# Gammapy coding sprint introduction

Christoph Deil (MPIK)
Feb 5, 2018

# Introduction

- I will skip the usual Gammapy introduction, except
  let's have a look at gammapy.org and docs.gammapy.org

- We still have prototype 1D/2D/3D analysis code,
  but everything needs more work

  - For 1D spectra we polish what we have

  - For 2D / 3D maps we are developing something new

  - See the following presentations today

# Organisation

- Today brief status reports and discussion on tasks

- Ideally small teams (usually 2 or 3 people) form for each task that then proceed independently

- In-depth discussion & start work this week, continue work and collaboration in the coming months

- Short report back Thursday afternoon to the whole group

- Bi-weekly regular Gammapy user / dev calls (Friday 11 am). Probably separate focused calls as needed on each topic / in the small team would work best to get it done.

- Gammapy 2018 second coding sprint in June / July https://goo.gl/forms/493orc8xrkg1QQYK2

# Today

- General Gammapy status update (C. Deil)

- Gammapy in CTA; 1DC R. Zanin

- Gammapy in HESS, and joint Crab paper with MAGIC and Fermi-LAT (C. Deil)

- IRFs: K. Kosack

- PIG 2 - New low-level analysis code: R. Terrier

- Modeling (TODO: add link to PIG): C. Deil

- Spectrum: R. Terrier

- Light Curves: J. Lefaucheur

- Sensitivity scripts: B. Khelifi

- High Level Interface: C. Deil

- Documentation: R. Zanin

- AoB, email Bruno if you want to discuss or present something!

# User ~ contributor

- Probably only a few people here can spend a significant amount of time to contribute to Gammapy. This is normal for science codes.

- If you're mainly here because you want to use Gammapy:

  - Hopefully the talks today and tutorials this week will be useful for you!

  - If you have any questions about coding or analysis, use this week to ask an expert!

  - If you want to attempt a first pull request: start with a small edit to the documentation or one function or class. Something that takes an hour, not a day or a week. Try pair coding, i.e. now that we're in the same room, sit together with someone that helps you and then reviews and mergers your pull request directly.

- All: please fill the Gammapy installation questionnaire! https://goo.gl/forms/0QuYYyyPCbKnFJJl3

# Tuesday - Friday

- Gammapy developer tutorial
  https://github.com/gammapy/gammapy-dev-tutorial
  Tuesday, Wednesday, Thursday 9:00 - 12:30
  Please have a look at the "Prepare" section

- In parallel in the morning, and for everyone in the afternoons: completely free to work on what you like / in small groups

- Maybe we can fix times for some sessions on the board, and avoid parallel discussion sessions?

# Developer tutorials

This is just a suggestion. Let me know what you want to do!
Bring questions / ask anything concerning Python, Git, Gammapy, …!
If one of you wants to show or teach something, let me know!

**Tuesday 9:00 - 10:30**

- Executing Python code (Python, ipython, Jupyter); understand imports, modules, def, class, ...
- Installing Python packages (setuptools, pip, conda); understand sys.path and sys.modules

**Tuesday 11:00 - 12:30**

- Writing Python code (with an editor like emacs/Atom/Visual Studio Code or an IDE like PyCharm)
- Debugging Python code (print, with IPython, with PyCharm)
- Reading / navigating a large codebase (Gammapy, using PyCharm)

