growing. Standards/recommendations TBD.

Gammapy data: outputs of Gammapy should be FAIR. Underway...

# FAIR4RS principles

**I** Software interoperates with other software by exchanging data and/or metadata, and/or through interaction via application programming interfaces (APIs), described through standards.

I1. Software reads, writes and exchanges data in a way that meets domain-relevant community standards.

Python eco-system: Gammapy uses libraries (ex: astropy) and other libraries (ex: Fermipy) use Gammapy
Our rigorous design using Python standards permits this interoperability.
Ex: Naima, agnpy, pyirf, pyswgo, COSMICS under discussion

# FAIR4RS principles

**I** Software interoperates with other software by exchanging data and/or metadata, and/or through interaction via application programming interfaces (APIs), described through standards.

I1. Software reads, writes and exchanges data in a way that meets domain-relevant community standards.

I2. Software includes qualified references to other objects.

# FAIR4RS principles

**I** Software interoperates with other software by exchanging data and/or metadata, and/or through interaction via application programming interfaces (APIs), described through standards.

I1. Software reads, writes and exchanges data in a way that meets domain-relevant community standards.

I2. Software includes qualified references to other objects.

Within Gammapy, the metadata and our documentation refer to other libraries.

Is it enough? Is is reciprocal?

➲ Notion of `associated software' is emerging

# FAIR4RS principles

R Software is both usable (can be executed) and reusable (can be understood, modified, built upon, or incorporated into other software)

R1. Software is described with a plurality of accurate and relevant attributes

R1.1. Software is given a clear and accessible license

R1.2. Software is associated with detailed provenance

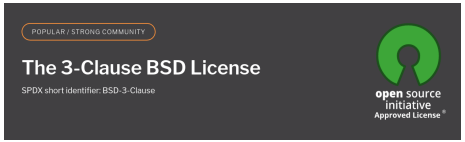
# FAIR4RS principles

Software is both usable (can be executed) and reusable (can be understood, modified, built upon, or incorporated into other software)


R1. Software is described with a plurality of accurate and relevant attributes

R1.1. Software is given a clear and accessible license

R1.2. Software is associated with detailed provenance




Provenance: to be done
See Servillat (2022) SF2A

# FAIR4RS principles

R Software is both usable (can be executed) and reusable (can be understood, modified, built upon, or incorporated into other software)

R1. Software is described with a plurality of accurate and relevant attributes

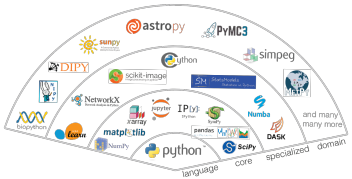
R2. Software includes qualified references to other software

# FAIR4RS principles

R Software is both usable (can be executed) and reusable (can be understood, modified, built upon, or incorporated into other software)

R1. Software is described with a plurality of accurate and relevant attributes

R2. Software includes qualified references to other software




Metadata and documentation mention our dependencies

Our documentation starts to precise the `associated libraries' (Fermipy, Naima, agnpy, pyirf, pyswgo, etc).

Better reference and better coordination: TBD.

# FAIR4RS principles

R Software is both usable (can be executed) and reusable (can be understood, modified, built upon, or incorporated into other software)

R1. Software is described with a plurality of accurate and relevant attributes

R2. Software includes qualified references to other software

R3. Software meets domain-relevant community standards

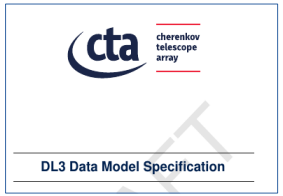
# FAIR4RS principles

R Software is both usable (can be executed) and reusable (can be understood, modified, built upon, or incorporated into other software)

R1. Software is described with a plurality of accurate and relevant attributes

R2. Software includes qualified references to other software

R3. Software meets domain-relevant community standards




CTAO provides in-depth formalism of our TeV knowledge.

KM3NeT has also this potential: cross-link TDB.

