



Sustaining Open Source Software in the Research Enterprise

Findings from a One-Day Workshop

Chelsea McCracken
Dylan Ruediger
Mark McBride
Patrick Masson
Josh Baron

March 25, 2026



Through research, collaboration, and innovation, Ithaka S+R informs policy, guides strategies, and builds evidence-based tools to broaden access to postsecondary education, improve student and workforce outcomes, and advance scholarship.

Ithaka S+R is part of ITHAKA, a nonprofit with a mission to improve access to knowledge and education for people around the world. We believe education is key to the wellbeing of individuals and society, and we work to make it more effective and affordable.

Copyright 2026 ITHAKA. This work is licensed under a Creative Commons Attribution Non-commercial 4.0 International License. To view a copy of the license, please see <https://creativecommons.org/licenses/by-nc/4.0/>.

The Apereo Foundation is a global nonprofit consortium that advances open source software to support teaching, learning, and research in higher education. Formed in 2012 through the merger of the Sakai and Jasig foundations, bringing together more than 25 years of combined experience in open source educational technology, Apereo provides governance, community coordination, and incubation support for a portfolio of open source projects.

Through open governance and community-driven innovation, the Foundation fosters a collaborative ecosystem grounded in shared stewardship of the academic digital commons. With deep expertise in open source sustainability and cross-sector engagement, Apereo enables institutions worldwide to retain greater control over their digital environments while advancing interoperability, resilience, and the public good.



This material is based upon work supported by the National Science Foundation under Grant No. 2512157. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

This material is based upon work supported by Alfred P. Sloan Foundation under Grant No. G-2025-25223.

This material is based upon work supported by a gift from the Chan Zuckerberg Initiative.



Table of contents

Introduction	3
Executive summary	6
Workshop activities	8
Defining sustainability in open source research software	10
Motivational sustainability	11
Goals of the project	11
Goals of the people	13
Infrastructural sustainability	16
Awareness of dependencies	16
Resource allocation	18
Archival sustainability	21
Relational sustainability	24
Summary of challenges	27
Challenge 1: What is the value proposition for universities? Why should they prioritize OSRS?	28
Mission alignment	29
Reputational benefits	30
Challenge 2: How can we know who is or could be using an OSRS product?	32
Technical solutions for direct tracking	33
Market research	34
System-level mapping	35
Challenge 3: How can we overcome researchers' cultural barriers to open culture?	37
Rethinking collaboration	38
Credit for software	40
Community norms	41
Challenge 4: Who should be responsible for sustaining OSRS?	42
Project level	43
Institutional level	44
Research enterprise level	47
Challenge 5: What extra-institutional support is needed to sustain OSRS?	

	50
Leadership	50
Discoverability	53
Policy recommendations	55
For funders	55
For institutions	55
Glossary	57
Appendix 1	66
About the workshop organizers	66
Highlights of recent work	67
Ithaka S+R's recent work on open science	67
The Apereo Foundation's recent work on open source in higher education	67
Appendix 2	68
Sustaining Open Source Software in the Research Enterprise: Call for participants	68
Participant list	71
Appendix 3	74
Workshop agenda	74

Introduction

Open source software (OSS) is increasingly recognized as a core component of open science.¹ Higher education has served as the seedbed for highly successful OSS such as Apache, Linux, R, Python, Jupyter, and Scalar.² Universities have also played a significant role in advancing OSS for their own use, including for teaching and learning and administration, partly in response to concerns about the cost of higher education and the movement towards open academic and administrative systems. The most visible and largest-scale products to result from these investments include learning management platforms like Open edX, Sakai, and Moodle, as well as administrative platforms such as Kuali and OpenCampus.³

While the open source software used for scholarly research (referred to throughout this report as “open source research software,” or OSRS) often shares with these products an origin in higher education, it has significant features that differentiate it from these other contexts. For example, researchers who develop OSRS are often more focused on the scholarship they are pursuing than on the software they develop to facilitate that pursuit. Indeed, in an environment that incentivizes peer-reviewed journal articles over other research outputs, open source research software is often regarded as little more than a means to a specific end.

Sustainability is a major challenge for even the most successful open

¹ Wilhelm Hasselbring et al., “Open Source Research Software,” *Computer* 53, no. 8 (2020): 84-88, <https://doi.org/10.1109/MC.2020.2998235>; Ian Sullivan, Alexander DeHaven, and David Mellor, “Open and Reproducible Research on Open Science Framework,” *Current Protocols* 18, no. 1 (2019): e32, <https://doi.org/10.1002/cpet.32>; Lorena A. Barba, “Defining the Role of Open Source Software in Research Reproducibility,” *Computer* 55, no. 8 (2022): 40-48, <https://doi.org/10.1109/MC.2022.3177133>; Jaqueline J. Brito et al.,

“Recommendations to Enhance Rigor and Reproducibility in Biomedical Research,” *GigaScience* 9, no. 6 (2020): 1-6, <https://doi.org/10.1093/gigascience/giaa056>.

² Throughout the text, please see the Glossary for more information about open source organizations, software, and technical terminology.

³ Guohua Pan and Curtis J. Bonk, “The Emergence of Open-Source Software in North America,” *The International Review of Research in Open and Distributed Learning* 8, no. 3 (2007), <https://doi.org/10.19173/irrodl.v8i3.496>; David Wiley, “Open Source, Openness, and Higher Education,” *Innovate: Journal of Online Education* 3, no. 1 (October 2006), <https://www.learntechlib.org/p/104321/>.

source software. OSS with any meaningful user base typically requires ongoing community engagement to improve and maintain code.⁴ OSS sustainability also includes identifying a viable model for funding or institutional support; project governance (e.g., decision-making on product roadmaps); technology infrastructure (e.g., code repositories, bug-tracking software, etc.); business operations (e.g., accounting, invoicing, etc.); and navigating legal and licensing issues. Indeed, estimates suggest that many academic open source software projects are abandoned within their first few years because of limited understanding of sustainability needs and models.⁵ Open source research software often faces further challenges, such as a reliance on grant funding and the devaluation of open source development and maintenance in the academic incentive structure.⁶ Dispersed across specialized research communities, OSRS in particular has yet to secure a firm institutional footprint, and its “maintenance, continued growth, and improvement historically has been deprioritized by institutions, publishers, and funders as a less important byproduct of the research enterprise.”⁷ Opportunities to lower the barriers to open source

⁴ Jonathan Corbet, “The Challenge of Maintaining Curl,” *LWN*, Eklektix, August 29, 2025, <https://lwn.net/Articles/1034966/>.

⁵ Jiayuan Zhou et al., “Studying Donations and their Expenses in Open Source Projects: A Case Study of GitHub Projects Collecting Donations through Open Collectives,” *Empirical Software Engineering* 27, no. 24 (2022), <https://doi.org/10.1007/s10664-021-10060-y>; Rajdeep Kaur and Kuljit Kaur Chahal, “Exploring Factors Affecting Developer Abandonment of Open Source Software Projects,” *Journal of Software: Evolution and Process* 34, no. 9 (2022), <https://doi.org/10.1002/smr.2484>; Adem Ait Fonollà, “Survival Rate of GitHub Projects - An Empirical Study,” *Liveable Software*, Jordi Cabot, April 25, 2022, <https://livablesoftware.com/survival-rate-github-projects-empirical/>; R. Stuart Geiger, Dorothy Howard, and Lilly Irani, “The Labor of Maintaining and Scaling Free and Open-Source Software Projects,” *Proceedings of the ACM on Human-Computer Interaction* 5, no. CSCW1, article number 175 (2021): 1-28, <https://doi.org/10.1145/3449249>; Addi Malviya Thakur et al., “Scientific Open-Source Software Is Less Likely to Become Abandoned Than One Might Think! Lessons from Curating a Catalog of Maintained Scientific Software,” *Proceedings of the ACM on Software Engineering* 2, no. FSE, article number FSE099 (2025): 2216 - 2239, <https://doi.org/10.1145/3729369>.

⁶ Danielle Robinson and Joe Hand, “Sustainability in Research-Driven Open Source Software,” *Code for Science & Society*, last updated October 7, 2022, <https://www.codeforsociety.org/resources/report-sustainability-in-research-driven-open-source-software>.

⁷ Carly Strasser et al., “Ten Simple Rules for Funding Scientific Open Source Software,” *PLOS Computational Biology* 18, no. 11 (2022): e1010627, <https://doi.org/10.1371/journal.pcbi.1010627>; Adam Siepel, “Challenges in Funding and Developing Genomic Software: Roots and Remedies,” *Genome Biology* 20, no. 147 (2019), <https://doi.org/10.1186/s13059-019-1763-7>; Julian Nowogrodzki, “How to

adoption and increase the likelihood of long-term OSRS sustainability could be created by developing robust, easy-to-follow guidance, specific to the OSRS context, designed to help researchers identify key questions to ask, practical steps to take, existing institutional resources and networks to leverage, and health metrics to adopt along the path to software development and implementation.

OSRS sustainability will not be achieved quickly. Through the concerted efforts of many stakeholders in the research enterprise and the development of and investment in supportive infrastructure, it has taken over 10 years to make data sharing an established practice, and over 20 years to reach a point where Open Access publication is widely accepted. A comparable level of investment is likely necessary to normalize code sharing and OSRS sustainability.

“Sustaining Open Source Software in the Research Enterprise” (SOSSRE), a one-day in-person workshop made possible with generous funding from the National Science Foundation,⁸ Alfred P. Sloan Foundation,⁹ and a gift from Chan Zuckerberg Initiative, was designed to accomplish the following goals in support of bolstering the ecosystem of open source research software and developing holistic pathways for sustaining it within higher education:

1. Define sustainability in the context of open source research software and articulate unique sustainability challenges and resources in that context.
2. Identify potential methods for sustaining OSRS within higher education, determine their feasibility, and prioritize them for implementation.
3. Strengthen a sense of community among people who use and/or develop OSRS and catalyze relationships between this group and people in other OSS communities.

For details about the workshop organizers, see Appendix 1. For information about the participants, see Appendix 2. This report primarily focuses on OSRS in the higher education context, although it also has relevance to OSRS in other parts of the research enterprise including

Support Open-Source Software and Stay Sane,” *Nature* 571, no. 7763 (July 1, 2019): 133-135, <https://www.nature.com/articles/d41586-019-02046-0>.

⁸ National Science Foundation grant no. 2512157

⁹ Alfred P. Sloan Foundation grant no. G-2025-25223

national laboratories and other government labs as well as industry research labs.

Executive summary

The SOSSRE workshop took place on August 8, 2025. The following report provides a detailed account of the workshop activities and findings about the sustainability of open source research software.

Several key challenges and opportunities emerged from workshop discussions:

- **Define a value proposition:** Until a clear and meaningful value proposition for OSRS sustainability is defined and widely accepted, it will be difficult to achieve.
- **Recognize software as scholarship:** Existing incentive structures for researchers do not adequately acknowledge the scholarly value of open source research software. Sustainability requires a movement to reform incentives to recognize software as a research artifact and to transform existing citation practices and requirements throughout the research enterprise.
- **Embrace student workforce development:** Students have much to contribute to the open source workforce given their willingness to do routine tasks in service of their own learning. Students also benefit from contributing to OSS, gaining critical workforce skills and experience solving real-world problems. This alignment between open source goals and the teaching and learning mission of universities creates opportunities for sustainability, both in terms of strengthening OSRS communities and in terms of convincing campus partners to support open source projects.
- **Coordinate campus-wide OSS activity:** A central point for institutional OSS coordination is integral to OSRS sustainability. Whether that is an administrator, an Open Source Program Office (OSPO), a Tech Transfer Office (TTO), or the library, it is valuable to have someone at an institution whose job it is to advocate for open source trainings, develop organizational open source policies (e.g., licensing, RTP), and serve as a knowledgeable link between

individual researchers developing OSS and the wider OSS ecosystem.

- **Break down academic silos:** OSRS sustainability depends on integrating the distinct perspectives of the researcher community and the open source community and creating more opportunities for them to work together. Researchers looking to sustain OSRS should look beyond their institution to the wider OSS ecosystem and form collaborative relationships with outside entities.¹⁰
- **Fund OSS as infrastructure:** Perpetual grant funding is not a viable sustainability solution for most OSRS. The types of funding systems used to finance buildings and utilities may be a better approach. Many potential models could work, including government budget line items, endowments, new funding models for grants (e.g., the FAIR model),¹¹ institutional membership dues, stock options, and user donation subscriptions. While not all these ideas are feasible in the short term, the common underlying factor is the provision of a continuous, reliable funding stream that OSS projects could rely upon.
- **Build software catalogs:** A key challenge for OSRS sustainability is discovery. A system for categorizing and tagging software—likely developed by an entity outside the academy—would improve many sustainability challenges, although it is not clear what model would be most effective. Proposed ideas include: software archives for the purpose of reproducibility and transparency; software maps of usage and dependencies in order to calculate impact factor; and software "nutrition labels" to encourage adoption and trust.
- **Develop sector-wide standards:** Sustainability differs depending on the goals of the project: while some OSS should be actively maintained indefinitely, others should be documented and archived. The OSS community would benefit from collectively

¹⁰ Molly Galvin, “Now Is the Time to Imagine the Research Enterprise We’ll Need for the Future,” *National Academy of Sciences*, October 11, 2022, <https://www.nationalacademies.org/news/now-is-the-time-to-imagine-the-research-enterprise-well-need-for-the-future>.

¹¹ Joint Associations Group on Indirect Costs (JAG), “The Financial Accountability in Research (FAIR) Model,” The Association of American Universities (AAU), September 30, 2025, <https://www.aau.edu/key-issues/financial-accountability-research-fair-model>.

developing a set of standards whereby researchers can determine which of these two categories their project belongs to at a given time. Funder mandates, which have been essential to other open science cultural shifts, would help disseminate standards throughout the research enterprise.

This is one of two reports on the SOSSRE workshop. A companion report, published by the Apereo Foundation, will include a practical guide and recommendations for researchers.

Workshop activities

The one-day SOSSRE workshop comprised four sessions. The two morning sessions were aimed at establishing a shared vision of the problems to be solved, while the two afternoon sessions were designed to generate solutions to the problems.

The morning sessions began with a plenary roundtable discussion to begin articulating the success factors common to all OSS. Plenary participants included:

- Patrick Masson, Executive Director, The Apereo Foundation (moderator)
- Cat Allman, VP, Open Source, Digital Science
- Dr. Karmen Condic-Jurkic, Executive Director, Open Molecular Software Foundation
- Clare Dillon, Community Lead, CURIOS
- Dr. Allison Randal, Senior Researcher, Capabilities Limited

The panelists described the definitions, metrics, and working models of sustainability in their sector(s). Then, participants were divided into five assigned groups of eight people each for the first breakout session, where they discussed sustainability factors relevant to their experience and identified five critical challenges specific to open source research software:

- Challenge 1: What is the value proposition for universities? Why should they prioritize OSRS?

- Challenge 2: How can we know who is or could be using an OSRS product?
- Challenge 3: How can we overcome researchers' cultural barriers to open culture?
- Challenge 4: Who should be responsible for sustaining OSRS?
- Challenge 5: What extra-institutional support is needed to sustain OSRS?

The afternoon sessions began with a breakout into five groups, each responsible for solving one of the five challenges. Participants selected the group they wanted to work on and completed a Think-Pair-Share exercise to generate a mock grant proposal around a solution to the table's challenge. After solving the first challenge in depth, participants completed a lightning round, during which they were encouraged to quickly move to each of the other tables in turn and generate solutions to the remaining four challenges. At the end of the day, each table's moderator summarized for the entire group the solutions to their challenges discussed during the afternoon sessions. The full workshop agenda is presented in Appendix 3.

The thought-partnership of the workshop continued in the post-workshop survey, where participants were asked to describe the solutions or ideas presented at the workshop that they found most promising and that they would be most interested in developing. Workshop participants were also given access to a conference website on the open source XWiki platform, where 12 people chose to contribute to a discussion board with sustainability advice and useful references.¹² The outcomes from the discussion board, along with the survey results, have been incorporated

¹² Laurie Gemmill Arp and Megan Forbes, "It Takes a Village: Open Source Software Sustainability: A Guidebook for Programs Serving Cultural and Scientific Heritage," *LYRISIS*, 2018, <https://itav.lyrasis.org/guidebook/>; "Academic Open Source Workshop at CHAOSSCon Europe 2025," *CURIOSS*, February 27, 2025, <https://curioss.org/news/2025-02-27-academic-open-source-workshop-chaoscon/>; Karl Fogel, *Producing Open Source Software: How to Run a Successful Free Software Project*, 2nd ed (O'Reilly Media, 2020), <https://producingoss.com/>; Kate Hertweck, Carly Strasser, and Dario Taraborelli, "Insights and Impact from Five Cycles of Essential Open Source Software for Science," Chan Zuckerberg Initiative, 2024, <https://doi.org/10.5281/zenodo.11201216>; Richard Littauer et al., "10 Quick Tips for Making Your Software Outlive Your Job," arXiv, 2025, <https://arxiv.org/abs/2505.06484>; John Meluso et al., "Opening Up: Interdisciplinary Guidance for Managing Open Ecosystems," SSRN, May 8, 2024, <http://dx.doi.org/10.2139/ssrn.4821969>.

into the discussion below. Additional commentary has been added to contextualize participants' comments.

Defining sustainability in open source research software

In the SOSSRE workshop's morning sessions, participants examined the concept of sustainability and its relevance to OSRS with an eye toward establishing common ground and a shared understanding of the problems to be solved.