**Wednesday 9:00 - 10:30**

- Writing automated tests (with pytest)
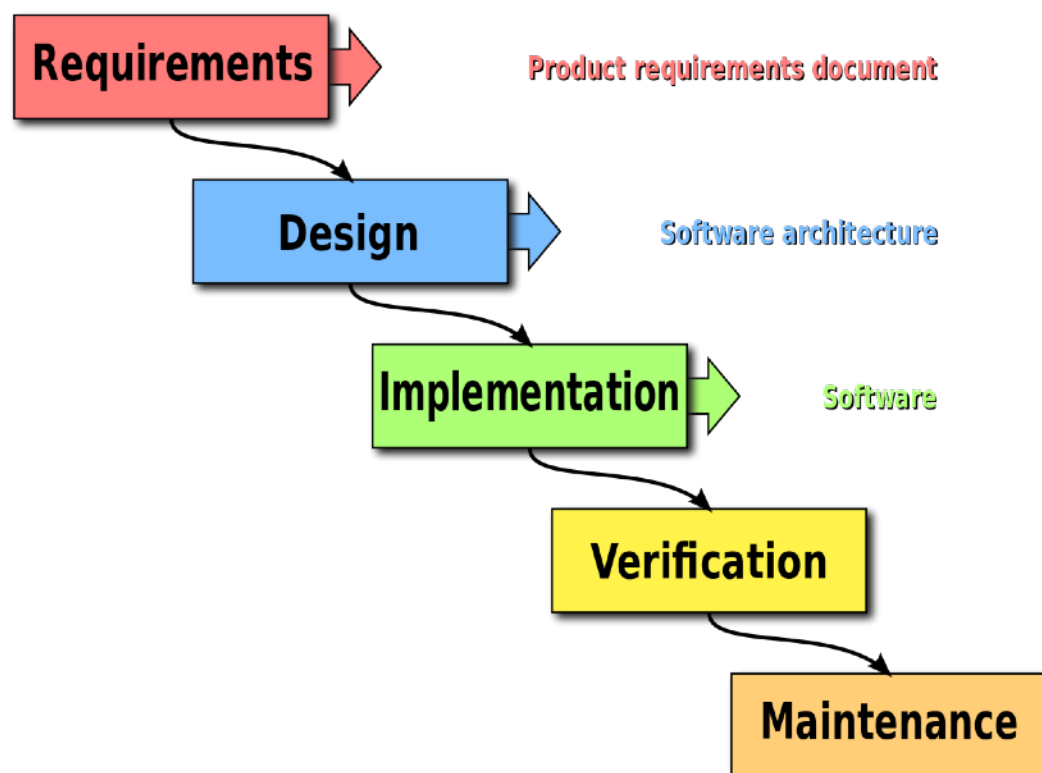- Writing documentation (with ReStructuredText and Sphinx)

**Wednesday 11:00 - 12:30**

- Learn git and Github
- Code review from the contributor and reviewer perspective

**Thursday 9:00 - 10:30**

- A big exercise for the whole workflow, to be done in pairs. Team up with your neighbor now!

# How to develop Gammapy?

## Waterfall

**Requirements** → Product requirements document

**Design** → Software architecture

**Implementation** → Software

**Verification**

**Maintenance**

## Agile

- **Individuals and Interactions** *over processes and tools*
- **Working Software** *over comprehensive documentation*
- **Customer Collaboration** *over contract negotiation*
- **Responding to Change** *over following a plan*

*There's many software development approaches, most for professional teams, not astronomers coding on the side.*
**What do we do?**
**What's good, what's bad?**

- Recently we introduced the PIGs for Gammapy. There have been discussions how to organise the work and distribute responsibility. The last months development hasn't been very active.

- If anyone has thoughts what does and doesn't work well for Gammapy development (or experience from similar software development projects e.g. in H.E.S.S. or CTA), we could discuss a bit now, or anytime this week.

# Today

- General Gammapy status update (C. Deil)

- Gammapy in CTA; 1DC R. Zanin

- Gammapy in HESS, and joint Crab paper with MAGIC and Fermi-LAT (C. Deil)

- IRFs: K. Kosack

- PIG 2 - New low-level analysis code: R. Terrier

- Modeling (TODO: add link to PIG): C. Deil

- Spectrum: R. Terrier

- Light Curves: J. Lefaucheur

- Sensitivity scripts: B. Khelifi

- High Level Interface: C. Deil

- Documentation: R. Zanin

- AoB, email Bruno if you want to discuss or present something!