A path to better science through co-creation and open infrastructure

Tasha Snow[1,2,3], Christopher Holdgraf[4], Wilson Sauthoff[3,5], Jessica Scheick[6,7], Ellianna Abrahams[8,9], Joanna Millstein[3], Sanjay Bhangar[10], Carl Boettiger[11], James Colliander[4], Luis Alberto Lopez Espinosa[12], Elizabeth Holmes[13], Joseph H. Kennedy[14], Julia S. Lowndes[15], Alex I. Mandel[10], Yuvi[4], Fernando Pérez[4,11], Joe-Paul Swinski[2], Andy Teucher[15], Matthew R. Siegfried[3,5]

[1]University of Maryland College Park
[2]NASA Goddard Space Flight Center
[3]Colorado School of Mines
[4]2i2c.org
[5]Colorado School of Mines
[6]University of New Hampshire Institute for the Study of Earth, Oceans, and Space
[7]University of Washington, Department of Earth Science, eScience Institute”
[8]Stanford University
[9]Stanford University
[10]Development Seed
[11]University of California Berkeley
[12]National Snow and Ice Data Center, CIRES, University of Colorado Boulder
[13]NOAA Fisheries, Northwest Fisheries Science Center
[14]Alaska Satellite Facility, University of Alaska Fairbanks
[15]Openscapes

Key Points:

- A co-creation community model can produce geoscience solutions that are more impactful and more efficient than proprietary methods

Corresponding author: Tasha Snow, `tsnow03@umd.edu`

Abstract

Proprietary development solutions are often perceived as being delivered more quickly and easily than open methods, fueling the misconception that they inherently produce better overall outcomes. However, we argue that open development practices can lead to both efficient and maximally impactful outcomes and that one successful strategy for accelerating open approaches in the geosciences is co-creation. This collaborative flywheel dynamic – pioneered in the geosciences by the Pangeo project – encourages teamwork, standardizes access to shared infrastructure, and facilitates iterations of stakeholder feedback and development to accelerate solution finding.

Plain Language Summary

Proprietary development solutions are often perceived as being delivered more quickly and easily than open methods, fueling the misconception that they inherently produce better overall outcomes. However, we argue that open development practices can lead to both efficient and maximally impactful outcomes and that one successful strategy for accelerating open approaches in the geosciences is co-creation. This collaborative flywheel dynamic – pioneered in the geosciences by the Pangeo project – encourages teamwork, standardizes access to shared infrastructure, and facilitates iterations of stakeholder feedback and development to accelerate solution finding.

During the Palisades Fire, researchers were unable to access critical real-time fire mapping for over 12 hours preventing them from tracking the fire's spread to better warn communities in the potential path Pagán (2025). This was not due to a lack of technology or data, but rather because access was locked behind proprietary walls. In contrast, open infrastructure ensures that science and technology reach and serve the broadest possible audience.

Adoption of open science and open-source development principles is growing in science and technology as 35 years of mounting evidence from software communities has shown they lead to better outcomes Free Software Foundation (1989); The Royal Society (2012); UNESCO (2021); Besançon et al. (2021). Open-source development is a collaborative approach to creating software with publicly available code, and open science is the principle and practice of making research processes and products accessible, reproducible, and usable by the public. Disciplines across science are transitioning to more openness to improve research efficiency Donoho (2024), enhance impact Woelfle et al. (2011); Antelman (2004), and ensure that science created with public funding is available to the public.

At the same time, there is tension between the collaborative nature of open approaches against the ambition to build faster and more efficiently, often accomplished via proprietary technology. Proprietary development models reduce the number of voices that must be considered and narrow development approaches Aksulu & Wade (2010). As a result, these solutions are often perceived as delivered more quickly and easily than open methods, fueling the misconception that they inherently produce better overall outcomes Rocklin (2024).

We are a group of scientists, software developers, and engineers that have separately taken open approaches in our research or development. We all sit near or at the intersection of technology and science, with computing projects designed to facilitate research. Some focus on developing tools or analytical methods, others on enhancing research capabilities, and others on training and education. Our communities range from dozens needing a specific tool, to hundreds using data from a single NASA mission, to thousands of students requiring compute power for coursework. Across our respective research and technology collaborations, we have seen the benefits of open practices not only for improving the quality of our solutions, but also our speed of development.