The concept of sustainability was borrowed from the environmental sciences. It was not originally a software term, as plenary speakers explained, yet it provides a helpful model for understanding OSS communities. Several participants throughout the day accordingly used botanical analogies to define OSS sustainability. For example, one participant compared OSS to a plant that needs time and resources to grow. In another example, participants compared OSS to a rainforest, a finite resource that will be depleted if we consume it without contributing to it. However, a limitation with this analogy is its faulty assumption that all OSS *should* be sustained (see "Motivational sustainability" below). Perhaps a better analogy proposed by participants is that OSS is a garden, a space maintained by people for a purpose. Other workshop participants referenced past efforts to define sustainability, which emphasized the persistence of software over time alongside its ability to meet new needs and fulfill its intent.¹³

Throughout the morning session, participant discussions revealed four interdependent dimensions that together define OSRS sustainability:

¹³ Daniel S. Katz, "Defining Software Sustainability," *Daniel S. Katz's blog*, September 13, 2016, <https://danielskatzblog.wordpress.com/2016/09/13/defining-software-sustainability/>; Colin C. Venters et al., "Software Sustainability: The Modern Tower of Babel," in Vol. 1216, *CEUR Workshop Proceedings* (CEUR, 2014), 7-12, <http://ceur-ws.org/Vol-1216/paper2.pdf>.

- **Motivational:** Sustainability is defined by the purpose of the software and the individual incentives of the people working on it.
- **Infrastructural:** Sustainability is defined by the extent to which software is considered to be infrastructure, a communal resource to which other communal resources flow.
- **Archival:** Sustainability is defined by preservation in the scholarly record and is conceptualized as similar to cataloguing and documenting the provenance of historical artifacts.
- **Relational:** Sustainability is defined by the strength and resilience of relationships between people and depends on the leadership and culture of communities.

Each of these four approaches to defining sustainability surfaced different challenges and opportunities, as we describe below.

Motivational sustainability

Goals of the project

One of the most important prerequisites for defining sustainability in OSRS is identifying why you are sustaining the software in the first place. Sustainability only has meaning in the context of the purpose or goal of the software itself. Participants noted that sustainability can be measured by the difference between what a piece of software is and what the community of users and developers would like it to be.¹⁴

In some cases, the purpose of the software may be defined by the number of users: plenary speaker Cat Allman memorably described purpose-built software for a single person as “more of a Q-tip than a hammer—you don’t reuse Q-tips.” Someone who creates OSRS to solve a specific research problem may have no interest in sustaining it beyond that use. Some projects may never be large enough to sustain a community or business model and may have only been open-sourced due to funder requirements, in which case the software should be sustained by being archived or forked. However, software that can be used by others or repurposed for other functions should be designed with these other uses in mind in order to be sustainable.

¹⁴ “Open Source Archetypes: A Framework for Purposeful Open Source (version 2.0),” Open Tech Strategies and Mozilla, November 2019, <https://opentechstrategies.com/archetypes-files/open-source-archetypes-v2.pdf>.

The purpose of the software can also be defined by the type of users. OSRS projects with a small individual user base and little potential for expansion may operate below the notice of their home institutions. However, for OSRS with wider applications, the potential for institutional adoption brings larger questions about institutional policy to the fore. Institutional adoption of OSRS projects occurs at three levels. At the enterprise level, OSRS can be installed and maintained by central IT, available to anyone on campus. While this is uncommon, Harvard's Dataverse and some other project management tools and data repositories provide examples. At the integrated level, OSRS can be adopted and supported by an individual campus department or unit, such as Dartmouth's Geospatial Learning Hub which supports several open source GIS applications.¹⁵ At this level, OSRS may be installed on campus machines in a computer lab or be maintained by a departmental IT tech who handles troubleshooting; some degree of institutional sanction of the software is necessary to receive install permissions and allocate resources for maintenance. Finally, at the individual level, some university personnel are writing their own code. Institutions may maintain standards and requirements for software created on campus that delineate everything from copyright and licensing to how code is posted to GitHub. For example, MIT's SuperCloud HPC includes instructions for installing R and Python.¹⁶ Allowing the creation of OSRS that conforms to institutional standards can be considered a tacit form of institutional adoption. For these varying levels of institutional adoption, sustainability depends not only on the software developers and the individual users, but also on the institutional procurement practices of the adopters.

The purpose of the software can also be defined by its real-world impacts on the research enterprise. That is, what sustainability means for a given software project depends on whether the software is 1) commercially viable; 2) not commercially viable but socially valuable in terms of its impact (e.g., "if five people make software that cures cancer, that's a big impact"); or 3) neither commercially nor socially valuable, but valuable in

¹⁵ "Accessing Software," Geospatial Learning Hub, Dartmouth College Department of Geography, accessed December 17, 2025, <https://sites.dartmouth.edu/gis-geography/software/>.

¹⁶ "Software and Package Management," MIT SuperCloud and Lincoln Laboratory Supercomputing Center, March 26, 2025, <https://mit-supercloud.github.io/supercloud-docs/software-packages/>.

terms of scientific transparency. Software’s social impact affects how important it is that it be actively maintained, how many resources should be allocated to it, etc. One challenge to sustainability is that many higher ed administrators are unaware of the social impacts of the open source research software being created and maintained at their institutions.

The purpose of the software can also be defined by its adaptability. Some OSS tools are foundational, and there’s a desire for them to never change; for example, “no one should touch” the COBOL code. But with other software, there is a need for constant maintenance to keep pace with scientific advances, add new features, and respond to the user community’s needs. Sustainability means something different in each of these cases. Sustainability can also shift quickly when new security issues arise.

Thus, sustainability has a different meaning for research software across various axes, such as the number and type of users, the real-world impact, and the code’s adaptability. Participants again drew on an environmental model of sustainability to note that, according to evolutionary principles, it may be inevitable that some software dies or forks. However, it may be valuable to develop criteria to determine which OSRS should be maintained and which should not. One participant suggested that the Open Source Program Offices (OSPOs) that have opened at several universities could play a role in helping make these judgment calls.

Goals of the people

Sustainability can also be defined by the motivations of the people working on it. Historically, many OSS workers have been motivated by “fun” and “profit;” that is, intrinsic and extrinsic motivations, respectively.¹⁷

The earliest developers were volunteers who viewed their work on OSS as addressing a personal need, and many people who work on OSS today still view it in this way. Some OSS workers are happy to give their labor away for free because they enjoy the process and the products. However, while this is a strength, it is also a weakness when it comes to sustaining open source research software: much of the critical maintenance work that

¹⁷ Christopher Tozzi, *For Fun and Profit: A History of the Free and Open Source Software Revolution* (Cambridge: MIT Press, 2017); Georg von Krogh et al., “Carrots and Rainbows: Motivation and Social Practice in Open Source Software Development,” *MIS Quarterly* 36, no. 2 (2012): 649–676, <https://doi.org/10.2307/41703471>.

needs to be done to sustain OSRS is “toil,”¹⁸ or “grunt work,” as one of the participants put it, and there is a tension between what users need and what the community of contributors want to spend their free time working on. This gap, between the work that must be done and the fact that no one will do it for fun or profit, is a major sustainability challenge in OSRS.

Some researchers today are motivated to work on research software out of a desire to monetize it, creating a disconnect between what might be most profitable and what the project needs. However, the desire to earn money from labor on OSRS can help sustain motivation. The rise of the research software engineer (RSE) career path is one major area where the financial motive intersects with OSRS. RSEs are highly specialized software developers who are also scholarly researchers, often with a PhD in their research domain, and who may or may not have a passion or interest in open source principles.¹⁹ RSEs are paid to develop research software, but one participant characterized them as expensive and hard to retain. Though RSEs’ expertise makes them a vital part of sustaining open source research software, some participants also spoke about RSEs’ responsibility to recognize the other, important non-development labor that makes a project successful and to mentor others with less advanced coding literacy.²⁰ One participant stated that it would be useful to understand how the day-to-day working lives and needs of RSEs differ from those of people who work in the commercial tech industry, proposing this as an area for further study.

This brings us to the other personal motivations of OSS workers, beyond “fun” and “profit,” that might sustain open source research software. Most OSRS is developed and used in the context of academia, and academia has a built-in teaching and learning aspect. Students are often willing to

¹⁸ Betsy Beyer, Chris Jones, Jennifer Petoff, and Niall Richard Murphy, *Site Reliability Engineering: How Google Runs Production Systems* (O’Reilly, 2016), <https://sre.google/sre-book/table-of-contents/>.

¹⁹ Florian Goth et al., “Foundational Competencies and Responsibilities of a Research Software Engineer: Current State and Suggestions for Future Directions,” *F1000Research* 13 (2025): 1429, <https://doi.org/10.12688/f1000research.157778.2>; Robert Baxter et al., “The Research Software Engineer,” paper presented at *Digital Research 2012*, Oxford, United Kingdom, September 10-12, 2012, <https://www.research.ed.ac.uk/en/publications/the-research-software-engineer/>; The United States Research Software Engineer Association (see Glossary).

²⁰ John Meluso et al., “Invisible Labor in Open Source Software Ecosystems,” *Proceedings of the ACM on Human-Computer Interaction* 9, no. 7 (2025): 1-32. <https://doi.org/10.1145/3757417>.

do the “grunt work” that no one else wants to do if it can serve as a learning experience or an apprenticeship of sorts, an onboarding pathway to other open source work that they would rather do in the future. One participant noted that at their institution, “we have an army of students who are glad to work” on open source research software. Including open source maintenance and contributions as part of coursework, independent studies, or work study programs could help sustain projects while also providing students with workforce skills.

Traditional academic incentive structures can motivate faculty and staff in higher ed to work on open source research software. For example, faculty creation of or contribution to OSRS can be considered scholarship or service and thus count towards retention, tenure, and promotion, graduate fellowships can be awarded to open source contributors, and staff performance evaluations can factor in open source contributions. Some institutions have already made strides towards incorporating data and software into traditional incentive structures; for example, the University of Maryland considers “software applications or other media products” to meet the criteria for modified tenure.²¹ However, most institutions do not currently incentivize researchers to produce long-term outputs like OSRS, instead encouraging serial publications and treating OSRS as an add-on to scholarship rather than as a research output in its own right. This creates a challenge for OSRS sustainability since researchers have limited time and prefer to spend it on tasks that will advance their careers, earn them institutional recognition, and allow them to keep their jobs. One participant also pointed out that the dichotomy between “fun” and “profit” as motivations for researchers misses the goal of utility. OSRS tools are built to support bleeding edge research where the tools researchers need don’t already exist.

An aspect of OSS sustainability that takes on unique features in the academic research context is the impact of novelty on individual motivation. As one participant put it, “is the continuous cycle of new people [in academia] an opportunity or a challenge?” On the one hand, novelty is an opportunity. Participants noted that it is human nature to find

²¹ Iratxe Puebla et al., “Ten Simple Rules for Recognizing Data and Software Contributions in Hiring, Promotion, and Tenure,” *PLoS Computational Biology* 20, no. 8 (2024): e1012296, <https://doi.org/10.1371/journal.pcbi.1012296>; “Information for Faculty Administrators: Appointment Considerations, Modified Tenure Criteria and Agreements,” APT Manual & Guidelines, University of Maryland, June 4, 2024. <https://faculty.umd.edu/apt/167>.

new projects in OSRS to be exciting. Novelty motivates people to work and additional motivations are not needed. One participant shared that postdoctoral fellowships at their institution are short and that this poses an exciting and motivating challenge for the postdocs to see what they can accomplish in that short time. Novel OSS projects are also easier to fund. However, the flip side of this is that software operating costs and overhead—or any project to improve what already exists rather than create new features—are “boring” and thus difficult to fund. OSRS developers in academia typically work under PIs whose academic incentives prioritize novelty, which can make these PIs non-ideal custodians of OSRS. This, and the constant churn of new student workers, could make it difficult to establish a sustainable governance model.

Infrastructural sustainability

After motivation, participants’ most salient approach to defining sustainability in open source research software was to consider OSRS as a form of infrastructure.²² In this view, sustaining OSRS—the virtual spaces where research takes place—is understood as equivalent to maintaining physical research spaces such as labs and equipment, buildings, roads, utilities, etc. This dimension of OSRS sustainability focuses heavily on open source software as an ecosystem with complex and unexpected dependencies, on open source funding and business models, and on support systems within and outside institutions.

Awareness of dependencies

A report on US *physical* infrastructure describes it as “taken for granted.”²³ This seems to be an inevitable characterization of all infrastructure, since it is human nature to be incognizant of the proverbial water we swim in, the systems we rely on every day without recognizing the circumstances that allow them to persist. This same lack of awareness

²² Elinor Ostrom, *Governing the Commons: The Evolution of Institutions for Collective Action* (Cambridge: Cambridge University Press, 1990) <https://doi.org/10.1017/CBO9780511807763>; Nadia Asparouhova, “Roads and Bridges: The Unseen Labor Behind Our Digital Infrastructure,” Ford Foundation, 2016, <https://www.fordfoundation.org/learning/library/research-reports/roads-and-bridges-the-unseen-labor-behind-our-digital-infrastructure/>.

²³ “2025 Report Card for America’s Infrastructure,” American Society of Civil Engineers, 2025, <https://infrastructurereportcard.org/wp-content/uploads/2025/03/Full-Report-2025-Natl-IRC-WEB.pdf>.

applies to the view of OSS as infrastructure. Many researchers who are not computer scientists depend on these tools without knowing their origin or their maintenance needs, never anticipating that they could disappear one day. As one SOSSRE participant noted, OSS is similar to a forest where "you're consuming more than you produce. Eventually you'll have no trees left. Software is similar, we're consuming it and not thinking about the fact that someone had to create it, fix the bugs, provide the hosting, pay the workers. It's very easy to use OSS and not think about how someone, somewhere, had to produce it." As with university indirect costs and overhead funding,²⁴ it is not immediately obvious to the untrained how the resources that sustain OSS are used, so their necessity is thrown into doubt. Participants spoke about convincing senior administrators in higher ed to care about open source research software at their institutions by emphasizing that it is being used and developed under their noses, without their knowledge or oversight. Each of these examples emphasizes the importance of awareness (or a lack thereof) in the infrastructural dimension of defining sustainability.

Because obliviousness presents a risk to sustainability, participants advocated for a marketing campaign to draw attention to OSS's centrality among university stakeholders, funders and policymakers, and the general public. "Building more awareness of the importance of OSS in our lives is critical to gaining more support for sustainability (from the public, funders, institutions, etc.)," one participant noted. Participants proposed marketing campaigns such as a Day Without OSS where all OSS was turned off, or a translation of campaigns on rainforest deforestation that would "show" people what the tech debt of sustaining OSS looks like and bring consciousness to the loss that would take place if OSS disappeared.

The complex dependencies within the open source ecosystem make awareness critical. Participants stressed that the value proposition of OSS is its ubiquity: 90 percent of all software is OSS, it is "the underpinning of the entire digital infrastructure."²⁵ Our economy is dependent on OSS in the same way as it is dependent on raw materials. In order to innovate,

²⁴ Laura Spitalniak, "National Science Foundation Faces Lawsuit Over 15% Indirect Research Cap," *Higher Ed Dive*, TechTarget, May 7, 2025, <https://www.highereddive.com/news/national-science-foundation-faces-lawsuit-over-15-indirect-research-cap/747385/>.

²⁵ Jason Perlow, "A Summary of Census II: Open Source Software Application Libraries the World Depends On," *The Linux Foundation*, March 7, 2022, <https://www.linuxfoundation.org/blog/blog/a-summary-of-census-ii-open-source-software-application-libraries-the-world-depends-on>.

you need a foundation to innovate from: OSS is that foundation. In this way OSS serves as a “commons”—a resource used by everyone but which no one is responsible for.²⁶

Participants stressed that it is critical for administrators to understand how dependent the university is on OSS. On the one hand, this dependency can serve as a metric for the maturity and real-world impact of open source research software, showcasing the good work researchers contribute to the community. On the other, if administrators do not know which university operations depend on OSS, they can't ensure operational systems are designed properly, and they may be blindsided by security threats. For example, departmental computer labs with OSRS installed may be at risk if the software is not kept up to date.

One participant proposed a model for OSRS projects to test and reveal their infrastructural dependencies. Just as banks perform “stress tests” to game out the consequences of shifts in the financial system, so these software “challenge events” could be used as a scenario to strengthen OSRS project sustainability. For instance, when presented with a hypothetical crisis such as “your primary developer has a major health crisis and can no longer be involved,” the OSRS community could work to document how they would respond, in the process uncovering sustainability challenges that they could then mitigate.

Resource allocation

Participants noted that, like other infrastructure, OSRS needs a continuous, reliable infusion of funding. Participants referenced the Principles of Open Scholarly Infrastructure—a values-based framework adopted by many open science organizations—and its assertion that one-time funding should not be used for ongoing needs.²⁷ Participants gave several examples of the current state of affairs where OSS does not receive the sustainable funding it needs: funders supporting software development but not maintenance; projects with year-to-year budgets; the

²⁶ Elinor Ostrom, *Governing the Commons: The Evolution of Institutions for Collective Action* (Cambridge: Cambridge University Press, 1990)

<https://doi.org/10.1017/CB09780511807763>.

²⁷ Geoffrey Bilder, Jennifer Lin, and Cameron Neylon, “Principles for Open Scholarly Infrastructure-v1,” *Science in the Open*, February 23, 2015,

<http://dx.doi.org/10.6084/m9.figshare.1314859>.

stress of continuous grant applications; and institutional funding for open source software that excludes OSRS.

In contrast, participants pointed to funding systems for existing infrastructure as models for what they consider to be sustainable funding. For example, campus research infrastructure is often funded with indirect costs received from grants (although this arrangement is currently being reexamined). Universities support some of their operations with interest earned on endowments. Public universities receive government appropriations from tax dollars. Infrastructure outside the academy (for example, some nonprofits) can on occasion be funded sustainably through crowdsourcing or through a subscription model. Participants also referenced funding models that have worked for other OSS: Drupal and WordPress thrive through multi-vendor ecosystems and broad adoption, while GIS software has longevity partly due to state-level investments and public sector support. While none of these models is a perfect fit for funding the majority of OSRS, participants hoped that they could serve as an inspiration to create a sustainable funding stream for OSRS.

