

# Sustainability and software ecosystems



James Howison UT Austin

- Sustainability means having the resources available to do the work needed to keep code scientifically useful, as the world changes.
  - Reduce the work needed (modularity, automation esp. CI/CD, “quality”)
  - Attract the resources needed to do the work.
- Beyond commercialization, peer production promising route to sustainability (aka “open source community”)
- Studied 114 grants funded the the NSF sustainability focused SI2 program, through online presence and interviews.
- Emerging results on the importance of being in ecosystems and the type of work needed in ecosystems.

**RQ:** How can grant funded organizations transition their software to (some) peer production?

**Option 1:** Change your organization so that it becomes a peer production community through **transmutation**

**Option 2:** Hand over the software to an existing peer production community through **migration** or through **transposition**

Configuration at End of Grant

		peer production	tool group	lab	author group	business	consortium
Configuration at Start of Grant	peer production	15				2	
	tool group	5	27	4		1	1
	lab		1	41*	1	1	
	author group	2	3	1	6		1
	business		1				
	consortium						1

1 Tool Group to PP transpositions

4 Tool Group to PP transmutations

1 Author Group to PP transposition

**Only 7 of 99 projects transitioned to peer production (15 began as peer production).**

# Guidance



James Howison UT Austin

- Transition is *organizational change*. And it is *hard*.
  - Best route was beginning as peer production. Which suggests funding projects already doing peer production.
  - Some examples of sustainability through handoffs to existing communities planned from the start (innovations with a *destination*)
- Also participation by participants experienced with open tooling (git, CI/CD, issue trackers) and open collaboration (decision making, mentoring).
  - Hire for open collaboration experience, mentor staff on collaboration.
- Being embedded in local software ecosystems seems to matter
  - Upstream and downstream dependencies are a pipeline for skilled, motivated, contributors.

# Ecosystem Work



James Howison UT Austin

- Ecosystems of package interdependencies and complementary uses
- Ecosystems form around:
  - Specific languages and *sufficiently broad* shared tasks
    - R's tidyverse, or the NumPy/Pandas/Jupyter Notebook world, Eclipse, Linux Foundation
  - Specific packaging systems (e.g., pip, cran, npm, deb)
  - Users who *redistribute code*, not so much end-users.
- The work is beyond maintenance:
  1. Sensing: How are ecosystem packages being combined?
  2. Adjustment: Reacting to nearby changes/opportunities, upstreaming.
  3. Synchronization: Managing change cascades through shared timed releases

*"Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Networking and Information Technology Research and Development Program."*

The Networking and Information Technology Research and Development  
(NITRD) Program

**Mailing Address:** NCO/NITRD, 2415 Eisenhower Avenue, Alexandria, VA 22314

**Physical Address:** 490 L'Enfant Plaza SW, Suite 8001, Washington, DC 20024, USA Tel: 202-459-9674,  
Fax: 202-459-9673, Email: [nco@nitrd.gov](mailto:nco@nitrd.gov), Website: <https://www.nitrd.gov>