We believe that open development practices can lead to both efficient and maximally impactful outcomes. We argue that a model called *co-creation* is a powerful strategy for accelerating progress through open approaches. This collaborative flywheel dynamic Collins (2009) — pioneered in the geosciences by the Pangeo project — encourages teamwork, standardizes access to shared infrastructure, and facilitates iterative stakeholder feedback and development to drive solutions forward more efficiently. When combined with thoughtful technological vision, co-creation helps open-development and open-science initiatives build better solutions for wider audiences while also keeping pace with proprietary alternatives.

1 From cost savings to community impact: the case for open development

The key strength of open development methods is integrating the micro-expertise and perspective of many, ensuring that development goals serve a broader base of users rather than just those who can pay. As a result, open-development methods often create technology that has many benefits over closed and proprietary methods. For example:

- **Lower cost to use.** Open development usually results in open source technology that can be used for any purpose at no cost Hoffmann et al. (2024).
- **Easier to reach.** Tools are designed to serve the needs of broader audiences, regardless of funding, location, or infrastructure availability.
- **More innovative solutions for a wider variety of problems.** By involving more perspectives in the *direction* of a project, tools solve challenges on broader and different kinds of fronts.
- **More reproducible.** By focusing on accessibility across the development pipeline, collaboratively developed products are often easier to reproduce Barba (2022).
- **More customizable.** Openly developed tools empower users, providing more choice and flexibility than most proprietary tools. This encourages re-mixing and re-use, allowing users to customize existing tools themselves without starting from scratch Medappa & Srivastava (2019).
- **Community empowerment.** By opening pathways for modification and development, users are converted into *collaborators*. They can provide their unique skills and perspectives to a problem in the way that best fits their interests and the project's needs Lerner & Tirole (2005). This often leads to network effects where a project team becomes greater than the sum of its parts Medappa & Srivastava (2019).

2 Co-creation: *How* you build influences what you build

Building on the strengths of open development across many communities we participate in, the most value comes from what we call *co-creation*: the collaborative dynamics enabled by shared infrastructure across communities and the feedback loops we can rapidly cycle through with open infrastructure, end-to-end transparency, and agency over a shared platform. Co-creation requires both having shared access to data, computational environments, and computational infrastructure, and also community investment in their social and collaboration practices, including development of a unique community identity.

In this way, the infrastructure that communities use influences *how and what they create*. By using tools and services that follow the same open development values, shared infrastructure creates its own virtuous cycle with community workflows, enabling communities built on these values to do their work more openly, and thus, *create* scientific and technological outcomes that were more impactful. The history of innovation highlights the critical importance of these connected spaces where a wide range of people naturally encounter and exchange ideas Johnson & S. (2011).

As an example of co-creation, the eScience Institute (the center for data science collaboration at the University of Washington) regularly hosts five-day hackweeks, workshops centered around training participants in the most up-to-date methods for accessing, developing tools, and analyzing scientific data while working in a cooperative environment. Attendees at one such workshop (Earth Science Hackweek) included new and experienced science users and team members from a variety of privately and publicly funded projects that included dataset producers, open-source data science tool developers, and cloud-computing infrastructure engineers. These participants used shared infrastructure and collaborative practices to complete many instances of a core *co-creation feedback loop* for delivering effective and efficient development. This allowed technology developers to:

- Work with scientific users to learn about their workflows;
- Collaboratively identify ways the open-source ecosystem could solve end-user problems more effectively;
- Immediately make progress on one or more open-source projects, with rapid feedback from domain scientists; and
- Deploy prototypes into production for communities to test.

As a result, these co-creation cycles made improvements throughout the open source ecosystem, quickly deployed new technologies into scientists' hands, and accelerated scientific impact Huppenkothen et al. (2018).