Participants gave particular attention to solutions for a constant funding stream that revolved around an office that already exists and is institutionalized at many universities: the Tech Transfer Office (TTO). Given the slow speed and difficulty of change in academia, linking maintenance of OSRS to infrastructure that already exists holds a lot of appeal. However, there were barriers to this idea. TTOs are designed to generate revenue which supports their own operations through software developed at the institution. While the market for some software (e.g., for collaboration or file management) might be able to generate profit and be commercially viable for a private company, the market for research software is usually too small to be profitable; instead, research software generally aims to be self-sustaining, though it still requires initial funding outlays to get to that point. This creates a structural disadvantage for open source projects in comparison with proprietary research outputs: OSRS projects need adoption before investment, but they need investment before adoption. Furthermore, TTOs can be skeptical of OSS;²⁸ one participant described interacting with a TTO that did not see OSS as part

²⁸ Daniel S. Katz, "Open Source Software and University Intellectual Property Policies," *Daniel S. Katz's Blog*, March 4, 2015, <https://danielskatzblog.wordpress.com/2015/03/04/open-source-software-and-university-intellectual-property-policies-2/>.

of their mission. See “Institutional level” below for potential solutions involving TTOs.

Participants’ discussions of how funding should flow to OSRS projects also focused on the question of responsibility for sustaining those projects. When infrastructure is held in common by many people who depend on it—and who have varying levels of awareness of that dependency—whose responsibility is it to keep that software running? One participant noted that this question “gets into capitalism and socialism and worldviews. There’s not one right answer.” It’s clear that OSRS projects need a better safety net or checkpoints for sustainability, but it’s not clear who will provide those things.

Part of the difficulty of determining responsibility for sustaining OSRS is the distributed nature of decision making in higher ed. While companies might declare software to be “mission critical” and enforce this hierarchically, academia’s shared governance model means that researchers work for themselves and that institutions cannot mandate their behavior. Different stakeholders pursuing their own objectives can create a responsibility vacuum: one participant gave an example of a particular IT department which runs a high performance computing (HPC) resource on campus without understanding how faculty members at the institution are using the resource. There is no one in a position of authority above both groups who is responsible for integrating their two perspectives, so there is no one to determine how resources should best be allocated to address all stakeholders’ needs.

The question of responsibility also relates to project ownership. When it is difficult to determine who owns an OSS project (the researchers, the institution, the funder, etc.), it is also difficult to determine who is responsible for sustaining it. Participants asked: “Who pays to maintain datasets, core infrastructure, or software after the initial funding lifecycle ends?” This is a question, not just of where the money comes from, but of determining whose job it is to ensure the project has the resources it needs to continue to exist.

Defining open source research software as infrastructure necessitates raising public awareness about its complex dependencies, creating a reliable and constant funding stream, and assigning responsibility to particular entities to carry out these tasks.

Archival sustainability

The third dimension of defining sustainability that featured prominently in SOSSRE discussions was the lens of the scholarly record. In this view, open source research software is treated as a form of scholarship, and sustainability is primarily accomplished by ensuring that OSRS is cataloged, tracked, and properly attributed. “Discoverability is critical for the reproducibility of science,” participants noted. Just as the Open Science agenda has raised expectations that researchers will publish their findings in open access venues and append data to their publications,²⁹ funders are increasingly requiring that sponsored software development result in open source applications. This momentum towards publishing the code used to realize research findings pertains regardless of whether the code is being actively maintained—even “Q-tip software” or one-time-use code must be made available in the interests of transparency, as other scholars will want to look at, review, and run the code as part of future research. Participants mentioned Software Heritage, an organization whose mission is to create a “Library of Alexandria” for publicly available code, and Invest in Open Infrastructure’s Infra Finder, noting that software can only be repurposed and reused if it can be found. Preserved in this way, OSS can still be sustained even if it is not maintained.

As with data publication, code discoverability is especially important for publicly-funded research. When taxpayers fund its development, it is important that OSS be available as a public good (according to the “Public Money, Public Code” principle), yet participants felt that funder support for maintaining these tools over time should be expanded and universalized. Some federal funding application requirements are purposefully vague as to whether software is a research product; this leaves consideration of OSRS sustainability up to the researcher. One participant described a situation where they relied explicitly on funder requirements for OSRS sustainability to push back against institutional resistance.

Discoverability is also important in terms of trust. For individual researchers who consider adopting an OSRS application, it may be difficult to determine whether the software should be used for a research task: Does the software do what it is supposed to? Is it scientifically up-to-date? Is it being actively maintained for troubleshooting and so that new

²⁹ Chelsea McCracken and Ruby MacDougall, “Researcher Challenges and Experiences with Data Services,” *Ithaka S+R*, March 27, 2025, <https://doi.org/10.18665/sr.322388>.

feature requests can be incorporated? Potential institutional adopters may also have concerns about the trustworthiness of code: Will it be fixed if it breaks? Does it present a security risk to install on campus computers? Will it be around long enough to merit the resource expenditure of maintaining it? Institutional IT procurement procedures typically ensure due-diligence for the acquisition of proprietary software through Requests For Proposals (RFPs). Such assessments determine whether the proprietary software vendor is solvent, how many employees they have, whether they are involved in litigation, etc.³⁰ For OSRS that might be adopted at the enterprise or departmental level, it is possible to adapt these same procurement procedures by applying the same due-diligence: Is the project active according to its current version date, release cycle frequency, etc.? How many contributors does it have? How many bug reports/issues are unresolved? CHAOSS's metrics models could be incorporated into existing procurement processes.³¹ However, central IT staff may not always be familiar with how to do so. One participant explained that at their institution, IT would rather invest in a commercial product than in OSRS because they do not know how to determine the reliability of the code. Other participants gave the cautionary example of Blacklight, an open source library discovery software that initially saw wide adoption, but devolved into chaos due to poor open source management as individual institutions adapted it to their local needs. Some participants called for an independent body that would assess and report on the stability of OSRS, emphasizing the need for more readily available information about OSRS to ensure sustainability.

With traditional forms of scholarship, one method of gauging the trustworthiness of a publication is in the number of subsequent works that cite it (also called “impact factor,” although participants took pains to differentiate this from real-world impact, discussed above in “Goals of the project”). Similarly, some participants argued that metrics of usage, adoption, reach, and maturity could serve as indicators of the trustworthiness of OSRS (and thus sustainability), if only there was some

³⁰ Robert Heckman, “Managing the IT Procurement Process,” in *Enterprise Operations Management Handbook, 2nd ed.*, ed. Steven F. Blanding (New York: Auerbach Publications, 2000), <https://doi.org/10.1201/9781003069386>; Thomas Trappier, “Is There Such a Thing as Free Software? The Pros and Cons of Open-Source Software,” *EDUCAUSE Review*, July 30, 2009, <https://er.educause.edu/articles/2009/7/is-there-such-a-thing-as-free-software-the-pros-and-cons-of-opensource-software>.

³¹ “Metrics and Metrics Models,” CHAOSS, accessed February 17, 2026, <https://chaoss.community/kb-metrics-and-metrics-models/>.

way to track them. Unfortunately, as one participant pointed out, “a trick with OSS is that once it is released, you don’t know who your users are and what the use cases are” because in most cases, open source projects do not ask someone to explain how they plan to use the software before they download it. Open source cultural norms tend to minimize barriers to software reuse. Nonetheless, participants proposed a few potential metrics to track adoption, such as the number of publications or presentations which cite an OSRS tool; the number of industry partners who support an OSRS tool; or the number of contacts the project receives from people interested in OSRS. When one participant commented on the need for a knowledge graph that cataloged software dependencies in OSRS, other participants recommended OpenAlex and Digital Science’s Dimensions tool.

Which bucket of faculty responsibilities does software development fall into—teaching, research, or service, or perhaps professional development?

The matter of tracking the usage and adoption of open source research software (necessary for archival sustainability) is further complicated by a lack of consensus around determining credit, authorship, and ownership. The rotating cast of people doing the work on an open source project doesn’t always match up with the official governance structure, and it may even be difficult to determine which people belong to which group. Then there is the issue of whether OSRS is owned by the researcher(s), the institution, or the funder, and whether they are aware of what ownership means. According to SOSSRE participants, “universities are overly protective of intellectual property, and researchers are afraid of being scooped,” particularly when financial risk is involved. Funders who are knowledgeable about open source issues may be able to pressure institutions’ legal departments to allow software to be open-sourced, but this workaround does not apply in all cases. And finally there is the question of whether the software itself should count as a publication, or if it is instead a tool that *leads* to publications and is cited in them. Which bucket of faculty responsibilities does software development fall into—teaching, research, or service, or perhaps professional development? This thorny set of problems around who has permission to publish OSRS and how it should be cited must be solved before archival sustainability can be achieved.

Relational sustainability

The fourth prominent approach to defining sustainability in open source research software focuses on the management and leadership of the community and group dynamics of the workers on an open source project. Separate from individuals' motivation for participating, this approach focused on the persistence of the group over time and its ability to self-perpetuate, in terms of change management, leadership, governance, communication, and culture. As one participant put it, having “a dedicated passionate group of people” is the most essential feature for ensuring project sustainability.

Planning for the future is an important metric of organizational success: sustainability should be a concern from the inception of a project. Yet few standardized playbooks exist for sustaining OSS projects. Without a widespread, shared body of practice, projects often reinvent the wheel or collapse when leadership or funding ends. One participant stressed the importance of open-sourcing a project from the very beginning;³² postponing the open source release often makes it more difficult to get all the stakeholders on board. Participants described change as a constant in OSS projects and stated that the most sustainable projects are those that build resilience and adaptability into their leaders and developers, citing studies of change management that show that regular small changes are more tolerable than occasional large changes. Teaching the skill of change management (which one participant compared to “therapy”) to community members is necessary in the context of open source research software in order to deal with the “artificially forced timescale of engagement at the four-year university”—which makes knowledge transfer difficult—and to promote innovation. Regular monitoring and review of community goals can identify misalignment and provide a direction for needed changes.

According to some participants, change should be driven by leaders, but leaders can be resistant to change. Meritocracies may not always create leaders with the skills to promote long-term sustainability—if everything must go through one person it can create a bottleneck that strangles the organization. It can be difficult for leaders to let go of control; as one participant put it (referencing the term first used for Guido van Rossum, the creator of Python), “‘benevolent dictators for life’ tend to result in ugly

³² Nadia Asparouhova, *Working in Public: The Making and Maintenance of Open Source Software* (San Francisco: Stripe Press, 2020).

coups,” so it’s necessary to develop succession plans and exit strategies for leaders. For open source research software, different academic roles—e.g., a post-doc who will leave in a few years vs. a tenured researcher—represent different levels of sustainability risk. One participant suggested that developing personas or profiles of OSS leadership styles could help researchers determine where they fit.

Embrace of change is also necessary to ensure that the culture of an OSS community remains open to new contributors. Without regular reflection about their governance model, community members can develop private in-jokes, jargon, and assumptions about others’ knowledge base, making it difficult for newcomers to get involved or for research collaborators to be converted into technical (code) collaborators. Participants emphasized that representation of all stakeholders—developers, users, and institutions—in governance of open source research software was a core concept of democracy and essential for sustainability. Representation feeds adaptability to change because users know what a project needs and developers know what works. The challenge is in making sure that the institution (which likely contributes many of the resources) doesn’t dominate decision making. Participants also spoke about the importance of including non-core developers and non-tech contributors like librarians or archivists. The value of this openness extends to cultivating collaboration over forks, leaving space for people who have left the project to return. Flexible information tools that enable read permissions (transparency), write permissions (inclusion), and administrative permissions (decision rights) can enable the necessary degree of openness to change.

Communication is another tool for relational sustainability that participants considered to be non-negotiable, but the research enterprise tends not to recognize the essential nature of communication-related labor (e.g., organizing, coordinating) to the success of a project as much as it recognizes software development. Communication can mean people skills, standards of propriety exemplified in codes of conduct, or documentation and comments in the code. Participants gave many examples of collaboration challenges in OSRS. For example, a project was supposedly transparent about being open source, but its legal department was unaware, emphasizing the importance of communicating with non-technical partners and external infrastructure. In another example, faculty who were highly invested in their own software projects were

uncomfortable contributing to a central resource and sharing with each other in a highly productive R1 environment.

Communication can refer to both internal and external collaboration, and participants also stressed the importance of establishing an ecosystem of open source research software into which new projects can be socialized. Here, sustainability means taking advantage of existing work, projects, and groups; it means cultivating communities of practice across campus (disciplines, departments) and within and outside institutions; it means leveraging potential partners' skills, resources, and expertise. It means gaining an awareness of users and potential users and connecting them with each other, not just through the core project. It means working with OSPOs (or other parts of the campus) as a neutral resource to help all potential contributors, projects, departments, and disciplines be more collaborative. According to participants, "growing the community is the only way for sustainability to work, so try to find collaborators with a long-term interest in the software."

Sustainability means taking advantage of existing work, projects, and groups; it means cultivating communities of practice across campus (disciplines, departments) and within and outside institutions.

Finally, relational sustainability is defined by culture, which participants mentioned in a variety of contexts. Institutions differ in the extent to which they subscribe to "hacking culture," impacting both user awareness of the capabilities of OSRS and expectations of whether software will be free or proprietary. Institutions also differ in terms of their research focus; at emerging research institutions, faculty may not know what they don't know. Projects differ based on whether they cater more to "software-centric culture" or "end-user-centric culture." The culture of open source in general tends to be less competitive than the culture of research software, where researchers can be hesitant to be open due to the influence of academic culture. Participants noted cultural differences in terms of speed: software adoption happens quickly at the institutional level but slowly in research communities; and in OSRS, both short-term processes (grants, grad students) and long-term processes (PIs with vision) happen on a different timeline from corporate software, where commercial interests drive new and updated products.

Summary of challenges

In discussing these definitions of sustainability and what they mean for open source research software, participants emphasized some challenges more than others. The five most salient challenges were:

- Challenge 1: What is the value proposition for universities? Why should they prioritize OSRS?
- Challenge 2: How can we know who is or could be using an OSRS product?
- Challenge 3: How can we overcome researchers' cultural barriers to open culture?
- Challenge 4: Who should be responsible for sustaining OSRS?
- Challenge 5: What extra-institutional support is needed to sustain OSRS?

In the following sections, we discuss the results of the afternoon session, where participants generated solutions to each challenge.

Challenge 1: What is the value proposition for universities? Why should they prioritize OSRS?

In order for institutions to assume responsibility for sustaining open source research software, they need to agree that they should do so, and then they need to incorporate this belief throughout all levels of the institutional structure. SOSSRE participants spoke about their experiences trying to gain support for OSRS efforts from upper administrators, and some shared sample “elevator pitches” for how to convince university leaders that OSRS is important. One participant stated that working together to create an effective value proposition would be the most effective step participants could take after the workshop, and that the value proposition should encompass not only open source research software but also open source enterprise software such as learning management systems, etc.

However, participants were not able to define a clear and compelling OSRS value proposition for institutions. It is likely that, rather, several value propositions are necessary to speak to the specific interests and motivations of key university decision makers. Funders and other stakeholders hoping to build on the work of SOSSRE should focus their investment on identifying these value propositions.

Participants determined that, once value proposition(s) for OSRS are defined, they need a standardized process for propagating them throughout the institution and throughout the research enterprise as a whole. For example, Offices of Research, OSPOs, and Tech Transfer Offices could play an educational role by embedding open source awareness into compliance and onboarding processes for employees and administrators. Open source advocates could create materials such as playbooks, governance templates, and frameworks to communicate the value of OSRS to policymakers and funders, which institutional leaders

could then use to advocate for OSRS funding at the federal and state levels.

While brainstorming OSRS value propositions, participants identified two main arguments for prioritizing OSRS:

- **Mission alignment:** OSRS is a good fit for universities' teaching & learning and research missions.
- **Reputational benefits:** Institutional investment in OSRS could lead to reputational prestige with the public and other institutions.

Mission alignment

Most universities incorporate teaching and learning into their mission; many also incorporate research leadership; and many incorporate both. Participants argued that universities should prioritize OSRS because it aligns with both their teaching and learning and their research missions.

In teaching and learning, participants argued, engagement with OSS is a high-impact practice that contributes to workforce development, producing graduates better prepared for collaborative, digital-first careers.³³ The OSS community is highly collaborative across geographical and political borders, requiring no official credentials to participate. This creates a low barrier to entry for students at the course and lab level. With a little instructional design assistance, faculty whose courses already include fundamental instruction in coding, computing, and collaboration would be well equipped to build hands-on, active learning experiences with OSRS that strengthen students' skills in these areas. OSRS also provides opportunities for student internships and work-study, applied research, and interdisciplinary practice, and under some circumstances could earn students a line on their CV or a publication. Graduates with open source skills and hands-on experience in foundational computing, coding, and collaboration will have a leg up in a technology-focused job market that favors teamwork and communication.

³³ George Kuh, *High-Impact Educational Practices: What They Are, Who Has Access to Them, and Why They Matter* (Washington, DC: American Association of Colleges and Universities, 2008),

<https://web.archive.org/web/20220208083536/https://www.aacu.org/publication/high-impact-educational-practices-what-they-are-who-has-access-to-them-and-why-they-matter>.

Open source communities also epitomize the forward-looking research currents of open science, interdisciplinary collaboration, and community engagement. OSRS projects offer opportunities for scholars in many disciplines to contribute: in addition to computer scientists and research field-specific experts, these projects have potential roles for business and law school faculty, data librarians, education researchers and social scientists specializing in organizational or social psychology, and designers or cognitive scientists who work on human-computer interaction. Furthermore, by allowing disparate researchers to come together on a single project, OSRS reduces duplication of effort across the research enterprise, saving money and improving productivity. Through its typically asynchronous processes, OSRS enables distributed research communities to work more efficiently and to easily incorporate community members both within and outside the academy. OSRS facilitates the transparency and reproducibility of science, furthering public trust, and ensures that if one researcher or institution drops out, the project can continue.

Reputational benefits

Participants also maintained that reputational benefits could be part of the institutional value proposition for OSRS and that, at some point in the future, prioritizing OSRS could improve universities' standing both with the general public and among their peer institutions. However, a great deal of change is needed before this value proposition becomes persuasive.

