Open Source Funding

Open funding models and strategies for open research and communities

The Economics of Open Source

By Neil P. Chue Hong (@npch)

Open Source is free...

Photo of John Lewis by Unseen Histories on Unsplash

Photo by Pradnyal Gandhi on Unsplash

Photo by Jari Hytönen on Unsplash

Goods and Resources

- Rivalrous vs Non-rivalrous
  - Does the consumption by one person prevent simultaneous consumption by others?
- Excludable vs Non-excludable
  - Can you prevent people who haven't paid for it from having access to it?

By this definition, open source looks like a public good

In reality, open source is more like a common-pool resource because it takes effort to maintain the resource so everyone can benefit

	Excludable	Non-excludable
Rivalrous	**Private goods** Food, clothing, cars, parking spaces	**Common-pool resources** Fish stocks, timber, coal
Non-rivalrous	**Club goods** Cinemas, private parks, satellite television	**Public goods** Free-to-air television, air, national defense

https://en.wikipedia.org/wiki/Club_good

The Unseen Labor Behind Our Digital Infrastructure

"Nearly all software today relies on open source code, written and maintained by communities of developers and other talent. Much like roads or bridges, which anyone can walk or drive on, open source code can be used by anyone—from companies to individuals—to build software. This type of code makes up the digital infrastructure of our society today.

Just like physical infrastructure, digital infrastructure needs regular upkeep and maintenance. But financial support for digital infrastructure is much harder to come by. Currently, any financial support usually comes through sponsorships, direct or indirect, from software companies."

- Nadia Eghbal

https://www.fordfoundation.org/work/learning/research-reports/roads-and-bridges-the-unseen-labor-behind-our-digital-infrastructure/

Lifecycle of an Open Source Project

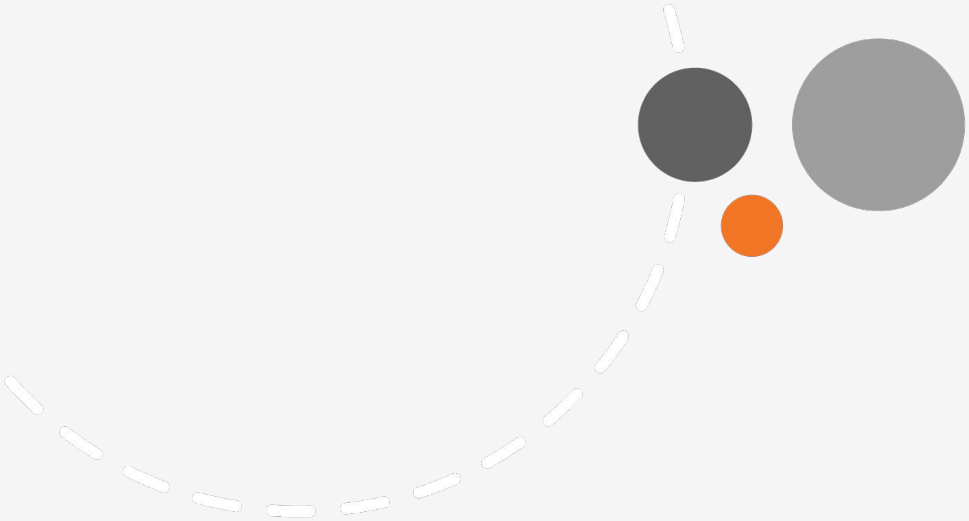
Different stages have different funding requirements and mechanisms

Stage	Key Milestones	Developers	Users	Support	TRL
Concept	Idea realised / tested	Creators	Creators	None / Limited	TRL1-2
Prototype	Minimal viable functionality, open source license	Creators / Close Collaborators	Creators / Close Collaborators	None / Limited	TRL3-4
Community Project	Core features others want, shared publicly, external contributors, governance model	Loose contributor community	Community	Community support	TRL5-7
Product	Release management, integrated w/ other software, more specific roles	Organised contributor community	Community and other end users	Formal support mechanisms	TRL8
Maintained	Undergoing continuous development with roadmap, self-sustaining	International, scaling, developer community	Any potential user	Formal support mechanisms	TRL9

Understanding the open source lifecycle: https://www.redhat.com/cms/managed-files/cp-open-source-software-life-cycle-brief-f23003pr-202004-en.pdf
Open Community Development Cycle: http://urssi.us/blog/2019/02/25/software-incubator-workshop-a-synthesis/
Routes to Sustainable Software: Transitioning to Peer Production: https://journals.aom.org/doi/abs/10.5465/AMBPP.2018.12182abstract
Technology Readiness Level: https://en.wikipedia.org/wiki/Technology_readiness_level

Economics of Open Source

- Open source requires effort, effort requires funding
- Different stages of the open source software lifecycle have different requirements
- Maintenance takes more effort than starting a new idea



Please visit the GitHub repository for full tutorial:

github.com/jupytercon/2020-OpenSourceFunding

Thanks