3 A feedback loop rooted in co-creation

The feedback loop described above is a common pattern seen across our communities and projects. We generalize the process as a *three-phase co-creation cycle* that repeats itself:

1. **Access.** A shared platform ensures all participants can immediately use the community-tailored tools needed for collaborating together.
2. **Learn.** A strong definition of community practices and values with investment in community skills in facilitating collaboration and learning lays a foundation for growing the co-creation skills of community members. A clear definition of community workflows, enabled by their shared platform, lets new members quickly learn and begin contributing to the science or development collective intelligence.
3. **Create.** By giving community members the *skills* and *tools* they need for co-creation, they can answer modular questions, gain insight, and build tools on their own or in groups. With shared infrastructure, these ideas and improvements quickly permeate the broader community and, in many cases, the larger open-science ecosystem.

A co-creation dynamic can be implemented beyond in-person workshops to persistent workspaces, as demonstrated by the Openscapes Openscapes (2018) and CryoCloud Snow et al. (2023) communities. These long-term collaborative community workspaces foster the three-phase co-creation cycle through shared persistent computational hubs (e.g., a JupyterHub Project Jupyter Contributors (2014)), shared communication platforms with saved histories (e.g., Slack), open training resources for new users (e.g., JupyterBook), and inviting all users and developers from the product pipeline into the community.

Active co-creation occurs when a troubleshooting question generates dozens of responses that result in deploying a bug fix Yuvi (2024a) and sharing the solutions widely Yuvi (2024b) within a few hours. A new scientific community tool can be staged openly on a shared computing environment, instantly accessed and tested by the user base in the same shared computing space; users can readily ask for and receive assistance from

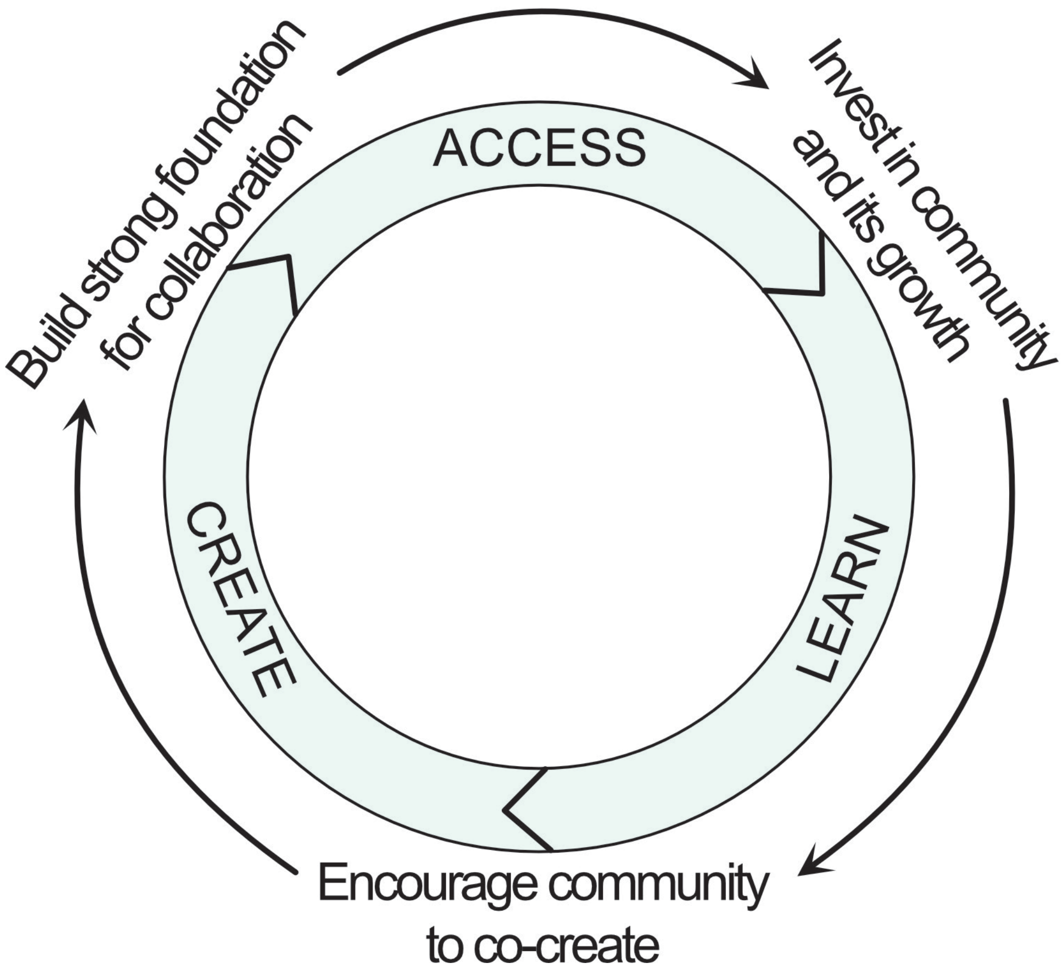

Figure 1. Three-phase co-creation cycle in the inner wheel with the model for replication outside.