In theory, by investing in OSRS, universities invest in the advancement of knowledge for societal benefit. Since OSRS is potentially transparent to the public and usable by, for example, private companies and citizen scientists, supporting OSRS could position universities as innovative contributors to the public good rather than as profit driven in an environment of rising student costs. Continued provision and maintenance of OSRS developed in academia demonstrates that universities are good stewards of the resources they have been allocated. The bottom-up structure of open source communities also provides a way for institutions to counter accusations of elitism. Many OSRS initiatives emerge organically from faculty labs, research centers, or student projects; universities benefit most when they recognize these grassroots efforts and provide legitimacy, infrastructure, and pathways to scale. However, in part

because it is difficult to measure the economic impact of OSS,³⁴ the public does not necessarily recognize its value, which throws the feasibility of this value proposition into doubt.

Participants also imagined that OSRS could serve as an arena where institutions compete with their peers for prestige. If OSRS were adequately recognized within traditional academic incentive structures, universities that prioritized OSRS could better attract faculty and students who valued openness and collaboration. OSRS provides tangible career roadmaps for students and, for early-career researchers, offers opportunities for CV lines, publications, and applied project experience. Faculty research projects involving OSRS are more appealing to some funders (particularly government funders) who increasingly value open, shareable outputs. These grants offer the potential to bring more research dollars into the institution, and faculty who wish to compete for them will gravitate towards institutions that will support them through that process. However, because the current landscape does not yet include a critical mass of researchers who recognize the scholarly value of OSRS, this part of the value proposition is also speculative. Participants suggested that existing studies of universities' return on investment in research competitiveness for high performance computing could provide a blueprint for demonstrating the value of investing in OSRS.³⁵

Crafting one or more convincing value propositions for OSRS will likely require a more concerted effort to identify important university decision makers, understand their goals and pain points, and provide OSRS metrics tailored to their role.

³⁴ Manuel Hoffmann, Frank Nagle, and Yanuo Zhou, "The Value of Open Source Software," *Harvard Business School Working Paper* 24-038 (2024), <https://papers.ssrn.com/sol3/Delivery.cfm?abstractid=4693148>.

³⁵ Amy W. Apon et al., "High Performance Computing Instrumentation and Research Productivity in U.S. Universities," *Journal of Information Technology Impact* 10, no. 2 (2010): 87-98. https://open.clemson.edu/cgi/viewcontent.cgi?article=1001&context=computing_pubs.

Challenge 2: How can we know who is or could be using an OSRS product?

Usage is a powerful metric for understanding a product’s significance and downstream influence on the research enterprise. Usage data—a standardized method of understanding who is using a piece of software and how they are using it—is vital for OSRS project developers and maintainers to know whether, how, and with whom to sustain a project. They may also need this information in order to justify the time and resources they spend on sustaining the project and to realize academic incentives based on their work. In addition, usage data is important for individual researchers or institutions who need to know whether a software application is trustworthy before deciding to adopt it. Finally, usage data is important for administrators, funders, publishers, and others who may be evaluating whether to invest in a researcher or their software. For OSRS, then, the usage metric is comparable to the impact factor of traditionally published scholarship.

The challenge in collecting usage data is that OSS projects are essentially designed to be frictionless at the point of download: while developers can track the number of downloads, there is typically no way for them to directly trace usage beyond that. (Contrast this system with “usage” tracking for traditional scholarship: anyone who “uses” an academic journal article, for instance, must cite it, and it is then possible to generate a list of every “user” of the article by name). One OSS community, for instance, opted against including any type of code that would alert the project when someone installed it, even though having that data would have been helpful to the project. The decision was primarily driven by privacy sensitivities and the values of the larger OSS community.³⁶ Due to conflicts like this one, OSS projects often need to resort to incomplete usage metrics, such as tracking pull requests, in order to estimate usage.

³⁶ “What is Free Software?” GNU Operating System, 2024, <https://www.gnu.org/philosophy/free-sw.html.en>.

Thus, OSRS communities struggle to overcome the paradox of needing to track usage in order to demonstrate an application’s value proposition, while also needing to not track usage in order to follow their values and maintain credibility. SOSSRE participants’ solutions to this challenge fell into three categories:

Direct tracking: Technical methods for measuring how many people are using a piece of software or code.

Market research: Solutions focused on estimating and understanding the usage of or need for particular OSS.

System-level mapping: Tracking OSS usage across the sector through publishers and funders with the use of knowledge graphs.

Technical solutions for direct tracking

Participants recommended a handful of hacks and software in order to count and observe open source code dependencies and users.

Some technical solutions were designed to track OSS usage within a given institution. One participant recommended partnering with the IT department to authorize a sanctioned port scan of all OSS infrastructure in use within an institution. Tools to do this include Nmap and Zenmap, Masscan, Angry IP Scanner, and Unicornscan. Another participant suggested generating a list of all open source software used on campus (a software bill of materials, or SBOM) by utilizing a software composition analysis (SCA) tool like FOSSA.

There are other technical solutions for measuring OSS usage outside a given institution, an environment where tracking is even more difficult (see the Glossary for more information on these solutions). Participants recommended that OSRS projects use their institution’s reverse proxy server—essentially a project “receptionist” that receives and routes external data requests from users, allowing the collection of usage data without personally identifiable information—to monitor users, hits, traffic, etc.³⁷ Some proprietary providers of remote proxy servers (e.g., Scarf) also offer the ability to embed tracking pixels in the software documentation

³⁷ Harry Abram, “What is a Reverse Proxy? Everything You Need to Know,” *Nostra*, June 10, 2025, <https://www.nostra.ai/blogs-collection/reverse-proxy>.

for an OSS project, allowing OSRS teams to collect limited information about the users who download their documentation. External websites could also be scanned with a profiler application (e.g., BuiltWith), which returns a list of all the technologies it can find on the page. These methods could be used to generate lists of off-campus users of open source research software.

Participants also recommended incorporating automated tracking into the software development process. OSRS projects which make their code available for download through GitHub have the ability to use GitHub's internal metrics for tracking users of the code, such as unique visitors, referring sites, forks, and commits, as well as the number of users who have starred a project (i.e. bookmarked it) or who receive notifications about it without being a contributor. Collecting this information automatically each time a project team member makes a contribution to the code would allow for continuous monitoring of the user base.

Market research

In addition to tracking usage by direct means, participants also advocated more indirect mechanisms for understanding a product's current or potential user base through market research or more traditional social science research. These methods, however, tend to be more labor-intensive than direct tracking.

Market research could involve analyzing existing data to determine OSS usage patterns. For example, researchers could analyze existing testimonials and posts on OSS discussion forums or keyword frequency using Google Ads Keyword Planner. Researchers could also scan open source libraries for references to a particular software product or count keyword tokens using a web search on Reddit, Stack Overflow, or a general search engine.

Market research could also involve actively generating new data about the existing or potential OSS userbase. Several participants proposed robust mixed-methods research projects along these lines. For example, one participant took a user experience design approach towards creating open source software products for research, asking themselves, "Did I build the right thing?" They envisioned a design ethnography with interviews to assess user needs and inform the development of prototypes, followed by

a survey to validate findings. Storytelling played an important role in another participant's proposal, which focused on stakeholder discovery through inviting the community to participate in impact storytelling. A third participant took more of a network referral approach to market research: they recommended focusing on the market analysis and the value proposition of a product, discovering potential customers using the networks formed by NSF's I-Corps training program and conducting focus groups to ask about challenges related to using or implementing OSRS that new software could address.

System-level mapping

Some participants' recommendations for studying OSS usage had implications across the entire OSS ecosystem, including for publishers, funders, and other system-level organizations. They often centered around the creation of some kind of database or knowledge graph that would track usage across the sector.

Several participants looked to the scholarly record to measure OSS usage. The system is already set up to track how and by whom scholarship is used and applied, and participants saw it as an obvious step to shift from there to tracking software as well. For example, projects could use the Zenodo /GitHub integration to build in citability and therefore trackability. One participant imagined creating a comprehensive knowledge graph of all software dependencies based on a survey of existing software references and mentions in published scholarship. Another imagined developing clear guidelines for adoption and usage of software citation practices to which publishers would opt-in. These guidelines could also be built into research compliance training around disclosure of resources.³⁸

Participants also proposed using the existing records from funding agencies as a starting point to track OSS usage across the sector. One participant suggested that foundations should create a public database of software identified in applications and proposals. In addition, they could require grant recipients to list all software used and developed during their funded research in their final grant report. Funders could then

³⁸ "Reminders of NIH Policies on Other Support and on Policies Related to Financial Conflicts of Interest and Foreign Components (NOT-OD-19-114)," National Institutes of Health, US Department of Health and Human Services, July 10, 2019, <https://grants.nih.gov/grants/guide/notice-files/NOT-OD-19-114.html>.

analyze this data in terms of institutions, software, and disciplines. This plan is similar to the Internet2 model for identifying and maintaining stakeholders' contacts and activities.

In addition to publishers and funders, there is also space for a new or independent body to create a system-level map of OSS usage. One participant suggested creating a consortium of institutions who undertake to study OSS usage on their campus (using the techniques mentioned in the previous two sections). A knowledge graph of the dependencies could be shared as an institutional repository or index and could serve as a tool for funders, sponsors, and member institutions to highlight the value of their campus projects. Researchers at the member institutions could also use the index to find each other for purposes of collaboration. Similarly, another participant suggested creating an independent body to curate data across institutions about OSS usage as a paid institutional membership service. Data could include adoption, scope, maturity, and total cost of ownership (TCO).

Usage data generated at any level—telemetry, market research, or the scholarly record—will be beneficial in generating a meaningful value proposition for OSRS sustainability.

Challenge 3: How can we overcome researchers' cultural barriers to open culture?

Developing a value proposition for OSRS sustainability will also require significant cultural change. Just as cultural shifts towards open access publication and data sharing throughout the research enterprise are measured in decades of investments and infrastructure development, we can expect a similar timeline for OSRS sustainability. This is, in part, due to a cultural barrier: researchers often have little interest in expending the time and resources required to build and maintain software, since they rarely see it as their core work, as a topic they are passionate about, or as a plausible pathway to prestige or career advancement. Developing supportive infrastructure around OSRS sustainability is essential because we cannot anticipate that researchers will do it on their own.

Despite identifying culture as a barrier, SOSSRE participants were not able to articulate a robust set of solutions, thereby illustrating another major finding of the workshop: the open source community and the research community, writ large, face difficulties in understanding each other's motivations. Software developers' intrinsic and extrinsic motivations do not always map easily onto researchers' motivations around scholarly contributions and academic incentives. This gap is evident in how participants discussed potential approaches to overcoming cultural barriers:

- (A) **Rethinking collaboration:** Researchers and software developers have different challenges in finding partners for collaboration. The need to address the challenges of both groups is paramount.
- (B) **Credit for software:** Researchers try to maximize the value of their time, so incentive structures play a role in influencing their behavior. If their discipline or institution does not reward them for pursuing openness, they may feel their time is better spent elsewhere.

- (C) **Community norms:** Increasing the participation of a wide variety of people leads to more innovation in OSS communities but presents challenges for collaboration. Communities need more effective methods for successfully integrating new researchers.

Rethinking collaboration

Potentially the most effective way to address collaboration challenges is to continue to create opportunities for cultural exchange between researchers and the open source community, so that these groups can better understand each other. Other proposed solutions feature designs for collaborative projects and systems such that the needs of both software developers and researchers are met.

SOSSRE participants identified competition as a common cultural barrier in software development communities, where a fear of perceived “scooping” fuels an attitude of “build your own” software, hindering early collaboration among software developers. Researchers, by contrast, are typically highly aware of and likely to collaborate with colleagues doing work in a similar subject area, subfield, or methodology, but do not typically seek collaborators working on similar software or code. Ithaka S+R’s research suggests that, rather than fear of having their code scooped, researchers are more likely to be hesitant to share their code because they are embarrassed about its poor quality.³⁹ One SOSSRE participant’s solution to the problem of fears around collaboration was inspired by CZI’s Essential Open Source Software for Science (EOSS) initiative, which provides “dedicated funding for maintenance, growth, development, and community engagement” of established OSS. The idea is that funders could establish a program that explicitly funds inter-institutional or cross-disciplinary collaborations with the goal of producing “built to share” OSRS known to not be commercially viable. The inter-institutional or cross-disciplinary aspect could allay the scooping fears of software developers, while a requirement to incorporate research software engineers on project teams could potentially mitigate researchers’ fears about code quality.

³⁹ Dylan Ruediger et al., “Big Data Infrastructure at the Crossroads: Support Needs and Challenges for Universities,” *Ithaka S+R*, December 1, 2021, <https://doi.org/10.18665/sr.316121>.

SOSSRE participants also described a common attitude among software developers—the belief that you can do a better job than the existing software—and how it impedes developers from joining and improving an existing project, impacting sustainability. Researchers, however, typically recognize the limitations of their coding abilities and are happy to use existing software whenever they can. The challenge for researchers is that their needs are bespoke; existing software is unlikely to serve their purposes as-is, and they may not know how to identify code that can be efficiently adapted for their use. Participants suggested that an educational campaign encouraging the modification of existing software might help solve this problem if it were combined with a system-level software catalogue which would allow researchers to easily locate code they could revise to meet their needs.

Participants noted that collaboration poses an especially complex challenge for graduate students who are required to publish something novel in order to receive their degree. In a saturated environment, these graduate students want to establish their own academic identity and make sure that their name is the one cited. For disciplines where code is considered a low-status research output, grad students who develop or maintain software as their main scholarly output—often as part of a PI’s project—may not be able to earn their degree by doing so. Even if they are able to earn a credential for creating software, these early-career researchers are unlikely to rack up citations of their work unless they also produce more traditional types of publications. One potential solution to this problem is the reform of citation practices across the board so that software users would cite the original code even if the developer hasn’t published about it in a traditional venue. Participants referenced the Zenodo / GitHub integration that facilitates this process.⁴⁰ This change, along with reform of dissertation guidelines, would allow OSRS development to be a sustainable path for graduate students to follow, regardless of discipline.

⁴⁰ Martin Fenner, “DOI Registrations for Software,” *DataCite*, May 17, 2018, <https://doi.org/10.5438/1nmy-9902>.

Credit for software

Overcoming cultural barriers requires giving researchers a reason to do so. According to participants, “it’s not easy to be open”—each step towards openness creates more work, and many researchers wonder “why they should care” about putting in that extra effort. Participants recommended that funders could create directives—for instance, requiring OSS sustainability plans as part of grant applications; requiring software to be publicly deposited; and requiring projects to include interviews with users—to force researchers to leave their labs and establish those communities that will sustain projects after the funding runs out. Mandates similar to these have played an important role in normalizing other open science practices, showing that this method can be effective. As directives around open access publishing and data sharing have shown, when many funders coordinate their long-term investments to create infrastructure, eventually these mandates transform into accepted and encouraged parts of research culture.

Participants also described the importance of fitting OSRS into academic incentive structures, including retention, tenure, or promotion (RTP) policies for faculty and other systems for staff, postdocs, students, etc. This would involve robust training and guidance in OSS citation practices for researchers;⁴¹ publisher requirements that all authors of a code release are credited as though it were an academic publication; support to store and publish software releases that are citable; and treating all contributions as publications. The automation of contribution records, potentially through AI, could serve as a metric that could be incorporated into calculations of impact factor. Participants also cited the *Journal of Open Source Software*’s EditorialBot, which automates checking references and repositories for submitted papers, as a useful model for other publications.

⁴¹ Stephan Druskat, “Practical Software Citation for Research Software Developers, Maintainers and Users,” In *HPC Best Practices Webinars*, IDEAS Productivity, June 4, 2025, Video webinar, <https://ideas-productivity.org/events/hpcbp-091-software-citation>; Smith et al., “Software Citation Principles,” *Peer Computer Science* 2 (2016): e86, <https://doi.org/10.7717/peerj-cs.86>; Daniel S. Katz, Neil Chue Hong, and Martin Fenner (Co-Chairs), “FORCE11 Software Citation Implementation Working Group,” GitHub, accessed November 12, 2025, <https://github.com/force11/force11-sciwg?tab=readme-ov-file#group-products>.

However, we can also approach the incentive issue with the opposite strategy: rather than making OSRS fit into the box of traditional incentive structures, we could reform these incentive structures to encompass a wider variety of activities. For example, one participant came up with a plan to reform retention, tenure, and promotion committees through scholarly societies. A joint task force of scholarly societies could develop new recommended guidelines which would incentivize OSRS labor to the same degree as traditional metrics. Scholarly societies could lead trainings for their members on how to implement the new guidelines and could liaise with accreditors to increase adoption. Related efforts are currently underway with the National Academies' HELIOS Open initiative, at the Open Research Community Accelerator (ORCA), and in a framework developed by Erin McKiernan.⁴²

Community norms

For some participants, the main barrier to open science was literally a cultural one: teams with a variety of researchers of different backgrounds and nationalities found it more challenging to enact democratic principles of governance. One participant developed a plan to investigate and document community norms in their OSS community. They would then enlist core members to model and enforce expected behavior and check in periodically to determine whether the code of conduct was adequate based on the demographic makeup of new contributors. Another participant developed a plan for revising their OSS community's existing code of conduct for international participation, incorporating national differences in terms of data sharing, available resources, and IP concerns, while also creating a reporting mechanism. Their proposal also encouraged researchers to contribute to another project in their field, increasing cross-cultural understanding.

Participants named several procedural solutions for successfully moderating OSS communities: using a software development forge site (e.g., GitHub) for communication; being strategic about the organizational information that is preserved (such as codes of conduct); and recording information about governance structures (who has rights to do what),

⁴² Erin C. McKiernan et al., "A Framework for Values-Based Assessment in Promotion, Tenure, and Other Academic Evaluations," (Pre-print, submitted in 2024), <http://dx.doi.org/10.31219/osf.io/s4vc5>.

labor (who does what), purpose (why tasks are important), and compensation (what benefits laborers receive). Participants cited several additional resources for strengthening the inner workings of OSS communities.⁴³

Challenge 4: Who should be responsible for sustaining OSRS?

The notion of responsibility implies that one has certain resources and the ability to allocate those resources. Participants' main concern was around who should have decision-making power around resource allocation, especially about which projects should be sustained and which should not be. As described earlier, sustainability for so-called "Q-tip" projects may consist minimally of licensing, commenting, documenting, and then archiving the code. However, for projects that should be sustained beyond this, participants determined that a roadmap was necessary, and recommended developing standardized processes and criteria for evaluating open source research software, determining whether it should be sustained, and if so, how, and how many resources should be allocated to it.