# Academic evaluation

In phase with the Appel de Paris, we pay a special attention to

Recognition of contributions and Valorization of activities


**Paris Call on Research Assessment**

*This text was prepared by the French Open Science Committee and presented to the Paris Open Science European Conference (OSEC) held in Paris on 4th and 5th February 2022, organised in the context of the French Presidency of the Council of the European Union, following the publication of the UNESCO recommendation on Open Science and the publication by the European Commission of Towards a reform of the research assessment system: scoping report.*

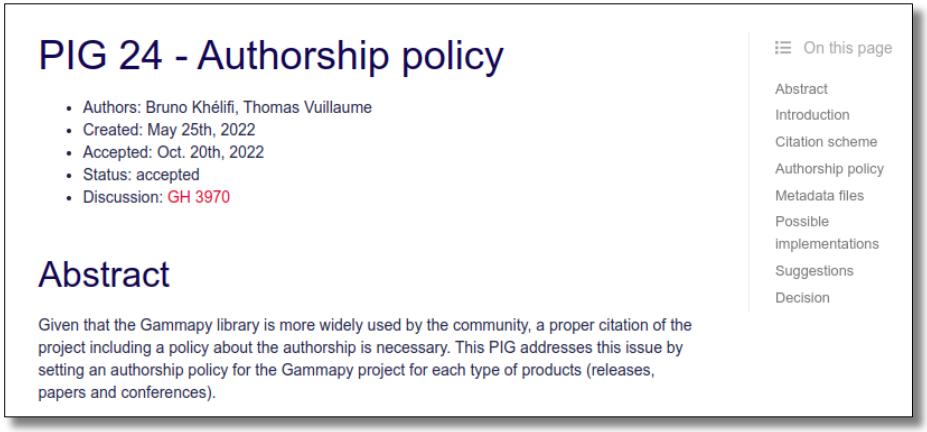
In the conclusions of its meeting of December 1st 2020 on 'the New European Research Area', the Competitiveness Council of the European Union highlighted that Open Science has a crucial role in boosting impact, quality, efficiency, transparency and integrity of research and innovation, and brings science and society closer together. The Council emphasised that bibliodiversity and multilingualism and the acknowledgement of all scientific productions are relevant elements of an European Research Area policy on Open Science.

The current system for assessing research, researchers and research institutions, however, does not incentivise or reward enough the quality of all research outputs in their diversity. It often relies on the quantity of publications in journals with high Journal Impact Factor and citations as mere proxies for quality and impact, thereby underestimating the value of other contributions, lowering reproducibility and holding back researchers from open sharing and collaboration.

# Academic evaluation

In phase with the Appel de Paris, we pay a special attention to

Recognition of contributions and Valorization of activities



## PIG 24 - Authorship policy

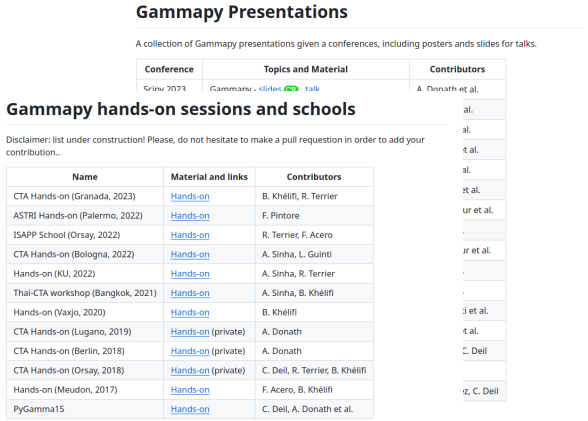
- Authors: Bruno Khélifi, Thomas Vuillaume
- Created: May 25th, 2022
- Accepted: Oct. 20th, 2022
- Status: accepted
- Discussion: GH 3970

On this page

Abstract
Introduction
Citation scheme
Authorship policy
Metadata files
Possible implementations
Suggestions
Decision

### Abstract

Given that the Gammapy library is more widely used by the community, a proper citation of the project including a policy about the authorship is necessary. This PIG addresses this issue by setting an authorship policy for the Gammapy project for each type of products (releases, papers and conferences).



