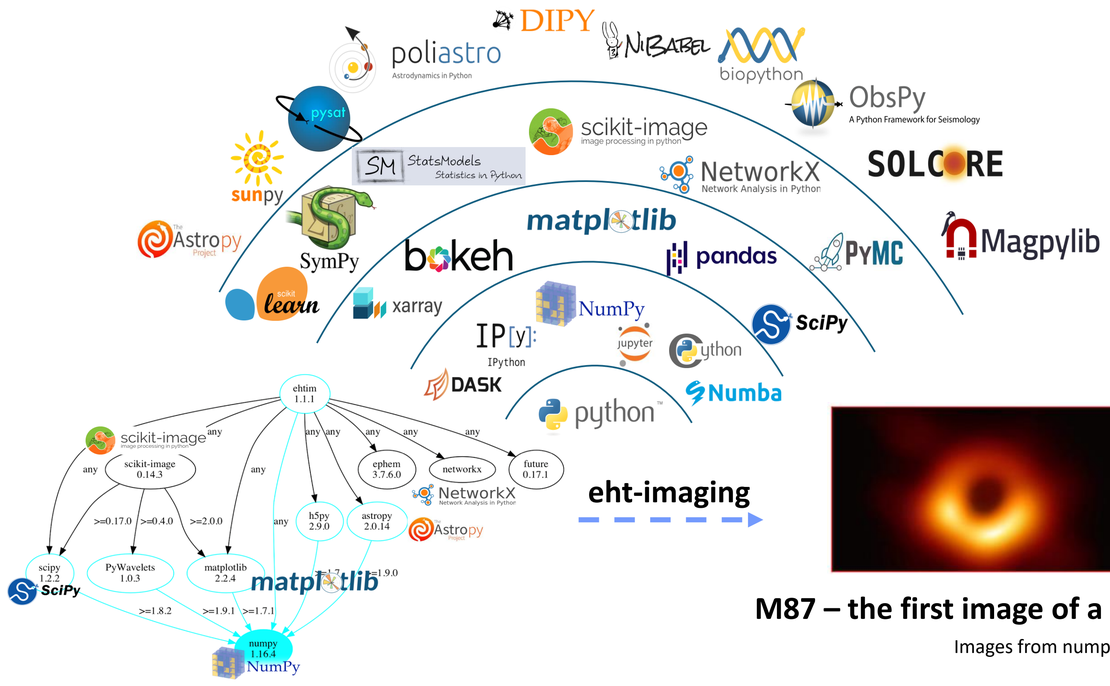
Sustaining Scientific Open-Source Software Ecosystems: Challenges, Practices, and Opportunities

Jiayi Sun (jysun@ece.utoronto.ca) University of Toronto



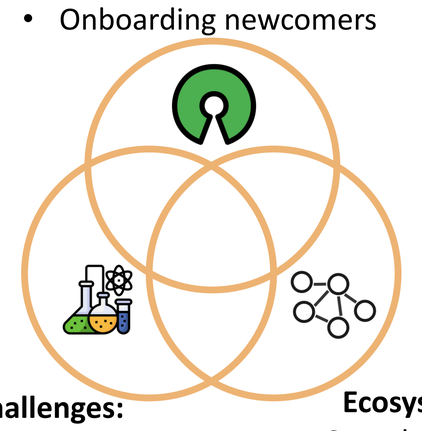
eht-imaging

M87 – the first image of a black hole

Images from numpy.org

Scientific open-source software (scientific OSS) ecosystems facilitate scientific research endeavors by streamlining research workflows.

OSS community challenges:
- Maintainers burnout
- Onboarding newcomers



Scientific software challenges:
- Interdisciplinary collaboration

Ecosystem challenges:
- Complex dependencies risks
- Multi-projects collaboration

Multifaceted challenges to sustain scientific OSS ecosystems: what are the emerging challenges and do the existing solutions & best practices still apply?

Research Plan Overview

Obj.1: Understanding the challenges in collaborative scientific OSS development — Work In Progress

Obj.2: Understanding the challenges of cross-project collaboration in the scientific OSS ecosystem — Case study with Astropy; Quantitative study with broader ecosystem

Obj.3: Designing interventions to support sustainable scientific OSS development — Work In Progress; Future Work

- Mindset conflicts on task prioritization
- High contribution barrier for code and science
- Redundant development in different projects within ecosystem
- Inefficient knowledge sharing across projects
- …

Problems

Tooling support	Domain-specific code review recommendation	Automated refactoring for improving code contribution from scientists	…
	Connecting code modules with required science for Good First Issue	Measuring the impact of code contribution	
Best practices	Development infrastructure support for onboarding scientists	Academic recognition & credit for code contribution	

Solution

Preliminary Results (Obj.1)

Interdisciplinary collaboration challenges in scientific OSS

Interview

Infrastructure-related contribution — Science-related contribution

While most contributor have scientific background, contribution preference spread across a spectrum.

TAKE AWAY

Mindset differences (**science-first** vs. **engineering-first**) can cause tension, impacting communication and decision-making.

*"…if a random **scientist contributes their very precious code**, which we really appreciate. But they copy paste it straight from their thesis…it needs a lot of cleanup; it has no test and documentation or only they can understand. **Then it becomes really hard to explain to them** why the thing from the thesis is not good enough for the package…"*

*"…**it's hard to reconcile that with pure software developers**, who have rigid standards that they want to adhere to, or who aren't as familiar with a slightly more flexible nature of scientific software collaboration…"*

OSS challenges for scientific community

Survey

Different from general OSS, which sees a growing trend of contributions driven by altruism and learning, the main motivation for scientific OSS contributions remains to be own-use.

TAKE AWAY

Inevitable turnover due to the shift of contributors' career focus: **Lowering the entry barrier** for both **science** and **code** to encourage newcomers to contribute.

Top motivation for contributing: **own-use**

"I had a specific research project and I saw a missing functionality"

Top reason for disengagement: **focus shift**

"After graduating I took a postdoc working on a different open-source astronomy project "

Opportunity: tooling support could help to better organize "Good First Issues" by **linking code** modules with required **scientific knowledge**.

Cross-project collaboration challenges in ecosystem

Cross-project communication links

Beyond dependency-related issues, people also collaborate across-projects because of:

- Knowledge sharing
- Coordinating shared functionalities
- Centralized infrastructure management

TAKE AWAY

Contributors **collaborate** across projects **beyond bugs fixing** between upstream and downstream packages: difficult to keep track of overview.

E.g., Cross-project knowledge sharing