SOSSRE participants observed that responsibility for sustaining open source research software exists at all levels. The community of users who rely on it must have some responsibility, as must the researchers who create it and the research institutions where they work. Those who fund OSRS and those who most benefit from it—that is, all of society—also bear responsibility. Both bottom-up and top-down responsibility is important to sustainability, and the responsibility should be distributed across a variety of organizations and individuals to avoid the possibility of a single point of

⁴³ Daniel Coyle, *The Culture Code: The Secrets of Highly Successful Groups* (New York: Bantam, 2018); Georg Link, "Metrics for Maintainers with Sarina, Feanil, and Felipe," August 7, 2025, in *CHAOSScast*, produced by the CHAOSS Project, <https://podcast.chaoss.community/116>; and the Organization and Relationship Systems Coaching (ORSC), <https://crrglobal.com/about/orsc>.

failure. While several of these layers of responsibility are well-established, others are more aspirational; in particular, participants emphasized that responsibility at the institutional, funder, and societal levels is essential to sustainability, but is not yet established. One participant proposed a study to map stakeholder interests and capabilities which would be conducted by interviewing people at all levels about their incentives for and the ways they contribute to sustaining OSRS. Such an examination could reveal pain points within the open source ecosystem that prevent stakeholders from taking responsibility.

Participants developed solutions to the question of who bears responsibility at three main levels of magnification:

- (A) **Project level:** Responsibility lies with individual researchers and users to sustain the work they contribute to.
- (B) **Institutional level:** Responsibility lies among institutional leaders and institution-level offices, especially OSPOs and/or TTOs.
- (C) **Research enterprise level:** Responsibility lies outside the academy in the government and private sectors, among funders and NGOs, and in academia-adjacent networks such as scholarly societies, membership organizations, and discipline-specific consortia.

Project level

At the most granular level, participants noted that university faculty, staff, and OSS workers (including those who are students) bear responsibility for sustaining open source research software.

Faculty and staff are often the researchers in question who develop OSRS, so they naturally should be responsible for the software they develop. Faculty and staff are also frequent users of software developed by others, and thus have a responsibility to help maintain it as well. However, participants stressed that without institutional recognition and support for their work, faculty and staff are often forced to deprioritize maintenance. Tapping into existing incentive structures, or reforming them to recognize OSS labor as scholarship, is necessary to allow faculty to take up this responsibility. For staff, annual performance evaluations should also be reformed to factor in contributions to OSS.

The sustainability of open source research software can also benefit from student labor. One participant proposed that OSS sustainability could be achieved through a guild-like model, where tasks are assigned to volunteers who have access to mentors for purposes of professional development. This model describes the structure of a university, where students' reason for enrolling is to learn and to gain real-world workforce skills. Another participant suggested taking a systems approach to leverage student labor—for example, computer science students could be engaged to help maintain code as part of their coursework, marketing students could help market OSRS products, and art students could help with software user experience design. Independent study, internships, work-study programs, and fellowships are existing models that would compensate students for their labor with money or credit alongside the work experience they would gain. Students could be supervised by departmental faculty or staff, or work for other administrative units such as OSPOs, TTOs, libraries, and IT. One participant voiced hesitation that maintaining code was too complex of a task to ask of students, but as long as students are not asked to shoulder the responsibility alone, without a mentor, there should be scaffolded tasks they are capable of doing as part of the learning process. OSPOs have so far been successful at supervising student workers.⁴⁴

Institutional level

Within an individual institution, participants attributed responsibility for sustaining open source research software to upper administrators, OSPOs and TTOs, and libraries. One participant considered establishing a campus organization or group capable of sustaining OSS to be the most actionable solution they heard during the workshop. While institutional and system-level OSPOs are still in the experimental phase, some participants saw the OSPO model as promising in this regard, a sentiment that Ithaka S+R's report suggests has merit.⁴⁵

Reiterating the need for a simultaneous bottom-up / top-down approach to OSS sustainability, a few participants recommended that institutions

⁴⁴ Dylan Ruediger et al., "University Open Source Program Offices," *Ithaka S+R*, August 14, 2025, <https://doi.org/10.18665/sr.323392>.

⁴⁵ Dylan Ruediger et al., "University Open Source Program Offices," *Ithaka S+R*, August 14, 2025, <https://doi.org/10.18665/sr.323392>.

appoint a “cabinet-level” open source director to drive institutional open source policy. While this level of seniority may be unnecessary, nonetheless assigning an existing high-ranking position (e.g., senior research officer) to include open source promotion as part of its scope could go a long way towards convincing administrators to value OSRS. A person in this role could also funnel information about OSRS accomplishments on campus to university marketing; good PR might make the institution more interested in sustaining and funding the work. One participant suggested that a senior open source leader could conduct a system assessment across campus information systems (such as learning management software, student information systems, etc.) and present the results to campus stakeholders in order to provide a foundation for developing institutional policy around open source adoption and maintenance.

Institution-level offices also bear responsibility for sustaining open source research software. Currently, Tech Transfer Offices (TTOs) sustain themselves and their work through revenue that is generated from the sale and licensing of technology developed at the university.⁴⁶ OSS revenue models could also be deployed to generate revenue from OSS developed at the university.⁴⁷ TTOs could use this revenue to fund staff time towards OSS promotion, both internally and externally, which would in turn increase revenue. For example, staff could work to maintain projects, driving more adoption, and could encourage private industry adopters to contribute back to the projects. Several participants suggested merging OSPOs with TTOs, although no clear roadmaps or goals for doing so emerged from the workshop.

⁴⁶ National Research Council, *Managing University Intellectual Property in the Public Interest* (Washington, DC: The National Academies Press, 2011), <https://doi.org/10.17226/13001>; “Technology Transfer Evolution: Driving Economic Prosperity,” Association of Public & Land-Grant Universities, 2017, Report of the Technology Transfer Evolution Working Group of APLU’s Commission on Innovation, Competitiveness & Economic Prosperity (CICEP), <https://www.aplu.org/wp-content/uploads/technology-transfer-evolution-driving-economic-prosperity.pdf>; Farah Gerdes, Jacki Lin, and Sophearay Smith, “University Technology Transfer: Best Practices Guide,” Wilson Sonsini Goodrich & Rosati, Advamed Accel, AUTM, 2019, <https://www.advamed.org/wp-content/uploads/2019/02/advamed-best-practices-guide-041719.pdf>.

⁴⁷ Chitu Okoli and An Nguyen, “Business Models for Free and Open Source Software,” SSRN, 2016, <http://dx.doi.org/10.2139/ssrn.2568185>.

Regardless of which institution-level office does the work, participants stressed the need for an office that provides strategic planning assistance on open source projects. This office would provide details on existing OSRS to encourage contributions to current projects and code reuse rather than launching new projects; would help researchers determine whether existing projects should be sustained and if so, what resources are available to do so; and would work to sustain projects that met certain criteria, for example by employing students as maintainers and engaging outside parties to contribute. This office could also serve as a clearinghouse for governance playbooks, licensing guidance, and contributor agreements, and could develop other resources targeted at particular campus audiences, such as roadmaps to sustainability for different types of OSRS projects.

Participants also described the important role of the university library in OSRS sustainability. Librarians often serve as the first point of contact for researchers, a gateway connecting them with governance and preservation practices. Further, as libraries already typically house and maintain institutional repositories for data and publications, they could take on the additional responsibility of holding and indexing an archive of retired OSRS projects with their licenses and documentation, and of developing resources on and instructing researchers how to transition a project into the archive. One participant envisioned a model whereby each institution operated its own Git repository for code (Git is the open source system behind GitHub) which was integrated with its document repository, facilitating the safeguarding of knowledge within the academy rather than on proprietary platforms like GitHub. Another participant created an alternative plan for how libraries could fit into a robust open source ecosystem on campus. Here, the library would serve as a gateway to connect researchers with the OSPO/TTO, a provider of institutional support for actively used OSRS infrastructure; and an institutional home for student workers. They noted that undergraduates can currently secure work-study in libraries; they should be able to get work-study to maintain code too. They also noted that graduate students or post-docs could be paid to help archive code.

Institutions can also help sustain open source research software by embedding open source literacy at every level. Sustainability ultimately depends on people, not just money or code, and projects thrive when they diversify their contributor base and train and mentor graduate students

and early-career researchers.⁴⁸ Institutions could normalize open source practices early to build a pipeline of contributors and maintainers, incorporating open source literacy into orientation programs for new faculty, graduate student onboarding, and training for research administrators.

Research enterprise level

SOSSRE participants spent the most time discussing solutions for sustaining open source research software at the research enterprise level; this was the level they felt most needed to be expanded. They attributed responsibility for sustaining OSRS to four main extra-institutional entities: the federal government; nonprofit foundations; private industry; and centralized networks across academia. They believed that this high-level support for OSRS will be essential to any sustainable model; however, most of their ideas in this regard are highly aspirational with no clear roadmap to achieving them. In effect, the lack of societal support for such a model is a major structural barrier to OSRS sustainability without a clear solution, thus placing a hard limit on the steps towards OSRS sustainability that are potentially achievable by POSE and related programs.

With the reasoning that society benefits most from open source research software and should therefore be responsible for sustaining it, participants posited that public infrastructure should be paid for by public tax dollars. Several participants speculated that Germany's Sovereign Tech Agency could be a good model for the US. This agency, which funds open source software in the name of national security, is a subsidiary of a cabinet-level ministry whose funding is appropriated by the German legislature. Germany's Sovereign Tech Agency recently inspired OpenForum Europe to call for a similar EU-wide initiative.⁴⁹ Replicating this

⁴⁸ Qingye Jiang et al., "Diversity, Productivity, and Growth of Open Source Developer Communities," arXiv, 2018, <https://doi.org/10.48550/arXiv.1809.03725>; Open Source Way contributors, "Building Diverse Open Source Communities by Making Them Inclusive First," in *The Open Source Way* (version 2.2), 2024. <https://guidebook.theopensourceway.org/attracting-users/building-diverse-open-source-communities-by-making-them-inclusive-first>.

⁴⁹ Nicholas Gates, "OFE Publishes Landmark Study Calling on Funding Europe's Open Digital Infrastructure through an EU Sovereign Tech Fund (EU-STF)," OpenForum Europe,

model in the US today would certainly pose significant challenges, including clearing the hurdle of congressional action. In the past, Congress has set a precedent with the use of federal funds for necessary services that are difficult to fund in other ways (e.g., the postal service, public utilities). SOSSRE participants imagined convincing the public of the need for establishing such an agency through a marketing campaign that would demonstrate the benefits of OSS, though they did not have a clear plan for how such a campaign would work.

Participants reasoned that anyone providing initial funding to science research should also be responsible for providing long-term maintenance of the work produced with that funding. Similarly, organizations that rely on software for their operations on a day-to-day basis should be responsible for making sure that software continues to exist. Several participants recommended an endowment model, where funders (potentially including federal, nonprofit, and industry) would place a percentage of the monies they award into an endowment that would then provide a long-term revenue stream for OSS sustainability. A related proposal was to establish a national non-profit to which all OSS stakeholders would contribute financially, creating a common pool of money to fund ongoing maintenance for open source research software. While the idea of a central authority likely would not appeal to the open source community, participants at the Open Source Congress a month later in September 2025 were thinking along the same lines, favoring a collaborative peer network “across foundations, academia, industry, and governments [to] build resilience, share expertise, and ensure that critical OSS infrastructure is maintained,” including through “developing usable platforms for global contributors, supporting multilingual ways to share knowledge, and fostering developer ecosystems in emerging markets.”⁵⁰

Participants did not believe responsibility was limited to any one industry sector. While the commercial open source world demonstrates the possibility of sustainability through community-driven business ecosystems, even commercially-backed projects struggle with scalability,

July 23, 2025, <https://openforumeurope.org/ofe-launches-landmark-study-calling-for-an-eu-sovereign-tech-fund-to-secure-europes-digital-future/>.

⁵⁰ Eclipse Foundation AISBL and contributors, “2025 Open Source Congress Report,” report on the third Open Source Congress, September 16-17, 2025, Brussels, Belgium, December 3, 2025, <https://outreach.eclipse.foundation/hubfs/2025%20Open%20Source%20Congress%20report.pdf>.

suggesting higher education must combine commercial models with public and institutional investment. On the other hand, foundations support only a small percentage of OSS projects. Foundations with broad research missions (not limited to open source) could serve as neutral hosts for open source research software, and affiliate programs (industry partnerships, consortial membership models) may provide recurring revenue and shared advocacy platforms.

In terms of selecting which projects to fund, participants favored the FOSS Fund model, where private companies (e.g., Microsoft) establish FOSS Funds which provide sponsorships to OSS projects voted on by their employees. Participants suggested that the systems discussed above for providing funding could be driven by votes from researchers on which projects to support.

Sustaining OSS at the research enterprise level requires organizations to take responsibility for providing labor as well as funding. Several participants suggested using new or existing networks of expertise to support OSS projects. For example, each institution could have an OSPO or access to a similar service (perhaps through system-level OSPOs, with which the Sloan Foundation is experimenting in the University of California and University of Texas systems).⁵¹ Some type of centralized process for seeking support from research software engineers would also be valuable; participants pointed to Internet2, a high-performance computing network, as a model. Participants also indicated a belief that journals, publishers, professional societies, and other extra-institutional stakeholders across all disciplines in the higher ed space have a responsibility to increase requirements for the citation of software which supports publications and to develop recommendations for RTP standards that acknowledge OSRS.

⁵¹ Faith Singer, "UT System Advances Open Source Leadership with New Program Office," Texas Advanced Computing Center, The University of Texas at Austin, October 16, 2025, <https://tacc.utexas.edu/news/latest-news/2025/10/16/ut-system-advances-open-source-leadership-with-new-program-office/>.

Challenge 5: What extra-institutional support is needed to sustain OSRS?

SOSSRE participants observed that entities at the research enterprise level have a role to play in open source research software sustainability. Their proposed solutions fell into two categories:

- **Leadership:** External entities could provide guidance, resources, training, and funding to OSRS projects.
- **Discoverability:** External entities could create systems for indexing the OSRS ecosystem, calculating impact factor, and depositing code.

Leadership

SOSSRE participants expressed that the type of extra-institutional support most needed is leadership across several areas: providing guidance and consultation; creating resources and tools; delivering training and education; and designing funding systems.

Participants were most likely to recommend some kind of consulting service that would supply guidance, expertise, and labor for a given project by request. Participants proposed quite a few different ideas about what service would be provided, who it would be provided by, and how the consultation would work. One idea was a matchmaking service that could pair OSRS projects with foundations or partner organizations (e.g., Code for Science and Society, Lyris) to provide administration and/or funding. Foundation-as-a-service is a potential support model for OSRS, but small OSRS projects don't have the infrastructure or knowledge to manage such relationships. A matchmaking service could help bridge the gap between them. Similarly, a nonprofit foundation specifically focused on research software could assist OSRS projects with business development or with finding a fiscal sponsor (a nonprofit that handles administration on their behalf). Corporations could also partner with campus OSRS, incentivized

by the good public relations they would earn and the potentially increased use of their products. In the discipline of engineering, project incubators (often university-based, such as the innovation incubators at UC Berkeley),⁵² product commercialization models, and the principle of productization create roadmaps for project development. Participants wondered what it would look like to replicate this support structure for open source research software.

Designing a system to supply the services of expert developers to OSS projects in need was another common theme. Participants proposed: a matchmaking service—perhaps operated through professional societies like IEEE—that would pair interested volunteer developers with projects in need of maintenance; a code review service (e.g., Princeton’s Repo Review Consultations); and an OSPO-as-a-service that would hire and then lend out research software engineers. Such a system already exists at some large tech companies; for example, software engineers from Google helped develop an OSS product at the University of Illinois at Urbana-Champaign, including by bringing in volunteers for sprints, and Amazon Web Services hosts a similar service where they donate developer labor with the hope that the resulting project ends up in the AWS ecosystem.

Participants also recommended that functional tools be developed by extra-institutional organizations in support of the whole research enterprise. For example, to address the need for a system to identify project maturity level, a set of standards and a classification system could be developed which included some form of “nutrition facts label” or badging for OSS projects. Templates for memoranda of understanding (MOUs) between OSS projects and partners for funding or data exchange are another potential tool. A third tool could be a new version control system specific to university research offices that could be spun up and sold to universities; this would replace GitHub’s version control system which conflicts with the needs of OSRS due to its incompatibility with binary data.

Informational resources could also be created to assist researchers. These resources could contain answers to frequently asked questions that developers of OSRS might have: How can researchers design an OSRS

⁵² UC Regents, “Incubators and Accelerators,” University of California, Berkeley, Office of Innovation and Entrepreneurship, accessed November 7, 2025, <https://iande.berkeley.edu/resources/incubators-and-accelerators>.

budget and an administrative plan? Who should researchers talk to at their institution (e.g., general counsel, TTO)? How can researchers who receive a grant hire open source workers on short notice? How can researchers understand and translate across different types of open source projects? What are the variables for success of an OSRS ecosystem in higher ed? Participants cited Lyrasis' It Takes A Village Toolkit which answers some of these questions.

Beyond these resources, participants envisioned training and education supplied by extra-institutional entities. Several referenced NSF's I-Corps model; one suggested that an "I-Corps for OSS" could be created to provide OSS workforce development at institutions which lack the resources to do so on their own; another participant recommended working with institutional chapters of student groups such as the National Student Data Corps (which "teaches data science fundamentals to students") to instruct students in OSS principles. Institutions have existing corporate relationships and alumni who could be recruiters within an I-Corps model. Relatedly, one participant held up the Data Curation Network (DCN) model and Europe's ENHANCE network, which "offers innovative educational programmes and initiatives that foster inter-institutional collaboration." Any such organization delivering education on OSS could offer a microcredential which students and researchers could document on their CV. One participant also kept the educational needs of non-academics in mind; they noted that open source support vendors could use training on how to be a palatable service provider to universities.