the developer when they run into roadblocks, and developers receive immediate, specific feedback on user experience. Development cycles are completed in as little as hours or days *because* of the one streamlined and interconnected set of workspaces.

In co-creation, using shared open technologies and leveraging a broad community of experts for science, development, and user support reduces the burden on project leads to provide all of the specialized expertise and time themselves. This enables the community to not only move faster but also achieve greater impact.

4 Co-Creation in Action: Lessons from the Pangeo Project

The open-source science Pangeo project Pangeo Project (2016) pioneered a powerful co-creation approach for the geosciences by uniting a strong technological and scientific vision with shared community infrastructure, collaborative workflows, and deep ties to open-source communities. Their goal was to build accessible, high-impact geoscience workflows through fully open tools and services. By forming cross-functional international teams, Pangeo created a tight co-creation feedback loop: new challenges receive immediate attention and solutions were quickly shared with the broader community.

Notably, Pangeo leveraged platforms like GitHub and standardized cloud computing environments to enable rapid iteration, ensuring that solutions developed for specific needs also benefit the public. This strategy helped produce or improve widely adopted cross-domain software (e.g., Xarray, Dask, Jupyter, Zarr, hvplot, Kerchunk) and cloud-computing infrastructure (e.g., JupyterHub distribution for Kubernetes), now used worldwide in both industry Holoviz (2025) and academia.

Pangeo's successes demonstrate how co-creation and shared infrastructure foster network effects that elevate the collective scientific enterprise. Communities benefit by using open standards, tools, and services that adapt to specific needs through co-creation, making any improvement immediately usable by all. That is the power of open technology, open science, and services that facilitate the co-creation of value within each.

5 Replicating co-creation in more communities

How do we replicate this co-creation model in practice? Below is a short list of items we believe can serve as a catalyst for co-creation in tool development teams, research teams, and/or funding agencies:

First, **lay a strong foundation for collaboration.** This creates the conditions for communities to connect and learn:

- Begin by designing a strong technological vision around a need that others can coalesce around Schweik & English (2013), then refine it collaboratively with the growing community Shah (2006).
- Disseminate and uphold a welcoming Code of Conduct that promotes a positive work environment Nielsen (2011).
- Create, fund, and maintain a shared, democratic, open communication space that efficiently directs users to relevant conversations (e.g., Slack, Zulip, GitHub)
- Provide a shared computing space with a shared environment (e.g., JupyterHub)

Next, **invest in your community and its growth** to define *modular* community workflows and build upon this foundation with a broad pool of members:

- Invite new and experienced users and developers from all pieces of the toolchain (e.g., infrastructure engineers, tool developers, data producers and users, domain practitioners/scientists)
- Provide structured training and encourage teaching spaces that resonate with *beginner* users
- Build documentation (e.g., JupyterBook, Quarto Executable Books Community (2020); Allaire et al. (2024)) that can guide beginners in contributing to the community and technology development Nielsen (2011)

Finally, **encourage co-creation cycles and sharing:**

- Encourage community members to share intermediate research ideas and products early and often to stimulate collaboration and feedback Nielsen (2011)
- Make upstream contributions to open source communities that are needed to complete your workflows
- Practice and encourage intellectual generosity, sharing, and iterating on ideas across all users
- Uphold your community's commitment to using open infrastructure and open collaboration Invest in Open Infrastructure (2025).

While not exhaustive, these practices have been irreplaceable in our projects. However, we note that co-creation can be hindered by a lack of community safety (if the Code of Conduct is not upheld), a clear beginner entry point, ineffective communication forums, or a community without a shared value system, like the scientific method Nielsen (2011).

