# fornax-demo-notebooks Onboarding

11 September 2025

# People (as of 11 September 2025)

- **Maintainers - IRSA Data Science Team**
  - **Troy Raen**
    - Team Lead. Developer: big catalogs (Parquet and HATS/LSDB); cloud services; alert streams. Astrophysics: stellar evolution.
  - **Brigitta Sipőcz**
    - Developer: maintainer of python libraries (pyvo, astroquery, astropy), python infrastructure, notebook infrastructure. Astrophysics: exoplanets and low mass stars
  - **Jessica Krick**
    - Scientist: galaxy clusters; exoplanets
  - **Andreas Faisst**
    - Scientist: formation of the first galaxies; JWST, ALMA, SPHEREx, Euclid
  - **Jaladh Singhal**
    - Developer: maintainer of IPAC's Firefly Python tools (astronomical data access & visualization), collaborator at TARDIS-RT, and contributor at several open source projects.

# People (as of 11 September 2025)

- Contributors
  - Jessica Krick (IRSA)
  - Troy Raen (IRSA)
  - Brigitta Sipőcz (IRSA)
  - Shoubaneh Hemmati (IRSA)
  - Andreas Faisst (IRSA)
  - Abdu Zoghbi (HEASARC)
  - David Shupe (IRSA)
  - Tom Donaldson (STScI)
  - Vandana Desai (IRSA)
  - Zach Claytor (STScI)
  - Jaladh Singhal (IRSA)
  - Sam Bianco (STScI)
  - Marziye JafariYazani (IRSA)

# Fornax Demo Notebooks – README.md

The Fornax Initiative is a NASA Astrophysics Archives project to collaboratively among the three archives HEASARC, IRSA, and MAST, create *cloud* systems, *cloud* software, and *cloud* standards for the astronomical community.

The **Fornax Science Console** is a *cloud* compute system near to NASA data on the AWS *cloud* which provides a **place where astronomers can do data-intensive research** with reduced barriers. The Fornax Initiative provides **increased compute, increased memory, increased ease of use by pre-installing astronomical software, increased reproducibility of big data results**, and increased inclusion by removing some of these barriers to entry, tutorial notebooks, and documentation.

This repo houses tutorial notebooks of **fully worked science use cases for all users**. Common goals of the notebooks are the **usage of archival data from all NASA archives, cross-archive work, big data, and computationally intensive science**. …

# Developing a Notebook

## Choose a science use case

- Combine data from multiple NASA archives.
  - This is a requirement for this repo that we have not compromised on.
- Interesting and relevant to Fornax Science Console users.
  - The goal of this repo is to support Fornax Initiative work and Console users.
- Data intensive and/or computationally intensive.
  - These are user needs that the Fornax Science Console is especially well suited for, so they are good things for notebook use cases to include.

This is not about producing original science.

It’s about helping Fornax users do their original science efficiently.

# Developing a Notebook

## Choose a science use case

There are a relatively small number of starting points for a large number of astronomy questions. These make good starting points for our notebooks. Examples (notebook motivating use cases in parentheses):

- collect multi-wavelength light curves (identify AGN subtypes and compare their behavior over time)
- collect spectra (physics of stars, galaxies, and AGN)
- perform forced photometry on images (study galaxy evolution)
- cross match catalogs (time-series photometry of stars)

Users want to focus on their science, not on technical difficulties. Provide robust solutions to common technical challenges so that users can easily apply them.

# Developing a Notebook

## Choose a science use case

Choose a science use case because it will guide important decisions that need to be made:

- Which datasets should be used? Why? What can be learned by combining them?
- What is the most efficient way to access the data?
  - Ask the archive what they recommend.
  - Prefer cloud data when it's available. Large-scale access is much more efficient than with traditional methods.
  - Don't expect a method that works well at small scale to automatically work well at large scale. Test it. Sort out the problems you encounter. Don't leave the user to figure this out on their own.
- What do users need to know about each dataset in order to use it appropriately?
  - For example, data filtering and cleaning.
- Most of our use cases benefit from parallelization. How should it be done?
  - How to parallelize efficiently can depend strongly on the use case. Demonstrate a solution for your use case.

# Developing a Notebook

## Use the template

https://github.com/nasa-fornax/fornax-demo-notebooks/tree/main/template

- Notebook (.md) and requirements (.txt) templates
  - Make a copy of both and use them as your starting point.
- Science and tech review checklists
  - These are the requirements for notebooks in this repo.
  - Notebooks must pass PR reviews based on these checklists before they will be published.
- Instructions are included in all files
  - These will be updated over the next few weeks as we continue to get our current processes written down.

Notebooks are organized into directories by topic.

- A new notebook should go into an existing directory if an appropriate one exists.

# Adding a Notebook: A Two-step Process

## 1. In to the repo

https://github.com/nasa-fornax/fornax-demo-notebooks/

- You can open the PR at any point in your notebook development process.
  - Open it as a draft PR if the notebook isn't ready to be reviewed yet. Once ready, click "Ready for review" to convert it.
- The PR will be reviewed by at least one repo maintainer.
  - Is the notebook in line with the repo's goals? Will give feedback about areas that need attention / changes / improvements / further development.
  - The notebook does not have to pass the science and tech review checklists at this point, but the reviewer will request minimum standards before merging.
  - (Optional) If you prefer, you can fully develop the notebook and request the science and tech checklists reviews in this step.

# Adding a Notebook: A Two-step Process

## 2. In to the published tutorials

https://nasa-fornax.github.io/fornax-demo-notebooks/

- (Optional) Continue to develop the notebook and submit additional PRs.
  - PR expectations are the same as in step 1.
- When your final PR is ready, request science and tech checklists reviews.
  - (Optional) This may be done in step 1.
  - For now, these reviews should be done by repo maintainers. In the future, this may expand to include others who have contributed notebooks.
- After your final PR is merged, a repo maintainer will add it to the published tutorials.
  - This is done through CI/CD.
  - Once it has been added, future changes to the notebook will be automatically pushed to the published website when the PR is merged.

# Ongoing Updates and Maintenance

- (Optional) You are welcome to continue iterating and improving the notebook.
  - These PRs don’t need to go through checklist reviews again, but those standards should be maintained.
- Be prepared to maintain the notebook.
  - The notebook’s first author has primary responsibility. If the first author leaves, primary responsibility will fall to the archive that the author was affiliated with.
  - It is common for bugs to crop up due to changes in archive services, python libraries, etc. You are responsible for addressing these issues. The notebook should continue to run bug free.
  - When users submit questions or bug reports about your notebook, you will be responsible for handling them.
  - If all dependencies are pip installable, the notebook should be automatically tested through CI/CD. If it cannot be automatically tested for any reason, you need to complete ongoing manual testing.
  - Notebooks that are not being maintained will be removed from the published tutorials.