Participants also voiced a belief that extra-institutional support should come in the form of funder leadership. As mentioned above, one model for this involved fiscal sponsorship or assistance by a private foundation established for that purpose. Other forms of support included making the current system of grant-funded OSRS more effective in terms of sustainability. For one thing, funders could ask researchers to build business development into their grant budgets, allocating time and money to create business plans and build community. Alternatively, funders could allocate grant funding to turn an OSRS product into software as a service, where users donate on a subscription basis to support the product through micro payments. Participants pointed out that it can be difficult to set up a system for individual donations to OSS, and recommended a fiscal sponsorship model where the university serves as the fiscal sponsor. Another participant called this model "Patreon for OSRS."

Discoverability

Discoverability of open source research software is another major challenge that participants maintained could be best solved by extra-institutional support for indexing, impact, and repositories.

Participants expressed that a catalog or index of open source research software, created outside the academy, could be highly beneficial. They explained how researchers' decisions to make their projects open source occur at different points in the development process, and many decide to do so after the project is complete—similar to how many researchers first think about data sharing at the end of a project.⁵³ Participants stressed the need for a database that would help researchers identify whether their software idea is actually novel, preventing them from reinventing the wheel. The tool could also include some kind of “nutrition facts label” or badge describing the status of existing software and providing information on who is using it, allowing the collection of large-scale data on open projects and relationships between projects. Participants posited that a nonprofit foundation for research software could create and maintain such a tool, and noted that Hugging Face's platform that allows users to share datasets fulfills some of this role already, as does Invest in Open Infrastructure.

Participants also pointed to the need for extra-institutional support with measuring the impact of open source research software. As they explained, it can be very difficult for some OSRS projects to assess their impact relative to other projects due to computational complexity or high data needs. In order to calculate an impact factor—and appeal to the vice provost for research's office—researchers might need to perform an analysis (connectivity or network) of the larger OSS ecosystem, for which they would need some kind of knowledge graph. Participants referenced Jonathan Starr's Map of Open Source Science as an example.

In addition, participants expressed an interest in establishing a “GitHub for research software,” a repository where research software could be tied to data, which would seem more natural to researchers and would contribute to the reproducibility of science. GitHub's limitations on the size of file uploads, project discoverability, etc., reflect that its business model

⁵³ Chelsea McCracken and Ruby MacDougall, “Researcher Challenges and Experiences with Data Services,” *Ithaka S+R*, March 27, 2025, <https://doi.org/10.18665/sr.322388>.

and use cases are not well-aligned with researchers' needs.⁵⁴ Participants voiced a desire to form a software development community that was more research-focused—they noted that Hugging Face's research segment has made steps towards this goal. These sentiments reflect a perception by some researchers that GitHub monopolizes coding: though it would be possible for an institution to create its own comparable research-focused repository using Git (the open source system behind GitHub), GitHub's status as the largest source code host creates a network effect which makes it difficult for researchers to go elsewhere. Participants also discussed the possibility of a code-and-data collaboration forge site optimized for the needs of research software engineers, a conversation that has progressed further since the workshop.

⁵⁴ GitHub, "About Git Large File Storage," accessed February 18, 2026, <https://docs.github.com/en/repositories/working-with-files/managing-large-files/about-git-large-file-storage>.

Policy recommendations

In this report, we summarize recommendations for funders and institutions that emerged from the SOSSRE workshop. See the companion report, published by the Apereo Foundation, for recommendations for researchers.

For funders

- Encourage or require applicants to submit detailed, robust sustainability plans as part of funding proposals and to include budgets and timelines for the development of business models and plans for future sustainability throughout the project life cycle.
- Encourage or allow applicants to use grant funding to hire research software engineers, community managers, UX or market researchers, and other personnel and activities that are common to many sustainable OSS initiatives.
- Consider creating dedicated funding opportunities specifically targeted at improving the sustainability of existing OSRS with high-impact and/or high-use potential.
- Integrate OSS into existing citation practices and open science requirements such as open access publication or data deposit.

For institutions

- Encourage cross-unit collaboration to leverage existing OSS expertise from across the university and designate clear responsibility for managing this collaboration to specific individuals, units, or bodies. Develop systems to funnel researchers to OSS resources and expertise.
- Encourage departments to develop standards to evaluate OSS outputs and activities in retention, tenure, and promotion decisions. Direct relevant university offices to integrate the

evaluation of OSS outputs into staff promotion guidelines.
Encourage university marketing to promote these research activities.

- Use onboarding, orientations, and similar opportunities to promote OSS values and practices among staff, faculty, and students.
- Tap into existing funding systems for institutional operations (e.g., state appropriations, private endowments, etc.) and existing OSS networks (e.g., institutional consortia, foundations, private industry) to find support for OSRS projects.

Glossary

Amazon Web Services (<https://aws.amazon.com/>): A cloud computing platform where users pay for computing power and storage. Many online applications rely on AWS's cloud infrastructure.

Angry IP Scanner (<https://angryip.org>): An open source port scanner.

Apache Web Server (<https://www.apache.org/>): An open source web server maintained by the nonprofit Apache Software Foundation (ASF). Originally released in 1995, Apache Web Server played a key role in the development of the World Wide Web.

API (application programming interface): Set of definitions, protocols, and rules associated with a software application that allows it to communicate with another software application.

Blacklight (<https://projectblacklight.org/>): An open source software first released in 2009, used by libraries and museums for cataloging and repositories.

BuiltWith (<https://builtwith.com/>): Technology profiling software which identifies the software platforms and services that a website uses.

Central Authentication Service (CAS) (<https://www.apereo.org/programs/software/cas>): An open source identity management software application used for single sign-on (SSO), created by Yale University and now supported through the Apereo Foundation.

CHAOSS (<https://chaoss.community/>): A project of the nonprofit Linux Foundation focused on creating metrics to evaluate the status of open source software.

COBOL (Common Business-Oriented Language): A programming language first developed in 1960 by the US Department of Defense, used today in many business, finance, and administrative systems for companies and governments.

Code for Science and Society (<https://www.codeforsociety.org/>): A nonprofit that promotes open source technology and public access to information to improve people's lives.

commit: A change to a codebase that is recorded in the version control history, similar to a saved file.

customer relationship management (CRM): A system for managing data on customers and interactions with them in order to personalize customer communications.

Data Curation Network (DCN) (<https://datacuration.network/>): A membership organization of institutional and nonprofit data repositories which promotes open research.

DevOps: The practice of integrating software development (making changes to code) with IT operations (implementing the changes).

Dimensions (<https://www.digital-science.com/product/dimensions/>): A searchable database of research funds created by Digital Science, a UK-based technology company, in 2016.

Drupal (<https://new.drupal.org/>): An open source web content management system (CMS) first released in 2001 which plays an important role on the back end of many high-traffic websites today.

EditorialBot

(https://joss.readthedocs.io/en/latest/editorial_bot.html): An open source bot developed by the Journal of Open Source Software in order to enable open peer review in a collaborative, iterative format.

ENHANCE (<https://enhanceuniversity.eu/>): An alliance of ten technology-focused institutions in Europe that offers collaborative inter-institutional educational programs.

Essential Open Source Software for Science

(<https://chan Zuckerberg.com/eoss/>): CZI's EOSS initiative supports software maintenance, growth, development, and community engagement for critical open source tools.

fiscal sponsorship: A model whereby a registered nonprofit organization adopts a mission-aligned open source software

project and uses its legal status as an incorporated entity to offer financial and business services.

FORCE11 (<https://force11.org/>): A free membership organization for technologists seeking to transform scholarly communication. Their Software Citation Implementation Working Group is currently working to implement the software citation principles they developed.

forge site: A collaborative online software development platform, such as GitHub, SourceForge, GitLab, Bitbucket, etc.

fork: A copy of an existing codebase that is then developed independently from the original.

FOSS Fund (<https://fossfunders.com/>): Fund established by a private company to provide sponsorships to OSS projects voted on by the company's employees.

FOSSA (<https://fossa.com/>): Software application used for software composition analysis (SCA).

foundation as a service (FaaS): A legal incorporated entity that provides business and operational infrastructure for groups who wish to form a foundation for advancing open source code but are not able to handle all the complex day-to-day tasks themselves.

Free Software Foundation Europe (<https://fsfe.org/>): A German charity founded in 2001 that advocates for Free Software in Europe.

Git (<https://git-scm.com/>): A free and open source distributed version control system often used for software development. Created by Linus Torvalds in 2005 for Linux version control, it is used in many forge sites today.

Google Ads Keyword Planner (<https://business.google.com/en-all/ad-tools/keyword-planner/>): A free tool that provides statistics on keyword search volume within Google.

Harvard Dataverse Repository (<https://dataverse.harvard.edu/>): A free and open source data repository open to all researchers in and outside the Harvard community. Developed at Harvard and originally released in 2006.

HELIOS Open (Higher Education Leadership Initiative for Open Scholarship) (<https://www.heliosopen.org/>): An initiative from the National Academies of Sciences, Engineering, and Medicine to create a cohort of higher education institutions who commit to advance open scholarship on their campuses.

Hugging Face (<https://huggingface.co/>): A US company whose Hub is an online collaboration platform where users share open source models and datasets.

I-Corps, or Innovation Corps

(<https://www.nsf.gov/funding/initiatives/i-corps>): A training program created by the US National Science Foundation to help scientists and engineers extend their projects towards commercialization.

IEEE (Institute of Electrical and Electronics Engineers)

(<https://www.ieee.org/>): A nonprofit professional organization for technologists.

Infra Finder (<https://infracfinder.investinopen.org>): A tool created by Invest in Open Infrastructure to catalog existing infrastructure services for open research and scholarship.

Internet2 (<https://internet2.edu/>): A nonprofit organization that provides a secure high-speed network to higher education and research institutions.

Invest in Open Infrastructure (<https://investinopen.org/>): A nonprofit focused on open source tools that enable the creation and dissemination of open content and data.

It Takes A Village Toolkit (<https://itav.lyrasis.org/toolkit/>): A toolkit developed by Lyrasis to help libraries, archives, and museums plan and manage sustainability for open-source software related to cultural heritage.

Journal of Open Source Software (<https://joss.theoj.org/>): A peer-reviewed open access journal which publishes scholarship on open source software.

Jupyter (<https://jupyter.org/>): An open source software environment first released in 2014, often used in cloud computing.

Kuali (<https://www.kuali.org/>): An open source software as a service (SaaS) first released in 2004 which is used for university finance, research administration, curriculum, and other functions.

Linux (<https://www.linux.org/>): An open source operating system first released in 1991. It is currently used on most internet servers and is the basis of Android smartphones.

Lyrasis (<https://lyrasis.org/>): A nonprofit that supports open access to cultural heritage through technology partnerships with libraries, archives, and museums.

Map of Open Source Science (MOSS)

(<https://www.opensource.science/moss>): A comprehensive repository for visualizing the “tools, people, papers, and data vital to scientific research.” MOSS is a project of the Open Source Science Initiative (OSSci) created by NumFOCUS, a nonprofit that serves as a fiscal sponsor for open source research software.

Masscan (<https://github.com/robertdavidgraham/masscan>): An open source port scanner.

memorandum of understanding (MOU): A document that outlines the terms of an agreement between two organizations.

Moodle (<https://moodle.org/>): An open source learning management system (LMS) first released in 2002.

National Student Data Corps (<https://nebigdatahub.org/nsdc/>): A community-developed data science training program for students, run by four regional nonprofit Big Data Innovation Hubs which are funded by the US National Science Foundation.

network effect: In economics, a phenomenon where the value a user receives from a good or service increases when there are more users of the good or service.

Nmap (<https://nmap.org/>): An open source network mapper used to identify open ports.

Open edX (<https://openedx.org/>): An open source learning management system (LMS) first released in 2012.

Open Research Community Accelerator (ORCA)

(<https://www.orcaopen.org/>): A project to advance open science, created by the nonprofit Multiplier.

Open Science agenda: A worldwide movement (also called ‘open research’ or ‘open scholarship’) which gained momentum in the 21st century, aiming to make the information and benefits of scholarly research available to the general public. The agenda is central to the United Nations Open Science and Open Scholarship Conference, held since 2018.

Open Science Framework (<https://osf.io/>): An open source project management tool for research, created by the nonprofit Center for Open Science.

OpenAlex (<https://openalex.org/>): An open access catalog of scientific scholarship created by the nonprofit OurResearch in 2022. It is intended to compete with commercial databases used to calculate journal impact factor.

OpenCampus (<https://www.opencampus.net/>): An open source student information system run by a software company of the same name that allows higher ed institutions to design their own administrative workflows without coding.

Opencast (<https://opencast.org/>): An open source video management system for academic institutions, optimized for recording and distributing lectures. Created by the University of California, Berkeley and currently supported by the Apereo Foundation.

OpenForum Europe (<https://openforumeurope.org/>): A Belgium-based nonprofit think tank that promotes openness in Europe.

Open Source Congress (<https://www.opensourcecongress.org/>): An annual international convening of leaders of OSS foundations. The first OSC was held in 2023.

OSPO as a service: A model where institutions pay a membership fee to a foundation, which acts as an OSPO for the institution.

Patreon (<https://www.patreon.com/>): A crowdfunding platform where users subscribe to receive access to digital products from a specific content creator.

persona: In user experience design, a persona is a personified representation of a specific customer segment or category of user.

port scan: A process of sending electronic packets to a range of network port addresses, with the goal of identifying ports which are actively being used to send and receive data.

Principles of Open Scholarly Infrastructure

(<https://openscholarlyinfrastructure.org/>): A set of guidelines for running and sustaining open scholarly infrastructure organizations in the research community, codified in 2020. Organizations self-select to adopt the POSI guidelines.

product commercialization model: A multi-stage process of bringing a new product or service to market, including such aspects as market research, product design and development, manufacturing, marketing, sales, etc.

productization: A process to standardize across the production process so that each project does not have to be created from scratch.

proxy server: A server that acts as an intermediary between a client making a request and a server that fulfills the request.

Public Money, Public Code (<https://publiccode.eu/en/>): A campaign by the Free Software Foundation Europe which argues for legislation requiring that publicly financed software developed for the public sector be made publicly available with an open source license.

pull request: A proposed change to a code repository; the contributor requests that the project maintainer pull their proposed change and merge it into the code base.

Python (<https://www.python.org/>): An open source programming language, first released in 1991.

R (<https://www.r-project.org/>): An open source programming language for statistical computing and data visualization, first released in 1993.

Repo Review Consultations

(<https://researchcomputing.princeton.edu/services/repo-review->

consultations ([https://www.princeton.edu/research-computing/consultations/](#)): A code review service for Princeton University researchers, staffed by Research Software Engineers (RSEs) from Research Computing.

reverse proxy server: Intermediate connection point that stands between the open internet and a private web server (where data and applications are stored). The reverse proxy receives and routes external requests from clients, directing traffic and improving the private web server's security, performance, and reliability.

Sakai (<https://www.sakailms.org/>): An open source learning management system (LMS) first released in 2005.

Scalar (<https://scalar.me/anvc/scalar/>): An open source publishing platform for digital scholarship, supported by the University of Southern California Libraries.

Scarf (<https://about.scarf.sh/>): Software application that provides usage analytics for open source software.

software as a service (SaaS): Rather than installing software locally, users subscribe to a cloud-based application that delivers the software over the internet.

software bill of materials (SBOM): A list of components / dependencies used to build a software product.

software composition analysis (SCA): A process for identifying all open source components of a software application (generating a software bill of materials) and then scanning each component for updates, compliance, and potential risks.

Software Heritage (<https://www.softwareheritage.org/>): A nonprofit whose mission is to collect, preserve, and share publicly available software.

Sovereign Tech Agency (<https://www.sovereign.tech/>): A German government agency established in 2022 to support open source software.

Stack Overflow (<https://stackoverflow.com/>): A web platform for programmers to ask questions and help each other solve problems.

tech debt, or technical debt: The idea that choosing a solution that is fast and low-quality creates future costs that have not yet been recognized.

total cost of ownership (TCO): An estimate of all the direct and indirect costs, both present and future, involved in the decision to purchase a product or service.

tracking pixel: A tiny, transparent image embedded in the HTML code of a webpage which contains a link to a server. When the user loads the page, their browser requests the image from the server, which then receives information about the user's IP address, device, etc. Commonly used in web marketing.

United States Research Software Engineer Association (US-RSE) (<https://us-rse.org/>): A fiscally-sponsored project of the nonprofit Community Initiatives (<https://communityinitiatives.org/>), US-RSE is a free membership association for all people who identify as Research Software Engineers which advocates for the importance of RSEs.

University of California Open Source Program Office Network (<https://ucospo.net/>): An initiative of the Alfred P. Sloan Foundation to create one OSPO which serves six institutions in the University of California system.

Unicornscan (<https://www.kali.org/tools/unicornscan/>): An open source port scanner.

user experience (UX) design: A subfield of design which centers ease and intuitiveness of use as its fundamental principle and then employs evidence about user experience to make design decisions.

version control system: A process where changes to files are tracked and documented over time.

WordPress (<https://wordpress.org/>): An open source web content management system (CMS) originally released in 2003 which is used on the back end of many high-traffic websites today.

XWiki (<https://www.xwiki.org/>): An open source platform that allows users to build a wiki, a website that is collaboratively edited through a browser.

Zenmap: See Nmap

Zenodo / GitHub integration

(<https://help.zenodo.org/docs/github/>): Automated system for creating a DOI for each new release published on GitHub.

Appendix 1

About the workshop organizers

Ithaka S+R helps academic and cultural communities serve the public good and navigate economic, technological, and demographic change. Ithaka S+R has over two decades of experience organizing large scale research projects on issues relating to scholarly communication, researchers' research and information practices, and other elements of the research enterprise. In that time, Ithaka S+R has worked with over 140 colleges and universities to create and disseminate data on topics of interest to academic research and teaching communities. Ithaka S+R staff have extensive experience designing productive convenings with diverse stakeholders on strategic issues in scientific research. Ithaka S+R recently published a report, funded by The Alfred P. Sloan Foundation, which assessed the effectiveness of university OSPOs in facilitating the development and maintenance of open source software. Ithaka S+R is a service of ITHAKA, a nonprofit with a mission to improve access to knowledge and education for people around the world.