### Gammapy Presentations

A collection of Gammapy presentations given a conferences, including posters ands slides for talks.

| Conference | Topics and Material | Contributors |
|---|---|---|
| Scipy 2023 | Gammapy - slides talk | A. Donath et al. |

### Gammapy hands-on sessions and schools

Disclaimer: list under construction! Please, do not hesitate to make a pull requestion in order to add your contribution..

| Name | Material and links | Contributors |
|---|---|---|
| CTA Hands-on (Granada, 2023) | Hands-on | B. Khélifi, R. Terrier |
| ASTRI Hands-on (Palermo, 2022) | Hands-on | F. Pintore |
| ISAPP School (Orsay, 2022) | Hands-on | R. Terrier, F. Acero |
| CTA Hands-on (Bologna, 2022) | Hands-on | A. Sinha, L. Guinti |
| Hands-on (KU, 2022) | Hands-on | A. Sinha, R. Terrier |
| Thai-CTA workshop (Bangkok, 2021) | Hands-on | A. Sinha, B. Khélifi |
| Hands-on (Vaxjo, 2020) | Hands-on | B. Khélifi |
| CTA Hands-on (Lugano, 2019) | Hands-on (private) | A. Donath |
| CTA Hands-on (Berlin, 2018) | Hands-on (private) | A. Donath |
| CTA Hands-on (Orsay, 2018) | Hands-on (private) | C. Deil, R. Terrier, B. Khélifi |
| Hands-on (Meudon, 2017) | Hands-on | F. Acero, B. Khélifi |
| PyGamma15 | Hands-on | C. Deil, A. Donath et al. |

<u>Each Gammapy release is an official publication</u>

- The SWH/ZenodoDOI has an author list
- **This transparency is very rare for a software!**

<u>Each presentation, hands-on session, school is promoted</u>

# Academic evaluation

In phase with the Appel de Paris, we pay a special attention to

Recognition of contributions and Valorization of activities

Anyone contributing to Gammapy can:

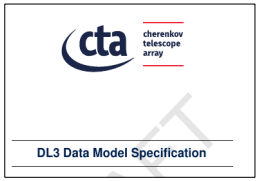
- develop her/his software skills with an high degree of quality
- enrich a highlight product of research (algorithm, core func., interoperability, DevOps)
- participate to the development of Open Science

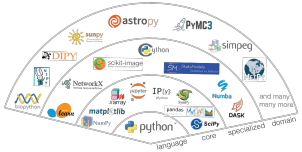
And any developer profile can bring its skills:

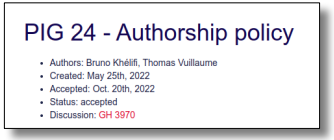
- Engineer, under-grad up to PhD student, post-doc, experimentalist, data scientist, data analyser, theorist, etc
- From any lab, any experiment, any country

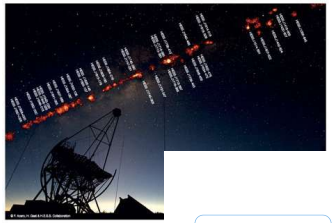
# Outline and Conclusion

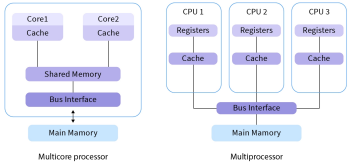
- Many activities beyond the pure coding
  - This presentation aims to present/remind the work made by the Gammapy team that goes beyond the Gammapy library and the CTAO SAT
- Gammapy is an open research software within the Python eco-system and follows the Open Science initiatives






# Thank you for your attention

# and

# Looking forward to meet you in Fall for the first `École Gammapy'!

Avec le soutien de l'API AHE de l'ObsParis

A **Python** package for
**gamma-ray** astronomy