6 Towards a global network of co-creation for open science

Pangeo's most innovative contribution to the scientific community has been an open development model guided by a powerful technological and scientific vision. This workflow reminds us that moving intentionally for the sake of allowing others to participate accelerates the creation of resilient, useful solutions by leveraging the micro-expertises and labor of many. In this way, open approaches can outperform proprietary ones.

In our research and development cloud communities, we aim to scale Pangeo's collaborative co-creation model, which is fundamentally rooted in **how we do our work, not just what we build**. Prioritizing open technology, standard workflows, and community-centric infrastructure fosters more meaningful improvements along the way, such as more discoverable and analysis-ready data, more useful and user-friendly tools, and more impactful and reproducible science. Expanding this co-creation model will foster a broader, more resilient network of communities that share knowledge and develop enabling technologies. Ultimately, the greatest impact comes from this interconnected collaboration.

Open Research

The code for this article is available on GitHub. No other data was used.

Acknowledgments
We thank NASA Transform to Open Science program (grant 80NSSC23K0002), and the NASA Cryosphere Program and ICESat-2 Project Science Office (grant 80NSSC22K1877) for funding this work. We thank the development teams, collaborators, and users within each of our communities: Openscapes, NASA OpenScienceLab, 2i2c, CryoCloud, NASA Veda, icepyx, NOAA Fisheries, NASA SlideRule, National Snow and Ice Data Center, Development Seed. The scientific results and conclusions, as well as any views or opin-

ions expressed herein, are those of the authors and do not necessarily reflect those of NASA, NOAA, or the Department of Commerce. We acknowledge ChatGPT o3-mini-high for help in translation from LaTex to MyST syntax.

References

Aksulu, A., & Wade, M. (2010). A Comprehensive Review and Synthesis of Open Source Research. *Journal of the Association for Information Systems*, *11*(11), 576–656. doi: 10.17705/1jais.00245

Allaire, J., Teague, C., Scheidegger, C., Xie, Y., Dervieux, C., & Woodhull, G. (2024, 11). *Quarto.* https://github.com/quarto-dev/quarto-cli. Retrieved from `https://github.com/quarto-dev/quarto-cli` doi: 10.5281/zenodo.5960048

Antelman, K. (2004). Do Open-Access Articles Have a Greater Research Impact? *College Research Libraries*, *65*(5), 372–382. doi: 10.5860/crl.65.5.372

Barba, L. A. (2022). Defining the Role of Open Source Software in Research Reproducibility. *Computer*, *55*(8), 40–48. doi: 10.1109/mc.2022.3177133

Besançon, L., Peiffer-Smadja, N., Segalas, C., Jiang, H., Masuzzo, P., Smout, C., … Leyrat, C. (2021). Open science saves lives: lessons from the COVID-19 pandemic. *BMC Medical Research Methodology*, *21*(1), 117. doi: 10.1186/s12874-021-01304-y

Collins, J. (2009). *Good to Great - Why some companies make the leap and other don't.* SAGE Publications Sage India: New Delhi, India.

Donoho, D. (2024). Data Science at the Singularity. *Harvard Data Science Review*, *6*(1). doi: 10.1162/99608f92.b91339ef

Executable Books Community. (2020). *Jupyter Book (v0.10).* Zenodo. Retrieved from `https://doi.org/10.5281/zenodo.4539666` ([Online; accessed 2025-03-24]) doi: 10.5281/zenodo.4539666

Free Software Foundation. (1989). *Gnu General Public License, Version 1.* https://www.gnu.org/licenses/old-licenses/gpl-1.0.html. Retrieved from `https://www.gnu.org/licenses/old-licenses/gpl-1.0.html`

Hoffmann, M., Nagle, F., & Zhou, Y. (2024). The Value of Open Source Software. *SSRN Electronic Journal.* doi: 10.2139/ssrn.4693148

Holoviz. (2025). *Holoviz Examples Gallery.* https://examples.holoviz.org/gallery/. Retrieved from `https://examples.holoviz.org/gallery/` ([Online; accessed 2025-02-15])

Huppenkothen, D., Arendt, A., Hogg, D. W., Ram, K., VanderPlas, J. T., & Rokem, A. (2018). Hack weeks as a model for data science education and collaboration. *Proceedings of the National Academy of Sciences of the United States of America*, *115*(36), 8872–8877. doi: 10.1073/pnas.1717196115

Invest in Open Infrastructure. (2025). *Statement of support for a global research ecosystem.* https://investinopen.org/blog/statement-of-support-for-a-global-research-ecosystem/. Retrieved from `https://investinopen.org/blog/statement-of-support-for-a-global-research-ecosystem/`

Johnson, & S. (2011). *Where good ideas come from: The natural history of innovation* (1st Riverhead trade pbk. ed. ed.). Riverhead Books.