The Apereo Foundation is a global nonprofit consortium that advances open source software to support teaching, learning, and research in higher education. Formed in 2012 through the merger of the Sakai and Jasig foundations, bringing together more than 25 years of combined experience in open source educational technology, Apereo provides governance, community coordination, and incubation support for a portfolio of open source projects.

Through open governance and community-driven innovation, the Foundation fosters a collaborative ecosystem grounded in shared stewardship of the academic digital commons. With deep expertise in open source sustainability and cross-sector engagement, Apereo enables

institutions worldwide to retain greater control over their digital environments while advancing interoperability, resilience, and the public good.

Ithaka S+R and Apereo staff comprised the core organizing committee for the workshop. The organizing committee was responsible for designing the workshop, creating the call for participants, evaluating applications and selecting participants, facilitating the workshop, and writing the report. Dr. Chelsea McCracken, Dr. Dylan Ruediger, and Dr. Mark McBride represented Ithaka S+R. Patrick Masson and Josh Baron represented the Apereo Foundation. The organizers also wish to recognize Deb Longino, Jenn Cummings, Fua Pan, and Jennifer Schopf for their assistance.

Highlights of recent work

Ithaka S+R's recent work on open science

Recent reports

- University Open Source Program Offices (2025)
- Researcher Challenges and Experiences with Data Services (2025)
- The Research Data Services Landscape at US and Canadian Higher Education Institutions (2024)
- Data Sharing Practices in the Humanities (2023)
- Leveraging Data Communities to Advance Open Science (2022)

The Apereo Foundation's recent work on open source in higher education

Reports

- The Open Source Software in Higher Education Preliminary Report (2025)

Community Engagement

- Openness Constituency Group co-leader, EDUCAUSE (2023, 2024, 2025)
- Track organizer, "Open source in Higher Education, OW2 Conference, Paris France (2024, 2025)
- Track organizer, "Open source in Higher Education, FOSSY, Portland, OR, USA (2025)
- Track organizer, "Open source in Higher Education, ScaLE, Los Angeles CA, USA (2026)

- Track organizer, “Open source in Higher Education, Eclipse OCX, Brussels, Belgium (2026)

Appendix 2

SOSSRE was conceptualized as a one-day workshop that would take place in-person in ITHAKA's New York City office, which has the capacity to seat about 40 people. Since one of the goals of the workshop was to bring together people from OSS communities who do not often have the chance to interact, the organizing committee invited approximately half the participants, deliberately selecting representatives of groups with different perspectives (e.g., OSS industry leaders, higher ed administrators, Open Source Program Office (OSPO) staff, research software engineers, etc.). In the spirit of openness, approximately a quarter of the participant slots were left open to be filled by applicants to the CFP (see text below). The remaining slots were filled by the organizing committee. In total, 39 participants attended the workshop (see participant list below).

Sustaining Open Source Software in the Research Enterprise: Call for participants

Ithaka S+R and the [Aperero Foundation](#) welcome applications to participate in *Sustaining Open Source Software in the Research Enterprise*, a workshop made possible by generous funding from the National Science Foundation, Alfred P. Sloan Foundation, and the Chan Zuckerberg Initiative.

Higher education serves as a seedbed for highly successful Open Source Software (OSS). Apache, Linux, R, and Python are all examples of currently widely-used, impactful, and sustained open source projects with roots in higher education. Colleges and universities have also implemented OSS to a significant degree in support of campus infrastructure, teaching and learning, and research computing. OSS serves a variety of functions in the academy, from addressing resource constraints to aligning with other open initiatives such as Open Science/Research, Open Educational Resources (OER), and others.

Nevertheless, sustainability is a major challenge for even the most successful open source software that, at a minimum, requires ongoing engagement to improve and maintain code and communities. OSS sustainability also includes the identification of a viable financial model,

project governance, technology infrastructure, business operations, and navigating legal issues such as copyright and licensing. Dispersed across distinct, specialized, research communities, OSS for research in particular has yet to secure a firm institutional footprint, and is often a low priority of institutions, publishers, and funders.

Successful applicants will receive funding to participate in a one-day, in-person workshop, bringing together OSS leaders from across higher education, open source foundations, and the private sector to discuss sustainability strategies and identify governance and infrastructure opportunities that spur broad and diverse adoption and sustainability of OSS. Participants will identify practical opportunities to develop holistic institutional pathways to better harness the power of OSS for open science and technological innovation by improving the resiliency and sustainability of the OSS ecosystem at their home institutions and across higher education.

Who is eligible to apply?

We welcome applications from practitioners with experience developing or contributing to OSS in higher education settings, especially those who develop OSS for research. We also encourage individuals affiliated with other stakeholder communities, such as the non-profit or private sector, to apply.

Applicants should have experience working on OSS projects, particularly those associated with higher education, and/or research experience that includes leveraging OSS. At this time we are not able to fund participants from outside the US.

When is the workshop?

The one-day, in-person workshop will be held at Ithaka S+R's offices in New York City on Friday, August 8, 2025 from approximately 8am to 5pm. Applicants are requested to be onsite during the entire duration of the workshop. There will be an optional reception on Thursday, August 7 at 4:30pm.

What support will Ithaka S+R and the Apereo Foundation provide?

The organizing committee will provide funding for travel and accommodations for workshop participants. Reimbursement for reasonable child care costs is available for participants.

When are applications due?

Applications are due on May 9, 2025, at 11:59pm Eastern Time.

What do I need to do to apply?

To apply, submit the following information to Chelsea.McCracken@ithaka.org via a PDF, DOC, or ODF/ODT document:

1. Your name, affiliation, and contact information.
2. A CV, resume, or biographical sketch.
3. A short statement of 250-500 words indicating your interest in the workshop topic and the perspectives you can contribute to the discussion.

If you have questions about this project or the application process, please email Chelsea McCracken at Chelsea.McCracken@ithaka.org.

Participant list

Members of the organizing committee and funders are marked with *.
Plenary speakers are marked with †.

Name	Title	Organization
Nathan Alexander	Faculty	Howard University
Cat Allman [†]	VP, Open Source	Digital Science
Chris Aniszczyk	CTO	Linux Foundation (CNCF)
Josh Baron*	Development Officer	Apereo Foundation
Katherine Brown	Sr Res Fellow	The University of Texas at Austin / University of Cambridge, UK
Scott Collard	Associate Dean for Research & Research Services	New York University Libraries
Karmen Čondić-Jurkić [†]	Executive Director	Open Molecular Software Foundation
Clare Dillon [†]	Community Lead	CURIOSS
Karl Fogel	Partner	Open Tech Strategies, LLC
Megan Forbes	Program Manager, Open Source Programs Office	Johns Hopkins University
Andrew Fullard	Research Consultant	Michigan State University
Kate Hertweck*	Program Manager, Open Science	Chan Zuckerberg Initiative
Josh Greenberg*	Program Director	Alfred P. Sloan Foundation
Muhammad Hossain	Director of Instructional Technology	Claflin University
Samir Iqbal*	Program Director	National Science Foundation
Ashwathi Iyer	Associate Director of Software Licensing and Research Partnerships	University of Michigan

Name	Title	Organization
Daniel S. Katz	Chief Scientist, NCSA	University of Illinois Urbana-Champaign
Stephanie Lieggi	Executive Director, CROSS	UC Santa Cruz
Georg Link	Open Source Strategist	Bitergia
Zi-Kui Liu	Dorothy Pate Enright Professor	Pennsylvania State University
Suresh Marru	Research Faculty	Georgia Tech
Patrick Masson*	Executive Director	Apereo Foundation
Mark McBride*	Director	Ithaka S+R
Chelsea McCracken*	Researcher	Ithaka S+R
John Meluso	Ezra Systems Postdoctoral Associate	Cornell University
Rosalyn Metz	CTO / Consultant	Emory Libraries and Museum
David Millman	Assoc Dean for Technology	NYU Libraries
Devora Najjar*	Program Associate, Technology	Alfred P. Sloan Foundation
Angela Newell	Executive Director of Innovation, Director of the UT-OSPO	The University of Texas at Austin
Duane Nykamp	Associate Professor	University of Minnesota
Jason Paluck	Associate Vice President	UMBC
Allison Randal [†]	Senior Researcher	Capabilities Limited
Kimberly Ray	Research Assistant Professor	University of Texas Austin
Dylan Ruediger*	Principal, The Research Enterprise	Ithaka S+R
Nathan Swain	Director	Tethys Geoscience Foundation / Aquaveo

Name	Title	Organization
Hector N. Torres	Professor of Mathematics and Science Education	Bethune-Cookman University, Daytona Beach, FL
Elizabeth Vu*	Senior Program Associate	Alfred P. Sloan Foundation
Gregory Watson	Research Scientist	Oak Ridge National Laboratory
Edward Zarecor	VP of Engineering for Open edX	Axim Collaborative

Appendix 3

Workshop agenda

8:30-9:30 - Networking Breakfast

9:30-9:50 - Welcome

Kevin Guthrie, President, ITHAKA

Mark McBride, Director, Academic Mission, Ithaka S+R

Patrick Masson, Executive Director, The Apereo Foundation

Chelsea McCracken, Researcher, Ithaka S+R

9:50-10:50 - Plenary panel: OSS Sustainability in Context

Moderated roundtable discussion. Panelists will describe what sustainability means to them, how they measure it, and what works to create sustainability in their sector.

- Patrick Masson, Executive Director, The Apereo Foundation (moderator)
- Cat Allman, VP, Open Source, Digital Science
- Karmen Condic-Jurkic, Executive Director, Open Molecular Software Foundation
- Clare Dillon, Community Lead, CURISS
- Allison Randal, Senior Researcher, Capabilities Limited

10:50-12:00 - Breakout 1: Challenges of Sustainability in Research OSS

Preassigned breakout groups with moderator. Participants will discuss sustainability factors relevant to their experience and define the challenges specific to OSS for research.

12:00-1:15 - Lunch

1:15-1:30 - Regroup

1:30-2:10 - Breakout 2: Identifying Solutions to Sustainability in Research OSS

Self-selected breakout groups with moderator. Given one sustainability challenge per group, participants will generate a solution using Think-Pair-Share.

2:10-2:45 - Break

2:45-4:00 - Breakout 3: Lightning Round Solutions to Sustainability in Research OSS

Self-selected breakout groups with moderator. Participants will quickly generate solutions to other sustainability challenges.

4:00-4:30 - Closeout

Moderators will summarize the sustainability solutions generated by the group, followed by a group discussion of feasibility.



Practical Guide for Sustaining Open Source Software in the Research Enterprise

A Business Process Workflow and Decision
Framework for Universities, Research Teams,
and OSS Communities

*Grounded in Findings from the Sustaining Open Source
Software in the Research Enterprise Workshop*

May 27, 2026



Purpose of the guide

Open Source Research Software (OSRS) is now central to discovery, collaboration, reproducibility, and interdisciplinary innovation across the research enterprise. Yet OSRS remains fragile: maintenance is undervalued, incentives misaligned, discoverability inconsistent, and institutional responsibility unclear. Many projects disappear within their first few years due to a lack of sustainability planning.

This practical guide provides:

- **A workflow** for researchers and institutions to identify sustainability needs and select appropriate models.
- **A logic model/decision tree** that visually maps sustainability pathways.
- **Concrete actions** for researchers, administrators, OSRS developers and communities, Open Source Program Offices (OSPOs), Tech Transfer Offices (TTOs), libraries, consortia, and funders.
- **Embedded insights, examples, and quotations** from the workshop's findings.

Summary of key challenges the guide addresses:

Five challenges were identified in the workshop:

1. What is the value proposition for universities? Why should they prioritize OSRS?
2. How can we know who is or could be using an OSRS product?
3. How can we overcome researchers' cultural barriers to open culture?
4. Who should be responsible for sustaining OSRS?
5. What extra-institutional support is needed to sustain OSRS?

This guide offers a decision process and sustainability models that directly respond to each challenge.

Core sustainability dimensions

Workshop participants surfaced four interdependent dimensions that together define OSRS sustainability:

1. Motivational Sustainability

Sustainability begins with **the goals of the project and the goals of the people** involved.

- Sustainability only has meaning in the context of the software's goal.
- Many OSRS tools are "purpose-built Q-tips"—not meant to be sustained beyond the immediate scientific need. Software should not be sustained simply for its own sake.
- Motivation varies: fun, profit, learning, academic rewards, or real-world impact. Each carries different sustainability risks.

2. Infrastructural Sustainability

OSRS is **core research infrastructure**—as essential as labs, buildings, and utilities.

- Many researchers use OSS without ever being aware that someone needs to maintain it.
- Infrastructure requires continuous, reliable funding streams—not episodic grants.

3. Archival Sustainability

Even if code is not actively maintained, it must be **citable, discoverable, and preserved**.

- A citation is required so that software can fit into existing systems for academic incentives. Also, according to the principles of open science, preservation of the code that produced certain findings is similarly important to preservation of the data.
- Discoverability is critical for the reproducibility of science.

- Preservation and discoverability enable future researchers to identify relevant/useful code sources that can be adopted and modified for new research endeavors.
- Preservation in the scholarly record includes documentation, provenance, licensing, and archiving in an accessible location.

4. Relational Sustainability

OSRS communities must be **governed, connected, and supported**.

- People are the most essential feature for ensuring project sustainability.
- Theoretically, OSRS could be sustained as a by-product of other institutional work (ongoing research, as a teaching and learning methodology, upstream/downstream of administrative computing).
- Communication, onboarding, mentoring, shared governance, and attention to culture are all central.

Business process workflow: Practical guide for sustaining OSRS

STEP 1 – Define purpose and intended impact of the software

Key Question: *What is this software trying to be?*

Actions:

1. Identify the archetype of the project (e.g., research tool, platform, library, infrastructure).
 - a. One resource identified by SOSSRE workshop participants to assess/align purpose and projects was “Open Source Archetypes: A Framework for Purposeful Open Source (version 2.0),” Open Tech Strategies and Mozilla, 2019, <https://opentechstrategies.com/archetypes-files/open-source-archetypes-v2.pdf>.
2. Assess expected user community size: single-use, niche, discipline-wide, or cross-disciplinary. While formal tools specific to market interest may not exist, several “desktop research” efforts using industry-recognized resources may help assess potential applicability and interest.
 - a. Stack Overflow Trends and Developer Surveys
 - i. Stack Overflow Developer Survey (<https://survey.stackoverflow.co/>)
 - ii. Stack Exchange Data Explorer (<https://data.stackexchange.com/>)
 - iii. Search tags related to your idea (e.g., rust, bioinformatics, GraphQL); assessing:
 1. Question volume over time
 2. Unanswered questions (indicates unmet need)
 - iv. Signal strength:
 1. High volume = large ecosystem
 2. Low but consistent = niche

3. Very low = likely single-use, early, or experimental
- b. [GitHub Inspiration and Metrics](#)
 - i. Github Innovation Graph (<https://innovationgraph.github.com/>)
 - ii. Github Topics (search directly, e.g., github.com/topics/<your-topic>)
 - iii. Consider stars, forks, contributors of similar projects, and growth trends (stars over time).
 - iv. Signal strength:
 1. 10k+ stars across multiple projects = large ecosystem
 2. 1k - 10k = niche
 3. <100 = likely single-use, early, or experimental
- c. Google Search and Trends
 - i. Advanced Search (https://www.google.com/advanced_search). Or search directly, e.g., “<your topic 1>” “<your topic 2>” site:.edu
 - ii. Trends (<https://trends.google.com/trends/>)
 - iii. Identifies related projects, communities, and organizations.
 - iv. Measures search interest over time
 - v. Signal strength:
 1. Multiple projects/communities, active through time = large ecosystem
 2. Some projects, active through time = niche
 3. Some projects, inconsistently active through time = likely single-use, early, or experimental
3. Determine desired lifespan: short-term (one project cycle), medium-term, long-term infrastructure.
4. Clarify motivation: scientific impact, novelty, funding, market viability, student training, etc.

Decision Point → Select a Sustainability Path and go to STEP 2

- **Archive-only Path** (Q-tip software)
- **Stewardship Path** (moderate maintenance for transparency and reproducibility)
- **Community Path** (active multi-user ecosystem)
- **Infrastructure Path** (mission-critical or widely adopted)

STEP 2 – Assess human capacity and governance needs

Key Question: *Who supports this software?*

Actions:

1. Identify potential/current core maintainers and contributors, their roles (e.g., faculty, students, postdocs, RSEs, external developers), and affiliations (e.g., campus, institution, company).
 - a. Data from STEP 1 can help identify peers, colleagues, and other relevant potential contributors.
 - b. Signal strength:
 - i. Multiple local/affiliated contributors = strong support environment
 - ii. Some local, potential affiliated contributors = limited support environment
 - iii. Few local, unknown affiliated contributors = low/no support environment
2. Identify resources (infrastructure, funding, non-technical collaborators, existing code/projects/communities, intellectual property, aligned non-technical organizations, etc.)
 - a. Again, STEP 1 may provide relevant leads for identifying non-staffing resources.
 - b. Signal strength:
 - i. Multiple local/affiliated resources = strong support environment
 - ii. Some local, potential affiliated resources = limited support environment
 - iii. Few local, unknown affiliated resources = poor support environment
3. Evaluate turnover risks, “continuous cycle of new people” (e.g., students) in academia.
 - a. Assess the roles of your potential community contributors and collaborators:
 - i. Signal strength:
 1. Primarily long-term faculty and staff (local and affiliated FTE) = strong support environment
 2. Few long-term faculty and staff (local and affiliated FTE), primarily students (local and affiliated, enrolled or grant-funded) and fixed-

- term faculty and staff = limited support environment
3. Primarily students (local and affiliated, enrolled or grant-funded) = poor support environment
4. Create or adopt governance models:
 - a. Decision-making process, see:
 - i. Fogel, Karl. "Social and Political Infrastructure." In *Producing Open Source Software: How to Run a Successful Free Software Project* (Second Edition). 26 September 2024, <https://producingoss.com/en/producingoss.html#social-infrastructure>.
 - ii. Bacon, Jono. "Governance." In *The Art of Community* (Second Edition). O'Reilly, 2012, <https://www.jonobacon.com/wp-content/uploads/2019/01/jonobacon-art-of-community-second-edition.pdf>.
 - iii. Neary, Dave, Josh Berkus, and Bryan Behrenshausen. "Project and Community Governance." In *The Open Source Way*. Edited by Brian Proffitt, et al. Red Hat, 2025, <https://producingoss.com/en/producingoss.html#social-infrastructure>.
 - b. Succession planning
 - i. "Practitioner Guide: Getting Started with Contributor Sustainability" CHAOSS. Linux Foundation, 2026, <https://chaoss.community/practitioner-guide-contributor-sustainability/>.
 - c. Codes of conduct
 - i. Fogel, Karl. "Codes of Conduct." In *Producing Open Source Software: How to Run a Successful Free Software Project* (Second Edition). 26 September 2024, <https://producingoss.com/en/producingoss.html#code-of-conduct>.
 - ii. Maffeo, Lauren. "Step Three: Create and Enforce a Clear Code of Conduct." In *The Open Source Way*. Edited by, Brian Proffitt, et al. Red Hat, 2025, <https://guidebook.theopensourceway.org/attracting-users/building-diverse-open-source-communities-by->

Decision Point → **Is the team stable enough to maintain the project?**

- If **Yes** → proceed to Step 3.
- If **No (Undetermined)** → seek support:
 - Campus resources: local OSPO (if one exists), campus library, Tech Transfer Office, etc.
 - Open source software foundations related to your project's scope, core technology(ies), academic domain, etc.: CURIOSS (<https://curiooss.org/>), SustainOSS: Open Source in Academia Map (<https://sustainoss.org/academic-map/intro.html>), FOSS Foundations Directory (<https://fossfoundation.info/listing>), Google Supported Organizations (<https://opensource.google/organizations-we-support>).
 - External community incubator or consultants

STEP 3 – Document, license, cite, and make discoverable

Key Question: *Are your practices mature, can new contributors engage?*

Actions:

1. Ensure code is findable:
 - a. Package Registries: Publishing in these and similar ensures projects show up in developer workflows (search, installs, dependency graphs). Examples include Npm (<https://www.npmjs.com/>), PyPI (<https://pypi.org/>), RubyGems (<https://rubygems.org/>), crates.io (<http://crates.io>), Docker Hub (<https://hub.docker.com/>).
 - b. Code repositories: Publicly accessible code repositories provide environments to expose code, issues, and roadmaps to peers in technology and/or disciplines: Examples include ASF Bitbox, Bitbucket, Codeberg, Forgejo, Gitea, GitHub, Gitlab, invent.kde.org, Launchpad, Savannah, SourceForge, etc.

- c. Your choice should align with your vision for the project; different project priorities (e.g., maximum visibility, enterprise workflows, open source values/nonprofit, self-hosting, decentralization, legacy/distribution, and advanced dev workflows) align with different code repositories.
2. Ensure code is citable:
 - a. Publish software in venues that recognize code as scholarship (in addition to discipline-specific publications). For example:
 - i. Journal of Open Source Software
 - ii. SoftwareX
 - iii. Journal of Open Research Software
 - b. Practices
 - i. Include a CITATION.cff file in your repository.
 - ii. Assign a DOI to releases (Zenodo, Figshare)
 - iii. Adopt clear versioning such as v1.2.0
 - iv. Add a “How to Cite” section in the project’s README.md
 - v. License clearly and appropriately: OSI Approved Open Source License (code), Creative Commons Licenses (non-code, text, graphics)
3. Document purpose, governance, and usage.
 - a. Fogel, Karl. "Documentation." In *Producing Open Source Software: How to Run a Successful Free Software Project* (Second Edition). 26 September 2024, <https://producingoss.com/en/producingoss.html#documentation>.
 - b. Hawthorne, Leslie. “Step Three: Communication Norms in Open Source Projects.” In *The Open Source Way*. Edited by Paula Dickerson, et al. Red Hat, 2025, <https://guidebook.theopensourceway.org/attracting-users/communication-norms-in-open-source-software-projects>.
4. Select an appropriate open source license. Due diligence dictates that founders of projects initiated through/with an institutional relationship consult the appropriate offices, for example, the campus’s Office of the General Counsel and/or the Technology Transfer Office/IP Transfer Office. In order to prepare for such discussions, a review of potential open source software licenses would be helpful.

- a. Open Source Initiative's Approved License List (<https://opensource.org/licenses>)
 - b. Creative Commons, Choose a License for your work (<https://creativecommons.org/chooser/>)
 - c. Carnegie Mellon University, Open Source License Comparison Grid (<https://www.cmu.edu/cttec/forms/opensourcelicensegridv1.pdf>)
 - d. Choose a License (<https://choosealicense.com/>)
 - e. Github License Selector (<https://ufal.github.io/public-license-selector/>)
5. Deposit code in a discoverable, persistent archive.
- a. Software Heritage (<https://www.softwareheritage.org/>)
referenced by workshop attendees
 - b. OpenAIRE (<https://www.openaire.eu/>)
 - c. Re3data (<https://www.re3data.org/>)

Decision Point → Does the project require active maintenance?

- If **Yes** → proceed to Step 4.
- If **No** → Follow Archival Sustainability Path.

STEP 4 — Evaluate funding pathways

Key Question: *Does the project need funding, and where might funding be obtained?*

Actions:

1. Identify required maintenance levels (minimal, moderate, high). Several tools are available to assess a new project's demands based on existing examples/references from peer projects.
 - CHAOSS
 - i. Metrics and Metrics Models (<https://chaoss.community/kb-metrics-and-metrics-models/>)

- ii. Software, eg., Augur, GrimoireLab
(<https://chaoss.community/software/>)
 - Multiple commercial options
(<https://github.com/topics/analytics>)
- 2. Determine if the project is a “by-product” of other funded operations.
 - When an open source software project emerges as a by-product of broader, funded university operations, it often does not require dedicated funding of its own. Instead, the resources needed to sustain and evolve the project—such as developer time, infrastructure, and maintenance—are naturally supported through existing budgets tied to administrative computing, academic programs, student services, and research initiatives. In this model, the software benefits from ongoing institutional investment in these areas, allowing it to grow organically without the overhead of securing separate funding streams, while still delivering value both within and beyond the university.
- 3. Determine if funding is available/possible for all aspects of the project.
 - [Aperio Foundation OSS Grant Funding Prioritization Framework](#)
 - [Aperio OSS Funding Tracker](#)
- 4. Consider alternatives to traditional grant-driven funding. “Business Models for Free and Open Source Software,” by Chitu Okoli and An Nguyen (SSRN, 2015, <https://dx.doi.org/10.2139/ssrn.2568185>) includes a variety of revenue models and, critically, their impact on both the project, developer, and end-user communities.
 - Funding OSS as infrastructure (Institutional Cost-Sharing/ Pooled Funding)
 - Service-Based Revenue (Support, Hosting, SLAs)
 - Membership & Consortium Models (Core + Tiered)
 - Corporate Partnerships & Sponsorships
 - Subscription-Based “Open Core Services” (Without Closing the Core)
 - Training, Certification, and Workforce Development
- 5. Choose from multiple organizational models:
 - Institutional (e.g., campus OSPO, library, department, research office)
 - Consortial (e.g., multi-institutional or discipline, academic or working groups)

- Foundational: (e.g., create or join a 501c3 or c6)
- Endowment/Collectives (new/emerging models as of 2026)
- Industry partnerships
- National or state agencies

Decision Point → **Is there long-term funding?**

- If **Yes** → proceed to Step 5.
- If **No** → shift to hybrid sustainability, reduce scope, or move toward archival path.

STEP 5 – Understand usage and impact

Key Question: *How will the project identify users and measure impact?*

Actions:

1. Use indirect measures where direct measurement (e.g., software and user tracking) is incompatible with OSS values:
 - Publications citing the software
 - Dependency analysis
 - Code repository metrics (e.g., downloads, stars, pull requests, issues submitted, contributors, forks, etc.)
 - Stories and use cases from the community
 - Community engagement (discussion forums, emails)
 - Participation (new committers, attendees at events/activities)
2. Adopt tools such as:
 - As suggested by participants in the SOSSRE Workshops
 - i. BuiltWith
 - ii. Scarf (privacy-preserving usage analytics)
 - Software Bill of Materials (SBOM) to map dependencies
 - Voluntary, inner, institution-wide port scans (in partnership with IT)

Decision Point → **Is there evidence of meaningful adoption?**

- If **Yes** → justify investment and move to Step 6.
- If **No** → consider repositioning, outreach, or winding down the project.

STEP 6 – Integrate project into institutional structures

Key Question: How will project sponsors/maintainers institutionalize the project and community?

Actions:

1. Determine which institutional unit(s) own stewardship responsibilities.
2. Determine which institutional unit(s) are dependent on or contribute to the OSRS.
 - Note: contributors from other institutions involved in the development and/or maintenance of the OSRS should be included.
3. Provide awareness and onboarding materials (as well as marketing and promotional) for faculty, students, staff, and administrators of current and potential adopting organizations.
4. Integrate OSRS into:
 - Research compliance workflows
 - Student training and work-study programs
 - Institutional PR and reputational strategies
5. Ensure alignment with institutional mission, i.e., research, teaching, public service.

Decision Point → Are the institution(s) or sufficient units prepared to steward OSRS?

- If **Yes** → proceed to Step 7.
- If **No** → escalate to research and or enterprise-level support.

STEP 7 – Connect to external ecosystems

Key Question: *What external relationships can be developed to increase resilience?*

Actions:

1. Identify multi-institution consortia, societies, and foundations aligned with your software domain.
2. Explore **Foundation-as-a-Service** models such as Code for Science & Society or Lyrasis.
3. Encourage participation in global OSS communities and standards groups.
4. Develop interoperability with other tools to increase value and reach.

Decision Point → Is the project positioned within a sustainable ecosystem?

- If **Yes** → Probability for long-term sustainability maximized.
- If **No** → Revisit earlier steps (funding, governance, adoption, archiving).

Logic model / Decision tree for OSRS sustainability

Key Question: Should this project be actively sustained, minimally maintained, or archived?

Decision tree structure

1. What is the project's purpose?
 - Single-use → **Archive Path**
 - Multi-user or foundational to ongoing research → go to step 2

2. Who are the maintainers?
 - Stable team with governance → go to step 3
 - High turnover or unclear roles → **Institutional Support Path**

3. Is the software discoverable, documented, and citable?
 - Yes → go to step 4
 - No → **Archival Preparation Path**

4. Is there sustained funding or institutional commitment?
 - Yes → go to step 5
 - No → **Hybrid/Minimal Maintenance Path**

5. Are usage metrics or community signals strong?
 - Yes → **Infrastructure / Community Sustainability Path**
 - No → **Reassessment Path**

6. Does the institution recognize OSRS as mission-aligned?
 - Yes → **Institutional Integration Path**
 - No → **Value Proposition Advocacy Path**

7. Are there external ecosystems to integrate with?
 - Yes → **Consortium / Foundation Path**
 - No → **Develop Partnerships Path**

Sustainability models for OSRS

A. Archival sustainability (minimum viable model)

- a. Document, license, archive.
- b. Achieves transparency and reproducibility.
- c. For Q-tip or one-time-use software.

B. Stewardship model (low/moderate maintenance)

- a. Department, lab, OSPO, library, or other unit provides light support.
- b. Works for tools with small but consistent user bases.

C. Community co-maintenance model

- a. Distributed contributions; governance shared.
- b. Best when community engagement is strong.

D. Institutionally supported infrastructure model

- a. OSPO, TTO, central IT, or library anchors sustainability.
- b. Ideal for widely used or high-impact tools.

E. Multi-institution consortium model

- a. Shared governance, funding, and evangelism.
- b. Enhances resilience and adoption.

F. Foundation-as-a-service model

- a. Outsource governance, accounting, legal, and community operations.
- b. Frees researchers to focus on science.

G. Endowment/continuous funding model

- a. Percent of grants or institutional budget supports maintenance.
- b. Provides strongest long-term stability.

Roles and responsibilities across the research enterprise

Researchers

- Define purpose and sustainability goals.
- Document, cite, archive; engage users.
- Advocate within departments.

Students (key workforce pipeline)

- Perform maintenance tasks; gain skills.
- Benefit from apprenticeship-like experiences.

Open Source Program Offices

- Provide governance, licensing, and sustainability planning.
- Serve as an institutional connector and community hub.

Libraries

- Steward archives, metadata, discovery systems.
- Provide training and preservation workflows.

Technology (IP) Transfer Offices

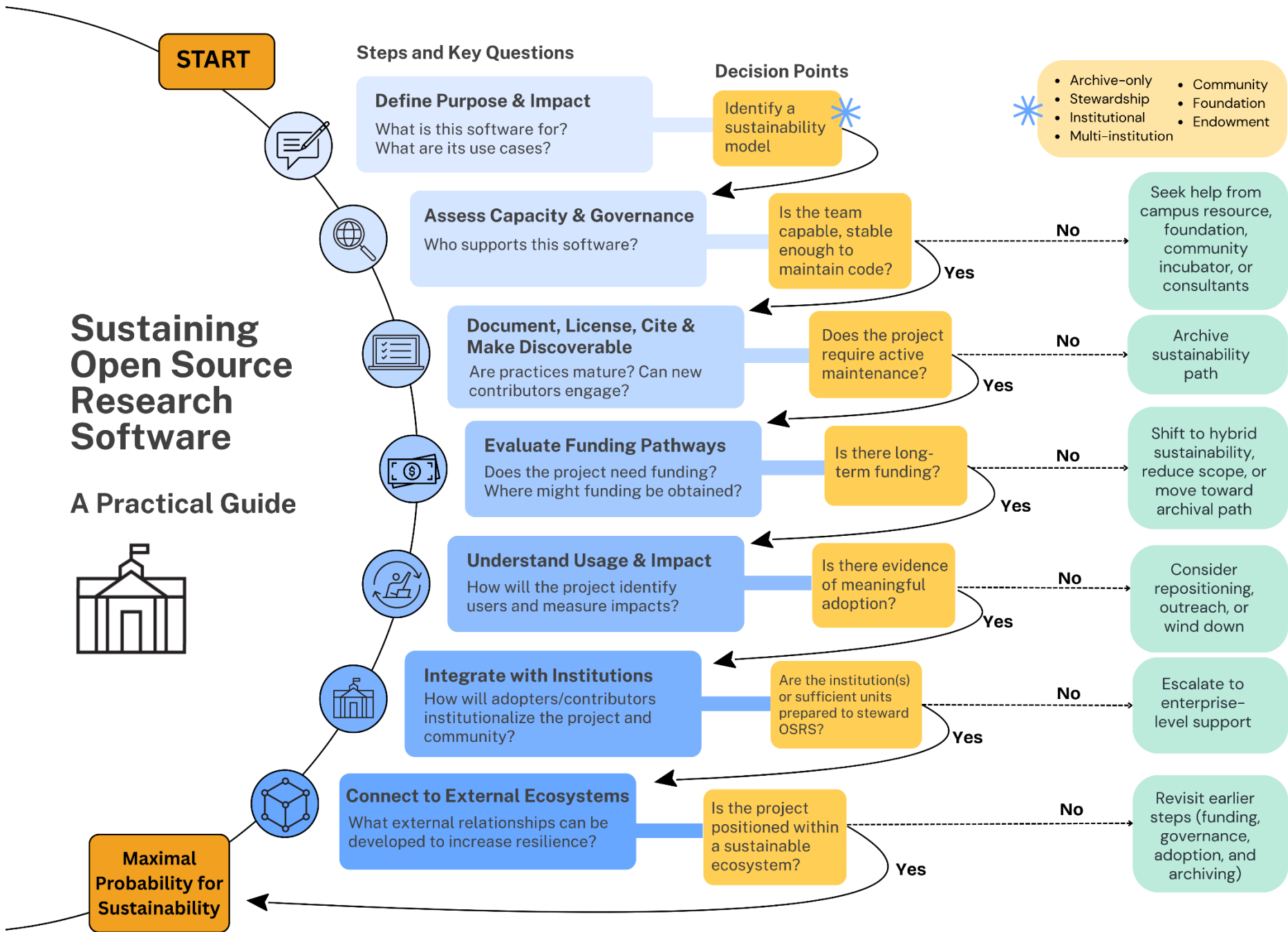
- Assist with legal, licensing, and business models.
- Adjust traditional commercialization frameworks to support OSRS.

Funders

- Require sustainability plans; support maintenance.
- Consider endowment-like approaches.

External partners (societies, foundations, NGOs)

- Provide training, consulting, matchmaking, tooling, evaluation.
- Help build sector-wide standards, e.g., “nutrition labels,” maturity models.





Sustaining Open Source Research Software

Steps and Action Items

1

Define Purpose & Impact

- Identify the archetype of the project
- Assess expected user community size
- Determine desired lifespan
- Clarify motivation

2

Assess Capacity & Governance

- Identify potential/current core maintainers and contributors
- Identify resources
- Evaluate turnover risks
- Create or adapt governance model

3

Document, License, Cite & Make Discoverable

- Ensure code is findable
- Ensure code is citable
- Document purpose, governance, and usage
- Select an appropriate open source license
- Deposit code in a discoverable archive

4

Evaluate Funding Pathways

- Identify required maintenance levels
- Determine if the project is a 'by-product' of other funded operations
- Determine if funding is available
- Consider alternatives to traditional grant-driven funding
- Choose from multiple organizational models

5

Understand Usage & Impact

- Use indirect measures where direct measurement is incommensurate with OSS values
- Identify and adopt assessment tools

6

Integrate with Institutions

- Determine which institutional unit(s) own stewardship responsibilities
- Determine which institutional unit(s) are dependent on or contribute
- Provide awareness and onboarding materials
- Integrate OSRS into research compliance workflows, student programs, PR strategies
- Ensure alignment with institutional mission

7

Connect to External Ecosystems

- Identify consortia, societies, and foundations aligned with your software domain
- Explore Foundation-as-a-Service models
- Encourage participation in global OSS communities
- Develop interoperability with other tools to increase value and reach