Lerner, J., & Tirole, J. (2005). The Economics of Technology Sharing: Open Source and Beyond. *Journal of Economic Perspectives*, *19*(2), 99–120. doi: 10.1257/0895330054048678

Medappa, P. K., & Srivastava, S. C. (2019). Does Superposition Influence the Success of FLOSS Projects? An Examination of Open-Source Software Development by Organizations and Individuals. *Information Systems Research*, *30*(3), 764–786. doi: 10.1287/isre.2018.0829

Nielsen, M. (2011). *Reinventing Discovery, The New Era of Networked Science.* Princeton University Press. doi: 10.1515/9781400839452

Openscapes. (2018). *Openscapes.* https://openscapes.org/. Retrieved from https://openscapes.org/ ([Online; accessed 2025-03-24])

Pagán, B. R. (2025). *When our community burned where was the satellite information?* https://www.linkedin.com/pulse/when-our-community-burned-where-satellite-information-pag Retrieved from https://www.linkedin.com/pulse/when-our-community-burned-where-satellite-information-pag%5C%25C3%5C%25A1n-phd-8rxwf ([Online; accessed 2025-03-18])

Pangeo Project. (2016). *pangeo.io.* https://github.com/pangeo-data/pangeo.io. Retrieved from https://github.com/pangeo-data/pangeo.io ([Online; accessed 2025-03-24])

Project Jupyter Contributors. (2014). *Jupyterhub.* https://jupyterhub.readthedocs.io. Retrieved from https://jupyterhub.readthedocs.io ([Online; accessed 2025-03-24])

Rocklin, M. (2024). *What does Pangeo 2.0 look like?* https://matthewrocklin.com/pangeo-2.0.html. Retrieved from https://matthewrocklin.com/pangeo-2.0.html

Schweik, C. M., & English, R. (2013). Preliminary steps toward a general theory of internet-based collective-action in digital information commons: Findings from a study of open source software projects. *International Journal of the Commons*, *7*(2), 234–254. doi: 10.18352/bmgn-lchr.397

Shah, S. K. (2006). Motivation, Governance, and the Viability of Hybrid Forms in Open Source Software Development. *Management Science*, *52*(7), 1000–1014. doi: 10.1287/mnsc.1060.0553

Snow, T., Millstein, J., Scheick, J., Sauthoff, W., Leong, W. J., Colliander, J., … Siegfried, M. (2023). *Cryocloud JupyterBook.* Zenodo. Retrieved from https://zenodo.org/record/7576601 doi: 10.5281/ZENODO.7576601

The Royal Society. (2012). *Science as an open enterprise* (techreport). https://royalsociety.org/topics-policy/projects/science-public-enterprise/report/. Retrieved from https://royalsociety.org/topics-policy/projects/science-public-enterprise/report/

UNESCO. (2021). Unesco Recommendation on Open Science. *UNESCO Digital Library.* doi: 10.54677/mnmh8546

Woelfle, M., Olliaro, P., & Todd, M. H. (2011). Open science is a research accelerator. *Nature Chemistry*, *3*(10), 745–748. doi: 10.1038/nchem.1149

Yuvi. (2024a, 8). *cryo: Enable larger /dev/shm for GPU users.* https://github.com/2i2c-org/infrastructure/pull/4564. Retrieved from https://github.com/2i2c-org/infrastructure/pull/4564 ([Online; accessed 2025-03-24])

Yuvi. (2024b, 8). *Enable higher limits for /dev/shm for all users.* https://github.com/2i2c-org/infrastructure/issues/4563. Retrieved from https://github.com/2i2c-org/infrastructure/issues/4563 ([Online; accessed 2025-03-24])